ROBOTIC ASSEMBLY OF INTERLOCKING CNC-CUT SHEETS INTO A WALL COMPONENT

Redesign the CNC-cut elements of the wall component in order to reduce the robotic automatic assembly time

Jeroen de Bruijn - 28 June 2019

Colophon

Master of Science (MSc) thesis
MSc Architecture, Urbanism and Building Sciences
Master's track: Building Technology
Graduation studio: New Techniques

Title
Robotic assembly of interlocking CNC-cut sheets into a wall component

Subtitle
Redesign the CNC-cut elements of the wall component in order to reduce the robotic automatic assembly time

Date
28 June 2019

Author
Student name: Jeroen de Bruijn
Student number: 4502019
Personal email: J.deBruijn1991@gmail.com

Mentors
First: Ir. Pieter Stoutjesdijk AE + T | Building Product Innovation
Second: Dr. M.Arch. Serdar Aşut AE + T | Design Informatics

Board of examiners delegate
Ir. Teake Bouma Urbanism

External contacts
Dr. Ir. Rufus Fraanje Lecturer in mechatronics at The Hague University of Applied Sciences
Jorin Seiffers Student supervisor at the HighTechCentreDelft

University
Delft University of Technology
Faculty of Architecture and the Built Environment
Address: Julianalaan 134, 2628 BL, Delft, the Netherlands

TU Delft

ROBOTIC ASSEMBLY OF INTERLOCKING CNC-CUT SHEETS INTO A WALL COMPONENT

Redesign the CNC-cut elements of the wall component in order to reduce the robotic automatic assembly time

Summary

Introduction

Robots are applied in many ways, such as: stunt doubles (Panzarino, 2018), autonomic farmers (Iron Ox, 2018), robot arms that can cook (Moley, n.d.) and a robot that makes hamburgers (Troitino, 2018). In the built environment, robots are also being used more often. Four examples are: the humanoid HRP-5 that installs drywall (AIST, 2018), the bricklaying robot SAM-100 (Construction Robotics, n.d.), the 3D concrete house printing robot used in Project Milestone (3Dprintedhouse, 2018) and the robot by Odico (n.d.) that preforms Robotic Hot-Wire Cutting to create moulds for concrete prefab elements. Robots play a significant role in Industry 4.0, which is also known as the fourth industrial revolution (Dhanani, 2015). Some believe that the application of robots on construction sites is a necessity due to a shortage of skilled labourers (Bremmer, 2018).

Research framework

TheNewMakers (n.d.), or TNM, want to radically change the way buildings are being constructed. In order to accomplish this, TNM is developing LEGO for buildings, which is their prefabricated construction system. The system contains a database of different building components. All components in the database are circular and digitally produced. Each component consists of several CNC milled elements, which are milled out wooden sheets. The elements are assembled into a component by hand. The prefabricated components are transported to the construction site where the installation can be completed by a maximum of two people.

Figure 1 presents a conceptual overview of the research design. The aim of this research is to deploy a robot to automatically assemble the CNC milled elements into a wall component and it should be faster than the current human process. This leads to the following main research question:

How could the elements of a wall component of TheNewMakers be redesigned for automatic assembly while taking less time to assemble than the manual process?

Literature review

A literature review is conducted in order to get an idea of the requirements for a robotic process and a few available systems are presented. The requirements for a robotic process are (Bouchard, 2017): a robot arm, a visions system, the end effector, software (such as artificial intelligence) and other hardware. Safety is an important aspect that will be discussed as well. An overview is presented of several available industrial robots and cobots, or collaborative robots, including their characteristics. Then a detailed comparison between the industrial robots and cobots is made based on six different criteria, being: maximum payload, maximum reach, maximum velocity, mass, repeatability and estimated Total Cost of Ownership (TCO). Furthermore, a conceptual level comparison between different robotic assembly systems is presented. At the end of the literature review the Design For Automatic Assembly method, or DFA2 method, is described (Eskilander, 2001). It provides redesign guidelines, which will help to become aware of all different factors that influence the time required for an automatic assembly process.

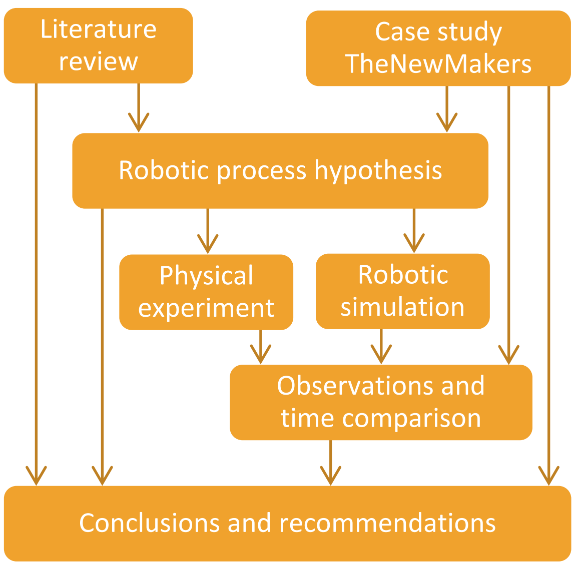


Figure 1: Conceptual overview of the research design.

Research by design

After the literature review, TheNewMakers and their building construction system is explained in further detail. As well as the construction system of Gilles Retsin (2017), whom also aims to radically change the way buildings are being constructed. Followed by a case study that examines a building block of TNM, which is a wall component that is applied in the façade of a tiny house. This wall component and the literature review lead to a robotic process hypothesis that provides input for:

- Physical experiment, where a small robot assembles a section of the wall component. The experiment is conducted in collaboration with HBO mechatronic students and it enables to hands-on test the proposed improvements of the wall component
- Robotic simulation, where a robot assembles the final redesigned version of the whole wall component. The simulation is run on a computer with the software RoboDK (2019b) and it provides insight in the time required to assemble the whole wall component.

The robotic process hypothesis also provides a robotic process for which a robot arm can be chosen. The most suitable robot is the industrial robot Smart5 NJ 110-3.0 from Comau (2019a). After the robot is chosen, the geometry of the wall component will be redesigned with consideration to the following two objectives: reduce assembly time and reduce CNC milling time. The redesign guidelines of the DFA2 method help to make the wall component suitable for robotic assembly and to reduce the assembly time (Eskilander, 2001).

Then, a time comparison in seconds is made of every assembly step between the manual and robotic assembly method for: the section (that is used in the physical experiment) and the whole wall component (that is used in the robotic simulation). In addition to that, the CNC milling time is considered for the whole wall component.

Conclusions

With the redesign guidelines of the Design For Automatic Assembly (DFA2) method the wall component is redesigned for automatic assembly. This resulted in three geometry versions:

- Version A - Original, which is the unmodified geometry.
- Version B - DFA2 *without* extra tolerances, which is the redesigned geometry with less assembly steps.
- Version B - DFA2 *with* extra tolerances, the same as the previous, but with extra tolerances at the connections that allow for assembly without a hammer.

Before the main question is answered, it needs to be stated that different electric screwdrivers were used in the two assembly methods. Therefore, the time required for screwing is omitted from the results described here below.

In order to answer the main question, the time required for the two assembly methods need to be compared based on the improved geometry version. So the simulated robotic assembly method that uses geometry version B - DFA2 *with* extra tolerances, takes 122 seconds to complete (without placing the screws) and it is 1,17 times faster than manual assembly of the same geometry. Hypothetically, it would mean the annual production of the robot is about 5,5 times higher than the manual method.

The CNC milling process takes much longer than the robotic assembly simulation. The latter is almost 8 times faster, which would mean the robot would be waiting for new elements most of the time.

Recommendations

Thirteen different recommendations are made, most of them are focussed on various aspects of the robotic process. The others are discussing the following topics: Total Cost of Ownership, CNC milling time, geometry, structure of the wall component and, lastly, a direction for a new construction method is presented.

Table of contents

Abbreviations

Abbreviation	Definition
AI	Artificial intelligence. “A collective term for science that attempts to imitate, or even surpass, human intelligence using computers. Devices become self-learning or self-organising to be able to make decisions without human intervention” (Wassink, 2018, p. 7).
CNC milling	Computer Numerical Control milling. “A machining process which employs computerized controls and rotating multi-point cutting tools to progressively remove material from the workpiece and produce a custom-designed part or product” (Ronquillo, n.d.).
DOF	Degrees Of Freedom. “Specific, defined modes in which a mechanical device or system can move. The number of degrees of freedom is equal to the total number of independent displacements or aspects of motion” (Rouse, 2009a). A human arm for example, has seven degrees of freedom (Kim, Miller, Al-Refai, Brand, & Rosen, 2011).
HBO	Higher professional education, or *Hoger beroeps onderwijs* in Dutch. A type of education provided by HBO institutions in the Netherlands (Government Of The Netherlands, n.d.).
N/A	“Not available or non applicable. The N/A abbreviation is used to fill up a blank portion of a form, chart, or other document so the reader knows the section is either not available at that time or non applicable” (Computer Hope, 2017).
PLC	Programmable Logic Controller. An industrial computer control system that controls the CNC milling machine. Based on certain input provided by sensors or humans, the system controls output, like the CNC milling head (AMCI, n.d.).
TCO	Total Cost of Ownership. “The purchase price of an asset plus the costs of operation” (Twin, 2019).
TNM	TheNewMakers. The company that provided the case study for this research.

Glossary

Term	Definition
Chamfer	"A cut on the edge or corner of something that makes it slope . . . rather than being perfectly square" (Chamfer, n.d.).
Characteristic	"A feature or quality belonging typically to a person, place, or thing and serving to identify them" (Characteristic, n.d.).
Cobot	A cobot, also known as collaborative robot, refers to "A computer-controlled robotic device designed to assist a person" (Cobot, n.d.).
Constraint	A goal with a specified limit (e.g.: weight should not exceed 46 kg).
Criteria	Plural of criterion.
Criterion	"A standard by which you judge, decide about, or deal with something" (Criterion, n.d.).
Element	A part of the wall component of TheNewMakers that is cut out of sheet material.
End effector	"A device or tool that's connected to the end of a robot arm where the hand would be. The end effector is the part of the robot that interacts with the environment" (Rouse, 2009b).
Geometry	"The shape or form of a surface or solid" (Geometry, n.d.).
Humanoid	"A machine or creature with the appearance and qualities of a human" (Humanoid, n.d.).
Objective	An indefinite goal without a specified value to attain (e.g.: minimise assembly time).
Repeatability	It is the average deviation measured during a movement sequences between three different points at the highest possible operating speed and while carrying the maximum payload (International Organization for Standardization, 1998).
Robot	"A machine controlled by a computer that is used to perform jobs automatically" (Robot, n.d.).
Vacuum gripper	A type of end effector "that uses a suction cup connected to a vacuum source to lift and handle objects" (Vacuum gripper, n.d.).
Wall component	A building block in the LEGO-inspired construction system of TheNewMakers.

INTRODUCTION

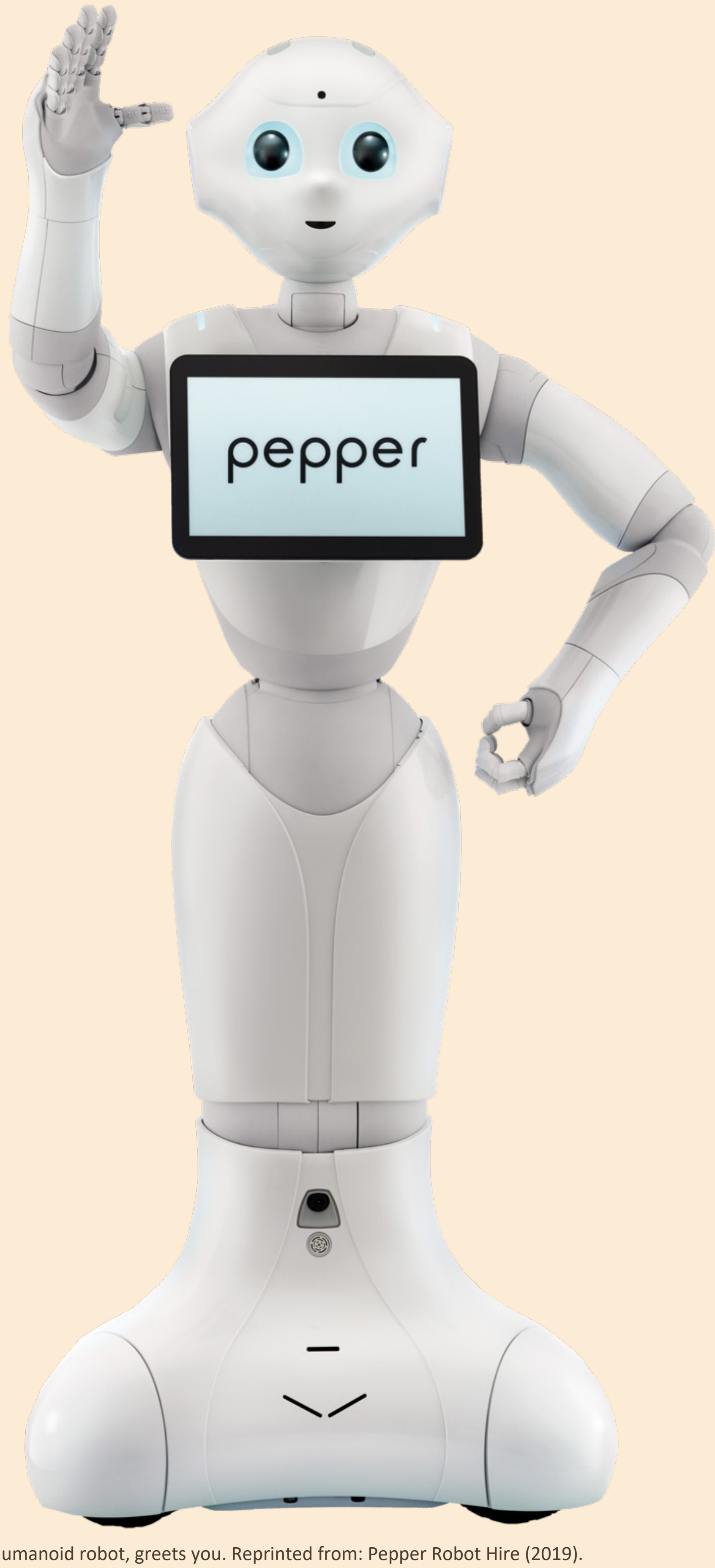


Figure 2: Pepper, a social humanoid robot, greets you. Reprinted from: Pepper Robot Hire (2019).

1 The imminent ubiquity of robots

The number of robots in the world is rapidly growing. Bouchard (2017) is one of the many authors who write about the prediction that more and more robots will play a significant role in multiple aspects of our lives. In his book, “Lean Robotics: a guide to making robots work in your factory”, he writes: “The fact is that robots will inevitably be put to work in greater and greater numbers” (p. 48).

Let us, therefore, examine some data regarding this statement. Research conducted by The International Federation of Robotics concluded that almost 3,8 million industrial robots will be operational worldwide in 2021 (IFR, 2018). The graph in Figure 3 visualises the increase in industrial robots over the past decades.

It is possible to place robots in two major categories, those of industrial robots and service robots (Bekey et al., 2006, p. 55). The former entails robots that are working in manufacturing processes and the latter refers to the other robots, which are used for a broad range of different applications. Just as with industrial robots, the number of service robots worldwide is increasing. At the end of 2016, the installed base of service robots was 29,6 million and is expected to reach 264,3 million by 2026 (BergInsight, 2017).

In order to get an impression of the broad range of areas in which service robots are being applied several examples are shown, which are:

An autonomous stunt double developed and used by Disney (Panzarino, 2018), which can be seen in Figure 4; Figure 5 shows a closet that folds and sorts laundry (Laundroid, n.d.); Figure 6 depicts a rendered image of a robot that is cooking in a domestic environment (Moley, n.d.); Cooking robots are already being used in the hospitality industry (Troitino, 2018), which is displayed in Figure 7. The hamburger restaurant gets mostly positive feedback. On Yelp, an online review platform, it gets an average customer rating of four out of five (Yelp, 2019); Agricultural tasks are carried out on a small scale in the form of FarmBot (Farm Bot, n.d.) as shown in Figure 8; Figure 9 illustrates a robot that focusses on the larger agricultural scale. The first farm of 750 square meters is up and running and has the potential to produce 30 times more per hectare than in traditional farming (Iron Ox, 2018); Robots which resemble the walking abilities of humans are also known as humanoids. A good example is the robot Atlas by Boston Dynamics (Boston Dynamics, 2018a). It is able to walk over uneven terrain, snow, stairs and it can even do a backflip. The latter can be seen in Figure 10; Social robots are also starting to make their way into our lives. Figure 11 shows a robot that is interacting with the elderly (Johnston, 2018). These examples and data present a glimpse of what can be expected in the future.

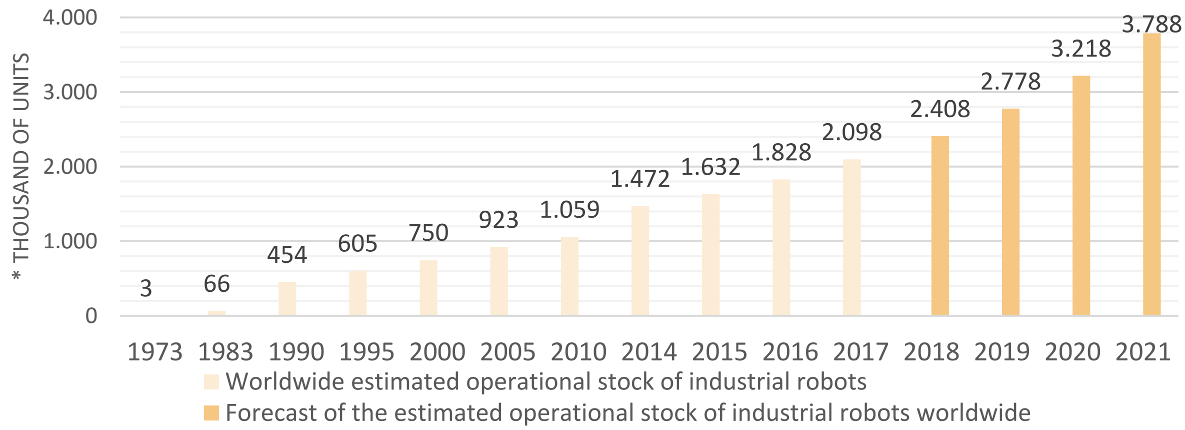


Figure 3: Worldwide estimated operational stock of industrial robots which shows the growth in number of industrial robots throughout the years as well as the forecasted growth. Adapted from: IFR (2018).

Figure 4: Three steps of the stunt double being launched, flying through the air and landing in a safety net. This prevents the need for stunt doubles to be put trough some of these dangerous stunts. Adapted from: Panzarino (2018).

Figure 5: Dried laundry can be dropped in the bottom drawer. Two robot arms will fold the clothes and sort it based on who wears them. Reprinted from: Laundroid (n.d.). Unfortunately, towards the end of the research, the company filed for bankruptcy (Seven Dreamers, 2019).

Figure 6: Computer generated image of robot arms cooking behind glass. Reprinted from: Moley (n.d.).

Figure 7: Robot which is able to make 120 hamburgers per hour for $6 each. Reprinted from: Troitino (2018).

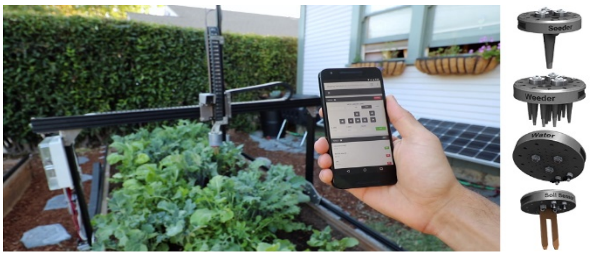


Figure 8: Farmbot being controlled using an app and some of its tools next to it. Adapted from: Farm Bot (n.d.).

Figure 9: Robot positions a growing container and can plant as well as harvest the crops. Adapted from: Iron Ox (2018).

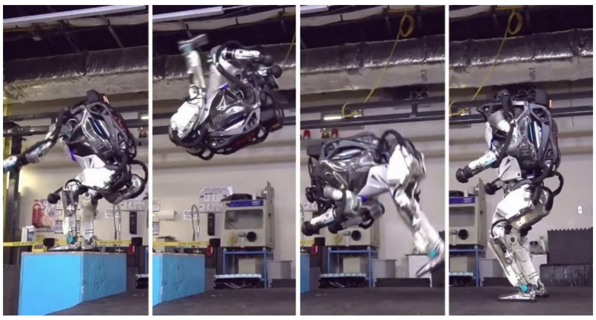


Figure 10: Four steps of Atlas performing a backflip. Adapted from: Boston Dynamics (2018a).

Figure 11: Social humanoid robot Pepper interacting with the elderly. Reprinted from: Japan Entertainment (2017).

2 Robots in the building industry

The previous chapter presented how more and more robots are being applied in a broad range of areas. The same is applicable within the building industry, it is exploring the application and implementation of robots as well. In this chapter examples of robots in the building industry will be shown.

2.1 Examples

2.1.1 Masonry robot

There are construction robots available that are able to lay bricks on-site. A company called Construction Robotics created a robot with these capabilities and called it SAM-100. SAM is an abbreviation of Semi-Automated Mason (Construction Robotics, n.d.). A photo of the robot in action can be seen in Figure 13.

2.1.2 On-site inspection robot

Another example can be seen in Figure 12. It contains a snapshot of a video from Boston Dynamics, which shows a robot they developed called Spot. It is a four-legged robot able to autonomously navigate around a building site and inspect work. At the end of 2018 on-site field tests were conducted and it will probably be available after mid-2019 (Boston Dynamics, 2018b).

Figure 12: Spot navigates a construction site and climbs a staircase. Adapted from: Boston Dynamics (2018b).


Figure 13: The robot SAM-100 is laying bricks on-site to construct the façade of The Lab School of Washington, DC. Reprinted from: Construction Robotics (n.d.).

2.1.3 3D concrete printing robot

There are numerous 3D concrete printing robotic projects that are prototypes or proof of concepts, but the following example takes the next step towards an actual concrete printed functional building. In Eindhoven, a team of six different partners, including the municipality, is working on a commercial housing project called Project Milestone. People will occupy the five houses, which requires the buildings to meet all modern rules and regulations. The first home is expected to be completed in 2019 (3Dprintedhouse, 2018). A render of the proposed design can be seen in Figure 14 and the robot in Figure 15.

Figure 14: The proposed design. Reprinted from: 3Dprintedhouse (2018).

Figure 15: The robotic 3D concrete printer. Reprinted from: 3Dprintedhouse (2018).

2.1.4 Foam cutting robot

A company called Odico (n.d.) developed a robot that performs Robotic Hot-Wire Cutting (RHWC) on a large scale. The robot controls a hot wire that slices through a foam block, which allows making complex geometries. These foam blocks will be applied as concrete formwork. Figure 19 shows the robot including its hot wire tool. Figure 18 shows how the foam is applied in the casting situation and the resulting concrete element can be seen in Figure 16. Figure 17 shows how multiple similar of these elements are applied in The Kirk Kapital building.

Figure 17: The Kirk Kapital building, located in Denmark. Reprinted from: Ravenscroft (2018).

Figure 16: The resulting prefab concrete element. Reprinted from: Odico (n.d.).

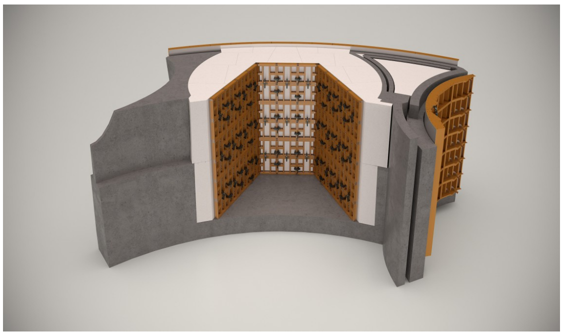


Figure 18: The foam elements applied as formwork. Reprinted from: Odico (n.d.).

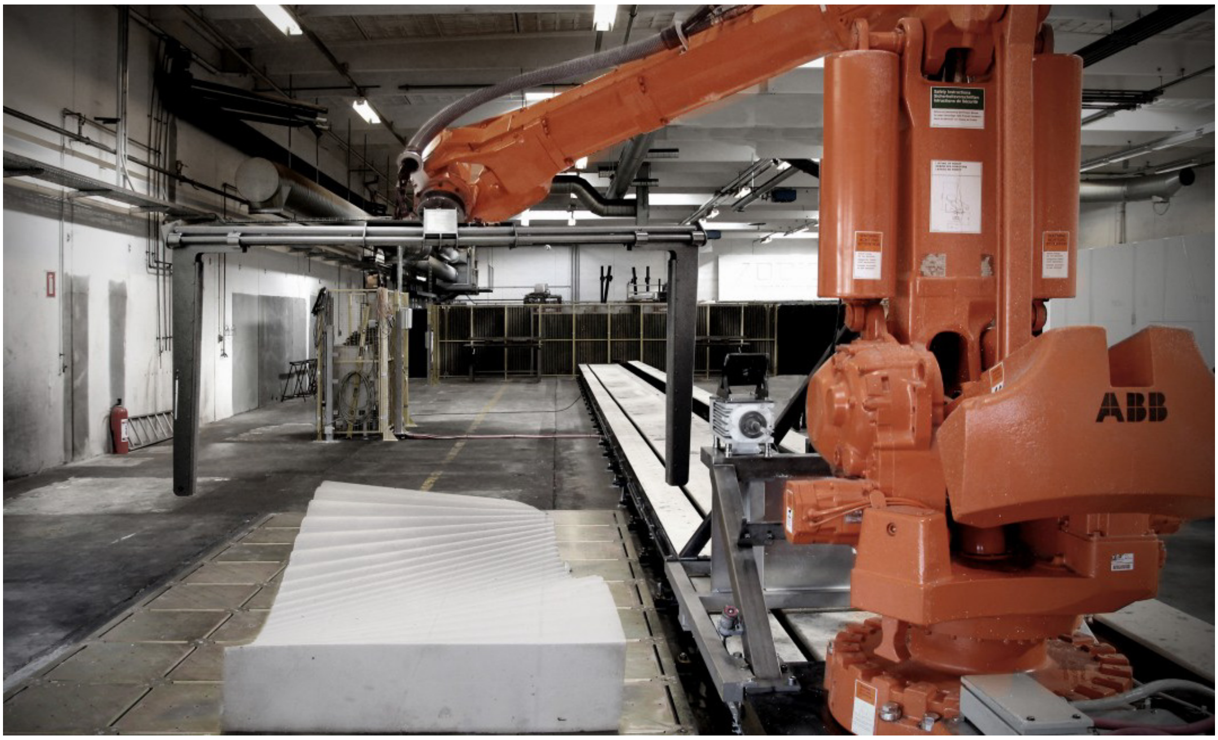


Figure 19: The robot including its hot wire tool and the resulting foam element which will be utilised as concrete formwork. Reprinted from: Odico (n.d.).

2.2 Assembly examples

This paragraph presents three state-of-the-art projects of robotic assembly that are applied in the building industry.

2.2.1 Ceiling robots erect a wooden frame

Researchers at ETH Zurich university applied a method in which robots work together with humans to build a wooden, load bearing structure that will be applied in the DFAB HOUSE project. Robots collaborate with a human to erect a construction made of wooden beams. Figure 21 shows the human worker screwing the beams together. The robots do everything else, such as: saw beams to size, drill holes and move beams to the correct position which can be seen in Figure 20 (ETH, 2019).

Figure 20: Robot arms of ABB Robotics position wooden beams as per a digital design. The robot arms are able to slide because they are mounted on a rail which hangs off the ceiling. Reprinted from: ETH (2019).

Figure 21: Worker fastens beams together while robot arms hold them in place. Reprinted from: ETH (2019).

2.2.2 Mobile robot erects a wooden frame

Compared to the researchers at ETH Zurich University, research towards robotic assembly was conducted on a smaller scale and with a different type of robot. Stumm, Devadass, and Brell-Cokcan (2018) tested their method on their project called the Twisted Arch that can be seen in Figure 24. Figure 25 shows how the mobile robot carries out several assembly tasks on its own, like: handling all wooden elements one by one, align each one, saw them to size and position every element. However, the latter requires human assistance. The robot is able to correctly align the wooden beam with the axis of its target location, but a human needs to gently guide the beam in its final position and then fasten it to another beam using screws.

2.2.3 Humanoid installs drywall

The National Institute of Advanced Industrial Science and Technology, abbreviated as AIST, in Japan has developed a humanoid robot prototype, which is shown in Figure 22. This prototype is capable of autonomously assembling a part of drywall. Starting by picking up a plywood sheet from a stack, to grabbing a battery screwdriver and fastening a board to a frame as can be seen in Figure 23. This prototype confirms some data of the research by Frey and Osborne (2017). This research looked into the probability of robots replacing any of the 702 detailed occupations examined. In this study, it is concluded that robots will replace the work of drywall installers with a probability of 79%.

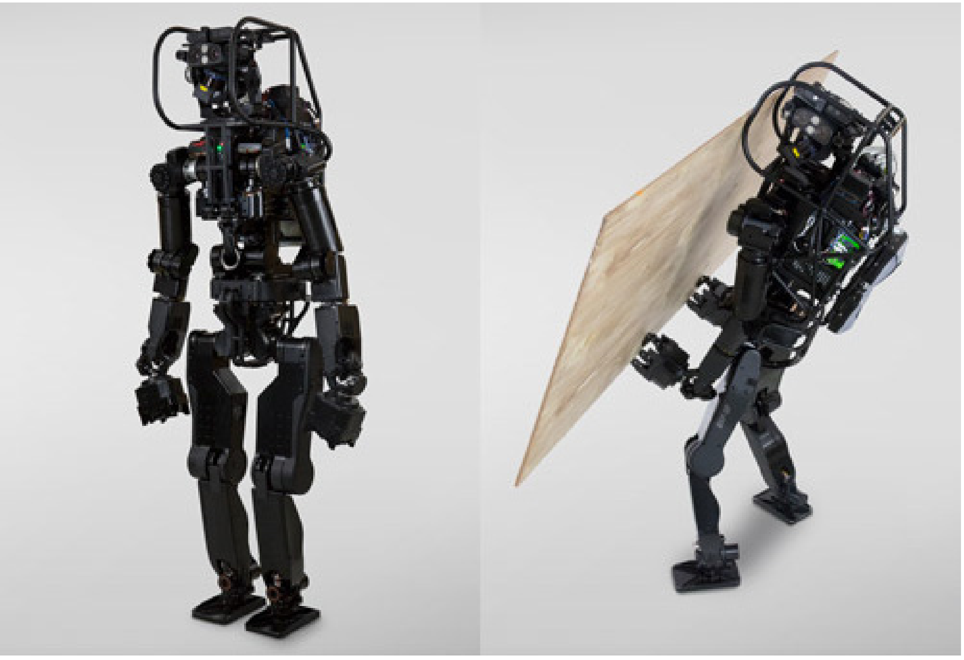


Figure 22: HRP-5P appearance (left) and HRP-5P carrying a plywood board that weighs approximately 13 kg (right). Adapted from: AIST (2018).

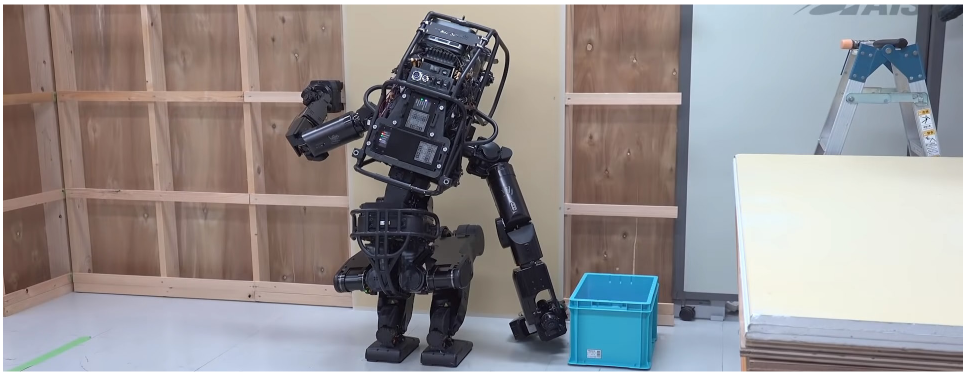


Figure 23: HRP-5P autonomously screwing the board into the wooden frame. Adapted from: AIST (2018).

Figure 24: The Twisted Arch is constructed with the help of a KUKA LBR iiwa (LBR refers to “lightweight robot” and iiwa stands for “intelligent industrial work assistant”) which is mounted on a KUKA mobile platform. Reprinted from: Stumm et al. (2018).

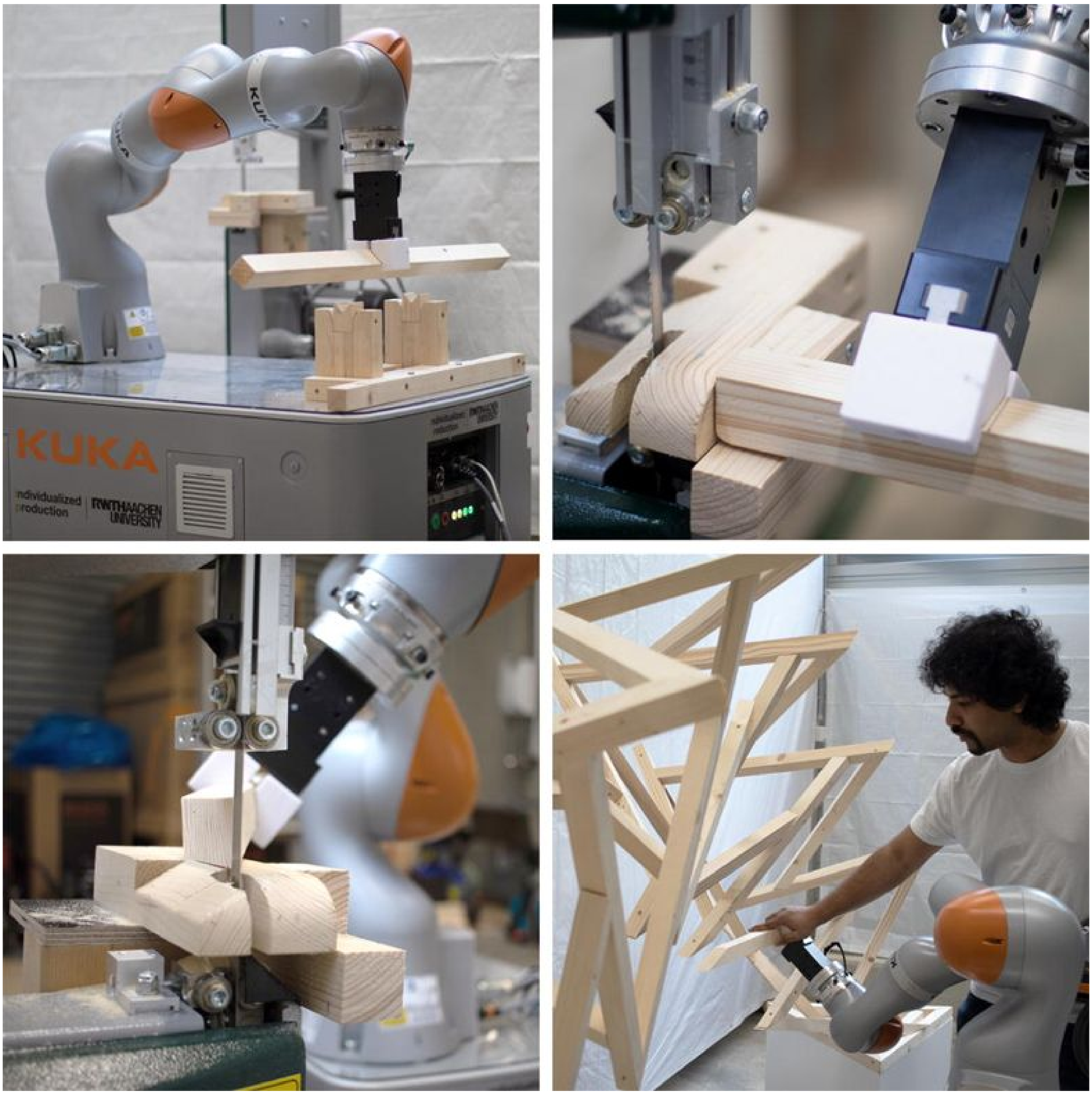


Figure 25: Assembly operation examples pertaining to the main categories of assembly. Upper left: handling; upper right: aligning; lower left: sawing; lower right: positioning. Reprinted from: Stumm et al. (2018).

2.3 Industry 4.0

Industry 4.0 is also known as the fourth industrial revolution. In this concept, robots play a large role in factories. It's a continuation on the first, second and third industrial revolution. These entail respectively, mechanization using the power of water and steam, mass production and electric powered assembly lines, IT systems to improve automation. The purpose of Industry 4.0 is explained in a paper by Zhou et al. (2015):

> The purpose of Industry 4.0 is to build a highly flexible production model of personalised and digital products and services, with real-time interactions between people, products and devices during the production process. (p. 1)

From the moment Industry 4.0 was officially presented in 2013, it will take time before the concept is broadly implemented in different fields. "The realization of the Industry 4.0 vision will most likely span a decade or two, but it is already impacting various system designs" Dhanani (2015). So it should be kept in mind when designing an industrial process.

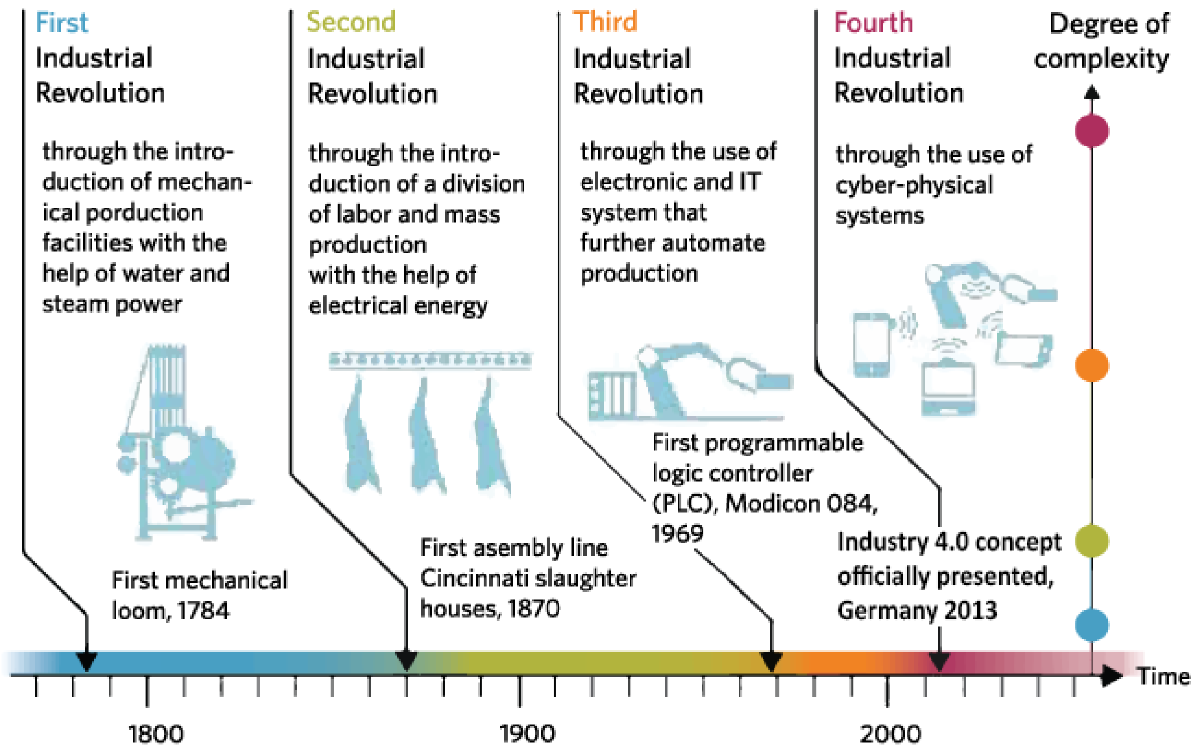


Figure 26: Timeline of the four industrial revolutions including a brief description and starting point of each industry. Adapted from: Dhanani (2015). Added the year and location of industry 4.0 from: Zhou et al. (2015).

2.4 Necessity or gimmick?

Are robotics in the building industry a necessity or a gimmick?

2.4.1 Labour shortage

Recent publications state there are not enough construction workers in the Netherlands. One article describes how the construction industry coped with a shortage of 48.000 skilled labourers in 2018 (Geertsma). Another news item writes that four out of ten construction companies coop with a labour shortage and states that this deficiency is here to stay. Therefore, they conclude, the construction industry needs to robotise (Bremmer, 2018).

As per an article in the TU Delft magazine Delft Outlook, it appears to be that a decrease of pressure on the building industry won't happen soon. It states that the Netherlands will need to realise a million new homes before 2030. A demand of this magnitude is comparable to previous eras in the Netherlands when the amount of construction work was high. This happened after the second world war and during the 1990s (Jongeneel, 2018).

2.4.2 Uncertainty

It is important to ask the question if robots are an absolute necessity or merely a gimmick. A lot of debate is going on regarding this question. It is a difficult question to answer, partially, because it is hard to predict the future of robotics accurately. This can be clearly seen in a data overview of multiple pieces of research on estimations regarding the influence of robotics on the global and USA job market. The results are far apart as shown in the graph in Figure 27. E.g., one research states that robots will destroy 2.000.000.000 jobs in the USA job market by the year 2025, while another research thinks 9.100.000 jobs will be lost to robots worldwide until 2030 (Winick, 2018).

Sami Atiya, president of the Robotics and Motion division of Swedish-Swiss automation company ABB Group, has an interesting approach for companies which are implementing robotics to be able to coop with the unpredictability. He said: "the same advice we give our customers every day: invest in automation solutions that provide flexibility and agility to grow in whichever direction the market goes" (ABB, 2018).

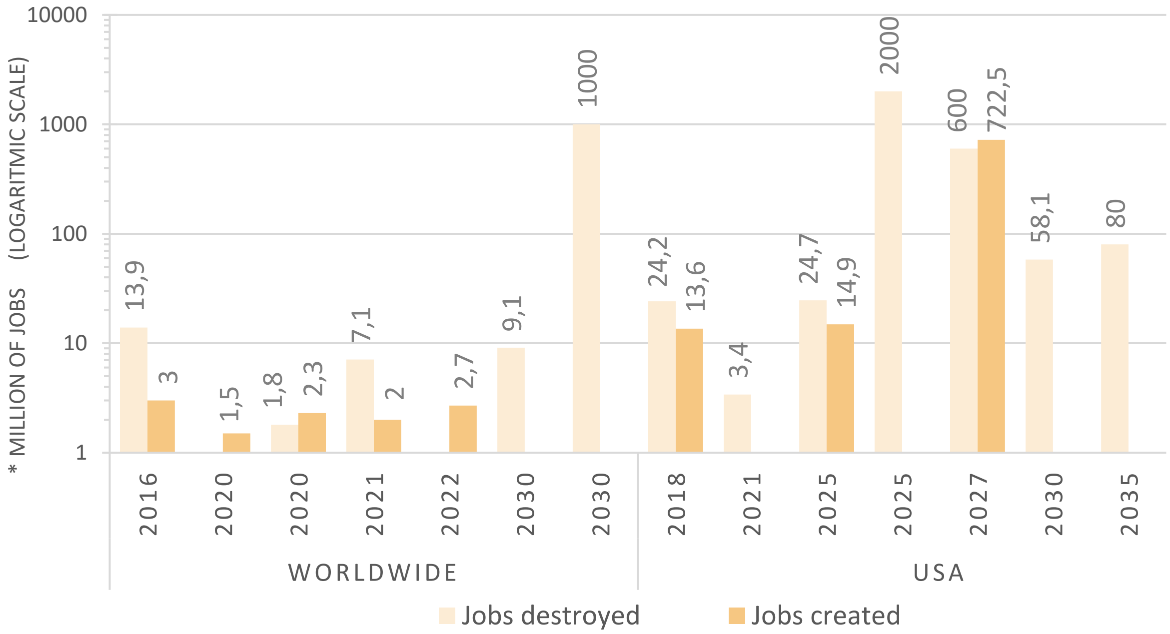


Figure 27: The predicted number of jobs automation will create and destroy based on 14 researches. Two nuances when interpreting the data: values have been rounded and ranges have been converted to averages. Adapted from: Winick (2018).

3 Research framework

3.1 TheNewMakers

TheNewMakers, abbreviated as TNM, is a small company that produces CNC milled components. Based on a digital design, precisely cut to size elements are milled from sheets of wood as per a file-to-factory process. The file-to-factory process is part of the encompassing Industry 4.0 concept (Oosterhuis, Bier, Aalbers, & Boer, 2004).

TNM designed and engineered their own modular construction system with different building components. One of these components is a wooden wall component that is designed for a modular façade. The wall component is composed of multiple CNC milled wooden elements. In short, the current process of TNM from design to completion goes as follows: a 3D design is made based on the building components of TNM, the building components are sent to the CNC milling machine, precisely CNC-cut elements are automatically produced from sheets of wood, the CNC milling machine is unloaded by hand, the elements are assembled manually into wall components, the components are transported to site, and lastly, all components are installed.

3.2 Problem statement

The CNC milling process of TheNewMakers is automated, but the assembly is done manually. Manual assembly can:

- Take more time
- Be more expensive
- Be less reliable
- Be less accurate
- Put certain restraints on the production quantity
- Put certain restraints on the product quality

3.3 Objective

The objective is: redesigning the geometry of the CNC milled wooden elements for automatic robotic assembly in such a way that it is faster than the manual assembly process. It has the potential to:

- Save time, which is the main focus of this research
- Save money
- Be more reliable
- Be more accurate
- Increase production quantity
- Improve product quality

Final products

The geometry of the CNC milled wooden elements will be redesigned on the basis of two objectives: reducing the robotic assembly time and reducing the CNC milling time.

In order to physically try and test the assembly steps of the redesigned geometry, an experiment will be conducted with a section of the wall component. It will be executed at the HighTechCentreDelft (HTCDelft) and in collaboration with HBO (professional bachelor's) mechatronic students from The Hague University of Applied Sciences, called *De Haagse Hogeschool* (HHS) in Dutch. The starting point of the experiment is a hypothesis about the robotic process, which is made based on the literature review and the case study of TNM. The experiment will help to rate certain aspects of this robotic process hypothesis plus it will provide input for a robotic assembly simulation of the whole wall component.

A time-based comparison will be made using the two redesign objectives. So the assembly time of the redesigned geometry during the manual and robotic process will be compared as well as the CNC milling time of the original and redesigned geometry.

Limitations

In order to prevent the scope of this research from becoming too big, two limitations are set:

- A single robot arm is used
- The robot arm is fixed to the floor

3.4 Research questions

Main research question

How could the elements of a wall component of TheNewMakers be redesigned for automatic assembly while taking less time to assemble than the manual process?

Sub-questions

Q1) What does the current manual assembly process of a wooden wall component of TheNewMakers look like?
Q2) How can the geometry of the CNC milled elements be redesigned to reduce the automatic assembly time as well as the CNC milling time?
Q3) What kind of robot arm is best suited to the robotic assembly process of TNM?
Q4) Does the proposed robotic assembly process take less time than the manual process?

3.5 Approach and methodology

The book Lean Robotics: a guide to making robots work in your factory by Bouchard (2017), provides an efficient method regarding the deployment of robotics in a production process. A graphic representation of the lean robotics methodology can be seen in Figure 28. This report will focus on the first phase, which is the design phase. Certain aspects of the lean robotics methodology will be implemented.

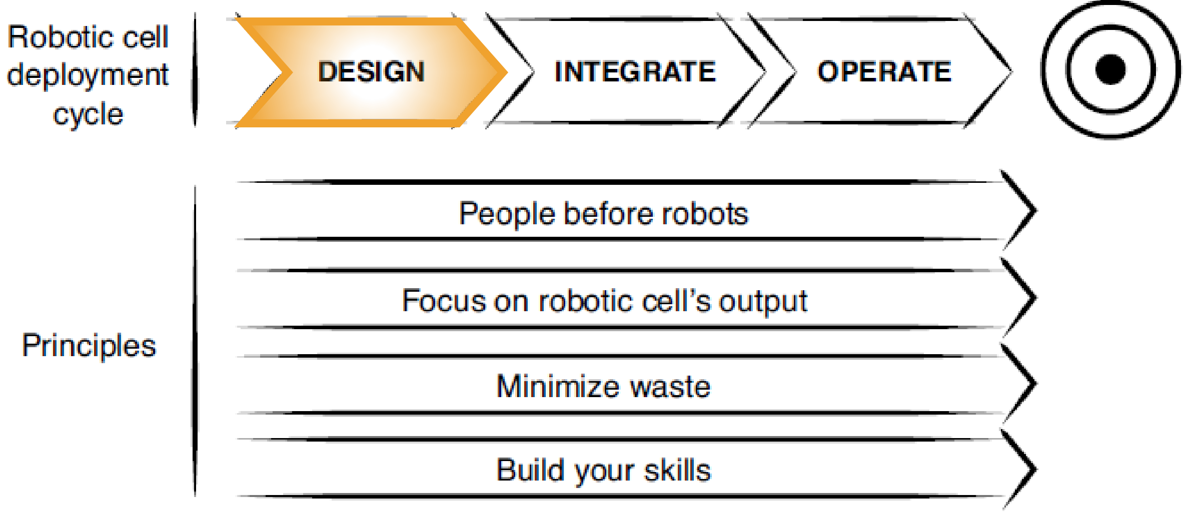


Figure 28: Lean robotics methodology overview. This research explores the highlighted design phase. Adapted from: Bouchard (2017).

The lean robotics book explains the methodology in further detail. It describes the design phase, along with a flowchart that visualises the phase, which is shown in Figure 29. Detailed information of this tried and tested method is going to be used as a rough guideline for the robotic process hypothesis and it will ensure that no vital steps are missed.

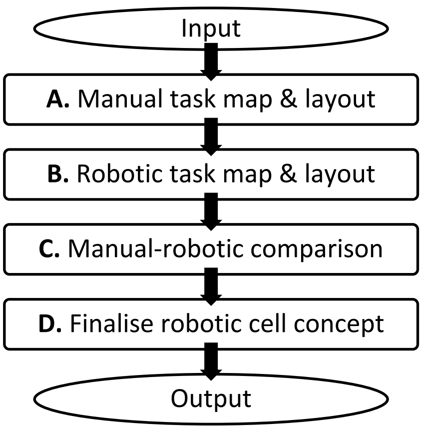


Figure 29: Design phase of lean robotics. Adapted from: Bouchard (2017).

Research design
On the next page, in Figure 30, a visual representation of the research design can be found

Robotic process hypothesis
The starting point is a robotic process, which is a hypothesis that is based on: the current manual process of TheNewMakers, findings in the literature and the Lean Robotics methodology.

Section assembly experiment (the experiment is marked with a brass colour throughout the report)
An experiment will be conducted in collaboration with HBO mechatronics students and is marked in Figure 30 with a brass coloured box and blue dashed line. Brass is chosen because the HBO students are studying at The Hague University of Applied sciences of which the logo colour is brass, just as blue is for the Delft University of Technology. Initially, the idea was to use a large robot to assemble the whole wall component, however, no such robot was available. Therefore, only a section of the wall component will be assembled with a small UR5 collaborative robot that is made available for testing at the HighTechCentreDelft. The aim of the experiment is to: physically test geometry modifications of the CNC milled elements with a robot, generate input required for the robotic simulation and briefly evaluate the five experiment topics in relation to the robotic process hypothesis.

Q1) What does the current manual assembly process of a wooden wall component of TheNewMakers look like?
Conduct a case study of the manual assembly process of a CNC milled wooden wall component of TheNewMakers. This entails working on-site and helping to assemble multiple wall components in order to be able to analyse the required steps. Then visualise each step schematically while describing the following aspects:

i. Movements. Describe the movements a human needs to perform (e.g., up, down).
ii. Actions. Include the required actions (e.g., rotate, lift).
iii. Tools. Mention the used tools (e.g., screwdriver, hammer).

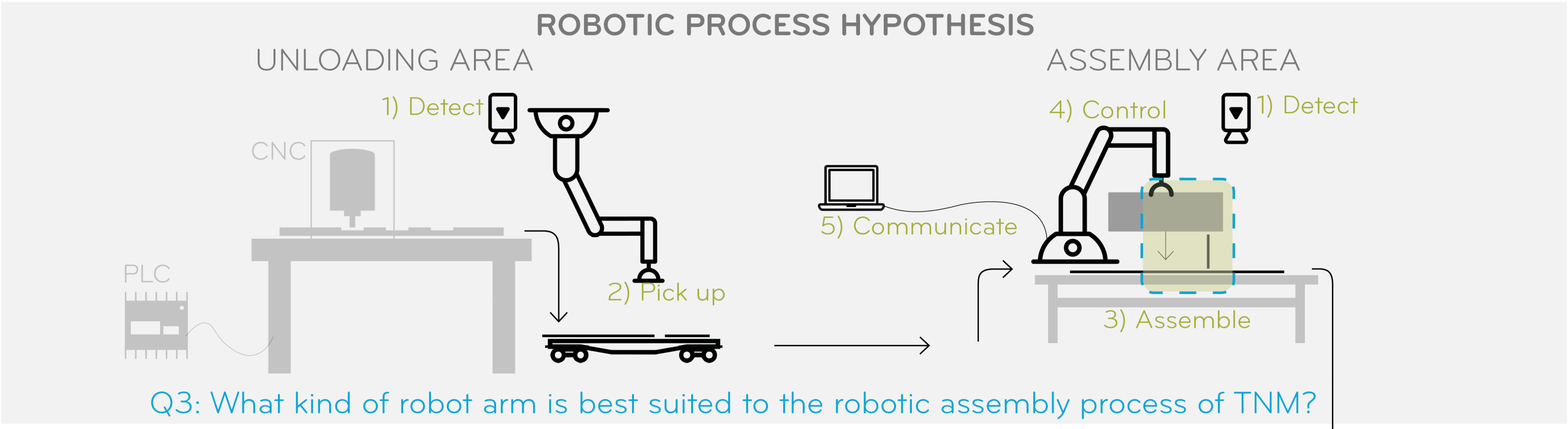
Describe three assembly detail principles, being: Q1A slide, which is a principle where one element slides in two other elements; Q1B screw, where elements are being screwed together; Q1C dog bone, when an element is pushed into a dog bone connection.

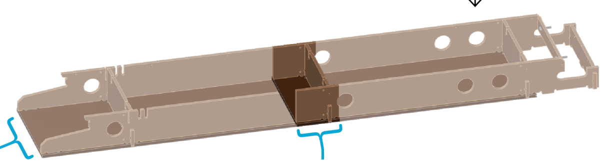
Q2) How can the geometry of the CNC milled elements be redesigned to reduce the automatic assembly time as well as the CNC milling time?
Redesign the geometry of the wall component, which aims to reduce the time required for robotic assembly and CNC milling. Use the Design For Automatic Assembly (DFA2) method to make the wall component suitable for an automatic robotic assembly process.

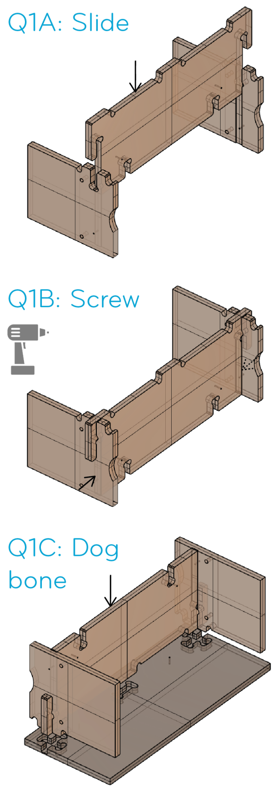
Q3) What kind of robot arm is best suited to the robotic assembly process of TNM?
Make an overview of different robots divided into two types, being industrial robots and cobots. Compare the robots on the following aspects: payload, reach, velocity, mass, repeatability, estimated Total Cost of Ownership (TCO). Propose a robot and state the number of robots required that is able to automatically assemble the wooden wall component of TheNewMakers.

Q4) Does the proposed robotic assembly process take less time than the manual process?
A time-based comparison will be made between the manual assembly process and robotic assembly process. This comparison considers the original geometry (version A) and the updated geometry (version B) *with* or *without* extra tolerances, plus how many seconds it takes to CNC mill the elements and to assemble the whole wall component. The available robot is not able to assemble a whole wall component, therefore, the robotic assembly process will be simulated.

Main research Q: How could the elements of a wall component of TheNewMakers be redesigned for automatic assembly while taking less time to assemble than the manual process?







Q1 What does the current manual assembly process of a wooden wall component of TheNewMakers look like?

Q2 How can the geometry of the CNC milled elements be redesigned to reduce the automatic assembly time as well as the CNC milling time?

Q4 Does the proposed robotic assembly process take less time than the manual process?

Assembly experiment with section wall component. In collaboration with HBO mechatronics students.

Experiment topics	Considerations
1) Detect elements (recognise and check)	1A) Vision (shape) 1B) 3D model 1C) RFID / QR
2) Pick up elements	2A) Impactive (e.g. gripper) 2B) Ingressive (e.g. pins) 2C) Astrictive (e.g. vacuum) 2D) Contigutive (e.g. adhesive)
3) Assemble elements	3A) Actions required 3B) Extra hardware required
4) Control robot arm	4A) Manual 4B) Teach pendant 4C) Software
5) Communicate	5A) Human & robot (HMI) 5B) Robot & detection 5C) Robot & end effector

Table to compare time in seconds for the whole wall component

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	U	V	W	R
2. CNC [s]	X	Y	Z	Z
1+2 = Total time	U+X	V+Y	W+Z	R+Z

Conclusions and recommendations

Figure 30: Research logical organisation scheme.

3.6 Relevance

3.6.1 Societal relevance

In the past, mankind spent the majority of their time growing food in order to survive. The first industrial revolutions allowed a lot of people to shift their focus from agriculture to other tasks, which helped humanity to significantly increase worldwide knowledge in a relatively short time span. Figure 31 shows the relation between the sheer drop in the percentage of the labour force working in agriculture and the starting point of industry 1.0, 2.0 and 3.0. This shows that automating processes help humans to allocate additional time to more beneficial tasks, which is also described in the book Lean Robotics:

> Humans can create much more value . . . when they're working creatively rather than repeating the exact same motions day after day. Robots can take care of this necessary yet low-value-added work instead; and workers can take on higher-impact tasks. (Bouchard, 2017, p. 40)

With this research, an interdisciplinary cross level collaboration will be established between students from HBO (professional bachelor's mechatronic students at *De Haagse Hogeschool*) and WO master level (graduate student of the master track Building Technology at Delft University of Technology). In this research the wall component of TheNewMakers will be redesigned for automatic assembly, while the mechatronic students will focus on the practical robotic challenges. Knowledge is going to be shared and an iterative feedback loop will be established between the two different levels.

3.6.2 Scientific relevance

This research contributes to a fully automated process from the digital design file to an assembled product, which is ready to be used on the construction site. Thus, taking the next step beyond a file-to-factory approach, which can be called: file-to-site.

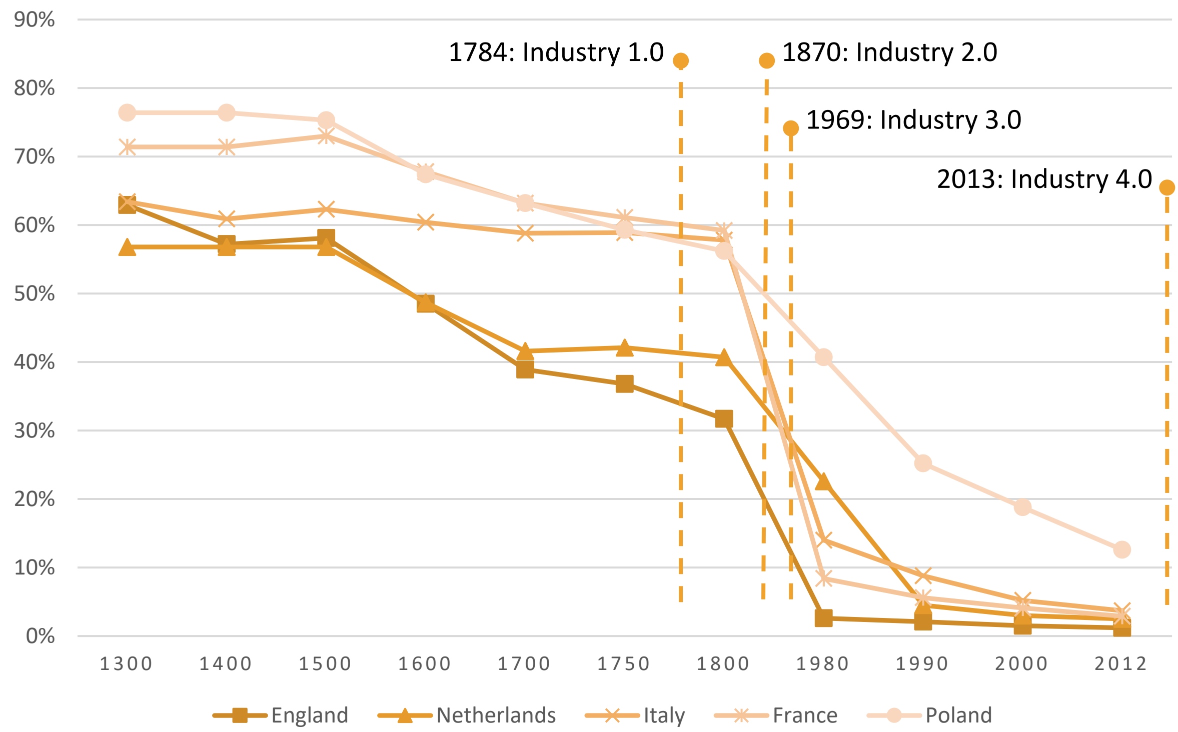


Figure 31: Percentage of the labour force of several countries working in the agricultural sector. Adapted from: Roser (2018). Added the starting year of Industry 1.0 up and till 4.0. Adapted from: Zhou, Taigang, and Lifeng (2015).

