Eindhoven University of Technology

MASTER

A framework on how to cope with geometric uncertainties of robotic manufacturing processes of brick structures

Hendriks, Amy L.

Award date:
2022

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Department of the Built Environment
Architecture, Building and Planning
Structural Engineering and Design

A framework on how to cope with geometric uncertainties of robotic manufacturing processes of brick structures

A.L. Hendriks
0990848

Version 2

Eindhoven, April 2022

Colophon

General

Title	A framework on how to cope with geometric uncertainties of robotic manufacturing processes of brick structures
Defense Date	08-04-2022

Student

Author	A.L. (Amy Louise) Hendriks
Student ID	0990848
Email	a.l.hendriks.1@student.tue.nl (al.hendriks@hotmail.nl)
University	Eindhoven University of Technology
Faculty	Department of the Built Environment
Master Program	Architecture, Building and Planning
Master Track	Structural Engineering and Design

Graduation Committee

Chairman	Prof. Dr. Ing. P.M. (Patrick) Teuffel p.m.teuffel@tue.nl
Supervisor	Ir. A.P.H.W. (Arjan) Habraken a.p.h.w.habraken@tue.nl
Supervisor	Dr.Ir. R.J.M. (Rob) Wolfs r.j.m.wolfs@tue.nl

Preface

This past year has been an absolutely joyful ride: I got to work on a project that kept me enthusiastic all the year around and with people that supported me during every step of the way. When you would ask my 18 year old self if I would ever graduate on a topic involving robots I would have probably never believed it. However, during the master track Structural Engineering and Design at the University of Technology Eindhoven (TU/e) my curiosity took over and before I knew it I was talking with Rob and Arjan about possible projects with these robots. This research kept me curious en motivated along the way and I hope that that is what you experience when you read this thesis.

As mentioned, I am grateful for every person that joined me along this project. Without the immense support of everyone around me I would not have been able to bring this graduation to a success. However, there are some people in particular that I would like to thank for their contribution.Firstly, I would like to express my gratitude towards my supervisors, Rob, Arjan and Patrick. Without your expertise on the subject and sometimes critical views on what I have been doing it would not have been possible to produce the same end result as I did now. During the supervisions I got excited by the way that you were enthusiastic about my research and I could not have had a better motivation than that, so I would like to thank you all for this! I would also like to thank my family for their infinite support throughout everything and every step that I have taken, not only in this research but in my academic career and basically my whole life up to this point and the years to come. I could always express my enthusiasm for the projects that I worked on even though they are sometimes not in your field of interest. However, you always taught me to be critical towards myself, open-minded and go with 100% for it and I appreciate it very much. The last person that I would like to thank is Sander. I know that sometimes I talk a lot about robots or pick-up points or Grasshopper and at some times is was probably not interesting at all but I am grateful for how you helped me along the way. Even when I was not that enthusiastic or optimistic at some moments you got me right back on track and reminded me why I do the things that I do. Of course I also want to thank everyone else that have been somehow a part of my journey so far, if it is in my personal, professional or academical development, I highly appreciate all the support, enthusiasm and feedback!
Thank you all!

I would like to invite you to read this thesis and experience the research that I have been conducting the past year. Hopefully you can understand all the steps that I have been taking and sometimes the difficulties that I faced but I hope that you can also feel the enthusiasm that I got to develop even more for this subject. If you have any questions or other insights, you are always welcome to contact me.

Amy Hendriks

Summary

A high demand on buildings is nothing new in the building industry. However, with a shortage of 200.000 craftsmen in 2020 in Europe (van Koert, 2021), this growing demand cannot be met and therefore the rules of traditional construction should change (Cao, 2019). Robotic manufacturing systems have a lot potential considering this shortage in physical labour and therefore, could be able to decrease the shortage of buildings in the future. However, robotic manufacturing is not yet used to its full potential in the building industry because there are too many uncertainties related to the manufacturing process and not enough research has been conducted at this moment. These uncertainties that could occur during a robotic manufacturing process vary from the surroundings, such as wind or uneven soil, to material properties, such as part tolerances, to construction uncertainties, such as manufacturing tasks by people or by robots. This research investigates how to cope with these uncertainties instead of minimizing or eliminating them to be able to work with these uncertainties in the future.

A proof of concept framework is set up within Rhinoceros' Grashopper. This framework starts with the design of a tower of bricks which moves in two directions. This tower is structurally checked by a centre of mass analysis to see if this structure is stable to built or will collapse. When there are structural checks that fail the requirements, the structure can be optimized within the framework to retrieve a structure without failed structural checks which is as close to the initial design as possible. When a structure is designed that has no failed checks, the robotic arm can start the built of this structure by the placement of the first brick. This brick is measured just after it is placed. To find the correct measurement method, several experiments are conducted and measuring techniques are tested. The measuring technique that showed the highest potential was a combination of two cameras and the program Vision Builder (NationalInstruments, nd). This measuring techniques makes it possible to capture the geometry of the built structure and analyze it inside Vision Builder by means of added marks on the bricks. This analysis is translated to coordinates of the centre points of the bricks which are transferred to Grasshopper. From these centre points a digital twin of the built structure is generated to which the initial design can be compared. When a digital twin is established, the next bricks can be added on top of this digital twin to see if these bricks can be built further or if there are any failed checks during a second structural analysis. The framework will then adjust the next brick or generate a new structure for the bricks that are not yet built to fulfill all the structural requirements and keep on building the structure. When the last step is conducted, the framework will loop back to the part where the next brick is built by the robot. This new brick is then measured and analyzed again in the same way as the first brick.

This framework has proven to work during multiple tests. However, there are also some discussion points on the results of the framework. From the final tests it was concluded that there are differences measured between the initial digital design and the built equivalent of this design. These differences are mainly devoted to the used measuring technique and the angles of the cameras, the possible vibrations caused by the robot, mark tolerances and possible imperfect placement of the robot. However, it was clear from all the tests that it was possible to measure differences between the models. From these measurements, the framework itself noticed if changes to parts of the structure, that were not constructed yet, needed to be made. The measuring technique proved to be sufficient for this research but could be highly improved for future research considering the manual actions that were needed in the set-up of the measuring technique and the application to an actual building site. Also, there are several parts on which this framework could be expanded such as: the addition of different types of structures, which could require a change in the structural analysis method, the use of multiple robotic arms and possible boundary conditions within the design phase to take the robotic manufacturing into account early in the design stage.

However, by the means of this framework where the structure is designed, structurally analysed and measured during construction, it becomes possible to use the measured data and conclude if the manufacturing process can still continue according to the initial design or not. If there is a brick that is not placed correctly, the framework will notice this and give new coordinates for the next bricks to build further. This framework was automated so that the manufacturing process does not have to be stopped to run the framework and analyse the data. This framework takes into account the fact that geometric deviations due to geometric uncertainties can happen and finds a way to build further and not having to start over with the whole building process. In order to enlarge the application of robotic manufacturing within the built environment it is important to acknowledge possible uncertainties and find a way to cope with them. This framework is a first step into that direction where real-time measured data is the key.

Contents

Contents **ix**

List of Figures **xi**

List of Tables **xv**

1 Introduction **3**

2 Literature review **5**
- 2.1 Digital manufacturing systems . . . 5
- 2.2 Robotic manufacturing processes . . . 8
- 2.3 Geometric uncertainties . . . 11

3 Framework design **15**
- 3.1 Design of structure . . . 17
- 3.2 Structural analysis . . . 19
- 3.3 Translation to the robot . . . 23
- 3.4 Data collection . . . 25
  - 3.4.1 Test 1: Distance sensor . . . 26
  - 3.4.2 Test 2: Image processing with one camera . . . 29
  - 3.4.3 Test 3: Image processing with two cameras . . . 35
  - 3.4.4 Test 4: Image processing with two cameras, focus on Y-direction . . . 41
  - 3.4.5 Test 5: Image processing with two cameras, bigger bricks . . . 46
- 3.5 Data processing . . . 50
- 3.6 Data conclusion . . . 54
  - 3.6.1 Data conclusion based on one brick . . . 55
  - 3.6.2 Data conclusion based on remaining structure . . . 56
- 3.7 Automation . . . 57
  - 3.7.1 Starting point . . . 57
  - 3.7.2 Conducted steps to full automation . . . 58
  - 3.7.3 Result of automation . . . 62

4 Final tests and results **63**
- 4.1 Test set-up . . . 63
- 4.2 Data results . . . 67
- 4.3 Full framework with automation steps . . . 73

5 Conclusion **77**

6 Discussion **81**
6.1 Discussion final test results . . . 81
6.2 Discussion specific framework . . . 85
6.3 Discussion general framework . . . 89
6.4 Recommendations for further research . . . 91

Appendix **97**

A Grasshopper model **97**
A.1 **Design of structure** . . . 97
A.2 **Structural analysis** . . . 101
A.3 **Optimization possibility** . . . 106
A.4 **Data collection and data processing** . . . 108
A.5 **Data conclusion** . . . 113
A.5.1 **Data conclusion based on one brick** . . . 113
A.5.2 **Data conclusion based on remaining structure** . . . 116
A.6 **Translation to the robot** . . . 118

B Margin component calculation **126**

List of Figures

2.1 Robotic timber assembly by ETH Zürich (ETHZürich, 2018) 6
2.2 Distributed robotic assembly (Leder and Weber, 2018) 7
2.3 Overview of digital manufacturing systems in the building industry in Dutch (Wolf et al., 2022) 8
2.4 Examples of construction robots 9
2.5 Examples of construction robots 9
2.6 The Hadrian X (Caballar, 2019) 10
2.7 Calibration set-up of laser profile scanner on robot (Khawli et al., 2021) 13

3.1 Schematic view of the framework 16
3.2 Directions of displacements of the brick structures 17
3.3 Brick structures 18
3.4 Brick structures 18
3.5 Brick structures 18
3.6 Schematic diagram of the combined centre of mass (Soomro et al., 2020) 19
3.7 Structural analysis based on centre of masses 20
3.8 Structural analysis based on centre of masses 21
3.9 Structural analysis based on centre of masses 22
3.10 Output structural check 22
3.11 Optimization of structure 23
3.12 Robot set-up in Grasshopper 23
3.13 Robot set-up in Grasshopper 24
3.14 Laboratory set-up 24
3.15 First trial set-up distance sensor 26
3.16 Distance sensor test with robot 26
3.17 Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 1 28
3.18 Image processing test with robot 29
3.19 Image processing test 29
3.20 Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 2 31
3.21 Visualizations of the outcomes of test 1 and test 2 33
3.22 Image processing test with two cameras 35
3.23 Adjustment marks on bricks 36
3.24 Adjustment pick-up point 36
3.25 Brick structures with deviations in two directions 37
3.26 Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 3 38
3.27 Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,30 of test 3 39
3.28 Visualizations of the outcomes of test 2 and test 3 40
3.29 Test set-up for test 4, image processing with two cameras 41

3.30 Vision Builder views 41
3.31 Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,60 of test 4 43
3.32 Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 4 45
3.33 Test set-up of test 5 46
3.34 Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 5 47
3.35 Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,30 of test 5 49
3.36 Overview of steps in Vision Builder 50
3.37 Acquire image and image logging 50
3.38 Region of interest and vision assistant 51
3.39 Match pattern and calculator 51
3.40 Data logging and CSV file 52
3.41 Vision Builder analysis 52
3.42 Data from Vision builder to points 53
3.43 Data from Vision builder to structure 53
3.44 Steps of the data processing part of the framework 54
3.45 Steps of the data conclusion part of the framework 55
3.46 Optimization in data conclusion 56
3.47 Schematic view of automation steps 57
3.48 Pick-up point with slide 58
3.49 Pick-up point with slide 59
3.50 Additional Vision Builder program for automated picture 59
3.51 Additional Vision Builder program for automated picture 60
3.52 Calculator step of the additional Vision Builder program 60
3.53 Grasshopper elements to import the CSV file 61
3.54 Remote connection component 62
3.55 Digital model and digital twin 62

4.1 Total set-up of the final test 63
4.2 Set-up of cameras for the final test 64
4.3 Set-up of programs for the final test 64
4.4 Set-up of robotstudio for the final test 65
4.5 Needed wiring for the final test 65
4.6 Final test 66
4.7 Visualization of the digital (blue) and the built structure (red) with X-amplitude 1,50 68
4.8 Trend of the differences between the two models in the X-direction 69
4.9 Visualization of the digital (blue) and the built structure (red) of Y-amplitude 1,00 71
4.10 Trend of the differences between the two models in the Y-direction 72
4.11 Sine structure of test case 1 73
4.12 Optimization of test case 2 74
4.13 Leaning tower of test case 3 75
4.14 Leaning tower of test case 4 76

6.1 Tilted camera view 82
6.2 Grasshopper model normal camera view 82
6.3 Grasshopper model tilted camera view 83
6.4 Possible improvements for marks on bricks 83
6.5 Structure of a wall 85
6.6 Possible other measuring technique 87

A.1 Definition of the brick . . . 97
A.2 Visualization of the brick . . . 98
A.3 Movement X-direction . . . 98
A.4 Curve X-direction . . . 98
A.5 Movement Y-direction . . . 99
A.6 Curve Y-direction . . . 99
A.7 Vertical distance . . . 99
A.8 Structure selection . . . 100
A.9 Structures . . . 100
A.10 Additional information on designed structure . . . 101
A.11 Generating datatree for structural check . . . 102
A.12 Calculation of boundaries for structural check . . . 103
A.13 Visualisation of safe boundaries . . . 103
A.14 Calculation of centre of masses for structural check . . . 103
A.15 Outcome of structural check . . . 104
A.16 Visualization of failed structural checks . . . 104
A.17 Visualization of failed structural checks . . . 105
A.18 Visualization of failed structural checks . . . 105
A.19 Gene pool for Octopus optimization . . . 106
A.20 Octopus optimization . . . 106
A.21 View of the running Octopus optimization . . . 107
A.22 Result of the Octopus optimization (white is initial design, green is optimized) . . . 107
A.23 Live connection robot to Grasshopper . . . 108
A.24 Imported data from Excel . . . 109
A.25 Transfer of Excel data to coordinates . . . 111
A.26 Data from the built model . . . 111
A.27 Digital models . . . 112
A.28 Comparison between the two models . . . 112
A.29 Addition of one brick from the initially designed structure . . . 113
A.30 Visualization of the newly added brick . . . 113
A.31 Output of structural check . . . 114
A.32 Visualization of failed structural checks . . . 114
A.33 Visualization of failed structural checks . . . 115
A.34 Adjustment to new brick . . . 115
A.35 Visualization of adjustment to new brick (green is adjusted) . . . 116
A.36 Structural check of adjusted structure . . . 116
A.37 Optimization of remaining structure . . . 117
A.38 Dataflow optimization . . . 117
A.39 Robot target filter component . . . 118
A.40 Preparation of brick for robot . . . 118
A.41 Fabrication parameters . . . 119
A.42 Starting point of robot path . . . 119
A.43 Targets for bricks . . . 120
A.44 Targets for pick-up point . . . 120
A.45 Actions end-effector . . . 121
A.46 Composition robot path . . . 122
A.47 Tool definition: vacuum gripper . . . 122
A.48 Robot code and robot path . . . 123
A.49 Robot path with targets . . . 125

List of Tables

2.1 Possible contributors to geometric deviations from literature 12

3.1 Overview of conducted tests 25
3.2 Distance sensor and digital model data centre of masses in mm (X-direction) . . . 27
3.3 Comparison distance sensor data and digital model (X-direction) 27
3.4 Image processing and digital model data centre of masses in mm (X-direction) . . 30
3.5 Comparison image processing with one camera data and digital model (X-direction) 30
3.6 Distance sensor and image processing data centre of masses in mm (X-direction) . 32
3.7 Comparison distance sensor and image processing with one camera data (X-direction) 32
3.8 Overview of possible reasons for differences between digital and built model 34
3.9 Image processing and digital model data centre of masses in mm (X-direction) . . 37
3.10 Comparison image processing with two cameras data and digital model (X-direction) 38
3.11 Image processing and digital model data centre of masses in mm (Y-direction) . . 39
3.12 Comparison image processing with two cameras data and digital model (Y-direction) 39
3.13 Image processing with one and two cameras data centre of masses in mm (X-direction) 40
3.14 Comparison image processing with one and with two cameras (X-direction) 40
3.15 Image processing data centre of masses in mm (Y-direction) 42
3.16 Digital model data centre of masses in mm (Y-direction) 42
3.17 Comparison image processing two cameras and digital model in mm (Y-direction) 42
3.18 Comparison image processing two cameras and digital model in % (Y-direction) . . 43
3.19 Image processing data centre of masses in mm (X-direction) 44
3.20 Digital model data centre of masses in mm (X-direction) 44
3.21 Comparison image processing two cameras and digital model in mm (X-direction) 44
3.22 Comparison image processing two cameras and digital model in % (X-direction) . . 45
3.23 Image processing data centre of masses in mm (X-direction) 46
3.24 Digital model data centre of masses in mm (X-direction) 46
3.25 Comparison image processing two cameras and digital model in mm (X-direction) 47
3.26 Comparison image processing two cameras and digital model in % (X-direction) . . 47
3.27 Image processing data centre of masses in mm (Y-direction) 48
3.28 Digital model data centre of masses in mm (Y-direction) 48
3.29 Comparison image processing two cameras and digital model in mm (Y-direction) 48
3.30 Comparison image processing two cameras and digital model in % (Y-direction) . . 49

4.1 Final test data centre of masses in mm (X-direction) 67
4.2 Digital model data centre of masses in mm (X-direction) 67
4.3 Comparison final test and digital model data in mm (X-direction) 68
4.4 Comparison final test and digital model data in % (X-direction) 69
4.5 Final test data centre of masses in mm (Y-direction) 70
4.6 Digital model data centre of masses in mm (Y-direction) 70
4.7 Comparison final test and digital model data in mm (Y-direction) 70
4.8 Comparison between final test and digital model data in % (Y-direction) 71

B.1 Differences final test and digital model data in mm (X-direction) 126
B.2 Differences final test and digital model data in mm (Y-direction) 126

Chapter 1

Introduction

Since the world is still demands more houses and the physical labor possibilities are not growing with the increased demand, a change in the traditional building processes is inevitable (Lalor, 2021). With a shortage of 200.000 craftsmen in 2020 in Europe (van Koert, 2021) the growing demand cannot be met and therefore the rules of traditional construction should change (Cao, 2019). Next to that, the building industry is generating too much waste, which is estimated at 30% of all the building materials that is delivered to a building site, which is leading to the possible depletion of several materials (Osmani, 2011). A transition of the building industry towards an industry 4.0 (King, 2017) can make changes as well as improvements to these aforementioned problems. With this industry 4.0, which relies on communicating robots based on computer-aided data, higher levels of precision can be achieved which might lead to less waste production, less needed physical labor, and a higher building speed (King, 2017). A higher level of efficiency could be realized which also produces more to the growing specific needs of the costumers (Paritala et al., 2017). Within this industry 4.0, robotic manufacturing is being researched as one of the innovative manufacturing systems. Robotic manufacturing, specifically with a robot arm, is already applied to several building techniques, such as realizing masonry dry brickwork (Molloy, nd), timber frame elements (ETHZürich, 2018) and magnetic brickwork (Usmanov et al., 2017).

However, robotic manufacturing is not yet used to its full potential due to the lack of knowledge in general and on the uncertainties that might happen during the manufacturing process (Waibel, 2011). So, to achieve a closer step to the application of robotic manufacturing, research needs to be conducted on possible uncertainties that might happen during such a robotic manufacturing process and how to cope with these uncertainties to make these manufacturing processes more reliable and applicable in every situation. It is chosen in this research to find a way on how to cope with these uncertainties rather than minimizing and eliminating the uncertainties. The minimization and elimination of the uncertainties is a useful research for the building site as well but it is assumed for this research that these uncertainties can be minimized to a certain extent and not be entirely neglected. Therefore, a way on how to work with these uncertainties should be investigated. These uncertainties can be caused during all stages of the manufacturing process and by different elements such as the building environment, the building materials, and the design of the structural element. When robotic manufacturing can be applied on a bigger scale, the aforementioned problems, such as housing shortage and low amount of physical labor, can be reduced by the addition of this manufacturing technique.
The last sentence states the ultimate goal, but before this goal can be reached small steps need to be taken on how to cope with these uncertainties. Therefore the main research question of this research is as follows:

How can geometric uncertainties of a robotic building process be dealt with within the design and manufacturing process of a brick structure, to enlarge the application of robotic manufacturing within the built environment?

In addition to the main research question, 3 sub-questions complete the scope of this research:

1. *Which geometric uncertainties can occur during the robotic manufacturing process and therefore influence the structural behaviour of the brick structure?*

2. *How can the results of the geometric uncertainties of a robotic manufacturing process be measured?*

3. *How can adjustments be made to the robotic manufacturing process to reduce structural consequences due to geometric uncertainties?*

The third sub-question is quit complicated and therefore three additional smaller questions are part of this sub-question:

3a. How can the results of the measured geometric uncertainties be processed and compared to the initial digital model?

3b. To what extend will the structural behaviour of the end-product be influenced by the geometric uncertainties?

3c. What are possible adjustments to the structural element?

With the answers to these questions a step forward can be taken in the application of robots on the building site where the process can be much more controlled no matter which uncertainties are present at the time. This control over the robotic manufacturing process and the end-product makes this process much more reliable since every step of the structure will be measured. All of these answers will lead to a framework in which several parts of the building and manufacturing process are defined. The framework will be made in such a way that every individual part can be replaced with another version of the part. For example, one part of the framework is the calculation method, which calculates if the structure is stable and will stand after construction. This calculation method which could be based on a centre of mass analysis can be replaced by another calculation method which uses other data to make a calculation. This should be no problem for the rest of the framework since the others parts stay the same and can still be used. In this case it becomes possible to conduct more in-depth research on specific parts of the framework but still have a working framework with which the replacements or adjustments can be tested.

This report starts with a literature review (Chapter 2) on digital (Section 2.1) and robotic manufacturing systems (Section 2.2). This chapter also mentions possible uncertainties of these manufacturing processes in Section 2.3. Chapter 3 will go into detail on the framework that is made during this research including all the tests (Section 3.4) conducted in the TU/e Structures Laboratory (University of Technology Eindhoven) to search for a measuring technique and data to improve this framework. The next chapter (Chapter 4) describes the last conducted tests of this research and their results on which several conclusions can be drawn (Chapter 5) and the research questions stated in Chapter 1 can be answered. Lastly, all the discussion points of this research are raised in Chapter 6 together with some recommendations for further research.

Chapter 2

Literature review

In order to understand the problems and uncertainties of a robotic manufacturing process, various conducted studies are investigated in this chapter (Sections 2.1 and 2.2). Additionally possible geometric uncertainties are investigated to understand what type of tests and measuring techniques are needed for this research (Section 2.3).

2.1 Digital manufacturing systems

Robotic manufacturing is part of a broader field of different digital manufacturing systems. However, robot arms are the most prevalent (Molloy, nd). These robot arms could be used for stacking elements, drilling but also for 3D printing. Other digital manufacturing systems that could be used in the build environment consists for example of gantry robots and drones (flight assembled architecture (Molloy, nd)).
Quite some research has already been conducted and still going on on these digital manufacturing systems and their advantages.
A first research project by the ETH Zürich shows load-bearing timber modules that are prefabricated by gantry robots in combination with human builders. One of the big advantages of digital manufacturing becomes clearly visible during this project since multiple robots can work together on the actions that need to take place in a short amount of time. Where the first robot takes the timber beam and guides it while it is sawed to size, the second robot, after an automatic tool change, drills required holes for connecting the elements (ETHZürich, 2018). Lastly, the robots work together, via a developed algorithm that prevents the robots from colliding, to position the element on the exact required location. All these actions can be conducted faster and in a more precise manner than when humans would be responsible for every action. However, it is not the case that there is no human interference or help anymore. As shown in Figure 2.1 the actual bolting is done by humans. Even though there is still some human action involved at the moment of this research, this research shows already the power of digital manufacturing. Another advantage of a digitally manufacturing process shows when the design is slightly changed because the computational model can constantly be adjusted to meet the new requirements by some parameters in the model. This can than be directly communicated to the executing robots. This kind of integrated digital architecture is closing the gap between design, planning and execution (ETHZürich, 2018).
This research in robotic timber construction pursues a radical shift in scales of application where complex and efficient non-standard timber structures can be realised (Willmann et al., 2016).

Figure 2.1: Robotic timber assembly by ETH Zürich (ETHZürich, 2018)

There are also other advantages already explored or waiting for the industry in the near future. One of these being the robot's ability to process real-time sensor data, making it possible for the robot to know at every moment where it is or where something else is. The question then arises where such an advantage can be used to its best potential. An example of this is that a robot could be able to build a perfect wall and build further from where it was by measuring where it already placed something. Due to the high level of precision, a robot can for example rotate every brick in a wall with just one degree where a human could maybe not even see this difference or achieve this by hand, many more designs and structures become available to build in a short amount of time. However, the building industry is not yet ready for the robots to take over the whole design process, due to the human factors that come into play during such a process, but they can help the designers by building a physical model or gathering data of the site. We should see robots more as an opportunity, instead of a threat to also enlarge the possibilities with these robots (Simondetti, nd).
Another advantage of digital manufacturing is the fact that mass customization can be made more executable since this manufacturing is driven by information technology. Mass customization becomes more desirable compared to mass production since businesses wish to serve an individual customer and their individual desires with the shortest possible development time and production time. Digital manufacturing systems enable a company to conceive products in a desired style and quantity in time scales shorter than the conventional methods while efficiently managing the entire product life-cycle by their digital and material advancements (Paritala et al., 2017).
As already mentioned briefly in the example with the timber assembly by ETH Zürich (ETHZürich, 2018), robots can work together resulting in less time consuming fabrication. This is also shown in a project from the University of Stuttgart where a system was developed of multiple collaborative single axis robots (Leder and Weber, 2018) (Figure 2.2). When this system would be deployed on-site, one can imagine multiple robotic teams working quickly and in parallel to create structures with long spans or large heights that are reversible and through that adaptive to change. As such, a construction future is conceptualized where distributed robotics can build around the clock, higher, faster, stronger and quieter (Leder and Weber, 2018).

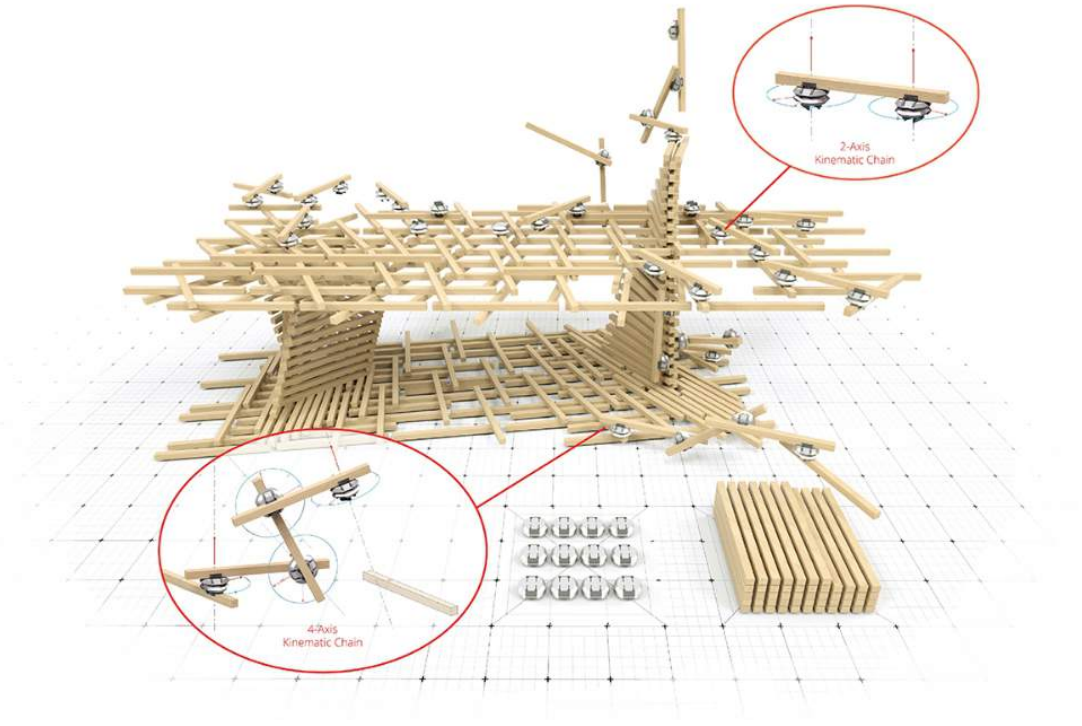


Figure 2.2: Distributed robotic assembly (Leder and Weber, 2018)

So, these multiple digital manufacturing systems have a lot of advantages that are already known or are becoming more evident during current research projects. The Dutch bank, ABN-AMRO, released a report in which they state that these manufacturing systems can make the building industry more sustainable, safer and lower the failure costs since there will be less failures on a building site (Wolf et al., 2022). The prediction that digital manufacturing systems will grow within the building industry between now and 2030 goes together with the crossing of certain current thresholds. These thresholds contain amongst others the lack in research and education, the conservative attitude and the lack in unambiguous data and software (Wolf et al., 2022). The report also makes a distinction between digital manufacturing systems as shown in Figure 2.3. All these systems are part of the digital manufacturing systems. Specifically, the construction robots will be investigated further in the upcoming section on robotic manufacturing processes (Section 2.2).

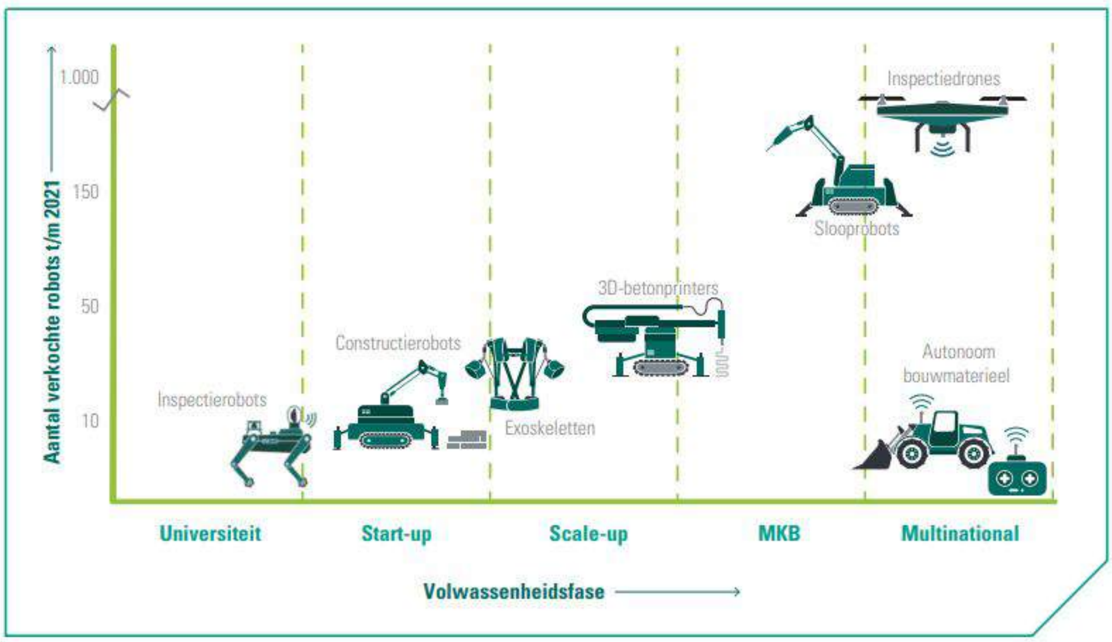


Figure 2.3: Overview of digital manufacturing systems in the building industry in Dutch (Wolf et al., 2022)

2.2 Robotic manufacturing processes

A more specific part of digital manufacturing is robotic manufacturing. This part of digital manufacturing will be used as the manufacturing process for this research, more specifically robotic manufacturing with a small robotic arm. In the report of Wolf et al. the following definition for this part of the digital manufacturing systems was given: robots that measure, lay bricks, glue, weld, drill or assemble (Wolf et al., 2022).

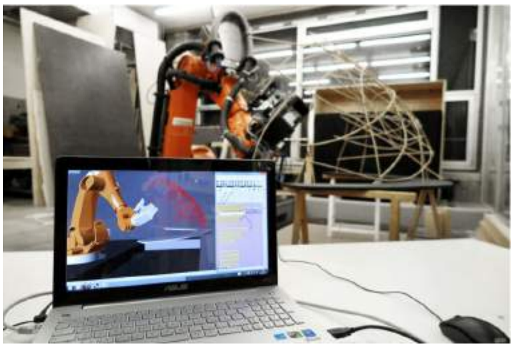
A first example of such a construction robot is given in Figure 2.4a where a robot carves into a wooden panel to make a timber sound diffuser. The fabrication of these sound diffusers are part of a larger research at the John H. Daniels Faculty of Architecture at the University of Toronto where the investigation of the potentials of robotic fabrication for architectural design and construction processes are the broader goal (Ergodomus, 2020). Next to using a single robot, multiple robots can work together to create an element as shown in Figure 2.4b. These two robots work together on a fiber reinforced building element that is created by the movements that both the robots make causing the fibre to wind in a certain pattern (Prado et al., 2014).

(a) The fabrication of a timber sound diffuser (Ergodomus, 2020)

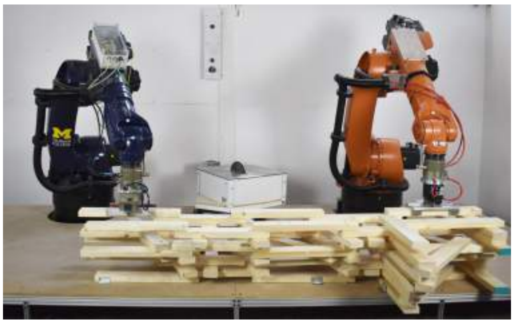
(b) Two robots working on a fibre reinforced building element (Prado et al., 2014)

Figure 2.4: Examples of construction robots

Other examples in Figure 2.5a and Figure 2.5b show assembly tasks conducted by robots. The first Figure (Figure 2.5a) shows a weaving process were a robot moves the strands in a certain pattern to create a structure. This means that the robot sometimes has to move a strand behind another strand and sometimes in front. The second Figure (Figure 2.5b) shows two robots working together on a timber structure. These elements are prefabricated by the robots in a separate space, then transported to the building site and on the building site assembled by humans. The robots can work together on the same assembly piece but on the ends of the piece making them not interfere with each other.



(a) The fabrication of a woven space structure (Brugnaro et al., 2016)



(b) Robotic timber assembly (Xie, nd)

Figure 2.5: Examples of construction robots

From multiple of these projects on robotic manufacturing a lot of advantages become clear. However, there are also multiple pitfalls still within these processes causing these robotic manufacturing processes to not being used to their full potential. One of these projects that mentions possible pitfalls developed the Hadrian X (Caballar, 2019), which is a block laying robot looking like a truck-mounted crane. This machine has several features that make it possible for the machine to work on its own such as: the identification of every block when they are loading into the machine, the ability to cut blocks into smaller pieces when needed and the possibility to store blocks for later use. Next to these features there is the layhead, where the block is kept at the precise location even with wind blowing and vibrations shaking the entire boom (Caballar, 2019). However, research is still conducted on this machine by bringing it into higher temperatures and other wind conditions to see if the machine is still capable of working. It is also not entirely clear if the machine processes the surroundings on the building site, such as the ground level that might not be completely straight.

Figure 2.6: The Hadrian X (Caballar, 2019)

Robots should not be seen as just fabricators or laborers but a full medium involved in the design and construction process. It is expected that robot arms will become sophisticated enough to work in tight and complex spatial conditions, enabling them to work directly on the construction site (Molloy, nd). This is already visible at the Hadrian X project where the robot works on the construction site itself.
More advantages of robots are quite similar to the advantages of all the digital manufacturing systems such as the fact that robots build faster than humans and they could even work during night hours since they do not get fatigued (HMCArchitects, 2019). Also, less people are needed to work on the construction site making the operation leaner (HMCArchitects, 2019) but also lower in construction costs (Usmanov et al., 2017). The technology that is used with these robots is less prone to error leading to more consistency in shape, construction and overall quality (HMCArchitects, 2019) but also to extreme precision and the ability to repeat actions frequently (Waibel, 2011). Lastly, robots can perform dangerous construction tasks (like demolition or complex crane work), which lessens the risk of injury to human workers (HMCArchitects, 2019).

As mentioned before, robotic manufacturing has a lot of advantages, but they are not used to its full potential due to the lack of research and trust that is in them. In that sense the building industry can learn a lot from other industries, for example the car industry, where robotics are already applied in big numbers and used to their strengths. At this moment there are already car manufacturers where artificial intelligence is combined with manufacturing lines to get even more speed in the manufacturing process, such as with Ford (Knight, 2021). Next to the lack of research and the trust in the robots, the disadvantages and possible uncertainties are apparently big enough to keep companies from using robotic manufacturing on a large scale in the past. These disadvantages are specified as the manipulation space, the ability to adjust and the purchase price (Usmanov et al., 2017). Next to these issues one of the biggest limitation of robotics in architecture and construction at this moment is that it lacks a human touch, primarily when it comes to creativity (HMCArchitects, 2019).

2.3 Geometric uncertainties

As mentioned in the last part of the previous section (Section 2.2) more research is needed into the possible geometric uncertainties that could occur during a robotic manufacturing process. It is stated that because of the difficulty of implementing various and dynamic work activities into robotic systems, the current construction industry drops behind in using robotic systems compared to other industrial branches (Usmanov et al., 2017). According to numbers of the Dutch Authority of Statistics, CBS, only 3% of companies in the building industry in the Netherlands made use of robots in 2018 (Cobouw, 2019). In 2020 55% of the building companies had at least one robot within their company (van Koert, 2021). However, that does not mean that these companies already make use of these robots and work with them on a regular basis. Nevertheless, it does show that companies are eager to invest in robots in the future as 81% of the researched companies in the building industry are willing to invest in robotics in the upcoming 10 years (van Koert, 2021). The reasons behind this high number of companies that want to invest in robots are mainly the safety aspects and the sustainability aspects together with the fact that the shortage of craftsmen is increasing (Industrievandaag, 2021). Other research pointed out that the biggest challenge is making sure the robot can handle construction tolerances and variations, adapting to changing conditions autonomously (Waibel, 2011). However, the question arises than what these construction tolerances and variations could be. If it is known what these construction tolerances and variations possibly are, then there could be thought of ways on how to cope with these tolerances and variations. ROB technologies stated that there are a lot of possible uncertainties. These possible uncertainties vary from part tolerances to uncertain environments to the necessity for complex jigs (ROBTechnologies, nd). They also state that work piece tolerances can be the consequence of manufacturing tasks by the robot itself or for example human workers (ROBTechnologies, nd). According to their research, which is still going on, the future is in the integration of sensor feedback.

Next to that it is not as easy as it may seem to use robots in the building industry. Companies should first look at their processes before adding robots to the equation (van Koert, 2021). By, for example, using a modular design and thinking about a robotic manufacturing process early in the development stage of a design, these factors can be taken into account. Also a building site is at this moment not a likely place for robots since it is an unorganised environment that can be highly unpredictable at moments (van Koert, 2021). This however, is also the reason for a high number of incidents on these building sites which could partly be prevented by robots that communicate with each other. It is also not unimaginable that robots will be present on a building site but they should be incorporated in a good and clear way into the whole building process (van Koert, 2021).

From these projects possible uncertainties of a robotic manufacturing process that could lead to geometric deviations are summarized in Table 2.1. Geometric deviations are the deviations that are present between what is designed to be build and what is actually build. With these uncertainties possible consequences are listed to get an idea of what could happen due to these uncertainties.