3.7 Time planning

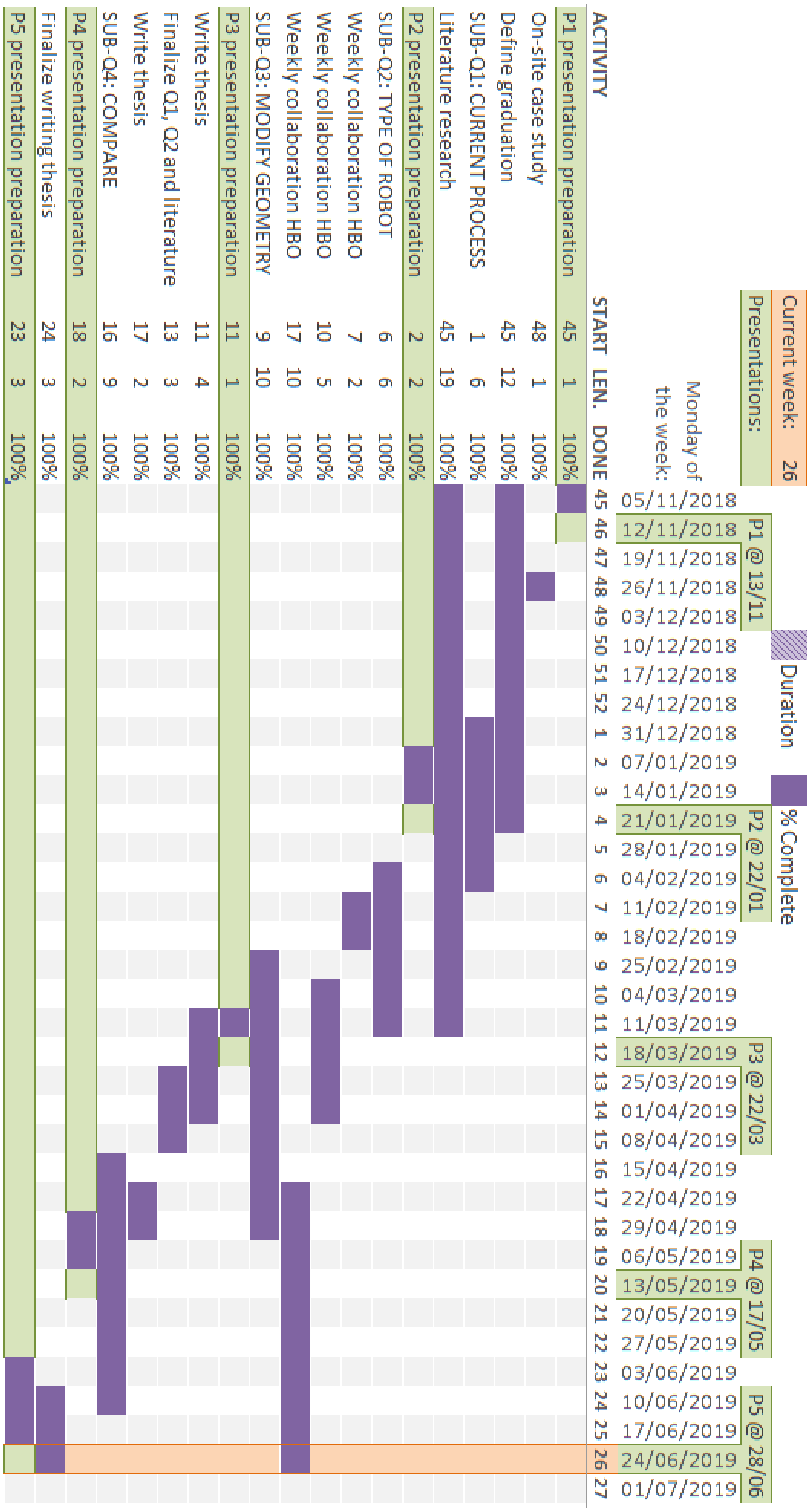


Figure 32: Finalised dynamic time planning in MS Excel that was used during the research.

LITERATURE REVIEW

Figure 33: Industrial robot arm lifting a car. Adapted from: Koot (n.d.).

4 Requirements robotic process

4.1 Lean Robotics

One of the most critical factors in a robotic process is the robot arm, which will be discussed in the next chapter. Besides the robot arm, there are other systems required to set up a robotic process. This chapter will describe these systems and present several examples. As described in the book “Lean Robotics: a guide to making robots work in your factory”, a robotic process consists of: the robot arm, a vision system (called sensors in the book), the end effector (the book calls it grippers), software, and other hardware (called other components in the book). Safety is an important aspect that will be discussed as well (Bouchard, 2017).

4.2 Vision system

This paragraph presents an overview of different vision system types and a few available vision systems are shown in order to develop an understanding of the possibilities.

It is important to understand that the vision system can influence the overall capabilities of the robotic process. For example, if a robot arm can move with high accuracy, but the vision system has low image quality then the overall accuracy is likely to be lower.

In research conducted by Pérez, Rodríguez, Rodríguez, Usamentiaga, and García (2016), different commonly used vision techniques are classified, which is shown in Table 1. On the next page in Table 2, a more detailed overview of these techniques is presented and Table 3 elaborates on the advantages and disadvantages.

Table 1: Classification of different vision techniques. Adapted from: Pérez et al. (2016).

Vision type	Single camera	Multiple cameras
Passive	2D.	Stereo vision. Photogrammetry.
Active	Time of flight. Structured light. Light coding. Laser triangulation.	Structured light. Projected texture stereo vision.

4.2.1 Available vision systems

Three available vision related systems are presented here. First, a widely used software package is shown followed by a vision system for industrial robots. Lastly, an example of a camera attached to a robot arm is given.

OpenCV

“OpenCV is a highly optimized library with focus on real-time applications” (OpenCV, 2019). OpenCV stands for Open source Computer Vision and it is a vision system that was released in 2000. It has many different application areas, just to name a few: object identification, stereo vision and motion tracking.

Vision for industrial robots

Figure 35 on the next page shows the application of a vision system that uses lidar sensors to detect when a human comes within a potential harmful reach of the industrial robot and its payload. The vision system will send a command to the robot to slow down or even come to a complete standstill. This recently released system is being sold by Veo Robotics that has partnered with four major players in the robotics industry, which are: KUKA, FANUC, ABB and Yaskawa. The goal of Veo Robotics is: “bringing perception and intelligence to industrial robots” (Veo Robotics, 2019).

UR wrist camera

The wrist camera made by Robotiq (2019b) is for a specific series of collaborative robots: the Universal Robot. It is a complete hard- and software solution with object recognition that is easy to install and use as is shown in Figure 34.

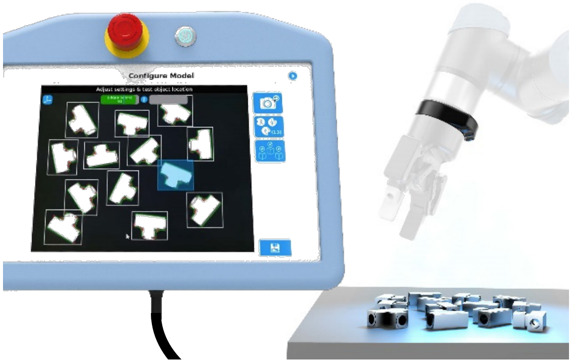


Figure 34: Wrist camera. Adapted from: Robotiq (2019a).

Table 2: Overview of different types of vision systems and their characteristics rated from bad (-) to good (+). Concrete measurements were available for the accuracy, therefore a best (++) rating was added. Adapted from: Pérez et al. (2016).

Vision type	Accuracy [mm]	Range [m]	Weight	Safe	Processing time	Environment influences
Stereo vision. Photogrammetry.	+ (0,050)	+	+	+	- (image processing)	- (brightness)
Projected texture stereo vision	+ (0,100)	- (0,25-3)	+	+	- (image processing)	- (brightness)
Time of flight	- (10,000)	- (0,8-8)	+	+	+	+
Structured white light	++ (0,127)	+	- (projector)	+	- (remain static)	- (light, brightness)
Structured blue LED light	++ (0,034)	- (0,16-0,48)	+	+	- (remain static)	+
Light coding	- (10,000)	- (1-3)	+	+	+	- (sun)
Laser triangulation	+	+	+	- (laser power)	+	- (brightness)

Table 3: Advantages and disadvantages of several types of vision systems. Adapted from: Pérez et al. (2016).

Vision type	Advantages	Disadvantages
Stereo vision. Photogrammetry.	Commonly used. High accuracy.	Influenced by environment. Physical marks required. Static object and camera needed.
Projected texture stereo vision	No physical marks required.	Influenced by environment. Static object and camera needed.
Time of flight	Independent of ambient light. Object and camera can move.	Low accuracy.
Structured white light	High accuracy.	Sometimes influenced by environment. Does not work with certain colours. Expensive. Sensors can be quite large. Static object and camera needed.
Structured blue LED light	High accuracy. Small sensor.	Expensive. Short working distance. Static object and camera needed.
Light coding	Inexpensive. Object and camera can move.	Low accuracy. Uncertified at industry level.
Laser triangulation	Inexpensive, depends on accuracy. Commonly used.	Dangerous for people, depends on laser. Short working distance. Line scanner.

Figure 35: Vision system that uses lidar sensors to enable a safe work environment for humans around industrial robots. Adapted from: Veo Robotics (2019).

4.3 End effector

The end effector of a robot is "a device or tool that's connected to the end of a robot arm where the hand would be. The end effector is the part of the robot that interacts with the environment" (Rouse, 2009b). The robotic end effector is also called the End-Of-Arm Tooling, which is abbreviated as EOAT. The number of different available end effectors is incredibly high. Besides these already available end effectors, it is relatively easy and cheap to develop one making the number of possible end effectors virtually endless. Some interesting end effectors are showed in this paragraph to develop an idea of the possibilities.

It is important to understand that the end effector can influence the overall capabilities of the robotic process. For example, if a robot arm is able to lift 50 kg, but the end effector can only lift 30 kg the overall payload capacity is reduced to 30 kg.

The robotic end effector can be either a tool (e.g. screwdriver, MIG welder, hammer) or a gripper. A gripper is used to grip an object in order to conduct a movement action or firmly hold it in place. Monkman, Hesse, Steinmann, and Schunk (2007) described four general categories in which all grippers can be positioned:

- Impactive, which physically clamps around an object.
- Ingressive, which pierces through an object on either an intrusive or non-intrusive way.
- Astrictive, which applies an attractive force to an object like vacuum suction, magneto adhesion or electro adhesion.
- Contigutive, which requires touching the surface of an object so an adhesion force can be applied like thermal or chemical adhesion or surface tension.

In the next subparagraph, examples are given of two end effector tools and four end effectors, one of each end effector gripper category.

4.3.1 Available end effectors

An interesting end effector tool is used in a robot called IkeaBot, which is able to assemble an Ikea table. The end effector tool screws in one leg at a time, which is shown in Figure 36. The researchers derived their inspiration from a similar tool used at the International Space Station. Figure 37 depicts a step by step process of the tool in action while rotating a battery so the rubber band mechanism of the tool is visible (Knepper et al., 2013). Figure 38 presents an example of a tool that is more widely known, it is a demolition hammer used as an end effector.

Guided by a 3D vision system, a special impactive gripper is able to pick many different objects autonomously from a bin, as can be seen in Figure 39. Figure 40 shows the different gripping positions of the end effector. It is developed by Soft Robotics and they describe it as an "air actuated soft elastomeric end effector" (Soft Robotics, 2019).

A gripper that is positioned within the ingressive category is shown in Figure 41. It is a needle gripper, which is mostly used to pick up fibrous materials such as woven carbon fibre or a sheet of insulation. The latter can be seen in Figure 42 (Schmalz, 2019).

Figure 43 shows a photo of an end effector with vacuum suction cups that carries a cardboard box, such an end effector is called a vacuum gripper and can be positioned in the astrictive category. The vacuum suction cups are connected to a vacuum source so objects can be carried (Vacuum gripper, n.d.). They are available in many different shapes and sizes, some small examples are shown in Figure 44.

There are only a few grippers that fall in the contigutive category. Figure 45 shows a photo of the Gecko gripper on display at Automatica 2018, which is "The Leading Exhibition for Smart Automation and Robotics" (Automatica, 2019). A image of the gripper itself can be seen in Figure 46. The white rectangles apply the same adhesive principle as a real gecko does, which eliminates the requirement for an air supply.

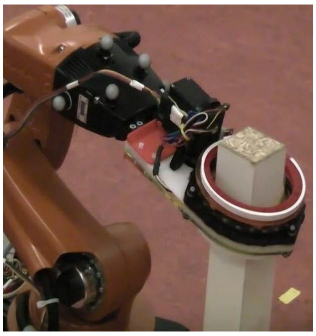


Figure 36: Tool screws in leg. Adapted from: Knepper, Layton, Romanishin, and Rus (2013).

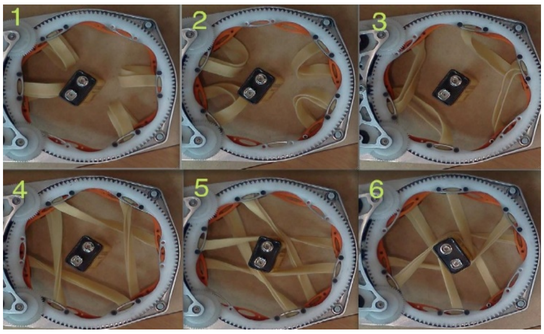


Figure 37: Six steps of the rotating process. Several rubber bands grab and rotate a battery. Reprinted from: Knepper et al. (2013).

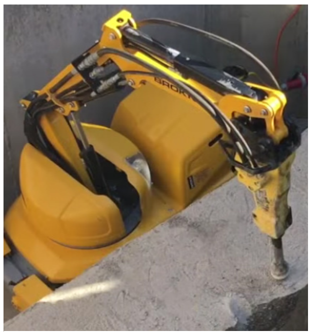


Figure 38: Robot arm with a pneumatic hammer. Adapted from: T&D World (2017).

Figure 39: Impactive gripper type that is able to bin pick many different objects. Adapted from: Soft Robotics (2019).

Figure 40: The different positions of the impactive gripper. From left to right: maximum open, parallel open and closed. Adapted from: Soft Robotics (2019).

Figure 41: Needle gripper with its pins protruding. Adapted from: Schmalz (2019).

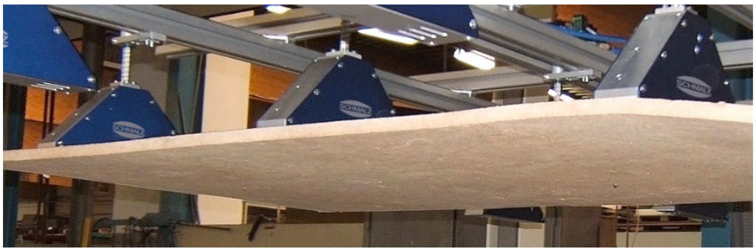


Figure 42: An three by one array of the ingressive gripper type carrying a sheet of fibrous material. Adapted from: Schmalz (2019).

Figure 43: A two by two array of vacuum suction cups lifting a cardboard box. Adapted from: Robotics Online (2019).

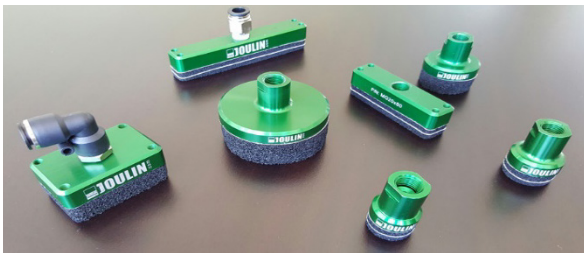


Figure 44: Several examples of small vacuum suction cups. Adapted from: Pneumatic Products (2017).

Figure 45: The Gecko gripper moving a small piece of glass at Automatica 2018. Adapted from: On Robot (2019).

Figure 46: The gripper that is able to mimic the sticking capabilities of a gecko. Reprinted from: On Robot (2019).

4.4 Software

This paragraph will describe some of the available software to control a robot, from a macro-, meso- and micro-level.

The software is vital for a robotic process, without proper software all components are practically useless. The software can also influence the overall capabilities of the robotic process. For example, a robot arm is designed to work on a certain maximum speed. However, a wrong software setting can unintentionally reduce the speed of the robot.

4.4.1 Macro

Every robotic system can be programmed for a specific task, which is how it is done before industry 4.0. A modern general approach to control a robotic process considers artificial intelligence, which is abbreviated as AI. As technology advances, AI is most likely to become the main approach to control robots and it already plays a large role within Industry 4.0. AI can be described as: "a collective term for science that attempts to imitate, or even surpass, human intelligence using computers. Devices become self-learning or self-organising to be able to make decisions without human intervention" (Wassink, 2018, p. 7).

AI looks very promising, but it will take a considerable amount of time before AI is able to imitate human thinking. During a very informal survey among eighteen AI experts, presented in a recent book by Ford (2018), in which they were asked to give their "best guess for a date when there would be at least a 50 percent probability that artificial general intelligence (or human-level AI) will have been achieved" (p. 528). The answers ranged from the year 2029 to 2200 with the mean being 2099. This indicates that it will take at least several decades until AI is able to reach an advanced level.

However, with a defined task and the right programming AI can tackle a problem with a seaming similar way of thinking as a human would do. This is demonstrated by a system called AlphaGo. It was specifically designed to play an ancient Chinese board game named Go. This game has so many possible moves that it is currently impossible to solve using a brute-force method. "The number of possible Go games is extremely large. It is often compared to the number of atoms in the universe (around 10^80), but it is in fact much much larger" (Traill, 2019). After years of teaching and improving the AlphaGo system to play Go, it was able to beat the human world champion (Kohs, 2017). Several months later AlphaZero was released, which beaten its predecessor by teaching itself Go in only three days using a method known as deep reinforcement learning. AlphaZero was generalised so it is also able to learn other games by itself (Silver et al., 2018).

Besides merely digital projects, AI is also interacting with real-world objects as is demonstrated in a research project by the Autodesk AI Lab. In this project, a robot named BrickBot taught itself to assemble a LEGO structure based on the 3D model of the completed structure. Two UR10 robot arms work together and pick each LEGO brick one by one. If needed the brick will be laid on the table and rotated to acquire the desired orientation (Terdiman, 2018). A photo of the two robot arms working together to assemble a LEGO structure can be seen in Figure 47.

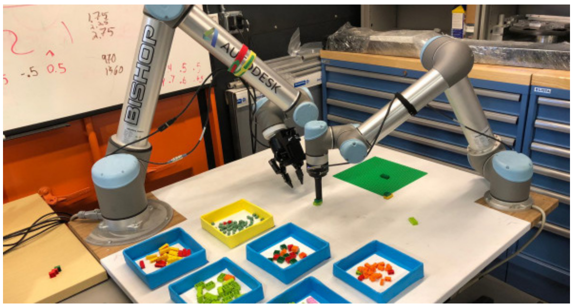


Figure 47: BrickBot autonomously assembling a LEGO structure. Adapted from: Terdiman (2018).

The next step is to apply this level of AI into factories. Yotto Koga Software Architect, Autodesk AI Lab said: "We're looking at ways to make robots easier to use so that we can put these sort of assembly lines together and make them accessible to more people—not just these big companies that have deep resources" (Redshift Video, 2018).

4.4.2 Meso

Two commonly used frameworks are described that are able to create a simulation and provide control commands for a robot arm. These frameworks are HAL-robotics and RoboDK.

Several robot manufacturers provide their own robotic simulation software. Usually, it is only possible to simulate their robots, or the software could even be tailored to merely a single robot. In this research it is undesirable to be bound to a very specific selection of robots, therefore, the use of software supplied by manufacturers is not considered.

HAL-robotics

HAL-robotics framework is a plugin for Rhino and Grasshopper. It is still in its beta phase, the 1.x-beta version is available, and it contains a relatively small database with 47 robots. It is possible to add other robot arms but this need to be programmed, which is a time consuming and difficult process. HAL-robotics provides an option to export the simulation to different common robot languages (HAL-robotics, 2019).

A commercial licence is available for about €1.060 per year for a small company with a team of up to ten people or €2.125 for bigger companies (HAL-robotics, 2019).

RoboDK

RoboDK stands for Robot Development Kit and is a standalone software package that was first released at the end of 2015. Currently, version 3.8 of RoboDK is available and it has a large database that contains 542 robots, of which 420 have 6 DOF or more. RoboDK provides an option to export the simulation to different common robot languages (RoboDK, 2019b). Although RoboDK is a standalone program a plugin for Rhino and Grasshopper is provided (RoboDK, 2018).

A one-time purchased commercial licence costs €2.995 and comes with one year support. Additional support afterwards costs €995 per year (RoboDK, 2019c).

4.4.3 Micro

Zoom in more and reach the robotic operating system and programming languages. Only one system is described since this is one of the most commonly used systems in the field.

Robot Operating System

"Robot Operating Systems (ROS) . . . is currently considered as the main development framework for robotics. . . . ROS has revolutionized the developers community, providing it with a set of tools, infrastructure and best practices to build new applications and robots" (Koubaa, 2016).

ROS is a free to use and open source operating system for robots. A key pillar of ROS are the software libraries that enable developers to easily add the required lines of code for several robotic capabilities, such as navigation. ROS is supported by a large community of amateur enthusiasts, academics and professional robot developers. They are supporting each other online and roughly 46.000 questions have been asked and most are answered (ROS, 2019a). Plus many people keep adding new libraries for others to use and expand upon. ROS is applied in many studies as well as available products, like Baxter, a robot developed by the company Rethink Robotics (ROS, 2019b).

The ROS online documentation page describes that the system is independent of any programming language because it is straightforward to implement ROS in a programming language. It has been implemented in Python, C++, and Lisp, plus beta libraries are made in Java and Lua (Dattalo, 2018). According to a software company that checks more than 1.000.000.000 lines of code each day: Java, Python and C++ are in the top five of the most popular scripting languages since 2015 (TIOBE, 2019). Meaning that ROS is able to provide a system that supports a lot of popular scripting languages and is, therefore, able to satisfy most programmers that need to work with a robotic system, which encourages the use of ROS.

4.5 Other hardware

Depending on the requirements of the robotic process other pieces of hardware might be required. One example is given in this paragraph, which is a slanted table that is part of the fully automated robotic process shown in Figure 48. The process is able to fabricate different types and sizes of wooden elements based on the needs of the customer.

Figure 49 depicts a photo of the slanted table being used to align a single wooden sheet. First, the robot arm picks up the sheet from a stack, but this stack can be slightly shifted so the exact position of the sheet is uncertain. The robot arm drops the sheet on the slanted table, which causes the sheet to slide to the lowest corner of the table. Now the robot knows the exact rotation and location of the sheet on the table.

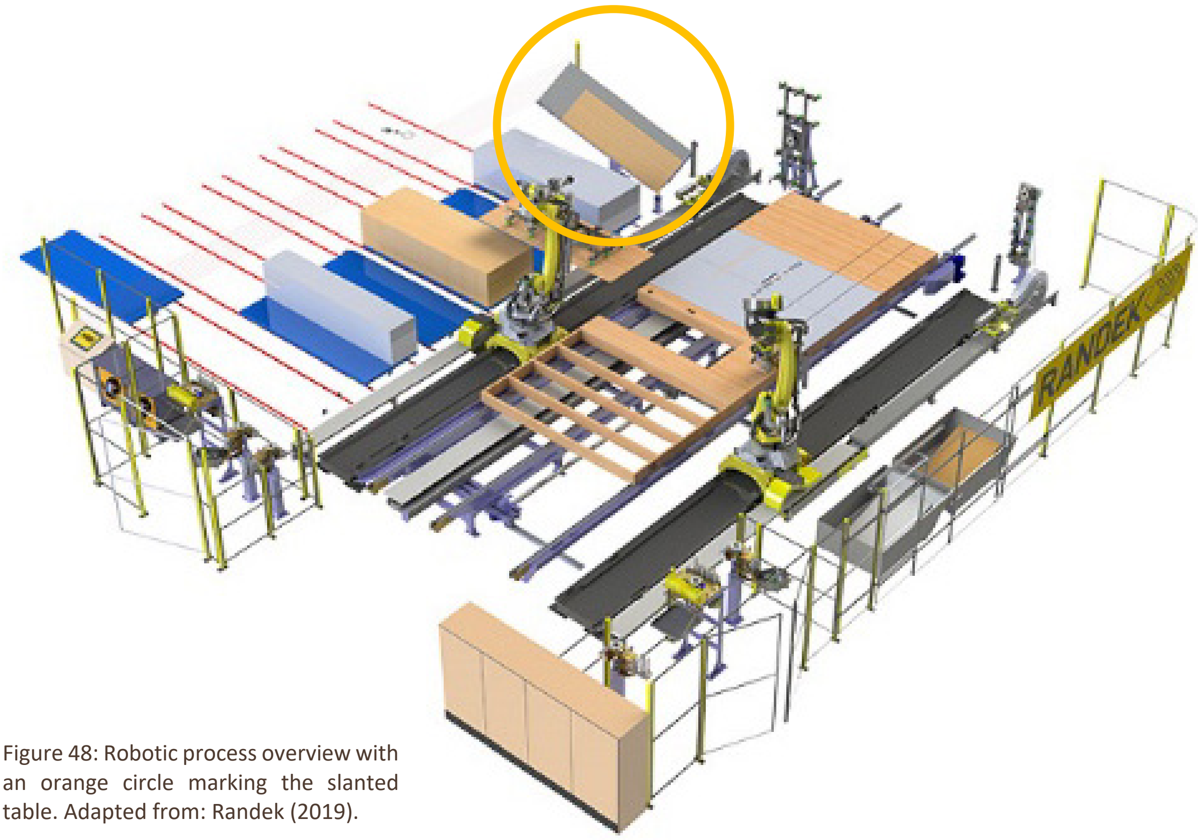


Figure 48: Robotic process overview with an orange circle marking the slanted table. Adapted from: Randek (2019).

Figure 49: Snapshot of a video with the slanted table highlighted. Adapted from: Randek (2019).

4.6 Safety measures

Since the early writings about robotics, safety has always been a concern by some people. Isaac Asimov, a famous science fiction author, came up with the three Laws of Robotics (Asimov, 1950, p. 27):

1. A robot may not injure a human being, or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.

As described in chapter 3 Research framework, the Lean Robotics book considers four principles of which the first one regards to the safety of humans. This principle is "People before robots" (Bouchard, 2017, p. 41) and consists of two parts. The first one regards safety and is titled: "Robotic cells must be safe for humans" (Bouchard, 2017, p. 41). It describes how a safe working environment for humans can be established by creating a risk assessment during the integration phase of a robotic process. Risks can come from many different sources as can be seen in Figure 50.

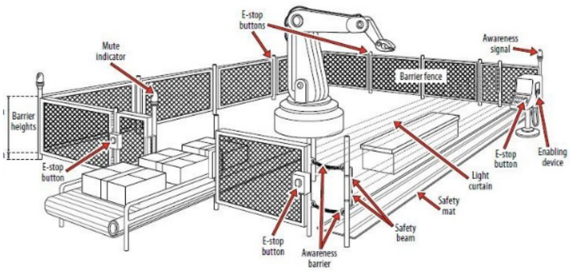


Figure 51: An example of a complete safety system for an industrial robot. Reprinted from: Jani (2016).

On the previous page in Figure 48 several fences can be seen that prevent people from walking into a potentially hazardous area, which reduces the risk of injury. Figure 51 shows more examples of risk-reducing measures, which are:

- Emergency stop button, when pressed the robotic process immediately stops.
- Mute indicator, a light that shines when a certain audio warning is turned off.
- Awareness barrier, notifies a person that he is approaching a hazardous area (USA Department Of Labor, 2013).
- Safety mat, a mat with pressure sensors so the system knows a person approaches a dangerous area.
- Safety beam, a laser that detects when someone enters a dangerous area.
- Light curtain, same as the beam, but with an array of beams.
- Awareness signal, audio signal in case of potential danger.

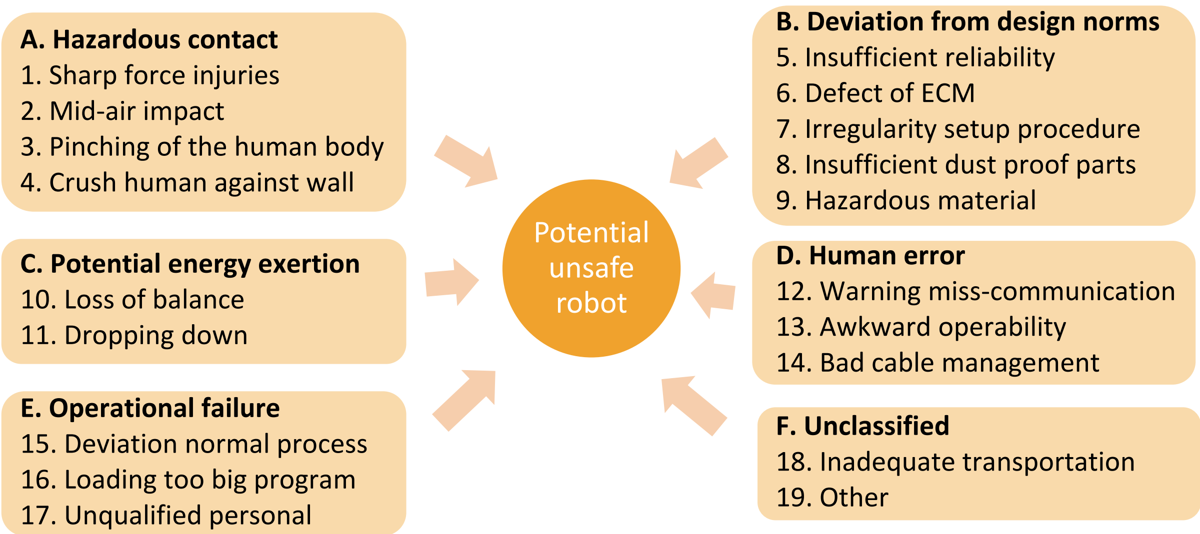


Figure 50: Overview of 19 different hazardous sources placed in six different categories. Adapted from: Jani (2016).

5 Industrial robots vs cobots

5.1 Characteristics and criteria

In this chapter an overview will be given of several available industrial robots and cobots, or collaborative robots, including their characteristics. Followed by a detailed comparison between the industrial robots and cobots based on six different criteria, being: maximum payload, maximum reach, maximum velocity, mass, repeatability and estimated Total Cost of Ownership (TCO). The TCO "is the purchase price of an asset plus the costs of operation" (Twin, 2019).

Besides the overview and detailed comparison, a conceptual level comparison between different robotic assembly systems will be shown.

Figure 52: Robot arm with its six Degrees Of Freedom visualised. Adapted from: Adapted from: Comau (2019a).

Two different tables will show a selection of two types of robot arms, being industrial robots and cobots. The following characteristics of both robot types will be presented:

- Maximum payload [kg], this is one of the most critical characteristics when choosing a robot. If the robot is not able to lift the elements it is going to work with it is pointless to acquire the robot.
- Maximum reach [mm], shows how far a robot can move its end from its base. This is also an important characteristic when a robot needs to be chosen.
- Degrees Of Freedom (DOF) [#], "specific, defined modes in which a mechanical device or system can move. The number of degrees of freedom is equal to the total number of independent displacements or aspects of motion" (Rouse, 2009a). The six DOF of a robot are visualised in Figure 52.
- Mass [kg], the weight of the robot. Very heavy robots can usually only be mounted on the floor, while lighter ones can be mounted to a wall or ceiling.
- Maximum velocity ± [m/s], the maximum speed how fast the robot end can move.
- Maximum velocity [°/s], the maximum speed how fast the joints in the robot arm can rotate.
- Repeatability ± [mm], it is the average deviation measured during a movement sequences between three different points at the highest possible operating speed and while carrying the maximum payload (International Organization for Standardization, 1998).
- Base type [mm], the shape and size of the installation base.
- Base area [m2], the area of the installation base. Some base types are rectangular and others are round, thus it is easier to compare the total area to get an impression of the different base sizes.

5.2 Total Cost of Ownership

Companies are not eager to share the costs of their robots, except with serious clients. Therefore, a more general approach is taken to roughly estimate the Total Cost of Ownership (TCO). The TCO will be based on multiple sources. These sources report prices in USD so they are converted to EUR using the annual average USD to EUR conversion rate of 2018, which is 0,85 (The Statistics Portal, 2019).

Generally speaking, the price of a robot arm is strongly related to its payload capacity and reach. The higher these characteristics are, the higher the price of the robot arm.

5.2.1 Industrial robotic processes

The price of an industrial robot arm ranges from €21.250 to €340.000. However, when taking into account all other aspects required for an industrial robotic process, such as required safety measures, the TCO of the robotic process could double (Thayer, 2017). Another source mentions a different method to calculate the TCO. It states that the cost of an industrial robot arm takes about 25% of the TCO of the robotic process (Motion Controls Robotics, 2017b), meaning the TCO could be four times as high. Taking the average of those two sources means the price of a robot arm should be multiplied by three to calculate the TCO of the robotic process. This results in in a Total Cost of Ownership range from (4 * €21.250 =) €63.750 to €1.020.000 (= 4 * €340.000) for an industrial robotic process. However, the latter seems too high. During a presentation given by ABB, a well-known company with a long track record in robotics, a TCO range of €127.500 to €510.000 over a time span of seven years is mentioned for an industrial robotic process with one robot arm (Gekas & Perera, 2007). This appears to be much more realistic. Another source shows that a TCO for an industrial robotic process of more than one million is extraordinary high. In this article, a TCO of about €130.000 is mentioned (Robot Worx, 2019b). To conclude, the maximum TCO for an industrial robotic process of €510.000 is considered to be a realistic value and will be used in this research.

5.2.2 Cobotic processes

For a collaborative robot arm, or cobot arm, the other required aspects, like safety measures, are lower in comparison to an industrial robot arm. This is because a cobot already has safety systems embedded in its arm. Therefore, the price of the cobot arm does not have to be multiplied by three to calculate the TCO of the cobot process, but with a lower number. An example is given that states an UR5 cobot arm costs about €30.000 and most of the times it results in a TCO of around €43.000 (Motion Controls Robotics, 2017b), which gives a multiplication factor of almost 1,5. The electronic book “Cobots ebook: Collaborative Robots Buyer's Guide” mentions the purchase price of several cobot arms (Robotiq, 2017). These values are converted to EUR, multiplied by 1,5 and added to Table 5, which is discussed later on in paragraph 5.4 Cobots.

5.3 Industrial robots

In order to get an impression of the wide variety of available industrial robot arms that would be suitable for a robotic assembly process, an overview is made. This overview shows the previously described characteristics of industrial robot arms that have a minimum payload capacity of 15 kg. A payload capacity of 15 kg is chosen so the robot would, including the weight of a simple end effector, be able to lift an object of approximately 10 kg.

The number of different available industrial robot arms is very high. In order to get an idea of the characteristics of the available industrial robot arms, in this research it is tried to find the extremes of all industrial robotic characteristics. Such as: highest payload capacity, furthest reach, highest velocity and lowest repeatability. Besides those robots with extreme characteristics, several industrial robots with more general characteristics are added as well in order to give a more generic overview that is presented in Table 4.

Table 4: Overview of several available industrial robots with a minimum payload capacity of 15 kg. The source can be found as a reference in the "Company and source" column. [1]these robots have two operation modes and are set to industrial mode. [2]a source verified 2.300 kg is currently the highest payload capacity available of an industrial robot arm (Robot Worx, 2019c).

Industrial robot	Company and source	Pay-load [kg]	Reach [mm]	DOF [#]	Max velocity ± [m/s]	Max velocity [°/s]	Repeat-ability ± [mm]	Mass [kg]	Base type [mm]	Base area [m2]
TX2-90L[1]	Stäubli (2018)	15	1.200	6	11,10	390-760	0,03	117	262x351	0,092
RS015X	Kawasaki (2018a)	15	3.150	6	19,90	180-610	0,06	545	400x530	0,265
TX2-90[1]	Stäubli (2018)	20	1.000	6	10,90	400-760	0,03	114	262x351	0,092
RS050NFE02	Kawasaki (2018b)	50	2.100	6	13,40	180-360	0,06	555	400x530	0,265
RS080NFE02	Kawasaki (2018c)	80	2.100	6	12,70	165-280	0,06	555	400x530	0,265
AURA-170-2.8[1]	Comau (n.d.)	170	2.790	6	2,00	85-190	0,10	1.615	1.100x900	0,990
ZX300S	Kawasaki (2015)	300	2.501	6	2,50	85-150	0,03	1.400	845x770	0,651
UP400RDII	Yaskawa (2017b)	400	4.909	6	1,70	80-160	0,50	2.500	935x860	0,804
MC470P	Nachi (n.d.)	470	2.771	6	1,90	95-180	0,20	1.620	834x734	0,612
IRB 8700-550	ABB (2019)	550	4.200	6	1,90	95-180	0,05	1.620	834x734	0,612
MH600	Yaskawa (2017a)	600	2.942	6	1,70	80-162	0,30	3.050	786x640	0,492
MPL800 II	Yaskawa (2018b)	800	3.159	4	1,30	65-125	0,50	2.550	640x640	0,410
IRB 8700-800	ABB (2019)	800	3.500	6	1,20	60-115	0,10	460	1175x920	1,081
KR 1000 titan	KUKA (2016a)	1.000	3.601	6	1,00	50-72	0,10	4.740	1350x1360	1,836
MG15HL	Kawasaki (2019b)	1.500	4.005	6		36-80	0,10	6.550	1200x1330	1,596
M-2000iA/2300	FANUC (2017)	2.300[2]	3.734	6	0,60	14-40	0,18	11.000	1540x1480	2,279

5.4 Cobots

In order to get an impression of the available collaborative arms, or cobot arms, that would be suitable for a robotic assembly process an overview is made. This overview shows the previously described characteristics of cobot arms that have a minimum payload capacity of 10 kg. Initially, the intention was to show cobots with a payload capacity of 15 kg, but then there would only be five cobots in the list, which would not give a good representation of the available cobots. Table 5 displays all available cobots that could be found online and that comply with the constraint of a minimum payload capacity of 10 kg.

It is interesting to remark that only three cobots were found with a payload capacity of 15 kg or more, these are 20 kg, 35 kg and 170 kg. Another mentionable fact regards the reach, the cobot with the highest payload capacity, the AURA-170-2.8, also has the longest reach, which is 2.790 mm. The cobot that has the second highest reach, is only able to extend its arm 1.813 mm. A third important fact to state is that no price was found for the AURA-170-2.8, which is the cobot with the highest payload capacity. Compare this cobot with the second one based on payload capacity and it is seen that there is almost a factor five difference in payload capacity. Hence, this will very likely result in a significant increase in the purchase price and TCO of the AURA-170-2.8. A final comment is about TAL BRABO, produced by an Indian company. No information can be found about the quality, which might explain the low price.

Table 5: Available cobots with a minimum payload capacity of 10 kg. The source can be found in the "Company and source" column. [3]these robots have two operation modes and are set to collaborative mode. [4]unlike the purchase price of the other cobots, the purchase price, and therefore the TCO, of this cobot is based on a different source (DUHAN robotics, n.d.).

Cobot	Company and source	Pay-load [kg]	Reach [mm]	DOF [#]	Max vel. ± [m/s]	Max velocity [°/s]	Repeat-ability ± [mm]	Mass [kg]	Base type [mm]	Base area [m2]	TCO ± [EUR]
UR10e	Universal Robots (n.d.-b)	10	1.300	6	1,00	120-180	0,05	33,5	Ø 190	0,028	51.000
KR810	Kassow Robots (n.d.)	10	850	7		225	0,10	23,5	130x130	0,017	
HC10	Yaskawa (2018a)	10	1.200	6		130-250	0,10	47,0	210x210	0,044	61.000
TAL BRABO	Tata Motors (n.d.)	10	750	5		300-600	0,20	80,0	?	?	13.000[4]
i10	Aubo Robotics (n.d.)	10	1.350	6	4,00	150-180	0,05	37,0	Ø 185	0,027	
Speedy 12	Mabi Robotic (n.d.-a)	12	1.250	6		75-275	0,10	35,0	Ø 200	0,031	
TM14	HMK robotics (n.d.-a)	14	1.100	6	1,10	120-180	0,10	32,6	Ø 220	0,038	
LBR iiwa 14	KUKA (2016b)	14	820	7		85-135	0,15	29,9	Ø 230	0,042	
TX2-90L[3]	Stäubli (2018)	15	1.200	6		390-760	0,03	117,0	262x351	0,092	
CR-15iA	FANUC (2018a)	15	1.441	6	1,50		0,03	255,0	340x340	0,116	
TX2-90[3]	Stäubli (2018)	20	1.000	6		400-760	0,03	114,0	262x351	0,092	
CR-35iA	FANUC (2018b)	35	1.813	6	0,75		0,03	990,0	700x700	0,490	111.000
AURA-170-2.8[3]	Comau (n.d.)	170	2.790	6	0,50	85-190	0,10	1.615,0	1.100 x900	0,990	

5.5 Industrial robot and cobot types

To give a visual impression of the robots in Table 4 and Table 5 a photo of each is shown. The robots are ordered on payload capacity from low to high. It is important to state the photos are not to scale.

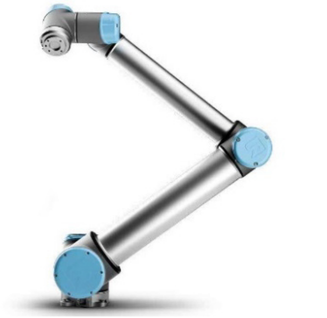


Figure 53: Cobot UR10e (10 kg). Adapted from: Universal Robots (n.d.-b).

Figure 54: Cobot KR810 (10 kg). Reprinted from: ESPS (2018).

Figure 55: Cobot HC10 (10 kg). Adapted from: Yaskawa (2016).

Figure 56: Cobot TAL BRABO (10 kg). Reprinted from: TAL Manufacturing Solutions (2017).

Figure 57: Cobot i10 (10 kg). Adapted from: Aubo Robotics (n.d.).

Figure 58: Cobot Speedy 12 (12 kg). Adapted from: Mabi Robotic (n.d.-b).

Figure 59: Cobot TM14 (14 kg). Adapted from: HMK robotics (n.d.-b).

Figure 60: Cobot LBR iiwa 14 (14 kg). Adapted from: KUKA (2016b).

Figure 61: Cobot CR-15iA (15 kg). Adapted from: FANUC (2018a).

Figure 62: Industrial robot RS015X (15 kg). Reprinted from: Kawasaki (2019a).

Figure 63: TX2-90L (15 kg), with industrial and cobot operation mode. Adapted from: Stäubli (2019).

Figure 64: TX2-90 (20 kg), with industrial and cobot operation mode. Adapted from: Stäubli (2019).

Figure 65: Cobot CR-35iA (35 kg). Adapted from: FANUC (2018b).

Figure 66: Industrial robot RS050NFE, 50kg, and RS080NFE (80 kg). Reprinted from: Kawasaki (n.d.-b)

Figure 67: AURA-170-2.8 (170 kg), with industrial and cobot operation mode. Adapted from: Comau (n.d.).

Figure 68: Industrial robot ZX300S (300 kg). Reprinted from: Kawasaki (n.d.-c).

Figure 69: Industrial robot UP400RDII (400 kg). Reprinted from: Yaskawa (2018c).

Figure 70: Industrial robot MC470P (470 kg). Reprinted from: Nachi (2016).

Figure 71: Industrial robot MH600 (600 kg). Reprinted from: Yaskawa (2017a).

Figure 72: Industrial robot MPL800 II (800 kg). Reprinted from: Yaskawa (2018b).

Figure 73: Industrial robot IRB 8700-550, 550 kg, and IRB 8700-800 (800 kg). Adapted from: ABB (2019).

Figure 74: Industrial robot KR 1000 titan (1.000 kg). Reprinted from: KUKA (2016a).

Figure 75: Industrial robot MG15HL (1.500 kg). Reprinted from: Kawasaki (n.d.-a).

Figure 76: Industrial robot M-2000iA / 2300 (2.300 kg). Reprinted from: Robot Worx (2019a).

5.6 Comparison

5.6.1 Types comparison

In this subparagraph, a comparison between the industrial robots in Table 4 and the cobots in Table 5 is made. The comparison is based on six different criterion, being: maximum payload, maximum reach, maximum velocity, mass, repeatability and estimated Total Cost of Ownership (TCO). For each criterion a graph is made that shows the average and maximum value. Except for the repeatability, the minimum values are shown as well since a lower repeatability is usually more desirable.

Figure 77 makes it clearly visible the maximum payload of industrial robots is far higher than cobots. As mentioned before, only three cobots were found with a higher payload capacity of 15 kg.

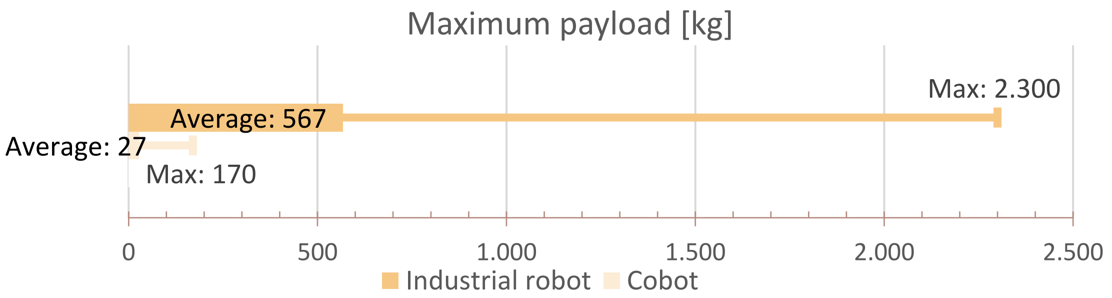


Figure 77: Graph showing the maximum payload of the industrial robots in Table 4 and the cobots in Table 5.

Figure 78 shows the maximum reach of industrial robots is close to five meters. This is almost three times as much compared to cobots, which are, except for one, not able to reach beyond 1,8 m.

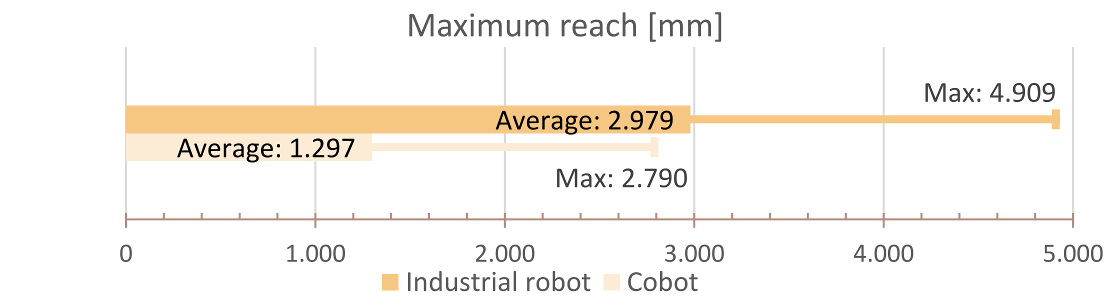


Figure 78: Graph showing the maximum reach of the industrial robots in Table 4 and the cobots in Table 5.

Figure 79 shows the maximum velocity, or movement speed, of the end of the robot arm. This is significantly higher for industrial robots compared to cobots. This is not due to a lack of mechanical capabilities in the cobot arm but caused by safety features installed in a cobot. These safety features ensure a safe working environment for a human and therefore put a restriction on the movement speed of the cobot.

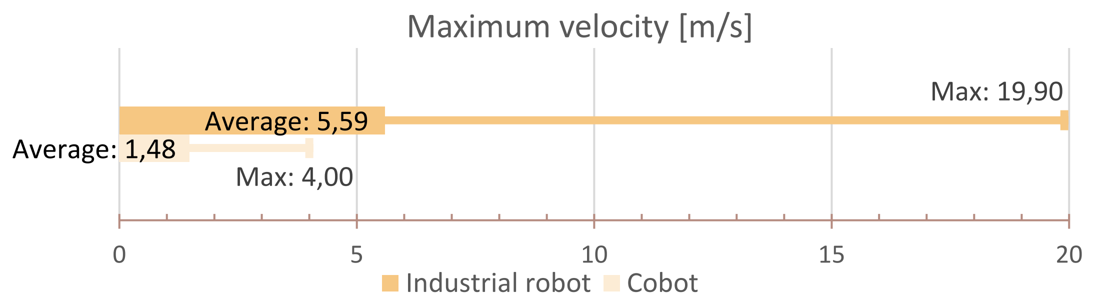


Figure 79: Graph showing the maximum velocity of the industrial robots in Table 4 and the cobots in Table 5.

Industrial robots and cobots can both be extremely precise, which is presented by the same minimal repeatability of 0,03 mm that can be seen in Figure 80. Some applications do not require extreme precision and there are robots available that provide this. Less precision required for a robot enables the use of less expensive parts, which reduces the purchase price of the robot.

The repeatability is not the same as the accuracy, the latter is usually worse in comparison to the first and differs over the working area. This effect is caused by link kinematics, which is the addition of small unintended deviations of the motors at each joint. Nevertheless, repeatability is a good indication of the accuracy (Yaskawa, n.d.).

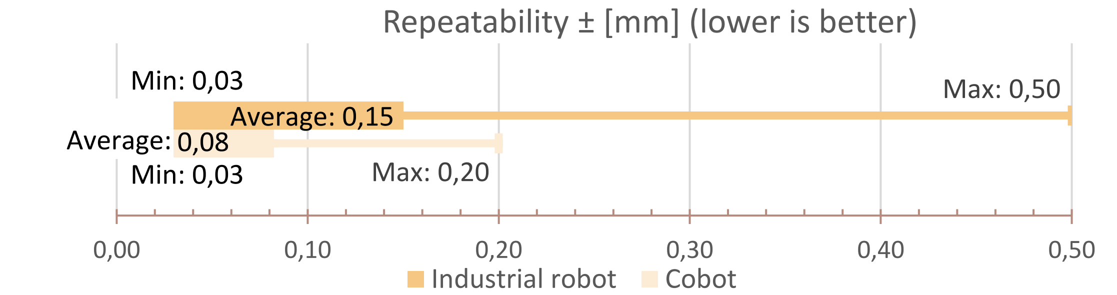


Figure 80: Graph showing the repeatability of the industrial robots in Table 4 and the cobots in Table 5.

It is interesting to notice how the maximum payload in Figure 77 looks, except for the numbers, almost exactly the same as the graph of the mass in Figure 81. This is logical since, generally speaking, the payload capacity of a machine is strongly related to its mass. This implies that a higher payload capacity results in a heavier robot.

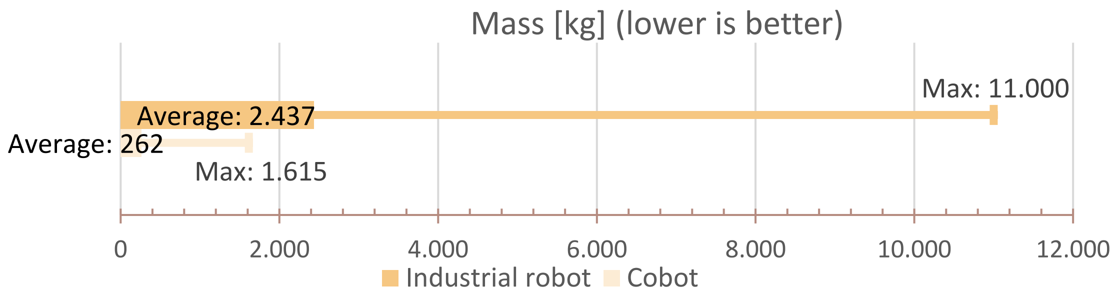


Figure 81: Graph showing the mass of the industrial robots in Table 4 and the cobots in Table 5.

Figure 82 shows the rough estimation of the TCO of industrial robots and cobots, which was explained in paragraph 5.2 Total Cost of Ownership. As mentioned before, the cobot with the highest payload capacity is extremely likely to be the most expensive one. However, this purchase price of this cobot is not considered in the cobot table and, therefore, not in this figure. Meaning the average and maximum value of cobots could be much higher.

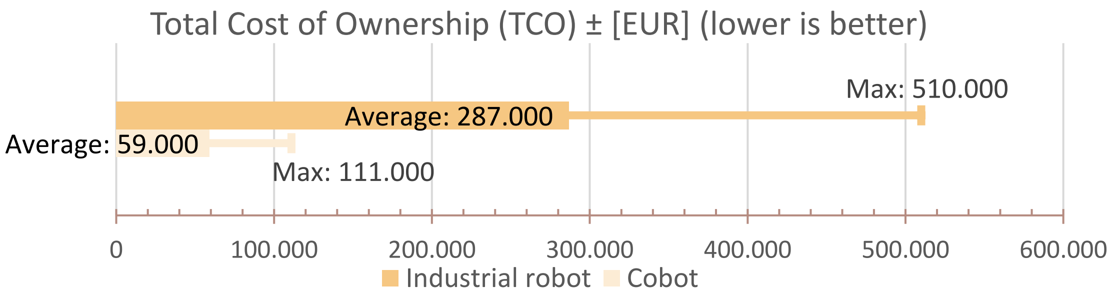


Figure 82: Graph showing the estimated Total Cost of Ownership of industrial robots and cobots in general.

5.6.2 Conceptual comparison

When zooming out from the individual robots to a more conceptual level, some interesting differences between different assembly concepts can be noticed.

A manual or completely human assembly process is highly flexible and can handle many different products, but the number of products produced is very low. On the opposite of the assembly processes spectrum sits automated assembly, meaning only robots. Automated assembly is inflexible and is not able to handle different products, but the product quantity is very high. Figure 83 presents a visualisation of the strengths and weakness of these two concepts, including an in-between variant, being a hybrid assembly where humans and robots work together (ElMaraghy, 2009).

During a workshop given by ABB named "Workspace Safety in Industrial Robotics: trends, integration and standards" (Matthias, 2014), the graph in Figure 84 that is placed on the next page was presented. It shows that for companies, whom have a low to medium production volume and make a product that is not very expensive per unit, it is advisable to aim for an automation solution that is either based on human-robot collaboration or robotic automation.

The previously shown process with the slanted table is part is part of a bigger fixed automation process. This big fixed automation process is presented in Figure 85. In this fixed automation process different wooden components are produced for either a wall, floor or roof Randek (2019).

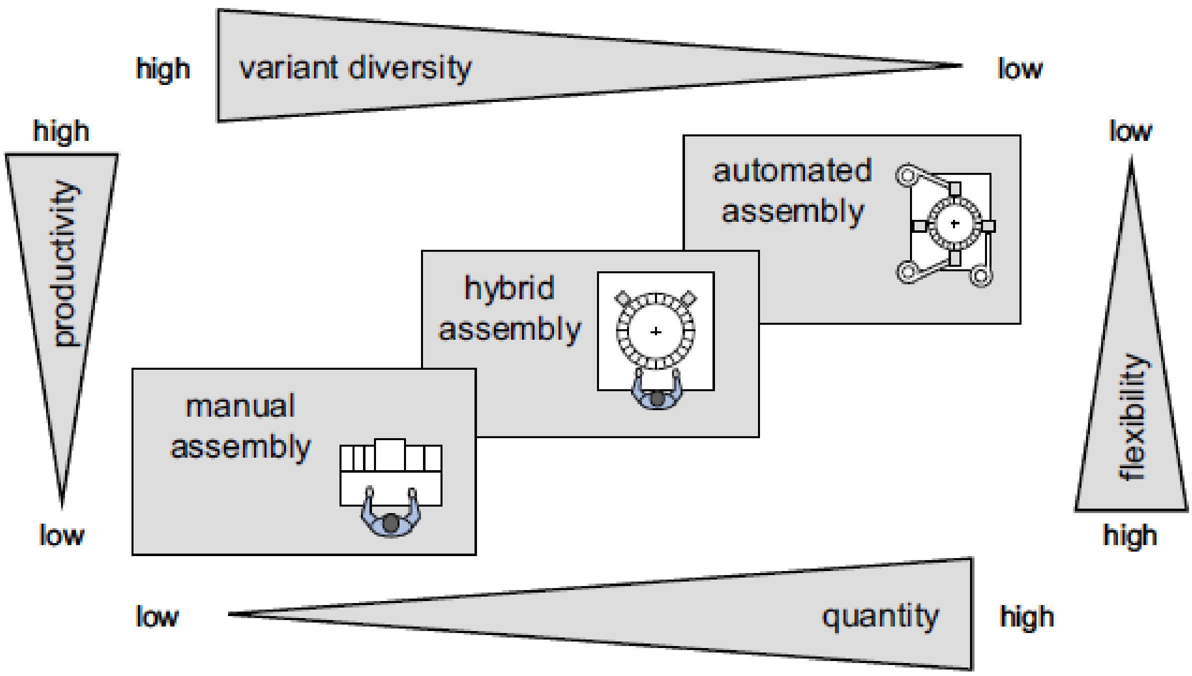


Figure 83: Utilization areas for manual, hybrid and automated assembly concepts. Reprinted from: ElMaraghy (2009).

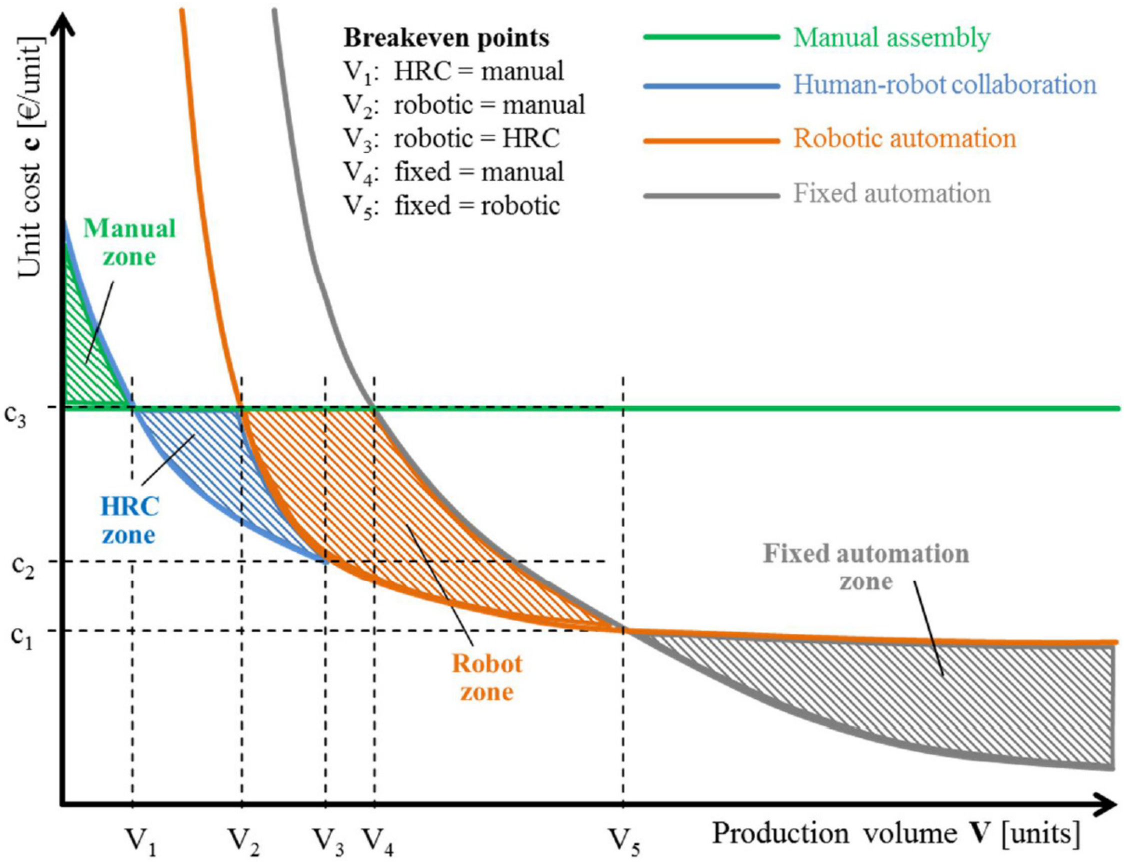


Figure 84: Graph showing different optimal zones for four types of assembly based on the cost per unit over the production volume. Reprinted from: Matthias (2014).