Insecurity	**Possible consequence**
Leveling of the ground (ROB-Technologies, nd)	If the ground is not even, the structure might not be even as well and might collapse earlier than expected
Vibrations (Caballar, 2019)	If the ground or surroundings vibrates the structure could move during construction
Wind (Caballar, 2019)	Due to wind the elements of the structure could start to move
Precipitation (ROBTechnologies, nd)	Due to precipitation the soil of the building site could become unstable
Manufacturing tasks by people (ROBTechnologies, nd), (Waibel, 2011)	People could enter the manufacturing process and adjust things which could lead to malfunctioning of the robotic system
Part tolerances (ROBTechnologies, nd), (Caballar, 2019)	If there are imperfections in the used material the structure could become unstable or positioned in the wrong place.
Manufacturing tasks robot (ROBTechnologies, nd)	There could be multiple uncertainties on the robots side for example the robot could be not installed properly, the robot could release a part too early or too late or the robot could grad the part in an unstable way
Deviation robot end-effector	The robot end-effector should work properly with the material otherwise the positioning of the elements could be not precise enough
Impact of placement of elements	If a new element is released onto a standing element it could be released with a small gravitational force which could lead to movement of other elements

Table 2.1: Possible contributors to geometric deviations from literature

The possible consequences stated in Table 2.1 are translated to a set of general consequences of what could happen to the structure due to the mentioned uncertainties:

- Wrongly placed element in terms of coordinates
- Wrongly placed element in terms of rotation
- Movement to the already placed elements
- Continuous movement of the structure
- Unstable connection between elements

The general consequences listed above to the uncertainties could occur separately but also combined with each other. This means that an element can only be placed wrongly in terms of coordinates for example but it could also be the case that in the same time there is continuous movement of the structure. The combination of the general consequences depends highly on the present uncertainties and this combination could also be between more than 2 general consequences.
A possible approach to cope with these uncertainties that might occur during the robotic manufacturing process is minimizing and eliminating the uncertainties and with that the consequences of these uncertainties. However, the question then arises if this is completely possible: can the uncertainties be completely eliminated? Or how much can the uncertainties be minimized? These questions are not in the scope of this project for the following reason: even if some of the uncertainties can be eliminated or minimized, it is no security that it can be done or what the influence of the insecurity will be if the insecurity can be minimized to a certain extend. It is still desirable that the uncertainties are eliminated or minimized to get an as clear as possible manufacturing process but that requires another research set-up. Therefore, within this research, it is accepted that uncertainties might be present during a robotic manufacturing process and for that assumption a framework is created to still work with a robotic manufacturing process even

though there might be uncertainties of some sorts to come as close to the reality of a building site as possible. This means that a combination with some sort of sensor and the robot should be made somewhere within the framework to get an idea of the geometric deviations caused by the geometric uncertainties.

There are already some studies conducted on the combination of robotics and sensors or feedback systems. One of these studies focuses on a subtractive manufacturing process of a timber element. To actually conduct the tasks, information was collected on the position of the robot during the carving operation to make the tasks of the robot as good as the same tasks conducted by a professional human worker (Brugnaro and Hanna, 2017). This research does not take into account environmental and geometric uncertainties but does collect data during the robotic tasks already. Also in other industries still research is conducted on the combination of robotics and sensor data, also in the aerospace industry. A recent study has pointed out that the absolute robot positioning is of high importance and that this accuracy could be effected by errors of the robot model, the tool calibration, the sensor and the product uncertainties (Posada et al., 2016). One last study shows the combination of a laser profile scanner in combination with a robotic arm. In this research the profile laser scanner was calibrated while connected to the robot to use this profile laser scanner from this position (Khawli et al., 2021). This research is conducted because it is stated that robotic manufacturing is useful in several manufacturing processes where manual labour intensive work is required (Khawli et al., 2021) such as the building industry.

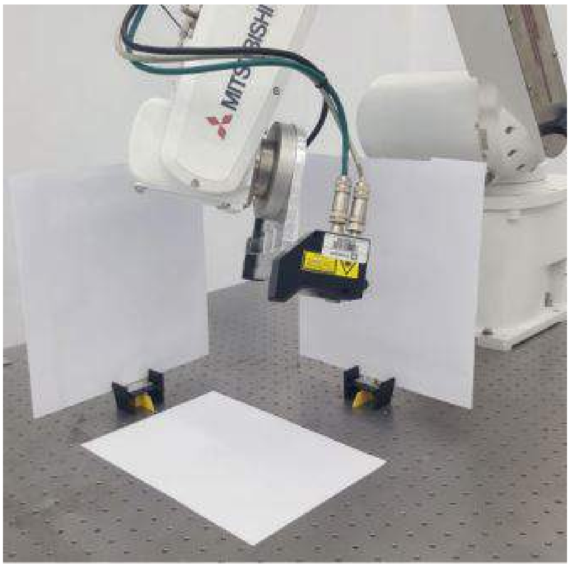


Figure 2.7: Calibration set-up of laser profile scanner on robot (Khawli et al., 2021)

It is however, not investigated how this scanner can be used in a construction set-up or how it can be applied. This is the where this research can be placed in the field of other studies since this research will look at the application of a measuring technique to a manufacturing process, how this collected data can be used and what can be concluded from it. It is basically a first step into the application of robotics on a building site.

Chapter 3

Framework design

To enable the application of robotic arms on the building site, a framework is designed in this research to work with the uncertainties of a robotic manufacturing process. This framework will measure the geometric deviations of a brick structure that is built by a robotic arm. First, an overview of this framework as a whole is given. Second, all the separate steps of the framework are explained in more detail including several tests that were conducted during the research to define the steps. Lastly, some additional information is given on the steps that are made to automate the framework. In the next chapter the framework will be used in the final tests to gain the results of the total framework.

The overall framework is designed from six different steps and visualized in Figure 3.1. These six steps represent the different steps of the robotic fabrication process from design to end result.

1. *Design of structure*
   The first step is creating the initial design of the structure (Section 3.1). Within this research, the initial design will consist of small bricks in a stacked design. However, this does not necessarily have to be the case and the designer of the structure should make a design in this stage of the framework what he/she wants to design and build with the robotic arm.

2. *Initial structural check*
   The second step in the framework is the initial structural check (Section 3.2). Within this check, the designed structure is checked to see if the structure will not collapse and therefore be save to build.

3. *Robot build*
   When the structural check suffices, the structure can be build with the robot (Section 3.3). To build the structure with the robot, the design is translated to robot code which will be read by the robot and used to execute the build.

4. *Data collection*
   During the construction of the structure, the structure will be measured in the data collection step (Section 3.4). This data collection step makes use of a measuring technique to gather the data of the placed bricks. Within Section 3.4 several tests are explained which have led to a conclusion on which measuring technique will be used for this research.

5. *Data processing*
   After the placement of every brick, this gathered data will be used and compared to the initial digital model in the data processing step (Section 3.5).

6. *Data conclusion*
   From this comparison between the build and digital model a conclusion can be drawn in the data conclusion step (Section 3.6). This data conclusion step contains multiple ways to draw a conclusion based on just one brick that is next to be placed or the whole remaining structure. This conclusion tells the robot if the next brick can be placed according to the initial design or if a change in coordinates is needed to the brick that will be placed. When the coordinates for the next brick are known, whether these are new coordinates or the coordinates of the initial design, the framework will loop back to the third step where the next brick will be placed with the robotic arm.

When a new brick is placed the framework will repeat itself from step 3 onward. The next brick will then again be placed, measured, compared to the digital model and a conclusion will be drawn for the next brick. This loop will continue until the whole design is build. By using this framework and measuring the build structure during the fabrication, the explained uncertainties can be taken into account and a structure can be build in one go.

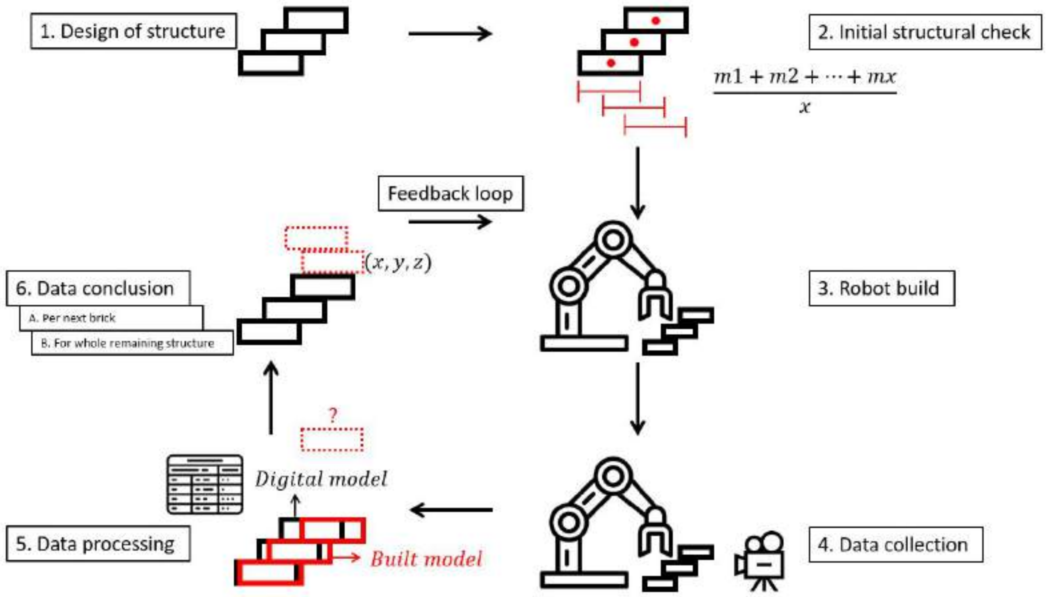


Figure 3.1: Schematic view of the framework

The framework is build for the largest part is Rhinoceros' Grasshopper. Grasshopper is a visual programming language in which it is possible to design structure, make calculations, upload data and use Python codes for example. For some smaller parts of the framework Python, Excel and Vision Builder are used as an addition to Grasshopper.

3.1 Design of structure

The first element of the framework consists of the set-up of a design for a structure that will be built with a robotic arm. For this research a case study of brick structures is chosen to work with, therefore the design in this research will consists of bricks. However, for the broader application of this framework this is no obligation since there are many more possible elements that could be used by a robotic arm to build a structure.
As for now, the small bricks are used as the basis of the design. These small bricks were chosen since many facilities for these small bricks are available in the TU/e Structures Laboratory so the experimental research with these small bricks could start early in the process of this research. To test multiple measuring techniques, which will be explored in Section 3.4, small structures of three bricks each are taken as the first structures. The structures are based on a sine curve of which the amplitudes in two directions can be altered. The amplitudes of the structures vary to generate 6 different structures with just three bricks. The structures only move, for now, to one side making the structures rather simple for this first measurement test. The digital models of the structures are shown in Figures 3.3a, 3.3b, 3.4a, 3.4b, 3.5a and 3.5b. Later during the different tests of the research the structures will become less simple in terms of movement to the other direction as well as becoming higher by the addition of more bricks. This is done to validate that the framework also works for other structures than the ones shown in this section.
To be consistent in the terms used for the displacements of the structures, the directions are defined beforehand. From Figure 3.2 the directions can be seen. The X-direction is defined as the direction of the long side of the brick and the Y-direction is defined as the direction of the short side of the brick. This means that if a displacement in the X-direction is mentioned that the structure of bricks move in the direction of the long side of the brick. The Z-direction is not mentioned here, this is defined as the height, which is not used in the conducted tests.

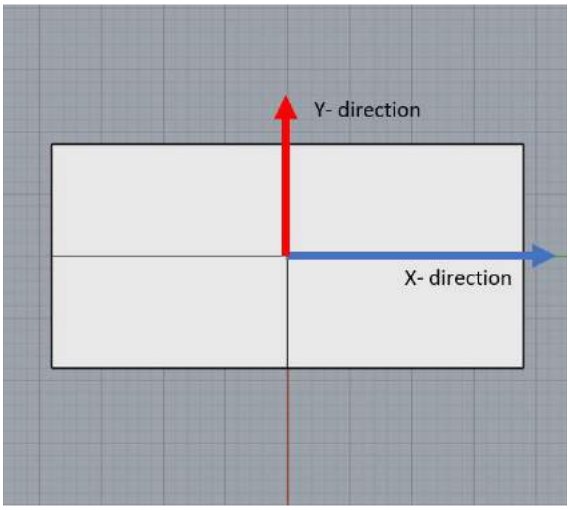
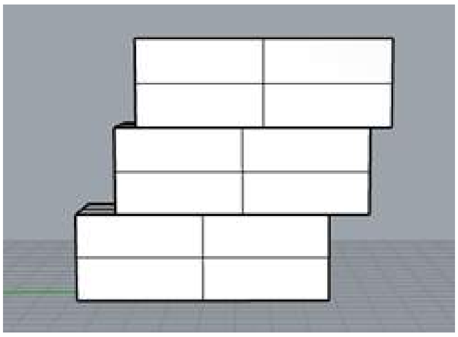
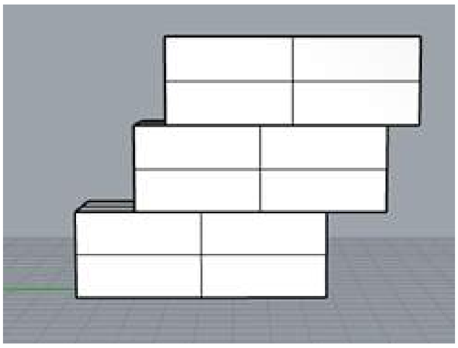


Figure 3.2: Directions of displacements of the brick structures

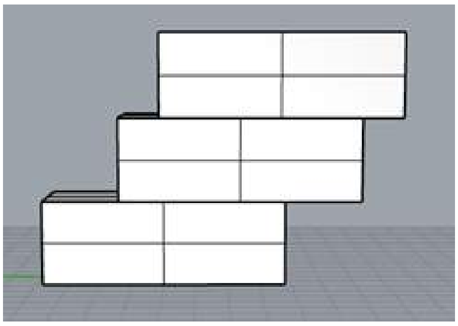


(a) Structure with X-amplitude 0,50

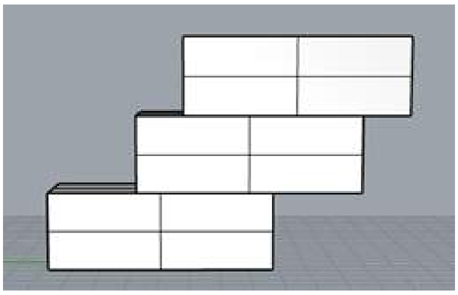


(b) Structure with X-amplitude 0,75

Figure 3.3: Brick structures

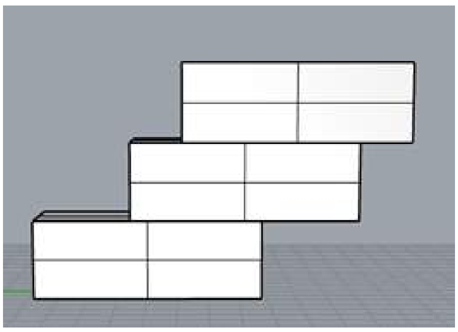


(a) Structure with X-amplitude 1,00

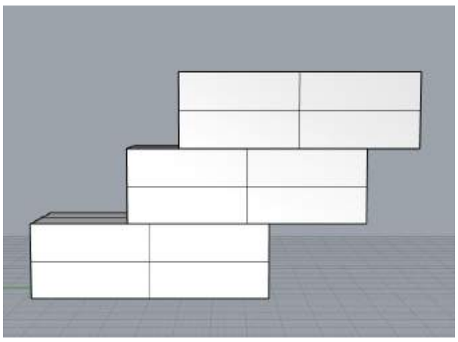


(b) Structure with X-amplitude 1,25

Figure 3.4: Brick structures



(a) Structure with X-amplitude 1,35



(b) Structure with X-amplitude 1,40

Figure 3.5: Brick structures

3.2 Structural analysis

From one of the sub-research questions in Section 1 the search and need of a structural analysis method is expressed when the influence of the geometric uncertainties on the behaviour of the brick structure is introduced. This influence by the geometric uncertainties needs to be calculated or visualized in a manner to see what actually happens due to the geometric uncertainties. Therefore, a structural analysis method is needed to draw conclusions on this part.
The structural analysis that will be used as a point of departure for this research is an analysis based on centre of masses. This problem is known in mathematical literature as 'the leaning tower of lire' (Soomro et al., 2020). This structural principle assumes that the mass of an object can be taken to be concentrated at one point (Soomro et al., 2020). The centre of mass can be evaluated per bricks but also of the structure as a whole leading to the following formula where n is the number of objects:

$$x_c = \frac{\sum_{i=1}^{n} m_i x_i}{\sum_{i=1}^{n} m_i} \tag{3.1}$$

x gives the centre of the mass of the n objects, m gives the mass and x gives the position of the i 'th object. These expressions can also be written for y and z if needed.
This means that when bricks are added to the structure the combined global centre of mass changes. An example of this is given in Figure 3.6.

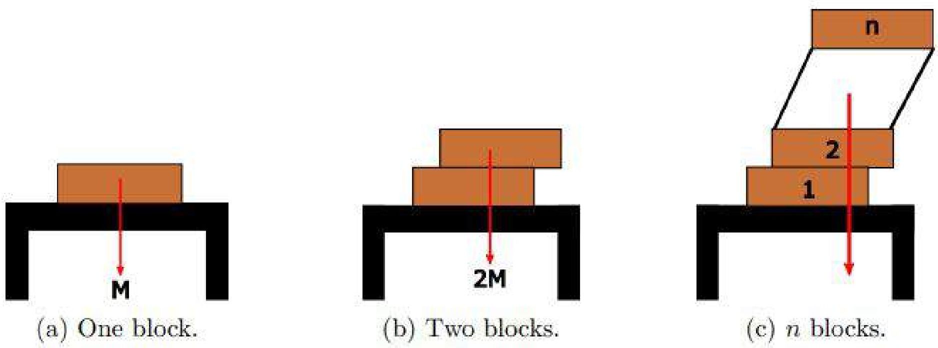
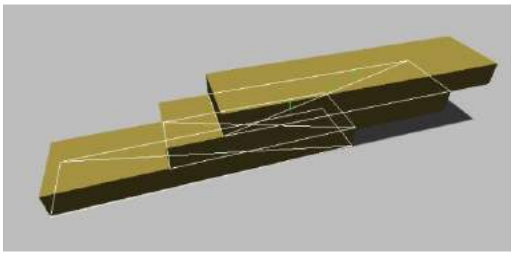
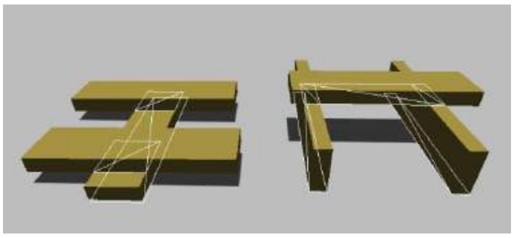


Figure 3.6: Schematic diagram of the combined centre of mass (Soomro et al., 2020)

When the bricks are constantly stacked on top of each other creating a tower, the analysis looks at the single object stability or single branch stacking (Figure 3.7a). This analysis will be the departure point for this research. When certain bricks in the structure would not lean on only one other element but multiple the analysis looks at multi-branch stacking (Thomsen and Kraus, 2014).



(a) Example of single branch stacking (Thomsen and Kraus, 2014)

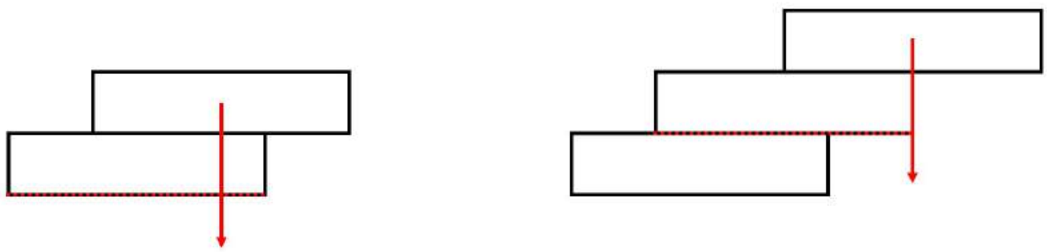


(b) Example of multi branch stacking (Thomsen and Kraus, 2014)

Figure 3.7: Structural analysis based on centre of masses

With a multi branch stacking analysis also other phenomenon like torque, the centre of contact area and an equivalent mass have a role. However, since the structures that are tested during this research are only towers stacked directly on top of each other on a single brick, this type of analysis is not needed at this moment. It is good to consider when the structures would change that the structural analysis might need a change as well. The choice is made for this research to focus on a rather simple structural analysis that gives already a lot of information on the structural behaviour of the towers of bricks that will be tested.

The structural analysis is the very first part of the framework after the design of the structure that is conducted and, as mentioned, based on an analysis of centre of masses. In Figures 3.8a and 3.8b a first example of such a structural analysis is shown. In both the figures the top of the red arrow represents the centre of mass of the top bricks. This centre of mass should be within the ground surface of the brick underneath it, indicated with the red dotted line in both cases, meaning that the structure is stable and will not collapse otherwise it is a failed check. These two checks are both individual checks meaning that the centre of mass of one brick is tested to the ground surface of the brick underneath.



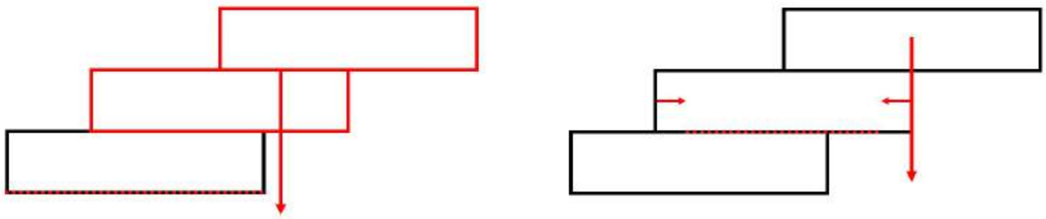
(a) Centre of mass analysis brick two

(b) Centre of mass analysis brick three

Figure 3.8: Structural analysis based on centre of masses

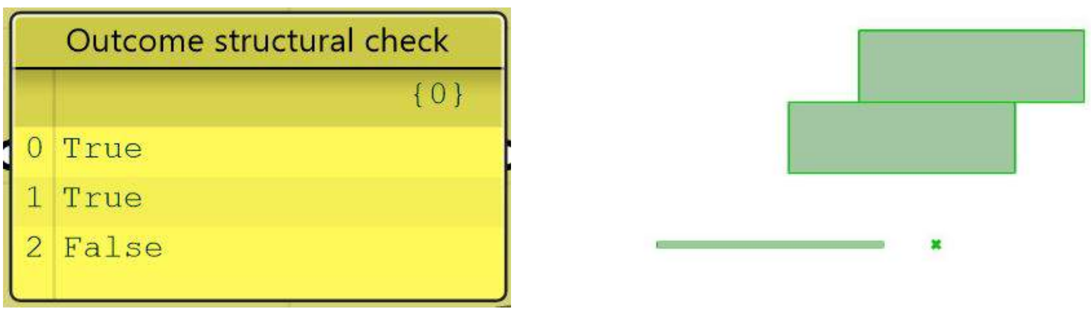
When a structure is higher than two bricks, so three bricks or more, also combined centre of masses should be taken into account. An example of a combined centre of mass is visible in Figure 3.9a, where a combined centre of mass is calculated for the top two bricks. This combined centre of mass is then checked to the ground surface of the lowest brick (red dotted line). It is also visible that this is a failed check since the arrow is outside the ground surface. The structure will therefore collapse due to the combination of the second and the third brick. Grasshopper also gives this as an output: the amount of failed checks and where they are in the structure (Figures 3.10a and 3.10b). These combined centre of mass checks are made for every possible combination of bricks in a structure. Especially when the structures become higher there are many more combined centre of masses and therefore structural checks that need to be conducted.
When it occurs that one or multiple structural checks fail, Grasshopper shows which checks these are and what the distance is that the individual or combined centre of mass is to the surface boundaries. An example is given in Figure 3.10b. Since one of the structural checks failed for this structure, Grasshopper gives one set of figures as an output. This figure shows that, in this case, the combined centre of mass from the top two bricks (shown on the 'ground') does not fall within the surface boundaries of the lowest brick, visualised in a small rectangle on the 'ground'. Grasshopper gives the horizontal distance between the combined centre of mass and the surface boundaries in a text form, which in this case is 8,98 mm.

Margin component
It could be the case that the structural check is just on the edge of a failed outcome, for example in Figure 3.8b where the individual check of the top brick is just within the ground surface of the brick underneath it. For safety reasons, since it is not entirely sure that this brick will be placed on the exact expected location, a margin component is added that scales the ground surface just a bit small which is visible in Figure 3.9b.
This margin component is added since from the first tests it became visible that building on the edge of the exact safe surface boundaries could cause the structure to collapse since there are some differences to the digital model. The very first value of this margin component was based on a test case with a structure with an X-amplitude of 1,35. This case was just safe to build according to the structural analysis (there were no failed checks) but in practice failed every time it was build. Therefore, the structure was build with several margin component values until the structures kept standing. The first version of the margin component was, as a result of these tests, set at 98% of the original surface boundaries.
When the surface boundaries are scaled with the margin component, the model can check if the centre of masses are within these safe surface boundaries. The model gives a boolean statement outcome of this check meaning true or false. If all the statements are true, then the structure should not collapse when being build.



(a) Combined centre of mass analysis (b) Explanation of margin component

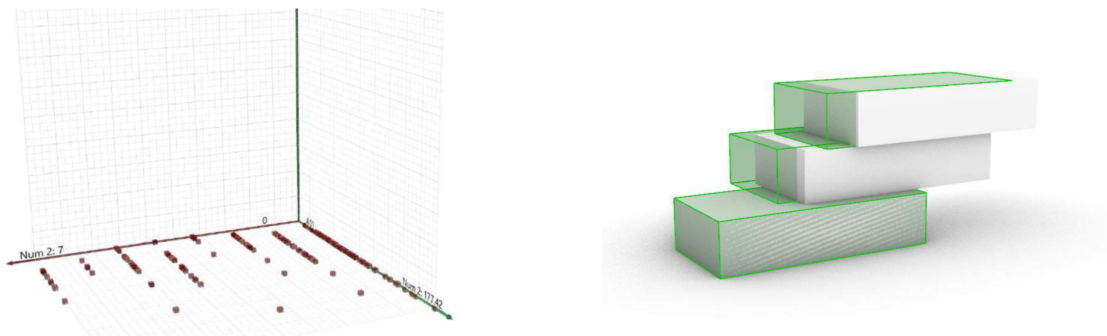
Figure 3.9: Structural analysis based on centre of masses



(a) Output of structural check (b) Visual output of the structural check that fails

Figure 3.10: Output structural check

Two things can be done if there are any failed structural checks. The first option is rather simple, meaning, that the original design can be adjusted or another design can be chosen. In this case the original design can be adjusted in Grasshopper and the framework shows then if there are any failed structural checks. There is, however, also a second option that makes it possible to use an optimization tool (the Grasshopper plug-in Octopus (Food4Rhino, nd)). This optimization tool searches for the structure that is as close as possible to the initial design without any failed checks. The optimization tool gives several outcomes in the form of small blocks visualized in a graph of which one should be selected manually(3.11a). In Figure 3.11b the white structure shows the initial design which had in this case one failed check. The green structure is the optimized structure that is as close as possible to the initial design but has no failed checks. This structure can then be used for the rest of the framework.



(a) The optimization tool from Octopus

(b) The initial structure (white) and the optimized structure (green)

Figure 3.11: Optimization of structure

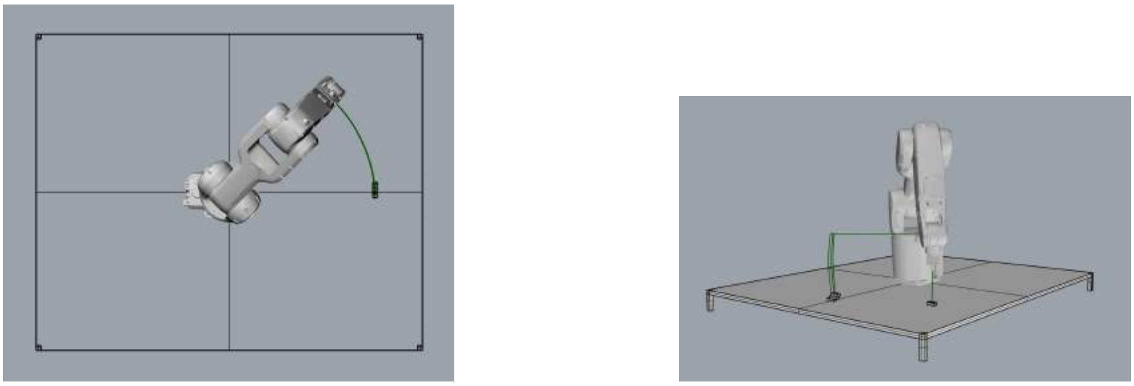
3.3 Translation to the robot

When the structural analysis, as described in Section 3.2, is conducted and all the checks suffice, the structure of small bricks can be build with the robotic arm. The robotic arm that is used during this research is an ABB IRB1200-5/0.9 robot (ABB, nd). Before the actual build is conducted with the robotic arm in the TU/e Structures Laboratory, a simulation is run inside Grasshopper to check the robotic path. To do this, a rapid code is generated by a separate part of the Grasshopper script where the design, described in Section 3.1, is used as a base. Within this part of the Grasshopper script also the robot end-effector is added to the script, which in this case is a vacuumtool. In Figures 3.12a, 3.12b, 3.13a and 3.13b the simulation of the robot set-up in Grasshopper is shown.

(a) Front view

(b) Side view

Figure 3.12: Robot set-up in Grasshopper



(a) Top view (b) 3D view

Figure 3.13: Robot set-up in Grasshopper

Besides a simulation in Grasshopper, also a simulation of the robot path can be run in Robotstudio, the program that is used to load the rapid code to the robotic arm itself. Both programs give the opportunity to see the path that the robotic arm will use before actually doing it. This makes it possible to filter out mistakes beforehand and check the whole robot path on insufficient or misplaced movements. After the simulation and the possible refinement of the robot path, the structure can be built with the robotic arm in the TU/e Structures Laboratory. A set-up is shown in Figure 3.14. To run this path the the robot arm, the base code and main code of the robot path need to be uploaded to robot studio. When this is done the robot can be moved by the control panel according to the lines of the codes.

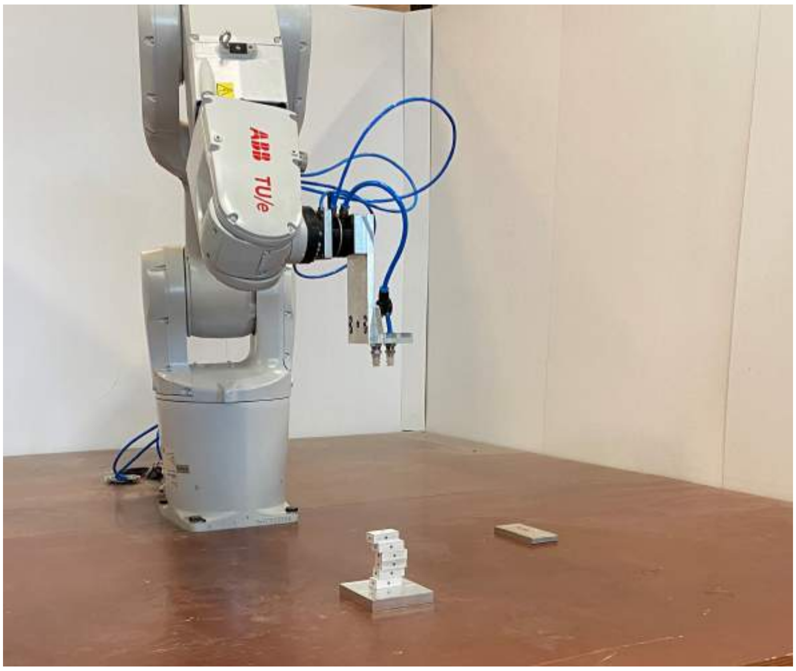


Figure 3.14: Laboratory set-up

3.4 Data collection

During the building of the structure by the robotic arm, data of the placed bricks should be collected. This was also formulated in the second sub-research question in Chapter 1, where a question specifically on measuring the geometric deviations caused by uncertainties of a robotic manufacturing process was introduced. However, it is not yet known beforehand what the best or even possible measuring techniques are to use for a robotic manufacturing process or on a building site. Therefore, several tests will be conducted to gain data of the build structures to improve the framework but first and foremost compare and test different measuring techniques. The tested equipment consists of a distance sensor but also multiple cameras. An overview of all the conducted tests is given in Table 3.1. Within this table, all the tests are numbered, the used measuring technique is mentioned and the focus points of the tests are stated.
It should be stated that there are many more possibilities in the field of measuring techniques to use for a robotic manufacturing process or on a building site. The measuring techniques that were tested during this research were already available in the TU/e Structures Laboratory and could therefore be used early in the process. This made it possible to do different tests and work with the outcomes of these tests. Besides being already available, both the measuring techniques from the TU/e Structures Laboratory work statically, meaning that the equipment is placed on one spot and is not moved during the build. However, this is not the case for all possible measuring techniques. As an extension, which will also be discussed in Chapter 6 to this first research other types of measuring techniques could be tested, such as a 3D hand scanner (Hartman, nd) or 2D/3D profile scanners (Micro-epsilon, nd). This profile scanner was investigated shortly which will be discussed in Chapter 6. One other option is the Microsoft Kinect (Microsoft, nd). The Kinect is a rather simple and inexpensive markerless motion capture sensor (Bilesan et al., 2018) but research was already conducted to show that the Kinect can also track markers. However, the Kinect was not directly available compared to the measuring techniques from the TU/e Structures Laboratory and therefore the available and known measuring techniques from the TU/e Structures Laboratory were taken as a starting point for the search to a measuring technique for this framework.

Test number	Measuring technique	Focus points
1	Distance sensor	3 bricks, only X-direction
2	Image processing with one camera	3 bricks, X-direction
3	Image processing with two cameras	3 bricks, X and Y-direction
4	Image processing with two cameras	6 bricks, Y-direction
5	Image processing with two cameras	6 bricks, X and Y-direction, bigger bricks

Table 3.1: Overview of conducted tests

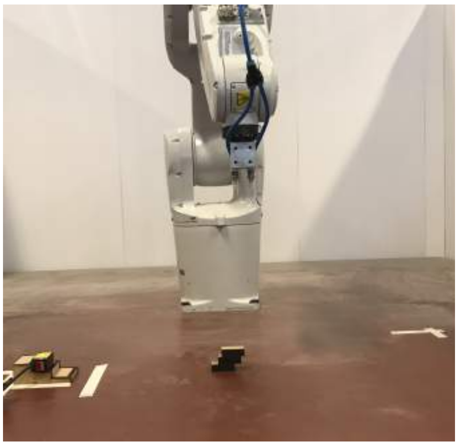
In order to test the measuring techniques as good as possible, the tests are conducted inside the TU/e Structures Laboratory and as much uncertainties as possible are eliminated. Examples of this are that the bricks are made with a lasercutter to minimize the imperfections of the material and the tests are conducted inside to minimize the effects of wind and vibrations of the ground. This keeps the focus on the measuring technique itself, when the best measuring technique is chosen, the uncertainties could be added to evaluate if the measuring technique still works under different circumstances within this research or further research.

3.4.1 Test 1: Distance sensor

The distance sensor test was set-up with a small step where the distance sensor (Panasonic micro laser distance sensor HG-C1400 (Panasonic, nd)) was laid on top (shown on the right side of Figure 3.15). This little step was built in such a way that there are three different levels in height which correspond to the heights of the bricks that will be placed on top of each other (Figure 3.15). During the built with the robot the distance sensor had to be moved higher onto the steps to measure the next placed brick, this movement was done by hand. The placement of the step was also measured by hand. This means that the distance sensor was set at a distance of twenty centimeters from the first placed brick. This distance was measured by hand and later measured with the distance sensor itself.

Figure 3.15: First trial set-up distance sensor

(a) Placement of first brick



(b) Total structure

Figure 3.16: Distance sensor test with robot

Table 3.2 shows the data (X-coordinate) of the centre of masses of the placed bricks that are measured with the distance sensor in millimeters on the left side of the table. On the right side of the table, the X-coordinates of the centre of masses of the bricks of the initial digital design are given. This latter data is from the digital model to which the measured data will be compared. Both this data from the distance sensor as well as the digital model will be used to make a comparison between the two in the next table.

	Distance sensor data			Digital model		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,33	7,17	10,97	0	5,52	8,46
0,75	0,33	6,90	10,37	0	8,29	12,70
1,00	0,03	11,20	16,87	0	11,05	16,93
1,25	-0,27	13,73	21,23	0	13,81	21,16
1,35	-0,33	15,40	-	0	14,92	22,85
1,40	-0,77	14,80	-	0	15,47	23,70

Table 3.2: Distance sensor and digital model data centre of masses in mm (X-direction)

Table 3.3 shows the differences between the data of the previous table in millimeters (left side) and percentages (right side). The percentages are normalized to the length of the brick according to the direction that they move. This means that the differences in the X-direction are normalized with the long side of the brick. A negative number in this table shows that the placed brick by the robot (the distance sensor data) is more to the right compared to the data from the initial digital model. A positive number shows the opposite, so the brick is placed more to the left than initially meant by the digital model. So, for example, the first brick of the structure with X-amplitude 0,5 was placed on the left side of the vertical axis when looking at Figure 3.17 with an amount of 0,33 mm. The third brick of the biggest two amplitudes does not have data in the table. These bricks could not be measured by the distance sensor since the structure collapsed with the placement of the third brick. All the structures that are measured with the distance sensor are built three times and measured three times. This means that the data in the tables is an average of three builds of the same structure.
Figure 3.17 is added to visualize the differences between the models. For this visualisation the structure with an X-amplitude of 0,5 was taken since this structure showed the biggest differences to the initial digital design. The blue structure represents the digital initial design and the red structure represents the built structure. The overlap in the figure is where the structures are in the same position. Where there is only blue or red it shows how much of a difference there is between the two models. Also for both structures the centre points are shown in the accompanying colour. It is now possible to actually see the differences between the two models and imagine how the other built structure look like compared to the digital model since there are also percentages of the other structures available in Table 3.3. The set-up of this figure is used for all the other comparison figures to come.

	Differences in mm			Differences in percentage (%)		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	0,33	-1,64	-2,50	0,88%	-4,32%	-6,59%
0,75	-0,33	1,39	2,33	-0,88%	3,65%	6,13%
1,00	-0,03	-0,15	0,06	-0,09%	-0,40%	0,16%
1,25	0,27	0,08	-0,07	0,70%	0,20%	-0,19%
1,35	0,33	-0,48	-	0,88%	-1,27%	-
1,40	0,77	0,67	-	2,02%	1,76%	-

Table 3.3: Comparison distance sensor data and digital model (X-direction)

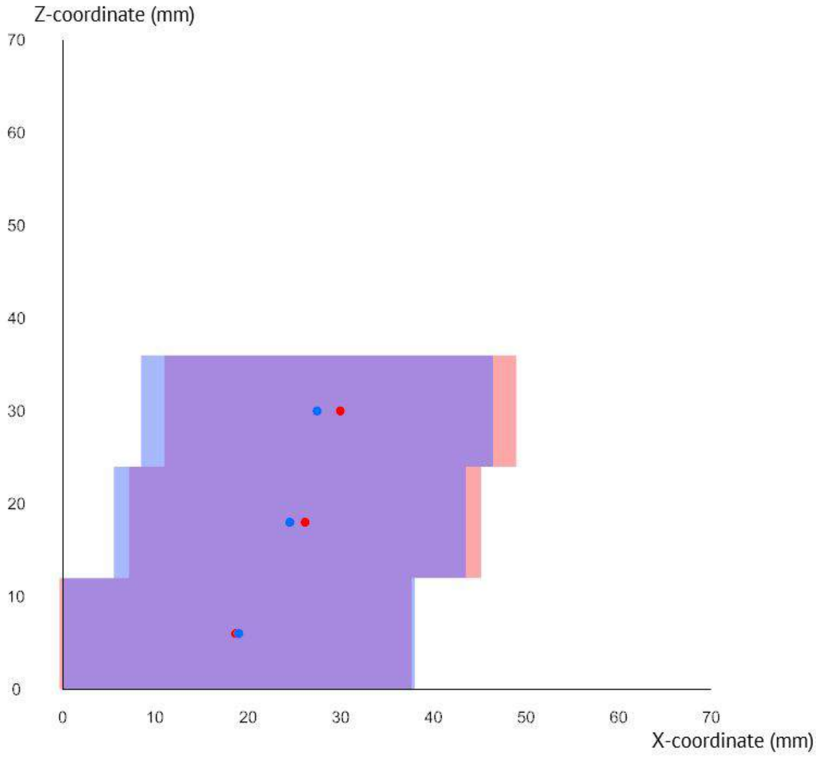
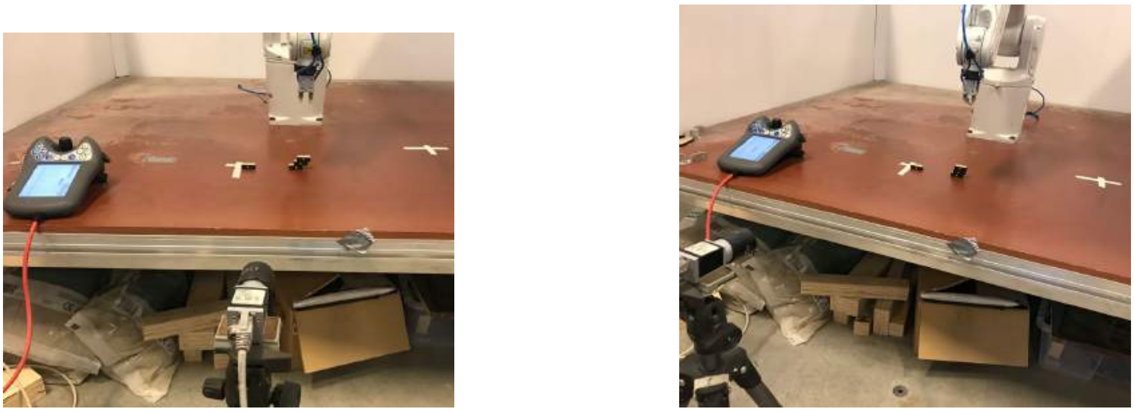


Figure 3.17: Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 1

From the data given in this section it can be concluded that the distance sensor is a good first measuring technique to get familiar with the idea of measuring a structure build with a robot. However, quite some actions needed to be done by hand causing some big differences between the measured data and the digital data. Since this measuring technique would also be quite hard to use with higher structures or structures with a completely different form, such as a wall, it is important to look at other measuring techniques as well.

3.4.2 Test 2: Image processing with one camera

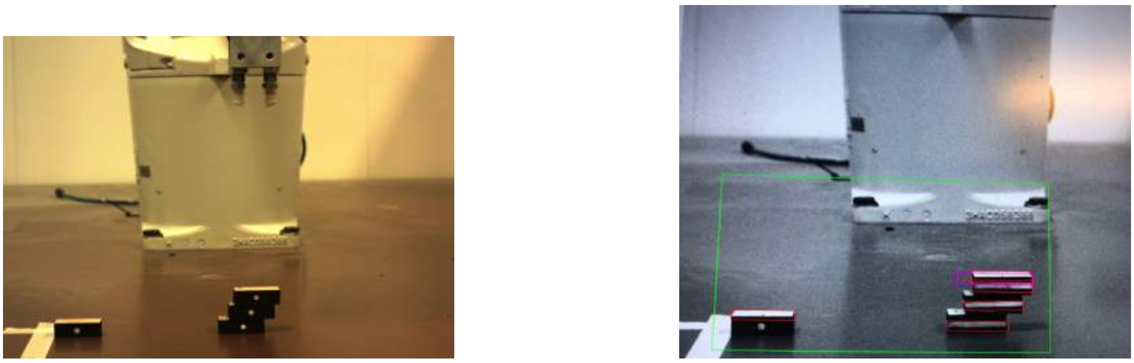
For the set-up of the first image processing test, one camera on a tripod was placed in front of the table where the bricks would be placed (Figures 3.18a and 3.18b). This camera, a Basler ace AC4600-7gc (Basler, nd), was placed once and did not need to be replaced after that, making the 'human errors' of replacing the sensor/camera already smaller compared to the distance sensor test. The camera made a live view of the bricks (Figure 3.19a) and when wanted the view could be translated to pixels which could be translated to measurements/coordinates of the introduced marks with the use of a calibration factor. This is all done in the program Vision Builder for automated inspection (NationalInstruments, nd). To capture the wanted coordinates of the centre of mass in this case, dots where placed in the middle of the bricks so that Vision Builder could recognize these dots and determine the exact middle point of the dots. This was done every time after the placement of one single brick which gives the coordinates of the just placed brick and the bricks that were already placed before.



(a) Test set-up image processing

(b) Test set-up image processing

Figure 3.18: Image processing test with robot



(a) Live view of camera

(b) Vision builder screen

Figure 3.19: Image processing test

During the tests with the image processing measuring technique, the same structures were built as with the distance sensor test explained in Section 3.4.1. The data of the X-coordinate of the centre of masses of the built structures and digital model is given in Table 3.4. The set-up of the table is the same as in the previous section. Again, all the structures are built three times so the stated numbers are an average of three builds. However, now the third brick of the two highest amplitudes do contain data. This could be achieved since the robot was put on hold just before the robot placed the third brick. By doing this, Vision Builder was able to take a picture with the third brick almost exactly at the location where the brick would be placed just before the structure collapsed due to the addition of this third brick to the structure.

	Image processing data			**Digital model**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	0,01	6,23	9,12	0	5,52	8,46
0,75	0,03	8,89	13,07	0	8,29	12,70
1,00	-0,01	11,57	17,19	0	11,05	16,93
1,25	-0,01	14,18	21,37	0	13,81	21,16
1,35	0	15,29	23,02	0	14,92	22,85
1,40	-0,02	15,87	23,91	0	15,47	23,70

Table 3.4: Image processing and digital model data centre of masses in mm (X-direction)

Table 3.5 shows the differences between the built model and the digital model in millimeters on the left of the table and in percentages on the right of the table. Figure 3.20 shows again the differences of the digital and the built model of the structure with X-amplitude 0,50.

	Differences in mm			**Differences in percentage (%)**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,01	-0,71	-0,65	-0,03%	-1,87%	-1,72%
0,75	-0,03	-0,61	-0,38	-0,08%	-1,59%	-0,99%
1,00	0,01	-0,53	-0,26	0,03%	-1,38%	0,-70%
1,25	0,01	-0,37	-0,21	0,03%	-0,98%	-0,56%
1,35	0	-0,38	-0,16	0,00%	-0,99%	-0,43%
1,40	0,02	-0,40	-0,21	0,05%	-1,06%	-0,56%

Table 3.5: Comparison image processing with one camera data and digital model (X-direction)

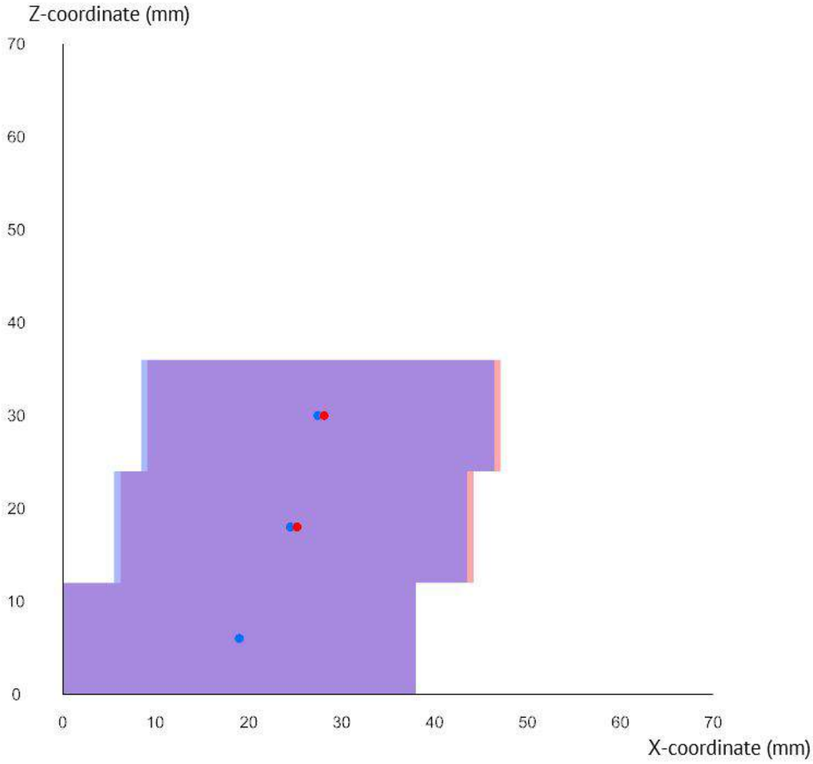


Figure 3.20: Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 2

During the test it was noticeable that less actions needed to be done by hand compared to the first test with the distance sensor (Section 3.4.1). The results of this can also be seen in Table 3.5 and Figure 3.20 since the differences between the built and digital model in millimeters are more consistent and smaller than the data from the distance sensor and the differences in percentages are also smaller. The first and second test, respectively with the distance sensor and image processing with one camera, are further compared in the next section (Section 3.4.2.1).

3.4.2.1 Comparison Test 1 and Test 2

From the data generated from the tests with both measuring techniques it can be concluded that the data obtained by the image processing technique is more consistent when looking at the sizing. Also, the data from the image processing measuring technique is closer to the data from the digital model. The image processing measuring technique is also more secure in the way that less human action is needed since the camera does not have to be moved where the distance sensor had to be moved when a new brick was placed. Next to that, a calibration factor is used within Vision builder. This calibration factor can be checked during a check test to see if this calibration factor is correct (by checking for example the length of the brick in Vision builder) whereas with the distance sensor the calibration was also done by hand.
One other advantage of the image processing test is that multiple cameras can be used together with a next test. This could also be done with the distance sensor but the biggest advantage of the image processing technique is that the cameras can both be connected to the same Vision Builder file which makes it possible to analyse multiple camera views at the same time.
Tables 3.6 and 3.7 give an overview of the data from the distance sensor test and the image processing test and the differences between the two. The data in Table 3.6 has no differences with the data that was presented in previous sections about the specific measuring techniques, this data is given here again for a complete overview of the comparison. The data in Table 3.7 however is new since the data of the distance sensor test and image processing test was not compared before. This table shows that there are quit some big differences between the outcomes of both tests. When looking at the data in Table 3.7, it is noticeable that there are some high numbers such as 2,71 millimetres (X-amplitude 0,75 brick 3) and 1,85 millimetres (X-amplitude 0,50 brick 3). This means that there are some big differences between the results of the measuring techniques which could be caused by the manual tasks that still needed to be conducted with the distance sensor test.

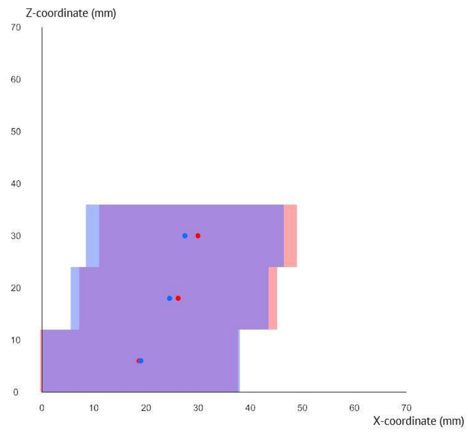
	Distance sensor data			**Image processing data**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,33	7,17	10,97	0,01	6,23	9,12
0,75	0,33	6,90	10,37	0,03	8,89	13,07
1,00	0,03	11,20	16,87	-0,01	11,57	17,19
1,25	-0,27	13,73	21,23	-0,01	14,18	21,37
1,35	-0,33	15,40	-	0	15,29	23,02
1,40	-0,77	14,80	-	-0,02	15,87	23,91

Table 3.6: Distance sensor and image processing data centre of masses in mm (X-direction)

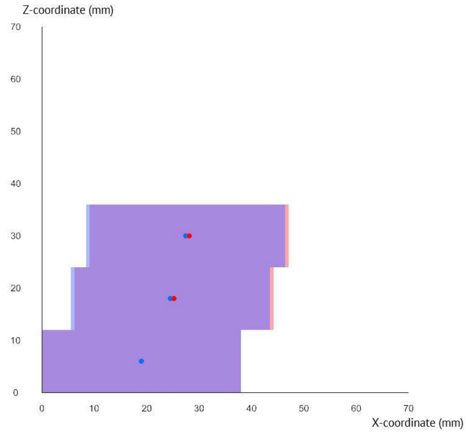
	Differences in mm			**Differences in percentage (%)**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,35	0,93	1,85	-0,91%	2,46%	4,87%
0,75	0,30	-1,99	-2,71	0,80%	-5,24%	-7,12%
1,00	0,04	-0,37	-0,33	-0,11%	-0,99%	-0,86%
1,25	-0,26	-0,45	-0,14	-0,67%	-1,18%	-0,37%
1,35	-0,33	0,11	-	-0,87%	0,28%	-
1,40	-0,75	-1,07	-	-1,97%	-2,82%	-

Table 3.7: Comparison distance sensor and image processing with one camera data (X-direction)

Figures 3.21a and 3.21b were already shown in the previous sections. However, when putting the two figures next to each other the differences between the two tests are clearly visible meaning that the distance sensor data has more differences with the digital initial model than the image processing data. Both the figures are again for a structure with an X-amplitude of 0,50.



(a) Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 1



(b) Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 2

Figure 3.21: Visualizations of the outcomes of test 1 and test 2

With the two tested measuring techniques came reasons for the differences between the built model and digital model. The possible contributors to these particular differences between the models are listed in Table 3.8 together with a short explanation and to which type of test they are applicable. This overview shows which contributor belongs to which type of test but also which contributors are applicable for both tests and can be improved anyway.

Reason	**Type of test**	**Explanation**
Pick-up point	DS and IP	The pick-up point is marked by hand every time a new test is conducted. This could lead to an actual pick-up that is not entirely the same as the digital one leading to deviations in the final positions of the structural elements. The pick-up point can therefore also vary per brick.
Marks on elements	IP	The marks on the elements for the Image Processing test are made by hand. Since the centre points of the marks are measured with the image processing technique, deviations could occur if the marks are not placed exactly in the middle of the element.
Gravitational acceleration	DS and IP	The placement of one brick could lead to a gravitational acceleration of that brick if the brick is released with a height to the bricks underneath it.
Movement of sensor	DS	Since the distance sensor could only measure the distance over one and the same height, the distance sensor was moved by hand to get to the next height to be able to measure the next element. This was done in combination with a holder made to the heights of the elements. Both the movement by hand as well as the holder could be an insecurity being it was all done or made by hand.
Robot end-effector	DS and IP	The robot end-effector, in the tests a vacuumtool, could be unstable which leads to small deviations from the planned coordinates. Since the vacuumtool 'sucks' the element in the air this could lead to the element being slightly out of place once connected to the vacuumtool
Calibration factor	IP	In order to calculate the coordinates of the elements with the image processing technique, a calibration factor is used to calculate pixels to millimeter. This calibration factor is based on the measurement of one of the elements which is measured with a digital caliper, if this is not entirely correct, all the coordinates could be slightly incorrect

Table 3.8: Overview of possible reasons for differences between digital and built model

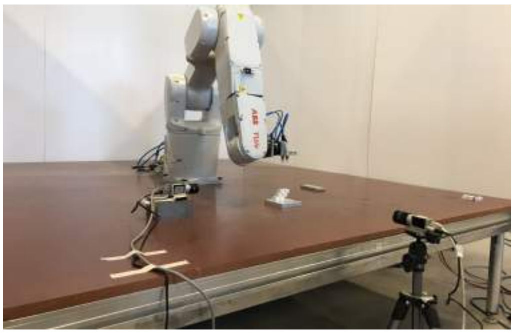
DS = distance sensor, IP = image processing

The possible reasons for the differences between the built and digital model will be improved in the upcoming tests to get to more consistent test results.
From all the gained information with the first two tests, it is concluded that the image processing measuring technique is researched further since a combination with different cameras from different angles can be made. The next step is to use two cameras from two different angles in combination with some small improvements to the fabrication process concluded from Table 3.8.

3.4.3 Test 3: Image processing with two cameras

As mentioned in Section 3.4.2.1, image processing is the measuring technique that will be investigated further with more tests. The second test with this measuring technique involves two cameras from two different angles, a front view and a side view (shown in Figures 3.22a and 3.22b). This makes it possible to capture the displacements in two directions (the X and the Y direction) where the previous set-up with image processing was only able to capture a displacement in one direction. By placing the cameras in the front and at the side, it is still possible to capture all the placed bricks by their marks when a new brick is placed. This also provides information on the locations of the bricks after movement has happened on top of them. This would not be possible if one of the cameras is placed from above since the marks on the bricks would then disappear by the placement of a new brick on top of the mark.

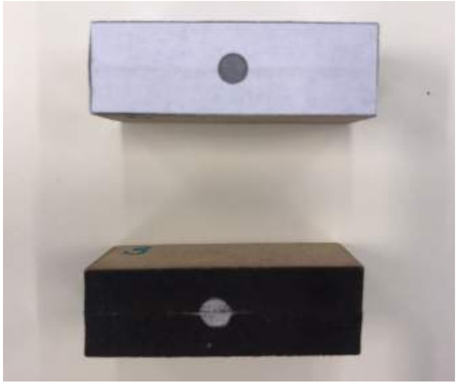
(a) Test set-up image processing



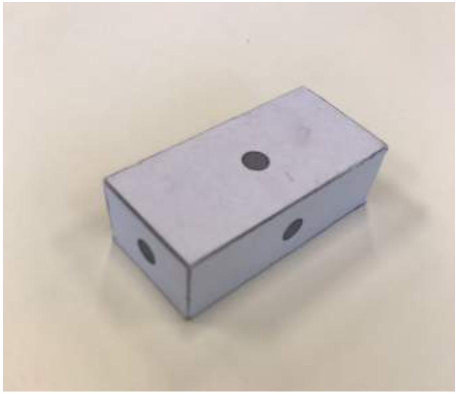
(b) Test set-up image processing

Figure 3.22: Image processing test with two cameras

With this second image processing a few adjustments were made to the set-up next to the addition of the second camera. These adjustments were done after the first conclusion on the differences between the digital model and the built model summed up in Table 3.8. The marks on the bricks, which are used to determine the coordinates of the centre of masses, were the first adjustment since these were set by hand for the previous image processing test. This caused uncertainties since the marks were not placed in the exact middle of the bricks and varied per brick. For this second image processing test the marks were printed to the exact size of the bricks and glued to three sides of the bricks. This resulted in less deviations in the marks and therefore less deviations in the measured data. In Chapter 6 it can still be found that these marks could contribute slightly to deviations between the digital and the built model but already less than with the previous marks.



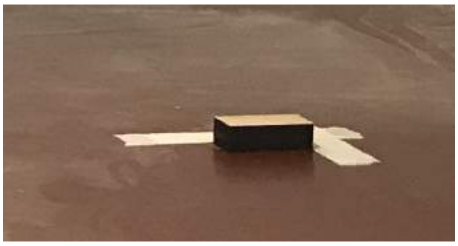
(a) Adjustment marks



(b) Three marks on one brick

Figure 3.23: Adjustment marks on bricks

The second adjustment compared to the first image processing test concerns the pick-up point. Previously the pickup point was marked by hand with tape to place the bricks at the same location every time the robot picks up a new brick. However, by using tape, the pick-up point becomes less accurate when used often since the lines of the tape become less sharp. For this second test with the image processing technique a pick-up point of metal was created. The plate with the pick-up point on top still needs to be placed by hand but the pick-up point itself will be the exact same every time a new brick is placed.

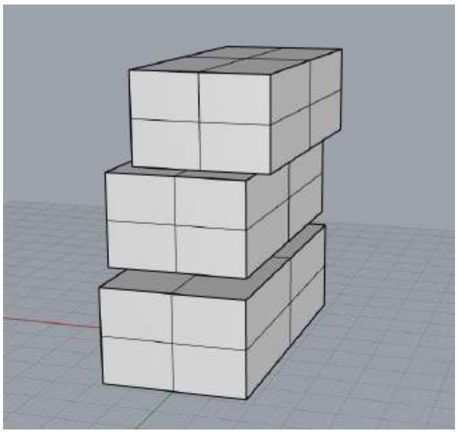


(a) Pick-up point tape

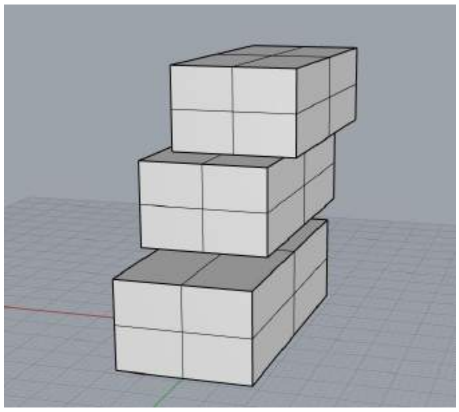
(b) Pick-up point metal

Figure 3.24: Adjustment pick-up point

Since this second image processing test makes use of two cameras from two different angles, the deviations in two directions can be captured, measured and compared to the digital model. Therefore the structures of three bricks shown in Figures 3.3a to 3.5b are changed a bit by the addition of a small deviation in a second direction (shown in Figures 3.25a and 3.25b. This deviation in the second direction (the Y-direction) has for this test been kept the same for all the brick structures. In an upcoming test this deviation will also vary just as the deviation in the X-direction already varies.



(a) Structure with amplitude 0,75 in X-direction and 0,30 in Y-direction



(b) Structure with amplitude 1,35 in X-direction and 0,30 in Y-direction

Figure 3.25: Brick structures with deviations in two directions

Tables 3.9 and 3.10 give the same data as the tables in the previous section (X-coordinate of the centre of masses of the bricks) but now from the test with the image processing measuring technique with two cameras plus the improvements of the marks on the bricks and the pick-up point. The data from this second image processing test comes closer to the digital model data compared to the test with only one camera. This can be explained by the conducted changes to the marks on the bricks and the pick-up point. These changes make the data more consistent and less calculation by hand needs to be conducted. The calculation by hand that was conducted in the previous test was regarding the marks on the bricks since the measured data before taking into account the quite large deviations of these marks was not comparable to the digital model data. This step, however, could be left out with this particular test since the marks on the bricks were much more reliable. These changes make the change in data from the previous test, not the measuring technique itself since this was the exact same as for the previous test. Figure 3.26 shows the differences between the digital and the built model for this third test of a structure with X-amplitude 0,50. There is almost no difference visible which also showed from the tables.

	Image processing data			**Digital model**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	0,04	5,67	8,44	0	5,52	8,46
0,75	0,00	8,45	12,71	0	8,29	12,70
1,00	0,05	11,15	16,78	0	11,05	16,93
1,25	-0,04	13,97	21,12	0	13,81	21,16
1,35	-0,06	15,08	22,99	0	14,92	22,85
1,40	0,01	15,57	23,83	0	15,47	23,70

Table 3.9: Image processing and digital model data centre of masses in mm (X-direction)

	Differences in mm			Differences in percentage (%)		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,04	-0,15	0,02	-0,10%	-0,40%	0,06%
0,75	0,00	-0,17	-0,01	-0,01%	-0,45%	-0,03%
1,00	-0,05	-0,10	0,14	-0,14%	-0,27%	0,39%
1,25	0,04	-0,16	0,04	0,10%	-0,43%	0,10%
1,35	0,06	-0,17	-0,14	0,17%	-0,46%	-0,37%
1,40	-0,01	-0,10	-0,13	-0,01%	-0,28%	-0,36%

Table 3.10: Comparison image processing with two cameras data and digital model (X-direction)

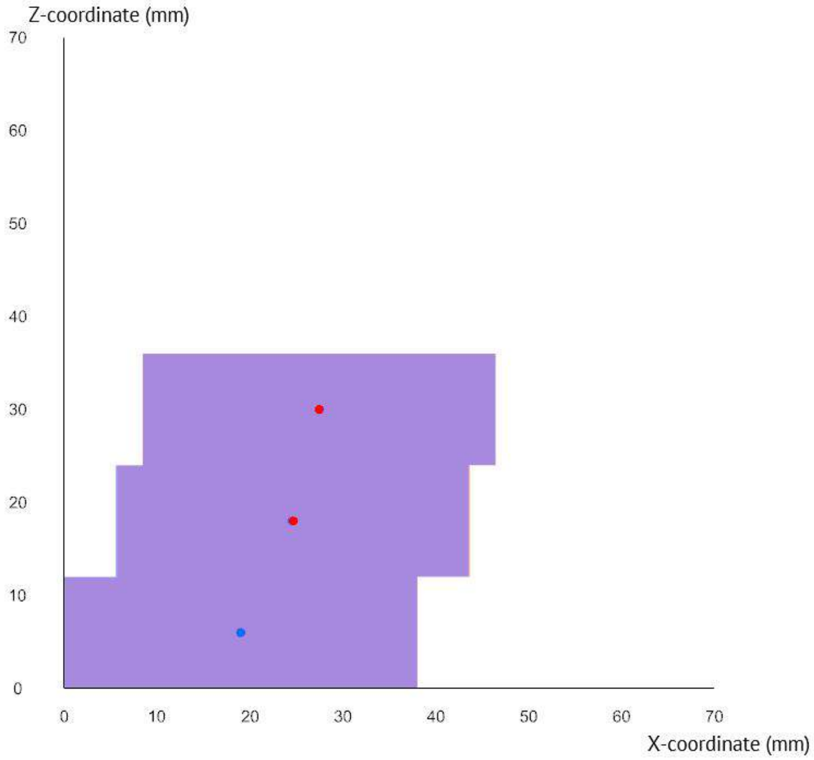


Figure 3.26: Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 3

Due to the addition of the second camera, the Y-direction of the centre of masses of the bricks can now be measured as well. Tables 3.11 and 3.12 have the same set-up as all the other tables but now for one Y-amplitude. Only one Y-amplitude was tested in this third test to get a first idea for this test set-up and data range. The negative numbers in Table 3.12 shows that the built structure is more to the right compared to the initial structure. This is also visible in Figure 3.27. However, there are no big differences which suggests that the set-up for the amplitudes in the Y-direction is good enough to conduct more tests. Therefore this Y-direction is tested more in the fourth test with structures with different Y-amplitudes.

	Image processing data			**Digital model**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,30	0,00	-2,04	-0,59	0	-2,54	-0,87

Table 3.11: Image processing and digital model data centre of masses in mm (Y-direction)

	Differences in mm			**Differences in percentage (%)**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,30	0,00	-0,50	-0,28	0,00%	-2,75%	-1,54%

Table 3.12: Comparison image processing with two cameras data and digital model (Y-direction)

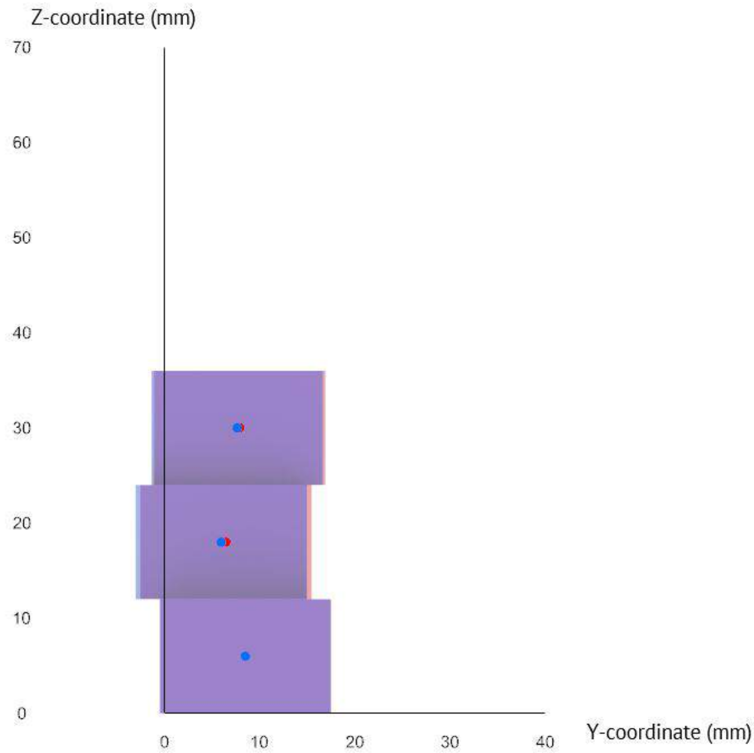


Figure 3.27: Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,30 of test 3

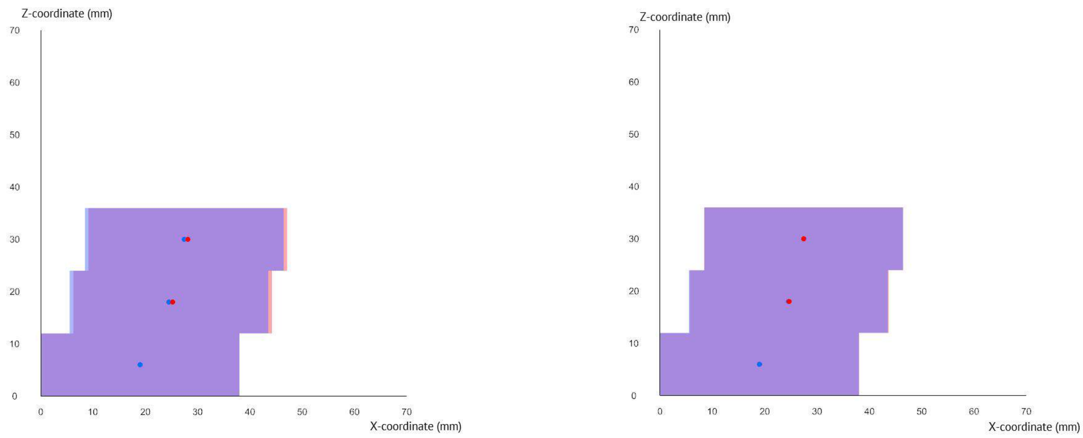
Tables 3.13 and 3.14 show the data from the image processing test with one camera compared to the same test with two cameras and the aforementioned improvements only for the data in the X-direction. The positive numbers in Table 3.14 show the distance that the measured data of the latter test is closer to the digital model compared to the test with one camera. This is also visible in Figures 3.28a and 3.28b since the latter of the two figures shows more overlap in the bricks meaning that the differences between the two models are smaller. It is therefore concluded to move further with this measuring technique into the fourth test.

	Image processing data 1 camera			**Image processing data 2 cameras**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	0,01	6,23	9,12	0,04	5,67	8,44
0,75	0,03	8,89	13,07	0,00	8,45	12,71
1,00	-0,01	11,57	17,19	0,05	11,15	16,78
1,25	-0,01	14,18	21,37	-0,04	13,97	21,12
1,35	0	15,29	23,02	-0,06	15,08	22,99
1,40	-0,02	15,87	23,91	0,01	15,57	23,83

Table 3.13: Image processing with one and two cameras data centre of masses in mm (X-direction)

	Differences in mm			**Differences in percentage (%)**		
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 1**	**Brick 2**	**Brick 3**
0,50	-0,03	0,56	0,68	-0,07%	1,48%	1,78%
0,75	0,02	0,44	0,37	0,07%	1,16%	0,96%
1,00	-0,06	0,43	0,41	-0,16%	1,12%	0,66%
1,25	0,03	0,21	0,25	0,07%	0,56%	0,66%
1,35	0,06	0,21	0,03	0,16%	0,55%	0,08%
1,40	-0,02	0,30	0,08	-0,06%	0,79%	0,21%

Table 3.14: Comparison image processing with one and with two cameras (X-direction)



(a) Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 2

(b) Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 3

Figure 3.28: Visualizations of the outcomes of test 2 and test 3

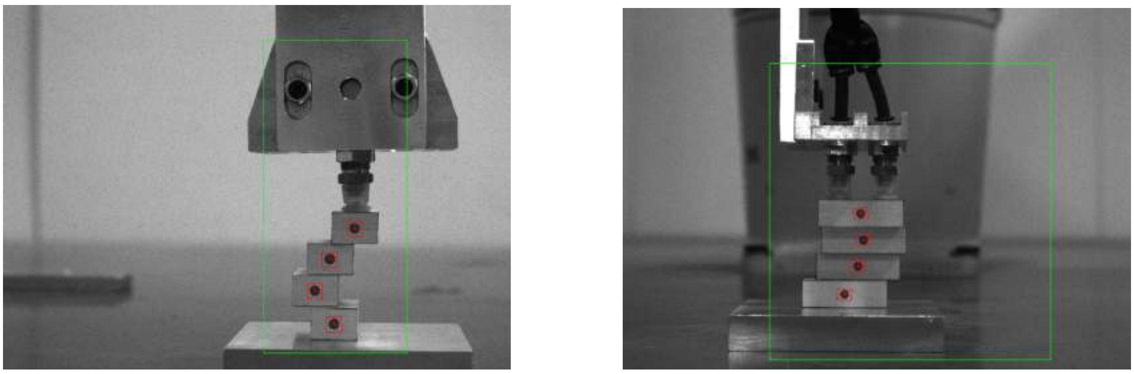
3.4.4 Test 4: Image processing with two cameras, focus on Y-direction

Several more tests are conducted with the same test set-up with two cameras using the image processing measuring technique. One of these tests made use of 6 bricks instead of 3 to see if the framework also works for structures already a step bigger than the first structures.
However, during this particular test, where the structure was increased in height with three bricks, the bricks started shaking from the moment when the fifth brick was placed. This shaking had quite an impact on the location of the bricks that where already placed. It seemed that the shaking is caused by the table to which the robot is attached. When the robot went to the exact place of the fifth brick with the last slow movement the table started shaking. This could for example be a contributor to the differences between the digital and the built model in the data in this section.
Next to the shaking, both the cameras had to be calibrated before the tests where conducted. However, especially with this test and having a closer look at the results, it became a bit unclear if the calibration was done correctly. The calibration factor was therefore checked with a first test run of a structure of two bricks. In this way there is more security that the results are useful after all the structures of the test are built with the robot.

(a) Robot set-up with 6 bricks

(b) Structure with 6 bricks with displacements in two directions

Figure 3.29: Test set-up for test 4, image processing with two cameras



(a) Vision Builder side view

(b) Vision Builder front view

Figure 3.30: Vision Builder views

From the conducted tests, data was collected from Vision Builder. This data is used to improve the tests, work on the Grasshopper model and look at the next steps and improvements of the model. The upcoming Tables give the same information as the previous tables but now for structures of 6 bricks and for deviations in the Y-direction since this direction and test set-up was tested more elaborately in this fourth test.

	Test 4, image processing 2 cameras Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	-0,60	-2,28	-0,43	2,65	1,81	-2,11
0,40	-0,29	-2,84	-0,45	3,73	2,57	-1,89
0,60	0,02	-4,11	-0,58	5,42	4,03	-2,64
0,80	0,41	-5,43	-0,86	6,82	-	-
1,00	0,46	-7,08	-1,20	8,43	-	-

Table 3.15: Image processing data centre of masses in mm (Y-direction)

	Digital model Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0	-2,54	-0,87	2,22	1,64	-1,64
0,40	0	-3,38	-1,14	2,95	2,16	-2,16
0,60	0	-5,06	-1,65	4,37	3,17	-3,17
0,80	0	-6,74	-2,12	5,75	-	-
1,00	0	8,40	-2,56	7,07	-	-

Table 3.16: Digital model data centre of masses in mm (Y-direction)

The differences between the built and digital model are again shown in Table 3.17 and 3.18. The differences are also visualized in Figure 3.31 where is becomes visible that the built structure is constantly placed more to the right compared to the digital structure.

	Differences in mm					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0,60	-0,25	-0,44	-0,43	-0,17	0,47
0,40	0,29	-0,54	-0,68	-0,78	-0,41	-0,28
0,60	-0,02	-0,95	-1,08	-1,04	-0,86	-0,53
0,80	-0,41	-1,30	-1,26	-1,07	-	-
1,00	-0,46	-1,32	-1,36	-1,35	-	-

Table 3.17: Comparison image processing two cameras and digital model in mm (Y-direction)

	Differences in percentage					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	3,33%	-1,41%	-2,43%	-2,36%	-0,95%	2,61%
0,40	1,62%	-3,00%	-3,80%	-4,35%	-2,26%	-1,53%
0,60	-0,13%	-5,28%	-5,98%	5,79%	-4,78%	-2,92%
0,80	-2,28%	-7,25%	-7,01%	-5,94%	-	-
1,00	-2,53%	-7,36%	-7,56%	-7,52%	-	-

Table 3.18: Comparison image processing two cameras and digital model in % (Y-direction)

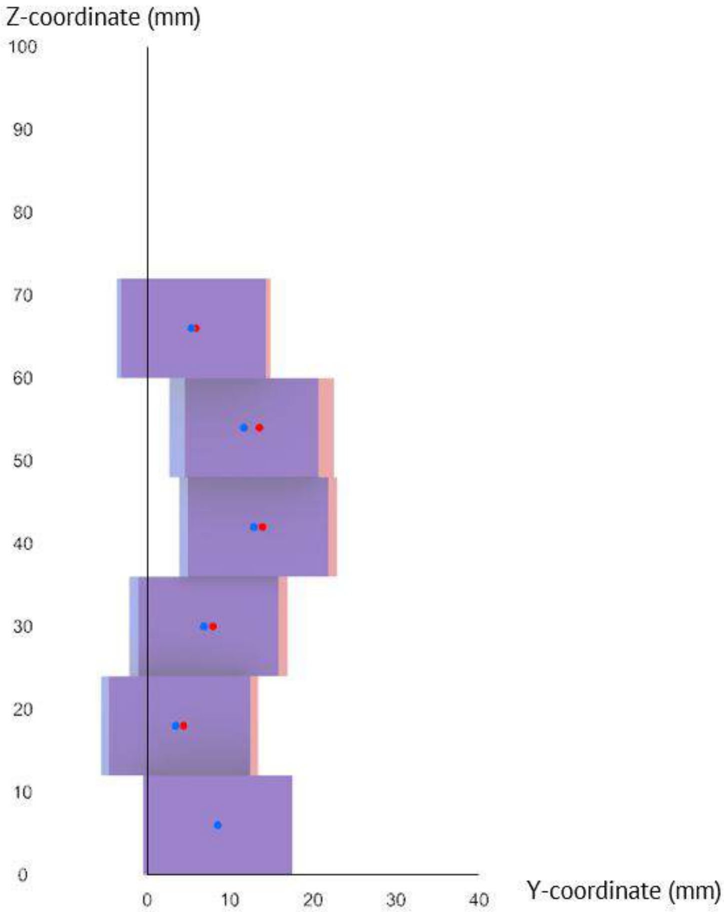


Figure 3.31: Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,60 of test 4

The same data is collected for structures in the X-direction with a height of 6 bricks. However, now only one amplitude is tested (X-amplitude of 0,50) since the focus for this test was laid on the different structures in Y-direction.

	Test 4. image processing 2 cameras X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	5,60	8,18	7,08	2,69	-2,72

Table 3.19: Image processing data centre of masses in mm (X-direction)

	Digital model X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	5,52	8,46	7,44	2,94	-2,94

Table 3.20: Digital model data centre of masses in mm (X-direction)

From the differences in Tables 3.21 and 3.22 and Figure 3.32 it is clear that the differences between the two models are rather small.