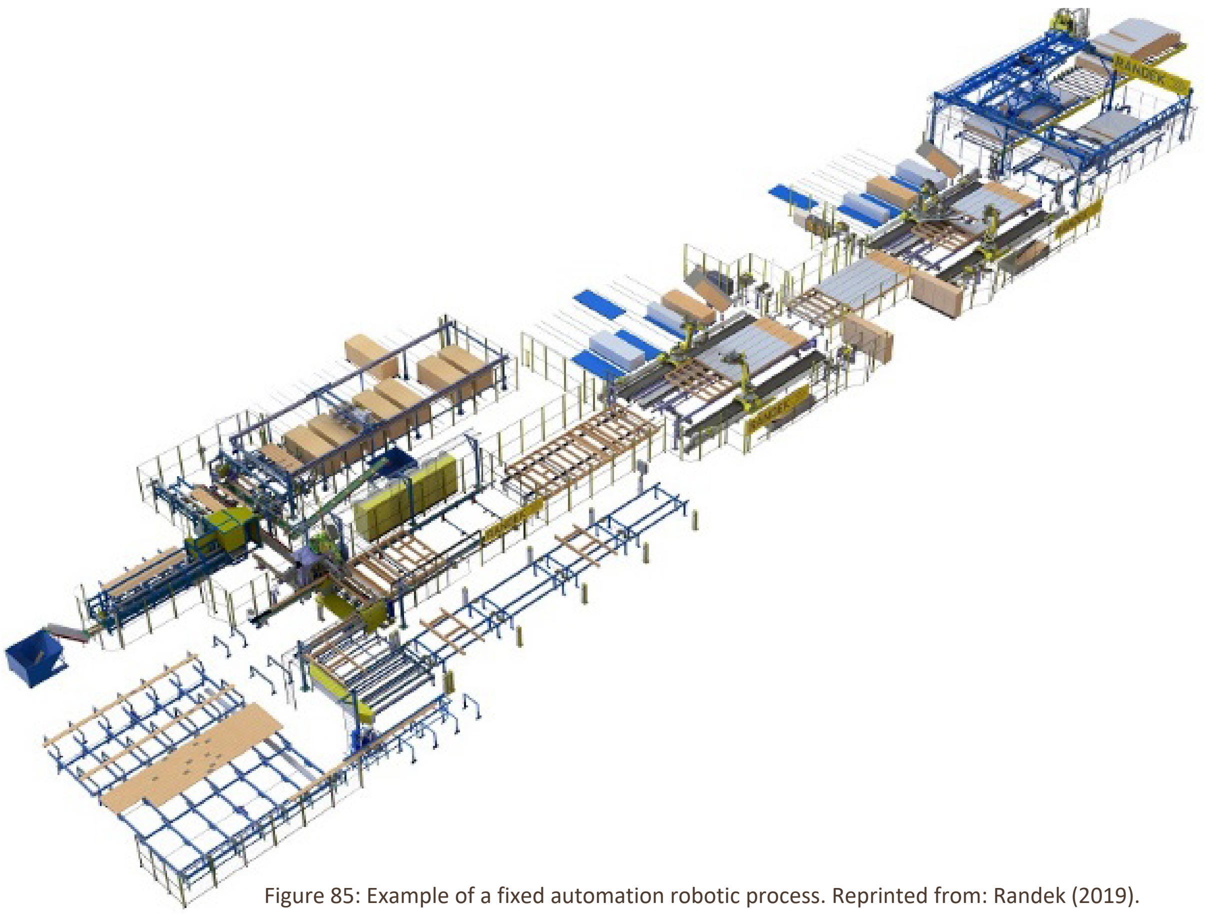


Figure 85: Example of a fixed automation robotic process. Reprinted from: Randek (2019).

6 Design For Automatic Assembly (DFA2)

6.1 DFA2 method

Simplifying an assembly process will enable the process to be completed quicker and easier. In order to simplify this process, the component should be redesigned from an automatic assembly point of view. In this chapter a method will be described that provides redesign guidelines, which will help to become aware of all different factors that influence an automatic assembly process.

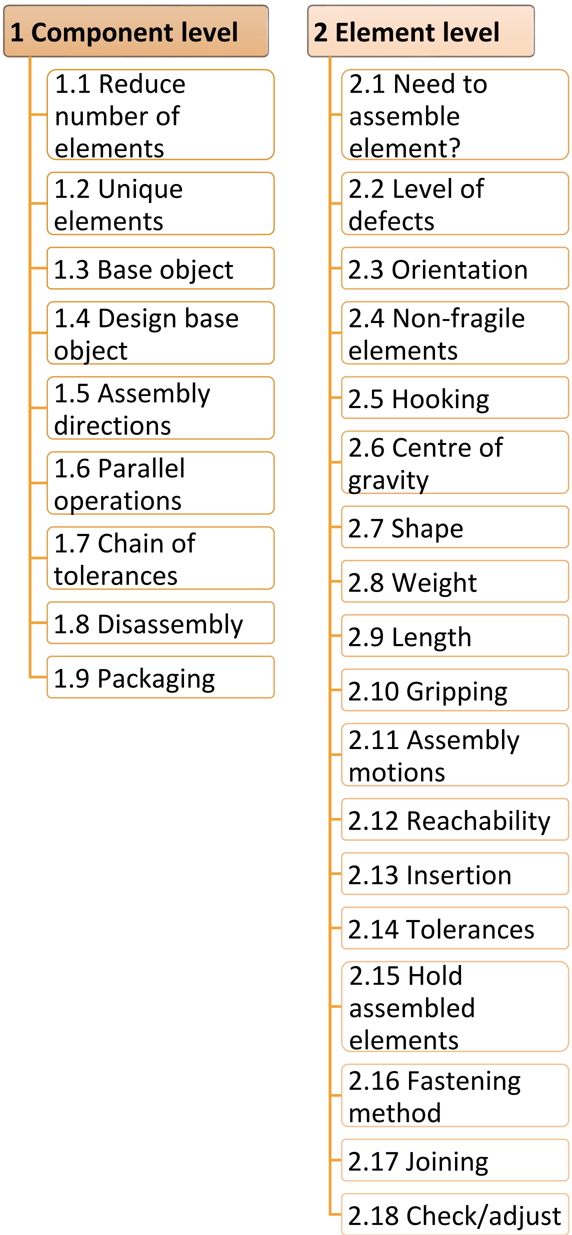


Figure 86: Structure of the Design For Automatic Assembly (DFA2) method. Adapted from: Eskilander (2001).

A method to improve a component for automatic assembly is described in a doctoral thesis written by Eskilander (2001). The industry, or field, in which this method is positioned is the Design For Assembly (DFA) field and the method is called DFA2, which refers to the Design For Automatic Assembly method. The main goal of the DFA2 method is to design a component “that is as simple (non-complex) as possible, which, in turn, means that the simplest possible assembly process can be used” (Eskilander, 2001, p. 154). DFA2 provides design, or redesign, guidelines based on more than 30 years of knowledge in the Design For Assembly field.

The DFA2 method is chosen because of three reasons. Firstly, unlike most other available DFA methods, it explains “how to avoid the features in the product where the evaluation indicates problems” and it “is focused on describing how to avoid several known assembly problems” (Eskilander, 2001, p. 60). Secondly, on an academic level it “constitutes one of the more comprehensive collections of design rules . . . aimed at automatic assembly” (Eskilander, 2001, p. 65). Lastly, it is established in collaboration with 18 companies in the industry and the DFA2 method “fulfils the needs expressed by engineers in [the DFA] industry” (Eskilander, 2001, p. 66).

Figure 86 shows the structure of the DFA2 method. Two different levels are considered, each with their own steps and (re)design guidelines. The first is the component level, or product level as to how it is called in the doctoral thesis. The second level focusses on the element level, called part level in the thesis. Besides the (re)design guidelines, every step contains certain evaluation criteria and is rewarded using a point scheme rated from an automatic assembly perspective: 9 points are given to the best solution, 3 for a solution that lacks complete satisfaction, and 1 for an unwanted solution (Eskilander, 2001). The subparagraphs on the following pages will elaborate on each step of the DFA2 method.

6.2 Component level

Each step on the component level will help to evaluate the component based on certain criteria and provide the previously mentioned 9, 3 or 1 rating. Some steps also provide a visualisation of this rating.

6.2.1 Reduce number of elements

Reducing the number of elements is very important. Too many elements have a negative influence on simplifying the assembly process (Eskilander, 2001).

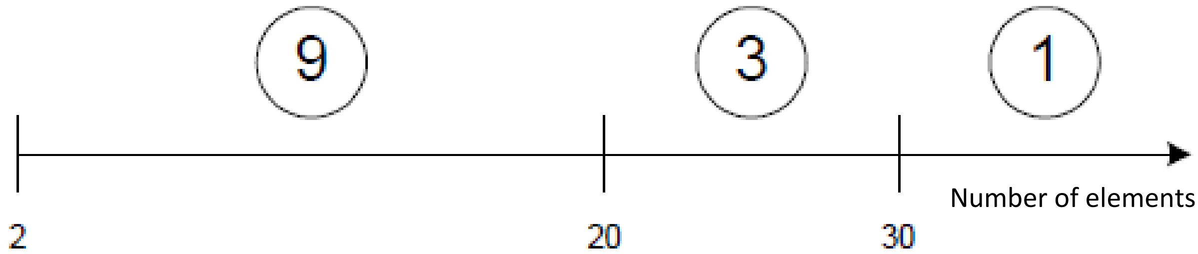


Figure 87: Evaluation rating regarding the number of elements. Adapted from: Eskilander (2001).

6.2.2 Unique elements

Applying mostly the same elements improves the assembly process. For example, using the exact same screws could reduce the need for different robotic tools (Eskilander, 2001).

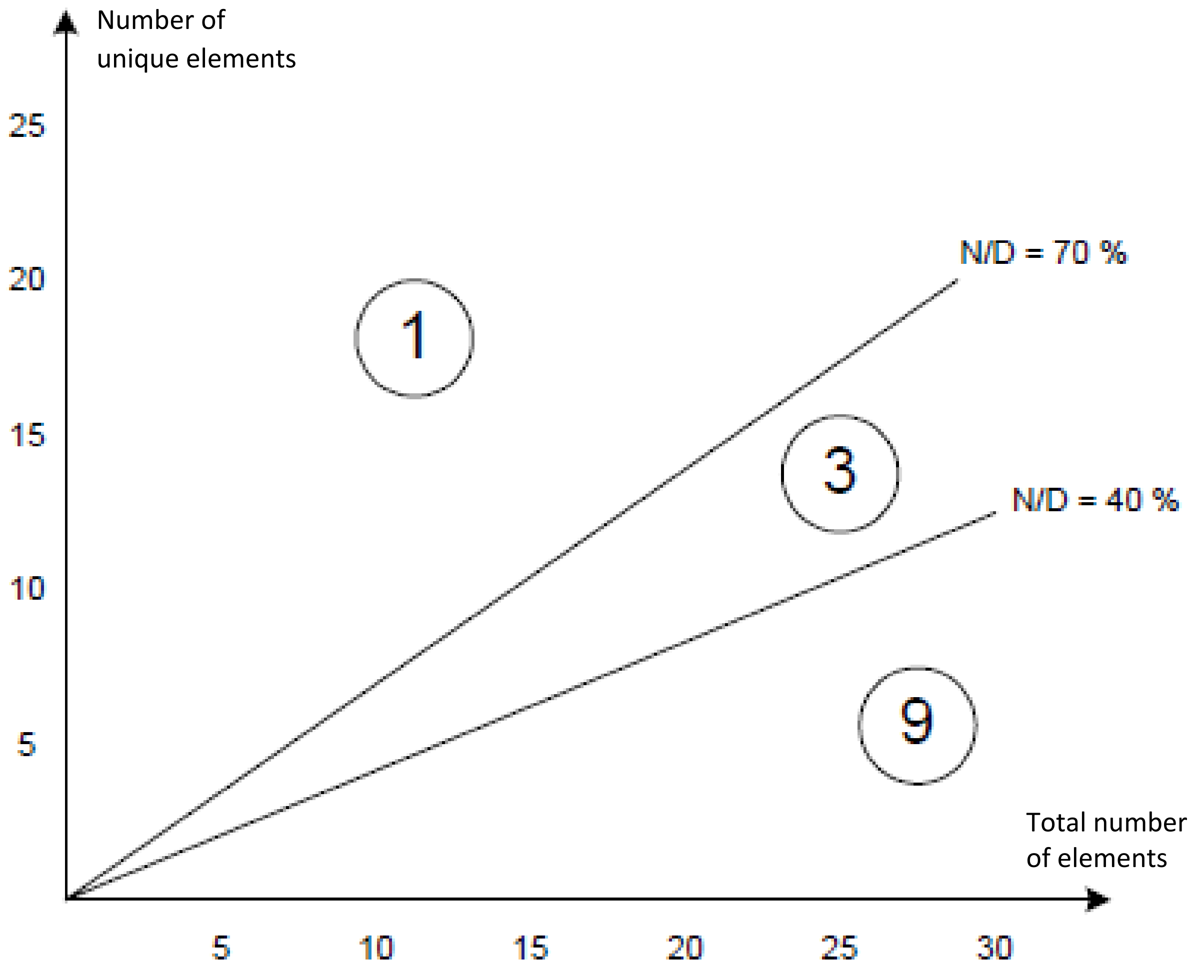


Figure 88: Evaluation of the number of unique elements divided by the total number. Adapted from: Eskilander (2001).

6.2.3 Base object

Having a base object in which the other elements can be assembled in improves the assembly process considerably. The base object can act as a fixture for the other elements, thus reducing the number of assembly fixtures required (Eskilander, 2001).

Table 6: Evaluation with or without a base object. Adapted from: Eskilander (2001).

The base object is the first element that the rest of the assembly can proceed from. All assembly operations are performed on the base object, which leads to simple fixtures and few assembly directions.	Points
With base object	9
Without base object	1

6.2.4 Design base object

There are multiple viewpoints to consider when designing the base object, just to name two: the base should not be flipped during the assembly process since extra equipment would be required plus any elements that are already assembled might shift; the ideal motions required should be horizontally and vertically (Eskilander, 2001).

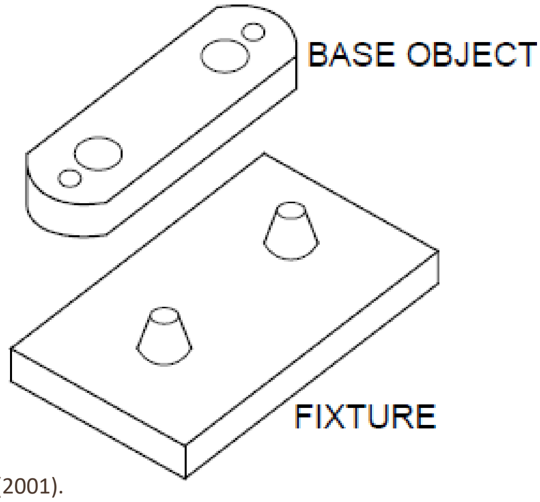


Figure 89: Example of a base object and fixture. Adapted from: Eskilander (2001).

Table 7: Evaluation of base object design. Adapted from: Eskilander (2001).

Design base object for easy fixture	Points
The base object is designed in a way that no further fixture, besides for the base object itself, is needed for the rest of the assembly. The base object does not need repositioning during assembly. One assembly direction.	9
Assembling the module requires multiple fixtures that each has only one fixed position. The base object has to be reoriented or transferred between fixtures during assembly.	3
Assembling the module requires one or multiple fixtures that have several movable positions. The base object must be transferred between and/or repositioned in the fixtures during assembly.	1

6.2.5 Assembly directions

The component should be designed so it can be assembled from top to bottom since this is the easiest assembly direction (Eskilander, 2001).

Table 8: Evaluation of assembly directions. Adapted from: Eskilander (2001).

Assembly directions, totally in the whole component	Points
One assembly direction into a fixed base object.	9
Two assembly directions into a fixed base object (alternatively one assembly direction in a movable base object with two different fixed positions).	3
Three or more assembly directions into a fixed base object (alternatively assembly in a movable base object with several different fixed positions).	1

6.2.6 Parallel operations

If certain elements can be assembled simultaneously, or in parallel, it can lead to a significant reduction of the total assembly time (Eskilander, 2001).

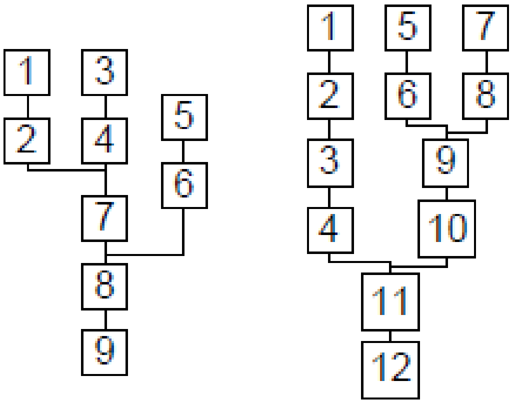


Figure 90: Two examples of different assembly sequences. The Left one has 7/9 parallel operations and the right has 10/12 parallel operations. Adapted from: Eskilander (2001).

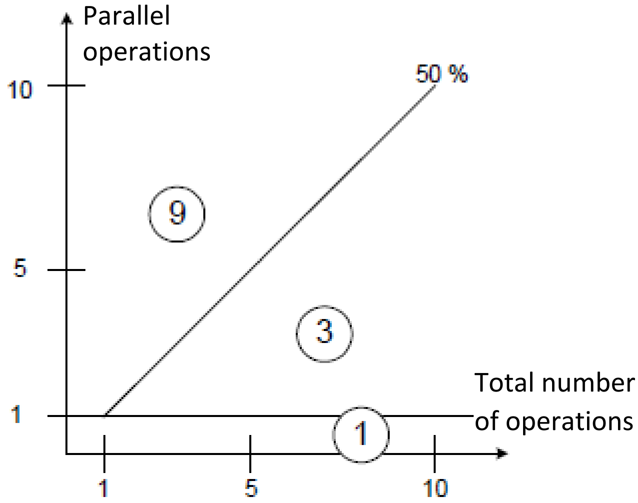


Figure 91: Ratio of parallel operations. Adapted from: Eskilander (2001).

6.2.7 Chain of tolerances

All tolerances must suit the assembly systems. For example, the gripper has a certain tolerance or accuracy, if the tolerances are smaller than this accuracy the assembly process is likely to come to a standstill (Eskilander, 2001).

Table 9: Evaluation chain of tolerances. Adapted from: Eskilander (2001).

Chains of tolerances should be minimised to have a more reliable assembly process.	Points
No chains of tolerances significant for the assembly process. Only the tolerance of each individual element is significant.	9
There are chains of two tolerances significant for the assembly process in the component.	3
There are chains of three or more tolerances significant for the assembly process in the component.	1

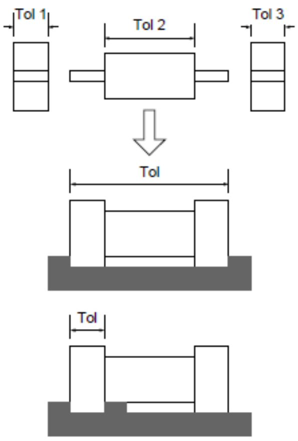


Figure 92: Avoid chains or sums of tolerances. Reprinted from: Eskilander (2001).

6.2.8 Disassembly

"No evaluation criterions for disassembly were found directly applicable or industrially verified" (Eskilander, 2001, p. 164). However, several suggestions are given to improve disassembly, to give an example: it is important to apply standard fastening elements. When fewer types of screws are applied the number of tools required will be reduced.

6.2.9 Packaging

Just as is the case with the previous step, disassembly, the same applies to packaging. There are no criteria available for this step. One recommendation is mentioned, which states the components should be transported while minimizing the amount of space and material required (Eskilander, 2001).

6.3 Element level

Each step on the element level will help to evaluate every element based on certain criteria and provide the previously mentioned 9, 3 or 1 rating. Some steps also provide a visualisation of this rating.

6.3.1 Need to assemble element?

Assembly should be avoided if possible in order to improve the assembly process. One approach to reducing the number of elements is by asking three questions, which are presented on the next page. Another approach, that unlike the three questions, takes the functionality of the element into account resulting in eliminating useless elements. An overview of the latter approach is shown in Figure 93.

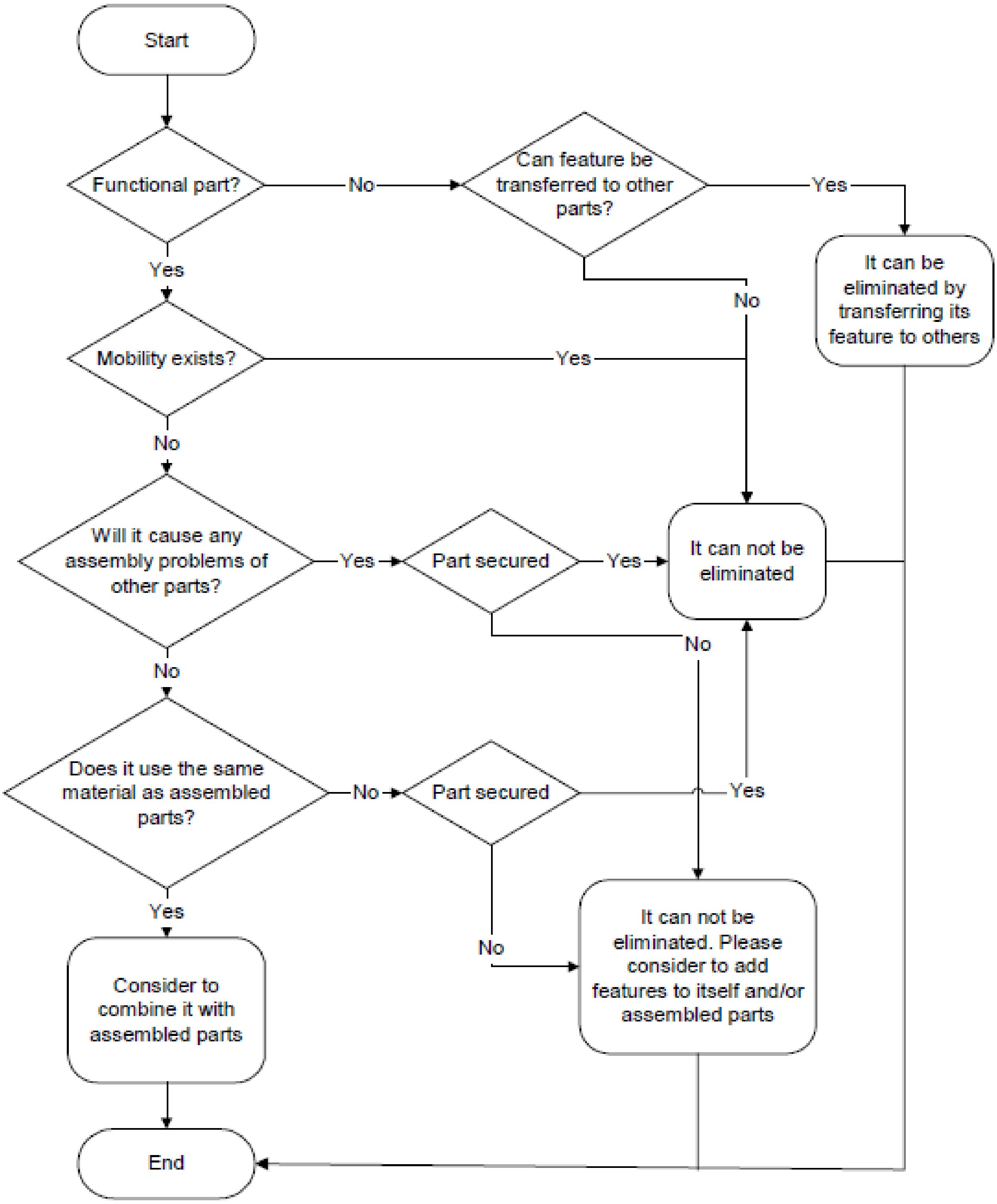


Figure 93: Flowchart of the approach that takes the functionality of an element, referred to as part in the source, into consideration. Reprinted from: Eskilander (2001).

To come back to the three questions, consider them for each element (Eskilander, 2001):

1. Does the element move, relative to other already assembled element during normal use of the finished component?
2. Does the element have to be of other material than already assembled elements, or isolated from them?
3. Does the element has to be separate from already assembled element because assembly or disassembly otherwise is impossible?

Table 10: Evaluation to determine if the element needs to be assembled. Adapted from: Eskilander (2001).

The three questions have to be answered for evaluation.	Points
The element has reasons for being separate ("yes" to at least one question).	9
The element should be eliminated/integrated ("no" to all three questions), but the element is still a separate element in the component.	1

6.3.2 Level of defects

Defects in all elements should be kept at a minimum, e.g.: rate of defect for screws should be lower than 0,1%, meaning a single defect per 1000. The same applies to the geometry of elements, they should not contain defects that might cause the process to stop unintendedly (Eskilander, 2001).

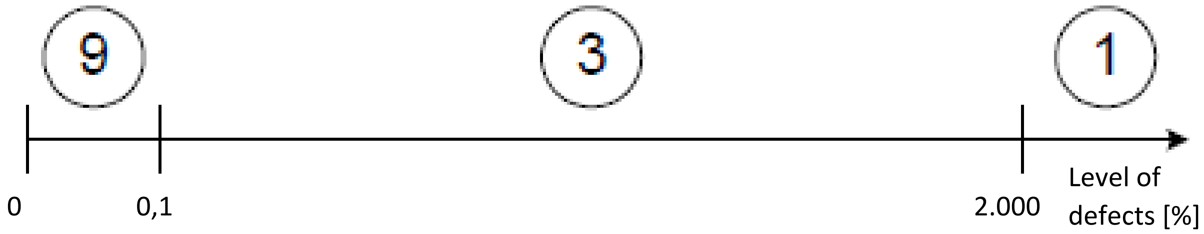


Figure 94: Level of defects. Adapted from: Eskilander (2001).

6.3.3 Orientation

When an element is delivered in the correct orientation it saves time, since it is not required to re-orientate the element. It might be required to place the elements in a fixture of some sort (Eskilander, 2001).

Table 11: Evaluation of correct orientation of an element. Adapted from: Eskilander (2001).

Orientation. If an element could be delivered oriented, cost and uncertainty in the process would be eliminated.	Points
No need for re-orientation of the element.	9
Element is partly orientated but needs final orientation.	3
Element orientation needs to be re-created.	1

6.3.4 Non-fragile elements

Most elements are not designed to be handled in a robotic process and therefore are at risk of breaking during a robotic assembly process. Two other design guidelines that are worth mentioning are: “Surface tolerances for parts [also referred to as elements], which an assembly system does not have to consider, should be avoided when possible” and “High friction for a part [or element] can be a drawback since e.g. gliding driven by gravity will be difficult” Eskilander (2001, p. 170).

Table 12: Evaluation of non-fragile elements. Adapted from: Eskilander (2001).

Feeding often requires non-fragile elements	Points
Element is not fragile.	9
Element can be scratched, which is not acceptable.	3
Elements cannot fall without deforming.	1

6.3.5 Hooking

Elements should not be able to tangle up with other elements when stored in bulk (Eskilander, 2001).

Table 13: Evaluation to consider unintended hooking of elements within each other. Adapted from: Eskilander (2001).

State during feeding, hooking: There should be no risk of elements hooking into each other for example in a bulk vibration feeder.	Points
Elements cannot hook into each other and tangle up.	9
Elements can hook into each other and tangle up.	1

6.3.6 Centre of gravity

To separate elements using a feeder its best to have an eccentric centre of gravity (Eskilander, 2001).

Table 14: Evaluation of the centre of gravity of an element from a feeder point of view. Adapted from: Eskilander (2001).

Centre of gravity for the element should be positioned for use in feeding. Drop the element repeatedly on a table to determine its state of rest. Simple orientation often means reliable and cost-effective feeding.	Points
Element has a stable state of rest and orients itself with the correct side upwards.	9
Element has a stable state of rest but orients itself with the wrong side upwards.	3
Element has an unstable state of rest and orients itself with different sides upwards.	1

6.3.7 Shape

It would be best if the shape of an element is symmetrical and has as few critical orientations as possible so it is easiest to orientate. To evaluate this aspect the symmetry is considered around two different axes, being the α and β symmetry. The latter regards the rotation about the axis over which the element is moved during insertion. The first, the α symmetry, considers the rotation about an axis perpendicular to this axis (Eskilander, 2001).

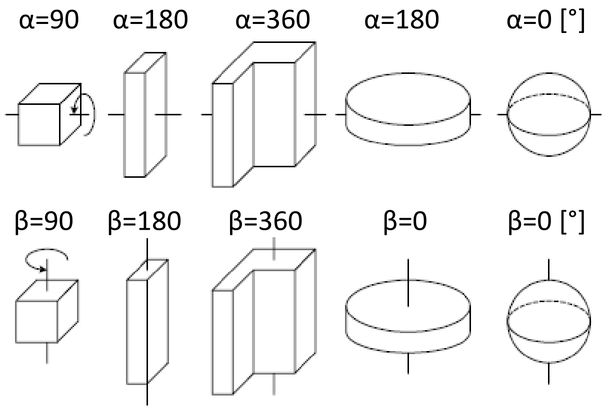


Figure 95: Examples of α and β symmetry of five different elements. Adapted from: Eskilander (2001).

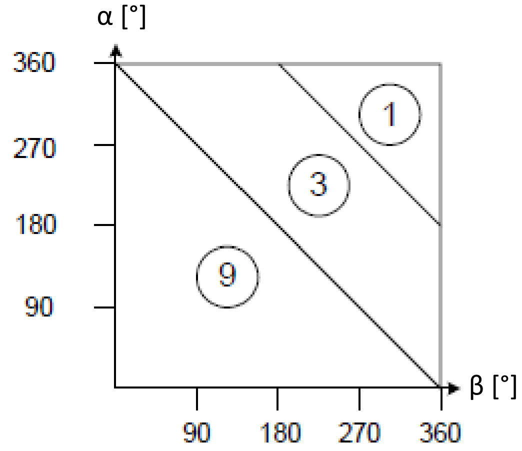


Figure 96:Symmetry. Adapted from: Eskilander (2001).

6.3.8 Weight

It is advised to try to keep the mass of an element as low as possible. However, if the mass is to low, complications might arise due to adhesive forces (Eskilander, 2001).

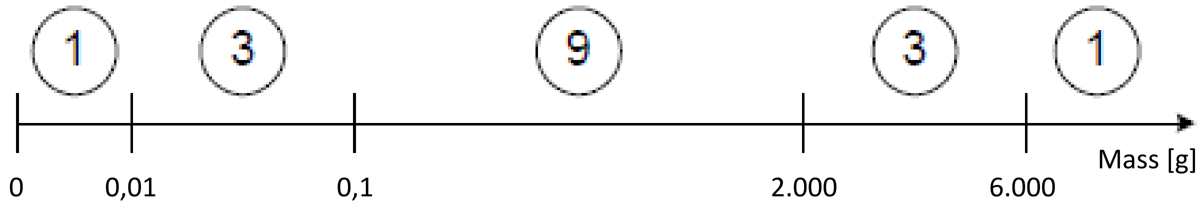


Figure 97: Visualisation of rating regarding the mass of an element. Adapted from: Eskilander (2001).

6.3.9 Length

The length of the elements has an influence on the chosen system required for a robotic system, like the gripper and robot arm (Eskilander, 2001).

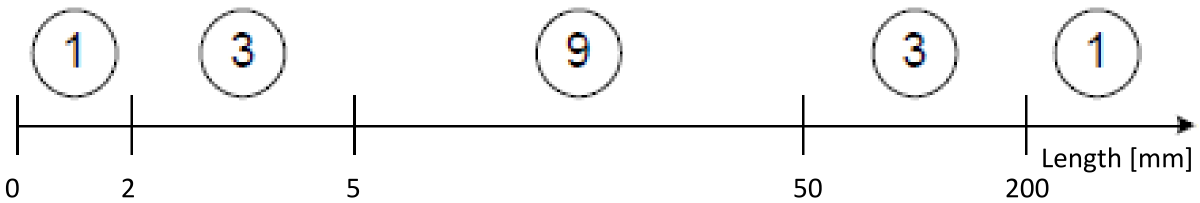


Figure 98: Visualisation of rating regarding the length of an element. Adapted from: Eskilander (2001).

6.3.10 Gripping

A design rule is to ensure all elements are easy to grip, which is influenced by the shape, size and material of the element. The ease of the automatic assembly process will be improved when most elements can be gripped using the same gripper (Eskilander, 2001).

Table 15: Evaluation to consider for gripping of elements. Adapted from: Eskilander (2001).

Gripping is simplified if there are defined surfaces with determined geometry for use. Soft elements, e.g. plastics and rubber, are difficult to grip with a mechanical gripper since they can deform due to the forces in the gripper.	**Points**
Element has surfaces for gripping and can be gripped with the same gripper as the previous element.	9
Element has surfaces for gripping but requires a new, unique gripper that could not be used for the previous element. Element has surfaces for gripping and can use the same gripper as used earlier, but not for the previous element.	3
Element has no surfaces for gripping or is flexible.	1

6.3.11 Assembly motions

It should be possible to manually assemble every element with one hand. When this is achievable it is very likely automatic assembly is possible. The preferred assembly motion is pushing one element within another (Eskilander, 2001).

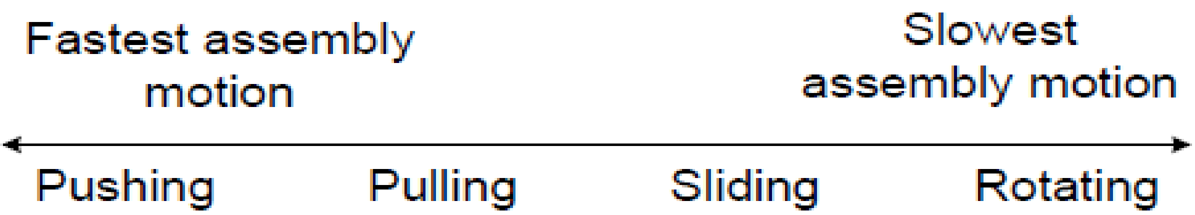


Figure 99: Four different assembly motions compared based on their assembly speed. Reprinted from: Eskilander (2001).

Table 16: Evaluation of different assembly motions. Adapted from: Eskilander (2001).

Assembly motions (during insertion) will be faster, the simpler they are.	**Points**
Assembly motion consists of a pressing motion with one element being assembled to already assembled elements.	9
Assembly motion consists of further motions than pressing motion with one element.	3
Assembly motion is an operation with multiple movable elements that simultaneously are assembled to already assembled elements with other motions than pressing motion.	1

6.3.12 Reachability

Enough space is required for the robot arm and gripper to position each element in its desired place. Extra room for special tools required for operations like screwing need to be taken into account. An extra design suggestion mentions increasing the number of screwdrivers so multiple screws can be inserted simultaneously (Eskilander, 2001).

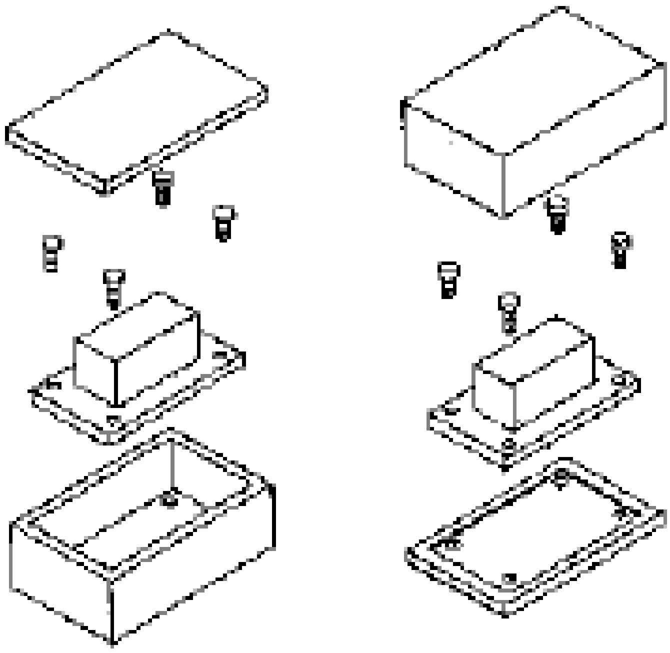


Figure 100: Example on how to simplify reachability. The original on the left and on the right is the improved version. Reprinted from: Eskilander (2001).

Table 17: Evaluation of reachability. Adapted from: Eskilander (2001).

Reachability for assembly operation should not be limited. All elements should be inserted in the same direction.	**Points**
No restrictions for reaching when fitting the element.	9
Reachability is limited. Other assembly direction than the previous element.	3
Reachability is limited and requires special tools or grippers to perform the assembly operation. Other assembly direction than the previous element.	1

6.3.13 Insertion

Some design rules to consider in order to simplify an insertion procedure: chamfer the destination geometry, apply guiding surfaces and minimise friction between elements. Plus some recommendations regarding screws: use long screws with a conical point, chamfer the holes, and avoid narrow holes (Eskilander, 2001).

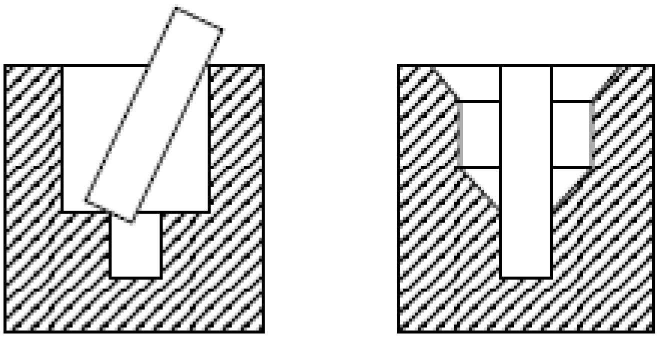


Figure 101: Chamfered geometry can help to improve insertion during automated assembly. The original geometry on the left and the improved on the right with guiding surfaces and chamfers. Reprinted from: Eskilander (2001).

Table 18: Evaluation of insertion. Adapted from: Eskilander (2001).

Insertion is simplified if there are chamfers or other guiding surfaces, e.g. an edge that can be used as a mechanical guide for the fitting operation, in the element.	**Points**
Chamfers exist to simplify the insertion operation.	9
No chamfers, but other guiding surfaces simplifies the insertion operation.	3
No chamfers or other guiding surfaces.	1

6.3.14 Tolerances

During an insertion procedure, tolerances need to be accounted for. In order to reduce the danger of blocking the insertion procedure, which results in stopping the assembly process, tolerances should not be too small. However, tolerances should not be unnecessary high since it can increase the production costs of the element and reduce the overall stability of the component (Eskilander, 2001).

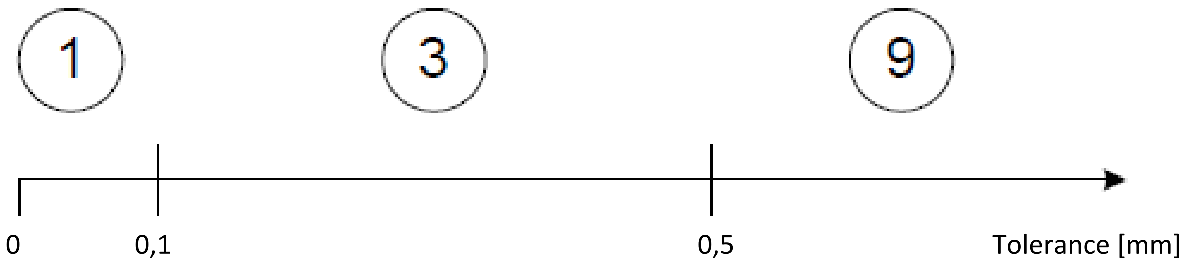


Figure 102: Visualisation of rating regarding the tolerance of an element. Adapted from: Eskilander (2001).

6.3.15 Hold assembled elements

After assembly, an element should be able to stand without support. It is possible to apply extra supports or temporary holding devices, but these can be expensive and have a negative influence on the reliability of the assembly process. A design rule is that an element should not be able to move after it is assembled (Eskilander, 2001).

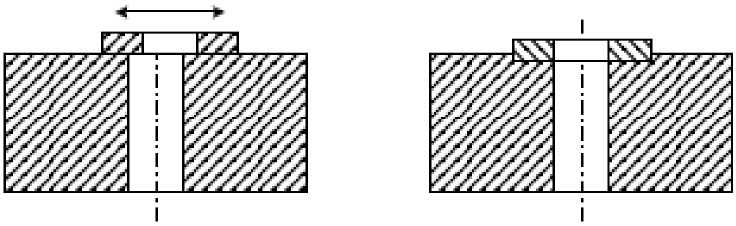


Figure 103: On the left an element is shown that can move after it is assembled. However, an element should not be able to move after it is assembled. Reprinted from: Eskilander (2001).

Table 19: Evaluation of holding assembled elements. Adapted from: Eskilander (2001).

Holding assembled elements is necessary if elements cannot keep orientation and position after assembly. Elements that are secured immediately, i.e. does not lose orientation or position if the assembly is turned upside down, ensures a more reliable assembly process.	**Points**
Element is secured immediately at insertion.	9
Element keeps orientation and position but is not secured.	3
Element must be held after insertion to keep orientation and position.	1

6.3.16 Fastening method

Since the number of fasteners has a big influence on the assembly time they should be minimised. This can be accomplished by integrating fasteners in other elements, using snap fits for example, or by standardising the used fasteners, by applying the same screw type and size (Eskilander, 2001).

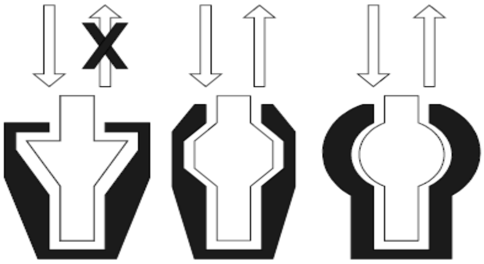


Figure 104: On the middle and right side are two snap fits that are specially designed for assembly and disassembly. Adapted from: Eskilander (2001).

Table 20: Evaluation of fastening elements. Adapted from: Eskilander (2001).

Fastening method. How is the analysed element itself fastened?	Points
No fastening method at all (the element is placed on or in an already assembled element), or only snap fits.	9
Screwing- or pressing operations.	3
Adhesive fastening methods, welding, soldering, riveting.	1

6.3.17 Joining

To improve the assembly process every joining action, like screwing, should be applied from the same direction. The best direction would be from the top since the force of gravity will assist in the process (Eskilander, 2001).

Table 21: Evaluation of joining action. Adapted from: Eskilander (2001).

Joining: Extra equipment or tools (e.g. press tools or screwdrivers) should not be needed to fit the element into place.	Points
No extra equipment is needed.	9
Extra equipment or tools are needed to fit the element in place and the extra operation is performed in the assembly direction.	3
Extra equipment or tools are needed to fit the element in place and the extra operation is not performed in the assembly direction.	1

6.3.18 Check/adjust

Checking or adjusting can be eliminated if it is impossible to assemble the whole component in the wrong way. Designing elements in such a way “that eliminate the risk of assembling the wrong way is called ‘poka yoke’ in Japanese” (Eskilander, 2001, p. 185).

Table 22: Evaluation of checking or adjusting an element. Adapted from: Eskilander (2001).

Check/adjust is not needed if the product is designed according to ”poka yoke”, i.e. it is impossible to assemble the element in more than one way. Every extra operation for checking or adjusting is extra work and a symptom of a design that is not quite satisfactory.	Points
Unnecessary to check if the element is in place.	9
Necessary to check if the element is in place or assembled correctly.	3
Necessary to adjust or re-orient element.	1

RESEARCH BY DESIGN

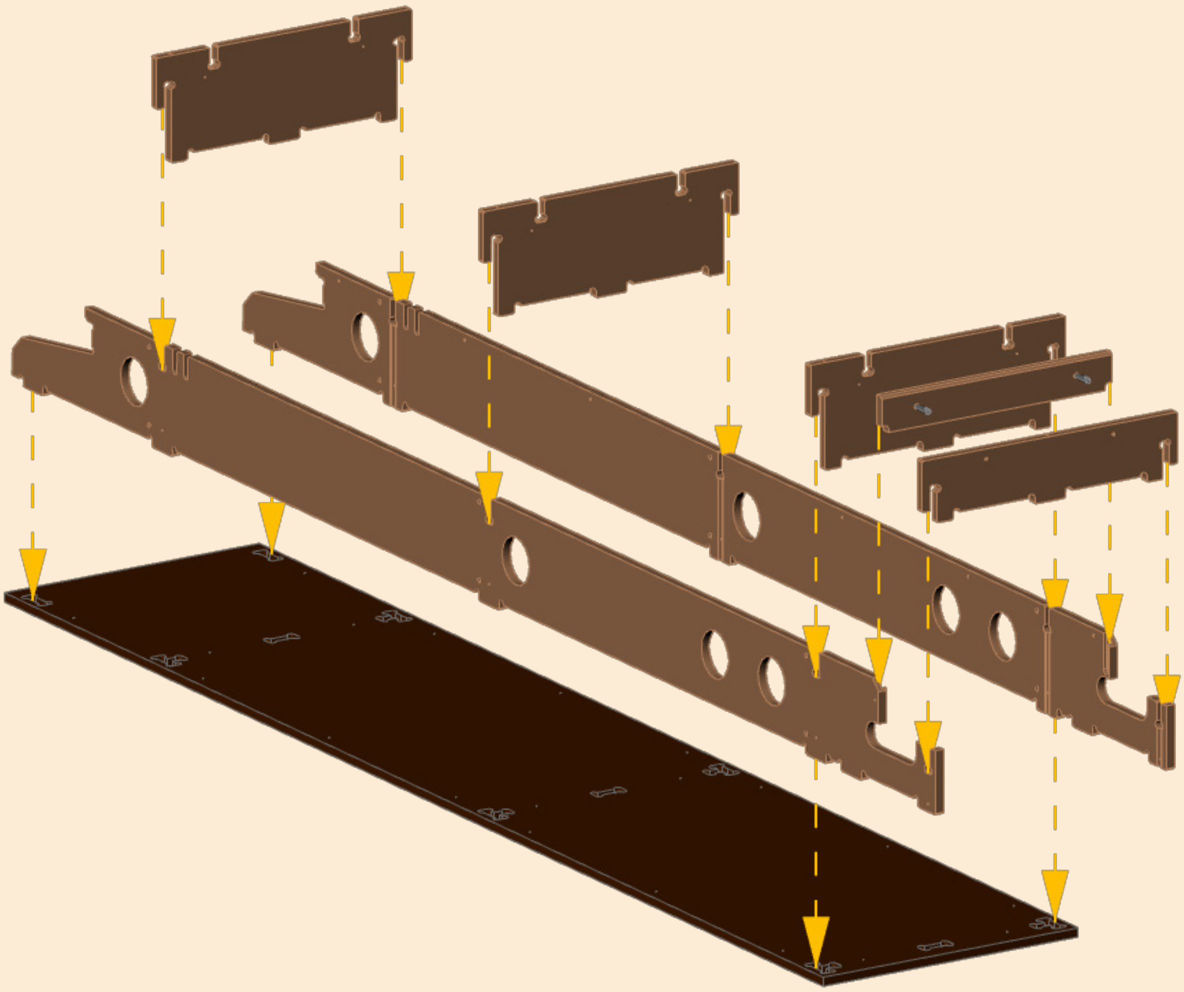


Figure 105: Exploded axonometric 3D view of the whole wall component with the improved geometry.

7 TheNewMakers

7.1 LEGO-inspired building system

7.1.1 TheNewMakers

"Onze gebouwde omgeving is niet flexibel en de realisatie kost nog altijd grote hoeveelheden grondstoffen en tijd. Volgens TheNewMakers (TNM) kan dat veel slimmer!" [Our built environment is not flexible and the realisation still requires large amounts of resources and time. According to TheNewMakers (TNM) this can be done much smarter!] (TheNewMakers, n.d.).

TheNewMakers want to radically change the way buildings are being constructed. In order to accomplish this, TNM is developing, and already applying in different projects, their LEGO-inspired building size scaled construction system that can be seen in Figure 106. The system contains a database of different building components made of sustainably grown wood. All components are circular and digitally produced. The database of building block is being expanded with new components. These components can be designed from a generic point of view to be applicable in many cases, or based on the requirements of a single client. Stijn Mertens is a client of the TNM and he required such a tailor-made component. The component developed for him will be used as a case study in this research.

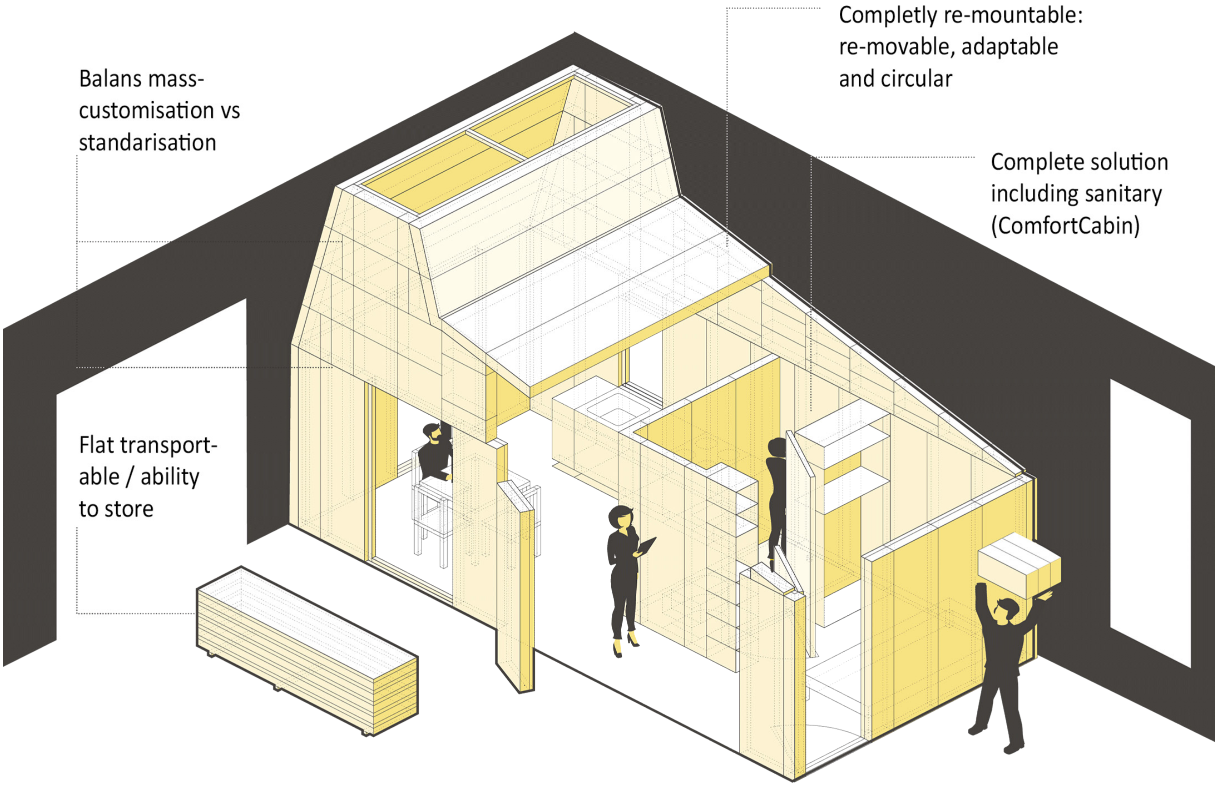


Figure 106: TheNewMakers LEGO-inspired construction system overview. Adapted from:TheNewMakers (n.d.).

7.1.2 Not the only one

Like the TheNewMakers, there are other people that share the same goal: they want to radically improve how buildings are being constructed. For example, the architect Gilles Retsin finds that current buildings consist of too many parts and the current construction methods are too complex, which both make the building industry unnecessary expensive. In order to question these current beliefs and trends in architecture, Gilles designed a unique building system that he applied in his winning design for the Tallinn Architectural Biennale in 2017. The design brief stated to propose a plywood pavilion constructed from LEGO-like building blocks. This design brief came forth from the goal of that year's Biennale, which aimed to inspire people in the building industry to further develop digital fabrication. Figure 108 shows the built pavilion designed by Gilles over which the jury said the following praising words: “the proposal critiques the traditional construction industry, whose need for thousands of different parts results in a slow, expensive and inaccessible process” (UCL, 2017).

More details about Gilles Retsin his building method are presented in images that can be found in Appendix C: Building method Gilles Retsin. All images are reprinted from the website of Retsin (2017).

Developing a new sustainable construction method does not have to be unattractive or mundane, on the contrary, it can lead to a unique expression. This is the case with the design of Gilles. One of the most prominent esthetical features, which are perceived by many as a quality, are the sharp edges that can be seen in Figure 107. These come forth from the construction method. During a presentation of Gilles at a robotic symposium, called ROBOUWTICS, he explained it is easier to align and assemble multiple rectangular blocks under a 45 degree angle instead of trying to assemble these blocks following a traditional orthogonal stacking pattern (Retsin, 2019).

Figure 107: Close-up of assembled components under a 45 degree angle. Adapted from: Retsin (2017).

Figure 108: The constructed pavilion designed by Gilles Retsin for the Architectural Biennale. Adapted from: Retsin (2017).

7.1.3 CNC milling

TheNewMakers and Gilles apply CNC milling in their production process. The main components of a CNC milling machine are: a Computer Numerical Control (CNC) router that is controlled by, as the name suggests, a computer; a milling head, which holds and rotates the milling bit; a worktable on which the workpiece is fastened. The CNC milling machine of TNM has the ability to cut elements up to 5 meters with an accuracy of 0,1 mm, a photo of it can be seen in Figure 109. Figure 110 presents a stack of elements cut by the CNC milling machine, followed by Figure 111 that depicts a photo of a stack of assembled components.

CNC milling time

It takes a certain amount of time to CNC mill an element. As described in research conducted by Altintas and Tulsyan (2015), it is possible to estimate the CNC milling time, although this is rather complex. The accuracy of this estimation can be up to 95%. However, for this research, the CNC milling machine of TNM is available and will be used to get an exact result by simply measuring the time required to complete the CNC milling process. This method is also used by TNM when they want to know the CNC milling time of a new component.

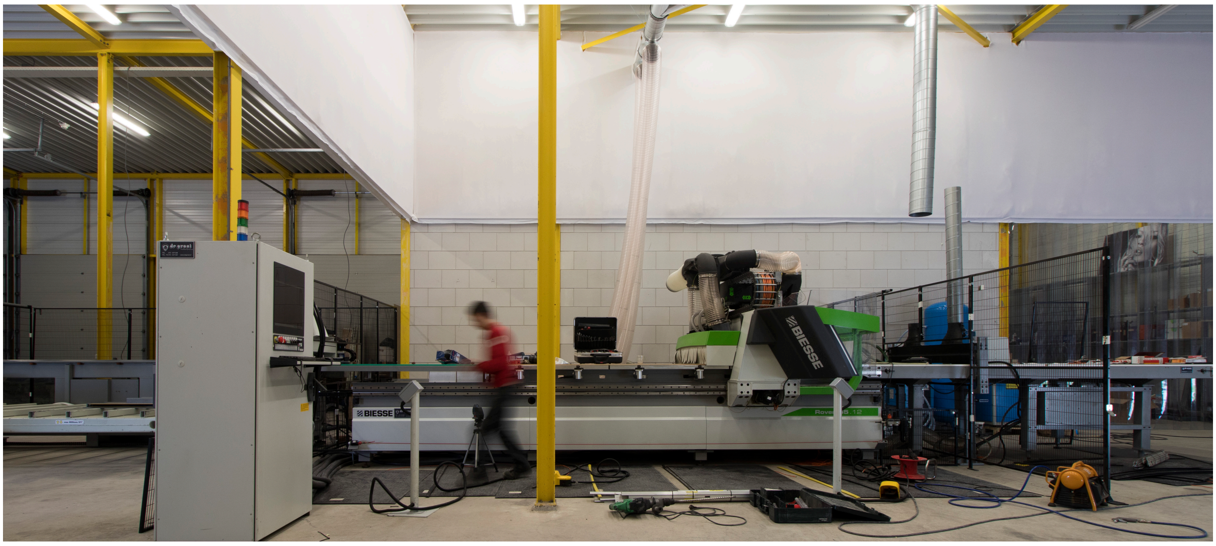


Figure 109: CNC milling machine in the factory of TheNewMakers. Adapted from: TheNewMakers (n.d.).


Figure 110: Stack of CNC milled elements ready to be assembled. Adapted from: TheNewMakers (n.d.).


Figure 111: Stack of assembled components waiting to be transported to site. Adapted from: TheNewMakers (n.d.).

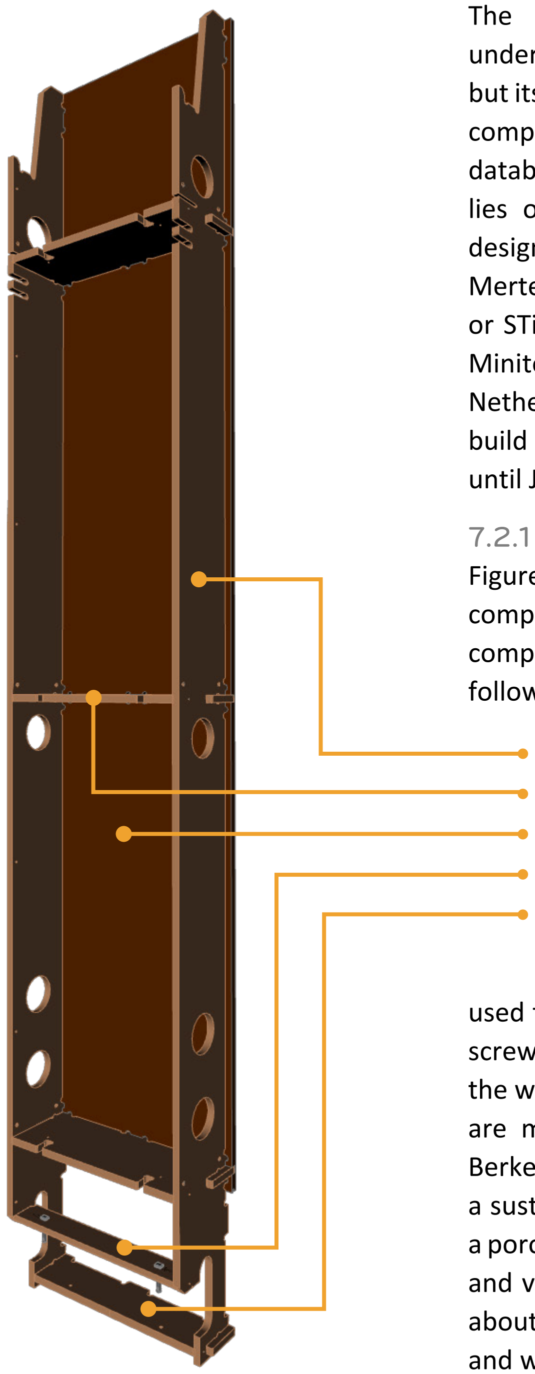


Figure 112: Rendered 3D view towards the interior side of a single wall component of TheNewMakers.

7.2 Case study

The case study helps to gain a better understanding of the building system of TNM, but its primary purpose is to analyse a particular component in the building components database of TNM. In this case study, the focus lies on the wall component of TNM that is designed for the façade of the tiny house of Stijn Mertens, who is a client of TNM. The tiny house, or STiny House, is built in Minitopia Poeldonk. Minitopia is a project in Den Bosch, the Netherlands, which allows several people to build their tiny house and live on that location until June 2023 (Minitopia, 2018).

7.2.1 Wall component

Figure 112 points out the elements on a computer-generated image of the wall component that primarily consists of the following eight CNC milled elements:

- 2 vertical beams (mirrored)
- 3 horizontal beams (identical)
- 1 outer layer
- 1 horizontal floor beam
- 1 horizontal bottom beam

Besides these elements, 16 screws are used to fasten the beams together. Another 25 screws with rubber washers are used to connect the wooden frame to the outer layer. All beams are milled from sheets of sustainably grown Berken multiplex. The outer layer is produced of a sustainable material called Celit 18S, which is a porous wood fiberboard that is water resistant and vapour permeable. The wall component is about 3,5 meters high, exactly 600 mm wide and weighs roughly 31 kg. Table 23 reports the total mass of the whole component and certain details of each element.

Table 23: Overview of the mass of each element of the wall component as well as the total mass.

Element	Volume [m3]	Density [kg/m3]	Mass [kg]	Amount [#]	Total mass [kg]
Vertical beam	0,0112	710	8,0	2	16,0
Horizontal beam	0,0021	710	1,5	3	4,5
Outer layer	0,0325	267	8,7	1	8,7
Horizontal floor beam	0,0007	710	0,5	1	0,5
Horizontal bottom beam	0,0013	710	0,9	1	0,9
Total mass					**30,6**

7.2.2 On-site assembly day

Usually, the elements are assembled into a component in the factory of TheNewMakers and then transported to site where they can be installed directly. However, in this case, the assembly process was conducted on-site.

In order to assemble the elements into the wall component, certain steps need to be followed. For example, the two vertical beams are placed on the assembly table and then the horizontal beams are slid into the vertical beams, which can be seen in Figure 113. Followed by fastening them together with screws and an electric screwdriver. All movements, actions and tools required for the manual assembly process can be found in Appendix A: Manual assembly steps of geometry version A - Original.