	Difference in mm					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	-0,08	0,28	0,36	0,25	-0,21

Table 3.21: Comparison image processing two cameras and digital model in mm (X-direction)

	Difference in percentage					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00%	-0,21%	0,74%	0,96%	0,66%	-0,56%

Table 3.22: Comparison image processing two cameras and digital model in % (X-direction)

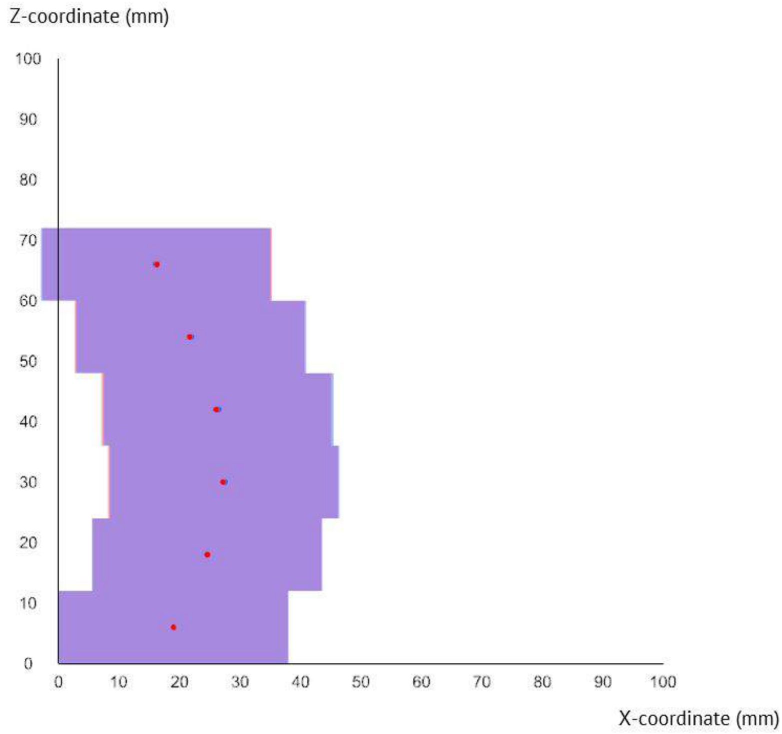


Figure 3.32: Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 4

The test set-up with the image processing measuring technique turned out to give useful data to this point. It is therefore concluded to move further with this measuring technique. However, for the fifth and last test the bricks will be increased in size. This change is supposed to have two positive effects on the tests. First, the weight of the bricks and the surface touching the working table will be bigger. By doing this, the structure should be less vulnerable for shaking of the table. This will be tested during the fifth test. Next to the shaking issue, with the usage of larger bricks, it can be investigated if the same percentage of deviations is present. Data from the tests with small bricks can then be compared to data from the tests with larger bricks which will be elaborated in Chapter 6.

3.4.5 Test 5: Image processing with two cameras, bigger bricks

As mentioned, for this fifth and final test bigger bricks were used. Next to the changed bricks there are no real differences to the fourth test in terms of the test set-up. For this first test with larger bricks only one amplitude in the X-direction and one amplitude in the Y-direction were tested. This was done since this test was more focused on the shaking issue and how a test with bigger bricks would work rather than the data from this test. More results with tests with the bigger bricks are included in Section 4.2. To avoid the shaking issue the speed of the robot was decreased to 50 % of the initial speed of the model.

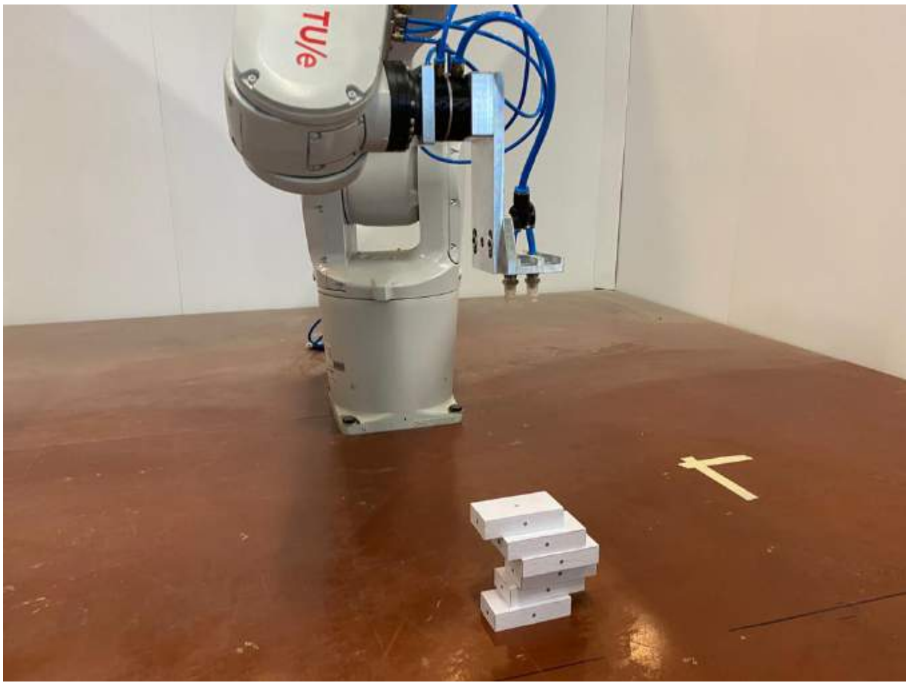


Figure 3.33: Test set-up of test 5

The same data as for all the previous tests are shown in the upcoming tables. The first four tables contain the data for the structure with an X-amplitude of 0,50.

	Test 5.Image processing 2 cameras X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	8,19	12,17	9,86	2,60	-6,60

Table 3.23: Image processing data centre of masses in mm (X-direction)

	Digital model X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	8,29	12,70	11,16	4,41	-4,41

Table 3.24: Digital model data centre of masses in mm (X-direction)

	Differences in millimeters					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00	0,10	0,53	1,30	1,81	2,19

Table 3.25: Comparison image processing two cameras and digital model in mm (X-direction)

	Differences in percentage					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,00%	0,13%	0,68%	1,69%	2,35%	2,84%

Table 3.26: Comparison image processing two cameras and digital model in % (X-direction)

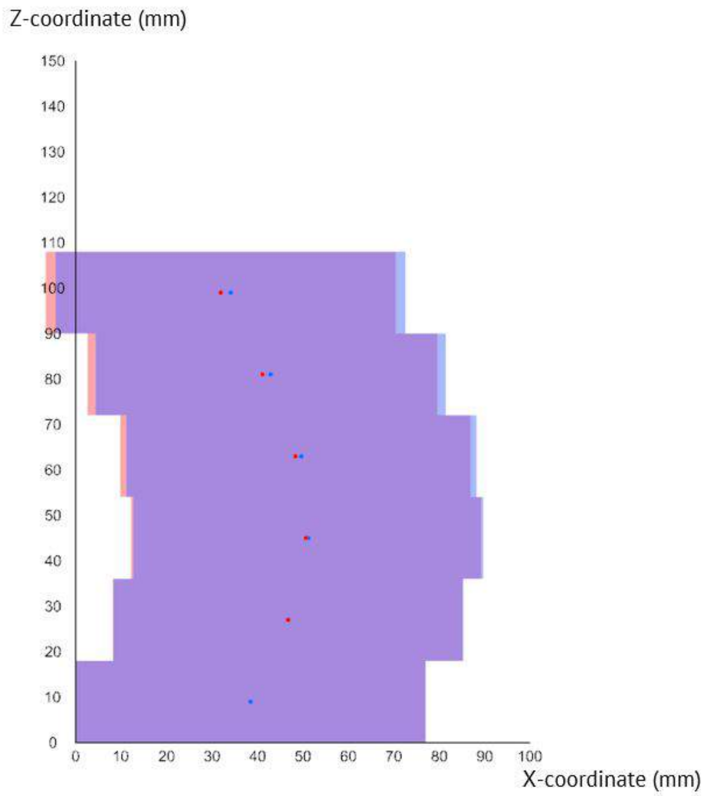


Figure 3.34: Visualization of the digital (blue) and the built structure (red) with X-amplitude 0,50 of test 5

The next set of 4 tables contains the data for the structure with an Y-amplitude of 0,30.

	Test 5. Image processing 2 cameras Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0,00	-3,30	-0,77	3,18	2,72	-1,84

Table 3.27: Image processing data centre of masses in mm (Y-direction)

	Digital model Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0,00	-3,81	-1,30	3,33	2,45	-2,45

Table 3.28: Digital model data centre of masses in mm (Y-direction)

	Differences in millimeters					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0,00	-0,51	-0,52	0,15	-0,26	-0,62

Table 3.29: Comparison image processing two cameras and digital model in mm (Y-direction)

	Differences in percentage					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,30	0,00%	-1,37%	-1,42%	0,40%	-0,71%	-1,66%

Table 3.30: Comparison image processing two cameras and digital model in % (Y-direction)

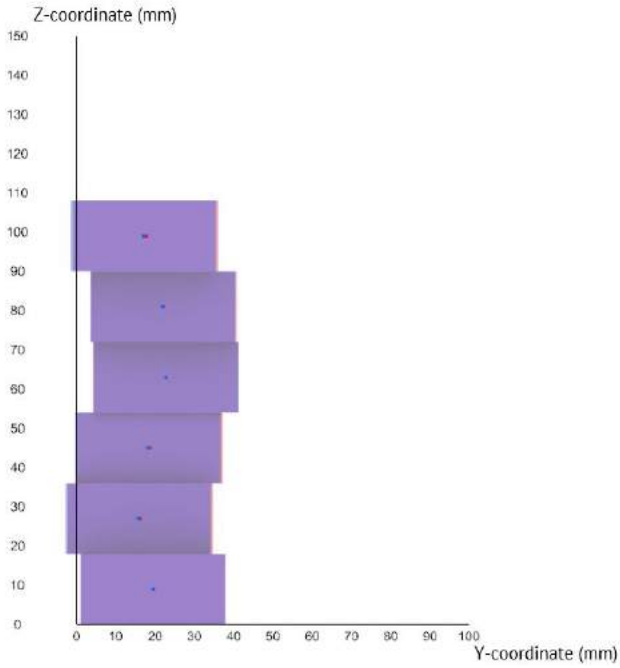


Figure 3.35: Visualization of the digital (blue) and the built structure (red) with Y-amplitude 0,30 of test 5

The conclusion from this fifth test is that this test showed less vibrations to the structures that were visible with the naked eye. There are some differences between the digital and the built model but it is found that these bricks should be used for the final test (Section 4.1) since these bricks are also already a step closer to bricks that could be actually used on a building site since they are just bigger in size. The differences between the models will be further investigated within the final test with more data and different structures to test.

3.5 Data processing

Within Section 3.4 different measuring techniques have been investigated and tested with the robotic arm. Within that section the program Vision Builder was mentioned since this program is used to process the data from the cameras. This section explains the work of Vision Builder for automated inspection in more detail together with the connection to the Grasshopper framework where the data from Vision Builder is used.

Figure 3.36 shows the overview of Vision Builder and all the steps that are taken within the program itself. Figure 3.36 shows the overview for just one camera. However, there are two cameras used for the test set-up. This means that the sequence of steps shown in the overview are conducted once more after the last step shown in the overview for the second camera.

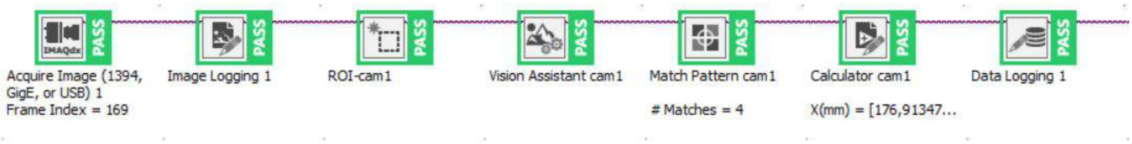
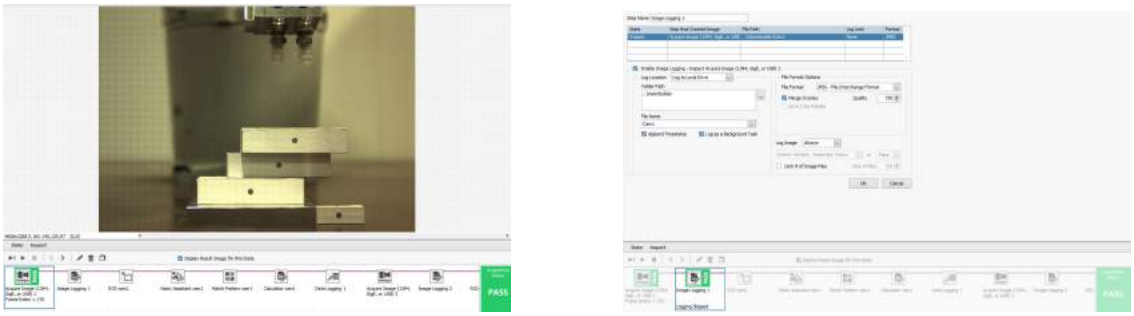


Figure 3.36: Overview of steps in Vision Builder

The first two steps that are conducted in Vision Builder are the 'acquire image' step (Figure 3.37a) and the 'image logging' step (Figure 3.37b). The 'acquire image' step selects a camera and uses the view of that camera as an input for the next steps within Vision Builder. The 'image logging' step saves the image that is made with that camera. This is not necessary for the output of Vision Builder. However, this step is incorporated at this point since Vision Builder can make the same analysis that is now executed from a camera view from a picture. Therefore, by saving the pictures, it is possible to do a new analysis of the picture at a later point in time if needed for this research.

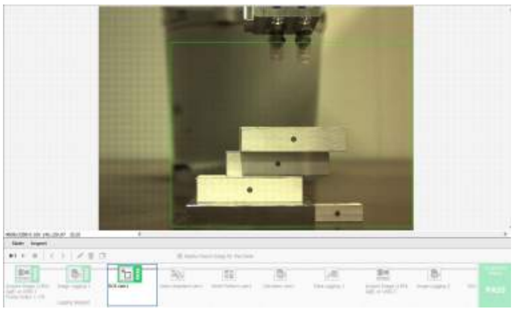


(a) 'Acquire image' step

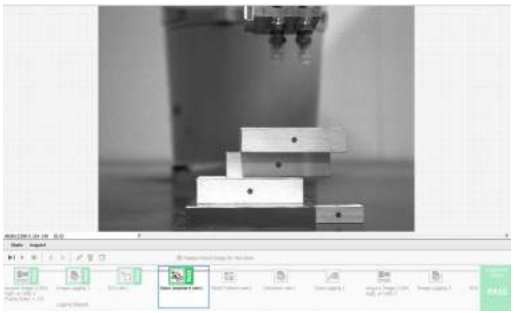
(b) 'Image logging' step

Figure 3.37: Acquire image and image logging

The next steps that are taken are the 'region of interest' step (Figure 3.38a) and the 'vision assistant' step (Figure 3.38b). The 'region of interest' step selects a region (marked with the green lines) in which vision builder later on searches for marks. By selecting this region, it can be made sure that only the bricks within this region are analysed instead of other bricks that are also in the camera view but do not need to be analysed for example. The 'vision assistant' step filters out the colour of the image, making it easier for Vision Builder to conduct the rest of the analysis.



(a) 'Region of interest' step

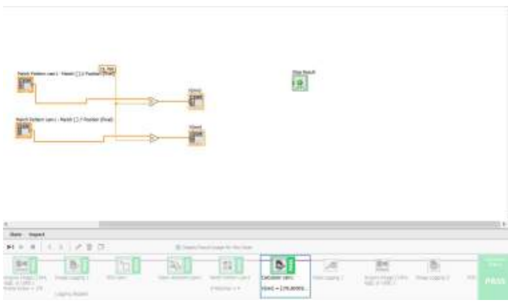


(b) 'Vision assistant' step

Figure 3.38: Region of interest and vision assistant

When the region of interest is set and the image's colour is filtered, the 'match pattern' step (Figure 3.39a) can be conducted. Within this step a pattern is selected for which Vision Builder will search within the set region of interest. In this case the pattern is the black dot that is placed multiple times on every brick. Within Vision Builder it is possible to say how much other patterns can be the same as the original pattern. This could be helpful when not all the patterns are the same for example. Vision Builder searched for all the possible matches and then calculates their middle point. This means that in this research case, the dotted marks are used to define the middle point of these dots. This information is than saved in Vision Builder in pixels. To get from pixels to millimeters, a 'calculator' step (Figure 3.39b) is added after the 'match pattern' step. Within this step a calibration factor is added which is used to calculate millimeters from the found pixels in the previous step. This calibration factor is defined with the first test run of the test set-up. Within this test run a distance is measured in Vision Builder, this could be for example the length of a brick. Vision Builder than gives the amount of pixels of this distance. However, the actual length of a brick is also known in millimeters. The amount of pixels in than divided by the length in millimeters to get to a calibration factor than can be used in the 'calculator' step. The information that Vision Builder gives first in pixels and later in millimeters is relative to the camera view. This means that the upper left corner of the camera view is point 0,0 and the outcomes from Vision Builder in millimeters are the distances from this upper left corner.

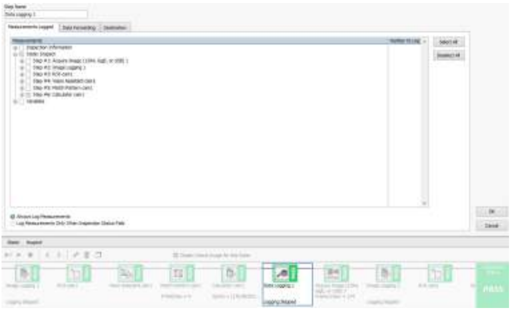
(a) 'Match pattern' step



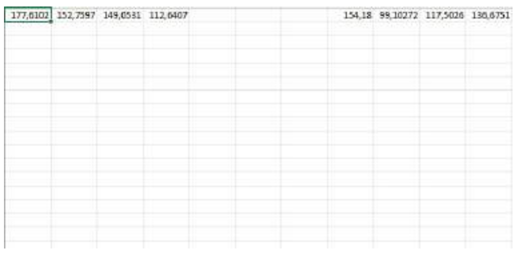
(b) 'Calculator' step

Figure 3.39: Match pattern and calculator

The last step that needs to be taken within Vision Builder itself is the 'data logging' step (Figure 3.40a). This step transfers the information obtained from the 'match pattern' and 'calculator' steps to a CSV file. Within this step it is possible to add or omit information to the file or file name, such as a timestamp. It is also possible to make one CSV file in a folder and add all the newly obtained information to that file. This approach is used for this research since all the data is then stored in one file and within the Grasshopper model the newest row of data is selected to be sure that all the data points of the last measurement are used. Figure 3.40b shows how the data from Vision Builder is translated to a CSV file.



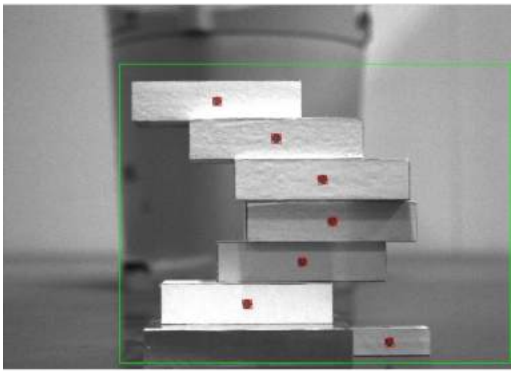
(a) Data logging



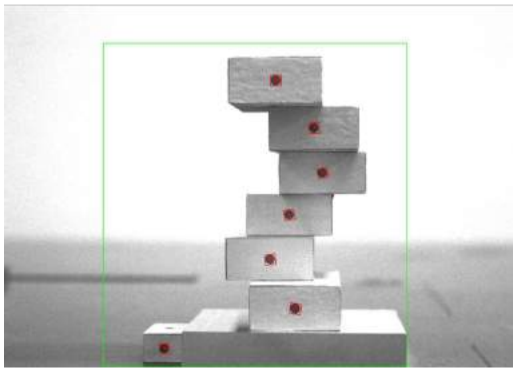
(b) CSV file from Vision Builder

Figure 3.40: Data logging and CSV file

Figures 3.41a and 3.41b shows the view that becomes visible within Vision Builder after the program has run. All the elements of the analysis are visible for both the camera views. Together with this view in Vision Builder itself, two images of these structures are saved and two CSV files are created (one for every image). This is done when the play button within Vision Builder is pushed. In Section 3.7 it will be explained how a picture can be taken at the right moment in time, within the fabrication process. For now, this button is pushed manually every time the robot lays one brick to obtain all the necessary information.



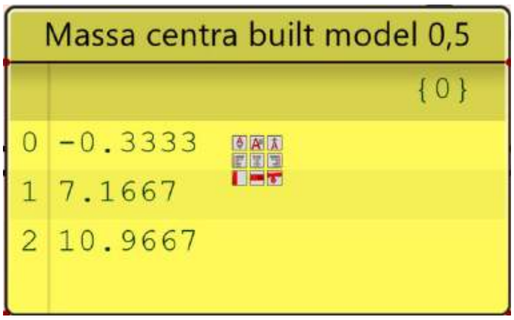
(a) Analysis front camera



(b) Analysis side camera

Figure 3.41: Vision Builder analysis

From Vision Builder a link to Grasshopper is made. In the first tests this link was created by hand by using the data from the CSV file (Figure 3.40b), obtained from Vision Builder, and adding it to Grasshopper manually (Figure 3.42a). When the data was entered by hand, the data from Vision Builder needed to be translated to coordinates of the centre of masses since the output of Vision Builder only gave distances relative to the camera view. This is also done in Excel beforehand by calculating the differences between the distances from the points. From this data, both X and Y-directions (in Figure 3.42b only data for the x-direction is given) points are made that represent the centre of masses of the built structure. This can be done by hand when a live connection to Excel is not acquired. However, when a live connection with Excel is made the steps are the same. The only addition to the Grasshopper model is the processing of the data of Excel to actual coordinates with the same results as shown in Figure 3.42b.

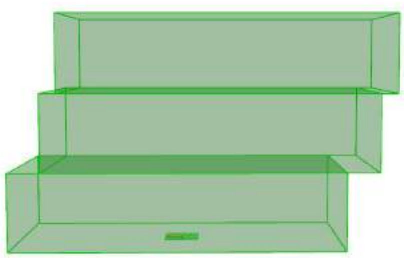


(a) Manually imported data

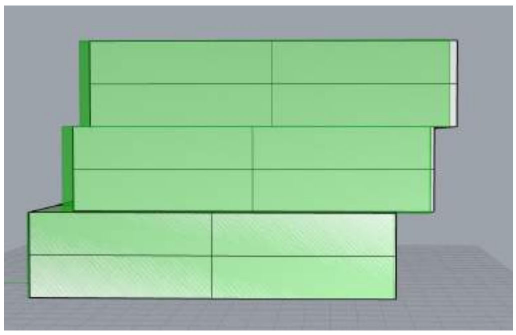
(b) Data translated to points

Figure 3.42: Data from Vision builder to points

From the coordinates of the centre of masses shown in Figure 3.42b a structure can be defined by placing bricks on the coordinates of the centre of masses (Figure 3.43a). This new digital structure of the built model can than be compared to the initial digital model (Figure 3.43b) where the green structure in this case is the built model and the white structure is the initial digital model.



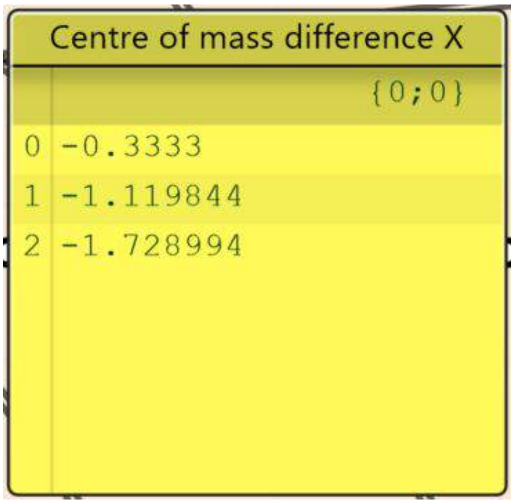
(a) Points translated to structure



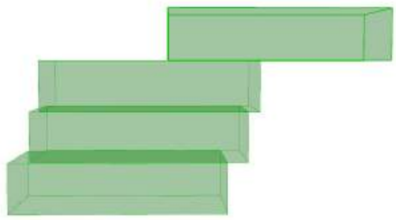
(b) Comparison built structure to initial digital model

Figure 3.43: Data from Vision builder to structure

The comparison between the digital model and the built model is also expressed in a panel where it can be seen what the difference in millimeters between the two models is. This is shown in Figure 3.44a where, for now, only the differences of the centre of masses in the X-direction are given for the stacked blocks. This is also done for the Y-direction in the Grasshopper model. The placed bricks are combined with the next brick that is about to be placed by the robot. This combination of the placed bricks and the yet to be placed brick is shown in Figure 3.44b. This Figure shows an extreme case where the brick that is yet to be placed as a more extreme amplitude than the bricks that are already placed. This is done to show the first version of the data conclusion in the next section (Section 3.6) works.



(a) Differences between built and digital model



(b) Combination of placed bricks and new brick

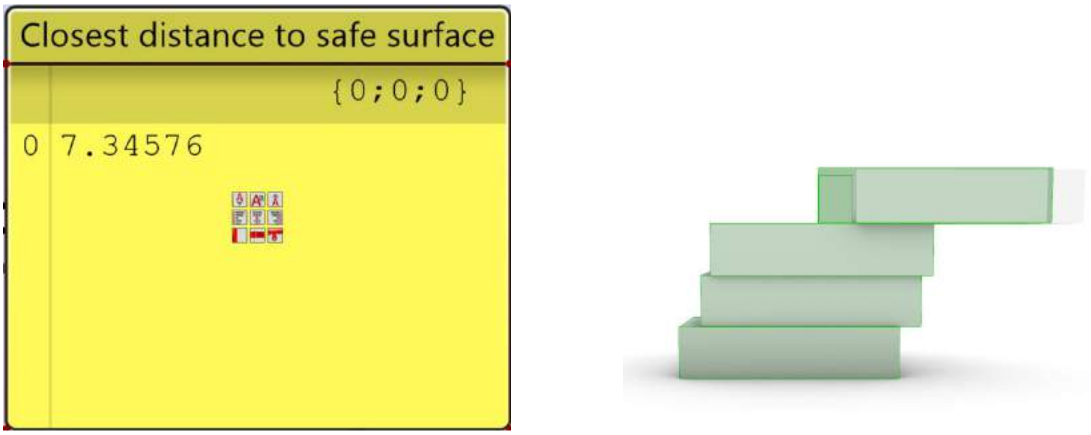
Figure 3.44: Steps of the data processing part of the framework

3.6 Data conclusion

As already mentioned, there are multiple ways of drawing conclusions for the further build of the structure. The first option that was used and explored in this research it to look at the first new brick that will be placed on top of the built structure. The second option is to look at all the bricks that are yet to be placed on top of the built structure. With the second option it becomes earlier visible what the consequences are for the rest of the structure due to a deviation between the digital and the built model. However, the first option was used in the early stage of the framework to see how this works and how new coordinates for bricks can be made. For both options a new structural check is conducted. This structural check is the same as the structural check described in Section 3.2, however, the input is slightly different. For the first option the input is as shown in Figure 3.44b where only one new brick is placed on the built bricks. For the second option the input is the same as for the first option but now also the other remaining bricks of the structure are added on top of the build bricks. Both the options result into outcomes of the structural checks that either all suffice or have failed outcomes among them.

3.6.1 Data conclusion based on one brick

When looking at the structure in Figure 3.44b it is quite obvious that the brick that will be placed next is not stable and will fall. The structural check gives this as an output. Since the output of the structural check fails, changes have to be made to the next brick. If all the checks were to suffice, there are no changes needed and the robot can continue with building the structure. From the structural check, an additional calculation can be made to gain knowledge on how far the centre of mass of the check that fails is from the safe ground surface. The outcome is given in a panel in Grasshopper (Figure 3.45a) and this outcome is also used to determine the right coordinates for the next block for this first option in responding to the output of the data model (referred to as the data conclusion). Since the outcome of this additional calculation is the distance that the centre of mass is away from the safe ground surface, the centre of mass of the next block is moved with this distance. When this is done, a final structure is completed (Figure 3.45b, the green bricks are the final structure). This final structure consists of the first three, in this case, already built bricks and the next brick that is yet to be placed, with a coordinate which is safe to built. This results in a structure that will not collapse even though the initial placement of this next brick would result in a collapse.



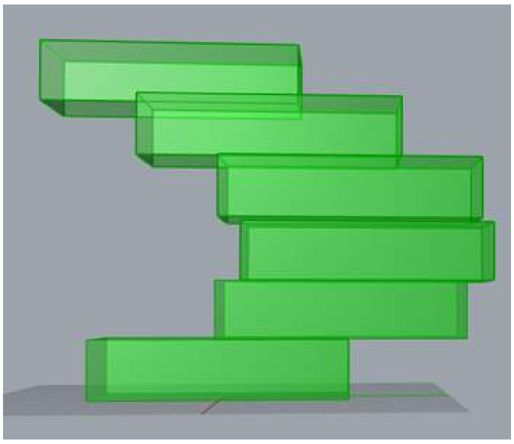
(a) Distance between centre of mass and safe structure

(b) Final structure (structure that will be built is green)

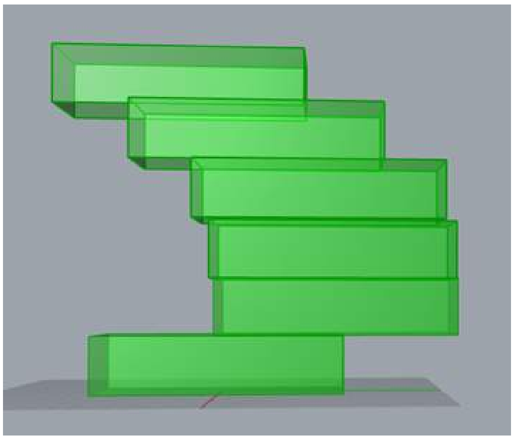
Figure 3.45: Steps of the data conclusion part of the framework

3.6.2 Data conclusion based on remaining structure

However, with the second option described, the calculation and adjustment to the coordinate is not the same. This is the case since there are possibly multiple checks that fail when the whole remaining structure is taken into account. It is then not as easy as with option 1 to say that there is one distance of a centre of mass to the safe surface that needs to shift. To get to a structure with all the remaining bricks taken into account when adjustments need to be made, the same optimization can be used as elaborated in Section 3.2. However, now the bricks that are already built are set as permanent and only the remaining bricks are used within the optimization to get to a result with no failed checks. Figure 3.46a and 3.46b show structure of which the first two bricks are already built and therefore not adjusted. The first figure shows these two bricks with all the remaining bricks of the original structure added on top. The second figure shows the two built bricks with an optimized remaining structure on top of which all the structural checks suffice.



(a) Two built bricks with remaining structure



(b) Two built bricks with optimized remaining structure

Figure 3.46: Optimization in data conclusion

Something that should be kept in mind is that option 1 can continue autonomously by the framework and for option 2 (the optimization) the optimization needs to be started manually at this moment and one of the outcomes should be selected before the next brick can be built by the robot. Therefore, the fabrication system needs to be stopped at this moment to conduct the optimization and select one of its options to continue.

The last part of this data conclusion step is the connection of the new coordinates of the brick that will be placed next to the robot code. From this last part of the framework, the framework will loop back to the part where the next brick will be built (Section 3.3). From this step the whole framework continues again with the data collection of the newly added brick.

Within this data conclusion part of the framework a choice is made to use the initial digital design of the structure and, if needed, alter the design to a design that is as close to the initial design as possible. The idea of this framework is, namely, that the designed structure is built in the best way possible and with the idea of first time right manufacturability. This also means that, when a brick is moved because it would otherwise collapse, it is still built on the edge of what is possible. With this principle also the least amount of material is used since the structure is prevented from collapse and no additional material is introduced. However, there are more possible solutions to tackle this problem which will be elaborated further in Chapter 6. This also depends on the situation and available material what other solutions could be possible.

3.7 Automation

3.7.1 Starting point

When combining all the steps of the framework, the framework works, however, at some points the data needs to be exported and imported from one program to the other. This, and some smaller steps, are not automatic. However, these small steps can be automated in order to get to a process where no human interference is needed. The steps that need to be taken to achieve a fully automated fabrication process for this research are the following:

1. The bricks that are to be placed need to be laid down by an automatic pick-up point.
2. The camera of the measuring technique should automatically take a view of the structure at the right time.
3. The measuring technique should process the taken view by itself and translate the taken view to useful data.
4. The data from the measuring technique should be loaded into Grasshopper automatically so that this data can be used for the data processing and data conclusion straight away.Data processing and data conclusion are together formulated as data analysis.
5. From the data conclusion a new robot code is made to build the next brick. This robot code should be directly and automatically uploaded back to the robot so that the robot does not have to stop building the structure.

All these steps are visualized in Figure 3.47. Section 3.7.2 explains all the steps taken in this research to obtain a fully automated framework.

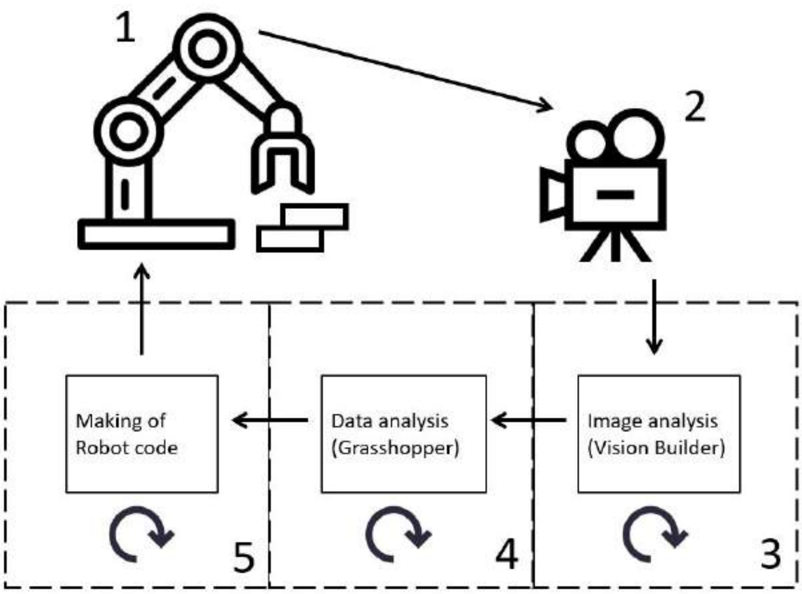


Figure 3.47: Schematic view of automation steps

3.7.2 Conducted steps to full automation

As explained in Section 3.7.1 there are five more steps to get to a fully automated framework. For each of these some additional steps to the framework or test set-up are added which are all separately explained.

3.7.2.1 Pick-up point

The pick-up point next to the robot can be improved in a way that a person does not have to place the next brick by themselves. On a real building site this would also not be the case since this is not the automated fabrication process that is wanted. Therefore, as a simple solution, a slide is made which fits the bricks used in this research in size and amount (Figure 3.48). This slide makes it possible to run the whole framework for the designed structure and without someone having to place every single brick in the pick-up point manually after the previous brick is picked up by the robot.

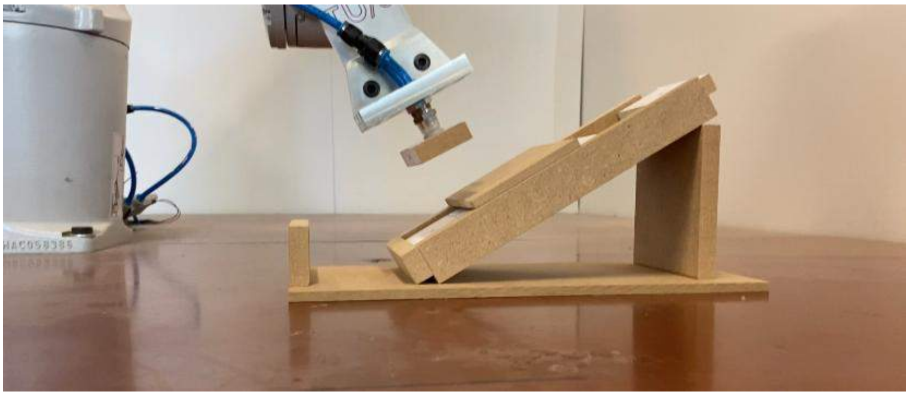


Figure 3.48: Pick-up point with slide

The slide is further automated with the addition of a pile of bricks on top of the slide and a small cylinder that pushes the bricks from the stock into the slide (Figure 3.49). This cylinder runs by a separate signal that is programmed in the robot code (DO-02). The DO-02 is a Digital Output that can be programmed within the robot code in Grasshopper. There is already a DO-01 signal that enables or disables the vacuumtool (the robot end-effector) and now this second signal is added especially for this cylinder in the automated pick-up point. This signal is set to true when the end-effector places a brick and is therefore away from the pick-up point. The brick is then loaded into the slide since the signal will cause the cylinder to push out. The signal is set to false when a brick is picked up meaning that the cylinder will be pulled back.

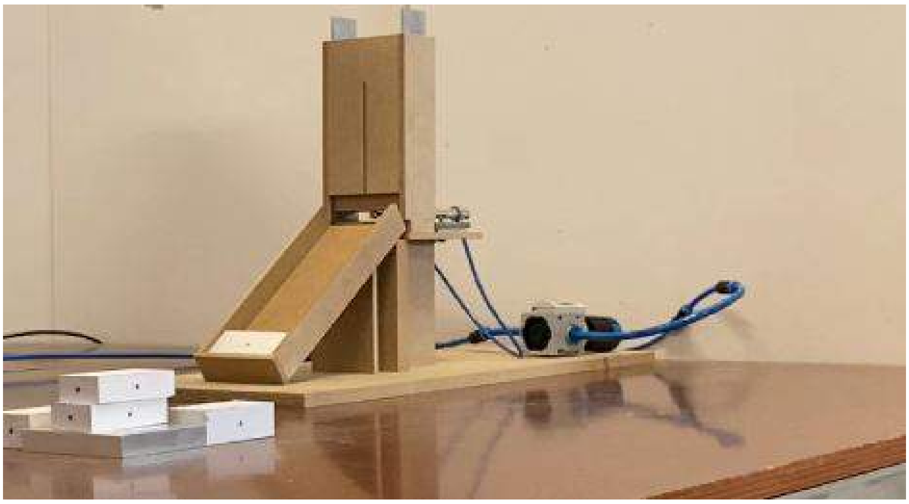


Figure 3.49: Pick-up point with slide

3.7.2.2 Automatic picture

The picture that will be analysed by Vision Builder can be made automatically by using the output signal of the end-effector (DO-01). The end-effector signal goes to true when the end-effector picks up a new brick. When this signal goes to true this also means that the robot is at the location of the pick-up point and not in the view for the picture therefore this signal is used to generate the pictures. The signal from the end-effector is translated to a digital signal that can be connected to a computer. This signal is then used inside Vision Builder by adding a small program that controls the data processing program as described in Section 3.5. This small additional program is shown in Figure 3.50. It shows that if the signal is true, the main analysis can be run and if the signal is false the signal is updated in its own small program again and no main analysis is run.

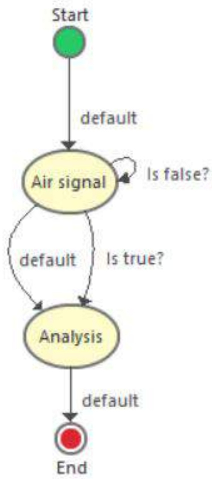


Figure 3.50: Additional Vision Builder program for automated picture

The small additional program (Figure 3.51) in Vision Builder that continuously runs first takes the signal from the end-effector. It stores this signal as variable 1 (Pactual), which is by default set to false. Then variable 1 is compared to variable 2 (Plast), also set to false as a default, in the calculator step (Figure 3.52). Variable 2 is the 'previous' signal. The picture should be taken when the signal goes from false to true since then once the picture is taken at the right moment and no additional pictures are taken. Therefore the calculator step says that the result is true when variable 1 is true and variable 2 is false. In all the other cases the result is false. If this result is true, the main analysis of the data processing will run.

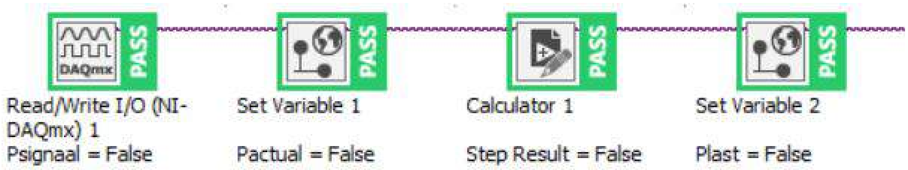


Figure 3.51: Additional Vision Builder program for automated picture

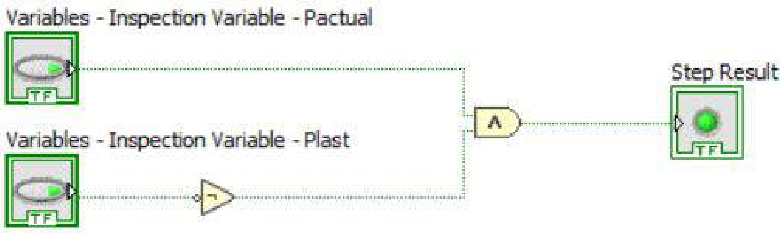


Figure 3.52: Calculator step of the additional Vision Builder program

3.7.2.3 Data production

When the picture is automatically taken by the end-effector signal inside Vision Builder, the whole main analysis will be run by itself since the previous step is now conducted in Vision Builder itself. The addition in this main analysis is the fact that every data analysis is added to the same CSV file and stored in a folder chosen before the whole manufacturing process starts.