Figure 113: Three photos made during the on-site assembly day. From left to right: a stack of elements, a horizontal beam sliding into a vertical beam and an overview of the partially assembled wooden frame.

Figure 114: Outside photo during the construction of the tiny house with several wall components installed on the red frame.

It is interesting to see how many different building components from the database of TheNewMakers are applied in the interior design of this tiny house. Figure 115 provides an overview of all the applied components and their different functions, such as a toilet or a bed. A photo of the exterior of the tiny house is presented in Figure 116.

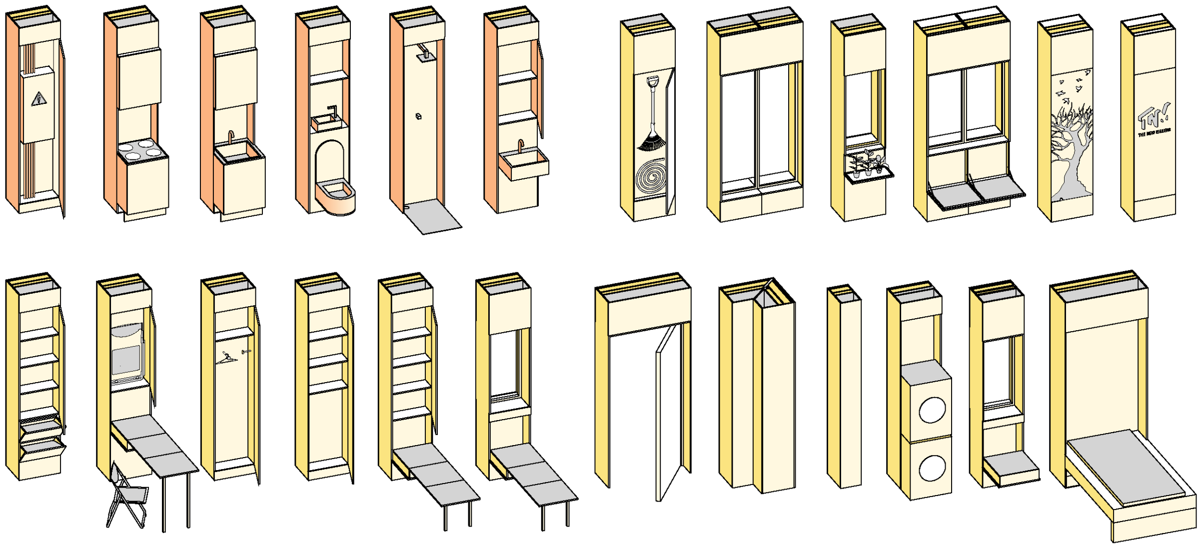


Figure 115: Overview of the applied components in the tiny house. Adapted from: TheNewMakers (n.d.).

Figure 116: Photo of the exterior of the tiny house when construction is almost finished. Adapted from: Mertens (2019).

7.3 Manual assembly process

The elements required for the wall component are CNC milled and assembled in the factory of TheNewMakers. A schematic overview of this manual process is presented in Figure 117. On the left is the unloading area, where an employee unloads the CNC milled element from the CNC milling machine on a pallet. The CNC machine is controlled by a PLC, which stands for Programmable Logic Controller. A PLC is an industrial computer control system that controls the CNC milling machine. Based on certain input provided by sensors or humans, the PLC controls output, like the CNC milling head (AMCI, n.d.). When the pallet is full with a stack of elements, an employee drives it to the assembly area, which can be seen on the right side of the figure. In this area, two employees assemble the different CNC milled elements into the wall component. The required steps to manually assemble this wall component of TheNewMakers can be found in Appendix A: Manual assembly steps of geometry version A - Original. The movements, actions and tools required for each step are described.

7.3.1 Component mass

The manual assembly process of all components in the database of TheNewMakers takes into account that it should be possible to assemble and carry a component with a maximum of two employees. The mass of the component is the most important aspect of this constraint. According to Dutch law one employee is allowed to carry a maximum of 23 kg (Arboportaal, n.d.), which implies that two people are allowed to carry 46 kg. Therefore, a single building component should have a maximum mass of 46 kg.

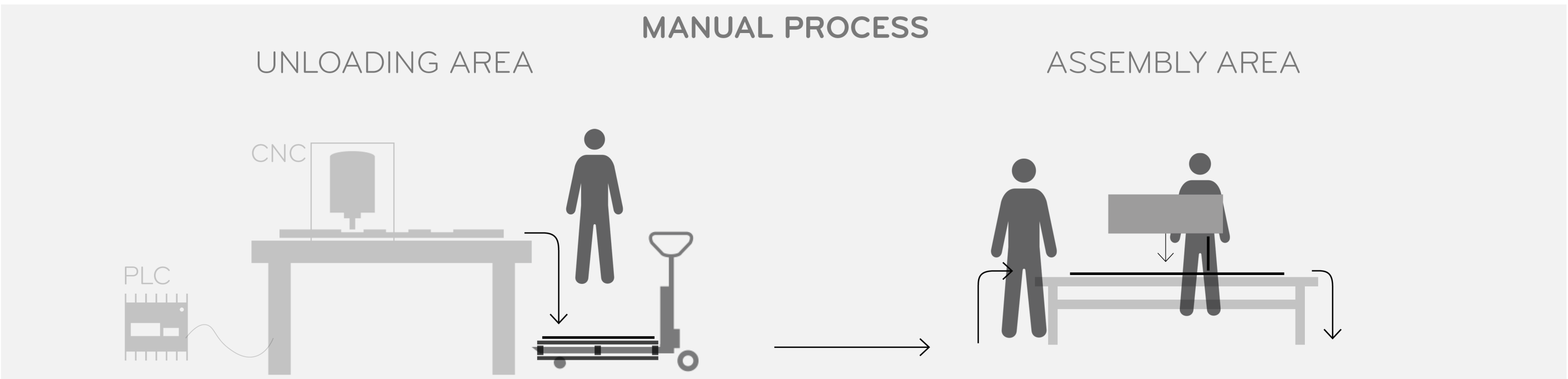


Figure 117: Manual process overview.

8 Robotic process hypothesis

8.1 Robotic process description

The Lean robotics book helped to create the robotic process hypothesis. The proposed robotic assembly process is based on the manual assembly process.

The primary use of this hypothesis is to generate a starting point in order to set up an experiment so the assembly sequence of the wall component can be explored hands-on and any proposed improvements to the wall component can be physically tested. Therefore, this robotic process is fairly generic and not all aspects are worked out in full detail.

Figure 118 proposes the robotic process hypothesis, which is based on the manual process. So the hypothesis also consists of an unloading and assembly area.

8.1.1 Unloading area

The CNC milling machine remains the same as in the manual process. The unloading of the CNC milling machine requires a vision system that can recognise where each CNC milled element is positioned. A robot arm is needed to lift every element of the CNC milling machine and place it on a mobile platform. When the vision system recognises the mobile platform is full, the software will send the mobile platform to the assembly area. The mobile platform will navigate autonomously towards the assembly area.

8.1.2 Assembly area

At the assembly area there is a parking bay available where the mobile platform can navigate to. The software will register when the mobile platform arrives at the assembly area and the robot arm in the assembly area will start moving. It picks up the first element and starts following the assembly process.

When taking into account the main goal of the DFA2 method, which is to use the simplest possible assembly process, it seems logical to apply a single robot in the assembly area in order to confine this complex task to a single entity. Adding a second robot will make the calculations that need to be made far more complex, as is described in the book “Robotics: Science and Systems VI”, it states: “Multi-robot path planning suffers from the inherent complexity resulting from the necessity of operating in Cartesian products of configuration and state spaces” (Matsuoka, Durrant-Whyte, & Neira, 2011). Besides that, a single robot provides flexibility. If expansion is required another robot can be acquired and a similar assembly area can be setup that incorporates the new robot. This is as per the advice of Sami Atiya, president of the Robotics and Motion division of Swedish-Swiss automation company ABB Group. He said: “invest in automation solutions that provide flexibility and agility to grow in whichever direction the market goes” (ABB, 2018). Therefore, one robot arm will be applied in the assembly area.

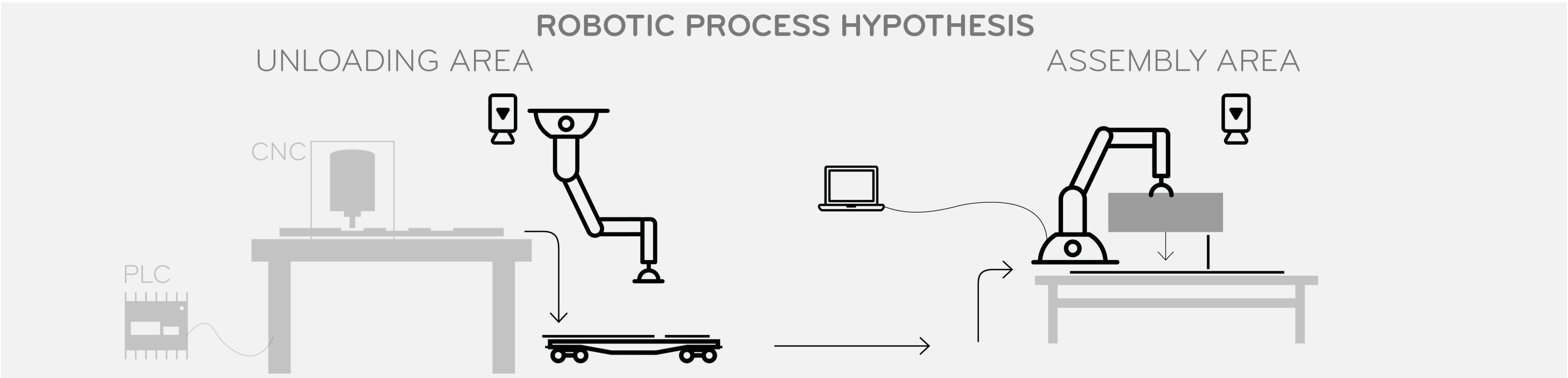


Figure 118: Robotic process hypothesis overview.

8.2 Assembly robot type

This paragraph considers what type of robot is suitable to assemble the wall component. The comparison graphs from paragraph 5.6 are used to mark the constraint or desired goal.

8.2.1 Payload capacity

As mentioned in chapter 5 Industrial robots vs cobots, the payload capacity is one of the most critical characteristics when choosing a robot. The payload that the robot arm needs to carry consists of two variables, being the mass of the component that needs to be lifted plus the mass of the end effector.

The mass of the whole wall component is about 31 kg. However, other building components of TheNewMakers can have a maximum mass of 46 kg. In order for the robot to be able to lift every component that might be in the database of TNM, the mass taken into account should be 46 kg.

Designing an end effector is beyond the scope of this research, therefore the exact mass of the end effector is unknown. However, without designing an end effector its mass can be assumed. Generally speaking, the mass of the end effector approximately equals the mass of the object that is going to be carried (Motion Controls Robotics, 2017a), which is the mass of the heaviest component of 46 kg.

Combining the mass of the wall component and the end effector results in a required payload capacity of 92 kg (= 46 kg + 46 kg). Two aspects are important when a robot arm carries a component: it should be picked up at the centre of gravity of the component and within close proximity of the base of the robot. Otherwise, the robot might not be able to lift the component or the robot could damage its motors. In order to leave some room for deviation of these two aspects the required payload capacity is increased with about 10% to 100kg.

The payload capacity has a big influence on the purchase price of the robot. Therefore, choosing a robot with a significantly higher payload capacity than required will likely result in an increased purchase price of the robot, which should be avoided.

Figure 119 presents the minimal required payload capacity in the graph that shows the maximum payload capacity of industrial robots and cobots.

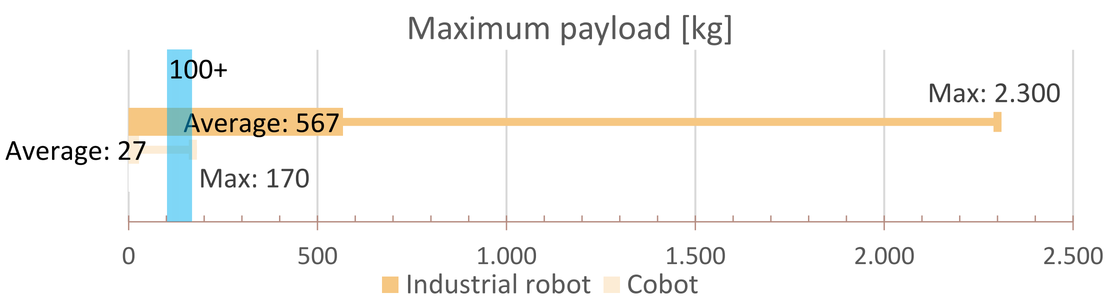


Figure 119: Required payload capacity marked with a vertical blue bar in the graph that shows the maximum payload of the industrial robots in Table 4 and the cobots in Table 5.

8.2.2 Maximum reach

In order for the robot to be able to assemble every element of the wall component, a minimal reach is required of about half the length of the wall component, which is 1,75 meter (= 3,5 m / 2). However, other building components of TheNewMakers can be bigger. This size of the elements is limited by the size of sheets the CNC milling machine can handle, which is 5 meters. Therefore, the elements can be up to 5 meters long, resulting in a reach of 2,5 meters (= 5 m / 2). Figure 120 visualises the required reach in the graph that shows the maximum reach of industrial robots and cobots.

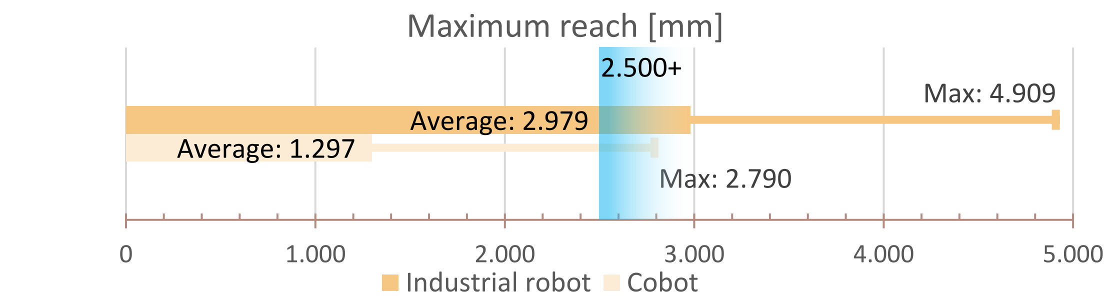


Figure 120: Required reach marked with a vertical blue gradient bar in the graph that shows the maximum reach of the industrial robots in Table 4 and the cobots in Table 5.

8.2.3 Velocity

The velocity of the robot arm influences how long it takes for the end of the robot arm to move over a certain distance. A higher velocity usually results in a faster assembly process and a reduced assembly time, which is desired. This is marked in Figure 121.

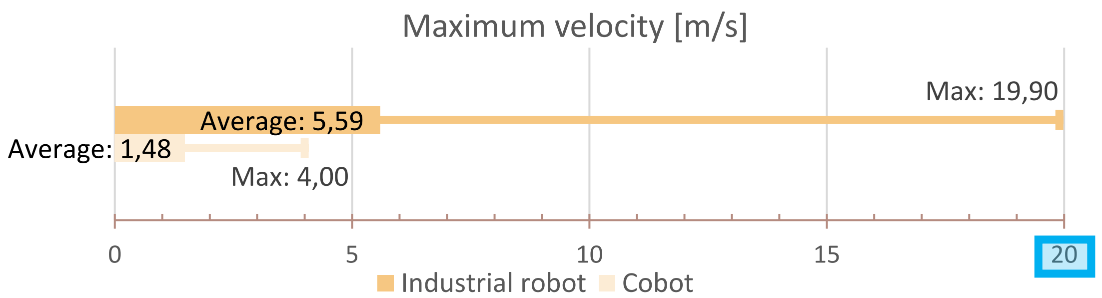


Figure 121: Desired velocity marked with a blue rectangle in the horizontal axis of the graph that shows the maximum velocity of the industrial robots in Table 4 and the cobots in Table 5.

8.2.4 Repeatability

Figure 122 presents the repeatability. For the assembly sequence, a low value for the repeatability and the accuracy is desired. The lower these values are, the more precise the robot arm is able to move the end effector.

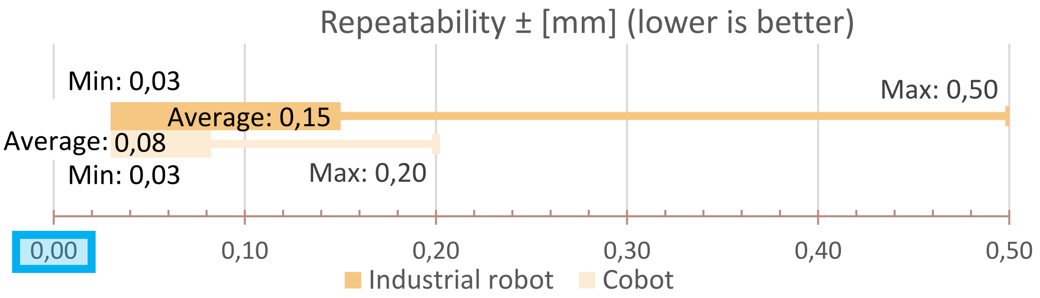


Figure 122: Desired repeatability marked with a blue rectangle in the horizontal axis of the graph that shows the repeatability of the industrial robots in Table 4 and the cobots in Table 5.

8.2.5 Mass

A robot with a lower mass could allow more installation options than only installation on the floor, like a ceiling installation. Besides, a lighter robot is easier to handle and install. Therefore, it is desired to choose a robot with a lower mass. In Figure 123 a low desired mass is marked.

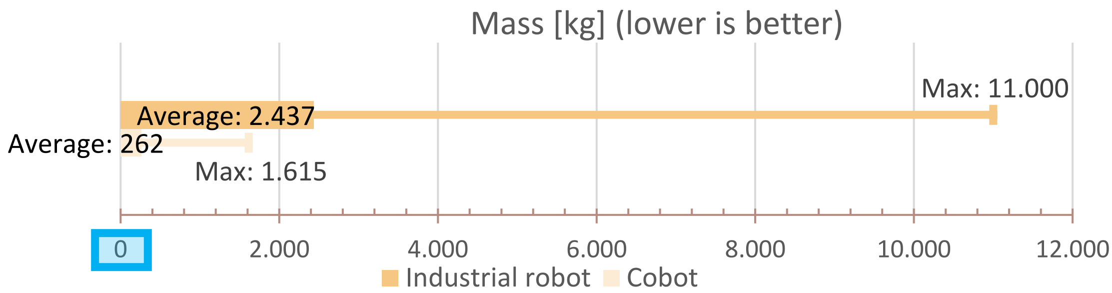


Figure 123: Desired mass marked with a blue rectangle in the horizontal axis of the graph that shows the mass of the industrial robots in Table 4 and the cobots in Table 5.

8.2.6 Total Cost of Ownership (TCO)

If the other characteristics of the robot are similar and all comply with the constraints, it is preferred to buy a cheaper one. Therefore, the desired Total Cost of Ownership is as low as possible, which is marked in Figure 124.

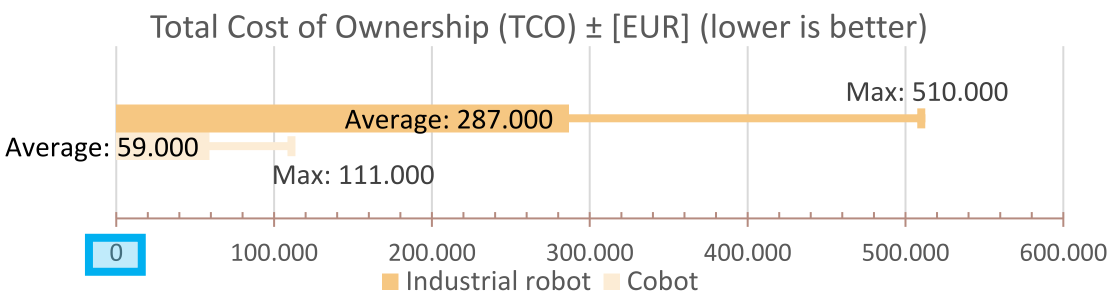


Figure 124: Desired Total Cost of Ownership marked with a blue rectangle in the horizontal axis of the graph that shows the TCO of the industrial robots in Table 4 and the cobots in Table 5.

8.3 Assembly robot choice

8.3.1 Criteria

In order to help choose a robot to operate in the assembly area a weight from 1 to 5 has been assigned to each criterion mentioned in the previous paragraph. An overview of the criteria including their weights and reasoning behind the weights is presented in Table 24.

Table 24: Weights on a scale of 1 to 5 and the reasoning behind each criterion.

Robot choice criterion	Weight	Criterion description and weight reasoning
Max. payload	5	Amount of mass the robot can lift. This is the one of the most important criterion when selecting a robot, thus it has the heaviest weight.
Max. reach	5	The distance over which the robot can pick up an object. This is very important in order for the robot to be able to assemble all the components of TNM.
Max. velocity	4	Movement speed of the robot. The geometry of the elements is going to be redesigned to reduce robotic assembly time, to reduce this time the speed of the robot is also an important factor.
Repeat-ability	3	How accurate the robot is. If the robot is not accurate enough the tolerances in the elements need to be bigger.
Mass	1	The mass of the robot itself, which is not very important for this assembly process.
TCO (Total Cost of Owner-ship)	1	Indication of the amount of money required to install and use the robot. Although this is very important from a company point of view, in this research the TCO is roughly estimated based on the maximum payload capacity and maximum reach. Due to the rudimentary nature of this estimation the TCO has the lowest weight possible.

8.3.2 Conceptual comparison

When considering the findings described in subparagraph 5.6.2 Conceptual comparison, it appears to be advisable for a relatively small company like TheNewMakers to invest in a hybrid assembly, also known as human-robot collaboration. This suggests looking into acquiring a collaborative robot, which is also known as a cobot.

8.3.3 Cobot

When looking at the first two criteria, being payload and reach, there is only one cobot left that fulfils these two constraints. This robot is the AURA-170-2.8, which features an industrial and collaborative mode. It has a payload capacity of 170 kg and a reach of 2.790 mm. Despite it matching the two constraints, the payload capacity is 70% higher than required, which likely results in an unnecessary high TCO.

The benefits of a cobot are diminished when a high payload is carried. The result of a cobot moving a high payload is a significant amount of inertia, which can be dangerous for a human. To prevent harm to employees in such a situation either the movement speed of the cobot needs to be very low or extra safety measures should be taken (Ryan, 2016).

8.3.4 Industrial robot

Instead of choosing a cobot, it is possible to apply an industrial robot together with the vision system described in paragraph 4.2 Vision. This system, provided by Veo Robotics, enables a choice from the wide variety of available industrial robots while one of the main benefits of cobots remain, being that there is no need for extra safety measurements such as a cage.

A safety vision system is also suggested to be applied when a collaborative robot needs to lift a relatively heavy payload (Baek, 2016). Due to such a vision system industrial robots and cobots can be considered equal for this application.

8.3.5 RoboDK

Since a robotic assembly simulation is going to be made in a simulation tool called RoboDK and this software contains a large library with industrial robots, a robot will be chosen from this library. The robot should fit the two constraints, payload and reach, plus closely matches the other criteria. Based on the two constraints, a preselection is made using the library on the website of RoboDK. The payload capacity is set to the range 100 - 150 kg, the reach is set to 2.000 - 3.000 mm and robot type is set to 6 DOF (RoboDK, 2019d). The latter is done in order to filter out other robot types, like delta robots. Filtering out the ones with a reach lower than 2.500 mm results in an overview with 11 robots that is presented in Table 25. The robots are sorted from low to high on payload and then on reach.

8.3.6 Weighted score

The robots in Table 26 on the next page are the same as the ones in Table 25. Table 26 shows the weighted score of each robot on the six criteria on a scale ranging from 1 to 3, where three is the best. The best 5% of the robots on a certain criterion are rewarded with a 3, the robots below the average on that criteria receive a 1, and the robots in between are awarded a 2.

The maximum amount of score is 54. Two robots are in the top 20% range, both robots are highlighted in Table 26 and a photo of each is shown in Figure 125 and Figure 126. The Smart5 NJ 110-3.0 industrial robot from Comau (2019b) has a score of 52, which is the highest. Besides that it has the best score, the two constraints, being the payload and reach, have plenty of room to spare. Also, the repeatability only deviates 0,02 mm from the best robot on this criterion and it is almost one of the lightest robots. Therefore, it is the best suitable robot to use in the robotic assembly process and will be applied in the simulation that is described in chapter 11 Robotic simulation.

Table 25: Overview of available industrial robots in the RoboDK library complying with the payload and reach constraints.
[1]unable to find documentation that exactly matches this robot and it appears to be a robot that is not sold new anymore.
[2]although proper documentation is found, it appears to be a robot that is not sold new anymore.

Industrial robot (from RoboDK database)	Company and source	Pay-load [kg]	Reach [mm]	Max velocity [°/s]	Repeat-ability ± [mm]	Mass [kg]	Base type [mm]	Base area [m2]
Smart5 NJ 110-3.0	Comau (2019b)	110	2.980	110-230	0,07	1.070	Ø 700	0,385
KR 120 R2500 pro	KUKA (2019a)	120	2.500	120-292	0,06	1.049	830x830	0,689
KR 120 R2700 HA	KUKA (2018)	120	2.696	101-292	0,05	1.104	830x830	0,689
IRB 6400 2.8 120[2]	ABB (2001)	120	2.800	90-190	0,10	2.230	Ø 800	0,503
KR 150 L120/2[1]	KUKA	120	2.810		0,20	1.140	?	?
KR 120 R2900 extra	KUKA (2019b)	120	2.896	115-292	0,06	1.084	830x830	0,689
Smart5 NJ 130-2.6	Comau (2019b)	130	2.616	110-230	0,07	1.050	Ø 700	0,385
RX260[2]	Stäubli (2010)	150	2.831	104-230	0,07	1.900	?	?
S-430iW[2]	FANUC (2001)	135	2.643	105-210	0,30	1.300	610x771	0,470
KR 150 R2700 extra	KUKA (2019c)	150	2.696	115-219	0,06	1.068	830x830	0,689
KR 150-2	KUKA (n.d.)	150	2.700	100-238	0,06	1.245	830x830	0,689

Figure 125: The chosen industrial robot with a score of 52, named Smart5 NJ 110-3.0. Adapted from: Comau (2019a).

Figure 126: The robot with a score of 50, named KR 120 R2900 extra. Reprinted from: KUKA (2019b).

Table 26: Weighted score from 1 to 5 for every robot on each criteria using a 1 to 3 scale. The best two are highlighted.

	Criterion:	Max. payload	Max. reach	Max. velocity	Repeatability	Mass	TCO	TOTAL weighted score
	Weight (1=min, 5=max):	5	5	4	3	1	1	
Robot	Company							
Smart5 NJ 110-3.0	**Comau**	**3**	**3**	**2**	**2**	**3**	**3**	**52**
KR 120 R2500 pro	KUKA	2	1	3	2	3	2	40
KR 120 R2700 HA	KUKA	2	1	3	3	2	2	38
IRB 6400 2.8 120	ABB	2	2	1	2	1	2	31
KR 150 L120/2	KUKA	2	2	3	1	2	2	41
KR 120 R2900 extra	**KUKA**	**2**	**3**	**3**	**2**	**3**	**2**	**50**
Smart5 NJ 130-2.6	Comau	1	1	2	2	3	1	30
RX260	Stäubli	1	3	1	2	1	1	19
S-430iW	FANUC	1	1	1	1	1	1	26
KR 150 R2700 extra	KUKA	1	1	1	2	3	1	23
KR 150-2	KUKA	1	1	1	2	2	1	30

9 Geometry

Modifying the geometry of the wall component enables the possibility to change the assembly steps. When the right modifications are made the assembly process will be easier, which is very likely to result in a faster assembly process.

9.1 Version A - Original

The wall component of TheNewMakers is shown in Figure 127. The required steps to manually assemble it can be found in Appendix A: Manual assembly steps of geometry version A - Original. The movements, actions and tools required for each step are described.

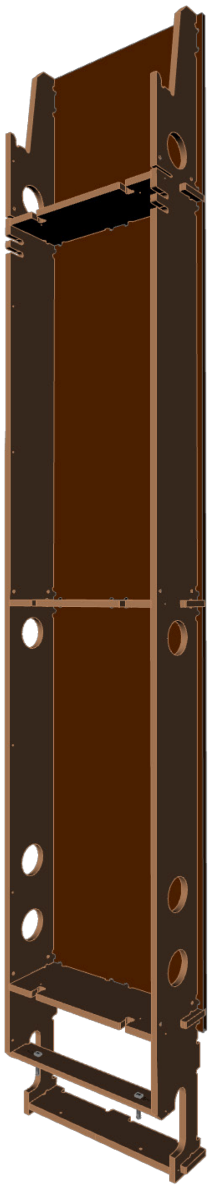


Figure 127: Rendered 3D view towards the interior side of a single wall component of TheNewMakers.

9.2 Version B - DFA2

Geometry version B is the result of a redesign process, which considered the following two objectives:

1. Reduce assembly time (primary)
2. Reduce CNC milling time (secondary)

Reducing the assembly time has the highest priority, therefore it is the primary objective and the main focus in this chapter. The method Design For Automatic Assembly (DFA2) will be applied to redesign geometry version A, which is the geometry of the original wall component. An overview of the DFA2 method can be seen in Figure 128. Applying this method will result in a redesigned geometry with a reduced assembly time.

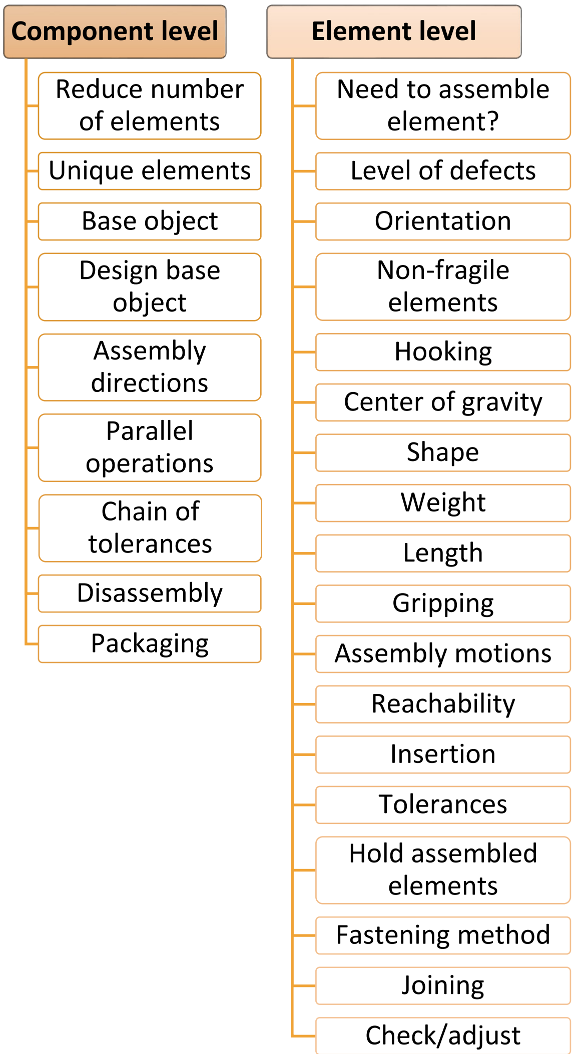


Figure 128: Structure of the DFA2 method. Adapted from: Eskilander (2001).

9.2.1 Component level

The consideration of each step on the component level is briefly described and presented in Figure 129.

9.2.2 Element level

Just as the component level, the consideration of each step on the element level is briefly described and presented in Figure 130.

Component level

- **Reduce number of elements**
  Remains the same.
- **Unique elements**
  Remains the same.
- **Base object**
  A base object was chosen, which made the assembly process much easier.
- **Design base object**
  The object with the best base design was chosen.
- **Assembly directions**
  The number of assembly directions has been reduced, which simplyfies and speeds up the assembly process.
- **Parallel operations**
  Remains the same.
- **Chain of tolerances**
  Remains the same, can not be imrpoved any further.
- **Disassembly**
  Remains the same (it is not rated in the component level assembly index).
- **Packaging**
  Remains the same (it is not rated in the component level assembly index).

Figure 129: Considerations made on the component level.

Element level

- **Need to assemble element?**
  Remains the same.
- **Level of defects** Remains the same.
- **Orientation**
  Improved by better positioning how the elements are fed to the assembly area.
- **Non-fragile elements** Remains the same.
- **Hooking** Remains the same.
- **Center of gravity** Remains the same.
- **Shape** Remains the same.
- **Weight**
  Could be reduced by milling material away that is not needed, but this would require extra CNC milling time so it is not considered. Applying a ligter material is not explored, it is beyond the scope of this research.
- **Length** Remains the same.
- **Gripping** Remains the same.
- **Assembly motions** Remains the same.
- **Reachability**
  The new assembly steps make it easier to reach the elements.
- **Insertion** Remains the same.
- **Tolerances**
  Improved and physically tested as shown in paragraph 10.3 Geometry tolerances.
- **Hold assembled elements**
  Slightly improved due to assembly steps.
- **Fastening method** Remains the same.
- **Joining**
  It is easier to join the elements due to the imrpoved assembly steps and tolerances.
- **Check/adjust**
  Less asjusting is required due to the extra tolerances.

Figure 130: Considerations made on the element level.

9.3 DFA2 comparison A and B

In this paragraph, geometry version A and B will be compared using the DFA2 method. Both versions are evaluated on efficiency from an automated assembly perspective, which will result in a rating called the assembly index. It will show if there is an improvement on component and element level.

9.3.1 Component level

Geometry version A – Original

The assembly steps are shown in Appendix A: Manual assembly steps of geometry version A - Original. It presents the 15 steps required to assemble the whole wall component. An exploded 3D view is presented in Figure 131 to give an overview of the wall component and its elements. Table 27 presents the assembly index on the component level.

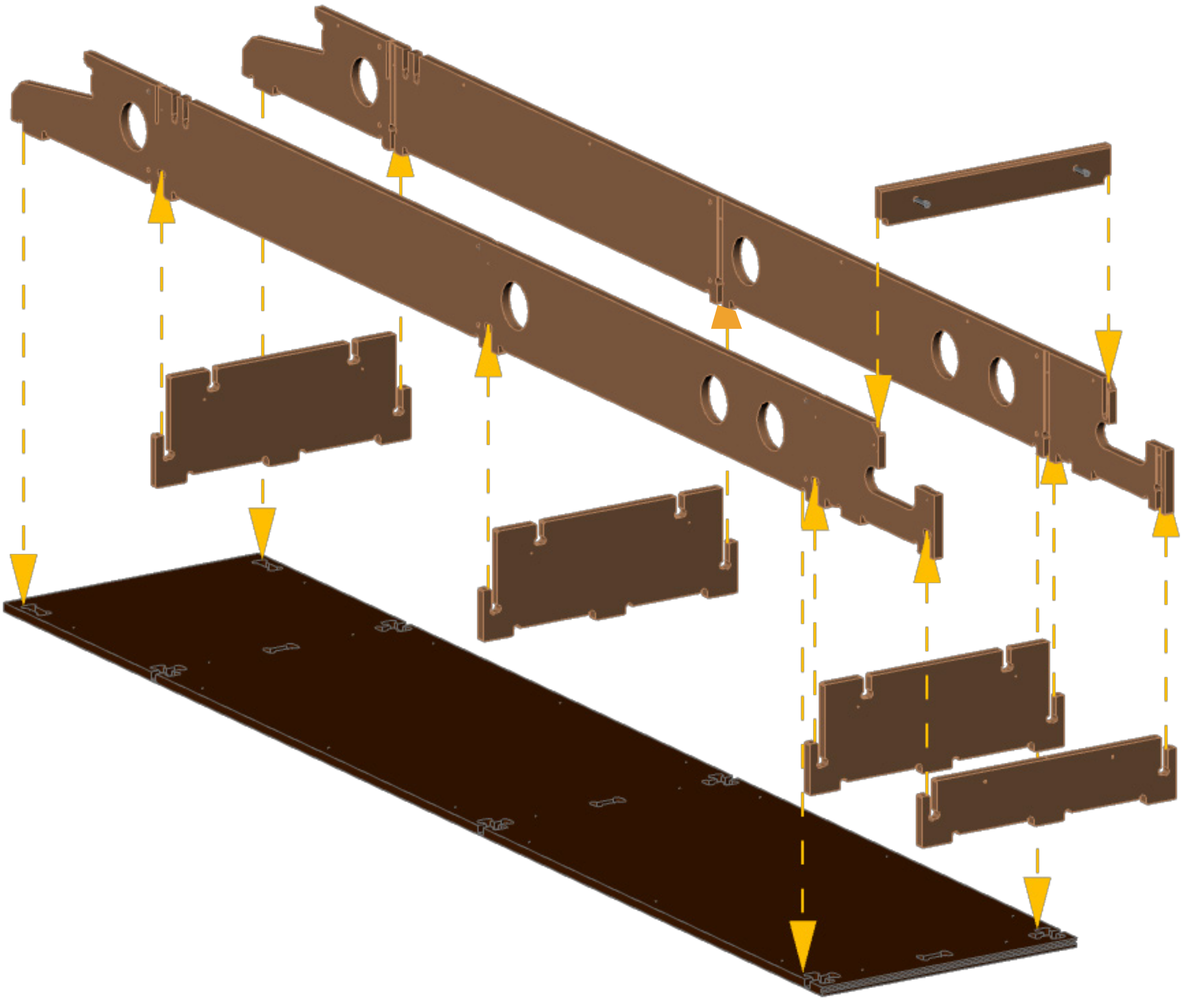


Figure 131: Exploded axonometric 3D view of the whole wall component with geometry version A - Original.

Table 27: Geometry version A rating on component level based on the DFA2 method. Adapted from: Eskilander (2001).

	1 Component level: Geometry version A - Original							
	1.1 Reduce number of elements	1.2 Unique elements	1.3 Base object	1.4 Design base object	1.5 Assembly directions	1.6 Parallel operations	1.7 Chain of tolerances	SUM
Value	49 elements	7/38=18%	No		3 directions	0 parallel	No chain	
Rating	1	9	1	1	1	1	9	23

$$\text{Assembly index} = \frac{\text{Total sum}}{\text{Maximum points}} = \frac{23}{63} = 36{,}5\%$$

Geometry version B - DFA2

The assembly steps are shown in Appendix B: Manual assembly steps of geometry version B - DFA2. It shows that the amount of assembly steps is reduced from 15 to 9. An exploded 3D view is presented in Figure 132 to give an overview of the updated wall component and its elements. Table 28 presents the assembly index on the component level.

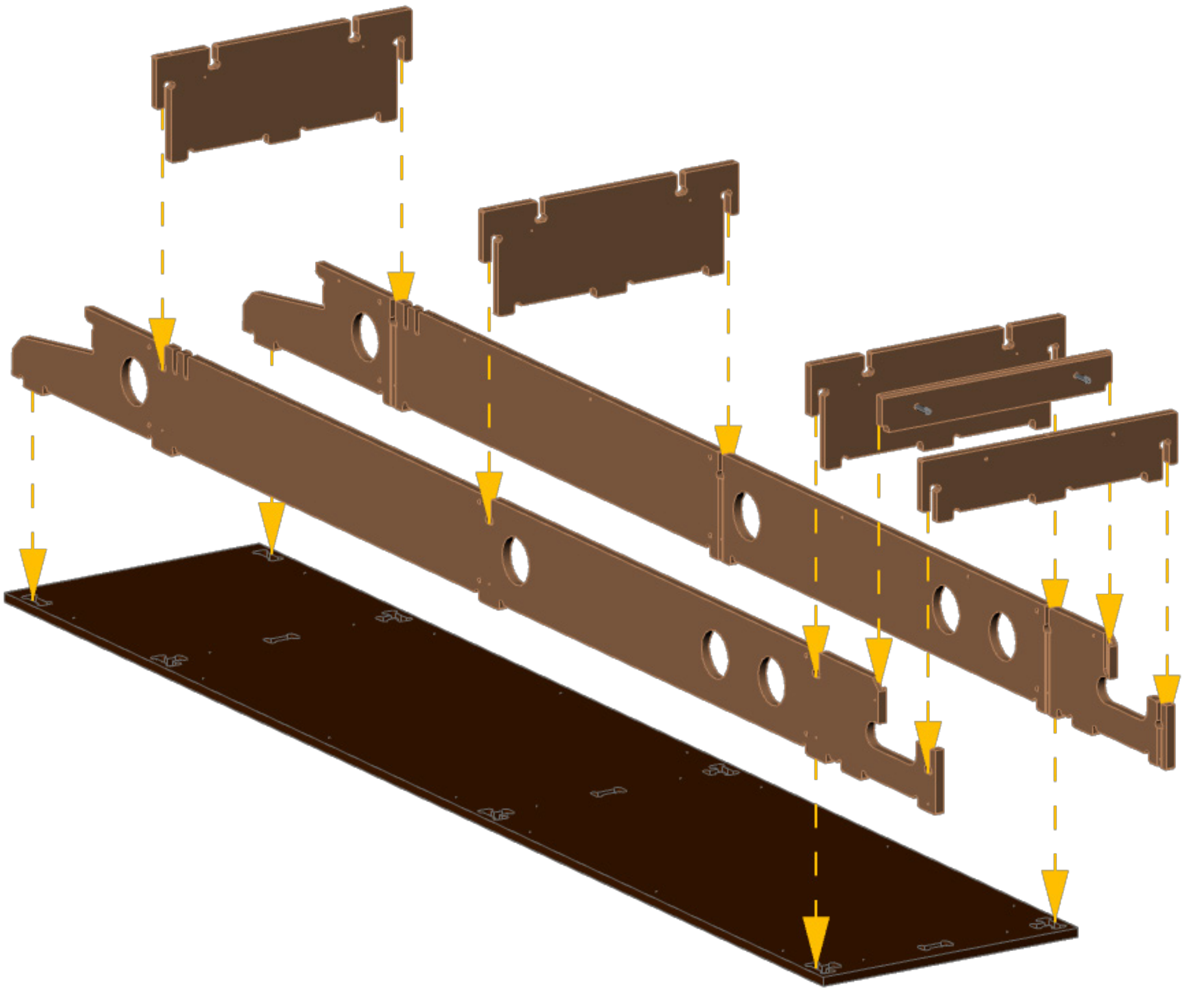


Figure 132: Exploded axonometric 3D view of the whole wall component with geometry version B - DFA2.

Table 28: Geometry version B rating on component level based on the DFA2 method. Adapted from: Eskilander (2001).

	1 Component level: Geometry version B - DFA2							
	1.1 Reduce number of elements	1.2 Unique elements	1.3 Base object	1.4 Design base object	1.5 Assembly directions	1.6 Parallel operations	1.7 Chain of tolerances	SUM
Value	49 elements	7/38=18%	Yes		2 directions	0 parallel	No chain	
Rating	1	9	9	3	3	1	9	35

$$\text{Assembly index} = \frac{\text{Total sum}}{\text{Maximum points}} = \frac{35}{63} = 55{,}6\%$$

9.3.2 Element level

Geometry version A - Original

Table 29 presents the assembly index on the element level.

Table 29: Geometry version A rating on element level based on the DFA2 method. Adapted from: Eskilander (2001).

		2 Element level: Geometry version A - Original																		
List of elements	Number of identical elements	2.1 Need to assemble element?	2.2 Level of defects	2.3 Orientation	2.4 Non-fragile elements	2.5 Hooking	2.6 Centre of gravity	2.7 Shape	2.8 Weight	2.9 Length	2.10 Gripping	2.11 Assembly motions	2.12 Reachability	2.13 Insertion	2.14Tolerances	2.15 Hold assembled elements	2.16 Fastening method	2.17 Joining	2.18 Check/adjust	SUM
Vertical beams	2	9	9	1	9	9	9	1	1	1	3	3	9	9	1	1	9	9	9	204
Horizontal beams	3	9	9	1	9	9	9	1	3	1	3	9	9	9	1	3	9	1	1	288
Horiz. floor beam	1	9	9	1	9	9	9	1	3	1	3	9	3	9	1	3	9	1	1	90
Horiz. bottom beam	1	9	9	1	9	1	9	1	3	1	3	9	9	9	1	3	9	1	1	88
Outer layer	1	9	9	1	9	9	9	1	1	1	3	9	3	9	1	3	9	1	1	88
Screws beams	10	9	9	1	9	1	1	3	3	3	1	1	1	9	1	9	3	1	9	740
Screws outer layer	20	9	9	1	9	1	1	3	3	3	1	1	1	9	1	9	3	1	9	1.480
Total	**38**																			**2.978**

Assembly index = Total sum / (Maximum points * number of elements) = 2.978 / (162 * 38) = 48,4%

Geometry version B - DFA2

Table 30 presents the assembly index on the element level.

Table 30: Geometry version B rating on element level based on the DFA2 method. Adapted from: Eskilander (2001).

		2 Element level: Geometry version B - DFA2																		
List of elements	Number of identical elements	2.1 Need to assemble element?	2.2 Level of defects	2.3 Orientation	2.4 Non-fragile elements	2.5 Hooking	2.6 Centre of gravity	2.7 Shape	2.8 Weight	2.9 Length	2.10 Gripping	2.11 Assembly motions	2.12 Reachability	2.13 Insertion	2.14Tolerances	2.15 Hold assembled elements	2.16 Fastening method	2.17 Joining	2.18 Check/adjust	SUM
Vertical beams	2	9	9	9	9	9	9	1	1	1	3	3	9	9	3	3	9	9	9	228
Horizontal beams	3	9	9	9	9	9	9	1	3	1	3	9	9	9	3	3	9	3	3	330
Horiz. floor beam	1	9	9	9	9	9	9	1	3	1	3	9	9	9	3	3	9	3	3	110
Horiz. bottom beam	1	9	9	9	9	1	9	1	3	1	3	9	9	9	3	3	9	3	3	102
Outer layer	1	9	9	9	9	9	9	1	1	1	3	9	9	9	3	9	9	9	9	126
Screws beams	10	9	9	3	9	1	1	3	3	3	1	1	1	9	1	9	3	3	9	780
Screws outer layer	20	9	9	3	9	1	1	3	3	3	1	1	1	9	1	9	3	1	9	1.520
Total	**38**																			**3.196**

$$\text{Assembly index} = \frac{\text{Total sum}}{\text{Maximum points * number of elements}} = \frac{3.196}{162 * \quad 38} = 51{,}9\%$$

9.3.3 Compare version A and B

The assembly steps of geometry B were improved significantly, which resulted in an improvement of the assembly index on a component level of more than 1,5 times. The assembly index on element level did not improve so much. This was partially due to the fact that the number of screws required for the component is relatively high, which had a major influence on the element level assembly index. To illustrate the influence of the screws they were temporarily disregarded, which resulted in an assembly index of 58,5% and 69,1% for geometry A and B respectively, thus improving with a factor of 1,18 instead of 1,07. The overview of the assembly index including screws is presented in Table 31.

Table 31: Overview of the assembly index of geometry version A and B on a component and element level.

Geometry	Component level assembly index	Element level assembly index
Version A - Original	36,5%	48,4%
Version B - DFA2	55,6%	51,9%
Improvement of version B compared to version A	**1,52**	**1,07**

10 Physical experiment

10.1 Aim

The aim of the experiment is to physically test geometry modifications of the CNC milled elements with a robot, generate input required for the robotic simulation and briefly evaluate the five experiment topics in relation to the robotic process hypothesis.

The experiment was conducted using a section of the wall component of TheNewMakers and in collaboration with a team of five HBO mechatronics students from The Hague University of Applied Sciences, called *De Haagse Hogeschool* (HHS) in Dutch. The experiment was executed at HighTechCentreDelft, or HTCDelft, an organisation that provides a technical workspace for students. In 2017 HTCDelft offered 538 students from MBO (vocational educational training), HBO (professional bachelor's) and WO (master) level a place to work on their projects (HTCDelft, 2017, p. 4). HTCDelft also provided the UR5 collaborative robot that was used in the assembly experiment.

The experiment only considered a section of the whole wall component, which is depicted in Figure 133. The reasoning behind using only a section and why this particular section is discussed in the next paragraph.

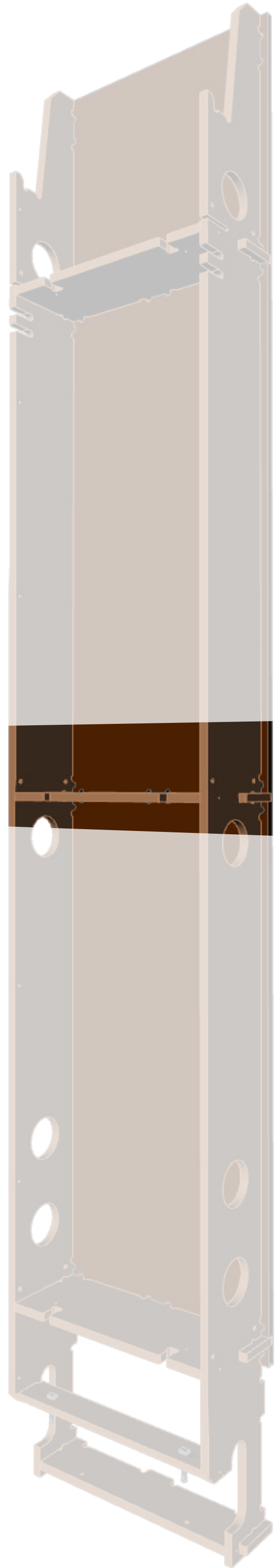


Figure 133: Rendered 3D view towards the interior side of a single wall component of TheNewMakers with the section for the experiment highlighted.

10.2 Section

The available robot that was used in the experiment, the UR5 cobot, has a maximum payload capacity of 5 kg. This is by far not enough to lift the heaviest element that weighs 8,7 kg, let alone the whole wall component, which has a total weight of about 31 kg. To reduce the weight a section was selected. A section was chosen where a horizontal beam meets the two vertical beams as well as the outer layer so it contains all three assembly detail principles, which are:

1. Slide, which refers to sliding one element into another.
2. Screw, which considers fastening elements together using screws.
3. Dog bone, which refers to pushing an element into a dog bone connection.

The section is highlighted in Figure 136. From the centre of the horizontal beam, a 200 mm measurement was set out in two direction parallel with the longest side of the whole wall component and on that location a cutting plane was made. All geometry outside these two cuts was deleted. The measurement of 200 mm was chosen to ensure enough material would be available to work with during the experiment. The resulting section can be seen in Figure 134. However, the mass of this section turned out to be very close to the payload capacity of the UR5 cobot as is presented in Table 32. It would leave only 0,4 kg (= 5 kg - 4,6 kg) for the end effector. Therefore, the size of the section was reduced, which can be seen in Figure 135. Table 33 shows the weight of each element of the updated section. This left 1,4 kg (= 5 kg - 3,6 kg) for the end effector, which was enough according to an estimation of the HBO students.

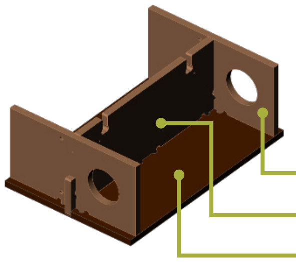


Figure 134: Axonometric 3D view initial section.

Table 32: Initial section of the wall component for the experiment.

Element	Volume [m3]	Density [kg/m3]	Mass [kg]	Amount [#]	Total mass [kg]
Vertical beam	0,0014	710	1,0	2	2,0
Horizontal beam	0,0021	710	1,5	1	1,5
Outer layer	0,0043	267	1,1	1	1,1
Total					**4,6**

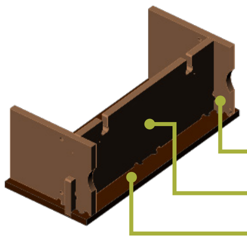


Figure 135: Axonometric 3D view updated section.

Table 33: Updated section of the wall component for the experiment.

Element	Volume [m3]	Density [kg/m3]	Mass [kg]	Amount [#]	Total mass [kg]
Vertical beam SMALLER	0,0010	710	0,7	2	1,4
Horizontal beam	0,0021	710	1,5	1	1,5
Outer layer SMALLER	0,0027	267	0,7	1	0,7
Total					**3,6**

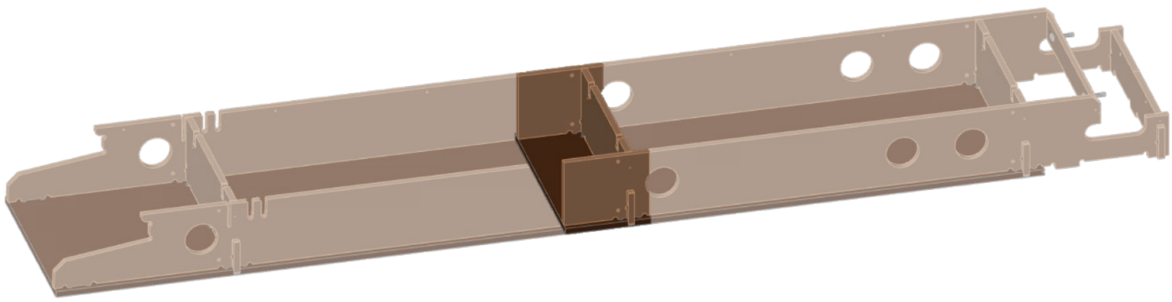


Figure 136: Axonometric 3D view of the whole wall component with the section for the experiment highlighted.

10.3 Extra tolerances

During the manual assembly process, a rubber mallet is required to hammer the elements in place. This is undesirable for the robotic process since it would require an extra end effector, which would complicate and slow down the robotic assembly process. Thus the goal is to assemble the section without the need to hammer the elements in place, which requires extra tolerances that allow the elements to slide smoothly. However, the tolerance should not be too high since it could cause the elements being to loosely connected, which results in a less rigid connection. Therefore, the tolerance should be just enough to enable robotic assembly without hammering. In order to increase the tolerance of the geometry an understanding of the different connections is required. This will enable identification of the potential areas in which extra tolerance can, and should be added.

Some of the assembly steps of the section are shown. Figure 137 shows how the vertical beams are slid in the outer layer. When this step is completed, the vertical beams can still move a few millimetres, as is marked with orange arrows in Figure 138. This figure also shows how the horizontal beam is slid in the vertical beams and the outer layer. Figure 139 shows the assembled section. To coop with the potential movement of the vertical beams the geometry of the horizontal beam is modified. In geometry version A - Original, all the corners had a chamfer of 3x3 mm. Increasing the chamfer on certain places would help to deal with the potential movement of the vertical beams. Figure 140 shows the horizontal beam with the increased chamfer in four locations. At the bottom, the original chamfer is increased with a factor 3, to 9x9 mm, because this chamfer needs to coop with the largest potential movement of the two vertical beams (and with 9 mm there is still 16-9=7 mm material left that rests against the side of the 16 mm deep dog bone slot). At the top, the original chamfer is increased with a factor 2, to 6x6 mm, because this chamfer needs to coop with less potential movement.

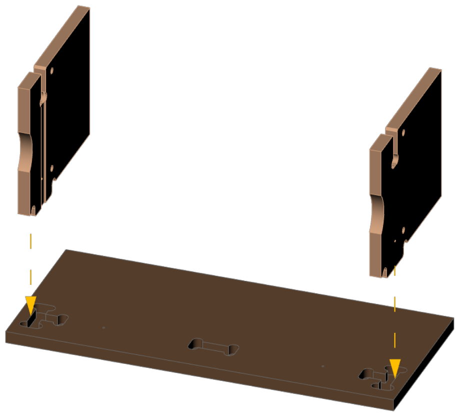


Figure 137: Axonometric 3D view of the vertical beams sliding in the outer layer.

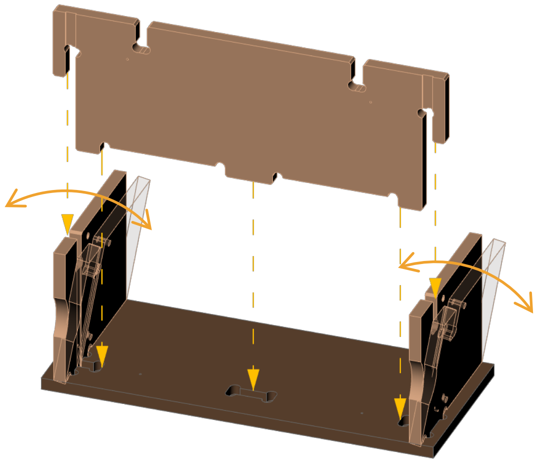


Figure 138: Axonometric 3D view of the horizontal beam sliding in the vertical beams. Plus orange arrows mark the slight potential movement of the two vertical beams.

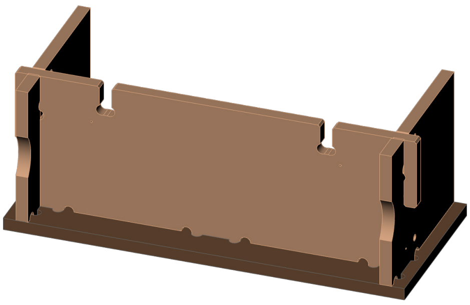


Figure 139: Axonometric 3D view of the assembled section.

Figure 140: Horizontal beam with the increased chamfers highlighted with orange circles.

Next to the increased chamfer, extra tolerances will be added and considered in further detail. The areas where extra tolerance can be added were localised, which resulted in three areas that can be seen in Figure 141. These three areas and the exact location of the tolerance are shown in Figure 142, Figure 143 and Figure 144.

The sheets out of which the elements are cut are 18 mm thick, but this can deviate up to 0,3 mm. To make the thickness accurate up to 0,1 mm (this is the tolerance of the CNC milling machine) a groove is made to ensure the thickness at the connection is 16 mm, marked with a T on Figure 143 and Figure 144.

Measurement P and Q are adjusted to find the optimal tolerance for the robotic assembly process. In geometry A - original, both measurements have 0 tolerance, meaning that P is 16 mm and Q is 18,3 mm. In order to be able to physically test different tolerances, five extra sections were produced with a measurement increment of 0,2 mm for P and Q. Photos of the six sections are placed in Appendix D: Six iterations of the section. Table 34 gives an overview of the tolerances of all six sections. It turned out that a margin of 0,8 mm enabled the UR5 to assemble it without requiring a rubber mallet to hammer the beams in place. Thus, measurement P is 16,8 mm and Q is 19,1 mm. These extra tolerances reduced the number of assembly steps even further, from 9 to 8 steps.

To prevent accidental mixing up any of the elements of the six sections, each element was engraved during the CNC milling with a number of circles corresponding to the section iteration number. The position of these circles was determined in collaboration with the HBO students and the design guideline of the DFA2 method was taken into account: “Shapes that can be used as means for orientation should be placed in the outer contours and preferably well visible” (Eskilander, 2001, p. 172). This ensured the circles would not hinder the assembly procedure and were clearly visible. In order to speed up the modelling process of the different elements, a Grasshopper algorithm including Python components was developed, which is revealed in Appendix E: GH modelling.

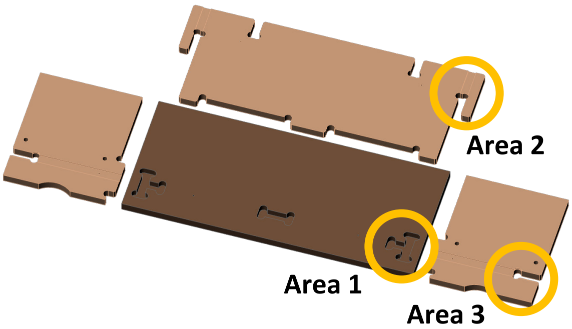


Figure 141: Axonometric 3D view of the section with the elements laid flat. The areas in which tolerances occur are highlighted with an orange circle and the area number.

Figure 142: Zoom in on area 1 showing the tolerances in the outer layer.

Figure 143: Zoom in on area 2 showing the tolerances in the horizontal beam.

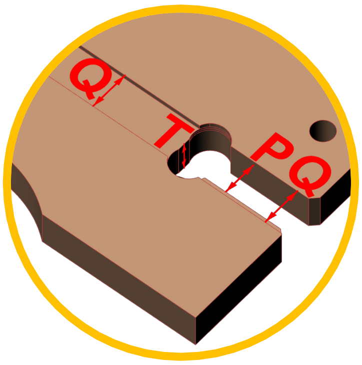


Figure 144: Zoom in on area 3 showing the tolerances in the vertical beam.

Table 34: Geometry margin of different section iterations.

Iteration	Margin [mm]	P [mm]	Q [mm]
1 (original)	0,0	16,0	18,3
2	0,2	16,2	18,5
3	0,4	16,4	18,7
4	0,6	16,6	18,9
5	0,8	16,8	19,1
6	1,0	17,0	19,3

10.4 Robotic process hypothesis

When looking at the robotic process hypothesis, which can be seen in Figure 145, there are five topics taken into account during the experiment. The focus area of the experiment is highlighted in brass with a blue dashed line. The HBO mechatronic students looked into further detail of the five experiment topics and described it in their report, which can be found in Appendix J: Dutch report HBO students. The five topics of the experiment are:

1) Detect elements. The robot should know where the elements are and if they are correctly assembled.
2) Pick up elements. An end effector should be able to assemble the elements and pick them up from both the CNC milling machine and the mobile platform.
3) Assemble elements. The robot should be able to perform the assembly actions. Extra hardware might be required for the assembly process.
4) Control robot arm. The robot arm can be controlled in three ways: manually by physically pushing it, via a teach pendant that basically is a remote control, or using software controlled via a different device such as a laptop.
5) Communicate. Refers to the streams of communication between: the operator & robot, the robot & the detection system, the robot & the end effector.

10.4.1 Not considered in the experiment

A robotic mobile platform that transports the elements from the unloading area to the assembly area is beyond the scope of the experiment. However, there are numerous robotic mobile platforms available that could perform. An example of a possible applicable robotic mobile platform is called Nipper, which is an intelligent pallet truck that is able to navigate autonomously and has a payload capacity of 1.000 kg. After charging for 10 minutes it is able to run for 8 hours straight, it has a top speed of 4 km/h and can be accurate up to approximately 1 mm (F3design, 2017). Due to its payload capacity, it is able to lift an euro pallet and about 31 complete wall components or 21 components of 46 kg, which is the maximum weight of a component of TNM. The number of components stacked on a single pallet will probably be lower than 21 since the stack is likely to become unstable due to its height.

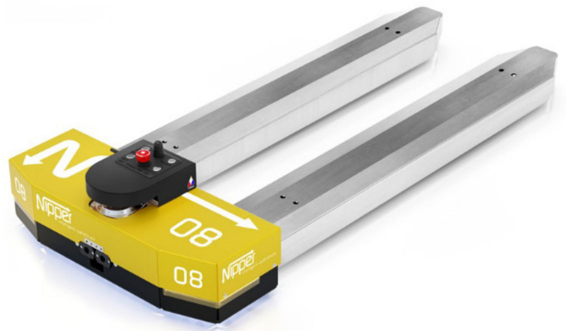


Figure 146: Nipper is an intelligent pallet truck. Reprinted from: F3design (2017).

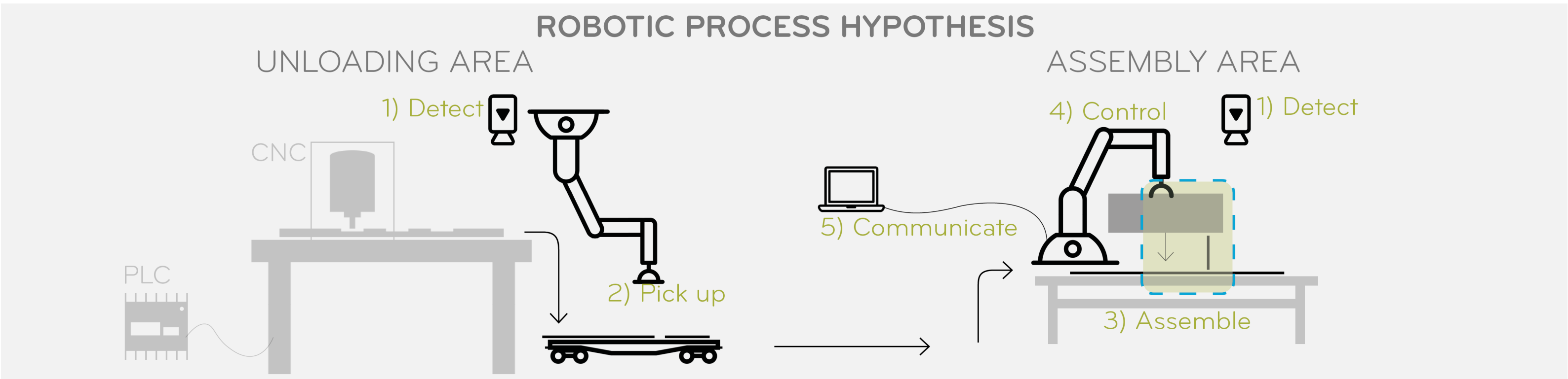


Figure 145: Robotic process hypothesis overview with the experiment aspects and focus area highlighted in a brass colour.

10.5 Layout

10.5.1 Robotic assembly

Figure 147 shows the layout for robotic assembly. It provided unobstructed movement for the UR5 cobot, which is only restricted by the cobot itself, the elements and the assembly table. During each experiment run a video was made from a similar position. The videos were later analysed in order to measure the number of seconds required for each assembly step.

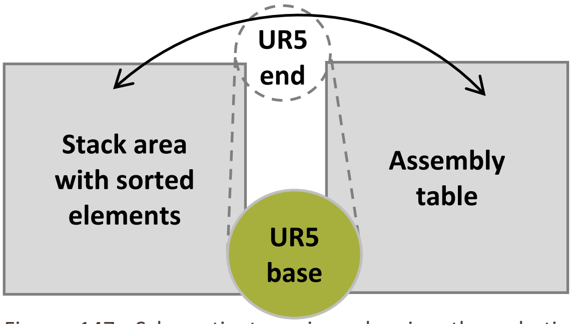


Figure 147: Schematic top view showing the robotic assembly layout for the section with the UR5 cobot.

The UR5 cobot used in the experiment can be seen in Figure 148. It has the following properties (Universal Robots, 2017):

• Payload:	5,0	kg
• Reach:	850,0	mm
• Mass:	18,4	kg
• Max velocity ±:	1,0	m/s
• Max velocity ±:	180,0	°/s
• Repeatability ±:	0,1	mm
• Base type:	Ø 149,0	mm
• Base area:	0,017	m2

Figure 148: UR5 cobot used in the experiment. Reprinted from (Universal Robots, n.d.-a).

10.5.2 Manual assembly

Just like the robotic assembly process, the manual assembly process is filmed and a similar layout was used that provides unobstructed movement as can be seen in Figure 149. In this case, a person stands next to a work table on which the elements are positioned on one side. The elements are sorted in exactly the same way as they are for the robot. Directly next to the elements there is space available to assemble the section.

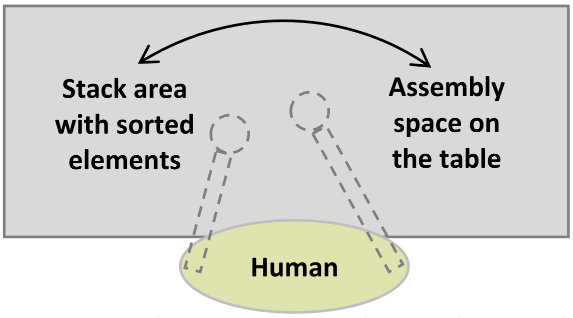


Figure 149: Schematic top view showing the manual assembly layout for the section and the standing location of a human worker.

10.5.3 Realistic time measurement

Some employees of TheNewMakers are very proficient in manually assembling the wall component because they have completed the assembly procedure many times. One skilled employee of TheNewMakers assembled the section in order to get a realistic time measurement. The assembly run was filmed and time squandered by incidental errors was subtracted to prevent errors influencing the time measurements. For example, during the manual assembly of the section, the battery of the electric screwdriver ran out of power. The time it took to change the battery was subtracted from the actual time.

For the robotic assembly method with the UR5, a similar approach was used to provide realistic data. The experiment run was filmed and the time required for the visions system to recognise the elements was subtracted. When a more decent computer is used, this time will be reduced to almost zero, therefore, it would be unfair to take this time into account.

10.6 Experiment findings

Besides the data explained in previous paragraphs, other findings were made during and at the end of the experiment. These are presented in this paragraph.

10.6.1 Robotic process hypothesis and the five experiment topics

The conducted experiment was based on the robotic process hypothesis and five topics were considered in particular, which are rated on a zero to five scale to represent the likelihood that the result is useful in the robotic process hypothesis. Where 0 means it was unable to determine if it is likely that the result is useful, 1 is very unlikely that the result is useful, 2 unlikely, 3 neutral, 4 likely and 5 is very likely. Table 35 shows the rating of all five experiment topics based on the following reasoning:

1) Detect elements. TensorFlow with deep learning, which is a form of artificial intelligence, is applied so the vision system is able to recognise which element it is looking at. Although the vision system is not accurate enough to pinpoint the location of the element within a few millimetres, it can be applied in the unloading area to recognise and pick elements from the CNC milling table.
2) Pick up elements. A vacuum gripper was designed, which proved to be a properly functioning approach. However, it might work less well in a dusty environment and it could not lift the porous outer layer.
3) Assemble elements. The robot is able to perform the required assembly steps and a slanted table was made and successfully applied. However, the screwing action was not tested.
4) Control robot arm. The robot arm was controlled by a combination of: receiving commands of TensorFlow; programming it via its teach pendant; manually guiding the arm. Manually a human is not capable of achieving the same level of accuracy as the robot. A simulation would be able to utilise the available accuracy of the robot.
5) Communicate. The UR5 robot, a camera, TensorFlow and the vacuum system are connected and able to communicate.

Table 35: Robotic process hypothesis rating based on the robotic assembly experiment with the section.

Experiment topics	Rating	Meaning
1) Detect	5	Very likely
2) Pick up	4	Likely
3) Assemble	5	Very likely
4) Control	3	Neutral
5) Communicate	5	Very likely

10.6.2 Geometry

The tested geometry modifications enabled robotic assembly. However, some of the CNC milled elements contain small defects, two examples are shown in Figure 150 and Figure 151. More photos of the sections can be found in Appendix D: Six iterations of the section. For a human, it is easy to deal with these imperfections, but they could potentially lead to a stagnation of the robotic assembly process since a robot might not be able to handle such irregularities.

Figure 150: Photo of the bottom part of the right vertical beam of section one. The small circle highlights a splinter sticking out. The big circle marks a very thin layer of wood that should have been milled out.

Figure 151: Photo of the right vertical beam of section four, which has splintering, or tearing, at the groove.

11 Robotic simulation

11.1 Aim

The aim of the robotic simulation is to test if the proposed robot is able to assemble the whole wall component. A simulation is made instead of a physical test, because the chosen robot, or any other robot of a similar size, was not available during the research.

Only one robotic simulation was made, because of two reasons. Firstly, it takes a considerable amount of time to program a robotic simulation. Secondly, and more importantly, one geometry version is redesigned for a robotic assembly process. This is geometry version B - DFA2, which was explained in chapter 9 Geometry, including the extra tolerances explained in paragraph 10.3 Extra tolerances.

Two software packages were considered in which the simulation could be made, being HAL-robotics and RoboDK. An initial test was conducted with the HAL-robotic plugin for Rhino and Grasshopper. A screenshot of this test can be found in Appendix F: GH test HAL-robotics. The test looks promising since the software appears to be functioning. However, as explained in subparagraph 4.4.2 Meso, the software is still in its beta phase and it contains a relatively small number of robots in its database. The user is able to add extra robots, but this would require very specific knowledge about the robot and it would take a substantial amount of time to do correctly. RoboDK, on the other hand, has a large database. Plus the software exists for a long time and has proven its quality and usability over the years. It is even referred to as "The future of robot off-line programming" (Nubiola, 2015). Due to its large database and proven track record, the robotic process simulation will be made in RoboDK.

11.2 Input

The following observations made during the experiment can be used input for the simulation. It provides the number of seconds required to:

1. Slide the vertical beams into the outer layer.
2. Slide the horizontal beam into the vertical beams.
3. Activate the vacuum gripper to attach a single element to it.
4. Deactivate the vacuum gripper to detach the element it was holding.

Next to these inputs, several boundary conditions are set for the simulation, which are:

1. Elements should be precisely fed to the robot. There are several methods possible to achieve this, two examples are: with an accurate camera that might be attached to the head of the robot or the slanted table that is presented in paragraph 4.5 Other hardware.
2. The only obstacles taken into account are: the floor, the assembly table, the elements, the robot arm and its end effector.
3. An actual 3D model that is able to perform the required screwing action is not applied. However, the screwing time is taken into account. The robot end effector will move to a specified location to pick up a screwing device, then it moves to the locations where a screw needs to be placed and here the end effector will perform the movement required in the amount of seconds it takes to complete the screwing action.
4. Forces on the elements, such as torsion, are not taken into account. A thorough analysis on this matter would make the scope of this research to big.

11.3 Robot movements

11.3.1 Kinematics

In order to make the simulation the movements of a robot arm need to be understood, which are called kinematics. In the "Springer handbook of robotics" by Siciliano and Khatib (2016) kinematics is described as follows:

> Kinematics pertains to the motion of bodies in a robotic mechanism without regard to the forces/torques that cause the motion. Since robotic mechanisms are by their very essence designed for motion, kinematics is the most fundamental aspect of robot design, analysis, control, and simulation. (p. 11)

The goal is getting the end effector to the desired location in 3D space, which is shown on the right side in Figure 152. All different angular joint positions combined, result in the end effector position, which is called forward kinematics and is rather easy to calculate. The other way around is also possible: calculating the joint positions based on the end effector location, which is called inverse kinematics and this is harder to calculate. This is mainly because several combinations of joint positions provide a valid solution to reach the desired end effector location (Avendano & Castro, n.d.). The chosen robot has six DOF, or Degrees Of Freedom, meaning it has six joints. Figure 153 reveals how the same end effector location can be reached by four different joint positions combinations. This particular end effector location can be reached by 42 different joint positions combinations.

11.3.2 Dynamics

Next step is to consider dynamics, which is "the relation between the applied forces/torques and the resulting motion of an industrial manipulator" (Melchiorri, 2014). Based on the angular position of each robot joint, every joint motor needs to provide a certain force and torque to reach the required robot joint angle.

11.3.3 Movement types

A robot arm can move its end effector from one point to the next with regards to the trajectory, which is called the toolpath. The toolpath can be generated by a linear or circular movement type. Another type does not takes into account the toolpath and only focuses on the joints of the robot, which is called a joint movement. This movement type uses the fastest way to change all joints from their current position to the required joint positions in order to reach the desired end effector location (Red, 1999).

It is important to note that movements within the simulation are either absolute, as per the world XYZ coordinate system, or relative, which means the movement is referencing to a frame, like the robot base or robot tool.

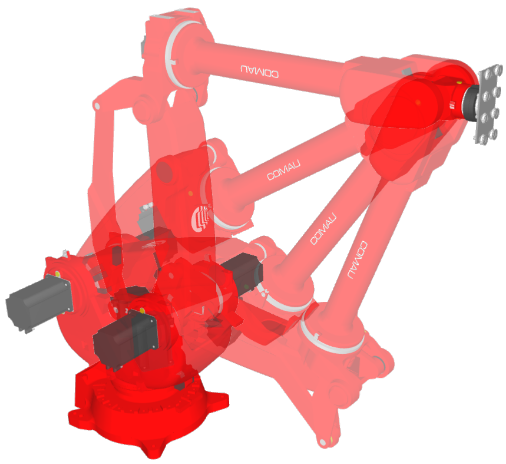


Figure 153: Four different combinations of joints of the robot arm to reach the same end effector location.

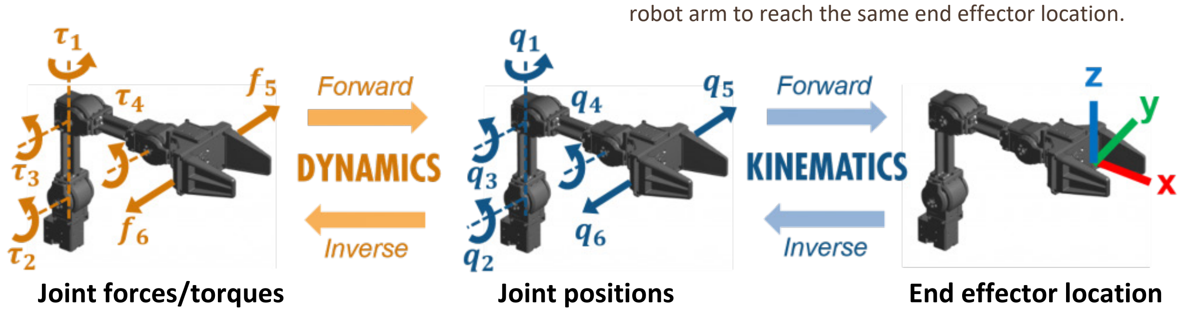


Figure 152: Relation between the joints and end effector position. Adapted from: Castro (2018).

11.4 The simulation

11.4.1 Elements start location

RoboDK refers to every single object as an item. An item can be a different type, such as: robot, geometry or location. In the simulation, the best suitable robot is applied, which is the Smart5 NJ 110-3.0 from Comau (2019b). The simulation basically consists of three phases: elements, screw and finalise. The simulation starts with all elements of the wall component laid flat as can be seen in Figure 154. In the real situation, there could be a stack of several elements on that location, but in the simulation, only the minimum number of elements required to assemble a single wall component is used.

The end effector approaches the start location of every element from the same absolute height of 500 mm, as highlighted with the orange dashed circle in Figure 155, and then moves down in a straight line until it reaches the element. This is required to ensure the end effector and the element that is being carried do not hit other elements. Furthermore, Figure 155 depicts the end result of the robotic simulation. The yellow line represents the toolpath over which the end effector moved during the simulation. This figure also shows an orange camera icon that marks the view location of Figure 156, which is presented on the next page.

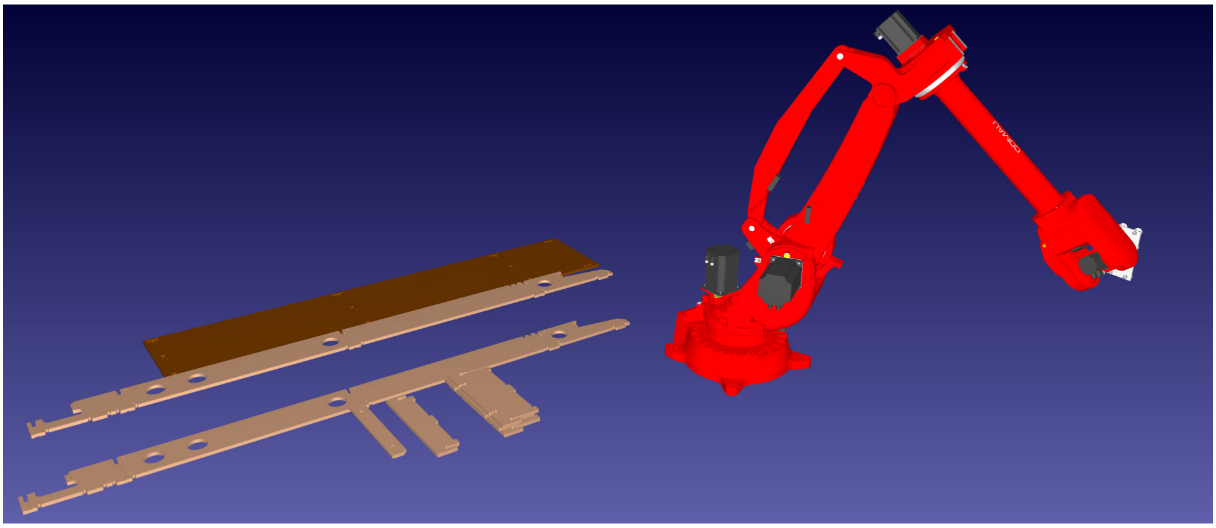


Figure 154: Robotic process simulation setup of the assembly process of the whole wall component in RoboDK.

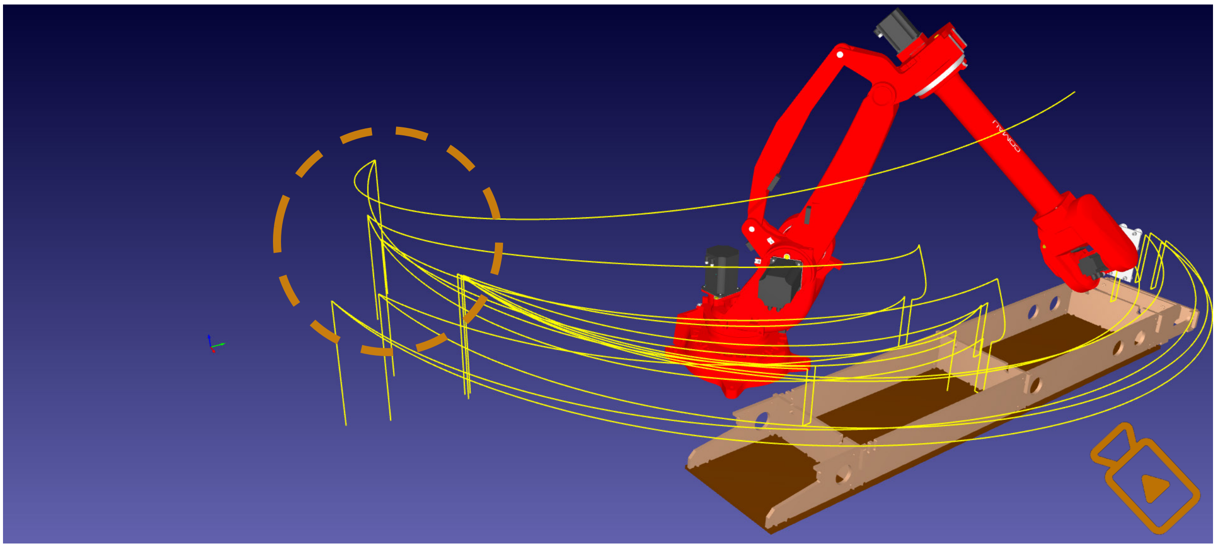


Figure 155: Result of all elements put together. The yellow line represent the toolpath of the robot during this part of the assembly process. The dashed orange circle marks the approach location above the flat laid elements. The orange camera icon marks the view location of Figure 156.

11.4.2 Elements final location

Figure 156 helps to understand how the end effector approaches the final location of an element and then leaves that location without hitting any elements. Step 1: The end effector is moved using a joint movement to the location 300 mm above the final location of the element. Step 2: The end effector makes a straight linear movement down and slides the horizontal beam in its final location. Step 3: The end effector moves 50 mm away from the element aligned with the main axis of the end effector. Step 4: Due to this 50 mm movement, the end effector can be moved up 300 mm without hitting other elements. Now the end effector can make a joint movement to approach the next element.

All beams need to be transferred from a horizontal to a vertical orientation. For the horizontal beams, this is no problem due to their relatively small size. However, for the big vertical beams, an extra movement is required to prevent the beam from hitting other elements or the robot. Figure 158 illustrates how the end effector transfers the right vertical beam from a horizontal to vertical orientation. Then it slides the beam in its final location with a linear movement.

When all elements are put together the first phase, called elements, is completed. It is followed by the screw phase, which will be explained on the other half of this page. The last phase, named finalise, simply consists of clearing the assembly table by moving the completed component to its final location.

11.4.3 Screw phase

When all elements are in their final location, they need to be screwed together. Figure 157 shows an automatic electric screwdriver with a screw head that activates when force is applied to it. It is designed to be used by people, but when placed in a custom made container the robot end effector can use it as well, without the need for a robotic tool change. It is efficient to apply an electric screwdriver that can be used by people as well. If the robot might not use it for a while then people can use the screwdriver. Plus, there is no need to acquire a specialised expensive end effector. It takes about 1 second to drill a screw in a wall (Festool, 2018).

Figure 157: Example of an auto-feed screwdriver, which has a force activated screw head. Adapted from: Festool (2019).

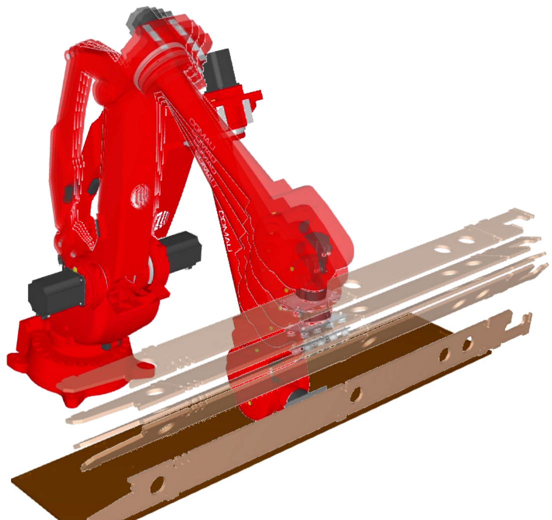


Figure 158: Chronophotographic image of the movement of the robot while the end effector slides the right vertical beam in its final location.

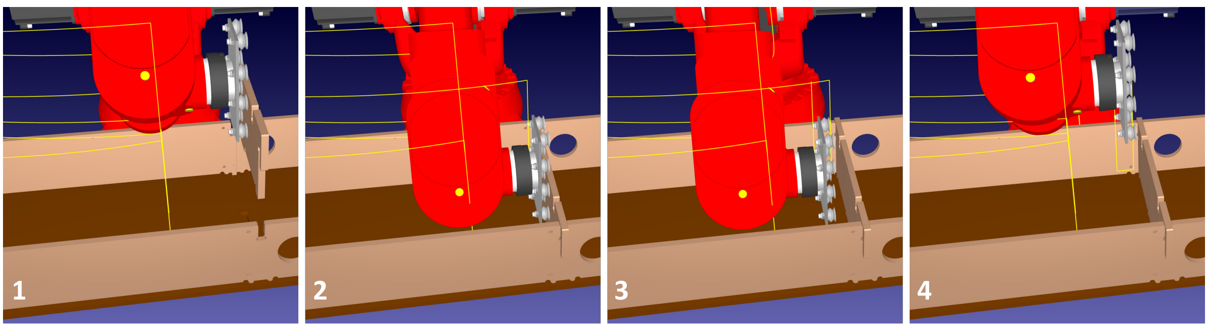


Figure 156: Steps of the robot arm to place the horizontal beam in its final location. Step 1 to 4 are shown from left to right.

11.4.4 Joint positions

The problem

As mentioned previously, for almost every end effector location there are multiple valid joint combinations possible. Due to all these different joint combinations, the robot arm can sometimes make movements that are undesirable. Several of such undesirable movements happened in the beginning of most simulations. One good example was during a test with the second best robot, the KR 120 R2900 extra from KUKA (2019b). The setup of the test with that robot can be found in Appendix G: RoboDK simulation test. Figure 159 presents the end result of this test and the toolpath clearly visualises how the robot arm made two big unintended movements that both look like a looping. Such movements are less likely to be a problem when the robot carries a small element. However, with the bigger elements, such unpredicted movements will most often result in the element hitting other elements or the robot arm. Some other unintended movements made a large element spin around viciously as if it was a two-blade propeller.

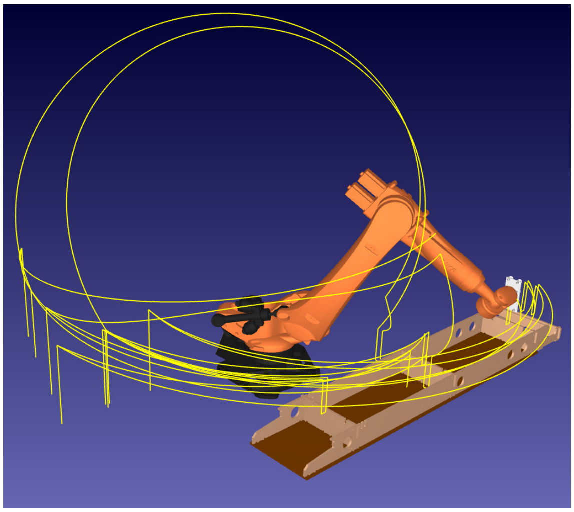


Figure 159: Result of all elements put together. The yellow line represent the toolpath of the robot during the assembly process. The big circles are clearly visible.

The solution

In order to prevent unwanted movements, the desired joint combination needs to be specified. On every location defined in RoboDK, the joint combination of the robot arm can be specified. When done correctly, the result is a smooth and safe movement as can be seen in Figure 160.

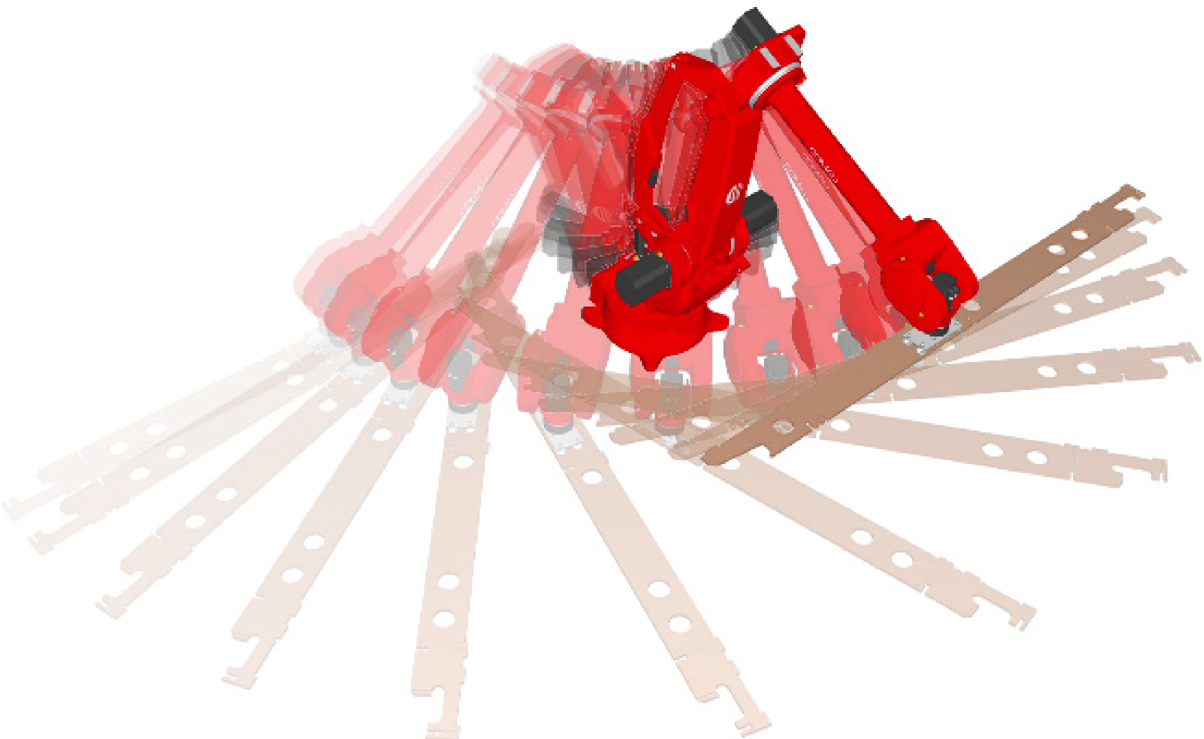


Figure 160: Chronophotographic image of the movement of the robot while the end effector carries the right vertical beam from its start approach location to the next location while keeping the element horizontally.

11.4.5 Collision detection

When the robot makes unintended movements as described on the previous page, the robot could slam the element it is carrying into itself or the carried element could collide with other objects. An example of the latter is marked in Figure 161. Such collisions should be prevented because they can cause serious damage to the robot, its payload or surroundings. In this simulation the collision detection was done manually and purely visually. When in doubt if a collision occurred during a particular movement, the simulation was slowed down and zoomed in on that area to make it clearly visible if a collision occurred or not.

RoboDK has an option to enable collision detection. The user can choose which objects in the simulation should be checked against each other for collisions detection. It could also be incorporated into the Python algorithm with a special function. Before a robot movement is made, this function first "checks if a . . . movement is feasible and free of collision" (RoboDK, 2019a).

11.4.6 Download

A video of the simulation is uploaded on YouTube and can be found by scanning the QR code in Figure 162 (the QR code redirects to the following URL: https://youtu.be/suKoIHKr0eg).

The Python script used for the simulation, which is explained later on, can be downloaded from GitLab via this shortened link: http://tiny.cc/gitlab (it redirects to this URL: https://gitlab.com/JBatGitLab/robotic-assembly-tnm/tree/master).

Figure 162: QR code to YouTube video of the simulation.

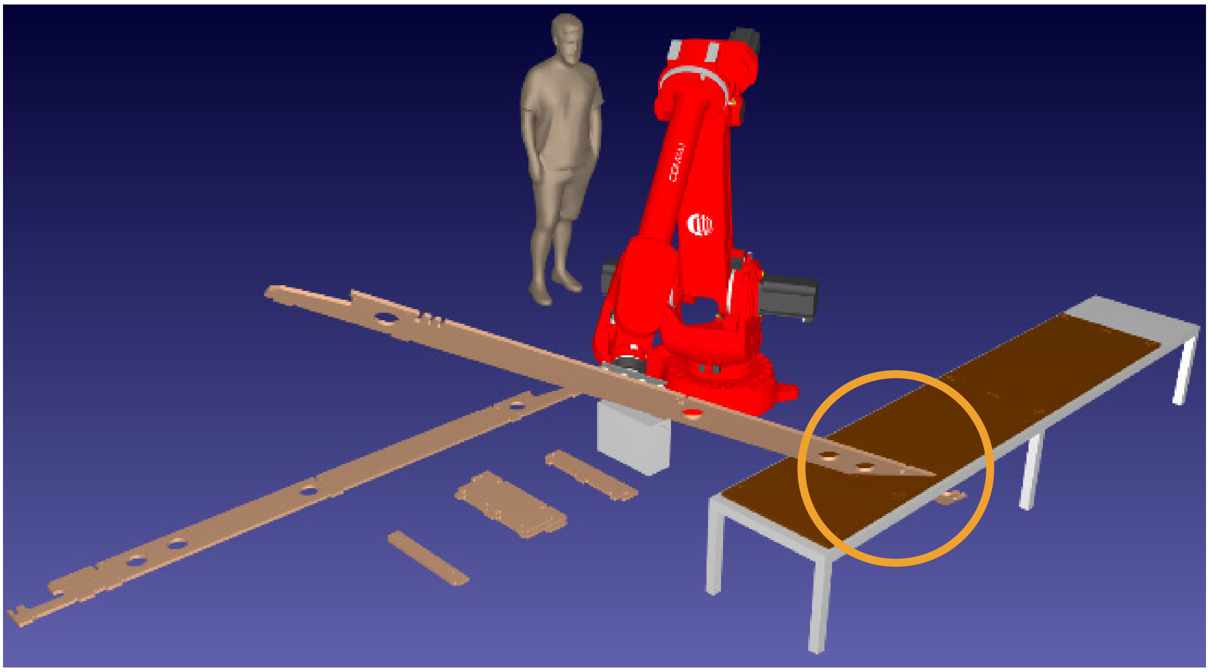


Figure 161: Snapshot of the vertical beam colliding with the table and the outer layer marked with an orange circle.

11.4.7 Force on elements

In the robotic simulation, the robot is able to move objects effortlessly. In reality, however, several forces are working on the elements when certain movements are made. As mentioned before, the forces are not taken into account. But, in order to illustrate this matter, one situation is described in which a too high torsion force might arise that maybe has the potential to damage the wall component.

Figure 163 shows a chronophotographic image of the simulation when the vacuum gripper is attached to the vertical beam and then the whole wall component is rotated while being completely airborne. The vertical beam will likely need to withstand a considerable amount of torsion force. If this torsion force would be too high, it could be reduced by rotating the whole wall component while it partially rests on the table as is presented in Figure 164.

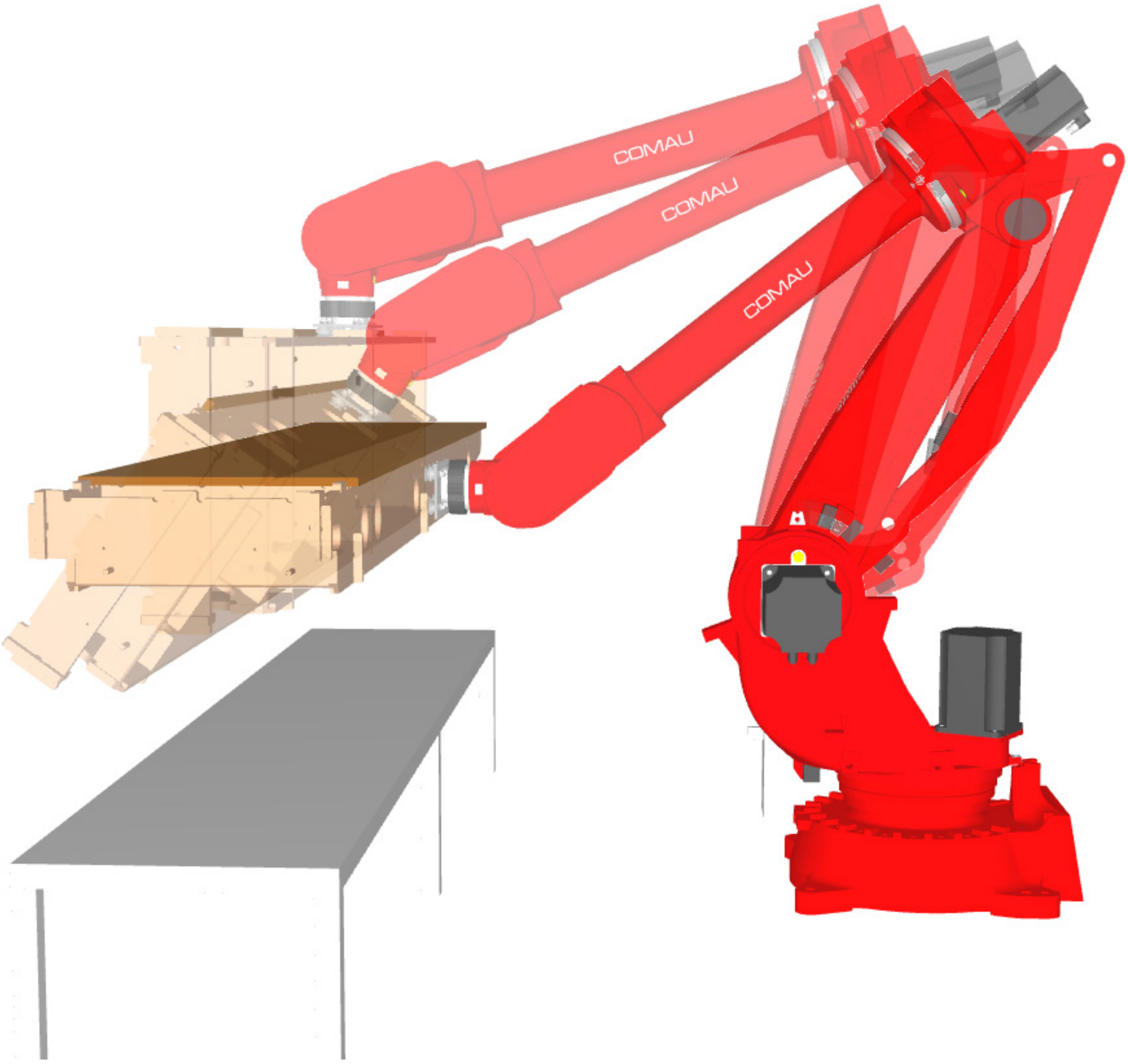


Figure 163: Chronophotographic image showing the movement of the robot while the end effector rotates the whole wall component in mid-air from a vertical to horizontal orientation.

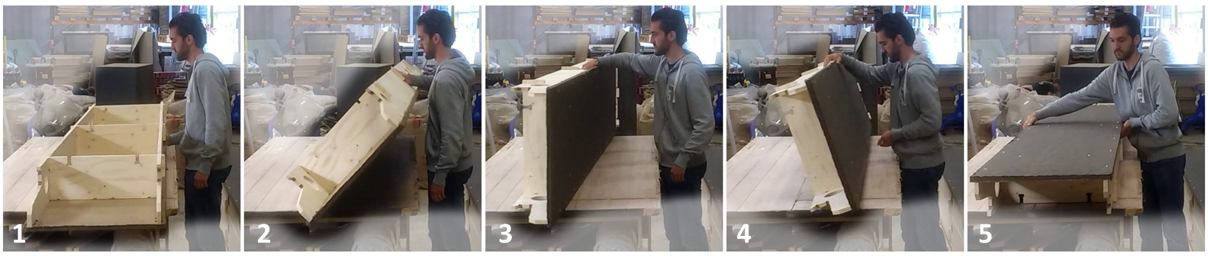


Figure 164: Manual steps to show how the movement could be carried out by the robot. Step 1 to 5 are from left to right.

11.4.8 Python script

In order to make the integration of RoboDK with other software easier, such as Rhino and Grasshopper, it was decided to control the simulation with a Python script. Python also provides the ability to create a dynamic assembly process with a generalised algorithm that could be applied to every component of TNM. Plus if the size of a component would change, the algorithm will adjust the movements of the robot arm accordingly.

In order to summarise the written algorithm and eliminate the need to scrutinise through the whole Python script, a pseudocode of the script is placed on the next page. But before the pseudo code can be understood, the logic behind the RoboDK item names needs to be clear.

Name logic

A logic for the RoboDK item names is required so the algorithm can recognise which items belong together. A single piece of geometry needs a minimum of two locations. The logic behind the item names is explained in more detail in Table 36. Table 37 presents an overview of all items used in the simulation.

When the elements are positioned on their final location screws should be drilled. This can be done by naming a location “SCREWS”. The Python script will send the robot arm to pick up the screwdriver and then goes to the location called “SCREWS”. Followed by going to all locations, and if “SCREW” is in its name a screw is inserted. If all elements need to be moved, a location should be named “ALL” so the script knows all geometry needs to be moved.

Table 36: Logic of the item names in RoboDK so the algorithm can recognise the locations that belong to the geometry.

RoboDK item	Name logic	Name explanation
The geometry of the element at its start location	ELM.[U].0.[V]	[U] = a number, starting from 0. It is used by the script to recognise the locations that belong to the geometry. 0 = the geometry should always have number 0. [V] = optional, an arbitrary name (e.g. The outer layer).
The locations over which the element is moved	ELM.[X].[Y].[Z]	[X] = [U] (script recognises the geometry and its locations). [Y] = a number, starting from 1 (except for special names, see explanation at [Z], which should start at 0). It refers to the start location where the robot will pick up the geometry. There are minimally 2 locations required, more are optional. [Z] = a special name: “SCREWS”, “SCREW” or “ALL”. Otherwise an arbitrary name (e.g. Start, Mid, Final, etc.).

Table 37: Overview of the item names of the geometries and their locations as used in the RoboDK simulation. [A]A total of 10 screw locations are added. [B]A total of 20 screw locations are added and it is possible to add extra move actions if needed.

Geometry or location	Location 1	Location 2	Location 3 and more are optional
ELM.0.0.Outer layer	ELM.0.1.Start location	ELM.0.2.Final location	
ELM.1.0.Right vert beam	ELM.1.1.Start location	ELM.1.2.Orientation	ELM.1.3.Final loc
ELM.2.0.Left vert beam	ELM.2.1.Start location	ELM.2.2.Orientation	ELM.2.3.Final loc
ELM.3.0.Horz beam 1	ELM.3.1.Start location	ELM.3.2.Final location	
ELM.4.0.Horz beam 2	ELM.4.1.Start location	ELM.4.2.Final location	
ELM.5.0.Horz bottom	ELM.5.1.Start location	ELM.5.2.Final location	
ELM.6.0.Horz floor	ELM.6.1.Start location	ELM.6.2.Final location	
ELM.7.0.Horz beam 3	ELM.7.1.Start location	ELM.7.2.Final location	
ELM.8.0.SCREWS	ELM.8.1.SCREW	ELM.8.2.SCREW	ELM.8.2.SCREW[A]
ELM.9.0.ALL	ELM.9.1.Flip it	ELM.9.2.Table location	
ELM.10.0.Extra move	(This single move ensures the robot arm does not hit other elements)		
ELM.11.0.SCREWS	ELM.11.1.SCREW	ELM.11.2.Move	ELM.11.3.SCREW[B]
ELM.12.0.ALL	ELM.12.1.Move	ELM.12.2.Finished	

Pseudocode

In the Python script, a reference is made to each numbered line of the pseudocode. So in script several notions of the text "PSEUDOCODE x" can be found, where x represents a number that is the same as the numbers here below. The whole Python script can be found in Appendix H: Python algorithm.

1. Provide input variables, such as the name of the robot.
2. Two functions, one to reset the simulation and another to drill a screw.
3. Import the required libraries, initiate the robot and call the function to reset the simulation.
4. Create a matrix, also known as a 2D list or a list of lists. Add all valid RoboDK items that start with "ELM." in their name to the right location in the matrix. Table 37 is a visualisation of this matrix.
5. Retrieve the reference frames the algorithm needs to work with and set the movements relative to the location of the assembled wall component.
6. Loop through all lists in the matrix that was created in step 4.
   - 6.1. A) Check if the first item in the list is geometry.
     - 6.1.1. Loop through all items in the list, except for the first one because that is the geometry.
     - 6.1.2. A) Check if it is the first or last step of this element.
       - 6.1.2.1. If it is the first step, set calculate the location above the element with an absolute height of 500 mm.
       - 6.1.2.2. If it is the last step, calculate the approach location 300 mm above the final location.
       - 6.1.2.3. Make a joint move to the calculated approach location then make a linear move to the location itself.
     - 6.1.3. A) Else make a joint move to the next element.
     - 6.1.4. B) Check if it is the first step of the element, if so: copy the geometry, hide the original geometry, link the copied geometry to the end effector and make a linear move up to the approach location previously defined.
     - 6.1.5. B) Check if it is the last step of the element, if so: connect the copied geometry to the assembled component reference, let the end effector move away without hitting other elements as explained in step 3 and 4 of Figure 156.
   - 6.2. B) Check if screws need to be added. Similar move actions are used as explained above in pseudocode 6.1. See the script itself for more details. Basically, the only different aspect is calling the "screw" function.
   - 6.3. C) Check if all geometry needs to be moved. Similar move actions are used as explained above in pseudocode 6.1. See the script itself for more details. Basically, the only different aspect is selecting the geometry of all elements.
   - 6.4. D) If it is not A, B or C, make a joint move to the first item in the list. The list should contain only one item, which is a location. This is useful because sometimes you would like the arm to make an extra move to avoid something.

11.4.9 Output

The following output resulted from the robotic simulation:

1. Robotic assembly time of the whole wall component and of every single assembly step.
2. The working area and height required for the robotic process.

The height factor mentioned in the second point is partially dependant on the movements of the robot. As explained before, if the joint combinations are not defined properly the robot can make big movements that might require much more free height.

12 Time

A time comparison will be made between the manual and robotic assembly method for the:

- Section, which was used in the physical experiment with the UR5 cobot.
- Whole wall component, which was used in the robotic simulation.

The CNC milling time of the different geometry versions of the whole wall component will be compared as well.

12.1 Section

This paragraph will compare the assembly time for the section by an employee of TheNewMakers and by the UR5 cobot. Towards the end of the experiment, the final geometry of the wall component was decided. This is geometry version B - DFA2, which was explained in chapter 9 Geometry, including the extra tolerances explained in paragraph 10.3 Extra tolerances. The time in seconds required to assemble the section of the wall component with this geometry version is measured for the robotic assembly method as well as the manual assembly method.

12.1.1 Assembly time

Based on the videos made during the experiment explained in chapter 10 Physical experiment, it is determined how many seconds it takes to assemble the section manually and with the UR5 cobot. The available results are presented in Table 38 and the data of each step, except for “Section TOTAL assembly time”, is graphically represented in Figure 165. Despite that “SUB TOTAL: Vertical beams” takes longer than 60 seconds, this figure is limited to 60 seconds in order to improve the readability.

The HBO students were able to make the UR5 cobot assemble the beams, which goes up to and including step 12. Therefore, only the time measured for these steps can be compared. Although the UR5 has an average repeatability and is not yet able to complete all the assembly steps, the physical experiment proves that the UR5 cobot is able to assemble the section with the redesigned geometry.

12.1.2 Section, compare time

Assembling the vertical beams with the UR5 cobot takes 82 seconds and just 12 seconds when done by a skilled employee of TNM. So a human is 6,83 (=82/12) times faster during these steps. The horizontal beam is assembled 3,69 (=48/13) times faster by an employee than the UR5. Although the UR5 is slow, the most important finding is that a robot is able to slide the elements of the section together.

Table 38: Overview of each assembly step for the section of the wall component with geometry version B - DFA2 that was used in the experiment.

Nr	Experiment section assembly method: / Experiment section steps	Manual	UR5
1	Move to stack area	1	7
2	Grab vertical beam 01	1	6
3	Move to assembly table	1	21
4	Slide in outer layer	3	4
5	Move to stack area	1	10
6	Grab vertical beam 02	1	6
7	Move to assembly table	1	22
8	Slide in outer layer	3	6
SUB TOTAL: Vertical beams		**12**	**82**
9	Move to stack area	1	9
10	Grab horizontal beam	1	6
11	Move to assembly table	1	21
12	Slide in vertical beams	10	12
SUB TOTAL: Horizontal beam		**13**	**48**
13	Move to screwdriver base	1	N/A
14	Grab screwdriver	1	N/A
15	Screw beams together	17	N/A
16	Move to screwdriver base	1	N/A
17	Place back screwdriver	1	N/A
SUB TOTAL: Screw beams		**21**	**N/A**
18	Grab screwed frame	1	N/A
19	Rotate screwed frame 180°	1	N/A
SUB TOTAL: Rotate frame		**2**	**N/A**
20	Move to screwdriver base	1	N/A
21	Grab screwdriver	1	N/A
22	Screw outer layer to frame	40	N/A
23	Move to screwdriver base	1	N/A
24	Place back screwdriver	1	N/A
SUB TOTAL: Screw outer layer		**44**	**N/A**
25	Grab assembled section	1	N/A
26	Move to stack area	1	N/A
27	Place section in stack area	1	N/A
SUB TOTAL: Move section		**3**	**N/A**
Section TOTAL assembly time		**95**	**N/A**

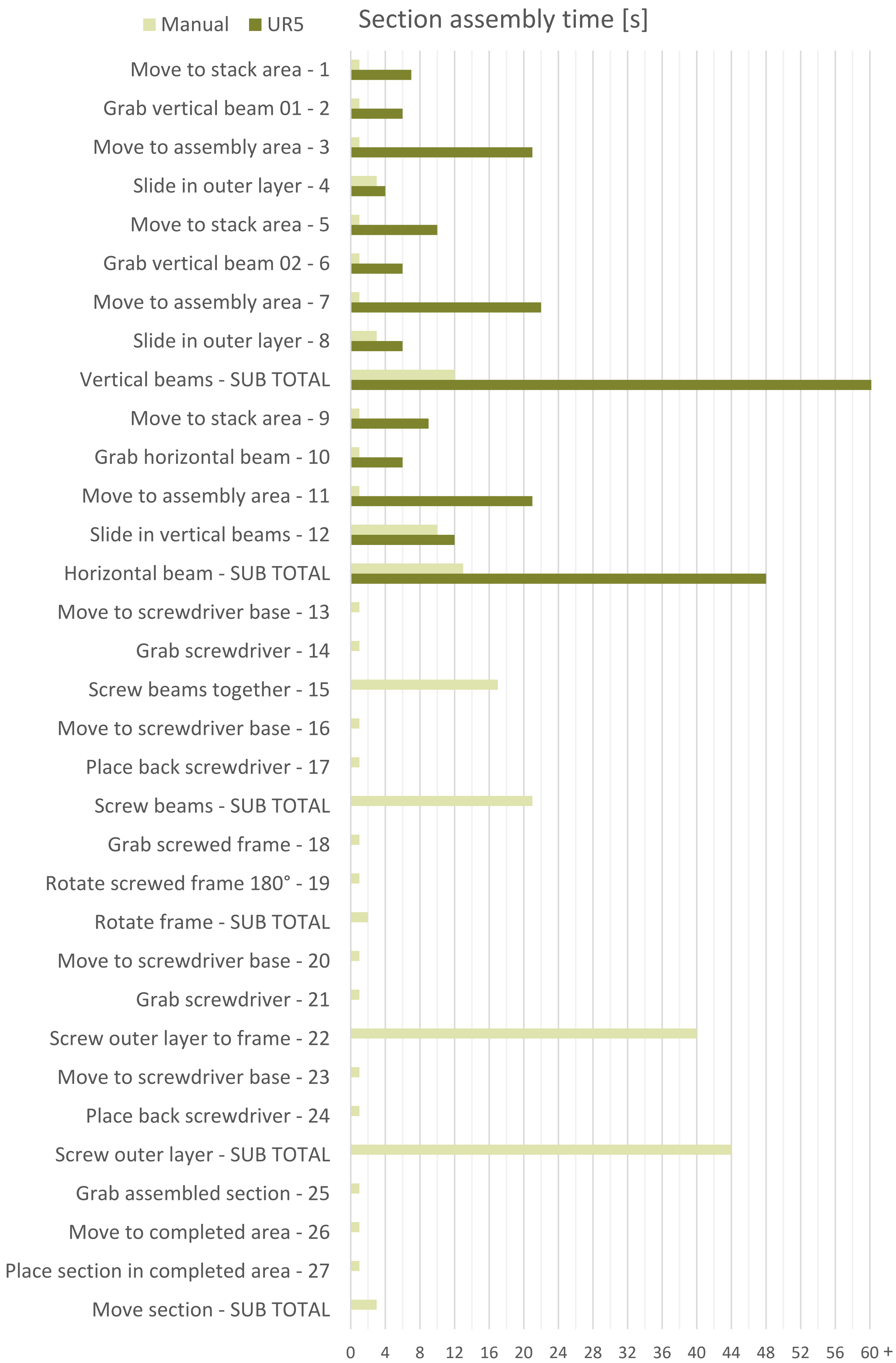


Figure 165: Graphic representation of the data in Table 38, except for “Section TOTAL assembly time”.

12.2 Whole wall component

This paragraph will compare the assembly time for the whole wall component by an employee of TheNewMakers and by the robotic simulation. The CNC milling time will also be compared. Both time comparisons are made between three geometry versions of the whole wall component:

- Version A - Original, which is the unmodified geometry of the wall component as to how it was designed and applied in the examined case study.
- Version B - DFA2 *without* extra tolerances, which is the redesigned geometry based on the guidelines of the Design For Automatic Assembly, or DFA2, method.
- Version B - DFA2 *with* extra tolerances, the same as the previous, but with extra tolerances at the connections as shown in paragraph 10.3 Extra tolerances.

The main reason for using both geometry B versions in the time comparison is to determine how much more seconds it takes to CNC mill version B - DFA2 *without* extra tolerances (variable Y in Table 39) as against version B - DFA2 *with* extra tolerances (variable Z in Table 39). If the CNC milling time of version B *with* extra tolerances will be significantly more, it is useful to consider other solutions instead of the extra tolerances required for the simulated robotic assembly method. Next to the main reason, this comparison also presents the time differences between both geometry versions for the assembly process.

The time comparison takes into account the two topics considered in the redesign of the geometry, being the assembly time and the CNC milling time. The number of seconds required to assemble any of the three geometry versions will be added up with the CNC milling time of that geometry version. It will determine the total amount of seconds required for each geometry version and corresponding assembly method. The following subparagraphs will elaborate on the details of Table 39. At the end of this chapter, the same table will be shown again but with all the data filled in.

The construction of the tiny house, which contains the wall component that is used as a case study, was completed before the geometry was redesigned. Therefore, the three versions of the wall component that were going to be made could not be used in this tiny house. Instead of making the wall components to gather the required data and then having to throw them away because there was no use for them, their height was adjusted with a few centimetres so they can be applied in a similar project that will be built in a few months. This slight height adjustment was implemented in all three geometry versions so the comparison will remain fair. Because the height adjustment was so small, it did not influence the assembly process. Therefore, the time required during the manual assembly method can still be compared with the time of the robotic simulation, which assembles a wall component that is not adjusted in height.

Table 39: Time comparison overview of the whole wall component of TheNewMakers.

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	U	V	W	R
2. CNC [s]	X	Y	Z	Z
1+2 = Total time	U+X	V+Y	W+Z	R+Z

12.2.1 Assembly time

The assembly time of the whole wall component is considered for the manual assembly method and the simulated robotic assembly method.

Manual assembly method

Some employees of TheNewMakers are very proficient in manually assembling the wall component because they have completed the assembly procedure many times. One skilled employee of TheNewMakers assembled all three geometry versions (A, and B *with*, or *without* extra tolerances) of the whole wall component in order to get an authentic time measurement. The setup of this manual assembly procedure is comparable to the robotic process simulation. Figure 166 presents an overview of this setup when the last wall component was going to be assembled and the orange circles mark different areas.

The three geometry versions of the whole wall component were assembled manually and the number of seconds measured during this process is shown in Table 40. It took:

- 456 seconds for version A - Original.
- 389 seconds for version B - DFA2 *without* extra tolerances.
- 366 seconds for version B - DFA2 *with* extra tolerances.

Robotic simulation assembly method

As explained in chapter 11 Robotic simulation, the robotic simulation was made based on geometry version B *with* extra tolerances. It took the robot in the simulation 187 seconds to assemble the whole wall component.

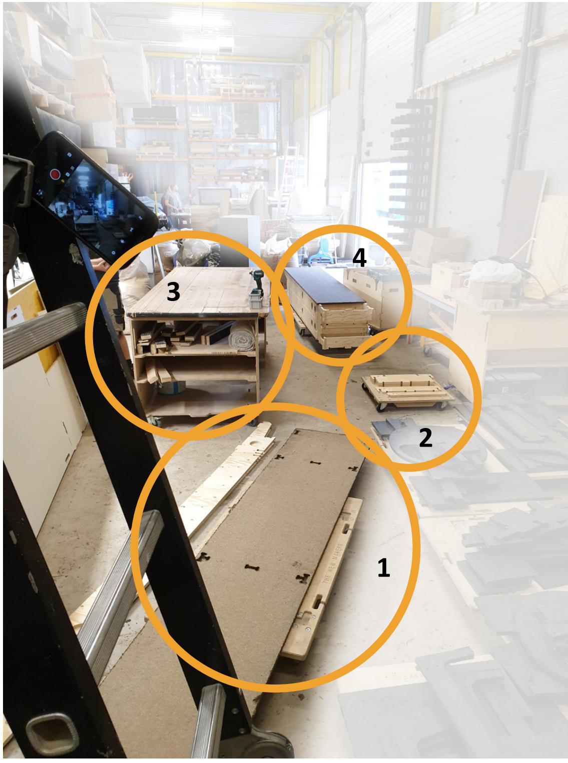


Figure 166: Manual assembly setup of the whole wall component in the factory of TheNewMakers. Circle 1 marks the stack of large elements, circle 2 marks the stack of small elements, circle 3 marks the assembly table and circle 4 marks the stack of finished wall components.

Table 40: Assembly time comparison for manual and robotic assembly of the whole wall component.

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	456	389	366	187

Version B - DFA2 with extra tolerances

Since geometry version B - *with* extra tolerances is the only version that is assembled using both assembly methods it is compared step by step.

Based on the videos made during the manual assembly of the whole wall component, it is determined how many seconds each assembly step takes. The data from the robotic simulation was extracted from RoboDK. The results are presented in Table 41 and each step, except for "Whole comp. TOTAL assembly time", is graphically represented on the next page in Figure 167. Despite the fact that some steps take longer than 60 seconds, this figure is limited to 60 seconds in order to improve the readability. The manual "Move to stack area" steps take 0 seconds because during the manual process multiple elements are picked up at once and already placed on the assembly table so there is no need to move to the stack area again.

It is important to state that one type of step had a big impact on the assembly time, which was the number of seconds required to drill the screws. During the manual assembly process, a standard cordless screwdriver was used, whereas the robotic simulation applied an automatic electric screwdriver. The latter tool works much faster because the screws are fed automatically. As opposed to the first tool, where a human needs to pick every single screw from a container and position it on the screwing head. The time difference caused by the two tools can be clearly seen in step 35 and 42.

Table 41: Overview of each assembly step of the whole wall component of geometry version B - DFA2 *with* extra tolerances. The table extends on the next half of this page.