3.7.2.4 Connection CSV to Grasshopper

The data gained from Vision Builder is stored in a CSV file in a folder on the used computer. However, this data needs to be imported into Grasshopper so that the analysis of the structure in Grasshopper can run. The easiest way to do this in the beginning of the research was to do this by hand and inserting the data from CSV into a Grasshopper panel (Section 3.5, Figure 3.42a). However, it would cost less time and less human action if this data from the CSV file was imported directly into Grasshopper. As a second step in this process, a Python component in combination with a boolean toggle was added in Grasshopper. The combination of these two components made sure that the data from the CSV file was imported into Grasshopper and stored in a datatree structure. This is only done for the last row of data since this is the data of the most recent picture. When this data is in Grasshopper, the same steps are conducted with this data es explained in Section 3.5. The exact Python code, the elements in Grasshopper and the outcomes of the elements in Grasshopper are explained in more detail in Appendix A. This step does not work completely automatically since the only thing that has to be done manually is switch the boolean toggle in Grasshopper from false to true, than the Python element loads the CSV file and the data is imported. Since the CSV file will be updated throughout the test, this toggle action needs to be switched after the placement of every brick.

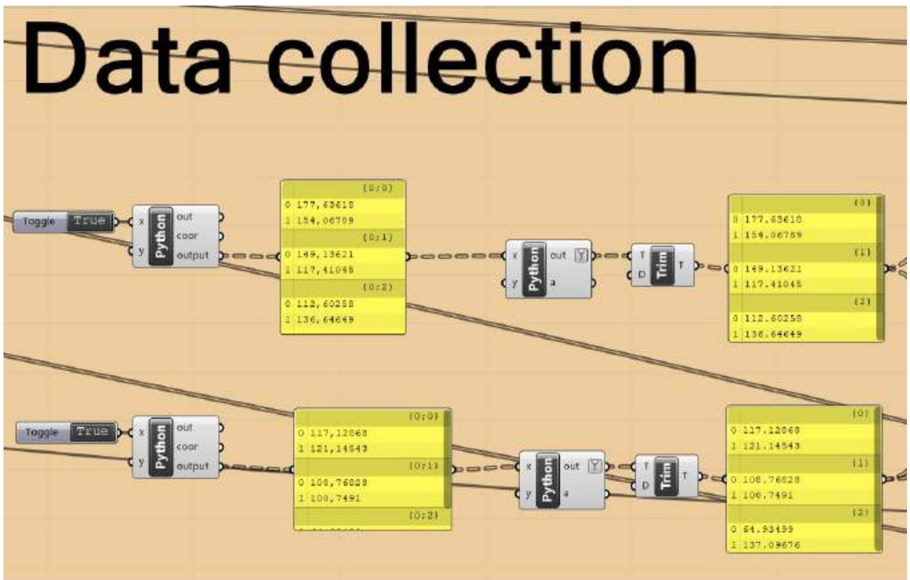


Figure 3.53: Grasshopper elements to import the CSV file

This last action of switching the boolean toggle can be automated as well by a live connection with the robot and using the same signal as used for taking the picture (Section 3.7.2.2). The same signal is used since this is also used for taking the picture and it is at that moment that the new data is uploaded to the CSV file. In order to be sure that the right data is taken a delay of 5 seconds is used for this upload. The remote connection (Figure 3.54) makes it possible to withdraw the status of the DO-01 signal of the robot (the digital output of the end-effector). In this case the signal is true which is collected from the remote connection component.

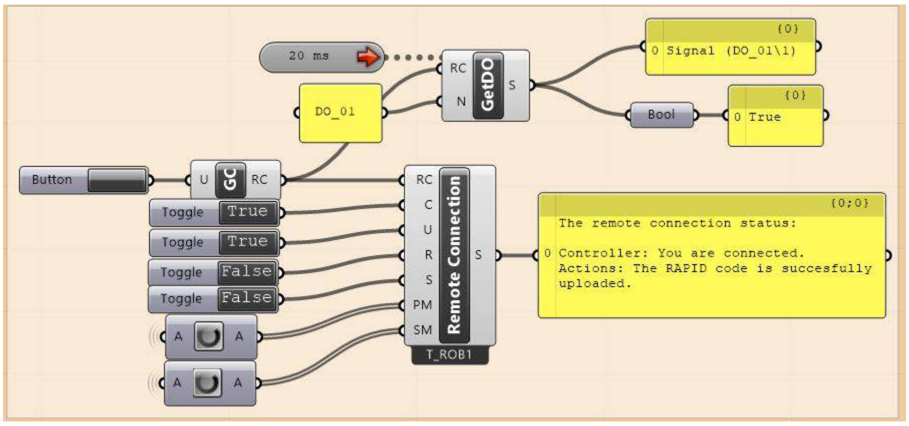


Figure 3.54: Remote connection component

3.7.2.5 Automatic update robot code to robot

The last step to get to an automated framework is the automatic update of the robot code to the robot. This step already goes automatically since the coordinates of the bricks that are used for the robot code are updated automatically. Therefore the robot code (the main and base code) are updated also automatically. When using a remote connection component and making a live connection to the robot, these pieces of code can be uploaded. Since the pieces of code update automatically and a choice can be made with the remote connection to upload constantly, the pieces of code can be uploaded to the robot also automatically. The only thing that still needs to be done is to grant permission to change the codes on the robot panel. This is still a setting of the robot in the TU/e Structures Laboratory because of safety measures. Since there is no entirely closed cage around the robot, it could be the case that the robot moves unexpectedly when the codes are just changed.

3.7.3 Result of automation

The final result of the framework combined with all the steps that are taken to automate the dataflow of the framework results in a digital twin. A digital twin is a frame where the dataflow between a physical object (in this case the build structure) and the digital object (the digital version of the built structure) goes automatic compared to a digital model where this dataflow goes manually. This digital twin offers an opportunity to simulate and optimize the production system (Kritzinger et al., 2018) and minimalizes the manual tasks leaving as less room as possible for manual errors.

(a) Representation of digital model (Kritzinger et al., 2018)

(b) Representation of digital twin (Kritzinger et al., 2018)

Figure 3.55: Digital model and digital twin

Chapter 4

Final tests and results

The framework, that was set up and explained in Chapter 3, is tested as a whole to gain results of this framework in terms of data and draw conclusions on its working. In the first section of this chapter (Section 4.1) the set-up of this final test to test the whole framework is explained. After the explained set-up of the test, the results are discussed in Section 4.2. Finally, this chapter contains the final tests of the framework where the automation steps are taken into account (Section 4.3) and bigger structures are tested to see the full working of the framework.

4.1 Test set-up

The test set-up contains parts of the set-ups researched and explained in Chapter 3. Within this test set-up not all the automation steps are incorporated since these tests had the goal to collect data in order to draw conclusions on the measuring technique and other specific parts of the framework. Section 4.3 explains the final tests with the automation steps incorporated.
Figure 4.1 shows the total set-up of the final test. The separate elements of this test set-up will be explained individually.

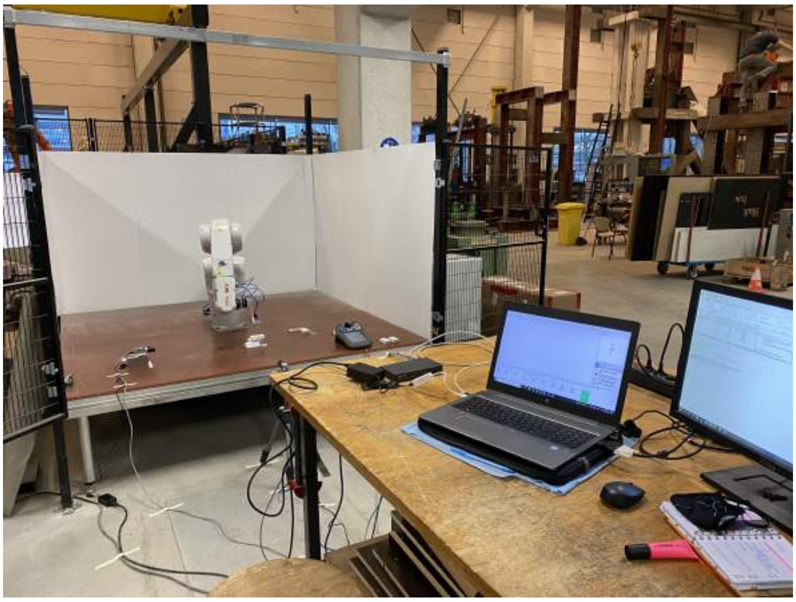


Figure 4.1: Total set-up of the final test

Figures 4.2a and 4.2b show both the cameras that are used during this final test. These cameras are both a Basler ace AC4600-7gc camera (Basler, nd) and this camera was first used with the second test elaborated in Section 3.4.2. Both the cameras are connected with two wires. One of these wires gives power to the camera's. The second wire is a data wire that is used to make a connection to a laptop.

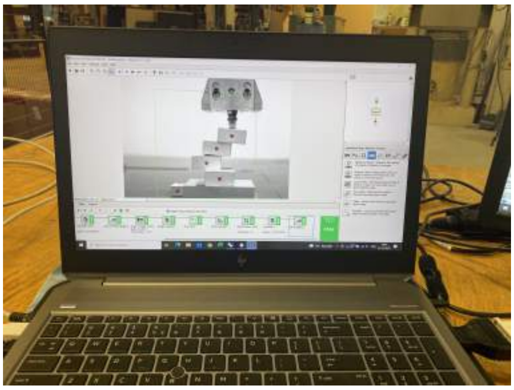
(a) Camera 1: front view

(b) Camera 2: side view

Figure 4.2: Set-up of cameras for the final test

Three programs are needed for this final test. The first program is Grasshopper (shown in Figure 4.3a). Grasshopper is used to generate the rapid code for the robot. Vision builder is the second program that is used for this final test, shown in Figure 4.3b.

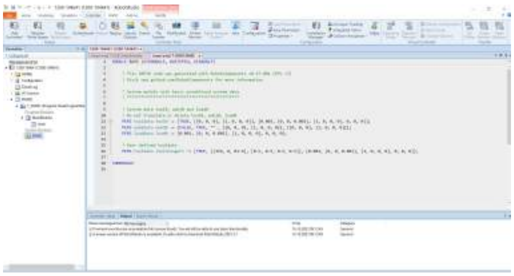
(a) Grasshopper screen



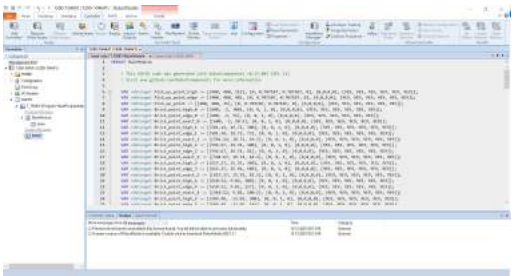
(b) Visionbuilder screen

Figure 4.3: Set-up of programs for the final test

Next to Grasshopper and Vision Builder, Robotstudio is used for this final test. Robotstudio is used to bring the rapid code, generated in Grasshopper to the robot itself. Two pieces of code are used for this, first the base code is used (shown in Figure 4.4a). This code contains information about the robot and the robot end-effector. This code only needs to be updated once, in the beginning of the test, since this code does not change when another structure is built. The second used code is the main code (shown in Figure 4.4b). This code contains the information on where the bricks need to be picked up and where they should be placed. This code therefore changes during the test for the multiple tested structures. This code will also be updated when an adjustment to the brick that is yet to be placed is made.



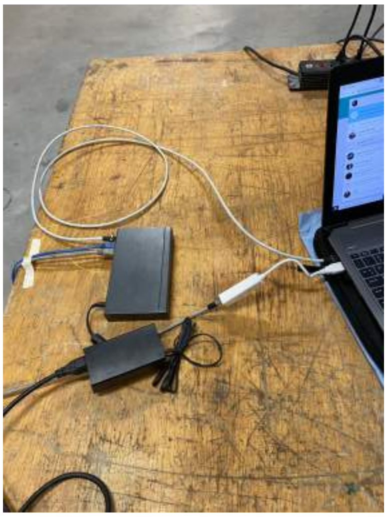
(a) Robotstudio: base code



(b) Robotstudio: main code

Figure 4.4: Set-up of robotstudio for the final test

Next to the cameras and the programs, some additional wiring is required to conduct this final test. The power and data wires of the cameras where already explained and shown in Figures 4.2a and 4.2b. The data wires of the two cameras come together in a switch (shown in Figure 4.5a). From this switch one wire goes into the computer to load the views of the cameras and use them in Vision Builder. Another new wire is connected to the laptop, this wire is used to make a connection directly to the robot (shown in Figure 4.5b).

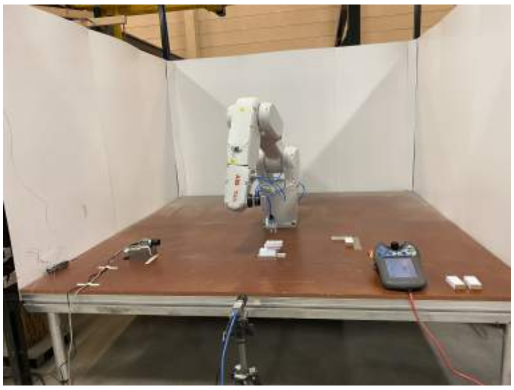


(a) Wiring of cameras to switch

(b) Wiring of robot

Figure 4.5: Needed wiring for the final test

When all the programs, wires and cameras are set up. The final test can be conducted. Figure 4.6a shows the test itself when one of the structures is built. Figure 4.6b shows this structure.



(a) Final test frontview

(b) Built structure

Figure 4.6: Final test

4.2 Data results

The results of this final test are described in the same way as the results of the tests described in Section 3.4. For both the X and the Y displacements a set of four tables is given. The first table shows the coordinates of the centre of masses in millimeters of the built model, which is the measured data. The second table gives the same information on the centre of masses in millimeters but now of the digital model. The third table shows the differences between the first and the second table in millimeters where a negative number means that the built model is more to the right compared to the digital model and a positive number shown the opposite, so the built structure is more to the left compared to the digital model. The last table in this set of four shows the differences of the third table but now in percentages normalized to the length of the brick. The length of the brick that is used for this normalization is the length in which the displacement is measured, so for the X-displacement the long side of the brick is used and for the Y-displacement the short side of the brick is used.
The amplitudes for both the X-direction and the Y-direction were chosen in a way that the steps between the amplitudes are the same and three of the five cases had no failed checks at the structural analysis. The last two of the five cases did have failed checks, these cases were used to see if the structures would actually collapse and if the model also concluded so after the measurements were done. The amplitudes were changed in comparison to the amplitudes used in Section 3.4 since the bricks used for the final tests are bigger bricks and therefore the previous amplitudes would not move the bigger bricks to the edges of the limit. Every amplitude was built 15 times so the numbers shown in the tables are the averages of 15 tests of structures with the same amplitudes. This amount of tests over which the average is taken is bigger than for the tests before in Section 3.4 making the data of these final tests more reliable.
The first set of four tables shows the data from the displacements in the X-direction. The placements of the first bricks are determined with the difference bricks that are placed next to the place where the structure is built (Figure 4.6b).

	Final test data X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
1,00	-0,32	16,96	26,15	23,31	9,13	-9,25
1,25	-0,32	21,07	32,52	29,12	11,37	-11,53
1,50	-0,17	25,11	38,86	34,74	13,43	-13,90
1,75	0,40	29,22	45,23	40,60	-	-
2,00	0,41	33,34	51,70	-	-	-

Table 4.1: Final test data centre of masses in mm (X-direction)

	Digital model X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
1,00	0	16,57	25,39	22,33	8,82	-8,82
1,25	0	20,72	31,74	27,91	11,02	-11,02
1,50	0	24,86	38,09	33,49	13,23	-13,23
1,75	0	29,00	44,44	39,08	-	-
2,00	0	33,15	50,78	-	-	-

Table 4.2: Digital model data centre of masses in mm (X-direction)

	Differences in mm X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
1,00	0,32	-0,39	-0,76	-0,98	-0,32	0,43
1,25	0,32	-0,35	-0,78	-1,21	-0,35	0,51
1,50	0,17	-0,25	-0,77	-1,25	-0,20	0,67
1,75	-0,40	-0,22	-0,80	-1,52	-	-
2,00	-0,41	-0,23	-0,91	-	-	-

Table 4.3: Comparison final test and digital model data in mm (X-direction)

From Table 4.3 it can be seen that most of the differences between the built and digital model are below one millimeter. The only differences that are bigger than one millimeter are with the placement of the fourth brick. When looking at the differences per brick, not looking at the different amplitudes, the differences are all in the same order of magnitude, meaning that the differences of the third brick for example for all the amplitudes are between -0,76 and -0,91. These margins within one brick are quite small. Another observation from the table shows that most of the numbers are negative which means that the built brick is placed more to the right than the digital brick and than it should be. In this case, when also looking at Figure 4.7, this is the side where the brick will be more likely to tip over, the more unsafe side. The positive numbers of brick 6 are also more on the unsafe side since the structure of bricks moves to the left side there. Figure 4.7 can be used in combination with Table 4.4. It shows the digital (blue) and the built model (red) of the structure with an X-amplitude of 1,50. With this visualization it can be seen that there are no big differences between the models since the colours in the visualization overlap a lot. There are some small parts where there is only blue or only red visible and that are the deviations between the models. For example. the fourth brick shows a difference between the two models, when looking at Table 4.4 this is a difference of -1,62%. With all the other percentages in Table 4.4 an idea can be gained on how the built structure looks like in comparison to the digital structure.

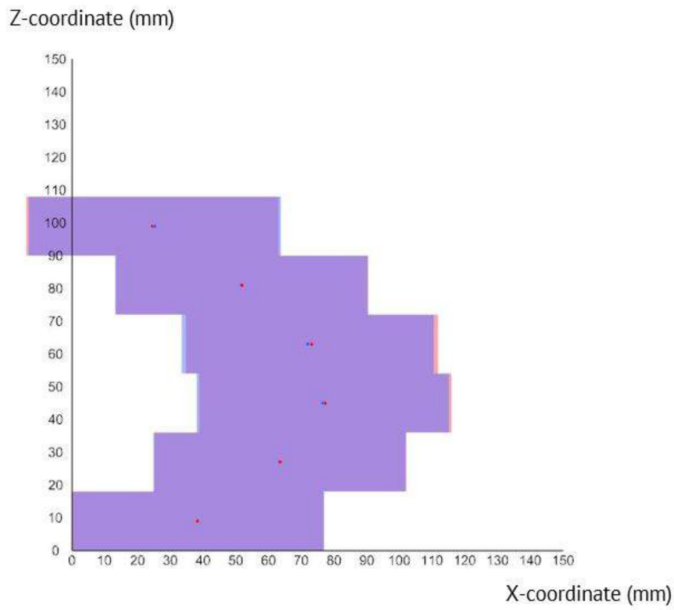


Figure 4.7: Visualization of the digital (blue) and the built structure (red) with X-amplitude 1,50

	Differences in percentage (%) X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
1,00	0,42%	-0,50%	-0,98%	-1,27%	-0,41%	0,56%
1,25	0,41%	-0.45%	-1,01%	-1,57&	-0,45%	0,66%
1,50	0,22%	-0,33%	-1,00%	-1,62%	-0,26%	0,87%
1,75	-0,52%	-0,28%	-1,04%	-1,98%	-	-
2,00	-0,53%	-0,30%	-1,19%	-	-	-

Table 4.4: Comparison final test and digital model data in % (X-direction)

Figure 4.8 is added to explain a trend in the differences between the digital and built model from the structure with an amplitude of 1,50 in the X-direction. This figure shows the difference between the two models in percentage relative to the height (which corresponds to the number of bricks added to the structure). From this figure it can be seen that the biggest differences are with the bricks in the middle of the structure. When looking back at Figure 4.7 the line corresponds with the bricks that are most placed to the edges of the structure.

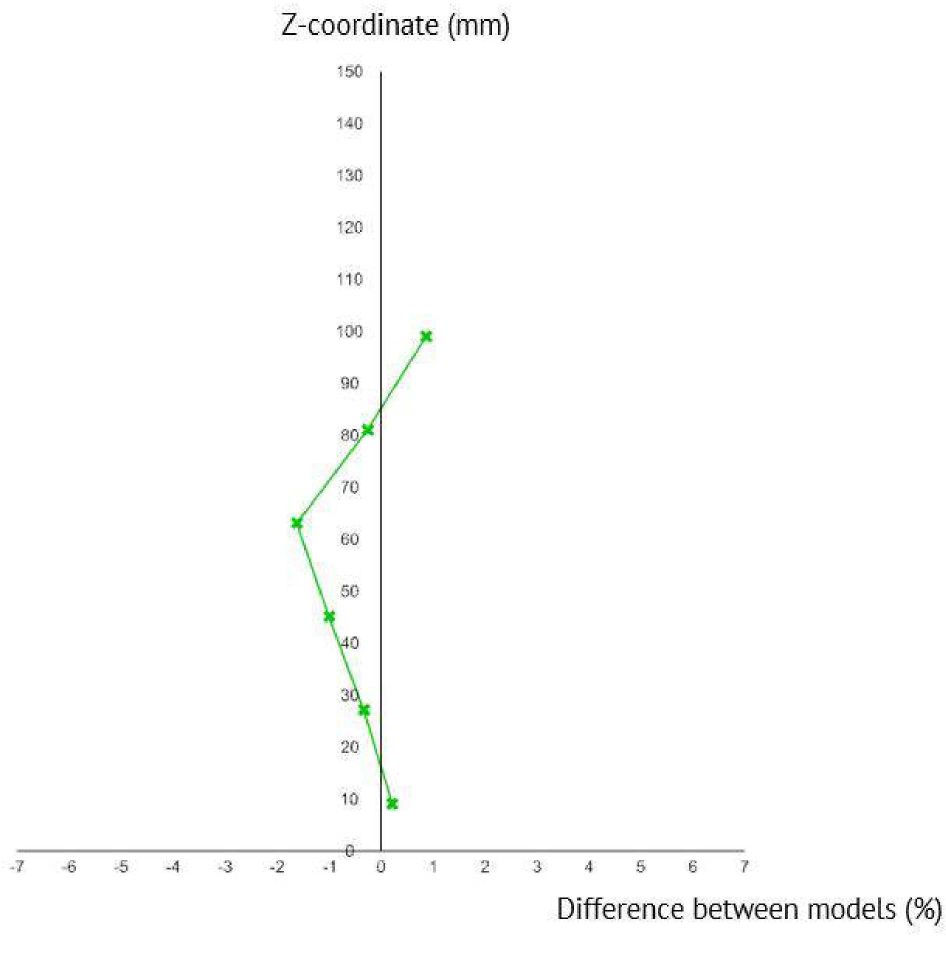


Figure 4.8: Trend of the differences between the two models in the X-direction

The next set of four tables show the data from the final test of the displacements in the Y-direction.

	Final test data Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0,07	-4,81	-0,84	6,15	4,18	-4,29
0,75	-0,03	-7,53	-1,57	8,65	6,17	-5,90
1,00	0,12	-10,11	-2,27	10,85	7,70	-7,56
1,25	-0,12	-12,84	-2,75	13,52	-	-
1,50	-0,05	-15,94	-3,40	15,24	-	-

Table 4.5: Final test data centre of masses in mm (Y-direction)

	Digital model Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	0	-6,34	-2,10	5,50	4,01	-4,01
0,75	0	-9,48	-3,01	8,12	5,82	-5,82
1,00	0	-12,60	-3,84	10,61	7,48	-7,48
1,25	0	-16,70	-4,60	12,99	-	-
1,50	0	-18,76	-5,33	15,27	-	-

Table 4.6: Digital model data centre of masses in mm (Y-direction)

	Differences in mm Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	-0,07	-1,53	-1,27	-0,65	-0,18	0,28
0,75	0,03	-1,95	-1,44	-0,53	-0,36	0,086
1,00	-0,12	-2,50	-1,57	-0,24	-0,22	0,08
1,25	0,12	-2,86	-1,86	-0,53	-	-
1,50	0,05	-2,81	-1,94	0,03	-	-

Table 4.7: Comparison final test and digital model data in mm (Y-direction)

From Table 4.7 it becomes clear that the differences between the digital and the built model for the Y-direction are bigger than the differences seen for the X-direction in Table 4.3. Possible reasons for this will be explained in Chapter 6. When looking solely at the differences between the two models for the Y-direction the same applies as for the differences between the two models for the X-direction, namely, that the differences between the two models when looking at the same numbered brick are in the same range. An example of this is the third brick since for all the amplitudes the differences between the two models are between -1,27 and -1,94 mm. Another observation from the table shows that most of the numbers are negative which means that the built brick is placed more to the right than the digital brick. This also becomes visible in Figure 4.9. Since the Y-direction of the structure moves differently than the X-direction, the deviation in this case does not necessarily mean that the brick is placed more to the unsafe side. Figure 4.9 shows to which side the actual model is built in comparison to the digital model.
Figure 4.9 can be used in combination with Table 4.8. Figure 4.9 shows the digital (blue) and the built model (red) of the structure with an Y-amplitude of 1,00.
With this visualization it can be seen that there are bigger and more differences between the models compared to the differences in the X-direction since there is more red and blue visible in this visualization. This visualization can be used in the same was as the previous visualization in combination with Table 4.8 since, for example, the second brick shows a difference of -6,75% and it can be imagined how the bricks of the other amplitudes are placed compared to this brick.

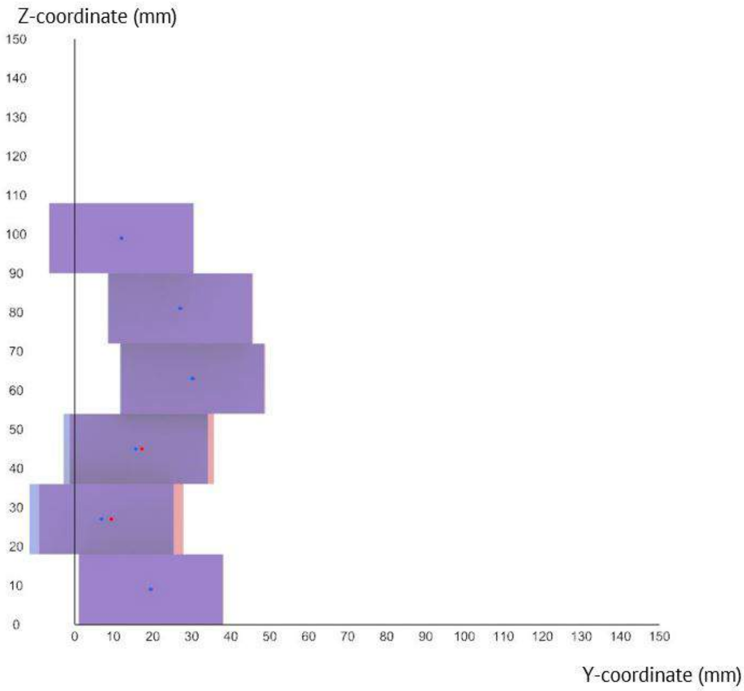


Figure 4.9: Visualization of the digital (blue) and the built structure (red) of Y-amplitude 1,00

	Differences in percentage (%) Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	-0,20%	-4,12%	-3,42%	-1,75%	-0,47%	0,77%
0,75	0,09%	-5,27%	-3,89%	-1,44%	-0,96%	0,23%
1,00	-0,33%	-6,75%	-4,24%	-0,66%	-0,60%	0,21%
1,25	0,31%	-7,73%	-5,03%	-1,44%	-	-
1,50	0,13%	-7,61%	-5,24%	0,09%	-	-

Table 4.8: Comparison between final test and digital model data in % (Y-direction)

Again a trend figure is added to show the trend in the differences between the digital and the built model (Figure 4.10). This figure is made with the same amplitude as Figure 4.9 namely Y-amplitude 1,00. This figure shows the difference between the two models in percentage relative to the height (which corresponds to the number of bricks added to the structure). From this figure it can be seen that the biggest differences are with the bricks in the lower to middle part of the structure. When looking back at figure 4.9 the trend line corresponds with the bricks that are placed most to the left edges of the structure.
From all these results it can be concluded that there were still some deviations between the digital and the built model present. These deviations could have multiple causes. These causes are explained and discussed in Chapter 6.

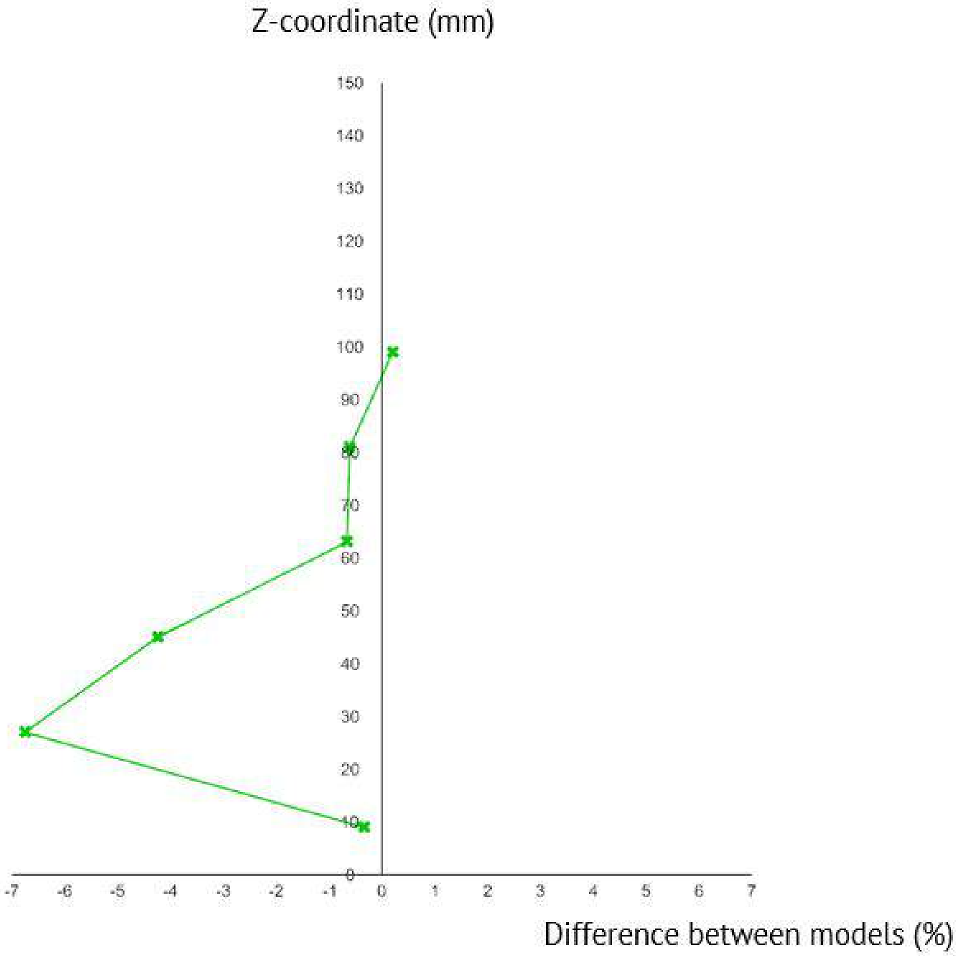


Figure 4.10: Trend of the differences between the two models in the Y-direction

4.3 Full framework with automation steps

Next to the final tests to gather data on the built structures and draw conclusions on the measuring technique of the framework, some final tests were run with all the automation steps combined into one test to see the whole framework at once. To test the whole framework, the same test set-up as explained in Section 4.1 was used. Some small additions were made due to the automation steps which were already explained in more detail in Section 3.7. In the end, all the separate automation steps worked. However, when combining all of these steps to the model at the same time the Grasshopper model crashed instantly after adding the remote connection of Robot Components to the data flow of the data processing part. To prevent this from happening three data dam components were added into the script. A data dam component can temporarily prevent the flow of data from the left side of the component to the right until it is activated. This prevented the whole model from crashing even though the automation steps could all be added to the same model at the same time.

During this test four different cases were tested, all with a height of 10 bricks. Also, two different types of structures were tested, the first two being based on the sine that was used for all the other tests during this research and the second being a leaning tower where the bricks are placed on a line that was formed between a fixed beginning point and a parametric end point. The line over which the bricks were placed in the second design was straight itself. So, two cases where tested per design: one of which is where all the structural checks suffice before the built, and one of which where not all the structural checks suffice before or during the built.

Test case 1: sine tower with all sufficient checks
For this first test case a tower just like all the other towers in this research based on a sine was taken. An X-amplitude of 1,00 and an Y-amplitude of 0,5 were used in combination with a height of 10 bricks to form this structure. This structure gave no failed checks. Also, no large deviations were measured during the built itself resulting in no changes in the coordinates of the bricks that were not placed yet. Therefore, this structure was built as a whole without any adjustments during the build of this structure. Also all the automation steps worked well besides the mentioned data dams and the clicks that were needed to let the flow of data pass through.

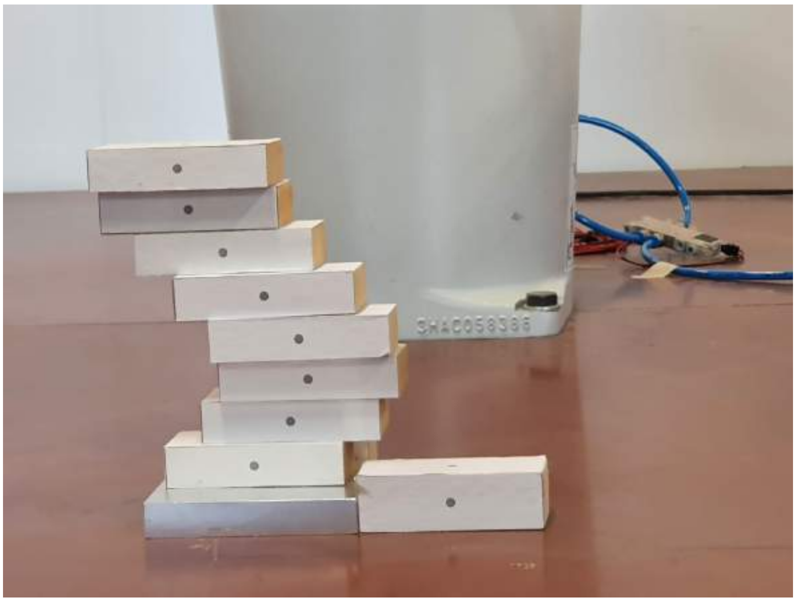


Figure 4.11: Sine structure of test case 1

Test case 2: sine tower with failed checks
This test case started with a sine tower of 10 bricks with an X-amplitude of 1,5 and a Y-amplitude of 1,00. Since 10 of the structural checks were not satisfactory, another initial design was needed before the built could start. This design was made with the optimization tool of the framework. The optimization tool (made with the Grasshopper plug-in Octopus) looked for a structure as close as possible to the initial structure with the failed checks but now with all sufficient checks (Figures 4.12a and 4.12b). With this new optimized structure without failed structural checks the built could start. However, since this structure was optimized to a structure still being on the edge of collapse there were some adjustments needed during the built because there were some deviations between the models measured. For this test case, the data conclusion part looked at the whole remaining structure after the built of a few bricks. Since there were multiple failed checks, the framework asked for a second run with the optimization tool to determine new coordinates for the bricks that are not placed yet. However, with all the equipment attached to the computer, the optimization would not start and could therefore not run. This happened multiple times and only when the equipment was all connected. What could be concluded from this test is that the optimization tool works and that there were deviations between the two models measured. However, the combination of all the equipment and the optimization tool could not be run and therefore this tower could not be finished.

(a) Front view of optimized structure (green is optimized)

(b) Side view of optimized structure (green is optimized)

Figure 4.12: Optimization of test case 2

Test case 3: leaning tower with all sufficient checks
This test case was almost the same as test case 1 with the only difference being the design of the structure. For this structure an X-coordinate of -50 and a Y-coordinate of 25 were taken together with the height of 10 bricks. This design gave no failed checks for the structural analysis before the built. During the built itself no large deviations were measured to the digital initial design and therefore this structure could be build according to the initial digital design (Figure 4.13).

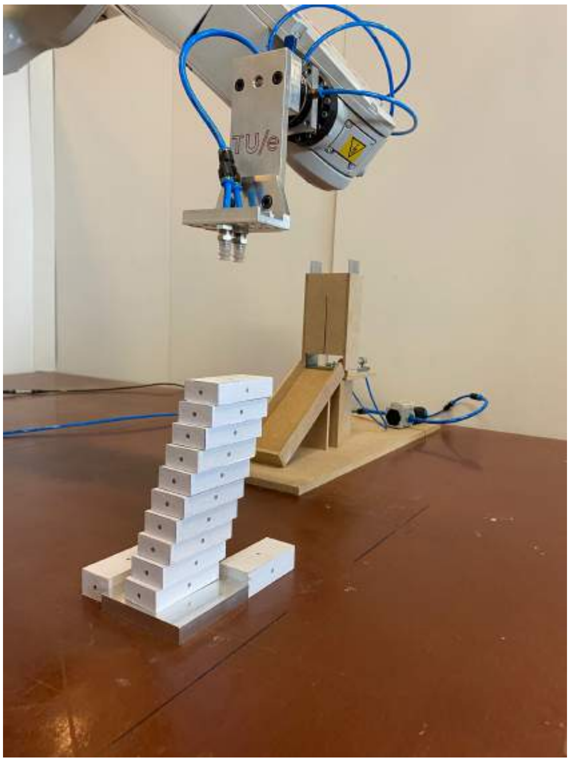


Figure 4.13: Leaning tower of test case 3

Test case 4: leaning tower, adjustments during building
For this test case the same initial structure was taken as a starting point as for test case 3. However, during the built a deviation between the digital and the built model was forced by moving one brick (the third brick) that was placed by the robot to the other side. By doing this a possible insecurity was forced. This was measured by the measuring technique, resulting into needed adjustments to prevent a collapse of the structure since there were failed checks at the structural analysis part. These adjustments were made per brick and this went well for a few bricks. However, when the structure was at a height of 6 bricks and a new adjustment was proposed by the model (in the same way as the ones before this brick were calculated), the structure collapsed. There are multiple explanations on why this could have happened. It could be that the margin component is too small and not accurate enough (explained further in Chapter 6). It could also be that the shaking of the table, which did happened during this test, had to do with this collapse. What can be concluded from this test case was that the measuring technique did notice the large deviation between the digital model and the built model and that it concluded to alter the coordinates of the next bricks. However, at some point this alteration was clearly not enough resulting in the collapse of the tower.

Figure 4.14: Leaning tower of test case 4

Chapter 5

Conclusion

In order to draw conclusions from this research, the main research question and the sub-research questions introduced in Section 1 are answered starting with the sub-research questions and ending with the main research question.

1. Which geometric uncertainties can occur during the robotic manufacturing process and therefore influence the structural behaviour of the brick structure?

From literature several possible geometric uncertainties were found, namely: the leveling of the ground, vibrations, wind, precipitation, manufacturing tasks by people, part tolerances and manufacturing tasks of the robot. Two additional possible geometric uncertainties were added to Table 2.1 being the deviation of the robot end-effector and the impact of the placement of elements since these were found during the research to play a role. These uncertainties could all lead to wrongly placed elements, movement of the elements and unstable connections between the elements which is not desirable during a manufacturing process.

2. How can the results of the geometric uncertainties of a robotic manufacturing process be measured?

Two measuring techniques were tested during this research, namely, a distance sensor and cameras in combination with Vision Builder. The latter of the two showed the most potential and was therefore explored further within this research. For this research this measuring technique showed to be suitable enough to finalize the framework and retrieve data on the built structures. However, this measuring technique also showed some discussion points and question marks if this would be suitable for bigger projects and more complex structures (Chapter 6). These discussion points mainly focused on the uncertainties and limitations of this measuring technique such as the importance of the levelling of the camera, the fact that elements behind each other cannot be captured and the uncertainty of usable data on a building site.

3. How can adjustments be made to the robotic manufacturing process to reduce structural consequences due to geometric uncertainties?

By measuring the built structure and processing the measured data it becomes possible to see if the next bricks can be placed according to the initial design or not. By adjusting the structure, when needed during building, the manufacturing process does not have to be stopped and start over again leading to the production of waste.

The third sub research question was completed with the following three smaller questions:

3a. How can the results of the measured geometric uncertainties be processed and compared to the initial digital model?

A combination with Vision Builder, Excel and Grasshopper was made to compare the built model to the initial digital model. From the data gathered from the pictures in Vision Builder, an Excel file was made of the coordinates of the introduced marks on the bricks. This resulted into two Excel files, one for each camera. These Excel files were imported into Grasshopper by the use of a small Python code. Once the data was in Grasshopper a digital twin was made of the built structure by combining the data from the two Excel files and projecting bricks on the measured centre points. With this digital twin of the built structure, a direct comparison could be made to the same part of the initially designed structure leading to possible differences in centre points and corner points in millimeters.

3b. To what extend will the structural behaviour of the end-product be influenced by the geometric uncertainties?

When the digital twin of the built structure is made a new structural analysis can be run to see if the structure can be built further. A distinction here was made between two approaches, the first being the addition of only one new brick to see if this brick leads to a collapse or not. If this new brick leads to a collapse of the structure the coordinates of this brick can be changed before the brick is placed leading to an adjusted structure. The second approach involves the whole remaining structure instead of just the next brick. When the whole remaining structure is analysed in combination with the already built bricks, again a structural analysis could be run to see if there are any failed checks and therefore possible collapses. If this is the case, an optimization tool can be run. This optimization tool adjusts the whole remaining structure at once and the built bricks will not be changed, to get a structure that has no failed checks and is as close to the initial design as possible. With this latter approach not only the next brick is taken into account but the whole structure making it possible to account for adjustments to come and adjusting the structure to something that is more controlled than adjusting every next brick.

3c. What are possible adjustments to the structural element?

The possible adjustment introduced in this framework is the adjustment of the brick or bricks that are next to be placed. This was introduced since these bricks are not yet placed and therefore the coordinates of these bricks could still be adjusted before the robot places the new bricks. This approach makes it possible to keep building and not needing to start over when something is built wrong leading to minimal waste. However, there are many more possible solutions to make adjustments during a manufacturing process which are introduced Section 6.4.

With all the answers to the sub research questions, the main research question can be answered as well.

How can geometric uncertainties of a robotic building process be dealt with within the design and manufacturing process of a brick structure, to enlarge the application of robotic manufacturing within the built environment?

By the means of this framework where the structure is designed, structurally analysed and measured during building, it becomes possible to use the measured data and conclude if the manufacturing process can still continue according to the initial design or not. If there is a brick that is not placed correctly the framework will notice this and give new coordinates for the next bricks to build further. This framework was automated so that the manufacturing process does not have to be stopped to run the framework and analyse the data. This framework takes into account the fact that geometric deviations due to geometric uncertainties during the manufacturing process can happen and finds a way to build further and not having to start over with the whole building process. The design process of this framework still has potential to grow in terms of taking into account the robotic manufacturing process early in the design phase with possible boundary conditions due to the specific manufacturing process.
In order to enlarge the application of robotic manufacturing within the built environment it is important to acknowledge possible uncertainties and find a way to cope with them. With the design and the steps of this framework it becomes possible to have more certainty when working with a robotic manufacturing process since the possible geometric deviations are analysed during the manufacturing process itself. The specific completion of all the separate steps of the framework may vary per case and per situation but the design is overall applicable to gain more certainty within a manufacturing process. This framework is a first step into the direction of a broader application of robotic manufacturing where real-time measured data is the key.

Chapter 6

Discussion

In this chapter all the points that are up for discussion considering this research will be elaborated. The discussion will be split into three parts: first the discussion on the results of the final test (explained in Chapter 4), second the discussion points on this specific version of the framework that was used to conduct the final tests and third some general discussion points related to a framework described in this research.

6.1 Discussion final test results

The first part of the discussion is regarding the results of the final test (Section 4.2). The differences between the digital and built models from the final test are in some cases quit large when looking at the percentages. These percentages range, for the X-direction, from -1,19% tot 0,87%. These percentages for the differences in the X-direction are not yet that large. However, the percentages, for the Y-direction, vary between -7,61% and 0,77% of the length of the block in the Y-direction. Especially the lower bound of this range is significant. There are multiple possible explanations for these differences between the digital and the built model. These possible explanations are explained in the following bullet points together with the likeliness that they contribute to the differences.

- *Part tolerances*
  It could be the case that the bricks are not exactly the same size and therefore part tolerances occur. Since these bricks are all made in one go with the same machines they look all the same size. The bricks are also measured with an electrical caliper where small deviations in the bricks were found. However, these tolerances are small and when using these more exact measurements of the bricks in the Grasshopper model, the centre of masses of the bricks were still all the same. This means that these small deviations in measurements of the bricks are negligible in these results. The bricks are also stiff and in this case it is assumed that they do not deform under the weight of the added bricks.

- *Mark tolerances*
  The round markers set on the bricks are placed in these positions by hand. Even though the markers are printed on a piece of paper the exact size of the brick, these papers are still put to the brick by hand meaning that these could slightly vary over the bricks. This is also quit difficult to measure compared to the measurements of the bricks were just an electrical caliper can be used. Since the electrical caliper cannot be put around something to measure the exact placement of the marker these measurements are not accurate enough. It could therefore be the case that the markers are not placed in the exact middle point of each side of the brick for all the bricks.

- *Leveling of the table*
  The table on which the structure is built is not entirely straight in some places due to the weight of the robot that is placed in the middle. However, to prevent these imperfections of the table from having an influence on the structure, a stiff small plate was added and secured to the table on which the structure is build. In this way the structure is build on the same plate at the same location every time and the imperfections of the table are not of influence on the differences between the digital and the built model.

- *Leveling of the camera*
  The leveling of the camera became a more important cause to the differences between the digital and built model than anticipated earlier in the process. Figure 6.1 shows how one of the pictures taken by one of the cameras looks when this camera is tilted a bit. The influence of this can be explained with a small Grasshopper model (Figures 6.2 and 6.3). Figure 6.2 shows a normal situation where the camera is entirely straight. Figure 6.3 shows a situation where the view is tilted. The middle points of the bricks are analysed and there is a difference between the two measurements of 0,5 millimeters in this case. This difference can become quite big if the brick is more to the outside of the camera view since the rotation is bigger compared to the middle of the camera view. This could explain some of the differences mentioned in Section 4.2 where the bricks that were placed more to the side that had bigger differences to the initial structure.

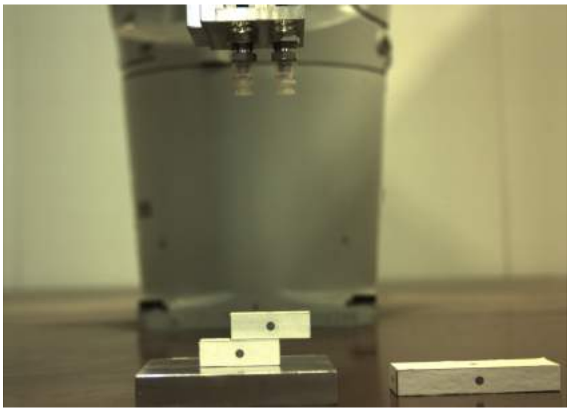


Figure 6.1: Tilted camera view

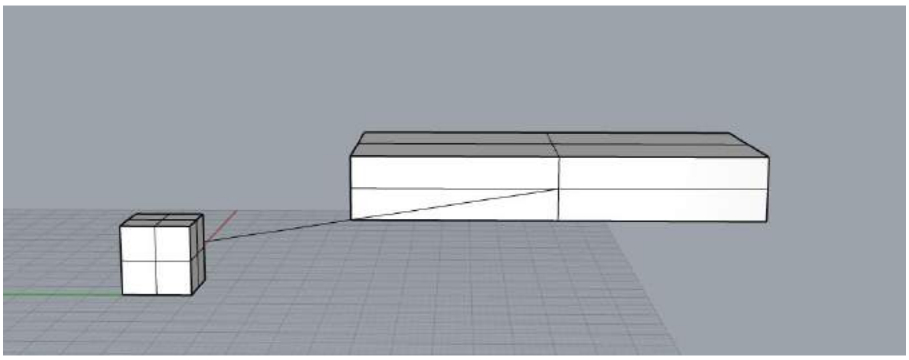


Figure 6.2: Grasshopper model normal camera view

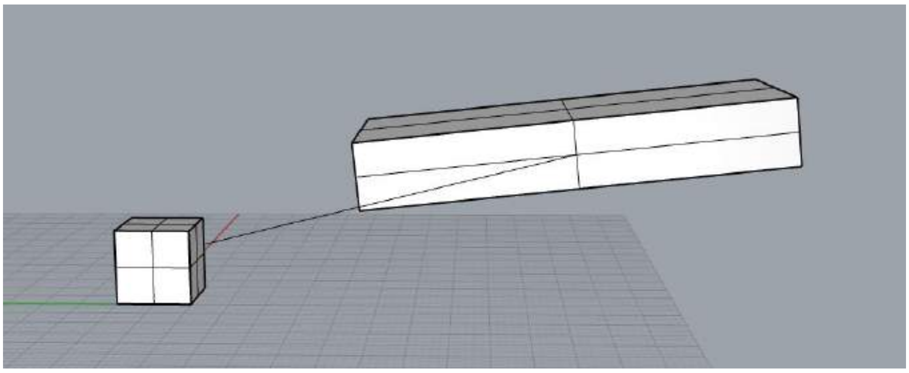


Figure 6.3: Grasshopper model tilted camera view

There is something that can be tried to prevent this leveling of the camera from happening for future projects with the measuring technique. With a small addition of a little stripe to the mark on the brick the angle of the selected mark can be measured in Vision Builder. This small feature could be used to get a better view on the placement of the cameras and how straight their view is. However, then the question arises what to do when this measured angle in Vision Builder is not entirely straight or cannot be placed 100 percent straight? What should then be done with the test result?

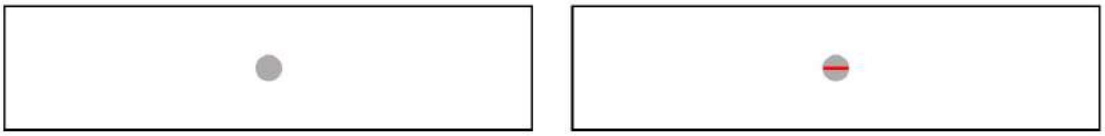


Figure 6.4: Possible improvements for marks on bricks

- *Calibration factor*
  The calibration factor that is used inside Vision Builder to calculate the centre points of the markers in comparison to the whole picture is calculated by the length of the bricks measured by Vision Builder in pixels divided by the actual length of the brick in millimeters. This is done twice before the actual measurements start to calibrate the calibration factor. However, when doing this multiple times the length of the brick in pixels kept shifting with every new picture that was taken. This gave small deviations when the calibration factor was calculated again. This kept on happening and therefore it could be the case that there is a small deviation between the actual centre point of the mark and the measured centre point of the mark.

- *Location pick-up point*
  The location of the pick-up point could be a cause to the differences between the digital and built model. However, the pick-up point was in this case fixed to the table on which the structure was built and the vacuum tool picked the bricks in the same pick-up point. There is a small amount of movement possible within the pick-up point otherwise the brick will be stuck inside the pick-up point. However, this possible movement is smaller than 0,1 millimeters making it negligible for this moment. If the total pick-up point was not entirely in the correct location, this error was transferred to all the bricks making the error non existing when the centre of masses of the bricks are calculated since the distances between

the centre of masses remain the same. Therefore, the contribution of the pick-up point to the differences between the models can be neglected.

- *Vibrations caused by the robot*
  The vibrations caused by the robot could cause some differences between the digital and the built model since the bricks can start shaking due to the vibrations and therefore shift slightly in their location. This seemed a problem by one of the tests 3.4.4, but this cause was eliminated by altering the robot speed and placing the structure in a location that is further away from the robot. However, during the last tests, the vibrations were visible again in the higher regions of the structures (from the seventh brick and higher). It can therefore not be assumed that these vibrations are not a cause of the differences between the digital and the built model.

- *Imperfect placement of the robot*
  It could be the case that the robot places the bricks not on the correct spot and that this differs per brick. However, it is unsure how much this contributes to the total differences between the digital and built model since this particular cause was not researched individually. However, this cause should be kept in mind to cause potential differences between the both models.

Concluding, the most likely causes to the differences between the digital model and the built model are the leveling of the camera, mark tolerances, calibration factor, vibrations and the imperfect placement of the robot.
Besides the difference between the digital and the built model from the last test there are also some differences between the last test with the big bricks (Chapter 4) and the last test with the smaller bricks (Section 3.4.4). The differences between the digital and the built models for these tests are as follows:

- Small bricks in X-direction between -0,24 and 0,46 millimeters
- Small bricks in Y-direction between -1,36 and 0,59 millimeters
- Big bricks in X-direction between -1,50 and 0,67 millimeters
- Big bricks in Y-direction between -2,86 and 0,28 millimeters

All the aforementioned causes could have an influence in these differences between the small and big bricks since new bricks were made with new markers, the test was conducted on another day so the calibration needed to be done again and the cameras needed to be placed again. The vibrations and the imperfect placement of the robot are not necessarily the cause of a new test. The differences between the two models are mainly bigger for the bigger bricks which could be explained again by the leveling of the camera since the markers on the big bricks are more to the outside of the camera view and could cause therefore some differences between the models when the camera is tilted.

6.2 Discussion specific framework

Design of structure
For this research the design that is used within the framework was kept quit simple. This design, of a tower of bricks stacked on top of each other, was not expanded further in complexity other than the height of the tower, and therefore the amount of used bricks, and the amplitude of the tower in two directions. The design was kept in this way throughout this research. The main reason for this was that the focus of this research was laid on the whole framework and completing all the needed parts for this. This meant that the time spent on this project was divided between the parts of the framework. Next to that, during the final tests it was concluded that the used measuring technique is suitable for this design of a tower but not for more complex designs where bricks are placed behind each other such as a wall (Figure 6.5). This is not possible in this case because if bricks are placed behind each other they are not visible anymore in the chosen camera views and can therefore not be analysed. Therefore, also during the final test, it was chosen to use this same design principle and show the working of the framework based on this design.

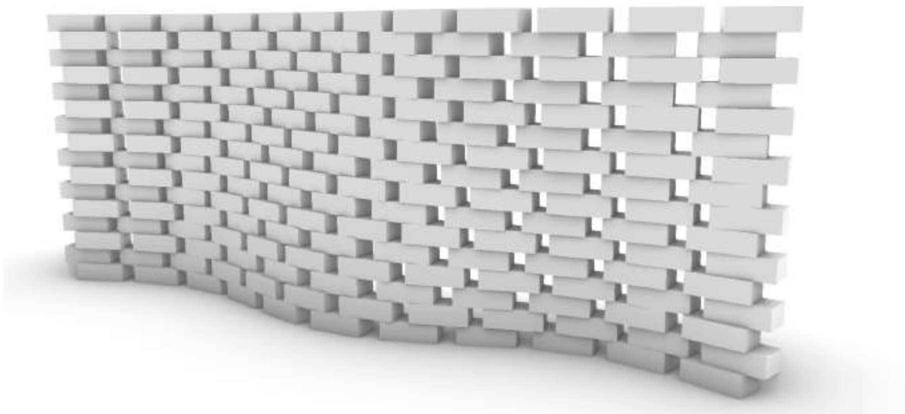


Figure 6.5: Structure of a wall

Structural analysis
The structural analysis that is now used in the framework is based on the principle of centre of masses. This structural analysis was chosen from the beginning of the research since this principle matches with the design of a simple tower. This structural analysis gave enough information about the structural behaviour of this design to continue with the framework and make conclusions on whether or not the structure would collapse. To leave enough time for the other aspects and parts of the framework, it was chosen to stick with this structural analysis. Also, since at a certain point in the process it became clear that the design principle would stay the same, it was not needed to expand the structural analysis. However, it is not obvious that this structural analysis principle would work for other kinds of structures are well. If, for example, a wall structure would be used, this structural analysis based on centre of masses would not fulfill the requirements since it is not suitable for this wall type of structure due to the fact that it only takes into account one brick support. Also, the structural analysis based on centre of masses does not take the total behaviour of the structure into account since principles such as stresses and friction are left out. It is also now the case that there are no load cases or additional forces added to this structural analysis which should be considered for the future when there are structures designed that will actually be built. For this proof of concept of the framework this is not a limitation since the centre of masses analysis describes the structural behaviour good enough to draw conclusions on the behaviour. However, when more in-depth information is required from this structural analysis or the structural analysis does not work well with another structural principle, this part of the framework should be extended or replaced according to the requirements.

Robot end-effector
The robot end-effector that was used for this research only had the opportunity to hold one single tool meaning, either a sensor, or a vacuumtool in the case of this project. For this first set-up of the framework this was not a limitation since this was not required for the used measuring technique. However, if the combination of multiple parts on the robot end-effector would be possible, more options, especially for the measuring technique, could be tested since a sensor could be connected to the robot end-effector at the same time as the vacuumtool. Therefore, this is also one of the recommendations for further research since a lot is still to gain in this area especially since the combination of multiple tools on the robot end-effector could lead to a better measuring technique where for example rotations could be taken into account.

Measuring technique
It was noticeable throughout the whole research that only displacements in X and Y direction were measured and possible rotations of the bricks were left out of the equation. The main reason for this is that this measuring technique with two cameras in this test set-up cannot measure the rotations. This could be improved by adding a third camera that takes a view from above or another type of camera/sensor. However, such an improvement and alteration to the test set-up would require more research into the test set-up and specific measuring technique. For this research it was chosen to use the measuring technique that uses two cameras in combination with Vision Builder to conduct multiple tests and work on the other parts of the framework as well. However, this does leave room for further research into measuring techniques that could also be more applicable on an actual building site or in a factory. It could be possible to combine a measuring technique with the vacuum tool on the robot end-effector.
One test was conducted with another sensor to see the potential of this measuring technique. This sensor is a 3D scanner (Figure 6.6) that measures profiles which is used also in combination with a 3D concrete printer. However, as already explained in the previous discussion point, it was not an option at this moment of this research to combine this scanner with the vacuumtool on the same robot end-effector. This measuring technique does have potential since some of the disadvantages of the cameras are not there anymore such as the leveling of the camera and the risk that the camera is touched on a building site. This would therefore be a good starting point for further research to see what kind of data can be retrieved from this measuring technique and how this data can then be processed as done in this framework.

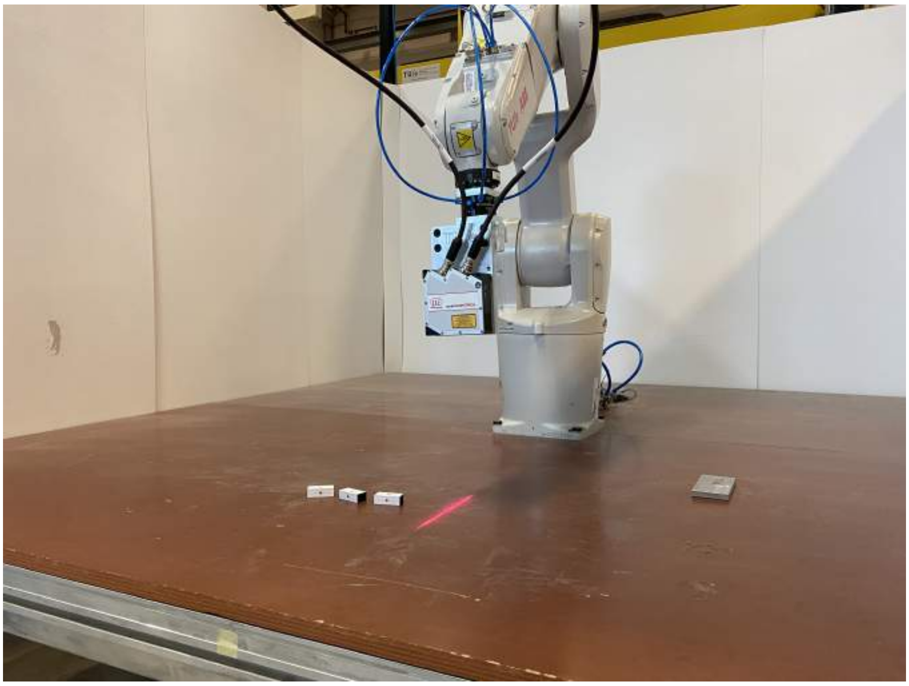


Figure 6.6: Possible other measuring technique

Margin component
The margin component, which was introduced in Section 3.2 is a part that causes some discussion and leaves a lot of room for further research. This is the case since the margin component for this research was firstly taken as a number that was based on one test case during this research. A margin component should be added to the structural check since otherwise the structures can be built on the ultimate edges and no safety margin of any kind is taken into account. Therefore it is chosen to add a margin component of some sort. However, during the different tests the margin component came up for discussion since it is not preferable to enlarge this component just for safety reasons since then the advantages of the robotic manufacturing are not used and the design could be limited more and more. Therefore this margin component should be chosen carefully. From the last test results it was the idea to calculate a margin component that takes into account a 95% security (explained further in Appendix B) because for this test it was tried to eliminate all the possible uncertainties. Since the results and differences between the two models from these tests fluctuate quit a lot, the question arose if this is the right way to calculate a universal margin component for the whole structure or if this should be done another way. From this it can be concluded that if the margin component is calculated with the mentioned formula that in some cases the bricks would still fall looking at the magnitudes of the differences between the two models. And also when looking at the differences in the X and Y direction, the values are not in the same range. So, if an universal margin component would be calculated, the calculation could become too much on the safe side. Therefore, another strategy to calculate and come to such a margin component should be considered such as a margin per direction, per brick and its size or per region in the camera view. However, this was not included in this research and the results that were gained with the tests were not specified for this calculation. If this were to be a bigger part of a research more focus can be laid on getting the right numbers and results to calculate and conclude something on this margin component. For the sake of consistency the margin component has been kept the same over this entire research even though this was not the intention from the beginning.

Data conclusion
There were two ways discussed in this research to draw conclusions on how to build further after measurements are conducted in the structure. The first option was to look per brick and the second option was to look at the whole remaining structure. Both these options showed potential. The first option could be automated within the framework and a conclusion from this analysis was made quickly. The second option took the whole structure into account which makes it possible to account for multiple failed structural checks. However, this option needed an additional optimization which was quite heavy to run in combination with all the other equipment on one laptop making it impossible to run this optimization during the build of the structure (this will be further elaborated in the next discussion point). Therefore both these options were used within this research where both of them have some advantages and disadvantages.

Automation
All the steps in the fabrication process with the robot were automated in the end. However, when all these automation steps were added to the Grasshopper model it became clear that the constant data flow was too much for the model causing the model to crash. Therefore three data dams were added to the Grasshopper model. The first data dam was placed directly after the component that gets the state of the digital output (DO-01) signal of the end-effector from the robot via the remote connection component. This signal (a boolean) then goes to the component that gets the data from the Excel file that was made by Vision Builder. When this data dam is not added this component, that gets the data from the Excel file, crashes causing the whole model to stop working. With the data dam, there are still some seconds needed when the signal is passed through but the model does not crash anymore making it possible to continue with the built. The second data dam was added just before the structure is translated to robot targets. This is not necessarily done to avoid the program from crashing but to stop the data flow at the point to see if the correct next coordinates are taken. The last data dam was added just before the new robot code is uploaded via the remote connection component. This is done once for the main code and once done for the base code. By putting the data dam open it is also required to grant the writing access on the robot panel by a person. By placing a data dam at that specific place the data flow is not directly uploaded to the robot which caused also some crashes before. In the ideal situation these data dams would not be present and the dataflow would run by itself through the model without crashing. However, it seemed to be a limitation to what Grasshopper or the computer can handle since all the other programs were still running and accessible at the moment of the crashes. During another research with this model or a model like this this should be taken into account to avoid the same problem from happening. During the final tests this became visible together with the fact that the optimization tool could not be used in combination with all this equipment on one computer. This slowed down the lasts checks of the whole framework (Section 4.3). It even became impossible to run the optimization during the test itself. The optimization could be run before and after the built of the structure but not during the manufacturing process itself making it impossible to use this tool during the second test of the final tests (Section 4.3).

6.3 Discussion general framework

Whole framework
A first discussion point in general on the framework is about possible future improvements and alterations. The aim of this research was to make a framework to work with possible uncertainties during a robotic manufacturing process, where parts of this framework could be altered/improved by more specific research or new technology for example. This is partly achieved in the way that parts of the framework can be altered but also some parts are dependent to build further from another part. An example of this is the structural analysis, which was now set at an analysis based on the centre of masses. This analysis was also used to determine how many millimeters a brick was moved if this was necessary considering the outcome of the structural analysis for one of the options of the data conclusion. If this structural analysis principle was to be changed, the way of calculating the movement of the brick that would otherwise fall will probably also need to be changed. The framework can be altered and improved by other research but it need to be kept in mind that the implementation of parts in the framework could lead to other alterations of other parts of the framework as well.
From the beginning of the research it is stated that the geometric uncertainties should be coped with. However, there are still some geometric uncertainties that are not embedded in this framework and would maybe need an additional part of script to be incorporated. The biggest examples of these uncertainties are the levelling of the ground and part tolerances. The consequences of these uncertainties are now not taken into account but could lead to some differences to an initial model. It is expected that some additional research scripting is needed to incorporate the consequences of these uncertainties completely.

Design process
Another limitation of this type of framework as a whole is the fact that the design process is not taken that much into account. Within the main research question, both the design process as well as the manufacturing process of a robotic building process was mentioned. However, in the final framework the manufacturing process is taken into account but the design process not that much since a design was inserted into the framework without having much boundaries for this design. The design could be altered according to the initial structural check, however, the manufacturing process is not taken into account for this design of a structure. When the robotic manufacturing process is taken into account during the design phase, several boundary conditions could be set up for example which are applicable for the robotic manufacturing process. This could result into designs that are easier to build with the robot and therefore could have more success within this framework.

Measuring technique
The used measuring technique was chosen since it was one of the possibilities that was already present in the TU/e Structures Laboratory and others had experience with the cameras and program, which made it quite easy to ask others for help. However, that does not mean that this measuring technique could be applicable or is representative for a building site. Since this measuring technique has quite some important aspects such as the constant straight angle of the camera view, as explained in Section 6.2, it could be the case that this measuring technique is not at all suitable for a building site where the ground could be uneven or vibrating for example. Also other unpredictable factors of a building site have influences on the camera view and the ability of Vision Builder to analyse the pictures. The lighting is one of them since the Vision Builder analysis could contain errors if the lighting is too dark or too bright. Also, the fact that people could touch the cameras could lead to slightly different positions of the cameras resulting into inaccurate calibration factors. This measuring technique is suitable for this experimental research in a laboratory to make the first set-up of this framework and gain results in the amount of time of this research. However, there could be improvements towards a measuring technique that could actually be used on a building site or at least research which measuring techniques would be suitable for such a situation.

Applicability
This specific framework has proven to be useful for this research and the tests that were conducted in it. The specific used methods for the separate parts of the framework however, are up for discussion when the framework would be used for other cases or in other situations as became clear from the other discussion points. That makes this specific framework not directly applicable for other situations. However, the overall framework design and steps are very much applicable to other research projects or building situations since the sequence of steps where real-time measurements are used within a digital twin makes it possible to conduct analysis during the manufacturing process and respond to this analysis. It should therefore, be kept in mind when such a framework is used for other cases, that the design of the framework and steps that are presented in this thesis are very useful, but the specific completion of the separate steps might differ from the completion presented in this thesis.

Allowable deviations
One last discussion point arose for the whole idea of such a framework namely the allowable deviations between the digital design and the build model. Until when do you want to make adjustments during the manufacturing process and from when do you start over since the built structure is too far from the initial design and the requirements of the design are not met anymore? This probably depends also on the design and the use of the structure. However, this could be an interesting case to keep in mind when further research is conducted on this framework.

6.4 Recommendations for further research

From this research multiple ideas for further research come to light. These ideas for further research are explained in this Section divided into four main topics: the translation to the building site, the measuring technique, the data conclusion/other solutions and the opportunities of the robot.

Translation to the building site
The higher goal, as briefly mentioned in Section 1, is to enlarge the application of robotic manufacturing processes in the built environment to fulfill the increasing demand of housing and buildings. This research was a first step into looking how this could be made possible. However, the focus of this research was on establishing a framework that measures a structure during the built and processes these measurements to a conclusion on whether or not the built further. During these steps the actual building site was not taken into account. When this would be taken into account it should be considered more carefully which measuring technique can be easily applied on a building site, where would these possible cameras be placed or would multiple robots work together where one of them measures the structure. Further research could be conducted on making such a framework easy applicable to a building site.

Measuring technique
The measuring technique with two cameras and Vision Builder was adequate for this first set up of the framework but this measuring technique also showed some flaws like the fact that measuring rotation was difficult and the angle of the camera was a big cause to the difference between the digital and the built model. There are also a lot more measuring techniques available that could measure the entire structure in a 3D environment. A separate research for this measuring technique, also taking into account the aforementioned further research idea on what would be applicable on a building site, could give more insights on what would be good to use for such a framework. It then also can be different on how the outcomes of this new measuring technique should be processed. It could be the case for example that the results of another measuring technique are not stored in an Excel file in the same way as it was during this research. This could require an alteration to the processing method of the data in this framework.

Data conclusion/Other solutions
Within this framework two ways of dealing with the differences between the two models (solutions) were considered. The first solution looked at the next brick of the initial design on top of the built structure and if the structural checks of this combination were sufficient. If so, the brick could be laid on the coordinates according to the initial design. If not, the coordinates of the next brick were altered with the distance that the centre of mass was outside of the safe boundaries. With this alteration the structural checks were sufficient and the next brick could be built. The second solution that was added to the framework looked at the whole structure that still needed to be built in combination with the bricks that were already built instead of just the next brick. By doing this more failed checks for higher bricks could come to the light. To still continue building and generate coordinates that do give all sufficient checks, an optimization option is added in the process. With this option, an optimization process is started with the conditions that the number of failed checks is zero and the difference with the initial digital design is as low as possible. From this optimization an option can be selected which is used to built the structure further. However, there are way more options to deal with the differences between the two models. For example when a brick is placed on the wrong coordinate it could also be an option to pick that specific brick up again and replace it before moving onto the next brick. Also other options such as the addition of an adhesive or using more bricks as a temporary support conditions could be considered to prevent the structure from falling. It then becomes the question which solutions can be added or executed during the building process or which ones should be considered before the whole building process starts. Also the question than arises which additional materials are needed to make these solutions possible.

Opportunities of the robot
At the moment of this research there are two separate ABB robots available for research at the TU/e Structures Laboratory. However, there are other options available a few months after the finishing of the research when two robots are moved to one table and they can work together with a multi-move option. With this option it becomes possible for the two robots to work together were one of them can built the structure and one can measure the structure for example. Next to the addition of one robot there could also be more research conducted on the robot end-effector and the possibilities of having both a vacuumtool and a measurement tool on one robot. This would also make it possible to measure simultaneously to the building process and bringing this framework yet another step closer to the building site.

Margin component
The margin component was mentioned a few times during this research as a limitation. It was not clear how to calculate this margin component with the data obtained during the experiments to get to a universal margin component for the whole structure. It even was questioned if it is desirable or possible to have a universal margin component for the whole structure. It could be that a margin component is better applicable for regions of the structure of even per brick. However, to determine this there is more specific research into such a margin component needed. When data is collected purely with this margin component in mind and tests are set up specifically for this research there is a bigger chance to get a reliable result for such a margin component compared to the data collected in this research (Appendix B).

Bibliography

ABB (n.d.). Irb 1200. *https://new.abb.com/products/robotics/industrial-robots/irb-1200*. 23

Basler (n.d.). Basler ace series. *https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/*. 29, 64

Bilesan, A., Owlia, M., Behzadipour, S., Ogawa, S., Tsujita, T., Komizunai, S., and Konno, A. (2018). Marker-based motion tracking using microsoft kinect. *IFAC-PapersOnLine*, 51:399–404. 25

Brugnaro, G., Baharlou, E., Vasey, L., and Menges, A. (2016). Robotic softness. *Posthuman frontiers*, pages 154–163. 9

Brugnaro, G. and Hanna, S. (2017). Adaptive robotic training methods for subtractive manufacturing. *ACADIA*. 13

Caballar, R. D. (2019). This bricklaying robot is changing the future of construction. *https://redshift.autodesk.com/bricklaying-robot/*. xi, 9, 10, 12

Cao, L. (2019). The evolution of bricklaying robots: Changing the rules of traditional construction. *https://www.archdaily.com/928440/the-evolution-of-bricklaying-robots-changing-the-rules-of-traditional-construction*. vii, 3

Cobouw (2019). Nauwelijks robots in de bouw. *https://www.cobouw.nl/marktontwikkeling/nieuws/2019/10/nauwelijks-robots-in-de-bouw-101277882*. 11

Ergodomus (2020). Robotic fabrication of mass timber sound diffusers. *https://www.ergodomus.it/2020/04/robotic-fabrication-of-mass-timber-sound-diffusers/*. 8, 9

ETHZürich (2018). Robotic collaboration in timber construction. *https://ethz.ch/en/news-and-events/eth-news/news/2018/03/spatial-timber-assemblies.html*. xi, 3, 5, 6

Food4Rhino (n.d.). Octopus. *https://www.food4rhino.com/en/app/octopus*. 23

Hartman, P. (n.d.). 3d handscanner van hoge resolutie. *https://meet-tekenwerk.nl/hoge-resolutie-3d-handscanner/*. 25

HMCArchitects (2019). Robotics in architecture and construction: an industry shift. *https://hmcarchitects.com/news/robotics-in-architecture-and-construction-an-industry-shift-2019-10-23/*. 10

Industrievandaag (2021). Abb robotics maakt bouwsector veiliger en duurzamer met automatisering. *https://industrievandaag.nl/abb-robotics-maakt-bouwsector-veiliger-en-duurzamer-met-automatisering/*. 11

Khawli, T. A., Anwar, M., Gan, D., and Islam, S. (2021). Integrating laser profile sensor to an industrial robotic arm for improving quality inspection in manufacturing processes. *Institutions of mechanical engineers*. xi, 13

King, M. (2017). How industry 4.0 and bim are shaping the future of the construction environment. *https://www.gim-international.com/content/article/how-industry-4-0-and-bim-are-shaping-the-future-of-the-construction-environment*. 3

Knight, W. (2021). Ford's ever-smarter robots are speeding up the assembly line. *https://www.wired.com/story/fords-smarter-robots-speeding-assembly-line/*. 10

Kritzinger, W., Karner, M., Traar, G., Henjes, J., and Sihn, W. (2018). Digital twin in manufacturing: A categorical literature review and classification. *Elsevier B.V.*, 51. 62

Lalor, A. (2021). Why is there a housing shortage in the netherlands? the dutch housing crisis explained. *https://dutchreview.com/expat/housing/why-is-there-a-housing-shortage-in-the-netherlands-the-dutch-housing-crisis-explained/*. 3

Leder, S. and Weber, R. (2018). Distributed robotic assembly system for in-situ timber construction. *https://www.icd.uni-stuttgart.de/teaching/master-theses/itech-m-sc-2018-distributed-robotic-assembly-system-for-in-situ-timber-construction/*. xi, 6, 7

Micro-epsilon (n.d.). Laser scanners for 2d/3d profile measurements. *https://www.micro-epsilon.com/2D_3D/laser-scanner/*. 25

Microsoft (n.d.). Kinect voor windows. *https://developer.microsoft.com/nl-nl/windows/kinect/*. 25

Molloy, J. C. (n.d.). 5 robot revolutionizing architecture's future. *https://www.archdaily.com/336849/5-robots-revolutionizing-architectures-future*. 3, 5, 10

NationalInstruments (n.d.). Vision builder for automated inspection. *https://www.ni.com/nl-nl/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/what-is-vision-builder-for-automated-inspection.html*. vii, 29

Osmani, M. (2011). Construction waste. 3

Panasonic (n.d.). Cmos type micro laser distance sensor hg-c. *https://www3.panasonic.biz/ac/ae/fasys/sensor/photoelectric/hg-c/index.jsp*. 26

Paritala, P. K., Manchikatla, S., and Yarlagadda, P. K. (2017). Digital manufacturing- applications past, current, and future trends. *Procedia Engineering*, 174:982–991. 3, 6

Posada, J. R. D., Schneider, U., Pidan, S., Geravand, M., Stelzer, P., and Verl, A. (2016). High accurate robotic drilling with external sensor and compliance model-based compensation. *6 IEEE International Conference on Robotics and Automation*. 13

Prado, M., Dörstelmann, M., Schwinn, T., Menges, A., and Knippers, J. (2014). Core-less filament winding. *Robotic Fabrication in Architecture, Art and Design 2014*, pages 275–289. 8, 9

ROBTechnologies (n.d.). Overcome irregularity. *https://rob-technologies.com/adaptive-processing*. 11, 12

Simondetti, A. (n.d.). New robots will revolutionise the built environment. *https://www.arup.com/perspectives/new-robots-will-revolutionise-the-built-environment*. 6

Soomro, H., Khan, A., Amna, K., and Anwar, M. S. (2020). The physics of stacking. xi, 19

Thomsen, K. K. and Kraus, M. (2014). Simulating object stacking using stack stability. 19, 20

Usmanov, V., Bruzl, M., Svoboda, P., and Šulc, R. (2017). Modelling of industrial robotic brick system. *34th International Symposium on Automation and Robotics in Construction (ISARC)*. 3, 10, 11

van Koert, L. (2021). Bouwen met robots. *https://www.robot-magazine.nl/article/Whitepaper/48446-bouwen-met-robots*. vii, 3, 11

Waibel, M. (2011). Architects using robots to build beautiful structures. *https://spectrum.ieee.org/robots-in-architecture#toggle-gdpr*. 3, 10, 11, 12

Willmann, J., Knauss, M., Bonwetsch, T., Apolinarska, A. A., Gramazio, F., and Kohler, M. (2016). Robotic timber construction - expanding additive fabrication to new dimensions. *Automation in Construction*, 61:16–23. 5

Wolf, C., de Waal, L., and Kemps, D. (2022). Bouwrobots winnen terrein; markt verdriedubbelt in 2030. xi, 7, 8

Xie, R. (n.d.). 21sp' robotic timber assembly. *https://ruxinx.com/21sp-robotic-timber-assembly*. 9

Appendix A

Grasshopper model

This appendix shows the whole Grasshopper model that was used for this graduation project in Grasshopper components, figures and text. The plug-in that was used for this Grasshopper script is Robot components Version 1.2.0. Another used plug-in in this Grasshopper model is Octopus which is used for the optimization part of the script.
The Grasshopper model is split into the parts of the framework as discussed in Chapter 3.

A.1 Design of structure

The first part of the Grasshopper model makes the design of the structure, which is in this case a tower of bricks. First the brick itself is defined with three parameters: the length, the width and the height of the brick. Secondly the brick is moved so that the centre point of the brick is now in the 0,0 coordinate (x,y).