	Whole wall component assembly method:	Manual	Robotic simulation
Nr	**Whole wall component steps**		
1	Move to stack area	2	1
2	Grab outer layer	2	4
3	Move to assembly table	4	3
4	Place on table	1	1
SUB TOTAL: Outer layer		**9**	**9**
5	Move to stack area	4	3
6	Grab vertical beam 01	2	2
7	Move to assembly table	2	2
8	Slide in outer layer	19	3
9	Move to stack area	0	3
10	Grab vertical beam 02	2	2
11	Move to assembly table	0	2
12	Slide in outer layer	20	3
SUB TOTAL: Vertical beams		**49**	**20**
13	Move to stack area	3	3
14	Grab horizontal beam 01	2	3
15	Move to assembly table	2	2
16	Slide in vertical beams	9	1
17	Move to stack area	0	3
18	Grab horizontal beam 02	2	3
19	Move to assembly table	0	2
20	Slide in vertical beams	8	1
21	Move to stack area	0	3
22	Grab horizontal bottom beam	2	3
23	Move to assembly table	0	2
24	Slide in vertical beams	10	1
25	Move to stack area	0	3
26	Grab horizontal floor beam	2	3
27	Move to assembly table	0	2
28	Slide in vertical beams	12	1
29	Move to stack area	0	3
30	Grab horizontal beam 03	2	2
31	Move to assembly table	0	3
32	Slide in vertical beams	5	1
SUB TOTAL: Horizontal beams		**59**	**45**
33	Move to screwdriver base	1	2
34	Grab screwdriver	1	2
35	Screw beams together	76	27
36	Move to screwdriver base	1	4
37	Place back screwdriver	1	3
SUB TOTAL: Screw beams		**80**	**38**
38	Grab screwed frame	2	3
39	Rotate screwed frame 180°	8	15
SUB TOTAL: Rotate frame		**10**	**18**
40	Move to screwdriver base	1	3
41	Grab screwdriver	1	2
42	Screw outer layer to frame	147	38
43	Move to screwdriver base	1	2
44	Place back screwdriver	1	2
SUB TOTAL: Screw outer layer		**151**	**47**
45	Grab assembled whole component	2	4
46	Move to completed area	4	4
47	Place whole comp. in compl. area	2	2
SUB TOTAL: Move whole component		**8**	**10**
Whole comp. TOTAL assembly time		366	187

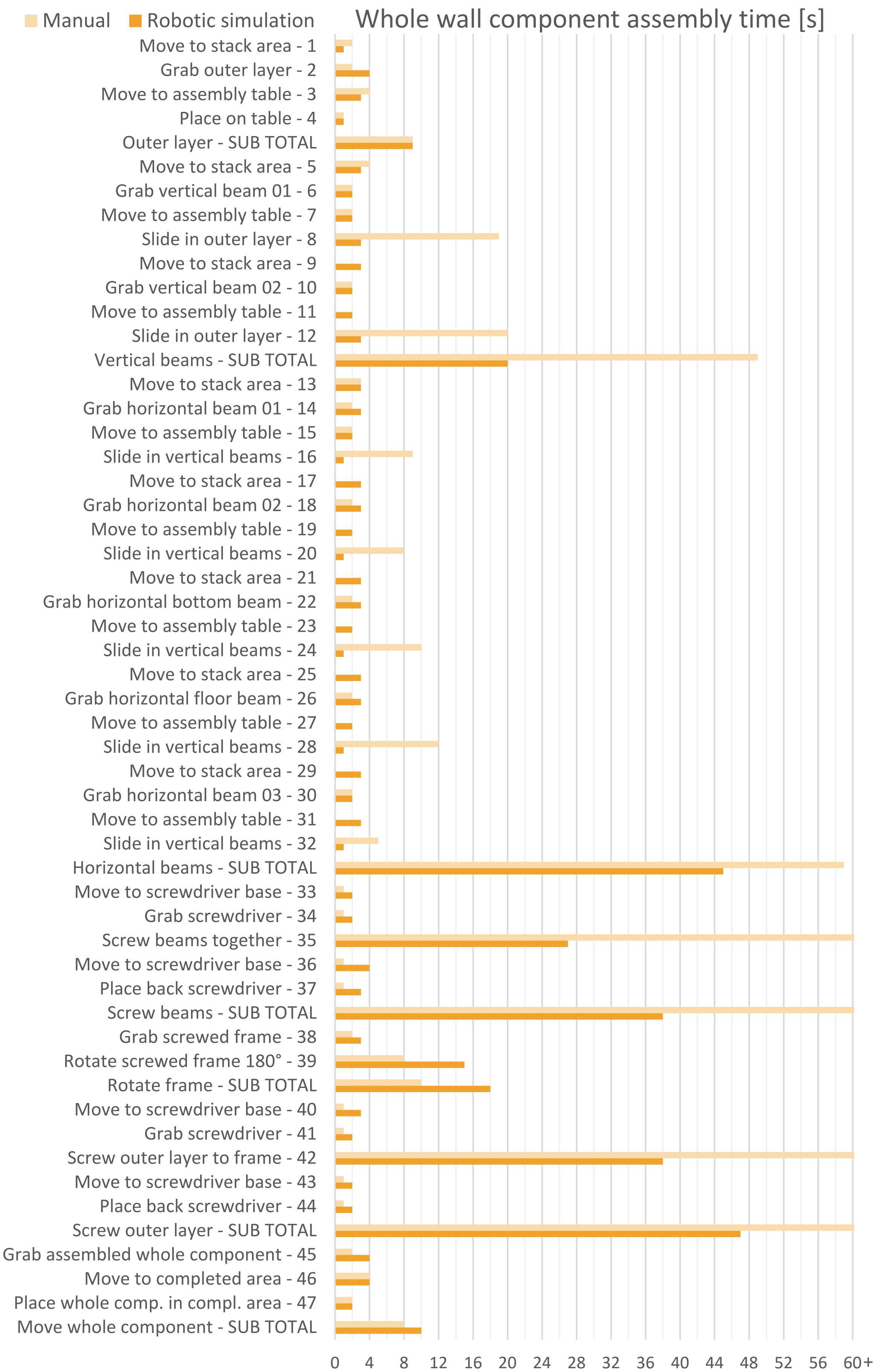


Figure 167: Graphic representation of the data in Table 41, except for “Whole comp. TOTAL assembly time”.

12.2.2 CNC milling time

The three geometry versions of the whole wall component were CNC milled and the number of seconds required during the CNC milling process was measured. As shown in Table 43, it took 1.493, 1.497 and 1.483 seconds to CNC mill geometry version A, and B *with*, or *without* extra tolerances, respectively.

During the CNC milling process, there should be only one changing variable, which was any one of the three geometry versions. All other variables, such as the materials, the CNC milling machine and its settings, remained the same. Figure 168 presents a photo of the CNC milling machine, which is a Biesse Rover G5.12. A stopwatch was started at the moment when the CNC milling head moved from its base position. The time measurement was stopped when all elements were milled and the milling head was returned at its base position.

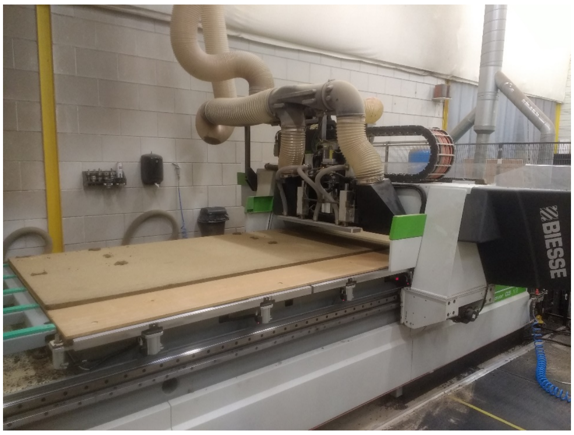


Figure 168: Photo of the CNC milling machine while it is milling the outer layer of geometry version A - Original.

Two variables slightly influenced the CNC milling time. The first could have been prevented by milling the required elements for each component from exactly the same amount of sheets. However, this would have significantly increased the amount of waste produced. Therefore, it was chosen to group and nest all horizontal beams of the three versions on a single sheet and all vertical beams on a different sheet as shown in Appendix I: Drawings for CNC milling. This slightly influenced the CNC milling time because the first and last movement of the CNC milling head from and to its base position were different for each version. The other variable that slightly influenced the CNC milling time was noticed when analysing the data. It was due to a minor difference in nesting of the horizontal beams for geometry version A, as opposed to the other two versions.

Table 42 shows the CNC milling time in seconds for each sheet and all three geometry versions. The total is calculated and added in Table 43. Comparing the CNC milling time between the three versions shows they only deviate 24 (=1493-1469) seconds at most.

Table 42: Overview of the CNC milling time per sheet of elements and per geometry version.

Version: / Sheet	A	B *without* extra tol.	B *with* extra tol.
Vertical beam	728	728	732
Horizontal beams	408	400	400
Outer layer	357	351	337
Total	**1.493**	**1.479**	**1.469**

Table 43: CNC milling time comparison of the whole wall component.

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	456	389	366	187
2. CNC [s]	1.493	1.479	1.469	1.469

12.2.3 Whole wall component, compare time

The total time consists of the assembly time and the CNC milling time. First, the number of seconds required for the assembly process will be discussed. Followed by the time required to CNC mill the elements.

1. Assembly [s]

Table 44 shows how many seconds it took to manually assemble the following geometry versions: A took 456 seconds, B *without* extra tolerances took 389 seconds and B *with* extra tolerances took 366 seconds. This means that geometry version B - DFA2 *with* extra tolerances is 1,25 (=456/366) times faster to assemble manually than geometry version A - Original.

The whole wall component based on geometry version B *with* extra tolerances was assembled during the robotic simulation in 187 seconds. This means it is 2,44 (=456/187) times faster than manual assembly with the original geometry. However, it is important to state that different electric screwdrivers were used in the robotic simulation and in the manual process.

Besides, it should not be forgotten that a robotic process hypothetically could run continuously, if there are enough elements to assemble. So on an annual base, the robot would be able to run for 8.760 hours (= 365 days * 8 hours) while a fulltime human worker would make about 1.872 hours (= 52 weeks * 5 days * 8 hours - 20 vacation days - 6 national holidays). Meaning the robot is able to work a factor of 4,68 (=8.760/1.872) more than an employee. Of course, influences such as robot maintenance and employee sick days should be taken into account to make this calculation more realistic.

2. CNC [s]

Table 44 shows the CNC milling time of the three different geometries. Version A was milled in 1.493 seconds. Version B *without* extra tolerances, was milled in 1.497 seconds and version B *with* extra tolerances, took 1.483 seconds to mill. These times are very close to each other and only deviate a maximum of 24 seconds, which is only 1,63% (=24/1.469) of the shortest CNC milling time measured.

It is interesting to note the CNC milling time for geometry version B *with* extra tolerances is milled the quickest. The geometry modifications that might have caused this reduced milling time are not investigated because it is outside the scope of this research.

Total time

Table 44 is visualised in Figure 169 and both show that the CNC milling process is much more time consuming than the assembly process. The robot will be standing still most of the time due to a lack of available elements that can be assembled. The manual assembly process with the improved geometry of version B *with* extra tolerances is 4,01 (=1.469/366) times faster than the CNC milling process. The robotic assembly simulation is even faster, it is 7,93 (=1.469/187) times faster than the mill process.

The total time for the robotic simulation including the CNC milling process is 1,18 (=1.949/1.656) times faster than total time required for the original process. The original process considered geometry version A, combined with a manual assembly method.

Table 44: Filled in time comparison overview of the whole wall component of TheNewMakers.

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	456	389	366	187
2. CNC [s]	1.493	1.479	1.469	1.469
1+2 = Total time	1.949	1.868	1.835	1.656

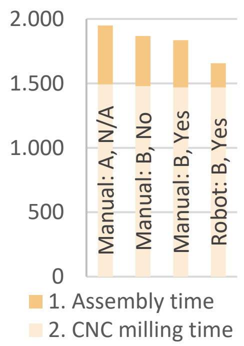


Figure 169: Visual representation of Table 44.

CONCLUSIONS AND RECOMMENDATIONS

Figure 170: A UR3 collaborative robot can work together with a human. Adapted from: Latmas (n.d.).

Conclusions

The conclusions are drawn based on the four sub-questions and are resulting in an answer to the main research question.

Q1) What does the current manual assembly process of a wooden wall component of TheNewMakers look like?

The current manual assembly process of a wooden wall component of TNM has been investigated physically, during an on-site case study day as well as with the robotic assembly experiment, and digitally, using the 3D Rhino file of the wall component. Three assembly detail principles were examined, which are:

1. Slide, which refers to sliding one element into another.
2. Screw, which considers fastening elements together using screws.
3. Dog bone, which refers to pushing an element into a dog bone connection.

It looks like the wall component is designed to suit the available CNC milling process of TNM. Plus, it is made in such a way to be suitable for manual assembly by two people.

Q2) How can the geometry of the CNC milled elements be redesigned to reduce the automatic assembly time as well as the CNC milling time?

The geometry was redesigned and aimed to reduce the time required for robotic assembly and CNC milling. The Design For Automatic Assembly method, or DFA2 method (Eskilander, 2001), was used as a guideline to modify the geometry of the wall component to make it suitable for a robotic assembly process.

The geometry modifications enabled automatic robotic assembly and made the assembly sequence easier. In combination with the extra geometry tolerances, certain steps of the process can be skipped plus some steps are performed faster and are less prone to failure.

Q3) What kind of robot arm is best suited to the robotic assembly process of TNM?

An overview was made of different robots divided into two types, being industrial robots and cobots. They were compared in terms of the following six criteria: payload, reach, velocity, mass, repeatability and estimated Total Cost of Ownership (TCO).

Out of this comparison, the following robot scored the highest and it was concluded that is able to automatically assemble the wooden wall component of TNM: Smart5 NJ 110-3.0 by Comau (2019b). The robot has a: payload capacity of 110 kg, reach of 2.980 mm, a mass of 1.070 kg, the joint speed lies between 115° and 292° per second and the repeatability is about 0,07 mm.

Q4) Does the proposed robotic assembly process take less time than the manual process?

A time-based comparison was made between the manual assembly process and robotic assembly process. This comparison considered the three geometry versions of the whole wall component, which are: version A - Original, version B - DFA2 *without* extra tolerances and version B - DFA2 *with* extra tolerances. This is coupled with a comparison between the number of seconds required to CNC mill the elements of the three different geometry versions.

The simulated robotic assembly process is 269 (=456-187) seconds faster than the manual assembly process. The CNC milling time of the geometry required for the robotic assembly process, which is geometry version B *with* extra tolerances, takes 1.469 seconds to mill. This is 24 (=1.493-1.469) seconds less compared to the original geometry (geometry version A). However, it is not investigated what caused this 24 seconds decrease in time.

How could the elements of a wall component of TheNewMakers be redesigned for automatic assembly while taking less time to assemble than the manual process?

With the redesign guidelines of the Design For Automatic Assembly (DFA2) method the wall component is redesigned for automatic assembly. There are three geometry versions:

- Version A - Original, which is the unmodified geometry.
- Version B - DFA2 *without* extra tolerances, which is the redesigned geometry. The number of assembly steps are reduced from 15 to 9.
- Version B - DFA2 *with* extra tolerances, the same as the previous, but with extra tolerances at the connections. This allows assembly without a hammer and reduced the assembly steps to 8.

In order to answer the main question, the time required for the two assembly methods are compared based on the improved geometry version. So the simulated robotic process with geometry version B - DFA2 *with* extra tolerances is 1,96 (=366/187) times faster than manual assembly method of the same geometry. Assuming the robot is able to work continuously for a year, it can work a factor 4,68 more than a fulltime employee. It would make the annual production of the robot 9,16 (=1,96*4,68) times higher as opposed to a human worker.

Table 45 is visualised in Figure 171 and both show that the CNC milling process is much more time consuming than the assembly process, which would cause the robot to stand still most of the time. The robotic assembly simulation is 7,93 (=1.469/187) times faster than the CNC milling process of geometry version B - DFA2 *with* extra tolerances.

The total time taken during the robotic simulation including the CNC milling process is 1,18 (=1.949/1.656) times faster than total time required for the original geometry with the manual assembly method.

Despite these very positive results, one important remark needs to be made, which is about the fact that different types of electric screwdrivers were used in the two assembly methods. In order to be able to make a fair comparison with the available data, the screwing time is temporarily disregarded. When the time required for the screwing steps are omitted for geometry version B - DFA2 *with* extra tolerances, the total time needed for the manual assembly method is reduced from 366 to 143 seconds and the robotic simulation time is reduced from 187 to 122 seconds. So the robotic process is still 1,17 (=187/122) times faster. Hypothetically, it would mean the annual production of the robot is 5,48 (=1,17*4,68) times higher than a fulltime human employee.

Table 45: Filled in time comparison overview of the whole wall component of TheNewMakers.

Assembly method:	Manual			Robotic simulation
Q2 geometry version:	Version A - Original	Version B - DFA2 (Design For Automatic Assembly)		
Extra tolerances:	N/A	no	yes	yes
1. Assembly [s]	456	389	366	187
2. CNC [s]	1.493	1.479	1.469	1.469
1+2 = Total time	1.949	1.868	1.835	1.656

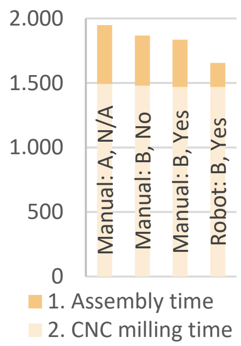


Figure 171: Visual representation of Table 45.

Recommendations

Robotic process, next step physically

As explained in paragraph 3.5 Approach and methodology, certain aspects of the methodology in the book "Lean Robotics: a guide to making robots work in your factory" by Bouchard (2017) are applied in this research. According to this methodology, a robotic process consists of three phases: design, integrate and operate (Bouchard, 2017). The proposed robotic process resulting from this research lies close to the end of the design phase. Some aspects described in this chapter need to be worked out in more detail before the next step, meaning the integration phase, can be taken. It is recommended to briefly exploring the integration phase to already develop an understanding of how the next step looks like.

Robotic process, next step digitally

A robotic simulation was made to measure long the robotic assembly process would take to assemble the wall component. It could be improved by adding collision detection. But the next big step is to integrate the simulation in the workflow of TheNewMakers. This integration would enable a link between the component of TNM and the assembly sequence of the robotic process. If this link is set up correctly, the dimensions of the specific component can change and automatically update the assembly sequence of the robot. This assembly sequence can then be digitally tested and verified.

The robotic simulation was made in a program called RoboDk, which provides a plugin for Rhino and Grasshopper. TNM is working with Rhino and Grasshopper. So it is suggested to integrate the robotic assembly simulation in the current workflow of TNM using RoboDK and its plugin for Rhino and Grasshopper.

It is also recommended to improve and optimise the toolpath. Besides the toolpath, the presented Python algorithm should be developed further and generalised. When made correctly, a dynamic assembly process arises that can be used for every component of TNM.

Robotic process, unloading area

The focus of this research was on the assembly area. It is recommended to further develop the other area as well, which is the unloading area that considers the unloading process of the elements from the CNC milling machine.

Robotic process, compare manual process

The factor time was compared between the robotic assembly process and the manual assembly process. The factor time consisted of the assembly time and the CNC milling time. To make the comparison fair and complete the assembly time should be measured again when the manual assembly method uses the same electric screwdriver as in the robotic simulation. Also, it is advisable to compare the two processes on other factors. Some topics that could be meaningful to investigate are: Total Cost of Ownership, productivity and efficiency.

Robotic process, end effector

Designing an end effector was beyond the scope of this research. But, the end effector has a big influence on the assembly process and TCO. For example, the HBO mechatronic students applied a vacuum gripper. Such a gripper requires a constant flow of air. The costs linked to keeping this airflow going can be significant. Plus, the designed vacuum gripper was not able to grip the outer layer of the wall component because it is made of a porous wood fiberboard called Celit 18S. A mechanical gripper does not come with the same costs required to provide a constant flow of air, but it has other costs and limitations. Thus, it is advised to design an end effector and further investigate the details of different available solutions. These details could entail: capabilities, speed and TCO.

Robotic process, robot table

The robot in this research was a robot arm. It might also be possible to apply a robot table in the assembly process. This suggested robot table should be able to clamp the elements to the table and then rotate 180 degrees about the longest axis of the wall component. This would reduce the required reach of the robot arm since it can drill all screws from the same side instead of reaching over the wall component. It might increase the speed of the process. The table needs to be designed in order to further investigate the speed and TCO.

Robotic process, multiple robot arms

At the beginning of this research, two limitations were set. One mentioned using only one robot arm. Multiple robot arms will add extra complexity to the assembly process, as explained in subparagraph 8.1.2 Assembly area.

Despite the extra complexity, the TCO of a robotic process with multiple robot arms could be lower because the sum of the TCO of several smaller robots might be lower compared to the TCO of a single bigger robot.

The assembly time could also be reduced. For example, two robot arms enable the possibility to move two elements at once instead of one. When properly synchronised the assembly process could be nearly twice as fast. So robot A can pick up the first element. When robot A reaches the first element robot B starts to move towards the second element. Robot A places the first element on its final location. Then robot B start moving the second element to its final location and robot A can pick up element number three. The two robots are also able to place screws simultaneously if two screwdrivers are provided.

Figure 172 presents how a robotic process with two robot arms might look like. The elements are sorted in two stacks so both robots have their own stack of elements. The arrowheads indicate that robot A is moving an element towards the assemble table and robot B is approaching the stack of elements in order to pick up a new element.

Robotic process, not fixed

The second limitation mentioned in the beginning of this research stated that the robot arm will be fixed to the floor. If this limitation is cleared the robot arm could, for example, be fixed to the ceiling or placed on a rail. The latter adds an extra Degree Of Freedom, which allows to choose a robot with a shorter reach, which significantly increases the possible robot choices. A different robot with a shorter reach could decrease the TCO of the process, however, the rail also comes with certain costs and it is a bit more complex to make the simulation due to the interaction between the rail and the robot arm.

Figure 173 shows that even the robot arm with the highest payload capacity can be placed on a rail. It might also be the heaviest robot, which indicates that it is likely that any robot arm can be placed on a rail. However, this has to be investigated further before such a conclusion can be drawn.

Figure 173: Industrial robot M-2000iA/2300 with a payload capacity of 2.300 kg and a mass of 11.000 kg moves over a rail. Reprinted from: AT-Aandrijftechniek (2019).

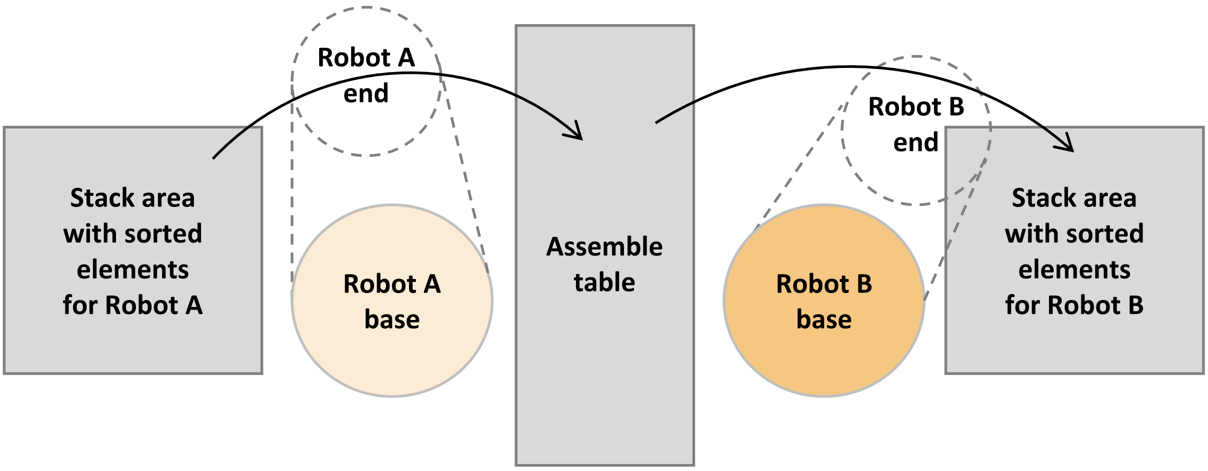


Figure 172: Proposed schematic top view showing a robotic assembly layout for the whole wall component with two robots.

Total Cost of Ownership

In this research, the Total Cost of Ownership, or TCO, was only considered in general. It is recommended to do more research towards the TCO of different suitable robots. This could be done by contacting robot suppliers and discuss the proposed robotic process with them.

CNC milling time

The main focus of this research was to reduce the robotic assembly time. The data shows that the CNC milling time for this specific case study is much more time consuming. The robotic assembly simulation is almost 8 times faster than the CNC milling process.

So it is advised to investigate how the milling of elements can be sped up to prevent a potential robot from standing still most of the time. There are many solutions conceivable and two are mentioned here. It might be achieved by increasing the output of the current tool, which is the CNC milling machine. However, it is unlikely that it will be able to increase its output per hour by a factor 8 but acquiring extra machines might be a solution. Another potential solution could be to use a CNC router as an end effector for the robot arm. Then the robot does not need to stand still and is less dependent on human workers or others machines since the robot arm covers both processes alone, from CNC milling its own elements to assembling them into a component.

Since the CNC milling time has such a big influence on the total time, it would help to have a clear understanding of the time required for different milling actions. This knowledge could be used in the design process of the elements. Software to simulate the CNC milling process would show the milling time required without the need to use the actual machine, which would take up valuable CNC milling time for testing. An employee of “De Groot”, which is the supplier of the CNC milling machine of TNM, advised using “bSolid” (Biesse, 2019). At first glance, the software looks very promising and able to simulate the CNC milling time. However, more research has to be conducted towards the suitability and implementation of bSolid in the workflow of TheNewMakers.

Geometry

Applying different chamfer solutions could influence the CNC milling time. It might also slightly reduce the robotic assembly time. In this research, only one improved chamfer solution was applied over a single axis. However, other chamfer solutions are possible, such as a round one, under a different angle or a second chamfer over the other axis. The latter might improve the assembly process further and make the proposed process even less prone to failure.

The geometry of CNC milled elements sometimes contained small defects. Extra care should be taken in the CNC milling process to prevent these because they could hinder or even stall the robotic assembly process.

Structure wall component

Although there is no indication it is the case, the structural integrity of the wall component might be compromised due to the geometry changes. Structural tests should be conducted on geometry version B *with* extra tolerances to verify its strength, stiffness and stability.

If the structural integrity of the wall component is compromised, measures should be taken to ensure the structure is safe and sound. The measures depend on the cause of the potentially compromised structural integrity. If, for example, it is caused by the tolerances there are different solutions possible. The following two solutions are given for inspirational purposes. The tolerances could be reduced by using special connectors that pull the elements together, such as a cam lock screw. Another potential solution could be to remove the tolerances and design a robotic end effector that is able to hammer the elements in place, which is how it was done in the original manual assembly process. A robot arm is able to perform hammering actions as was presented by the pneumatic hammer end effector shown in paragraph 4.3 End effector. The hammering required for the assembly process needs to be more subtle than the hammering action of that demolition robot. More research has to be conducted to investigate if there is a structural problem to begin with, followed by investigating what a suitable solution might be.

A new construction method

Due to the limitations of current technology, it is more efficient to apply a robot for a specific task instead of it trying to mimic humans. As shown in the introduction, the robot specialised in making hamburgers is much more efficient than the robot arms that are trying to cook.

A similar approach would be advised for the building industry. So instead of robots that struggle to thrive in an environment suited to human capabilities, like the humanoid HRP-5 that installs drywall and the bricklaying robot SAM-100, there are construction methods designed from a robotic point of view. Two examples are: the 3D concrete house printing robot used in Project Milestone and the robot by Odico that preforms Robotic Hot-Wire Cutting to create moulds for concrete prefab elements. Both examples helped to realise unique buildings. Linking the design of the building with a robotic process is in line with the ideas of Frank Lloyd Wright, as he once wrote

> If I was to realize new buildings I should have to have new technique. I should have to so design buildings that they would not only be appropriate to materials but design them so the machine that would *have* to make them could make them surpassingly well. (James & Wright, 1968, p. 34)

Several people believe that robots are required on the Dutch building sites due to two factors. First, there are not enough construction workers in the Netherlands. In 2018 there was a shortage of 48.000 skilled labourers. Second, the pressure on the Dutch housing market is high and will stay high, because a million new homes need to be built before 2030.

Instead of applying robots on building construction sites, there are companies, such as TheNewMakers and Gilles Retsin, who want to radically improve how buildings are being constructed. These improvements will result in a much simpler and faster on-site construction method, which would reduce the need for either skilled labourers or on-site construction robots that do not thrive in a working environment specifically tailored for people.

The existing components of TNM have very interesting features that can be incorporated into a new construction method. Plus the most prominent esthetical feature of Retsin his method, being the positioning of components under a 45-degree angle, is likely to improve the robotic assembly process. This idea of positioning components under a 45-degree angle might also have potential on an element level. For example, by giving the element a double 45-degree edge as shown in Figure 174. This probably makes it is easier to position and slide the element in a groove of another element. However, the CNC milling process should be able to support the required operation to create such geometry or a different geometry modification process could be considered.

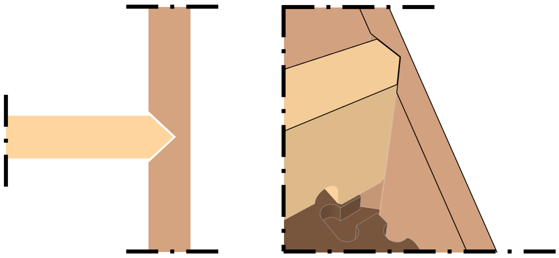


Figure 174: Two fragments of an element with a double 45-degree edge sliding in one with a double 45-degree groove. Left the plan view and right a 3D view with a clipping plane.

The following statement is presented in the DFA2 thesis, which suggests applying the presented DFA2 method and its guidelines in the initial design phase:

> Usually, DFA analysis is performed only when the design details are known and the product is more or less finished. As a result, designers tend to view a DFA analysis as an extra step or burden (Hsu, Jerry Fuh, & Zhang, 1998). . . . A DFA analysis should be used to guide the designer or the product development team in the initial search for a “good” design. (Eskilander, 2001, p. 59)

To conclude: it is recommended to design a new construction method based on a robotic assembly process and inspired by the presented construction methods of TheNewMakers and Gilles Retsin. The DFA2 design rules should be used as a guideline from the start of the design.

References

3Dprintedhouse. (2018). Project Milestone. Retrieved from https://www.3dprintedhouse.nl/en/project-info/project-milestone

ABB. (2001). Product Specification - IRB 6400R. Retrieved from https://library.e.abb.com/public/4b47c4bc123fe9e8c1257b4b0051f5a7/IRB_6400R_Rev3.pdf

ABB. (2018, October 27). ABB to build the world's most advanced robotics factory in Shanghai. Retrieved from https://new.abb.com/news/detail/9410/abb-to-build-the-worlds-most-advanced-robotics-factory-in-shanghai

ABB. (2019, January). IRB 8700 - High payload robot. Retrieved from https://library.e.abb.com/public/c351681e07f84c8cb9fb4135c4e02fa6/IRB8700-9AKK106713A0323-Rev.C.pdf

AIST. (2018). Development of a Humanoid Robot Prototype, HRP-5P, Capable of Heavy Labor. Retrieved from https://www.aist.go.jp/aist_e/list/latest_research/2018/20181116/en20181116.html

Altintas, Y., & Tulsyan, S. (2015). Prediction of part machining cycle times via virtual CNC. *CIRP Annals, 64*(1), 361-364. doi:https://doi.org/10.1016/j.cirp.2015.04.100

AMCI. (n.d.). What is a PLC? Retrieved from https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/what-plc

Arboportaal. (n.d.). Hoeveel mag een werknemer tillen? Retrieved from https://www.arboportaal.nl/onderwerpen/tillen-en-dragen/vraag-en-antwoord/hoeveel-mag-een-werknemer-tillen

Asimov, I. (1950). *I, Robot*. London: D. Dobson.

AT-Aandrijftechniek. (2019, May 7). Grootste Fanuc robot verplaatst over 100 meter. Retrieved from https://www.at-aandrijftechniek.nl/algemeen/%EF%BB%BFgrootste-fanuc-robot-verplaatst-over-100-meter/60493

Aubo Robotics. (n.d.). AUBO-i10: Collaborative Lightweight Robot. Retrieved from https://www.motoman.com/hubfs/Collaborative/HC10.pdf?hsLang=en-us

Automatica. (2019, Ápril 2). Automatica - The Leading Exhibition for Smart Automation and Robotics Retrieved from https://automatica-munich.com/index-2.html

Avendano, J., & Castro, S. (Producer). (n.d.). Controlling Robot Manipulator Joints Video. *MATLAB and Simulink Robotics Arena*. Retrieved from https://mathworks.com/videos/matlab-and-simulink-robotics-arena-controlling-robot-manipulator-joints-1521714030608.html

Baek, S. (2016, November 11). Demystifying cobot safety: The four types of collaborative operation. Retrieved from https://blog.universal-robots.com/demystifying-cobot-safety-the-four-types-of-collaborative-operation

Bekey, G., Ambrose, R., Kumar, V., Sanderson, A., Wilcox, B., & Zheng, Y. (2006). International assessment of research and development in robotics. *World Technology Evaluation Center Panel Report*.

BergInsight. (2017). *Summary: The global service robotics market*. Retrieved from http://www.berginsight.com/ShowReport.aspx?m_m=3&id=262

Biesse. (2019). bSolid. Retrieved from https://www.biesse.com/ww/wood/software/bsolid

Boston Dynamics. (2018a, December 28). Atlas | Boston Dynamics. Retrieved from https://www.bostondynamics.com/atlas

Boston Dynamics. (2018b, October 11). Spot Robot Testing at Construction Sites. Retrieved from https://www.youtube.com/watch?v=wND9goxDVrY

Bouchard, S. (2017). Lean Robotics: a guide to making robots work in your factory.

Bremmer, D. (2018, November 15). Bouw moet robotiseren: tekort aan bouwvakkers is blijvend. Retrieved from https://www.ad.nl/economie/bouw-moet-robotiseren-tekort-aan-bouwvakkers-is-blijvend~aaad9f09

Castro, S. (2018, April 25). Robot Manipulation, Part 2: Dynamics and Control. Retrieved from https://blogs.mathworks.com/racing-lounge/2018/04/25/robot-manipulation-part-2-dynamics-and-control

Chamfer. (n.d.). In *Cambridge English Dictionary*. Retrieved from https://dictionary.cambridge.org/dictionary/english/chamfer

Characteristic. (n.d.). In *Oxford English Dictionary*. Retrieved from https://en.oxforddictionaries.com/definition/characteristic

Cobot. (n.d.). In *Oxford English Dictionary*. Retrieved from https://en.oxforddictionaries.com/definition/us/cobot

Comau. (2019a). NJ-110-3.0: characteristics and technical specifics. Retrieved from https://www.comau.com/EN/our-competences/robotics/robot-team/nj-110-30

Comau. (2019b, February). Robotics Compact Catalogue. Retrieved from https://www.comau.com/Download/robot/BROCHURE/Comau_Robotics_Compact%20Catalogue.pdf

Comau. (n.d.). AURA - Advanced Use Robotic Arm. Retrieved from https://www.comau.com/Download/our-competences/robotics/Automation_Products/Folder_Aura%20Doppie.pdf

Computer Hope. (2017, 10 November). What is N/A (NA)? Retrieved from https://www.computerhope.com/jargon/n/na.htm

Construction Robotics. (n.d.). SAM100: Semi-Automated Mason. Retrieved from https://www.construction-robotics.com/sam100

Criterion. (n.d.). In *Cambridge English Dictionary*. Retrieved from https://dictionary.cambridge.org/dictionary/english/criterion

Dattalo, A. (2018, August 8). ROS Introduction. Retrieved from http://wiki.ros.org/ROS/Introduction

Dhanani, S. (2015). *Realizing Industry 4.0: essential system considerations*. Retrieved from https://www.maximintegrated.com/en/an5991

DUHAN robotics. (n.d.). Brabo Robotic Solution. Retrieved from https://www.indiamart.com/proddetail/brabo-robotic-solution-11719085788.html

ElMaraghy, H. A. (2009). Changeable and reconfigurable manufacturing systems. London: Springer.

Eskilander, S. (2001). *Design For Automatic Assembly - A Method for Product Design: DFA2.* (Doctoral thesis), Royal institute of technology, Stockholm.

ESPS. (2018). Kassow Robots - ESPS | Hands-on Robotimation. Retrieved from https://esps.nl/products/kassow

ETH. (2019, March 22). Robotic collaboration in timber construction. Retrieved from https://www.ethz.ch/en/news-and-events/eth-news/news/2018/03/spatial-timber-assemblies.html

F3design. (2017, October 6). Nipper intelligent pallettruck. Retrieved from https://www.dinostretchhood.com/wp-content/uploads/2017/11/20171006_NipperSpecsheet_EN_lr.pdf

FANUC. (2001). S-430i. Retrieved from https://www.gentic.nl/upload/custom/pdf/S-430i_Serie.pdf

FANUC. (2017). M-2000iA/2300. Retrieved from https://www.fanuc.eu/~/media/files/pdf/products/robots/robots-datasheets-en/m-2000ia/datasheet%20m-2000ia-2300.pdf?la=en

FANUC. (2018a, October). Collaborative Robot: Robot CR-15iA. Retrieved from https://www.fanuc.co.jp/en/product/catalog/pdf/robot/RCR15iA(E)-02.pdf

FANUC. (2018b, October). Collaborative Robot: Robot CR-35iA. Retrieved from https://www.fanuc.co.jp/en/product/catalog/pdf/robot/RCR-35iA(E)-02.pdf

Farm Bot. (n.d.). FarmBot | Open-Source CNC Farming. Retrieved from https://farm.bot

Festool. (2018, June 22). Cordless drywall screwdrivers DWC 18-4500 | DWC 18-2500 - Festool. Retrieved from https://www.youtube.com/watch?v=Z8gjgmUm8OA#t=7s

Festool. (2019). Cordless construction screwdriver DWC 18-4500 Li-Basic DURADRIVE. Retrieved from https://www.festool.com/products/drilling-and-screwdriving/cordless-drywall-screwdrivers/574747---dwc-18-4500-li-basic

Ford, M. (2018). Architects of intelligence : the truth about AI from the people building it.

Frey, C. B., & Osborne, M. A. (2017). The future of employment: How susceptible are jobs to computerisation? *Technological Forecasting and Social Change, 114*, 254-280. doi:https://doi.org/10.1016/j.techfore.2016.08.019

Geertsma, P. (2018, May 23). Bouw heeft 48.000 vakmensen tekort in 2018. Retrieved from http://www.technischwerken.nl/nieuws/bouw-heeft-48-000-vakmensen-tekort-in-2018

Gekas, P., & Perera, K. (2007, May 30). *Economic Justification for Industrial Robotic Systems.* Lecture at the National Innovation Robotics Competition.

Geometry. (n.d.). In *Dictionary.com*. Retrieved from https://www.dictionary.com/browse/geometry

Government Of The Netherlands. (n.d.). Secondary vocational education (MBO) and higher education. Retrieved from https://www.government.nl/topics/secondary-vocational-education-mbo-and-higher-education/higher-education

HAL-robotics. (2019). HAL-robotics framework. Retrieved from https://hal-robotics.com

HMK robotics. (n.d.-a). TM12 & TM14 Series. Retrieved from https://www.hmkrobotics.com/images/1_products/actuators___robotcs/cobots/1tm12/robot_specs.pdf

HMK robotics. (n.d.-b). TM14 Collaborative Robot: 1100mm Reach & 14kg Payload. Retrieved from https://www.hmkrobotics.com/robots/tm14

Hsu, W., Jerry Fuh, Y. H., & Zhang, Y. (1998). Synthesis of design concepts from a design for assembly perspective. *Computer Integrated Manufacturing Systems, 11*(1), 1-13. doi:https://doi.org/10.1016/S0951-5240(98)80003-7

HTCDelft. (2017). *Jaaroverzicht 2017*. Retrieved from https://www.htcdelft.nl/wp-content/uploads/2018/06/htcdelft_jaarverslag_2017-def6-kleine-let-_rgb_1805.pdf

Humanoid. (n.d.). In *Cambridge English Dictionary*. Retrieved from https://dictionary.cambridge.org/dictionary/english/humanoid

IFR. (2018). Executive Summary World Robotics 2018 Industrial Robots. Retrieved from https://ifr.org/downloads/press2018/Executive_Summary_WR_2018_Industrial_Robots.pdf

International Organization for Standardization. (1998). *Manipulating industrial robots - Performance criteria and related test methods* (ISO/DIS Standard No. 9283). Retrieved from https://www.iso.org/standard/22244.html

Iron Ox. (2018, November 29). Iron Ox - Premium produce for everyone. Retrieved from http://ironox.com

James, C., & Wright, F. L. (1968). *The Imperial Hotel*. Rutland, Vermont, United States: Tuttle.

Jani, K. (2016). Standard safety practices for Industrial Robots.

Japan Entertainment. (2017, May 9). Pepper Robot - Future of elderly people healthcare in Japan. Retrieved from https://www.youtube.com/watch?v=5ei0xrC_L6A

Johnston, L. (2018, May 13). 'What about that human touch?' Elderly will be cared for by ROBOTS to solve staff shortage. *Express.co.uk.* Retrieved from https://www.express.co.uk/news/uk/958844/uk-care-homes-robots-elderly

Jongeneel, C. (2018, October). How to build a million new homes? *Delft Outlook, 35,* 22-25.

Kassow Robots. (n.d.). KR810. Retrieved from https://www.kassowrobots.com/products/kr810/

Kawasaki. (2015, May 13). Standard specifications ZX300SFE02001. Retrieved from https://robotics.kawasaki.com/userAssets1/productPDF/ZX300SFE02001-E.pdf

Kawasaki. (2018a, May 10). Standard specifications RS015XFE02. Retrieved from https://robotics.kawasaki.com/userAssets1/productPDF/RS015XFE02-E.pdf

Kawasaki. (2018b, May 10). Standard specifications RS050NFE02. Retrieved from https://robotics.kawasaki.com/userAssets1/productPDF/RS050NFE02-E.pdf

Kawasaki. (2018c, May 10). Standard specificsations RS080NFE02. Retrieved from https://robotics.kawasaki.com/userAssets1/productPDF/RS080NFE02-E.pdf

Kawasaki. (2019a). RS015X Robot. Retrieved from https://robotics.kawasaki.com/en1/products/robots/small-medium-payloads/RS015X

Kawasaki. (2019b, January 18). Standard specifications MG15HL*E58. Retrieved from https://robotics.kawasaki.com/userAssets1/productPDF/MG15HLFE58-E.pdf

Kawasaki. (n.d.-a). Kawasaki Robot MG15HLFE28. Retrieved from http://kawasakirobotics.nl/index.php?id_product=46&controller=product&id_lang=1

Kawasaki. (n.d.-b). Kawasaki Robot RS080NFE02. Retrieved from http://kawasakirobotics.nl/index.php?id_product=27&controller=product&id_lang=1

Kawasaki. (n.d.-c). Kawasaki Robot ZX300SFE02. Retrieved from http://kawasakirobotics.nl/index.php?id_product=31&controller=product&id_lang=1

Kim, H., Miller, L. M., Al-Refai, A., Brand, M., & Rosen, J. (2011, 30 Aug.-3 Sept. 2011). *Redundancy resolution of a human arm for controlling a seven DOF wearable robotic system.* Lecture at the 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society.

Knepper, R. A., Layton, T., Romanishin, J., & Rus, D. (2013). *Ikeabot: An autonomous multi-robot coordinated furniture assembly system.* Lecture at the Robotics and Automation (ICRA), 2013 IEEE International Conference on Robotics and Automation (ICRA).

Kohs, G. (Writer). (2017). AlphaGo. In. Netflix: https://www.netflix.com/title/80190844.

Koot. (n.d.). Koot Automation - Fanuc M2000. Retrieved from http://www.kootautomation.nl/koot/producten/fanuc-robotics/fanuc-m2000.html

Koubaa, A. (2016). *Robot Operating System (ROS)* (Vol. 1): Springer, Cham.

KUKA. (2016a, May 24). KR 1000 1300 titan PA, KR 1000 L950 titan PA. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/48ec812b1b2947898ac2598aff70abc0/spez_kr_1000_titan_pa_en.pdf

KUKA. (2016b, May 23). LBR iiwa. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/48ec812b1b2947898ac2598aff70abc0/spez_lbr_iiwa_en.pdf

KUKA. (2018, May 17). KR QUANTEC extra HA. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/48ec812b1b2947898ac2598aff70abc0/spez_kr_quantec_extra_ha_en.pdf

KUKA. (2019a, March 4). KR 120 R2500 pro. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000182735_en.pdf

KUKA. (2019b, March 4). KR 120 R2900 extra HA. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000182741_en.pdf

KUKA. (2019c, March 4). KR 150 R2700 extra. Retrieved from https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000182737_en.pdf

KUKA. (n.d.). KUKA KR 150. Retrieved from https://www.robots.com/images/robots/KUKA/High-Payload/KUKA_KR_150_Datasheet.pdf

Latmas. (n.d.). UR3 robot. Retrieved from http://www.latmas.lv/en/robotics/robot-hands/ur3-robot

Laundroid. (n.d.). Laundroid | A new way to liberate your loved one in automation age. Retrieved from https://laundroid.sevendreamers.com/en

Mabi Robotic. (n.d.-a). Datenblatt: Speedy 12. Retrieved from https://www.mabi-robotic.com/robotic/createpdf/pdfCreateProdukt.asp?IDProdukt=2

Mabi Robotic. (n.d.-b). Produkte und Lösungen. Retrieved from https://www.mabi-robotic.com/robotic/overview

Matsuoka, Y., Durrant-Whyte, H. F., & Neira, J. (2011). *Robotics: Science and Systems VI*: MIT Press.

Matthias, B. (2014). *Industrial safety requirements for collaborative robots and applications.* Lecture at the ERF 2014 Workshop: Workspace Safety in Industrial Robotics: trends, integration, and standards.

Melchiorri, C. (2014). Dynamic model of robot manipulators. Retrieved from http://www-lar.deis.unibo.it/people/cmelchiorri/Files_Robotica/FIR_05_Dynamics.pdf

Mertens, S. (2019). Stiny House - Volg mijn zoektocht naar geluk. Retrieved from https://stinyhouse.nl

Minitopia. (2018). Minitopia Poeldonk. Retrieved from http://www.minitopia.eu/

Moley. (n.d.). Moley - The world's first robotic kitchen. Retrieved from http://www.moley.com

Monkman, G. J., Hesse, S., Steinmann, R., & Schunk, H. (2007). *Robot grippers*. Weinheim: Wiley-VCH.

Motion Controls Robotics. (2017a). Range of Robot Cost - Robot System Cost Series. Retrieved from https://motioncontrolsrobotics.com/range-robot-cost/

Motion Controls Robotics. (2017b). Should You Purchase a Used Robot? Retrieved from https://motioncontrolsrobotics.com/should-you-purchase-a-used-robot/

Nachi. (2016). Robots for Handling Heavy Loads. Retrieved from http://www.nachi-fujikoshi.co.jp/eng/rob/juryo/mc350a.htm

Nachi. (n.d.). Standard specification MC470P-01-FD11. Retrieved from http://www.nachirobotics.com/wp-content/uploads/2018/07/NRS-MC-470P-StandardSpecification.pdf

Nubiola, A. (2015). The future of robot off-line programming. In.

Odico. (n.d.). Robotic Hot-wire cutting. Retrieved from https://odico.dk/en/technologies#robotic-hot-wire-cutting

On Robot. (2019). Gecko Gripper. Retrieved from https://onrobot.com/en/products/gecko-gripper

Oosterhuis, K., Bier, H., Aalbers, C., & Boer, S. (2004). *File to Factory and Real Time Behavior in ONL-Architecture.* Lecture at the Fabrication: Examining the Digital Practice of Architecture, Proceedings of the 23rd Annual Conference of the Association for Computer Aided Design in Architecture and the 2004 Conference of the AIA Technology in Architectural Practice Knowledge Community, Cambridge, November 8-14, 2004.

OpenCV. (2019). Open source Computer Vision - Home. Retrieved from https://opencv.org

Panzarino, M. (2018, June 28). Disney Imagineering has created autonomous robot stunt doubles. *TechCrunch.* Retrieved from https://techcrunch.com/2018/06/28/disney-imagineering-has-created-autonomous-robot-stunt-doubles

Pepper Robot Hire. (2019). Pepper Robot Hire & Rental UK. Retrieved from http://pepperrobothire.com

Pérez, L., Rodríguez, Í., Rodríguez, N., Usamentiaga, R., & García, D. (2016). Robot guidance using machine vision techniques in industrial environments: A comparative review. *Sensors, 16*(3), 335.

Pneumatic Products. (2017, February 27). Joulin Mini Grippers. Retrieved from https://pneumatics.com.au/joulin-mini-grippers

Randek. (2019). ZeroLabor Robotic System - House Production Technologies. Retrieved from http://www.randek.com/en/wall-floor-and-roof-production-lines/zerolabor

Ravenscroft, T. (2018, June 4). Olafur Eliasson's first building is a castle-like office in a Danish fjord. Retrieved from https://www.dezeen.com/2018/06/04/olafur-eliasson-fjordenhus-office-vejle-fjord-denmark-architecture/

Red, E. (1999, September 2). Robotics Overview. Retrieved from https://www.et.byu.edu/~ered/eaal/html/body_robotics_review.html

Redshift Video. (2018, September 6). Can Machine Learning and Robotics Transform Factories? Retrieved from https://www.autodesk.com/redshift/machine-learning-robotics

Retsin, G. (2017). Tallinn Architecture Biennale Pavilion. Retrieved from https://www.retsin.org/Tallinn-Architecture-Biennale-Pavilion

Retsin, G. (2019, February 28). *Modular systems.* Lecture at the ROBOUWTICS symposium, Faculty of Architecture and the Built Environment, Delft University of Technology, the Netherlands.

RoboDK. (2018, July 18). RoboDK plugin for Rhino and Grasshopper. Retrieved from https://www.food4rhino.com/app/robodk

RoboDK. (2019a, June 3). 2. robolink Module — RoboDK API Documentation. Retrieved from https://robodk.com/doc/en/PythonAPI/robolink.html?highlight=_Test

RoboDK. (2019b). RoboDK Home. Retrieved from https://robodk.com

RoboDK. (2019c). RoboDK Pricing. Retrieved from https://robodk.com/pricing

RoboDK. (2019d, April 7). Robot Library. Retrieved from https://robodk.com/library

Robot. (n.d.). In *Cambridge English Dictionary*. Retrieved from https://dictionary.cambridge.org/dictionary/english/robot

Robot Worx. (2019a). FANUC M-2000iA/2300. Retrieved from https://www.robots.com/robots/fanuc-m-2000ia-2300

Robot Worx. (2019b). How Much Do Industrial Robots Cost? Retrieved from https://www.robots.com/faq/how-much-do-industrial-robots-cost

Robot Worx. (2019c). What Are My Robot Options Based On Maximum Payload? Retrieved from https://www.robots.com/faq/what-are-my-robot-options-based-on-maximum-payload

Robotics Online. (2019, January 3). Is a Vacuum Gripper Right for Your Collaborative Robot Application? Retrieved from https://www.robotics.org/blog-article.cfm/Is-a-Vacuum-Gripper-Right-for-Your-Collaborative-Robot-Application/134

Robotiq. (2017). *Cobots ebook: Collaborative Robots Buyer's Guide*. In. Retrieved from https://blog.robotiq.com/hubfs/COBOT%20EBOOK%20FINAL.pdf

Robotiq. (2019a). Wrist Camera - Vision made for Universal Robot. Retrieved from https://blog.robotiq.com/hubfs/Product-sheets/Product_sheet-Wrist-Camera-EN.pdf

Robotiq. (2019b). Wrist Camera Vision System for e-Series Universal Robots - Instruction Manual. Retrieved from https://assets.robotiq.com/website-assets/support_documents/document/Vision_System_e-Series_PDF_20190329.pdf

Ronquillo, R. (n.d.). Understanding CNC Milling. Retrieved from https://www.thomasnet.com/articles/custom-manufacturing-fabricating/understanding-cnc-milling

ROS. (2019a, April 27). ROS answers. Retrieved from https://answers.ros.org/

ROS. (2019b). ROS.org | Testimonials. Retrieved from http://www.ros.org/testimonials

Roser, M. (2018, December 18). Employment in Agriculture. *Our World in Data.* Retrieved from https://ourworldindata.org/employment-in-agriculture

Rouse, M. (2009a, July). degrees of freedom. Retrieved from https://whatis.techtarget.com/definition/degrees-of-freedom

Rouse, M. (2009b, February). What is end effector? - Definition from WhatIs.com. Retrieved from https://whatis.techtarget.com/definition/end-effector

Ryan, C. (2016, May 4). *How To Create The Right Collaborative System For Your Application.* Lecture at the National Robotics Safety Conference, Hynes Convention Center, Boston, Massachusetts USA.

Schmalz. (2019). Special Grippers | Needle Grippers SNG-D / DV. Retrieved from https://www.schmalz.com/en/vacuum-technology-for-automation/vacuum-components/special-grippers/needle-grippers/needle-grippers-sng-d-dv

Seven Dreamers. (2019, April 23). [Important Notice] Notice of petition and decision to start bankruptcy proceedings. Retrieved from https://translate.google.nl/translate?hl=&sl=ja&tl=en&u=https%3A%2F%2Fsevendreamers.com%2Fnews%2F190423_corporate%2F

Siciliano, B., & Khatib, O. (2016). *Springer handbook of robotics*: Springer.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., . . . Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science, 362*(6419), 1140-1144. doi:10.1126/science.aar6404

Soft Robotics. (2019). SuperPick. Retrieved from https://www.softroboticsinc.com/superpick

Stäubli. (2010, September). Stäubli Robotics Controls software bring powerful and flexible solutions. Retrieved from http://eola.cz/images/staubli/download/Prehled_produktu_Stabuli_Robotics.pdf

Stäubli. (2018). TX2-90 Range. Retrieved from https://www.staubli.com/en/file/5260.show

Stäubli. (2019). Robotic arms. Retrieved from https://www.staubli.com/en-in/robotics/product-range/6-axis-scara-picker-industrial-robots/tx2-collaborative-robots

Stumm, S., Devadass, P., & Brell-Cokcan, S. (2018). Haptic programming in construction. *Construction Robotics, 2*(1), 3-13. doi:10.1007/s41693-018-0015-9

T&D World. (2017, April 29). Robotic Jackhammer. Retrieved from https://www.tdworld.com/td-how/robotic-jackhammer

TAL Manufacturing Solutions. (2017, August 21). Robot: Made in India! Retrieved from http://www.machinemarketindia.com/tooling-and-workholding/robot-made-in-india

Tata Motors. (n.d.). TAL BRABO robot with five axis, 750 reach,speed of 1000mm/ second,operate on single phase,10 kg Payload. Retrieved from https://sierracontrol.com/product-detail/TAL-BRABO-robot-with-five-axis,-750-reach,speed-of-1000mm-second,operate-on-single-phase,10-kg-Payload/22185

Terdiman, D. (2018, July 22). Autodesk's LEGO model-building robot is the future of manufacturing. Retrieved from https://www.fastcompany.com/90204615/autodesks-lego-model-building-robot-is-the-future-of-manufacturing

Thayer, K. (2017, April 27). What Is the Real Cost of an Industrial Robot Arm? Retrieved from https://insights.globalspec.com/article/4788/what-is-the-real-cost-of-an-industrial-robot-arm

The Statistics Portal. (2019). Euro (EUR) to U.S. dollar (USD) annual average exchange rate from 1999 to 2018. Retrieved from https://www.statista.com/statistics/412794/euro-to-u-s-dollar-annual-average-exchange-rate/

TheNewMakers. (n.d.). HOME - TheNewMakers. Retrieved from https://www.thenewmakers.com/

TIOBE. (2019, April 26). TIOBE Index | TIOBE - The Software Quality Company. Retrieved from https://www.tiobe.com/tiobe-index

Traill, P. (2019, January 24). Number of Possible Go Games. Retrieved from https://senseis.xmp.net/?NumberOfPossibleGoGames

Troitino, C. (2018, September 24). Meet The World's First Fully Automated Burger Robot: Creator Debuts The Big Mac Killer. *Forbes.* Retrieved from https://www.forbes.com/sites/christinatroitino/2018/06/21/meet-the-worlds-first-fully-automated-burger-robot-creator-debuts-the-big-mac-killer/#3a4645416a89

Twin, A. (2019, April 10). Total Cost of Ownership (TCO): What an Asset Will Cost You Over the Long Term. Retrieved from https://www.investopedia.com/terms/t/totalcostofownership.asp

UCL. (2017, December 13). Building-block pavilion wins Tallinn Architectural Biennale competition. *Bartlett School of Architecture.* Retrieved from https://www.ucl.ac.uk/bartlett/architecture/news/2017/dec/building-block-pavilion-wins-tallinn-architectural-biennale-competition

Universal Robots. (2017). UR5 manual. Retrieved from http://robotics.ee.uwa.edu.au/courses/robotics/project/ur5/UR5_User_Manual_en_Global-3.4.5.pdf

Universal Robots. (n.d.-a). UR5. Retrieved from https://www.universal-robots.com/media/1801303/eng_199901_ur5_tech_spec_web_a4.pdf

Universal Robots. (n.d.-b). UR10. Retrieved from https://www.universal-robots.com/media/1801323/eng_199901_ur10_tech_spec_web_a4.pdf

USA Department Of Labor. (2013, May 9). Machine Guarding eTool | Additional Safety Considerations. Retrieved from https://www.osha.gov/SLTC/etools/machineguarding/additional_considerations.html

Vacuum gripper. (n.d.). In *The free dictionary*. Retrieved from https://encyclopedia2.thefreedictionary.com/vacuum+gripper

Veo Robotics. (2019). Bringing perception and intelligence to industrial robots. Retrieved from https://www.veobot.com/

Wassink, J. (2018, October). Human control of artificial intelligence. *Delft Outlook, 35,* 7-9.

Winick, E. (2018). Every study we could find on what automation will do to jobs, in one chart. *MIT Technology Review*.

Yaskawa. (2016, June 28). Motoman HC10 collaborative robot - Safe and flexible interaction. Retrieved from https://www.yaskawa.eu.com/en/news-events/news/article/news/motoman-hc10-collaborative-robot-safe-and-flexible-interaction/?tx_news_pi1%5Bcontroller%5D=News&tx_news_pi1%5Baction%5D=detail&cHash=e91562b0cf2f02dac3884111cfacf6dc

Yaskawa. (2017a, April). MOTOMAN MH600. Retrieved from https://www.yaskawa.eu.com/en/products/robotics/motoman-robots/productdetail/product/mh600/

Yaskawa. (2017b, November). UP400RDII. Retrieved from https://www.motoman.com/hubfs/Robots/UP400RDII.pdf

Yaskawa. (2018a, November). HC10: Human-Collaborative Robot. Retrieved from https://www.motoman.com/hubfs/Collaborative/HC10.pdf?hsLang=en-us

Yaskawa. (2018b, January). MPL800 II. Retrieved from https://cdn2.hubspot.net/hubfs/366775/Robots/MPL800II.pdf

Yaskawa. (2018c). UP400RDII: High-Payload Industrial Robot. Retrieved from https://www.motoman.com/industrial-robots/up400rdii

Yaskawa. (n.d.). Robotics Glossary. Retrieved from https://www.motoman.com/en-us/about/company/robotics-glossary

Yelp. (2019, 3 January). Creator - SoMa - San Francisco, CA. Retrieved from https://www.yelp.com/biz/creator-san-francisco

Zhou, K., Taigang, L., & Lifeng, Z. (2015, 15-17 Aug. 2015). *Industry 4.0: Towards future industrial opportunities and challenges.* Lecture at the 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD).

APPENDICES

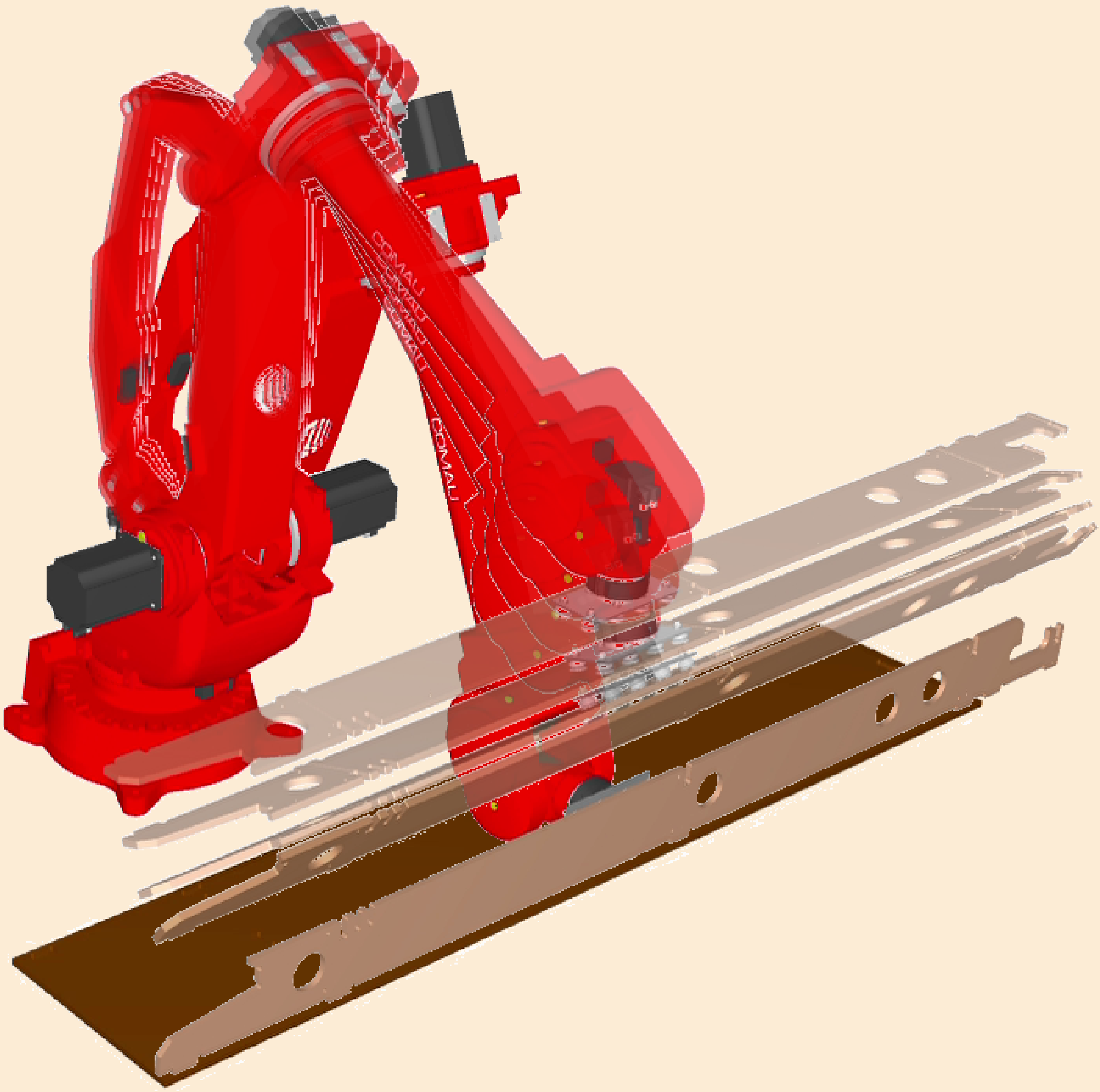
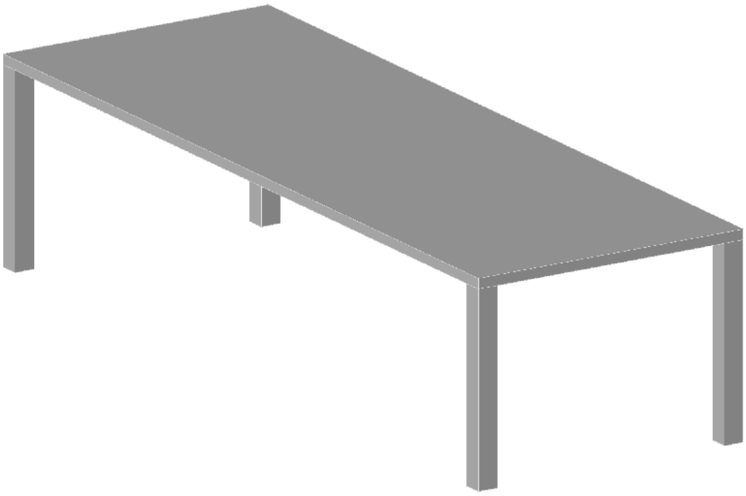
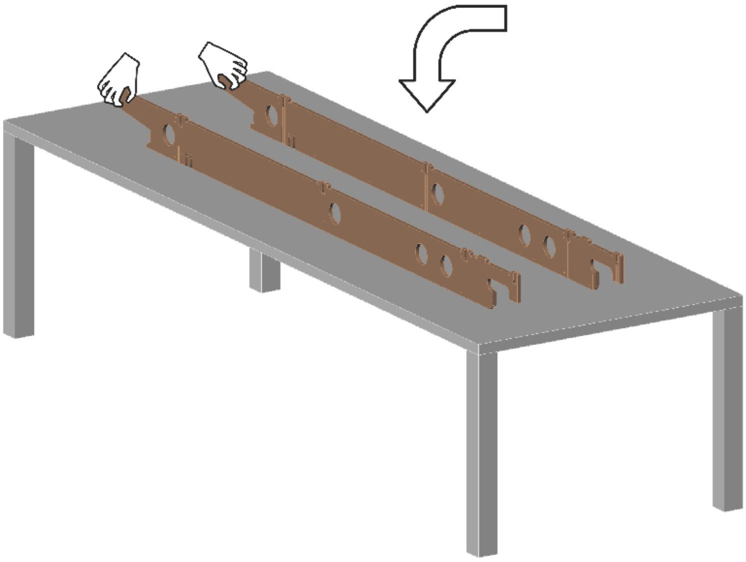


Figure 175: Chronophotographic image showing the movement of the end effector sliding the right vertical beam in its final location.

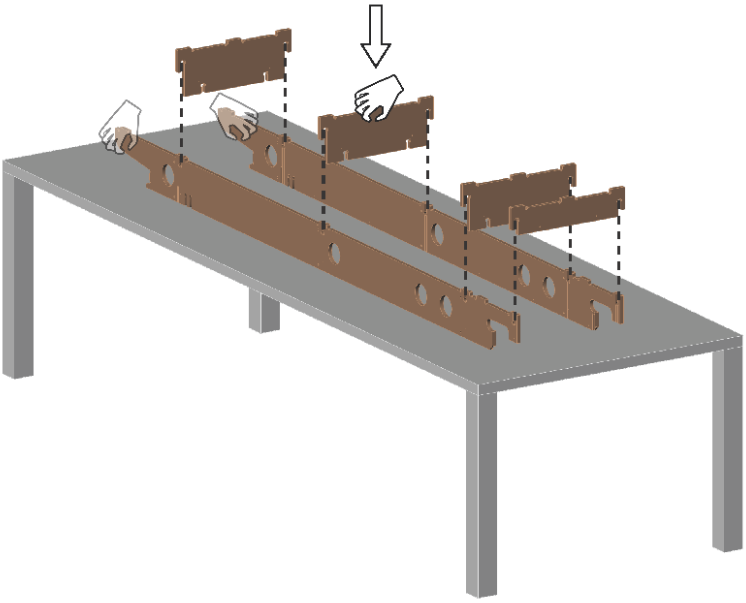
Appendix A: Manual assembly steps of geometry version A - Original



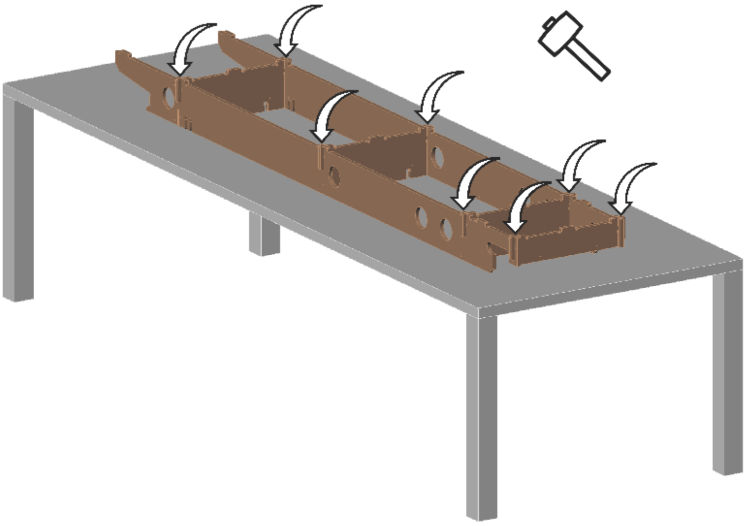
Geometry A - Step 1: Start with an empty assembly table.



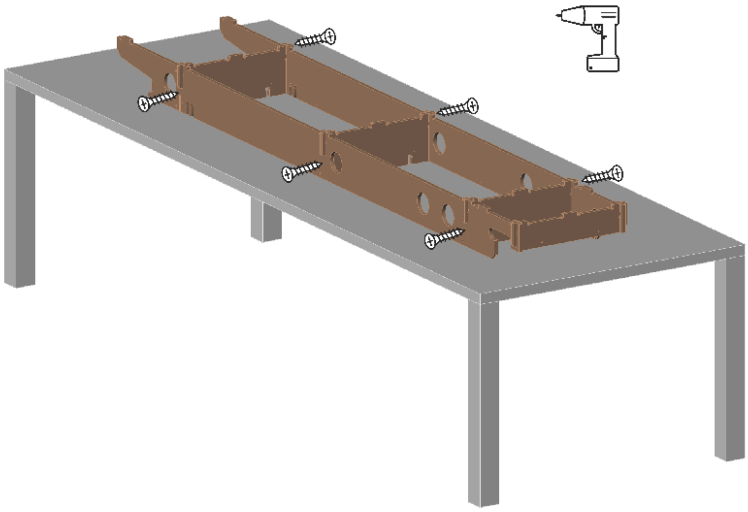
Geometry A - Step 2: Place the two vertical beams on the assembly table and have one person holding them upright.



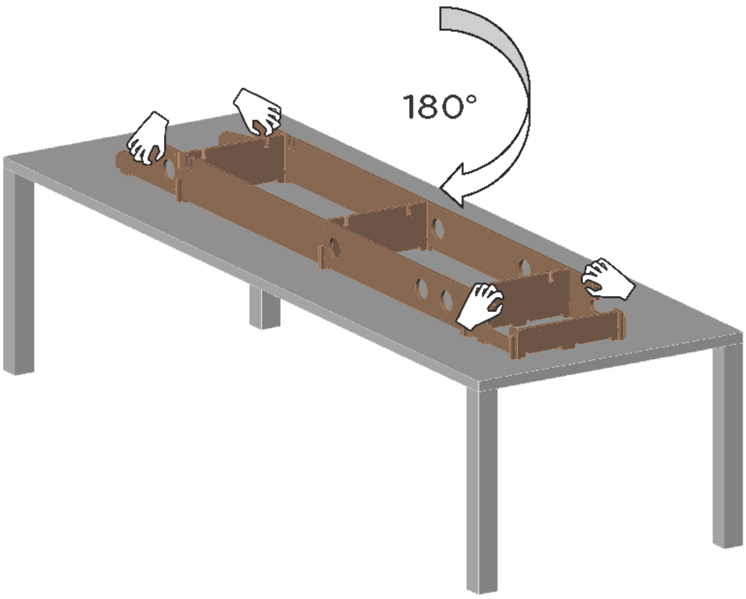
Geometry A - Step 3: Slide the three horizontal beams and the horizontal bottom beam in the grooves of the two vertical beams.



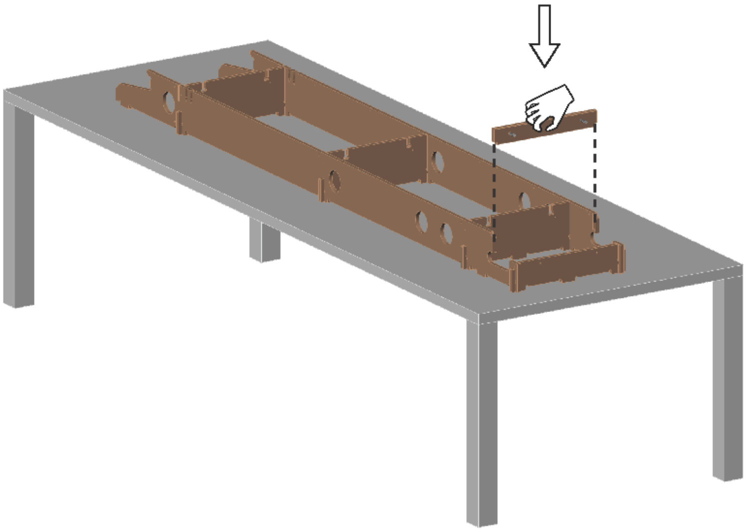
Geometry A - Step 4: Hammer all four horizontal beams in place using a rubber mallet.



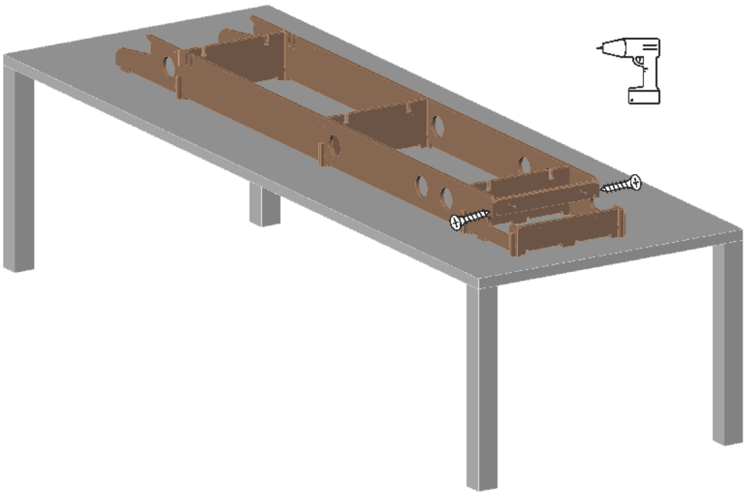
Geometry A - Step 5: Fasten the horizontal and vertical beams together with countersunk cross head wood screws using a cordless screwdriver.



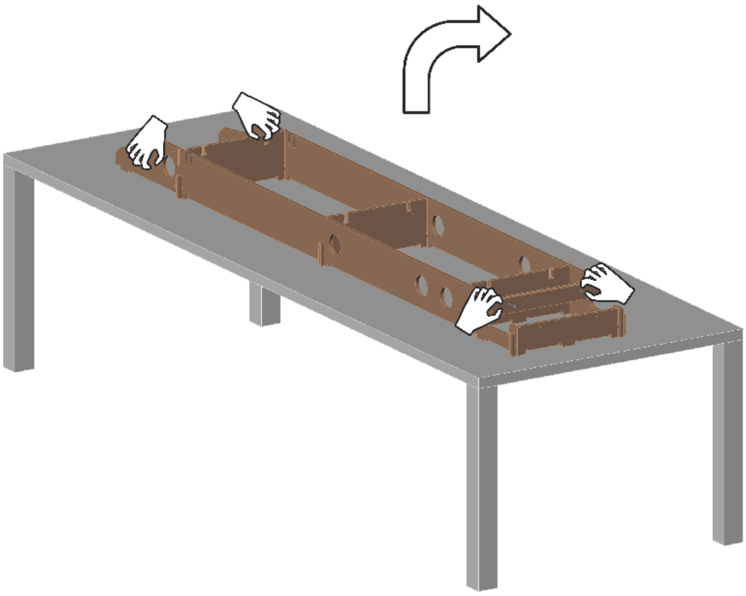
Geometry A - Step 6: Two people lift up the wooden frame and rotate it 180 degrees about its longest axis.



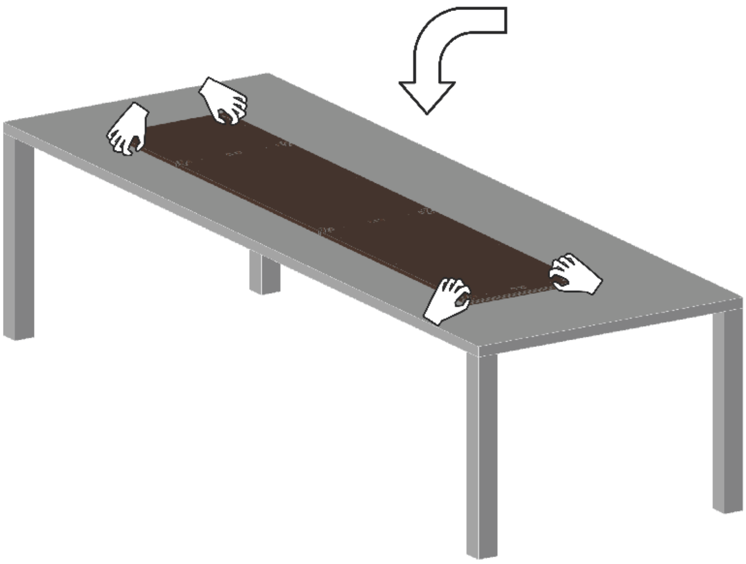
Geometry A - Step 7: Slide the horizontal floor beam in the grooves of the two vertical beams.



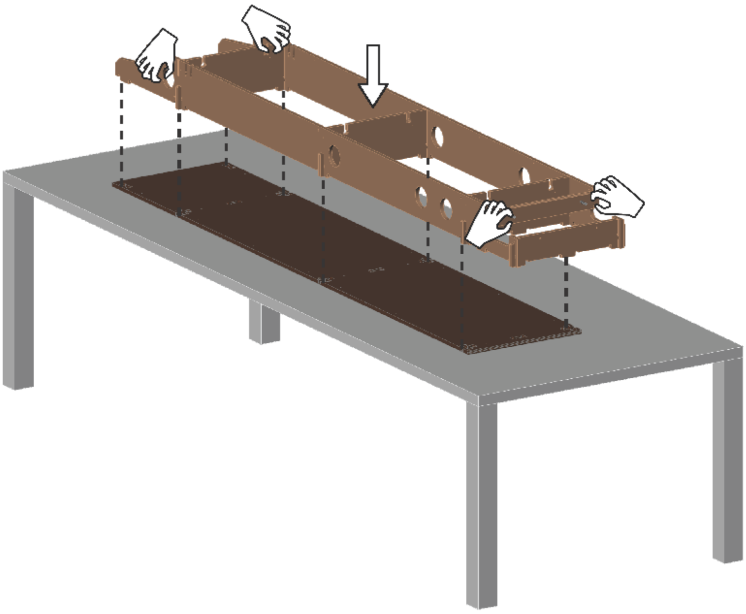
Geometry A - Step 8: Fasten the horizontal floor beam to the vertical beams with countersunk cross head wood screws using a cordless screwdriver.



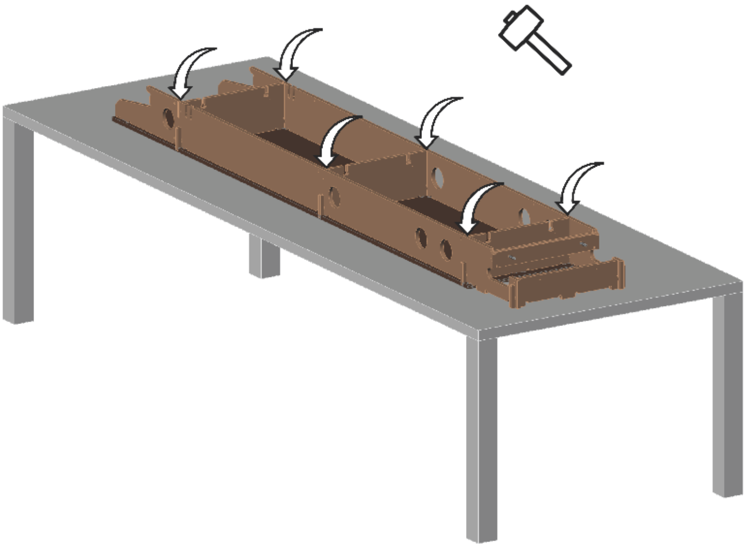
Geometry A - Step 9: Two people lift the wooden frame of the assembly table.



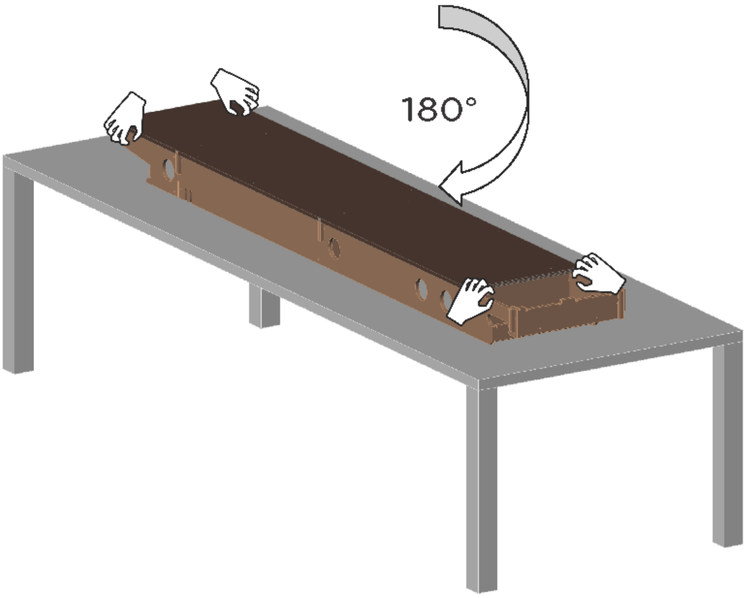
Geometry A - Step 10: Two people place the outer layer on the assembly table with the outside on the table so the inside, which contain dog bone slots, faces upwards. Clear residue from the dog bone slots.



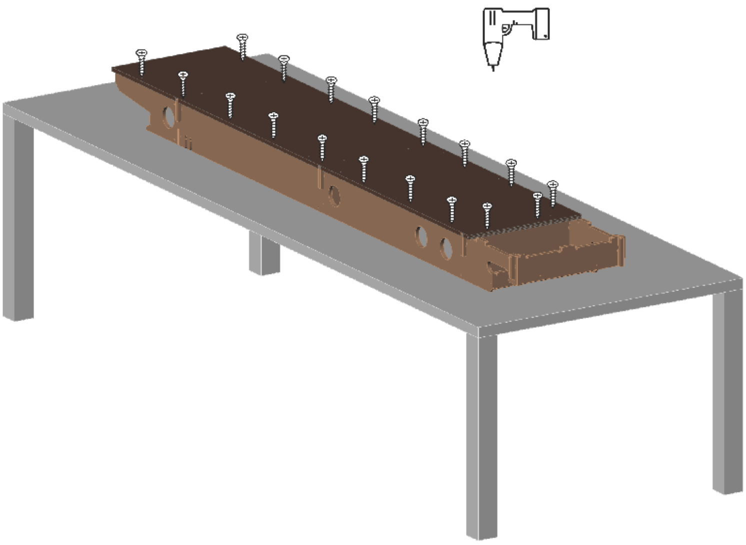
Geometry A - Step 11: Two people lift up the wooden frame and position it in the dog bone slots of the outer layer.



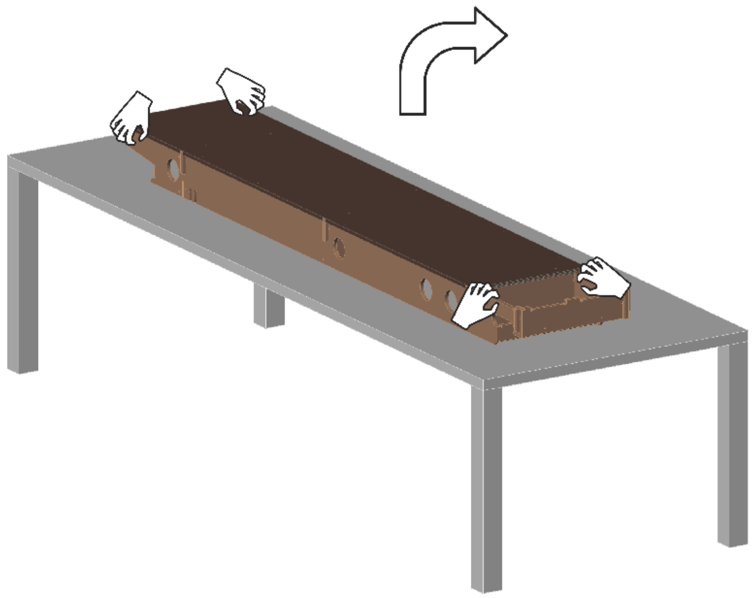
Geometry A - Step 12: Hammer the wooden frame in the dog bone slots of the outer layer using a rubber mallet.



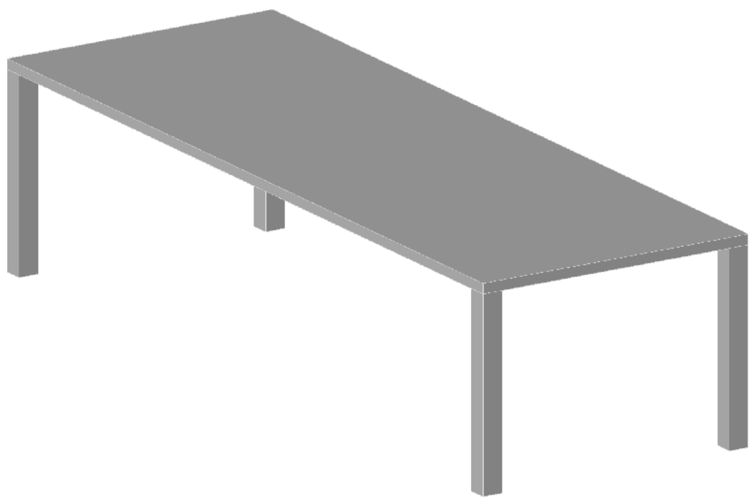
Geometry A - Step 13: Two people lift up the whole component and rotate it 180 degrees about its longest axis.



Geometry A - Step 14: Fasten the outer layer against the horizontal and vertical beams with round cross head wood screws using a cordless screwdriver. All screws have a rubber washer so the water protective capability of the outer layer remains intact.

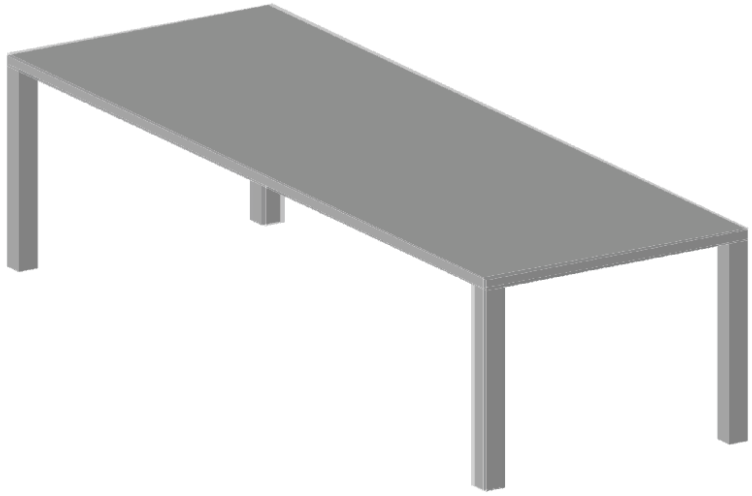


Geometry A - Step 15: Two people lift the completed wall component of the assembly table.

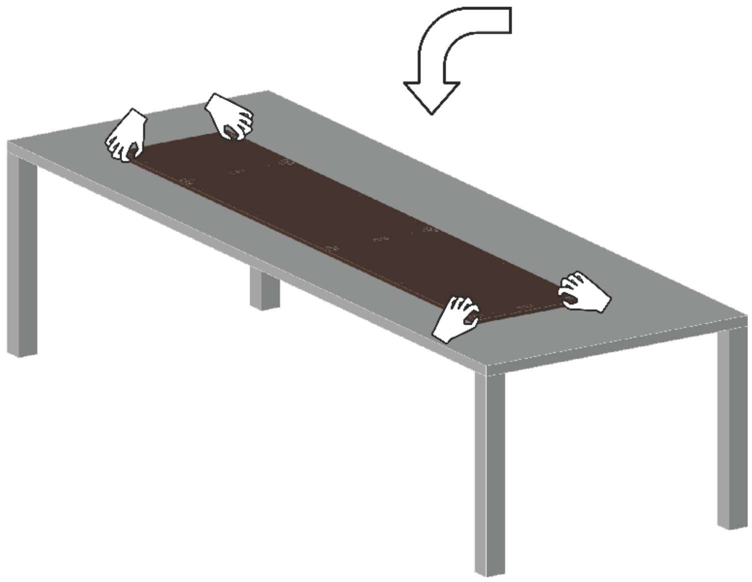


Geometry A - Step 1: Start with an empty assembly table.

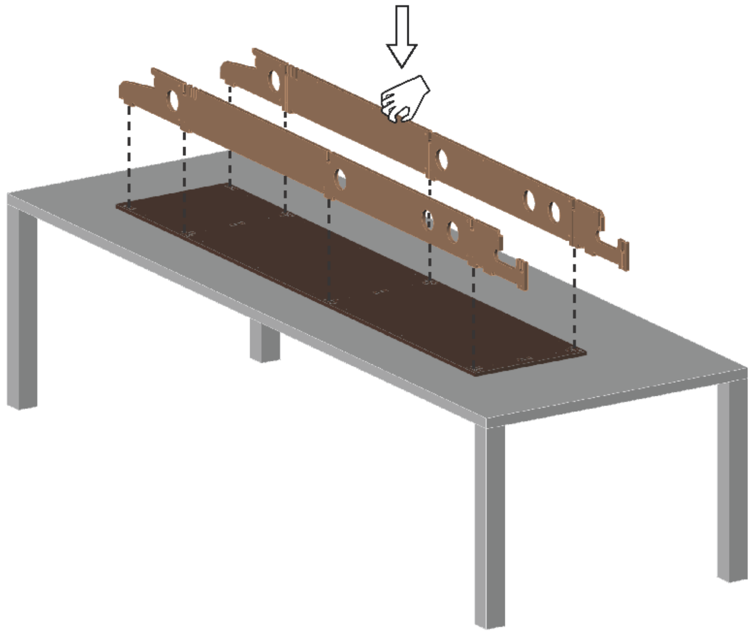
Appendix B: Manual assembly steps of geometry version B - DFA2



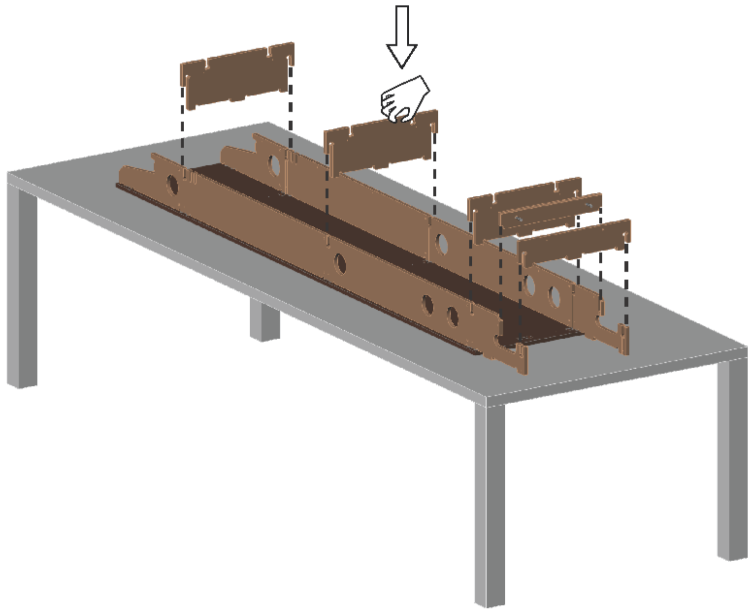
Geometry B - Step 1: Start with an empty assembly table.



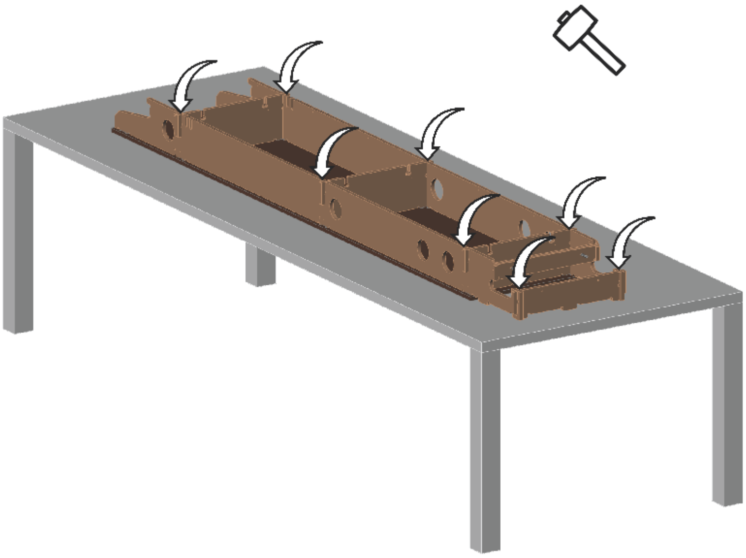
Geometry B - Step 2: Two people place the outer layer on the assembly table with the outside on the table so the inside, which contain dog bone slots, faces upwards. Clear residue from the dog bone slots.



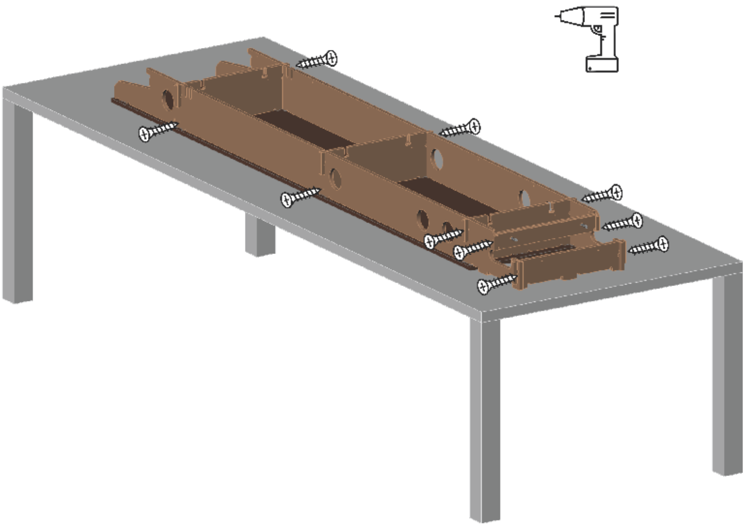
Geometry B - Step 3: Position the two vertical beams in the dog bone slots of the outer layer.



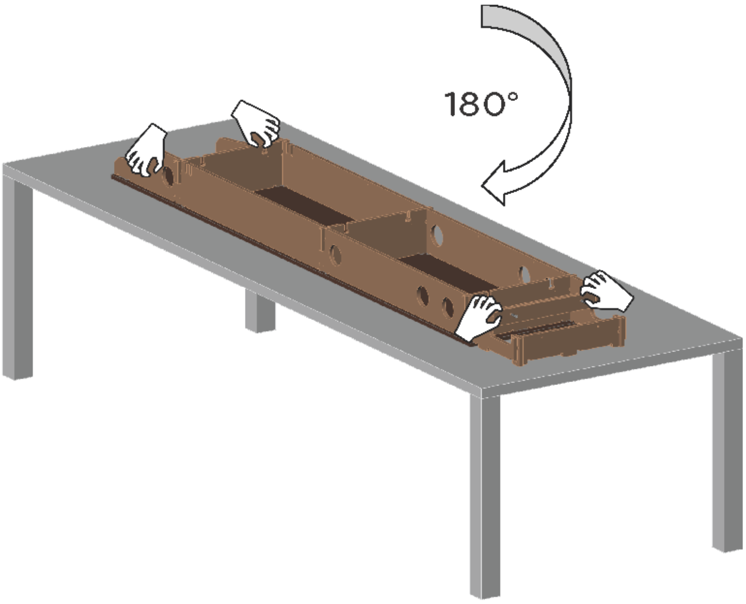
Geometry B - Step 4: Slide the three horizontal beams, the horizontal bottom beam and the floor beam in the grooves of the two vertical beams.



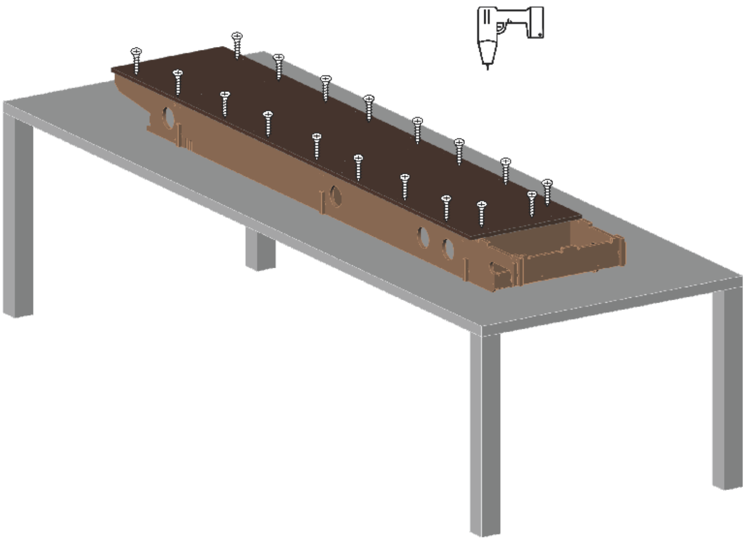
Geometry B - Step 5: Hammer all four horizontal beams in place using a rubber mallet. It will simultaneously ensure the wooden frame be hammered in the dog bone slots of the outer layer.



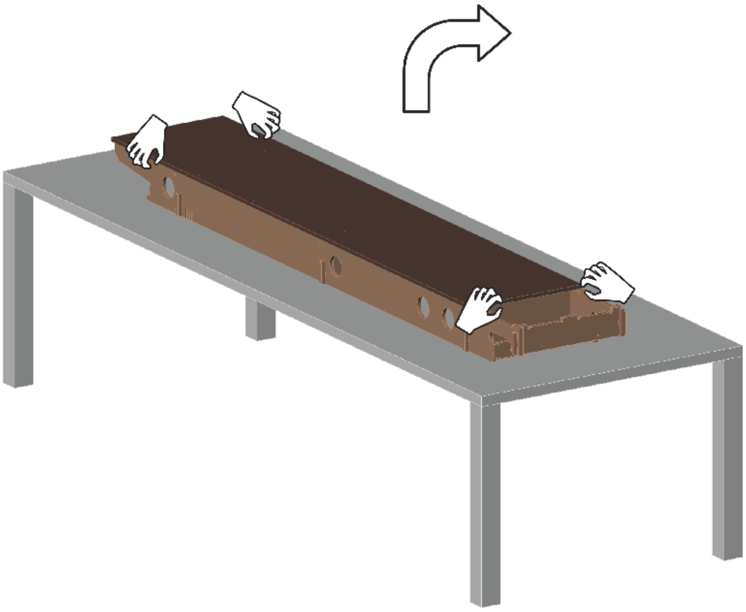
Geometry B - Step 6: Fasten all beams together with countersunk cross head wood screws using a cordless screwdriver.



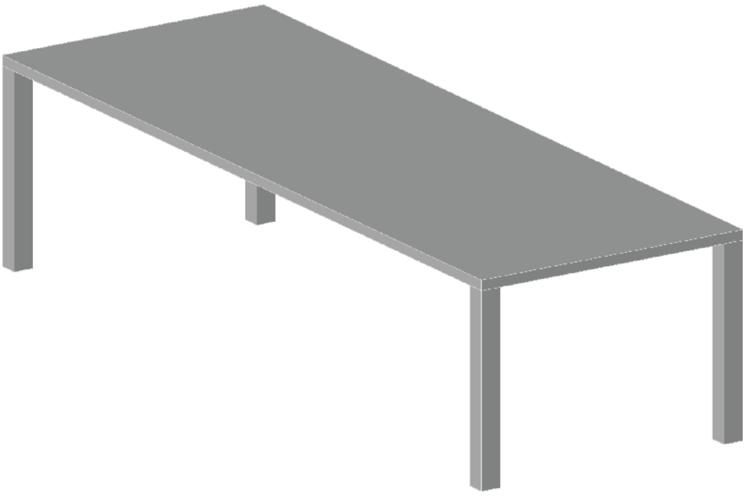
Geometry B - Step 7: Two people lift up the whole component and rotate it 180 degrees about its longest axis.



Geometry B - Step 8: Fasten the outer layer against the horizontal and vertical beams with round cross head wood screws using a cordless screwdriver. All screws have a rubber washer so the water protective capability of the outer layer remains intact.

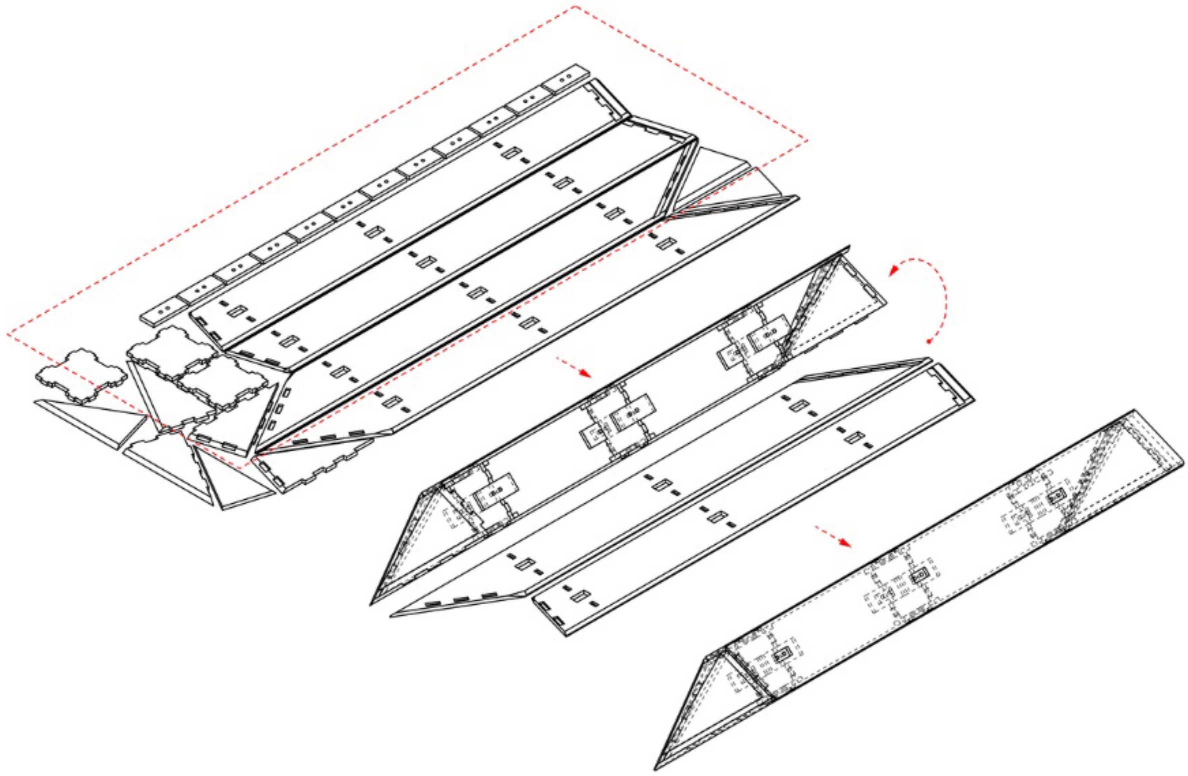


Geometry B - Step 9: Two people lift the completed wall component of the assembly table.

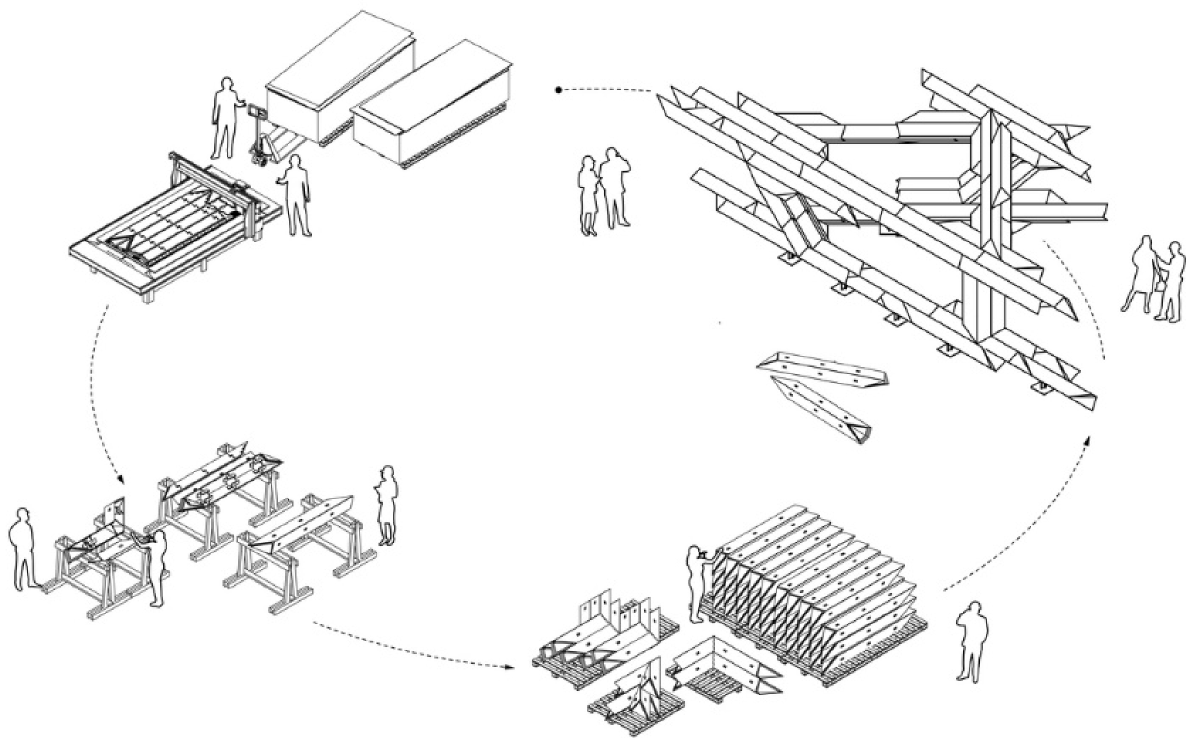


Geometry B - Step 1: Start with an empty assembly table.

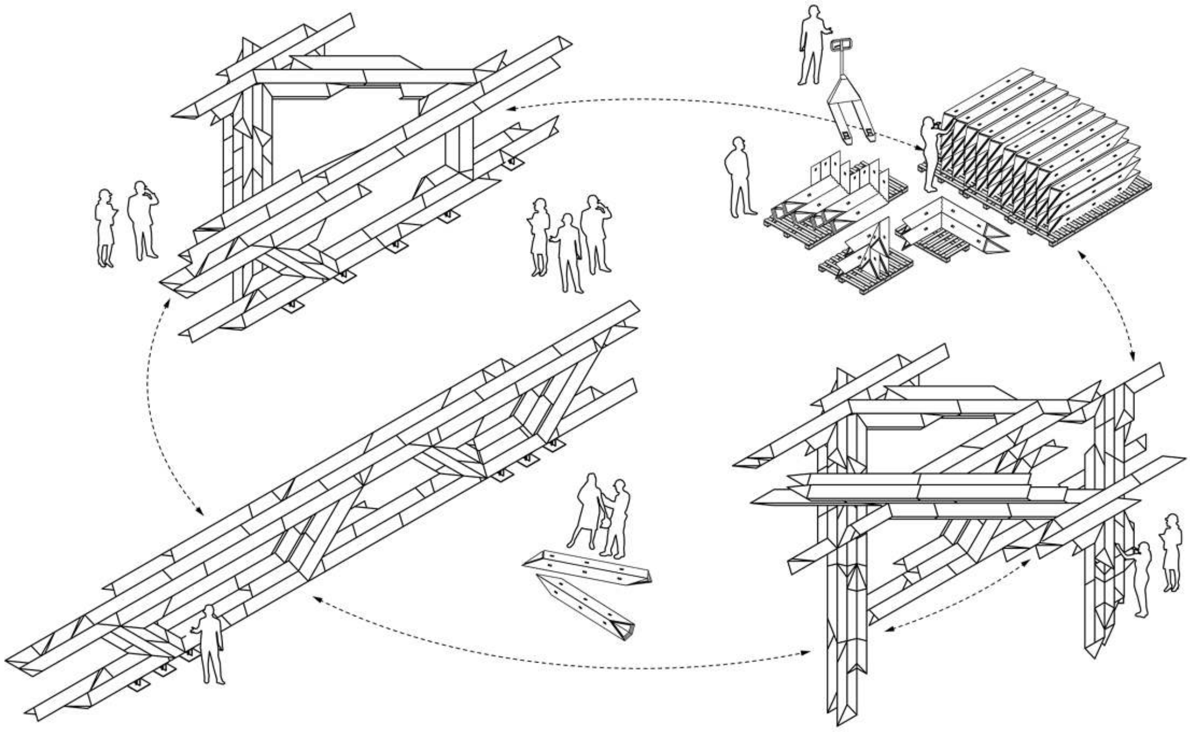
Appendix C: Building method Gilles Retsin



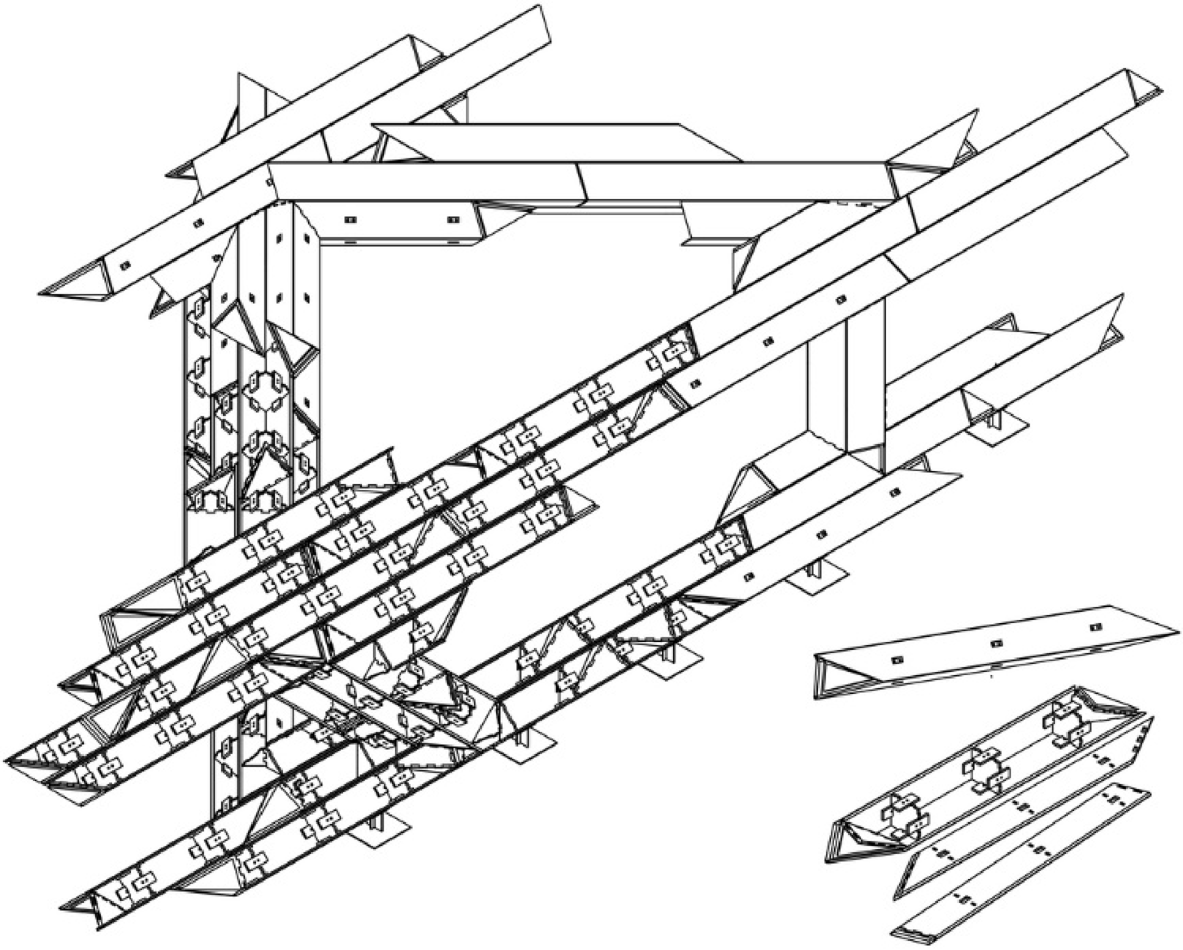
The elements are cut out of a sheet and assembled into a component.



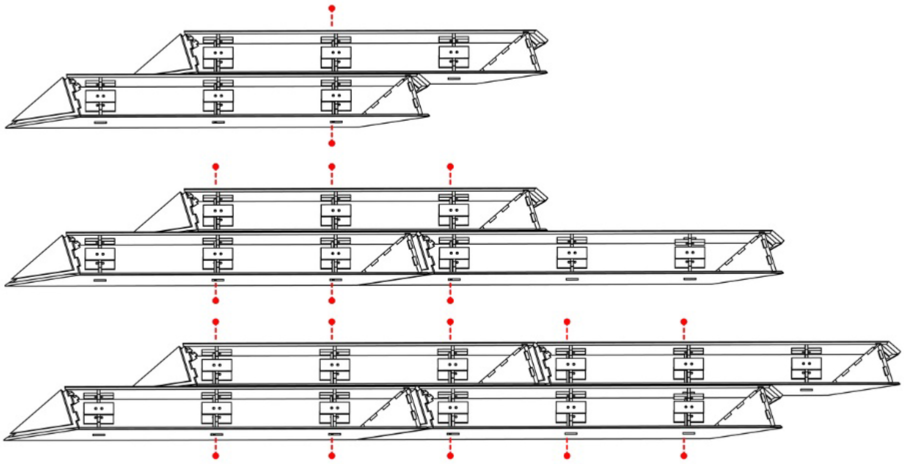
From flat wooden sheets to the constructed pavilion.



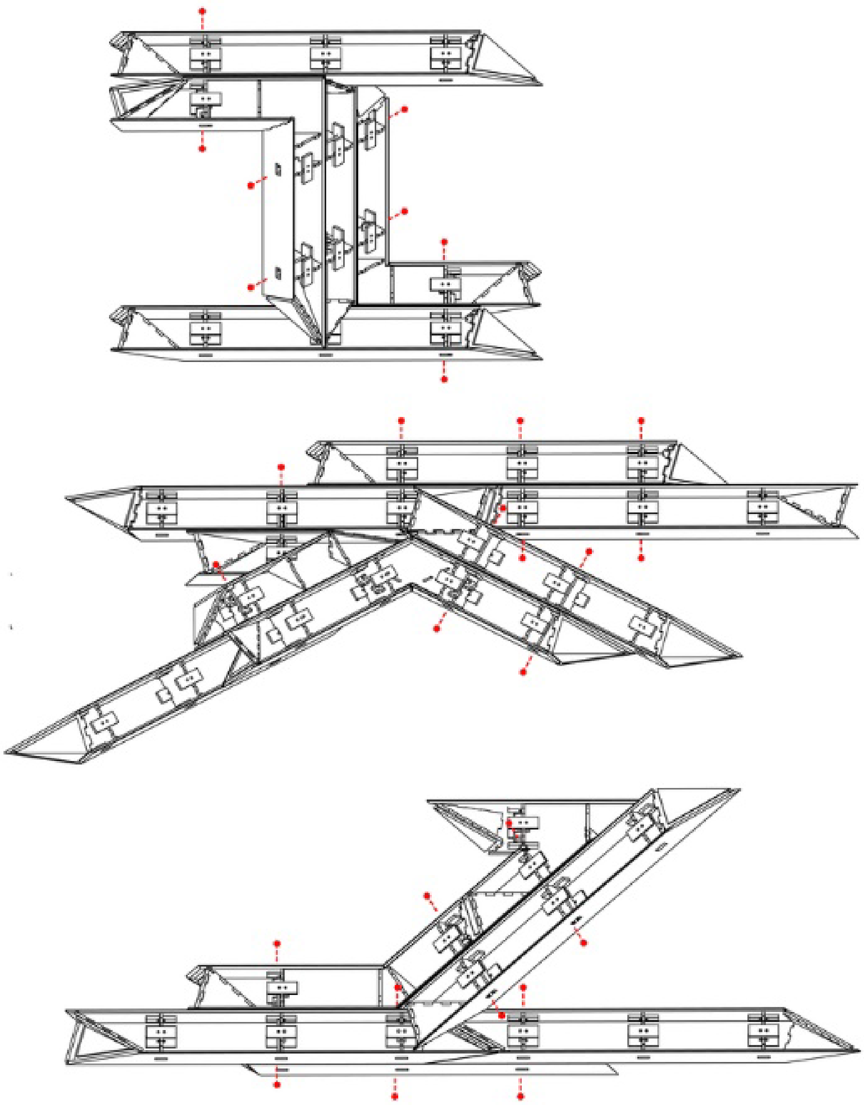
From a stack of assembled components to the constructed pavilion and back to a stack of components.



The constructed pavilion with some of the connection details shown.



Connection details of the components. The red lines represent the steel cables that tie the components together.



Different connection details of the components. The red lines represent the steel cables that tie the components together.

Appendix D: Six iterations of the section

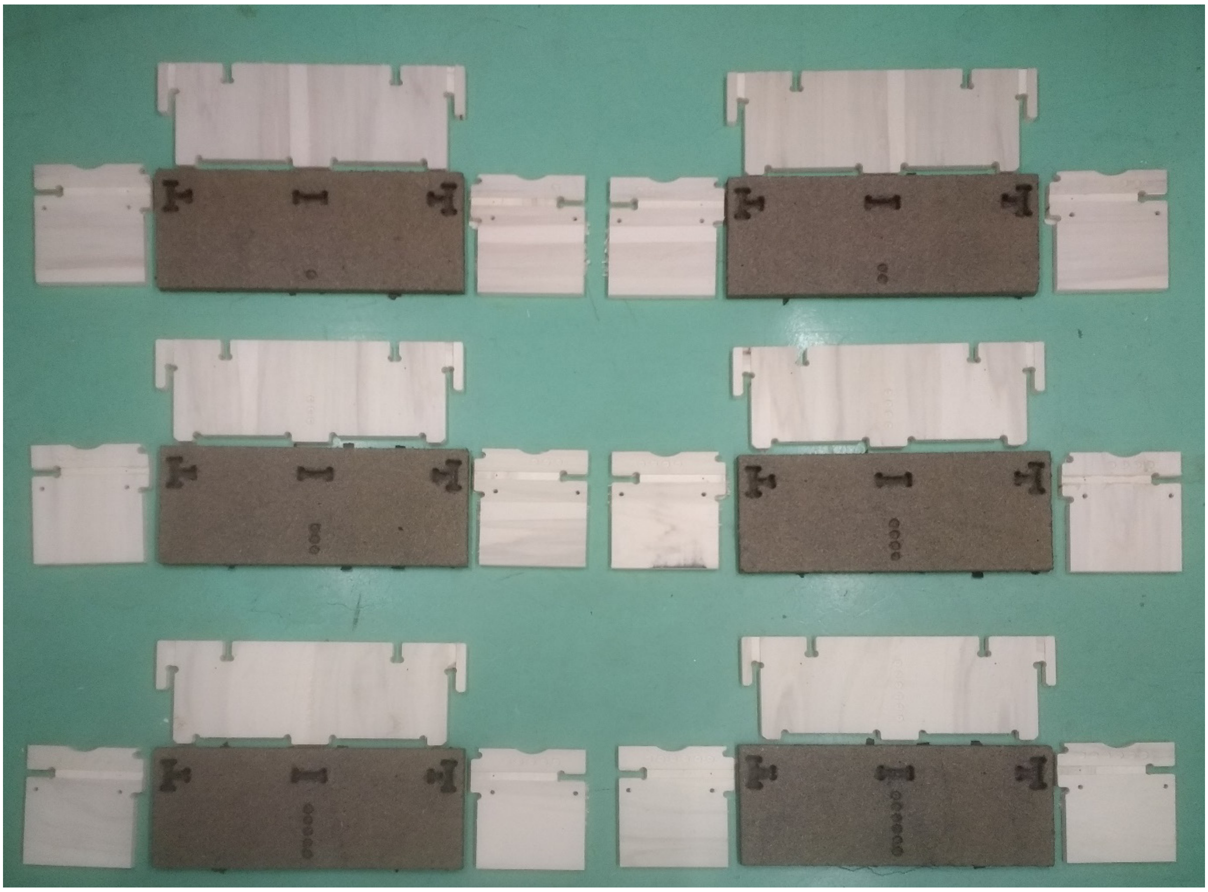


Photo with an overview of the six sections laid flat on the floor at HTCDelft.

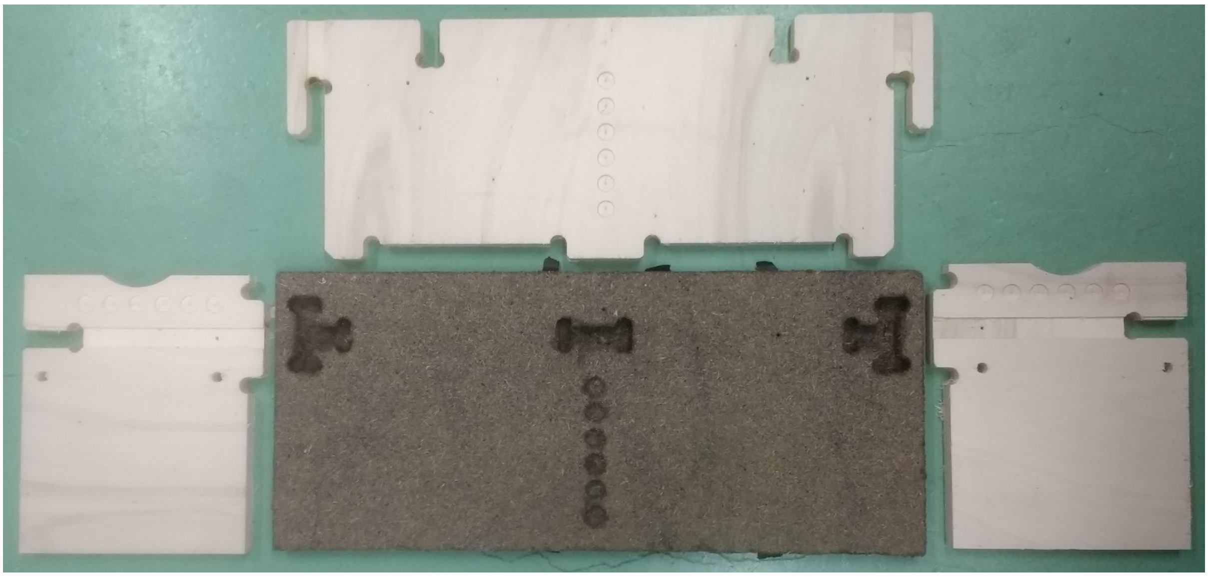
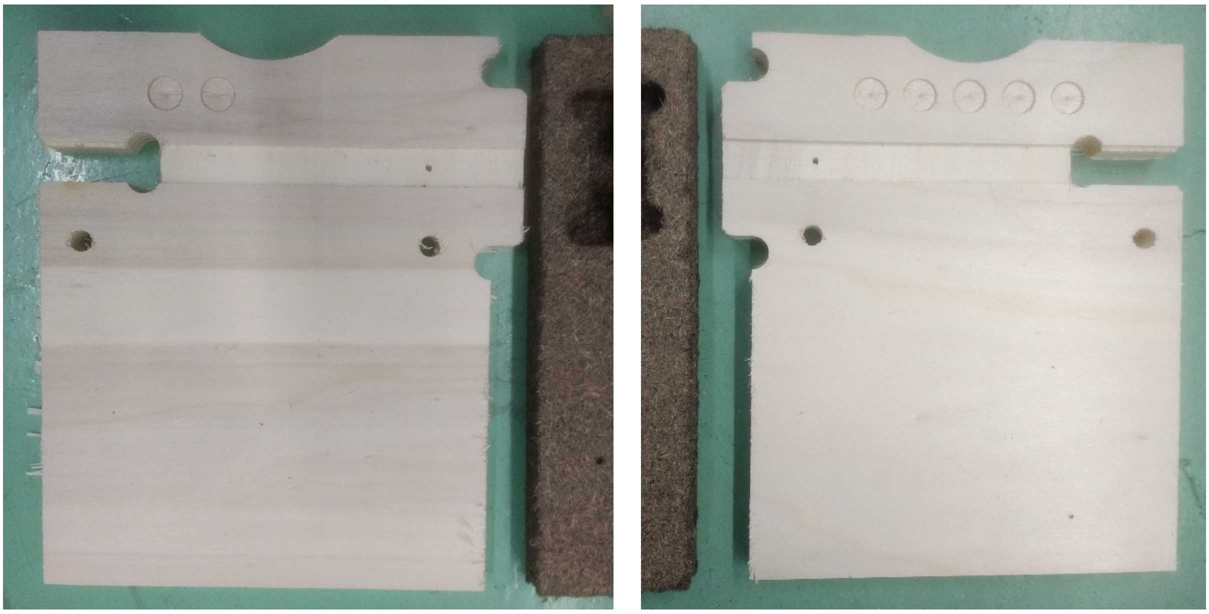


Photo of section number six laid flat.