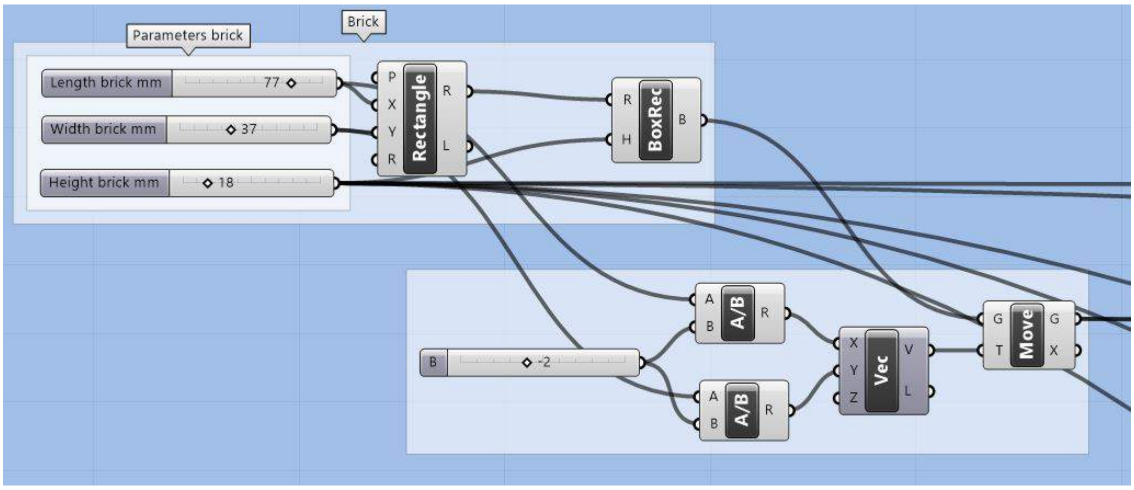


Figure A.1: Definition of the brick

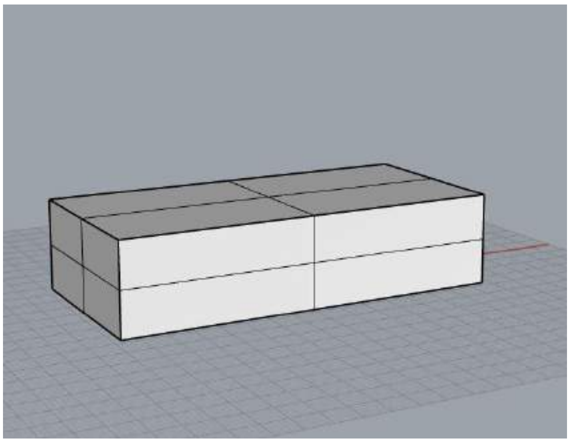


Figure A.2: Visualization of the brick

When the brick is defined, a line on which the bricks will be oriented is made. This is done in two separate parts, one part for the movement in the X-direction and one part for the movement in the Y-direction. First the movement in the X-direction is described. A curve with the shape of a sine was created where the amount of amplitudes is still variable. Also the value of the amplitudes of the curves is set as a parameter, making it easy to generate different structures to test with the robotic fabrication. This is the parameter that is used for the different structures and therefore mentioned and described throughout the whole thesis. The basic sine curve it scaled to fit a number of bricks, which is also set as a parameter and therefore variable, and rotated to the Z-axis.

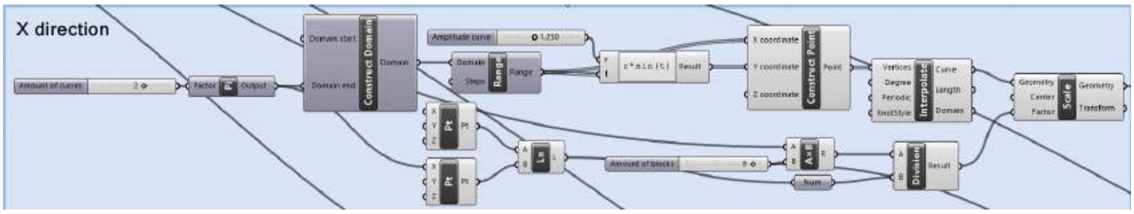


Figure A.3: Movement X-direction

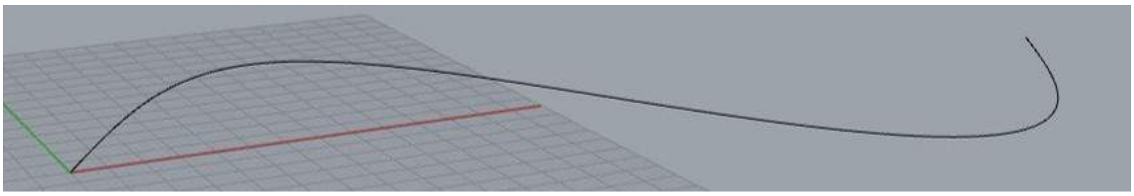


Figure A.4: Curve X-direction

The same steps are used for the movement of the bricks in the Y-direction as for the X-direction. The amount of curves and the amplitudes of these curves are set in separate parameters as these two variables in the part of the movement in the X-direction, making it possible to play with these parameters and generate multiple different structures of bricks. When the curve is finished it is translated into horizontal frames. The amount of horizontal frames is the number of bricks that was also used to scale the sine curve. The last step in this part for the movement in the Y-direction is the translation of the horizontal frames to unit X vectors.

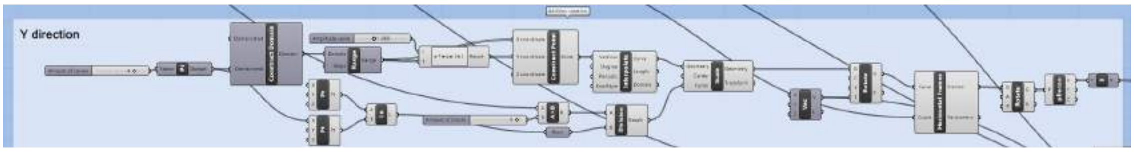


Figure A.5: Movement Y-direction

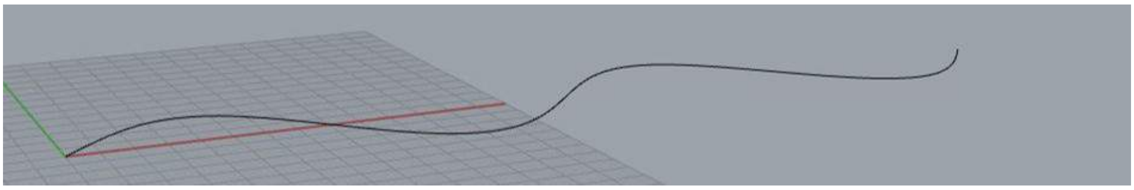


Figure A.6: Curve Y-direction

One last part before the structure is defined is a small step to get the height of the horizontal frames right for the height of the bricks. The horizontal frames that were created in the previous parts are not entirely on the correct height according to the heights of the bricks since the division over the curve to get to the horizontal frames was based on a division by number and not focused on the height. This is corrected by taking the sine curve of the movement in the X-direction and slicing the curve according to the height of the bricks. This then makes sure that the bricks fit on top of each other instead of there being some gaps between the bricks or some overlap in height.

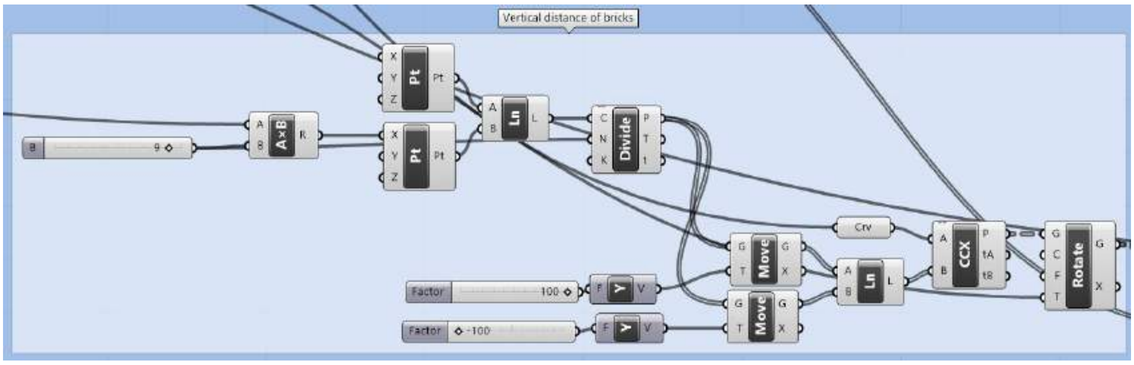


Figure A.7: Vertical distance

In the next part the points on the curves and the bricks are combined to form a structure that is used in all the other parts of the Grasshopper script. There is one set that can be used when only movement in the X-direction is wanted and there is one set that can be used when both movement in the X-direction and movement in the Y-direction is wanted. This was done to build the tests from a simple structure to a structure that is a bit more complex. When one of the sets of points is selected, the bricks are oriented on these points to form the structure of bricks. In the first case this is done for all the points of the curves but the components after the establishment of the structure make it possible to select a part of the structure.

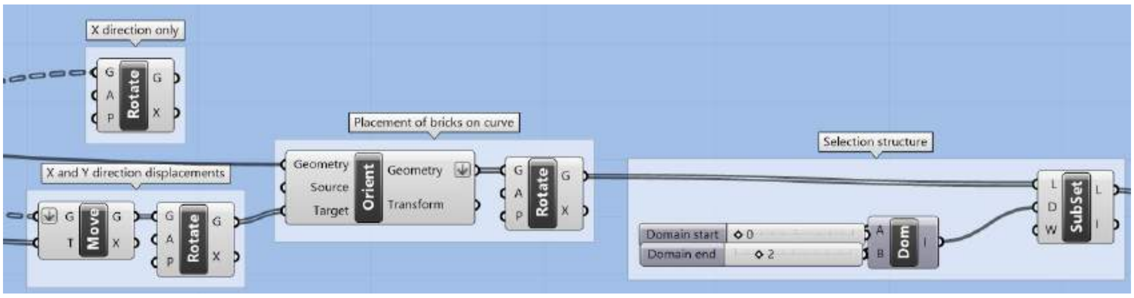
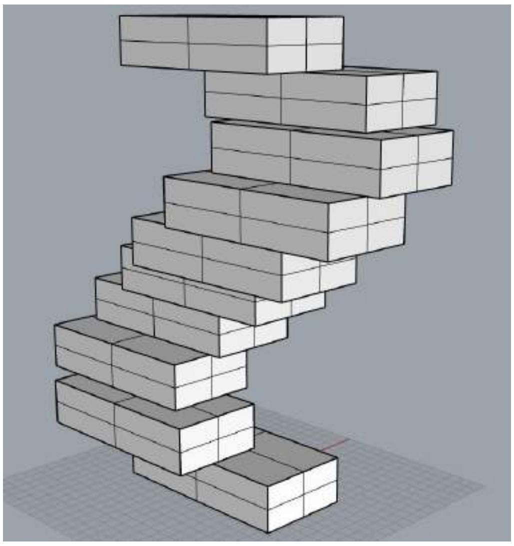
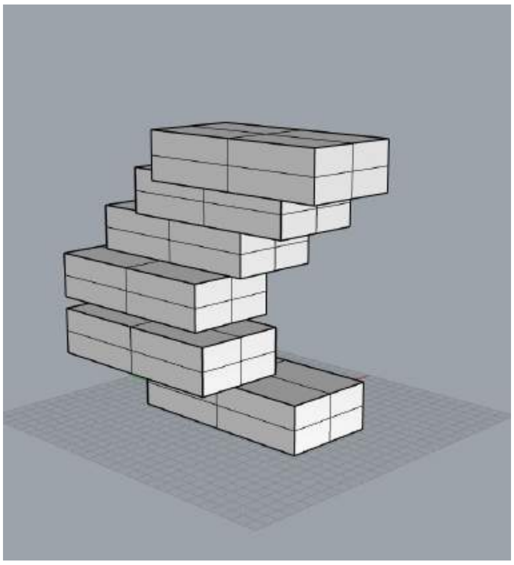


Figure A.8: Structure selection

In Figure A.9a the total structure is shown that is generated with the Grasshopper script. Figure A.9b shows a selection of the total structure with the amount of bricks that is chosen.



(a) Total structure



(b) Selected structure

Figure A.9: Structures

A.2 Structural analysis

When the final initial structure is completed during the last step of the design part of the Grasshopper model, the structural analysis can be executed. First some additional information is given on the chosen structure in the meaning of the coordinates of the corner points of the bricks and the coordinates of the centre of masses of the individual bricks (Figure A.10).

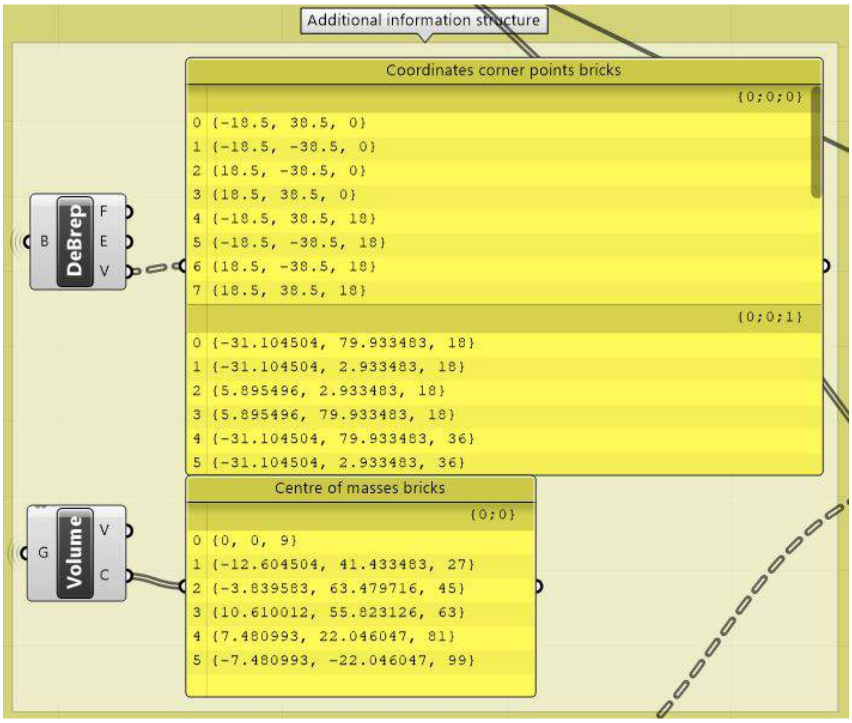


Figure A.10: Additional information on designed structure

The defined structure is subjected to a structural check based on centre of masses as explained in Section 3.2. In order to conduct the structural analysis, the structure was used as an input for a Python component where all the possible combinations and individual bricks are set in a data tree structure with the boundaries that the bricks should be checked to.
This data tree is then split into one data tree with the boundaries to which the centre of masses should be checked to see whether or not they fall within the boundaries. The other data tree contains all the combinations of bricks and individual bricks of which the centre of mass or combined centre of mass should be calculated. The order in these data trees are linked to one another meaning that the second boundary in the first data tree is the boundary to which the second item of bricks in the second data tree should be checked.

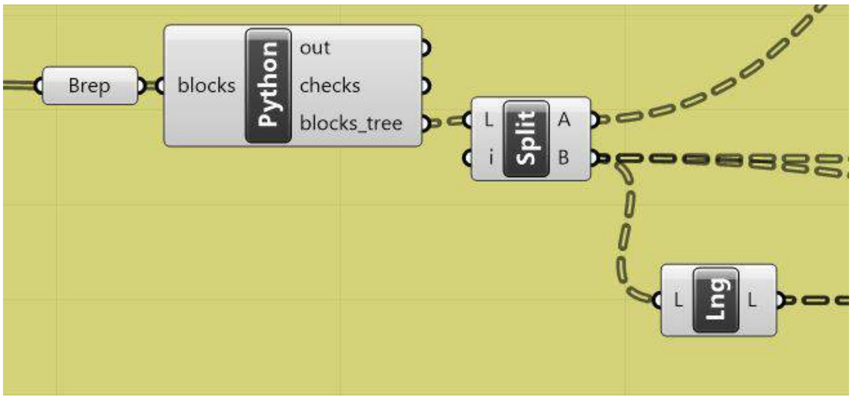


Figure A.11: Generating datatree for structural check

To get to the data tree and generate all the possibilities of individual bricks and combined bricks, a small Python script was used in a Grasshopper component.

```
import ghpythonlib.treehelpers as th

n = len(blocks)

block_num = range(n)
sizes = range(2, n + 1)

checks = []
blocks_tree = []

for size in sizes:
    for i in block_num:
        check = block_num[i:i + size]

        if len(check) == size:
            tree = []
            for i in check:
                tree.append(blocks[i])

            blocks_tree.append(tree)
            checks.append(str(check))

blocks_tree = th.list_to_tree(blocks_tree)
```

The datatree with the boundaries to which the centre of masses should be checked is led to a separate part in the Grasshopper script. In this part of the script the boundaries are set so that the structural check can actually be conducted. To get to that point, the boundaries are defined as a surface and after that as a box. These boxes are all moved to the Z-axis and scaled with the margin component (explained in Section 3.2 and Chapter 6). The margin component is in this example set at 0,98 meaning that the boundaries are scaled to 98% of the original boundaries. These scaled boxes are then the final boundaries to which the centre of masses will be checked.

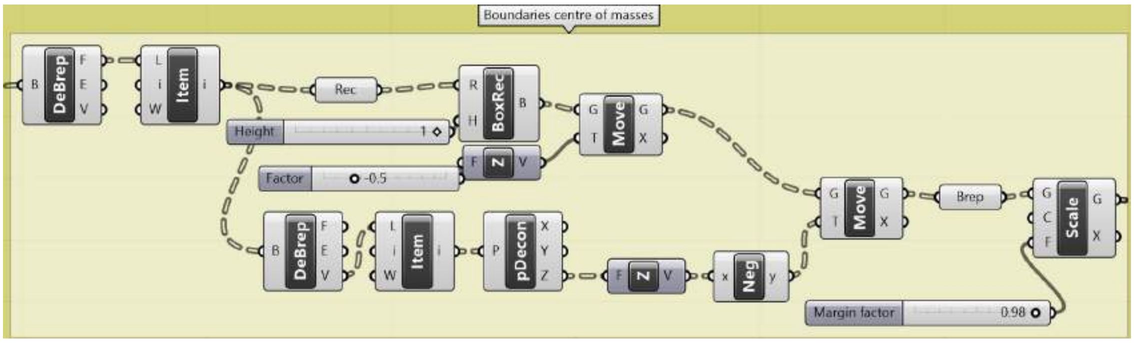


Figure A.12: Calculation of boundaries for structural check

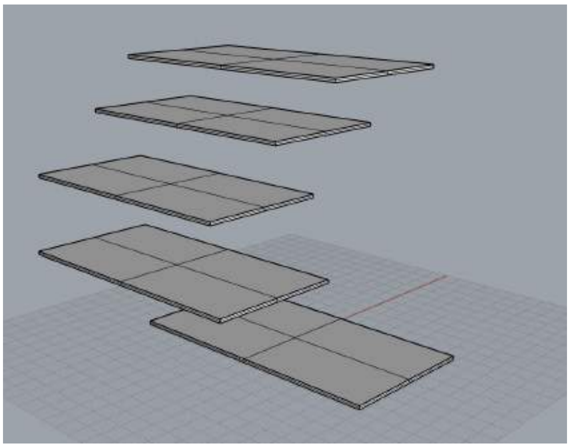


Figure A.13: Visualisation of safe boundaries

The datatree with the combinations of bricks and individual bricks that are needed to calculate the centre of masses are led to a second separate part of the Grasshopper script. Within this part of the script the centre of masses of all the bricks are calculated and if needed a combined centre of mass is calculated if more bricks need to be combined for one check. These centre of masses are then moved to the Z-axis so that it can be checked whether or not they fall within the boundaries.

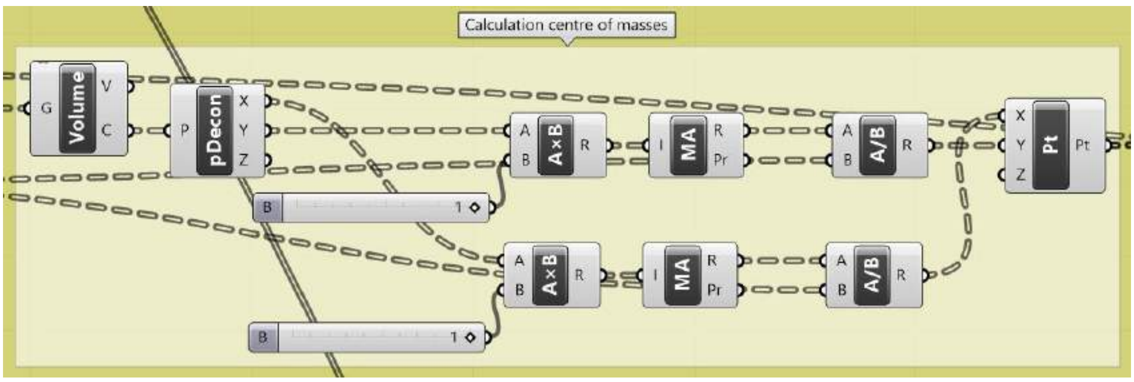


Figure A.14: Calculation of centre of masses for structural check

For the explanation of the Grasshopper script a structure with an X-amplitude of 1,75 and a Y-amplitude of 1 is taken. This combination is tested with the structural analysis. Figure A.15 shows the component which checks if the centre of masses are within the boundaries together with the outcome of this check. It shows that this particular structure has one false outcome of all the conducted checks.

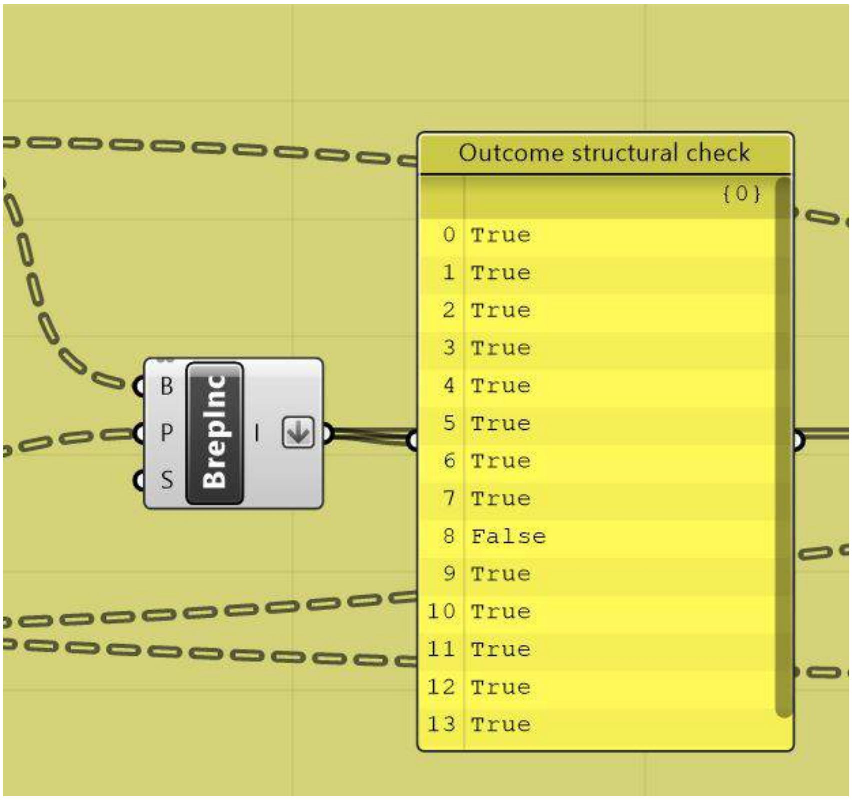


Figure A.15: Outcome of structural check

To understand the outcome of the structural check, some components are added to the script to visualize this outcome. The amount of failed checks and their index numbers are shown. Also the closest distance of the centre of mass to the boundary is given if there is a failed check.

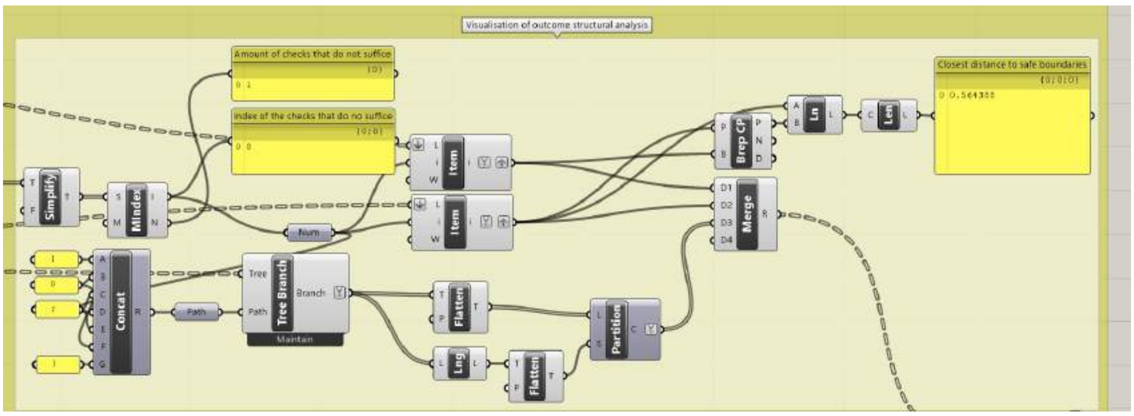


Figure A.16: Visualization of failed structural checks

If there are more checks that fail, which is for example the case with a structure with an X-amplitude of 1,75 and a Y-amplitude of 1,25. All the failed checks can be seen in a visualized manner by using the slider shown in Figure A.17. Figures A.18a and A.18b show the visualisation of two of the failed structural checks. The visualisation shows that that the combination of the shown bricks cause a collapse of the structure since the combined centre of mass of these bricks is not within the safe boundary of the brick directly underneath it. A small point close to the safe boundary shows the distance that the centre of mass is outside the safe boundary which is also given as a distance in millimeters in a panel.

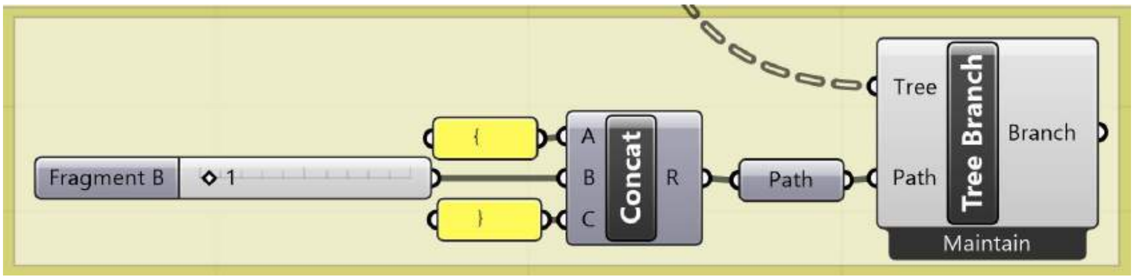
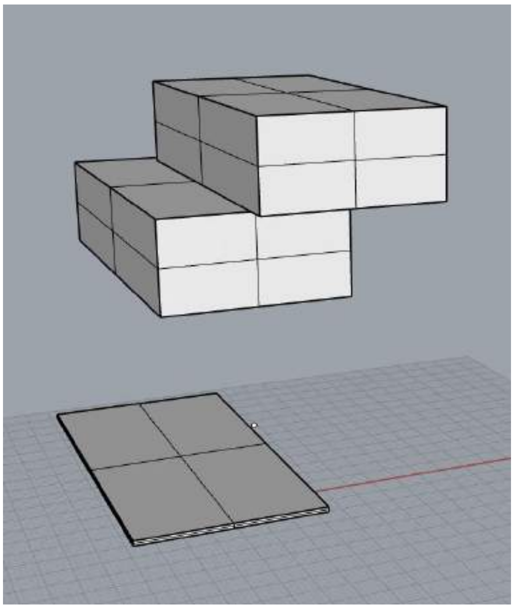
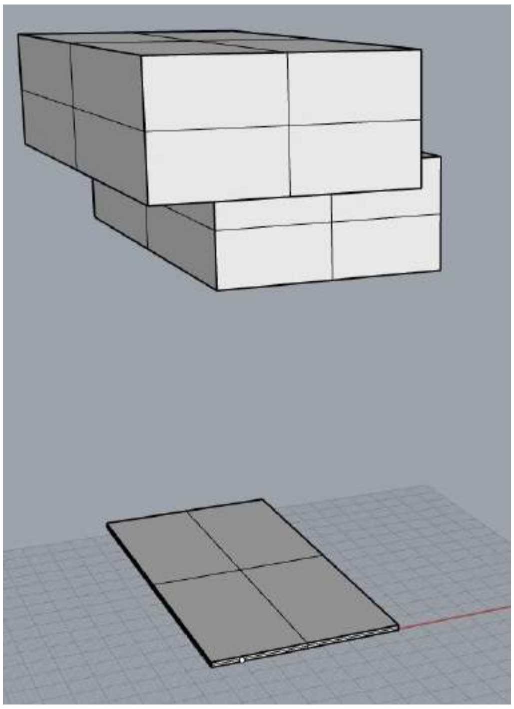


Figure A.17: Visualization of failed structural checks



(a) Failed structural check 1



(b) Failed structural check 2

Figure A.18: Visualization of failed structural checks

A.3 Optimization possibility

As explained in Section 3.2 there is an optimization option if the initial design has failed checks at the structural analysis and there is a need for a design as closely to the initial design as possible. This optimization part of the Grasshopper model uses the plug-in Octopus. The exact part containing the initial design and the structural analysis are copied for this optimization part. However, now the input for the values for the X and Y amplitudes are put in a gene pool (the first gene is the X-amplitude and the second gene is the Y-amplitude). The parameters within this gene pool are used to run the optimization.

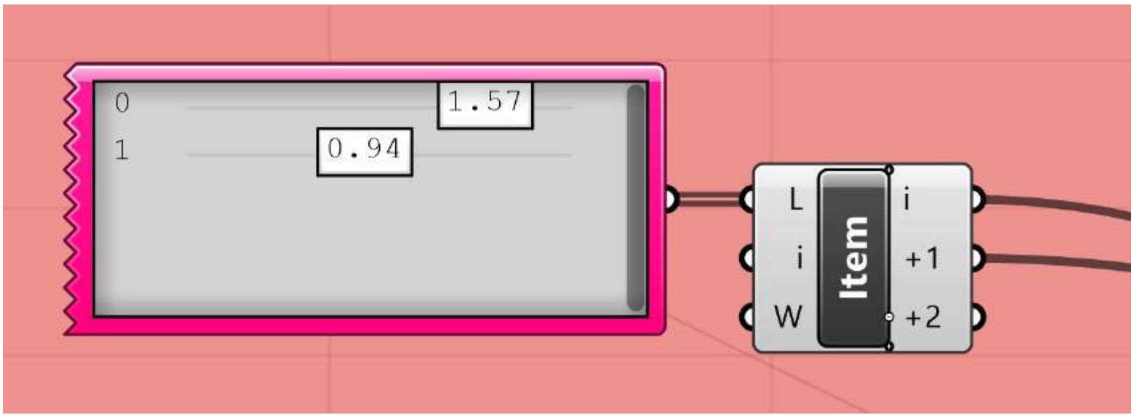


Figure A.19: Gene pool for Octopus optimization

To make the optimization complete there are two objectives added to the octopus component. The first objective is the number of failed checks from the structural analysis. The second objective is the distance between the centre of masses of the initially designed structure and the structure made with the octopus optimization.

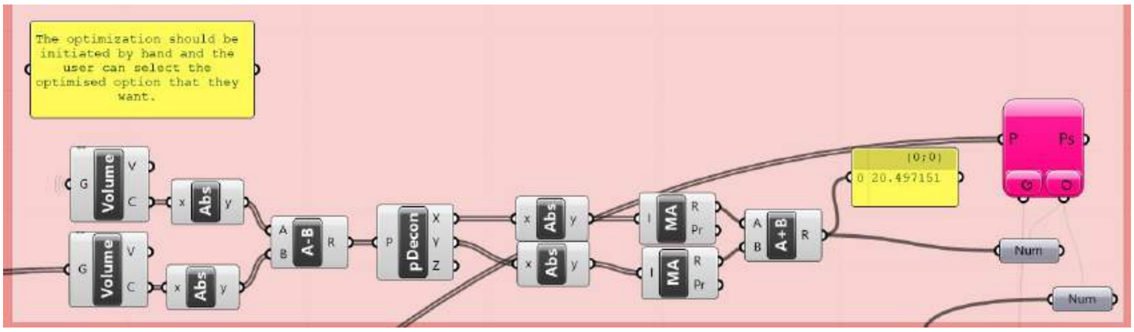


Figure A.20: Octopus optimization

When the Octopus component is run, the graph shown in Figure A.21 will appear. The amount of failed checks need to be 0 in order to get to a structure that can be built, therefore the chosen result should be on the right axis. The result most to the vertical axis is the result with the smallest difference to the initial model. Therefore the result on the right axis most closely to the vertical axis is in this case the best option to use. The optimization is activated and an outcome is selected manually. This is the case since the optimization can run for a very long time since there are a lot of options. During this research the optimization was therefore stopped at a certain moment and a solution was selected by hand.

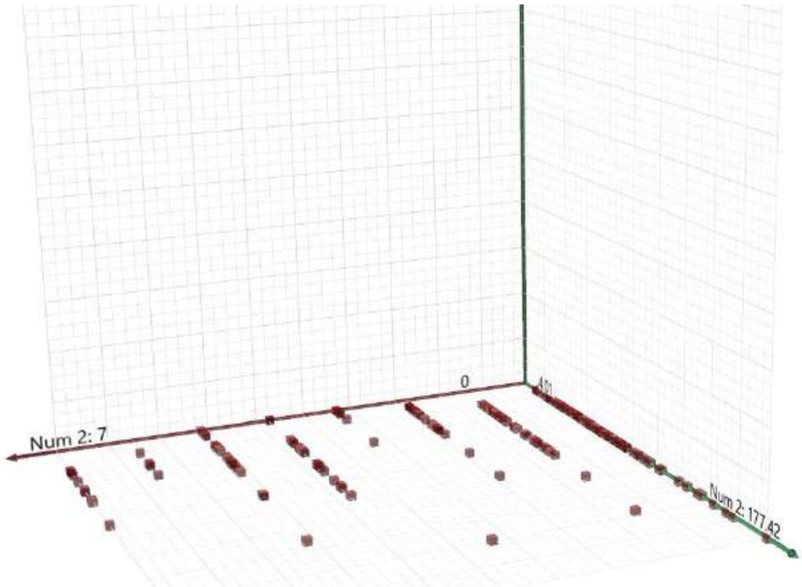


Figure A.21: View of the running Octopus optimization

A result of the optimization could look like Figure A.22 where the white structure is the initially designed structure and the green structure is the optimized structure that has no failed checks and can therefore by built by the robot.

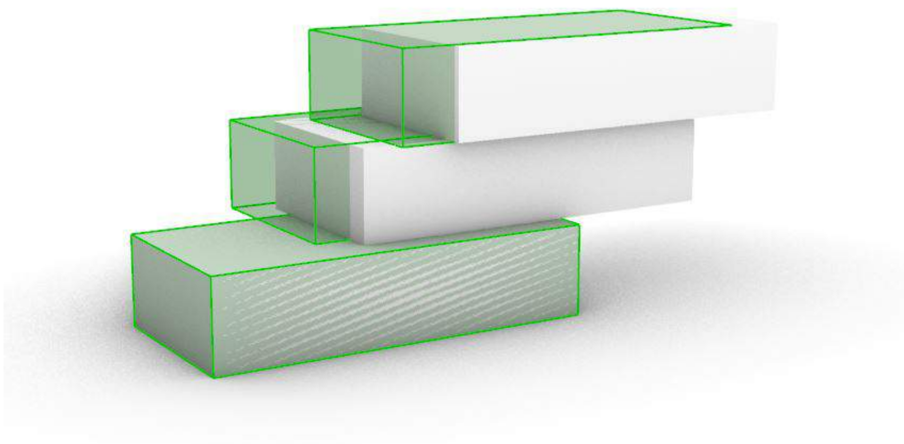


Figure A.22: Result of the Octopus optimization (white is initial design, green is optimized)

A.4 Data collection and data processing

During the built of the structure,the placed bricks are measured after every brick to see if there is a need for adjustments to the coordinates of the bricks that are not yet placed. In order to do this, the data generated by Vision Builder, as explained in Section 3.4, should be loaded into the Grasshopper model. The steps taken in this were already explained in Section 3.7.2.4 so this section only contains the final part of the model. During the built of the structure there needs to be a moment when it is save to take a picture, generate the data in Vision Builder and load that data into the Grasshopper model. From the Rapid Code that is used to control the robot, the digital output can be used for this since the digital output is true when the vacuum tool is on and the digital output is false when the vacuum tool is off. This line of code is therefore used to control the actions since it is a given that when the line of code is turned from false to true the brick is then picked up and the robot is not in the way of the picture.
First, to use this line of code and being able to see where in the code the robot is, a live connection is made between Grasshopper and the robot. When this live connection is made, this can be used to upload the main and base code directly to the robot and get the status of the code from the robot. The component Get Digital Output gives then the digital output of the DO-01 item as a Boolean.

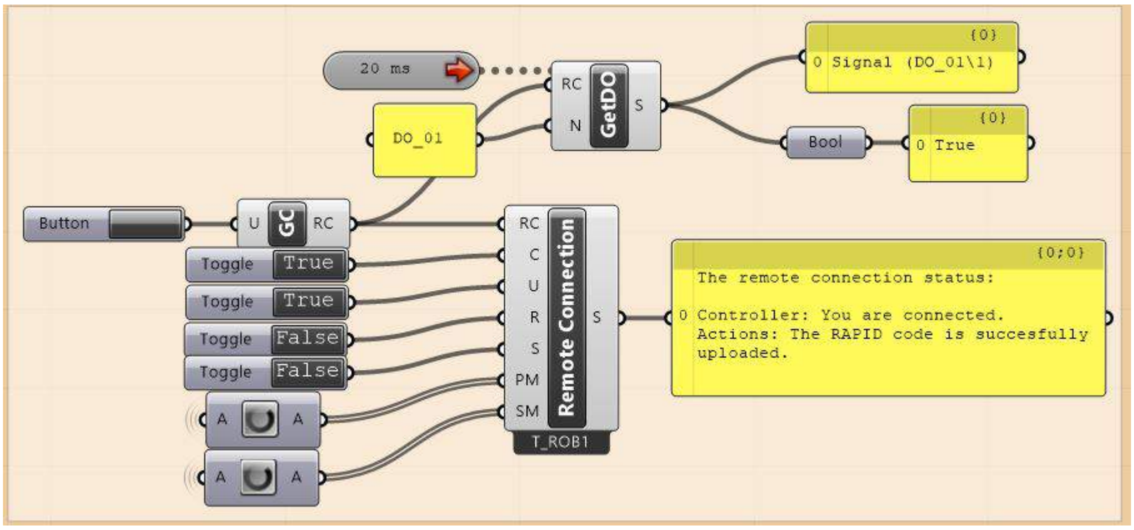


Figure A.23: Live connection robot to Grasshopper

To get the data from the Excel file , generated by Vision Builder, to Grasshopper, a small Python script is used to get the Excel data from a place on the computer that is given beforehand. The data in the Excel file is structured in multiple rows where the first couple of cells of a row are the X-coordinates of the placed bricks and the last couple of cells are the Z-coordinate of the placed bricks. The placements of the coordinates belong together, meaning that the first X-coordinate belong with the first Z-coordinate. This is of importance since, when you have a stack of bricks, Vision Builder does not know which brick is placed first or last and it organises the data in an ascending manner in the X-direction. This means that the coordinates are not necessarily in the right order that they are placed in yet, since the ascending order in X-direction is not how the tower is build but the order in the Z-direction is. This will be solved in Grasshopper in the next part of the script. The Python code takes the last row of data that is in the Excel file since this is the last added data and therefore the data that is needed at every point. There is one value that should be changed if the height of the build structure changes. The j value, which is now set at 7, gives the amount of cells after which the split should be made between the X-coordinates and the Z-coordinates. In this case it means that there are 7 X-coordinates (6 for the bricks and 1 for the reference point) and 7 Z-coordinates.