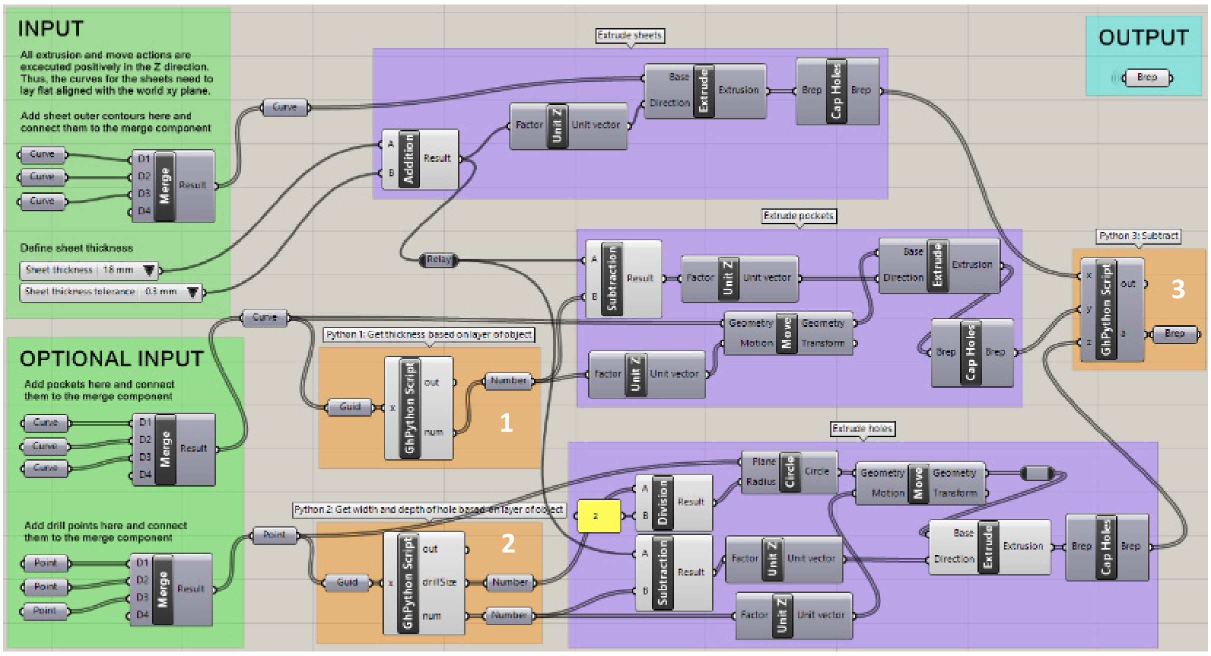


Two examples of how nicely milled elements look like. The left photo shows the left vertical beam of iteration two and the right photo shows the right vertical beam of iteration five.

Two examples of how the elements sometimes contains small defects. The left photo shows he bottom part of the right vertical beam of section one. The small circle highlights a splinter sticking out. The big circle marks a very thin layer of wood that should have been milled out. On the right photo the splintering, or tearing, at the groove is clearly visible.

Appendix E: GH modelling

Here below, an overview of the Grasshopper algorithm is shown. To make it readable a zoomed in image of the algorithm is presented after two pages. Although it is not readable, on the overview three orange rectangles can be seen. Each orange rectangle contains a Python script and a number in white. The scripts are shown here below with its corosponding number. Several comments are added behind the hashtag in green text in the Python scripts.



Overview of the whole Grasshopper algorithm.

Python script 1, which determines the depth of the pocket based on the layer of the object:

```
import Rhino
obj = Rhino.RhinoDoc.ActiveDoc.Objects.Find(x) # Refer to the object in Rhino
layerIndex = obj.Attributes.LayerIndex # Get layer number
layerName = Rhino.RhinoDoc.ActiveDoc.Layers[layerIndex].Name # Get layer name

splitChar = "+"
splittedName = layerName.Split(splitChar) # Split the layer name at the + symbol
num = splittedName[len(splittedName)-1] # Get the number
```

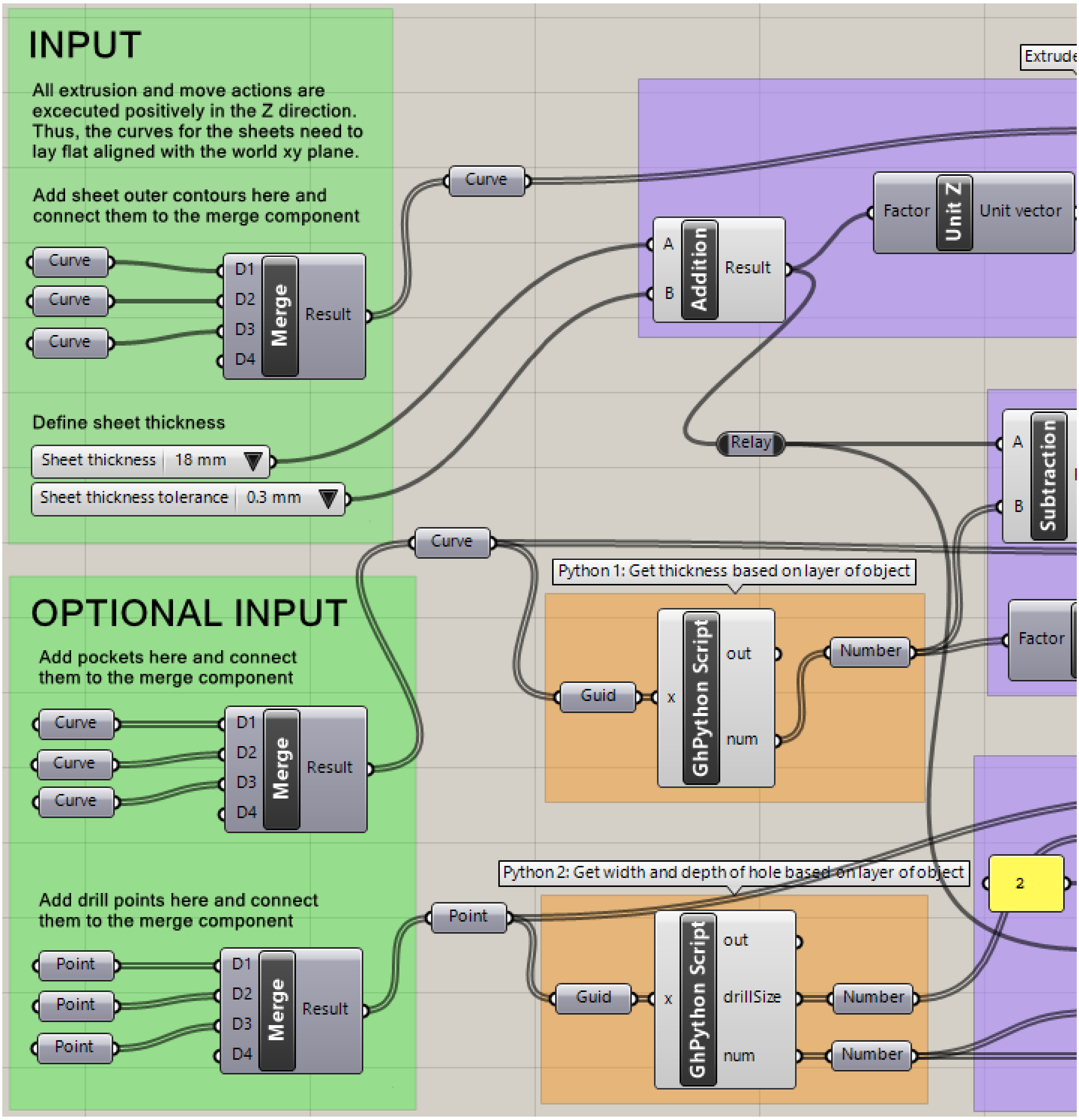
Python script 2, which determines the size of drill holes based on the layer of the object:

```
import Rhino
obj = Rhino.RhinoDoc.ActiveDoc.Objects.Find(x)
layerIndex = obj.Attributes.LayerIndex
layerName = Rhino.RhinoDoc.ActiveDoc.Layers[layerIndex].Name

splitChar = "+"
splittedName = layerName.Split(splitChar)
num = splittedName[len(splittedName)-1]

finds = "BOOR"
foundCharNum = layerName.find(finds)
diameterCharNumStart = foundCharNum+4 # Define string position of the drill depth
drillSizeString = layerName.Substring(diameterCharNumStart)

findStrMm = "MM"
drillSizeTotal = drillSizeString.split(findStrMm)
drillSize = drillSizeTotal[0] # Get first part of the string. This is the drill size
```

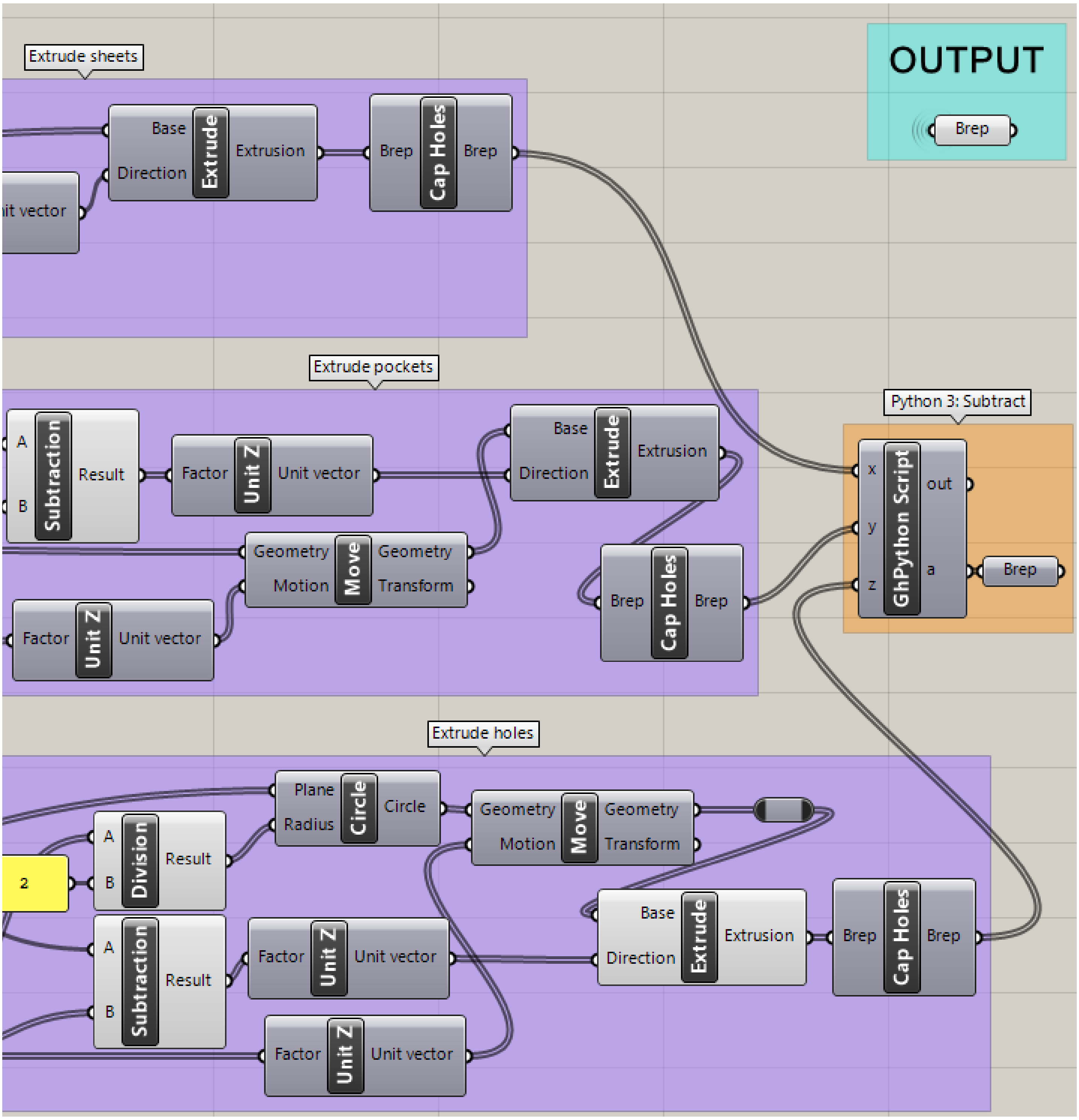
Python script 3, which subtracts the pockets and holes from the sheets and generates the 3D output:

```
import Rhino.Geometry as rg

tolerance = 0.01 # Set geometry tolerances required for brep subtract action

if x and not y and not z: # Check if only the sheet is given as input
    a = x
elif x and y and not z: # Check if the sheet and slots are given as input
    a = rg.Brep.CreateBooleanDifference(x,y,tolerance) # Subtract slots from sheet
elif x and z and not y:  # Check if the sheet and drill holes are given as input
    a = rg.Brep.CreateBooleanDifference(x,z,tolerance) # Subtract holes from sheet
elif x and y and z: # Check if the sheet, slots and drill holes are given as input
    temp = rg.Brep.CreateBooleanDifference(x,y,tolerance)
    a = rg.Brep.CreateBooleanDifference(temp,z,tolerance)
```

The Grasshopper algorithm is split up in two images to make it clearly readable, the left half is on this page and the right half is on the next page.

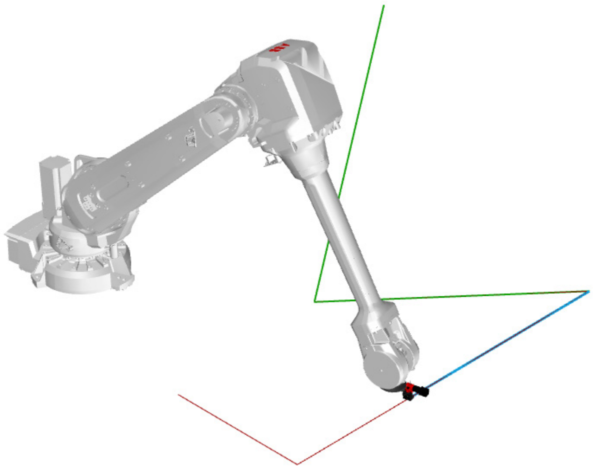


Left half of the Grasshopper algorithm.



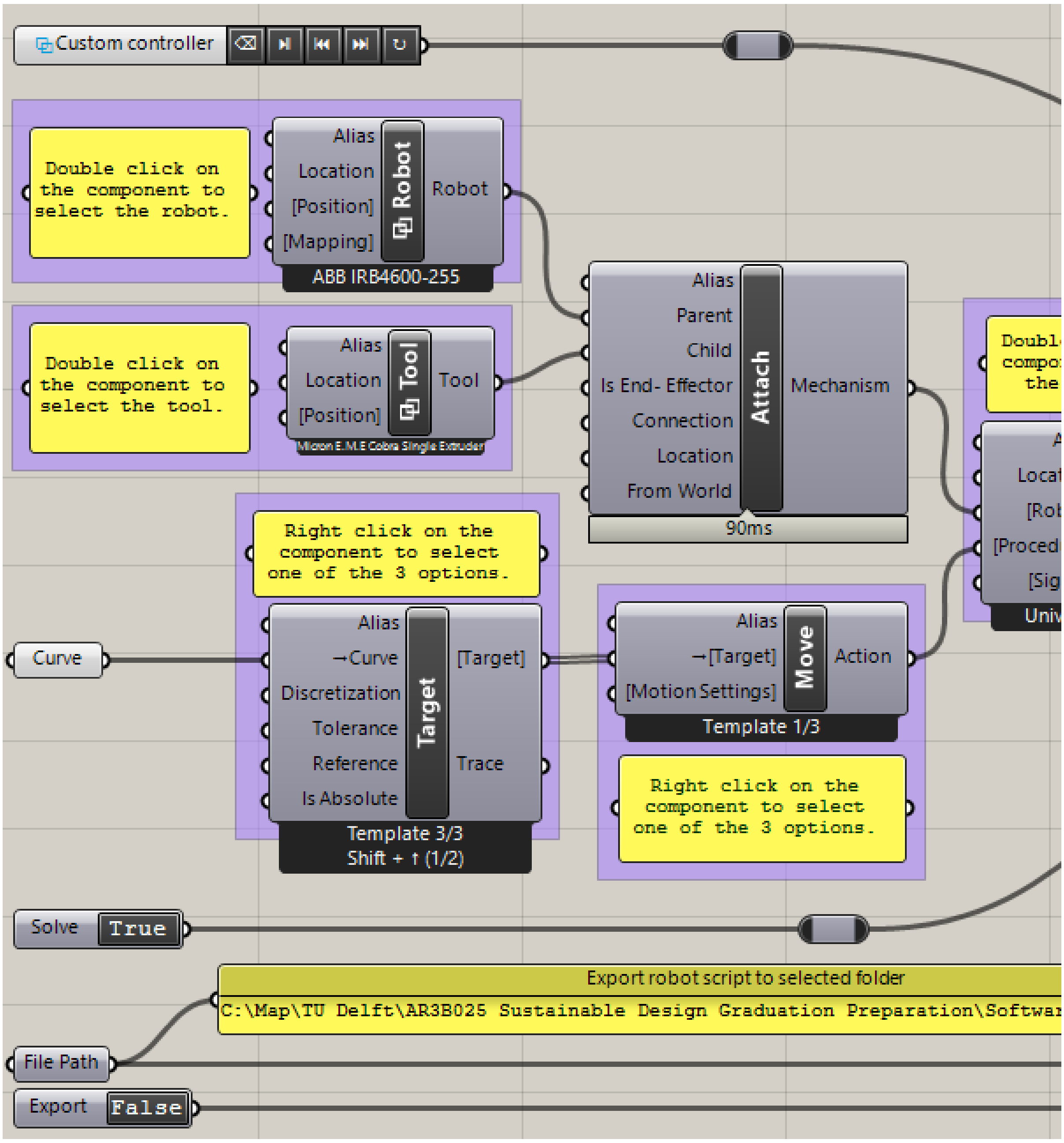
Right half of the Grasshopper algorithm.

Appendix F: GH test HAL-robotics

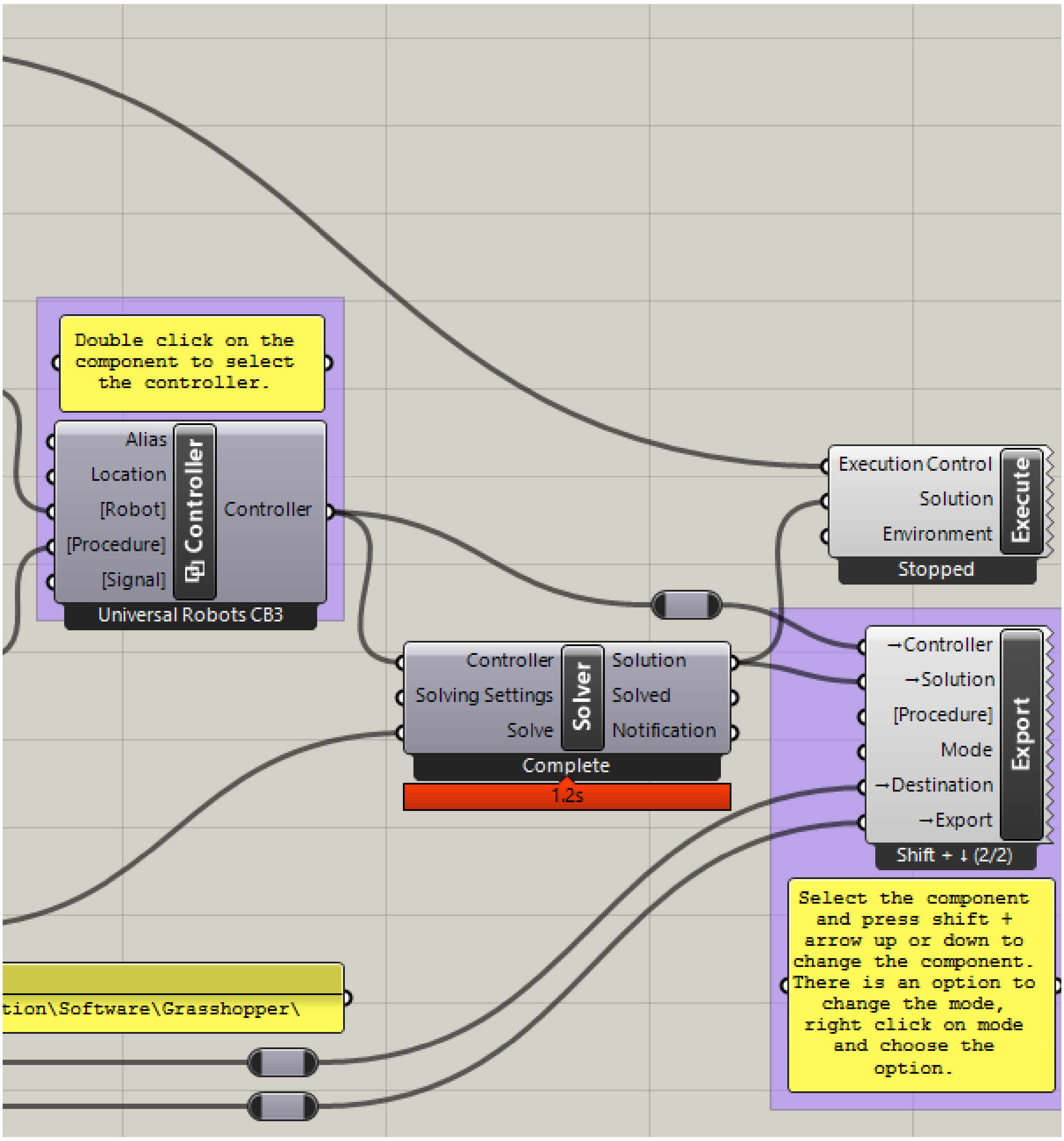


On the right, a 3D view in Rhino is shown in which a randomly chosen robot arm follows a curve drawn in Rhino. The robot is the IRB4600-255 from ABB.

The Grasshopper algorithm is split up in two images to make it clearly readable, the left half is on this page and the right half is on the next page.

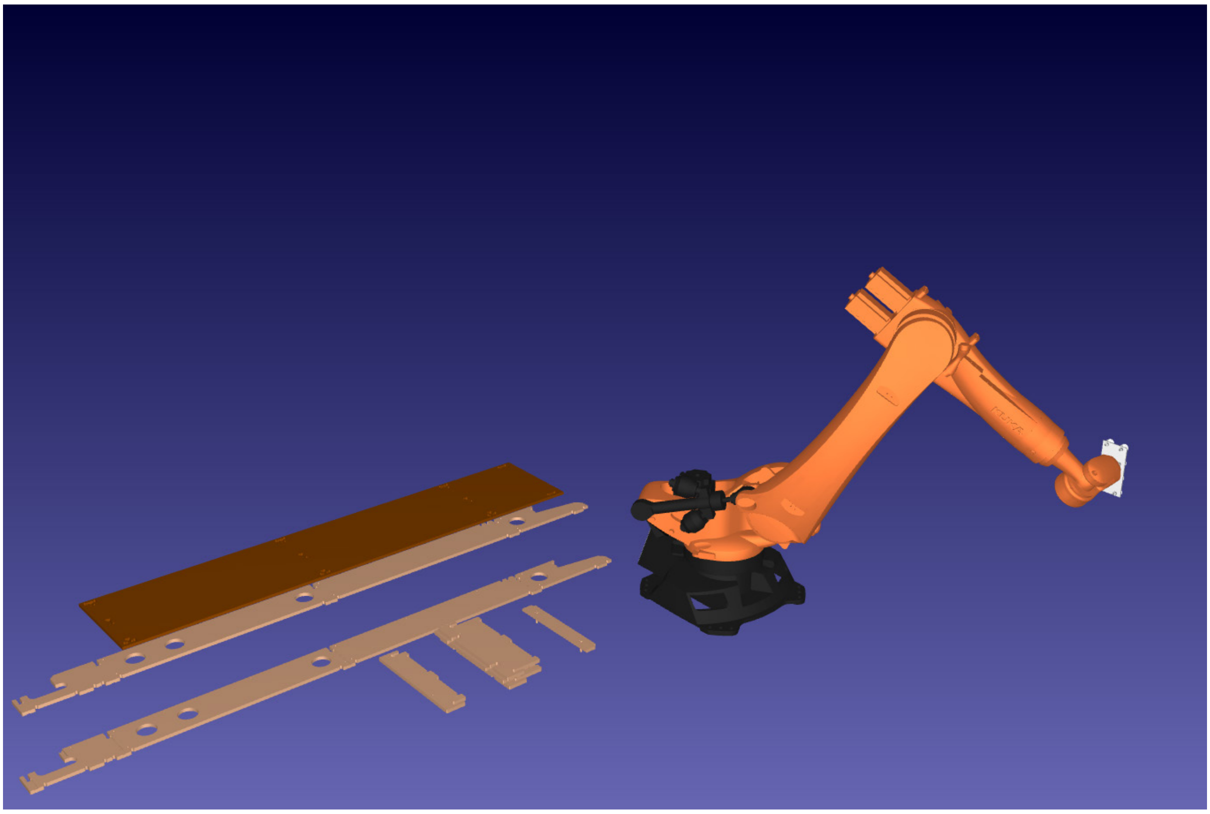


Left half of the Grasshopper algorithm.

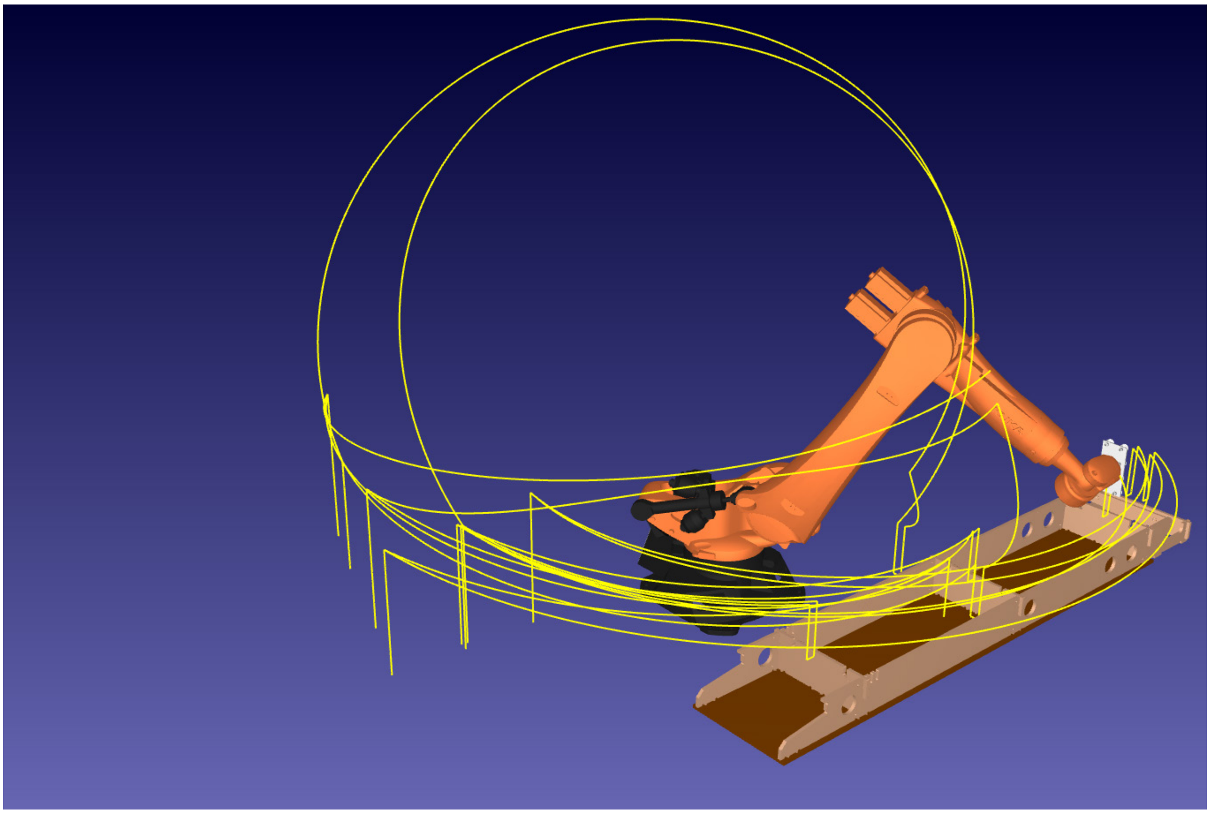


Right half of the Grasshopper algorithm.

Appendix G: RoboDK simulation test



Simulation test setup with the second best robot, the KR 120 R2900 extra.



Result of all elements put together. The yellow lines represent the toolpath of the robot during this part of the assembly process.

Appendix H: Python algorithm

The Python algorithm written for the simulation is shown here below. The file can be downloaded from GitLab via this shortened link: http://tiny.cc/gitlab (it redirects to this URL: https://gitlab.com/JBatGitLab/robotic-assembly-tnm/tree/master).

```python
# Assembly of a wall component of TheNewMakers www.thenewmakers.com

# Author: Jeroen de Bruijn
# Email: J.deBruijn1991@gmail.com

### INPUT

# PSEUDOCODE 1
# Variable for the robot name
RobotName = "Comau Smart5 NJ 110-3.0"

# Distance to move to on the selected frame Z axis when picking up an element
DistanceStartPos = 500
# Move distance over the selected frame Z axis before sliding an element in its final position
DistanceFinalPos = 450
# Move distance over the tool Z axis after sliding the element in its final position
DistanceTool = 50

# Variable for the frame in which the assembled component is located
FrameComp = "The component"

# Variable for the screwdriver tool
ScrewdriverGeom = "Screwdriver geometry"
ScrewdriveLoc = "Screwdriver location"



### FUNCTIONS

# PSEUDOCODE 2
# Function to reset the geometry
def ResetGeometry():
        # Show the screwdriver tool
        RDK.Item(ScrewdriverGeom).setVisible(True)
        # Define first part of the name of each element
        firstPart = "ELM."
        # Go trough each item
        for item in RDK.ItemList():
                if item.Valid():
                        name = item.Name()
                        # Check if the item is geometry
```

```
                        if item.Type() == 5:
                                # Check the name of the item to determine if it needs to be deleted
                                if name == tempGeom or name == "temp Screwdriver":
                                        # Delete the old stuff
                                        item.Delete()

                                # Check if first part of the item name matches with the beginning structure
                                if name[:len(firstPart)] == firstPart:
                                        # Show the geometry
                                        item.setVisible(True)

# Function to drill a screw
def screw(theItem):
        # Approach over the X axis of the screw tool
        approach = theItem.Pose()*transl(-DistanceTool,0,0)
        # Linear move to the approach position
        robot.MoveL(approach)
        # Move to the screw hole
        robot.MoveL(theItem)
        # Linear move to the approach position
        robot.MoveL(approach)


### START OF SCRIPT

# PSEUDOCODE 3
# Import library to communicate with RoboDK
from robolink import *
# Import library for robotics toolbox
from robodk import *

# Name for temporary geometry
tempGeom = "temp OJB geometry"

# Any interaction with RoboDK must be done through RDK
RDK = Robolink()
# Call the function to reset the geometry
ResetGeometry()

# Select the robot
robot = RDK.Item(RobotName)
# Get its tools, make sure in RoboDK that the vacuum gripper is the first tool and the screwdrive second
robotGripper = robot.Childs()[0]
robotScrewdriver = robot.Childs()[1]
# Hide the screwdriver tool in case it might still be visible
robotScrewdriver.setVisible(False)
# Just to make sure, set the vacuum gripper as active tool
robot.setPoseTool(robotGripper)

```

```
# Set robot joints to home position otherwise the robot sometimes makes unexpected maneuvers between two points
# The home position can be set manually in RoboDK.
# For the Comau Smart5 NJ 110-3.0 this is recommended: robot.setJoints([0,0,-90,0,0,0])
robot.setJoints( robot.JointsHome() )
#robot.setJoints([0,0,-90,0,0,0])

# PSEUDOCODE 4
# Construct empty matrix (2d list) as long and wide as there are number of elements
# The matrix is to big, but will be made shorter later on
    # If the matrix is to short the elements can not be placed on the right position in the matrix
    items = [[None for x in range(len(RDK.ItemList()))] for y in range(len(RDK.ItemList()))]

    # Define first part of the name of each element
    firstPart = "ELM."
    # Go trough each item
    for item in RDK.ItemList():
        if item.Valid():
            name = item.Name()
            # Check if first part of the item name matches with the beginning structure
            if name[:len(firstPart)] == firstPart:
                # Disregard the first part and split the string at a .
                data = name[len(firstPart):].split(".")
                # Add item to correct location in the matrix
                items[ int(data[0]) ][ int(data[1]) ] = item

    # Remove empty lists from matrix
    items = list(filter(any, items))
    # Remove empty items from lists in matrix
    for i, row in enumerate(items):
        items[i] = [x for x in row if x is not None]

    # PSEUDOCODE 5
    # It is important to provide the reference frame and the tool frames when generating programs offline
    robot.setPoseFrame(robot.PoseFrame())
    robot.setPoseTool(robot.PoseTool())

    # Retrieve the robot reference frame
    robotReference = robot.Parent()

    # Retrieve the assembled component reference frames
    componentReference = RDK.Item(FrameComp)
    # All  move actions should be made relative to assembled component reference
```

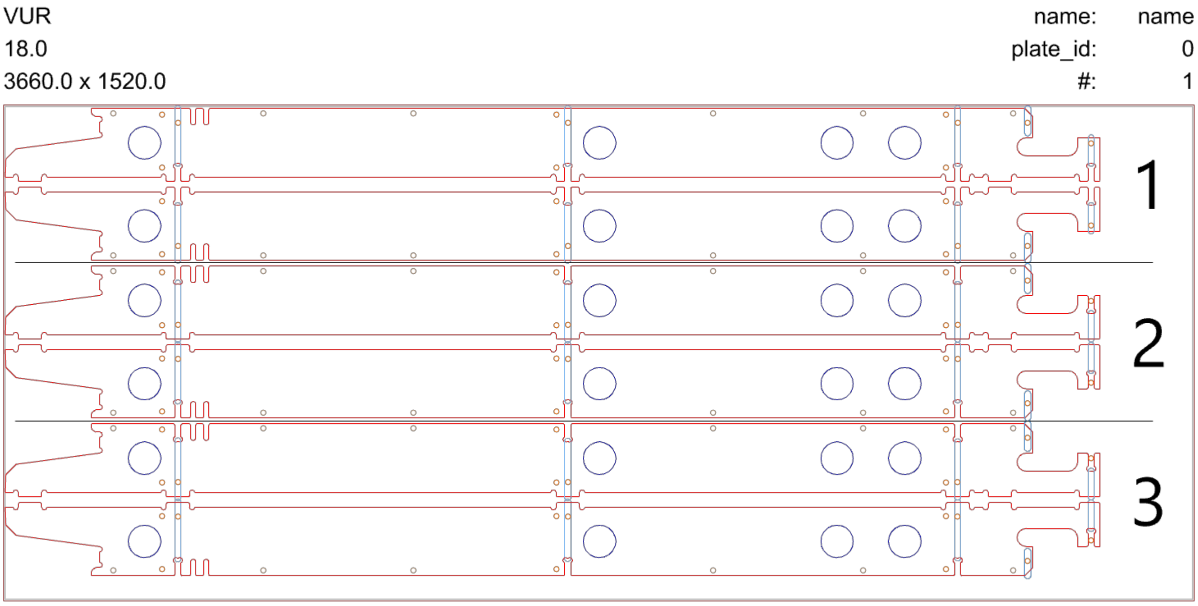
```
    robot.setPoseFrame( componentReference )

    # PSEUDOCODE 6
    # Loop trough all list in the matrix
    for aList in items:
        # Variable for the first item in the current list
        firstItem = aList[0]

        # PSEUDOCODE 6.1
        # Check if the first item in the list is geometry
        if firstItem.Type() == 5:
            # Determine last step number
            lastStep = len(aList[1:])-1
            # Loop trough each item, but skip first item because that is the geometry
            for i, item in enumerate(aList[1:]):
                # Check if it is the first or last step
                if i == 0 or i == lastStep:
                    if i == 0:
                        # Copy item location
                        approach = item.Pose()
                        posItem = approach.Pos()
                        # Set Z-value
                        posItem[2] = DistanceStartPos
                        # Update the approach location with the new Z-value so the approach will always have the same Z-value
                        approach.setPos(posItem)
                    else:
                        # Calculate approach position over the world Z axis
                        approach = Offset(item.Pose(), 0, 0, DistanceFinalPos)

                    # Joint move to the approach position
                    robot.MoveJ(approach)
                    # Linear move down to the item
                    robot.MoveL(item)
                else:
                    # Joint move to the item
                    robot.MoveJ(item)

                # Check if it is the first point and the geometry need to be picked up
                if i == 0:
                    # Copy the geometry
                    aList[0].Copy()
```

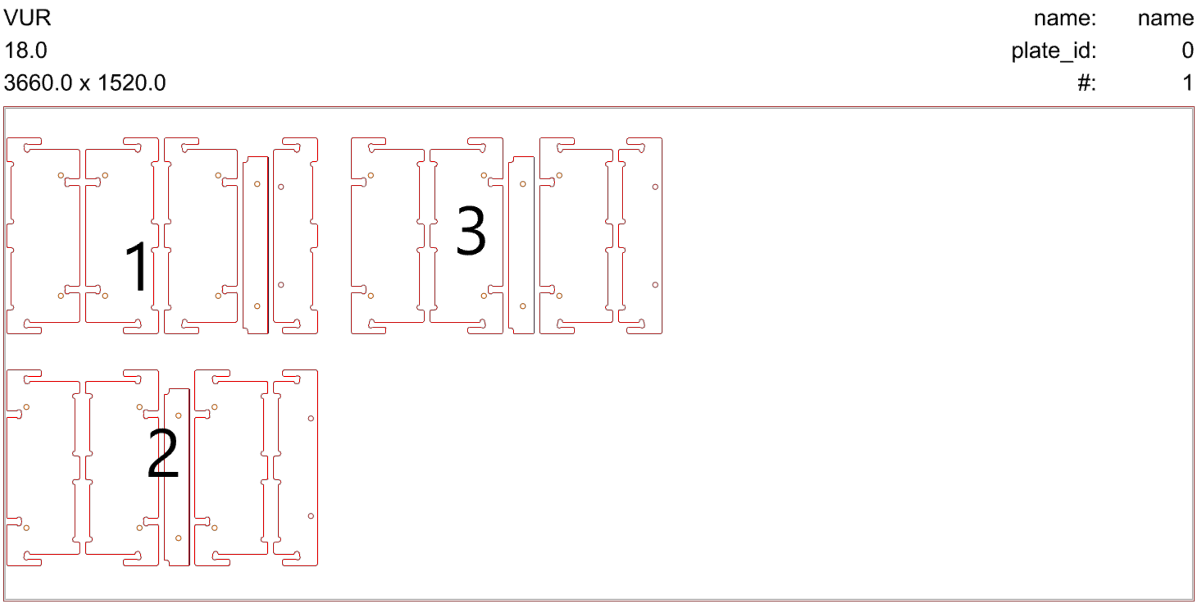
```
                    # Paste the geometry to the same parent and thus the same location
                    item_temp = RDK.Paste(aList[0].Parent())
                    # Rename the geometry
                    item_temp.setName(tempGeom)
                    # Connect the geometry with the robot tool without changing its absolute position
                    item_temp.setParentStatic(robotGripper)

                    # Hide the geometry
                    aList[0].setVisible(False)

                    # Linear move up to the approach position
                    robot.MoveL(approach)
                elif i == lastStep:
                    # Connect the geometry with the finished component without changing its absolute position
                    item_temp.setParentStatic( componentReference )

                    # Leave over the Z axis of the tool instead of the world Z axis
                    approachLeaveFirst = item.Pose()*transl(0,0,-DistanceTool)
                    # Linear move to the approach leave position
                    robot.MoveL(approachLeaveFirst)
                    # Calculate second approach position over the world Z axis
                    approachLeaveSecond = Offset(approachLeaveFirst, 0, 0, DistanceFinalPos)
                    # Linear move to the approach leave position
                    robot.MoveL(approachLeaveSecond)
        # PSEUDOCODE 6.2
        # Check if screws need to be added
        elif "SCREWS" in firstItem.Name():
            # Get location of the screwdriver tool
            locationScrew = RDK.Item(ScrewdriveLoc)
            # Copy location
            approach = locationScrew.Pose()
            posItem = approach.Pos()
            # Set Z-value
            posItem[2] = DistanceStartPos
            # Update the approach location with the new Z-value so the approach will always have the same Z-value
            approach.setPos(posItem)

            # Set the screwdriver as the active tool
```

```
        robot.setPoseTool(robotScrewdriver)
        # Joint move to the approach location
        robot.MoveJ(approach)
        # Linear move down to the location
        robot.MoveL(locationScrew)

        # Show screwdriver tool
        robotScrewdriver.setVisible(True)
        # Hide the screwdriver geometry
        RDK.Item(ScrewdriverGeom).setVisible(False)

        # Linear move up to the approach location
        robot.MoveJ(approach)

        # Loop trough all items in the list, except for the first item
        for item in aList[1:]:
            # Check if screw is in the item name
            if "SCREW" in item.Name():
                # Call the screw function
                screw(item)
            else:
                # Joint move to the item
                robot.MoveJ(item)

        # Joint move to the approach location
        robot.MoveJ(approach)
        # Linear move down to the location where the screwdriver should be placed
        robot.MoveL(locationScrew)

        # Set the vacuum gripper as active tool
        robot.setPoseTool(robotGripper)
         # Show the screwdriver geometry
        RDK.Item(ScrewdriverGeom).setVisible(True)
        # Hide screwdriver tool
        robotScrewdriver.setVisible(False)

        # Linear move up to the approach location
        robot.MoveL(approach)
    # PSEUDOCODE 6.3
    # Check if all geometry needs to be moved
    elif "ALL" in firstItem.Name():
        # Calculate approach location over the world Z axis
        approach = Offset(firstItem.Pose(), 0, 0, DistanceFinalPos)
        # Joint move to the approach location
        robot.MoveJ(approach)
        # Linear move down to the location
        robot.MoveL(firstItem)

        # Go trough each item in RoboDK
        for item in RDK.ItemList():
```

```
                if item.Valid():
                    # Check if name matches the geometry
                    if item.Name() == tempGeom:
                        # Connect the item with the robot tool without changing its absolute position
                        item.setParentStatic(robotGripper)

            # Linear move up to the approach location
            robot.MoveL(approach)

            # If the list contains 3 items or more, than it has points in between as well
            if len(aList)>=3:
                # Loop trough all items in the list, except for the first and last
                for item in aList[1:-1]:
                    # Joint move to the item
                    robot.MoveJ(item)

            # Get the last item
            lastItem = aList[-1]

            # Copy location
            approach = lastItem.Pose()
            posItem = approach.Pos()
            # Set Z-value
            posItem[2] = DistanceStartPos
            # Update the approach location with the new Z-value so the approach will always have the same Z-value
            approach.setPos(posItem)

            # Joint move to the approach location
            robot.MoveJ(approach)
            # Linear move down to the location
            robot.MoveL(lastItem)

            # Go trough each item in RoboDK
            for item in RDK.ItemList():
                if item.Valid():
                    # Check if name matches the geometry
                    if item.Name() == tempGeom:
                        # Connect the geometry with the finished component without changing its absolute position
                        item.setParentStatic( componentReference )

            # Linear move up to the approach location
            robot.MoveL(approach)
```

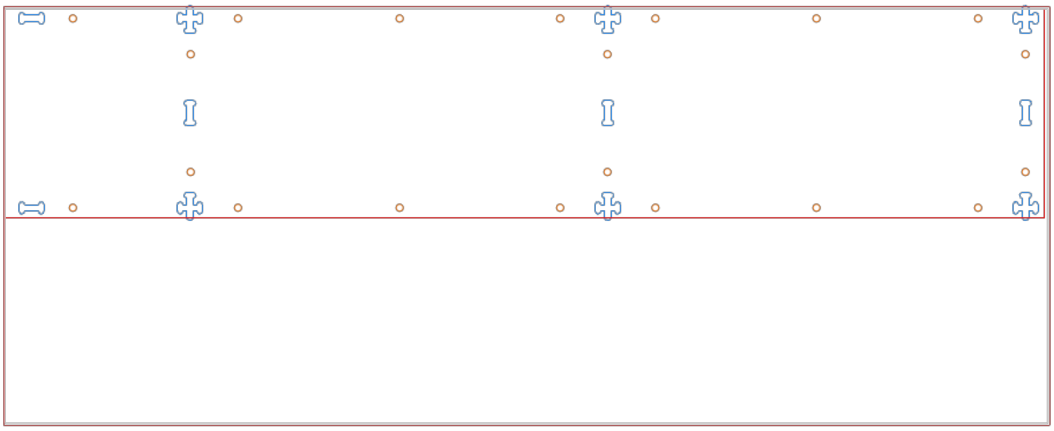
Appendix I: Drawings for CNC milling



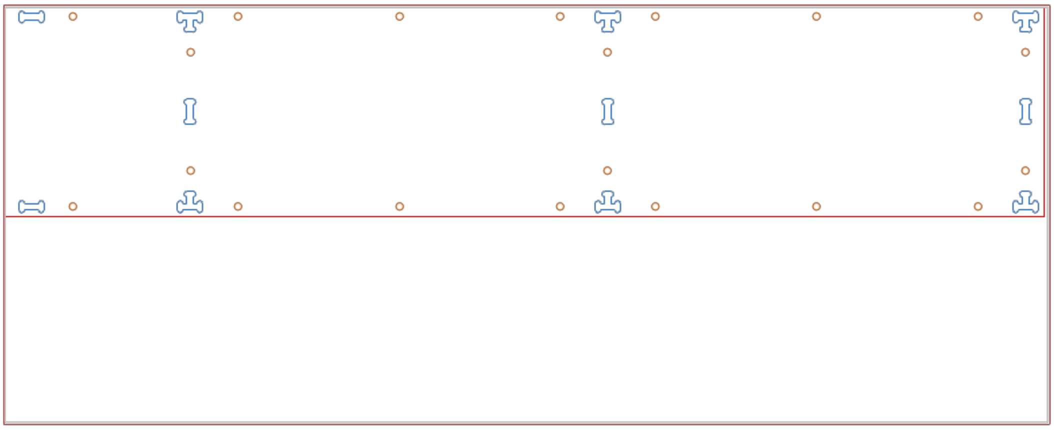


Drawing of the vertical beams nested on the top sheet and all the horizontal beams are nested on the bottom sheet.

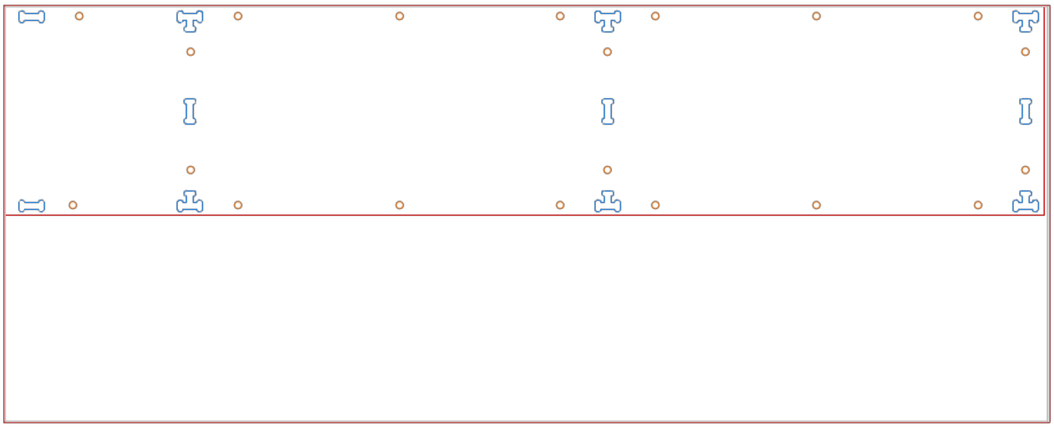
VUR name: name
18.0 plate_id: 1
3000.0 x 1200.0 #: 1



VUR name: name
18.0 plate_id: 2
3000.0 x 1200.0 #: 1



VUR name: name
18.0 plate_id: 3
3000.0 x 1200.0 #: 1



Drawing of each outer layer on three different sheets.

Appendix J: Dutch report HBO students

On the next page starts the final report that is written in Dutch by the HBO mechatronic students whom are studying at The Hague University of Applied Sciences in Delft. Two reports have been made, being an interim report and a final report. Only the following chapters are shown: summary, table of contents, introduction, summary of the interim report, conclusions and recommendations. The full version of both reports are available upon request.

THE NEW MAKERS

DE HAAGSE HOGESCHOOL

EINDVERSLAG

MeH3.C
Projectgroep: H

- Do van den Berg 16021053
- Dexter Bregman 16019733
- Jeroen Koot 14142783
- Per Kerssens 16023501

Samenvatting

The New Makers is een bedrijf dat zich bezighoudt met produceren en assembleren van houten constructies voor in kantoor ruimtes. Er is een slimme constructie bedacht om eenvoudig kantoor ruimtes op te bouwen. Ze proberen dit op een zo duurzaam mogelijke manier te doen.

The New Makers wil gaan kijken of het mogelijk is om het assemblage deel te automatiseren. Dit willen ze gaan doen doormiddel van een robot arm die verschillende stukken hout moet gaan oppakken en assembleren. Als dit mogelijk is dan kan dit veel mogelijkheden bieden voor het bedrijf om het groter te gaan oppakken. Ze zouden bijvoorbeeld snellers orders kunnen verwerken.

In dit rapport staat de volgende vraag centraal: Kan het assemblage onderdeel geautomatiseerd worden doormiddel van een assemblage robot en een vision systeem.

Om deze vraag te kunnen beantwoorden is er met behulp van een camerasysteem op een robotarm een prototype gerealiseerd. Dit prototype maakt gebruik van een combinatie van OpenCV2 en deep learning om de locatie, het type en de verdraaiing van een wandelement te detecteren. Vervolgens wordt het element opgepakt door een 6-assige robotarm met vier zuignappen. Na het oppakken wordt het element op een positioneertafel geplaatst, waarna deze wordt geassembleerd.

Uit de testresultaten blijkt dat het prototype voldoet aan 17 van de 20 vooraf gestelde testen. Hieruit kan geconcludeerd worden dat het systeem in staat is autonoom een wandelement te assembleren, zolang er geen verschillende elementen bovenop elkaar liggen en de vezelplaat reeds op de assemblageplaats is vastgezet.

De belangrijkste aanbevolen actiepunten zijn extra onderzoek in het vision-systeem, wat met een betere camera, een betere ondergrond/belichting en een snellere computer elementen beter kan herkennen, aanpassingen aan de robotarm waardoor het mogelijk is om de volledige wandelementen inclusief houtvezelplaat op te pakken en te assembleren. Tevens kan er worden gekeken naar een positioneertafel van betere materialen, wat de duurzaamheid en nauwkeurigheid van het systeem verhoogt.

Inhoudsopgave

1 Inleiding

Voor het assembleren van wandelementen worden nu werknemers gebruikt. Dit is zwaar en intensief werk wat kan leiden tot klachten in het menselijk lichaam. Daarom wordt er gekeken naar een alternatieve oplossing voor dit probleem. Het probleem kan op verschillende manieren worden opgelost er is gekozen voor een oplossing met een assemblage robot. Deze assemblage robot moet het werk van de werknemers gaan overnemen.

Aan Dhr. J de Bruijn is gevraagd om onderzoek te doen naar het globale productie- en assemblageproces, en om in samenwerking met de projectgroep een beeld te vormen naar de mogelijkheden voor het automatiseren van het assemblage-gedeelte van het proces.

De opdrachtgever van dit project is Pieter Stoutjesdijk, namens The New Makers. Dit bedrijf maakt modulaire prefab ruimtes voor verschillende doelgroepen. Deze ruimtes worden gemaakt van duurzaam plaatmateriaal wat na gebruik weer makkelijk demonteerbaar is en elders kan worden geplaatst.

De opdrachtnemer is Groep H. Studenten van de opleiding Mechatronica op de Haagse Hogeschool te delft. Mechatronica is een opleiding die opgebouwd is uit drie verschillende studies zoals Elektrotechniek, werktuigbouw en Technische informatica. De opdracht wordt uitgevoerd op de Haagse hogeschool en in het Hightechcenter te delft.

Dit einderslag beantwoordt de volgende vraag: Kan het assemblage onderdeel geautomatiseerd worden doormiddel van een assemblage robot en een vision systeem.

Als deze assemblage robot werkt, kan The New Makers overwegen om het product op ware grote te ontwikkelen, produceren en te gebruiken

De opbouw van dit eindverslag is als volgt. Na deze inleiding volgt in hoofdstuk 2 een samenvatting van de tussenrapportage uit maart 2019. In hoofdstuk 3 staat het concept van de operatie. In hoofdstuk 4 wordt de beeldherkenning besproken. Vervolgens beschrijft hoofdstuk 5 de robotarm Er wordt in hoofdstuk 6 ingegaan op de realisatie van de end-effector. Hierna wordt in hoofdstuk 7 inzicht gegeven in de positioneertafel. In hoofdstuk 8 wordt de datacommunicatie uitgelegd. Hierna wordt in hoofdstuk 9 de mogelijkheden voor een human machine interface verduidelijkt. Erna volgt in hoofdstuk 10 de acceptencetest. Vervolgens wordt er in hoofdstuk 11 een conclusie getrokken en in hoofdstuk 12 enkele aanbevelingen gedaan. In hoofdstuk 13 staan de figuren die in dit document zijn geplaats. Het verslag sluit af met verschillende bijlages, waaronder in bijlage 9 het research framework van Jeroen de Bruijn, een afstudeerder van de TU Delft die parallel aan dit project het volledige werkproces van TNM beschouwd.

2 Samenvatting tussenrapportage maart 2019

The New Makers wil gaan kijken of het mogelijk is om het assemblage deel te automatiseren. Dit willen ze gaan doen doormiddel van een robot arm die verschillende stukken hout moet gaan oppakken en assembleren. Als dit mogelijk is dan kan dit veel mogelijkheden bieden voor het bedrijf om het groter te gaan oppakken. Ze zouden bijvoorbeeld snellers orders kunnen verwerken.

Hiervoor zijn verschillende punten met elkaar vergeleken namelijk: doeltreffendheid, effectiviteit, haalbaarheid en efficiëntie. Daarbij zijn de belangen en wensen in kaart gebracht van de betrokken partij: The New Makers. Een belangrijke vraag voor The New Makers is of met een robot arm het mogelijk is om een gehele wand te assembleren.

Aan de hand van een methodisch ontwerpproces is ervoor gekozen om een prototype te realiseren wat met behulp van deep-learning beeldherkenning verschillende elementen herkent, waarna deze met behulp van een vacuümgrijper aan een robotarm de elementen oppakt en assembleert.

Uit de testresultaten blijkt dat het prototype nog niet voldoet aan de vooraf gestelde eisen. Hieruit kan geconcludeerd worden dat het systeem nog niet in staat is autonoom een wandelement kan assembleren. De deelsystemen zijn los van elkaar in staat om een of meerdere deelfuncties te vervullen, maar de deelsystemen werken nog onvoldoende samen en zijn nog niet volledig gerealiseerd.

De belangrijkste aanbevolen actiepunten zijn om de deelsystemen verder te realiseren, te onderzoeken of met een positioneertafel de nauwkeurigheid van het systeem kan worden verhoogt en om te onderzoeken of er een mogelijkheid is om nieuwe typen elementen aan het systeem toe te kunnen voegen. Ook is het verstandig om kritisch te kijken naar andere manieren van detectie van de wandtypen in het geval er gebruikt wordt gemaakt van de positioneertafel.

11 Conclusie

In dit dubbele project hebben wij ons bezig gehouden met het automatiseren van een productieproces voor The New Makers. Hiermee hebben we onderhand onszelf de volgende deelvragen gesteld: Is beeldherkenning een geschikte vorm van detectie van de elementen, kunnen we met een robotarm de elementen assembleren tot een wanddeel. Aangezien het ons gelukt is om met deze middelen een wanddeel te assembleren kunnen we deze vragen beantwoorden met een ja.

De robotarm beschikt over onze eigen ontworpen end-effector. Met de vier zuignappen kunnen we alle delen van de wandmuur oppakken behalve het bodem deel. Dit deel is namelijk gemaakt van geperste vezelplaat waar we door de vezels te veel luchtdruk verliezen om hem op te pakken. Dit was zeker een tegenslag in dit project en iets waarvan we geleerd hebben.

In tegenstelling tot het eerste deel van dit project beschikt de robotarm nu tot een camera met beeldherkenning waarmee coördinaten kunnen worden door gegeven aan de computer zodat er via deze camera elementen kunnen worden geplaatst. De beeldherkenning is in vergelijking met het vorige verslag verbeterd naar het herkennen van alle twee de onderdelen en berekend hierbij ook de hoek waaronder het element ligt. Hierdoor kan de robot alle elementen met de juiste hoek oppakken en plaatsen in de positioneertafel. Er zijn nog geen testen gedaan voor de beeldherkenning met overlappend elementen.

The New Makers krijgen met dit onderzoek een idee van hoe een automatische productie proces er voor hun uit komt te zien en kunnen hiermee verder werken naar een realistische automatisering op grotere schaal.

12 Aanbevelingen

12.1 Vision

- Om alle onderdelen goed te kunnen zien is er een grotere tafel nodig om de elementen op te leggen met een zelfde kleur achter als waarop getraind is. Vervolgens moeten aan deze tafel op elke hoek goede lichten worden gemonteerd zodat er weinig schaduw is van de onderdelen.
- Een camera met een grotere invalshoek waardoor de onderdelen in één geheel kunnen worden gezien.
- Een snellere computer zodat de software die het vision programma runt beter te werk kan gaan en zo sneller elementen kan herkennen.
- Een duidelijke markering vanuit de machine zodat onderdelen makkelijker herkent kunnen worden.

12.2 Robotarm & end-effector

- Een vacuümejector of vacuümpomp met een hogere flow testen.
- De houtvezelplaten zijn momenteel aan de onderkant gelamineerd. Hierop kan wel een geschikte onderdruk worden bereikt met de vacuümejectors. Een plaat kan aan de onderkant worden vast gepakt. De tafel of klem waarop de plaat geplaatst wordt moet dan voorzien zijn van openingen voor de zuignappen.
- De houtvezelplaten worden opgepakt door middel van een mechanische grijper. Dit moet echter wel voorzichtig gebeuren omdat de platen erg slap en bros zijn.
- Het oppakken van de houtvezelplaten kan worden gedaan met behulp van zogenaamde naaldgrijpers. Deze grijpers grijpen zich in het materiaal vast met kleine naaldjes. De kosten zijn echter aanzienlijk hoger dan de andere oplossingen.
- Voor het oppakken van het gehele wandelement is een robotarm nodig met een hogere maximale payload.

12.3 Positioneertafel

Voor het verbeteren van de positioneertafel kan er gekeken worden naar het ontwerp en de gebruikte materialen in het huidig ontwerp. Door de kleine omvang is er tijdens dit project voor gekozen om een combinatie van 3d-gepriten onderdelen en hout te gebruiken dit is namelijk snel toepasbaar. Op grote schaal kan er beter gebruik worden gemaakt van metaal om de stevigheid te garanderen en het glijvlak glad te houden. De constructie kan dan bestaan uit drie staanders van de juiste lengte en in de juiste hoek met de plaat dit zorgt nogmaals voor stevigheid en voorkomt over bepaling van de constructie.

12.4 Camera

Verbeteringen aan de camera kunnen gerealiseerd worden door in eerste instantie uiteraard het kopen van een meer professionelere camera. In de markt zien we dat er bijvoorbeeld veel Cognex camera gebruikt worden voor dit soort toepassingen. Ook kan er gekeken worden naar het licht rondom de camera door dit lichte constanter en lichter te maken kan er voor een beter beeldherkenning gezorgd worden.