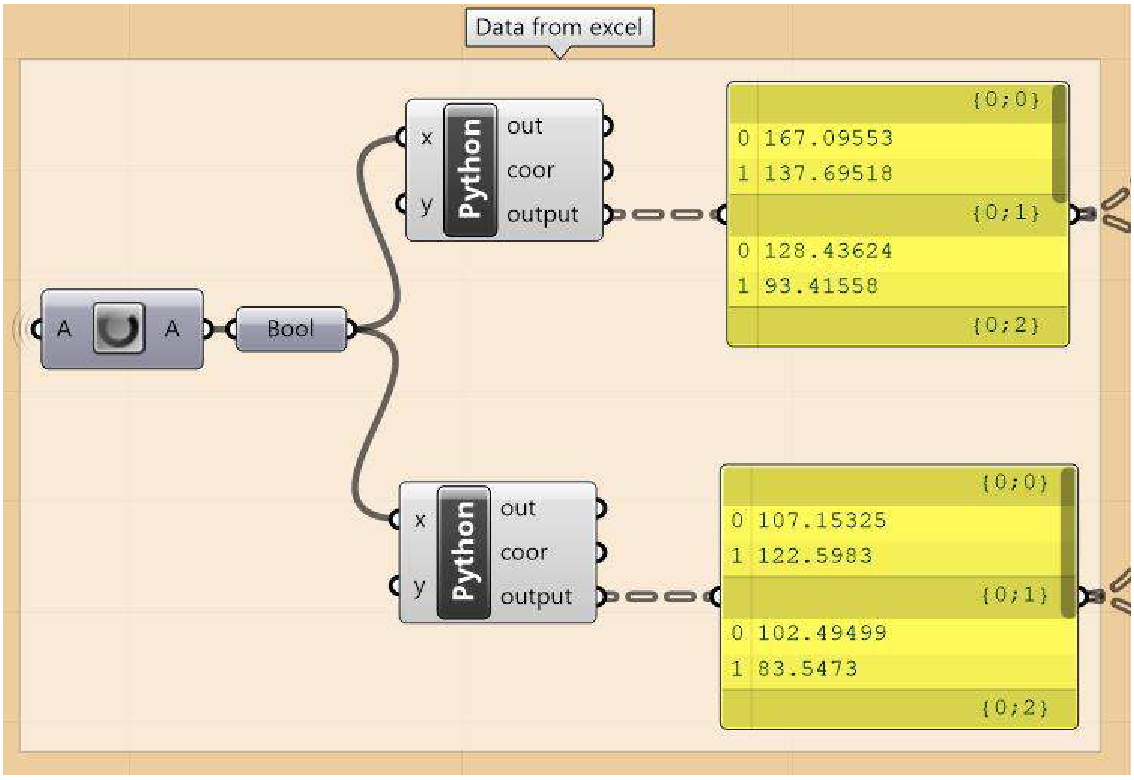


Figure A.24: Imported data from Excel

```
import rhinoscriptsyntax as rs
import ghpythonlib.treehelpers as th
import csv
import time

time.sleep(5)

if x:
    length_doc = -0

    f = open('C:\Users\Amy Hendriks\Desktop\Afstuderen\Visionbuilder\Visionbuilder\
        cam1.csv')
    reader = csv.reader(f)

    for row in reader:
        length_doc += 1
    f.close()

    current_row = 0
    output = []

    #output.append(test)
    file = open('C:\Users\Amy Hendriks\Desktop\Afstuderen\Visionbuilder\
        Visionbuilder\cam1.csv')
    file_reader = csv.reader(file)

    for row in file_reader:
        current_row += 1
        if current_row == length_doc:
            half = int(len(row) / 2)
            j = 7
            for i in range(len(row)):
                coor = []
                if i < half:
                    if row[i]:
                        #print(j)
                        value_one = row[i]
                        value_two = row[j]
                        j += 1
                        coor.append(value_one)
                        coor.append(value_two)
                        output.append(coor)

    output = th.list_to_tree(output)
    print(output)
    file.close()
    time.sleep(1)
```

When all of the above is conducted and Grasshopper understands the numbers in the datatree, the items in the datatree are organised by the Z-coordinates of the bricks. When that is done the Z-coordinates of the bricks are not needed anymore and the X-coordinates are combined with the Y-coordinates, since these where measured with the two different cameras and therefore processed in two different data streams. When all of that is done the points of the centre of masses of the placed bricks are defined and they are placed on the correct height with a series component. One last thing that is added at this part of the Grasshopper script is the reference point. This is a point that is put permanently in the test set-up and is used to determine the placement of the very first brick. For all the other bricks the spacing with the first brick can be taken to determine the distance between them. However, that is not possible to do for the first brick and assuming the first brick is put at the 0,0 (x,y) point is not correct since there could be a deviation at the placement of the first brick. Therefore, the reference point was introduced which needs to be entered before the whole process starts.

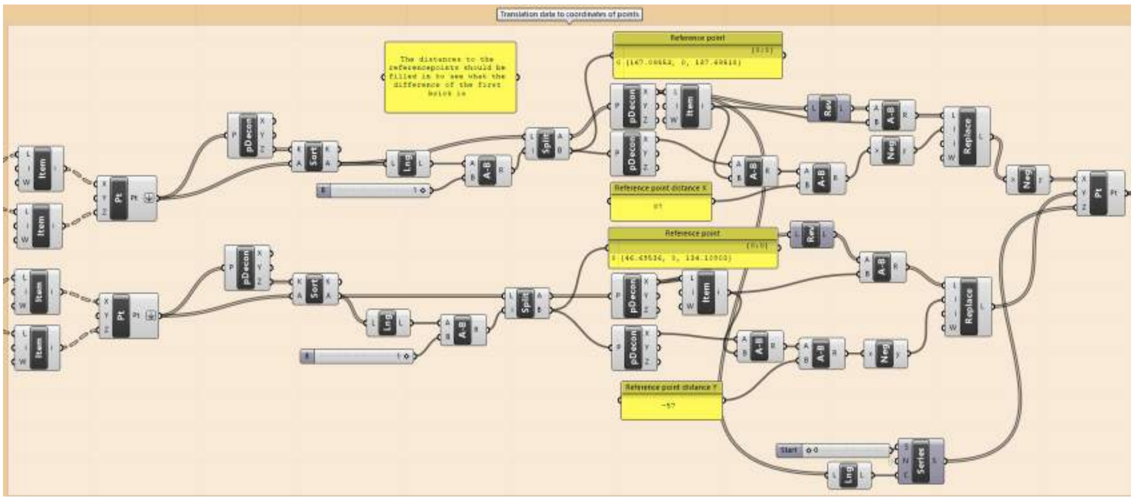


Figure A.25: Transfer of Excel data to coordinates

The coordinates of the centre of masses of the measured bricks are used with an orient components to place bricks on these coordinates. A digital twin of the built structure is then made. The same information is given on this structure as on the initial designed structure, being the coordinates of the corner points of the bricks and the centre of masses of the bricks. Lastly, the length of the datatree of the built structure is taken to get the same structure in amount of bricks from the initially designed structure. When this is not done the entire designed structure is compared to the built structure. This is a bit unpractical since they could not have the same height and comparing the two then might give some incorrect information. When the length of the initially designed structure is set as the same as the built structure, the comparison is made with the information that is needed at that moment in the process.

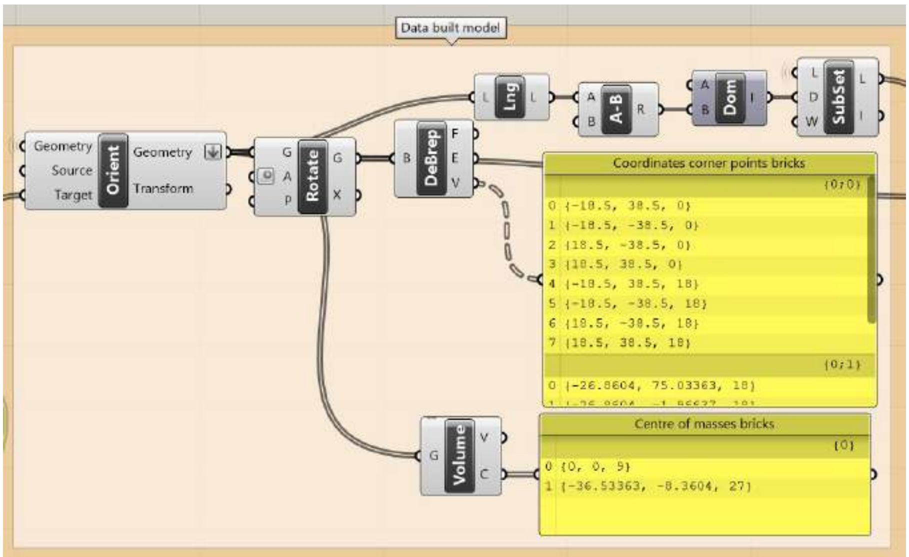
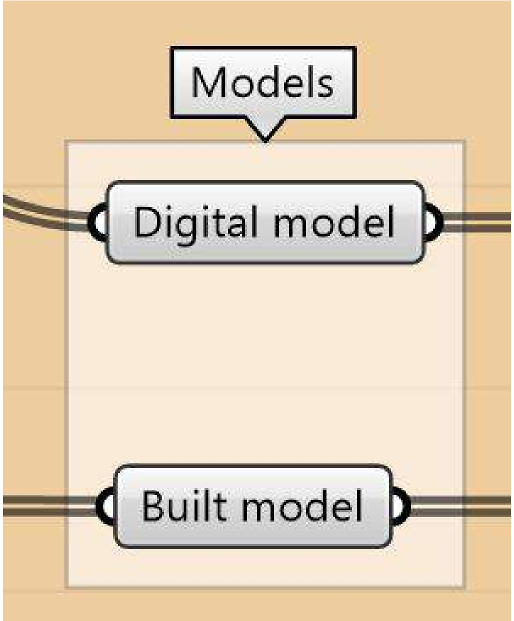
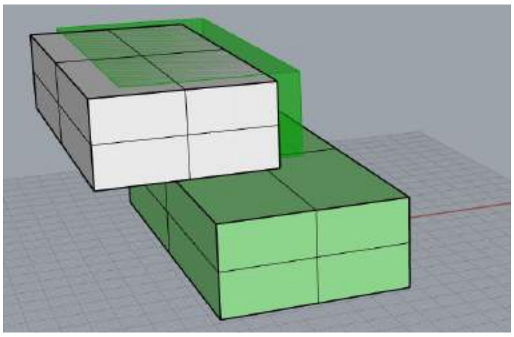


Figure A.26: Data from the built model

The two models, with the same amount of bricks are defined so that they can be compared in Grasshopper.



(a) Digital models



(b) Visualized digital models (white is designed, green is built)

Figure A.27: Digital models

In the comparison between the two models, the centre of masses are compared to one another since this is the most important information which is also measured. The Grasshopper file shows the differences in both the X-direction and the Y-direction for the centre of masses of the bricks that are already placed.

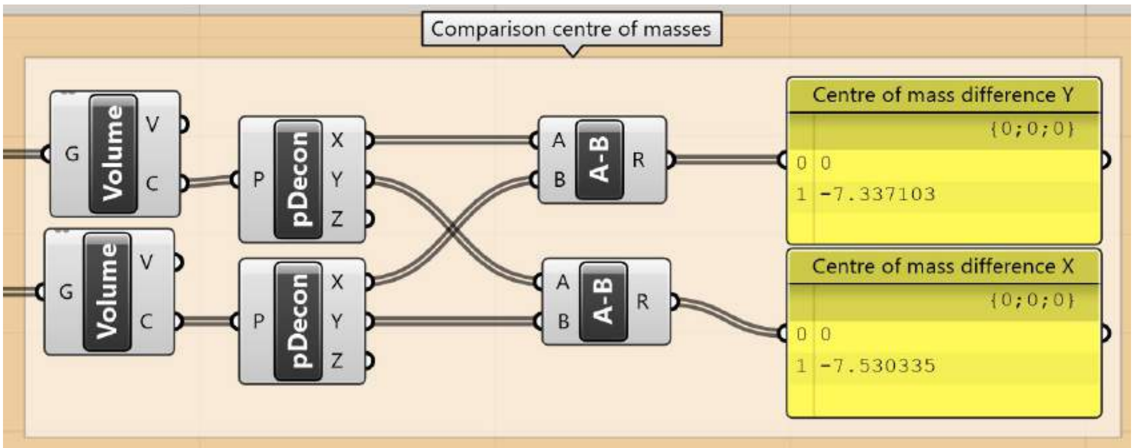


Figure A.28: Comparison between the two models

A.5 Data conclusion

As explained in Section 3.6 a conclusion on whether or not to built further could be made based on one next brick or the whole remaining structure. Therefore this part is split into two smaller parts. The first being the data conclusion based on one next brick and the second being the data conclusion part based on the whole remaining structure.

A.5.1 Data conclusion based on one brick

For the first data conclusion the next brick according to the initial design is added to the digital version of the built structure.

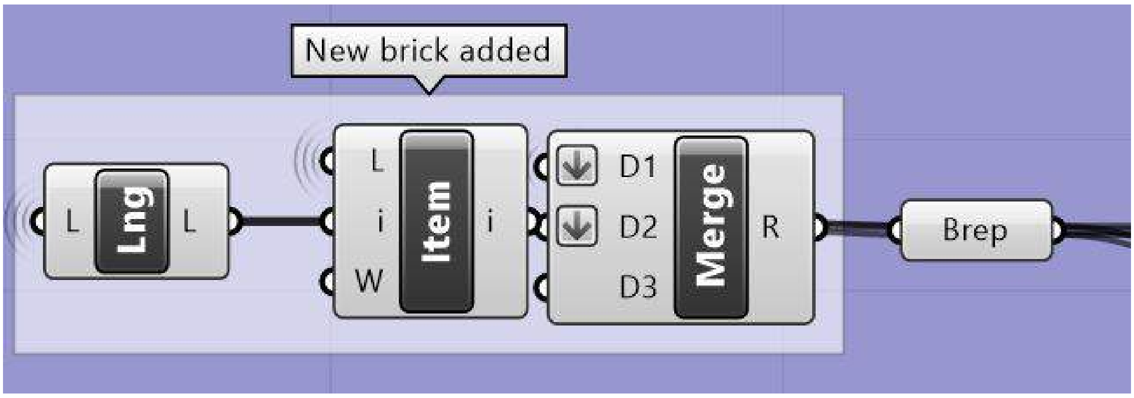


Figure A.29: Addition of one brick from the initially designed structure

This structure, the combination of the built structure and one new brick from the designed structure (in Figure A.30 the lowest 2 bricks are already built and the third is added according to the initial design), is then subjected to the same structural check as the initial structure was. This structural check was already explained in detail in Section A.2 so that is not done again. However, when at least one of the structural checks fails, further action is taken in the Grasshopper model since this structure does not have the potential to be stable. If it would be the case that all the check do suffice, the new brick can be placed as it is and the coordinates for this new brick are directly communicated to the robot. All the further steps that are here explained are then not taken.

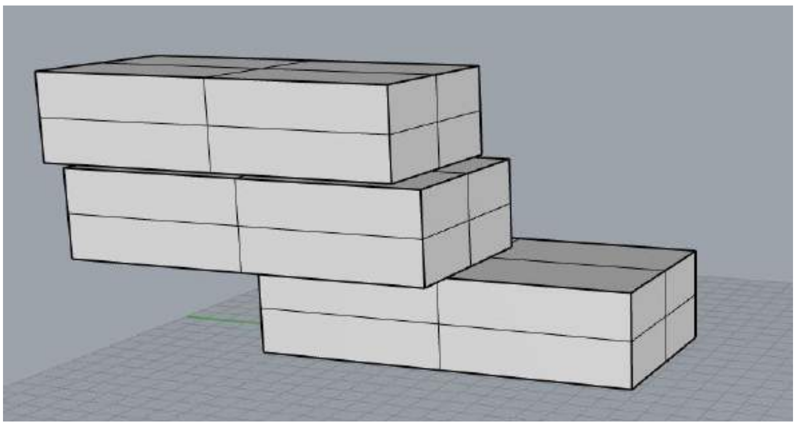


Figure A.30: Visualization of the newly added brick

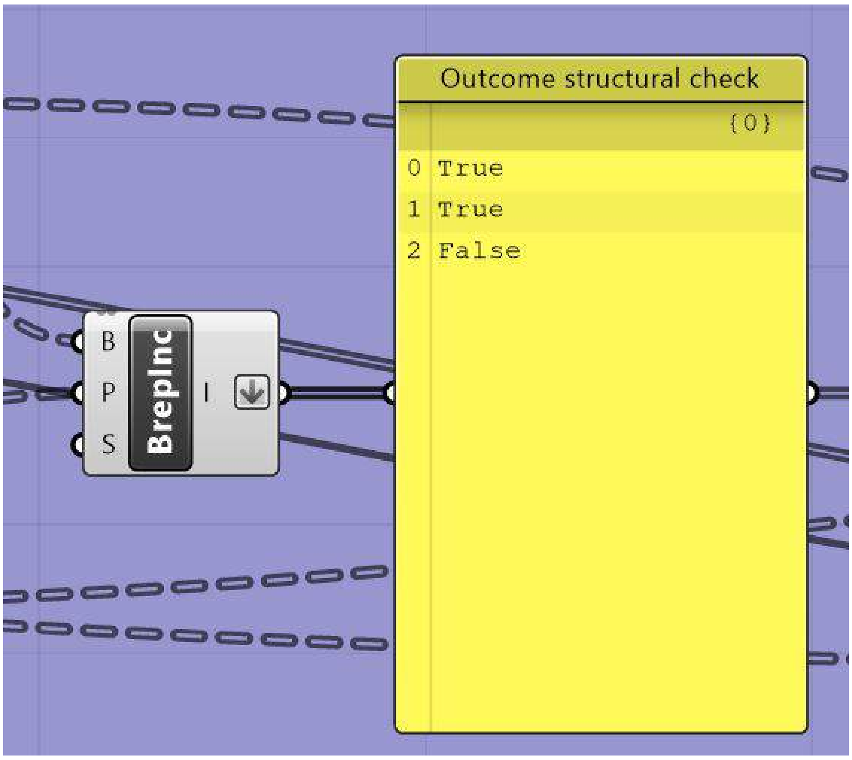


Figure A.31: Output of structural check

The same visualization is used to show which check fails, which is in this case always a check containing the new brick since that is the insecure factor here that is not placed yet. However, in comparison to the first structural check, now the length that the centre of mass is away from the safe boundary is more important and used further on the make an adjustment to the coordinates of the brick that will be placed next.

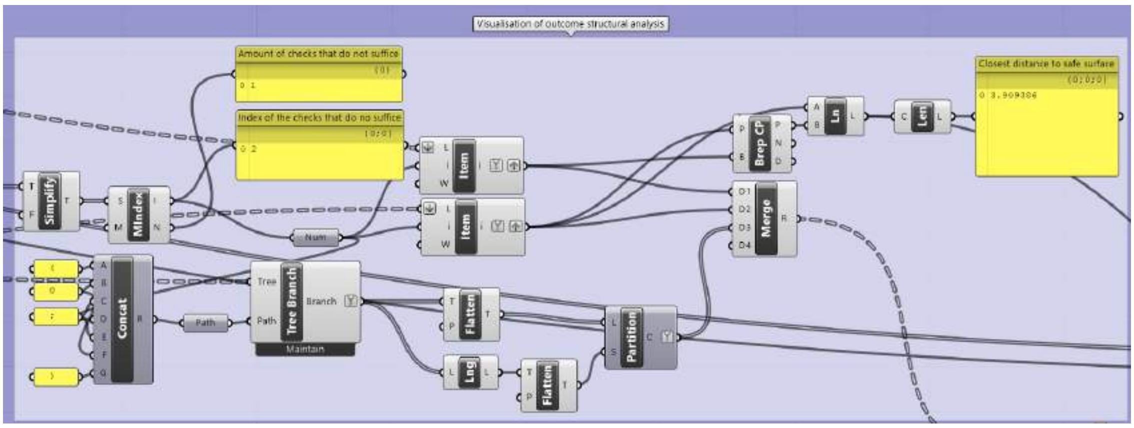


Figure A.32: Visualization of failed structural checks

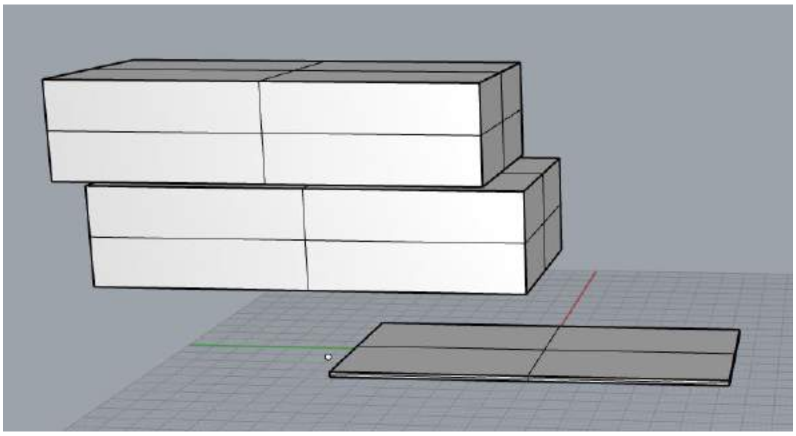


Figure A.33: Visualization of failed structural checks

The length between the centre of mass of the brick that caused the failed check and the safe boundary is taken to make an adjustment to the coordinate of the brick that will be placed next since this is the length to get to a sufficient check. This is therefore the length that the brick will move and be adjusted plus a really small length of 0,001 millimeters so that the next structural check does suffice since the centre of mass will then fall inside the safe boundary instead of on the edge. This structural check takes the same margin component into account as the initial structural check. The Grasshopper model itself determines to which side the brick should be moved from the length to the safe boundary and the centre of mass being positive or negative. The brick is then moved and a new structure is formed.

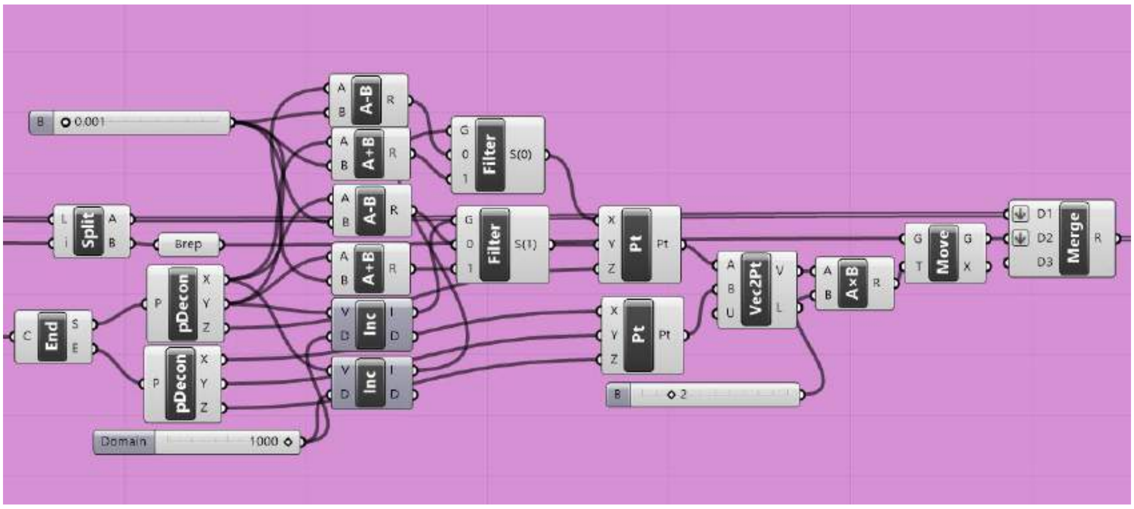


Figure A.34: Adjustment to new brick

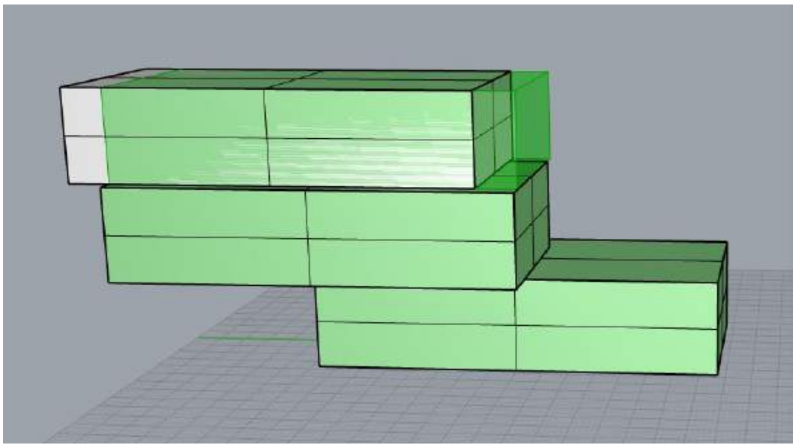


Figure A.35: Visualization of adjustment to new brick (green is adjusted)

This new structure is one last time subjected to the structural check, which is again the exact same as the first structural check that was explained in detail. This structural check should than always come back as sufficient since the checks that failed are now corrected.

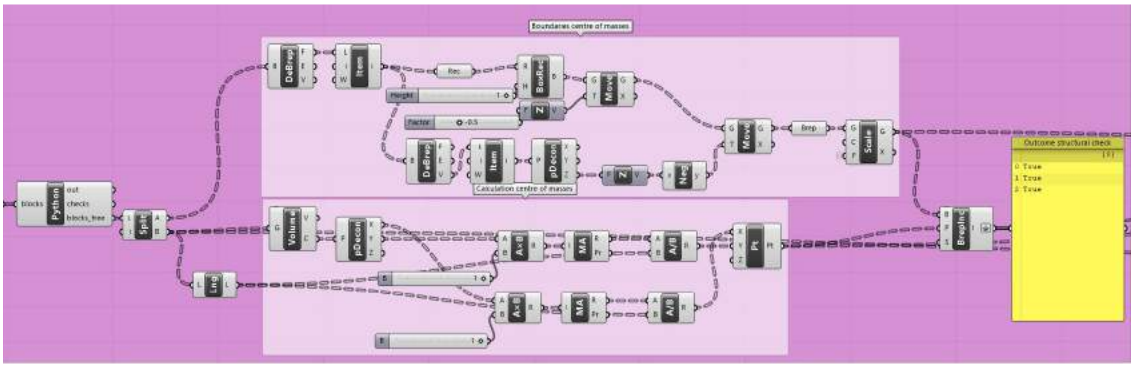
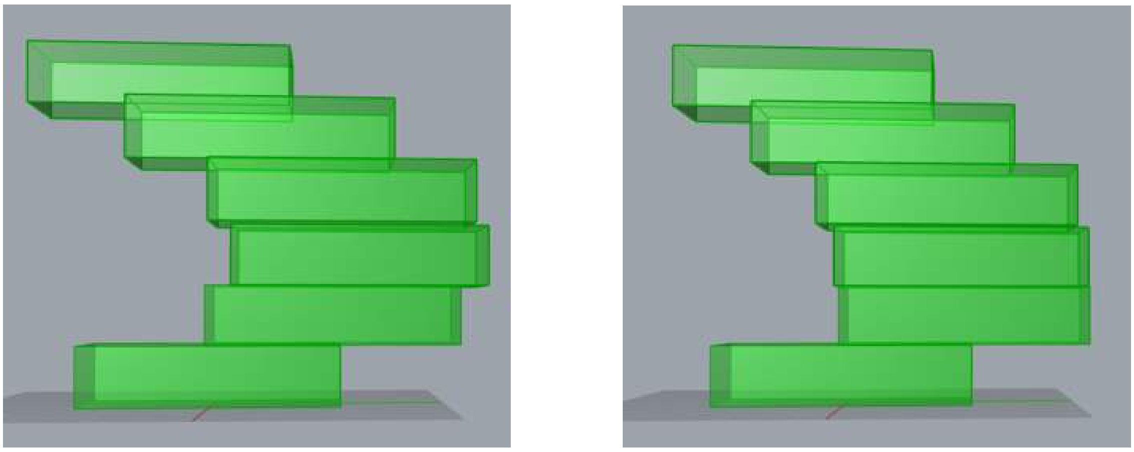


Figure A.36: Structural check of adjusted structure

A.5.2 Data conclusion based on remaining structure

The other way of drawing a conclusion on how to build further is looking at the whole remaining structure instead of just the next brick. As explained in Section 3.6 it could be the case, when looking at the whole remaining structure that there are multiple failed checks. The same optimization tool as explained in Section A.3 could then be used. However, now the bricks that were already placed will be permanent and only the other remaining bricks of the initial structure will be used for this optimization. An example is given in Figure A.37a where the first two bricks are built and therefore taken as permanent. The rest of the bricks are taken from this initial design of the structure. Figure A.37b shows the same first two bricks but now with the optimized version of the remaining bricks from the design of the structure. This last structure does not give any failed checks and can therefore be used to built further. The main advantages from this approach in contrast to the first one is the fact that the framework looks at the whole structure and it could be the case that more adjustments are prevented by looking at the whole structure. When looking per brick there could be unexpected adjustments after every brick when just one brick for example deviates from the design.



(a) Built bricks (first 2) plus initial remaining structure

(b) Built bricks (first 2) plus optimized remaining structure

Figure A.37: Optimization of remaining structure

In the Grasshopper model only a dataflow is added to the already existing optimization part of the script. When there is data from the Excel files, the built structure is taken and the remaining initial structure is put on top of this structure to be used for the optimization. If there is no data from the Excel files, the whole initial structure is used for the optimization which is required at the beginning of the process when the design is not sure yet.

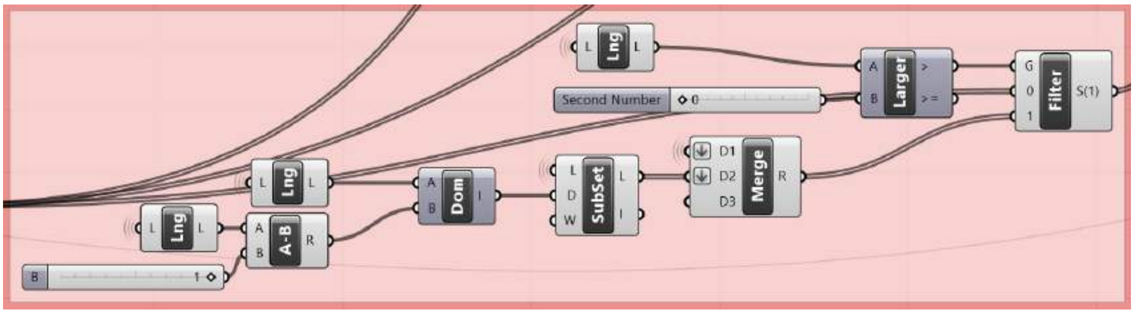


Figure A.38: Dataflow optimization

A.6 Translation to the robot

The structure, whether it is the originally designed structure or the adjusted structure, should be built by the robot. In order for the framework to select the right coordinates, the coordinates that should be built, a small filter part is added to the framework. The output of this filter component is one coordinate for the brick that is to be placed next, original, adjusted or optimized. This is done in this way since after every brick the brick will be measured and analysed. It is therefore not entirely sure that the next brick has the exact coordinates of the next brick of the initially designed structure. In this way it is also easier in the framework to overwrite the targets and see what the next brick that will be built is since this is updated after every brick. The filter components select the next brick from the initially designed structure if there is no failed check at the data processing part of the framework (Section A.4). If there is a failed check at the data processing part of the framework, the filter component will select the adjusted brick that is the outcome of the data conclusion part of the framework (Section A.5). If the optimization tool is used in the framework, not the adjusted next brick is taken but the next brick from the optimized structure is taken. In this way every time one brick is placed, the coordinates of the correct brick that is to be placed next are given to robot.

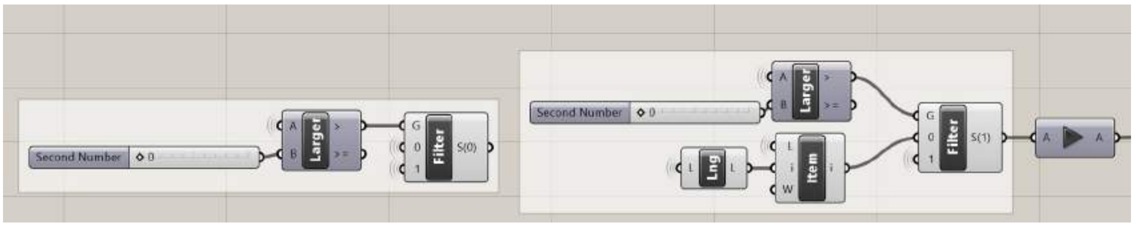


Figure A.39: Robot target filter component

When the 'correct' next brick is selected by the previous filter component, the brick can be prepared for the robot. This means that the brick should be moved to a certain location in front of the robot and already two points are defined, one point being the top of the brick on the exact location where the brick should be placed and one point being just a bit higher than the top of the brick.

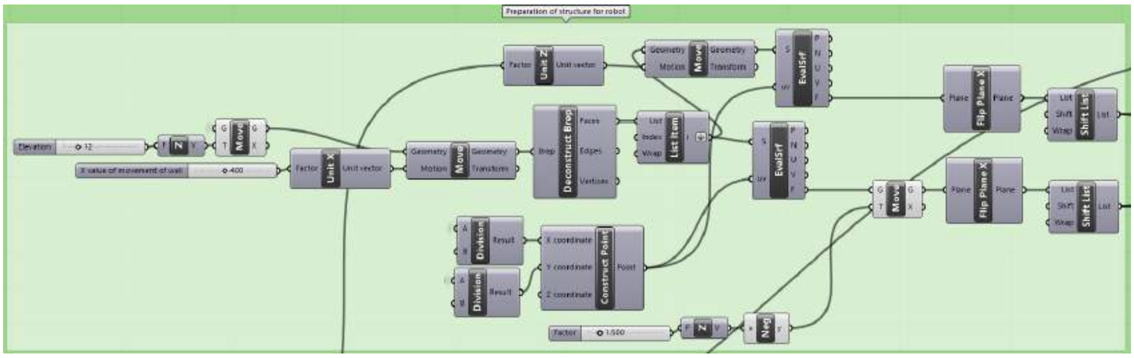


Figure A.40: Preparation of brick for robot

In the beginning of the robot part of the framework a set of fabrication parameters are defined that are used in all different parts of the robot part of the framework. These fabrication parameters contain the starting point, the point where the bricks will be picked-up, different Z-coordinates for safe points just over the bricks and high points where the robot can more freely move, different speeds for linear and absolute joint movements, precision parameters and a name for the digital output that is used.

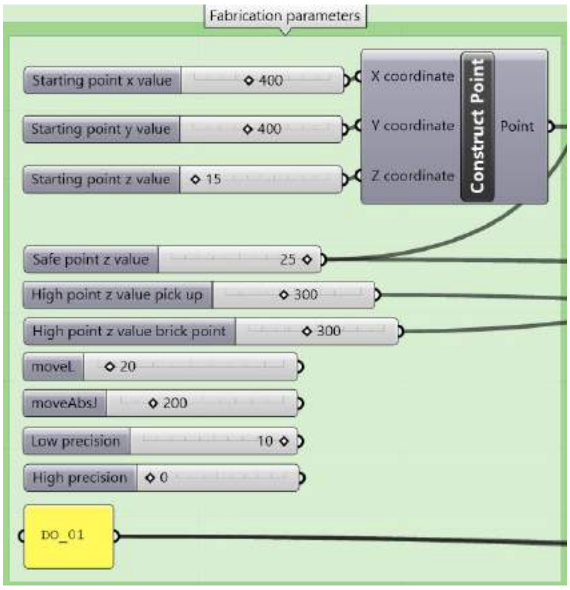


Figure A.41: Fabrication parameters

A starting point, from which the robot will always starts its path, is defined as the first point of the robot path. This starting point is a central point where all the axis of the robot are put to almost zero.

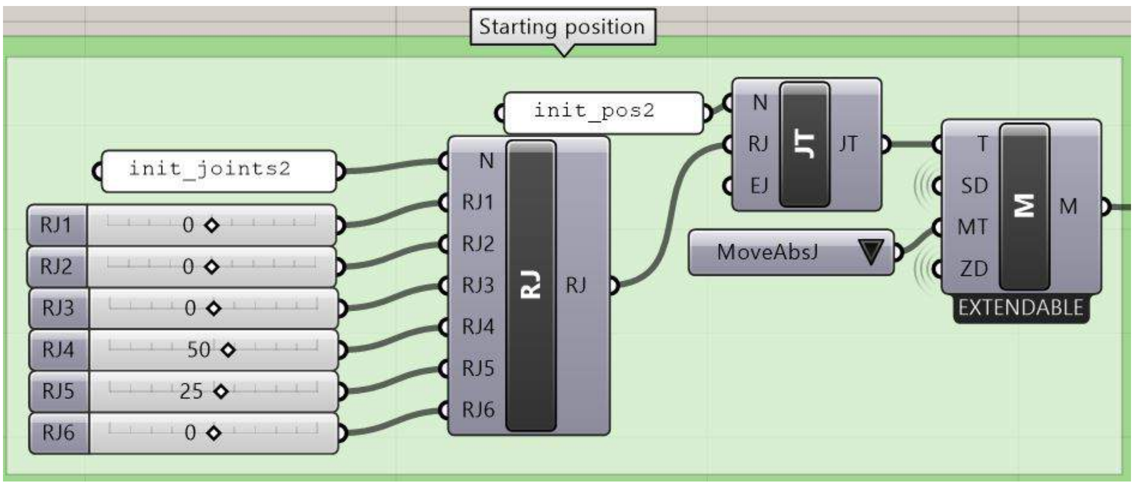


Figure A.42: Starting point of robot path

When the starting point of the robot path is defined it is time to define the targets of the brick. Later on all the targets will be put in the correct order so in this part of the framework it is important to define all the needed targets first. With the preparation of the robot (Figure A.40) already to different points where created, one being the top of the brick itself and one being a bit higher than the top of the brick. These points are used in this part of the framework as brick point exact and brick point edge. One last brick target is added being brick point high, this point is higher than the other two points. For both the brick point edge and brick point high two fabrication parameters are used, respectively safe point z value and high point z value brick point. These three brick targets are defined in this way since the robot movement that is higher and further away from the brick can be done with less precision and at a higher speed. The robot will start moving slower and with more precision between brick point edge and brick point exact since here it comes really precise where the brick is placed.

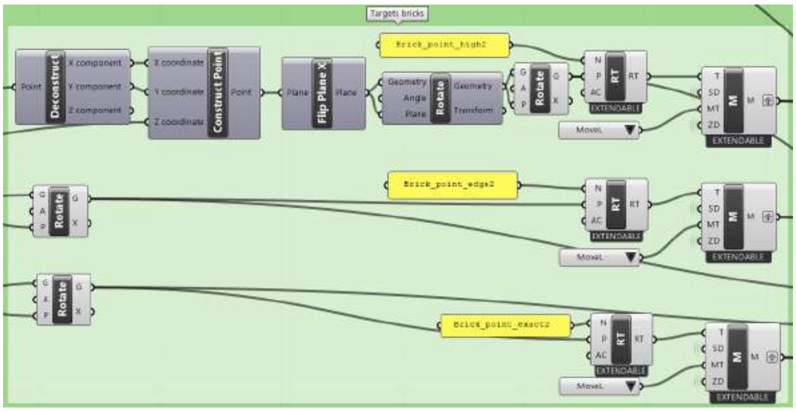


Figure A.43: Targets for bricks

The targets for the pick-up point, the point where the robot picks up a brick, are defined in the same way as the targets for the brick, with pick up point being the exact pick up point where the robot will move slower and more precise, pick up point edge being the from from which the robot can start moving faster going upwards and slower moving downwards and pick up point high where the robot moves faster and with less precision. This last target, pick up point high, is connected to the target brick point high so that the robot path is constructed is a logic manner.

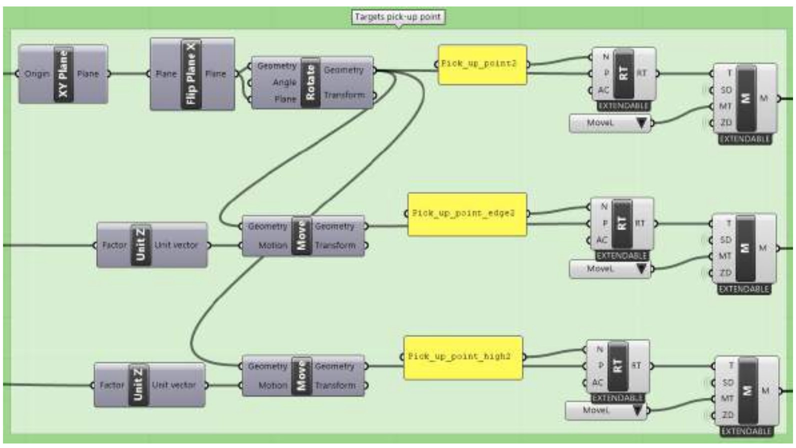


Figure A.44: Targets for pick-up point

Lastly, before the robot path is composed, the actions for the end-effector are defined. The digital output is once set at true and once set at false. These two actions will be added to the robot path for when the end-effector should be turned on or off. There is also a waiting time of one second added. This is done to make sure that the robot takes and places the bricks at the correct point since the robot is not doing any other movement during that one second.

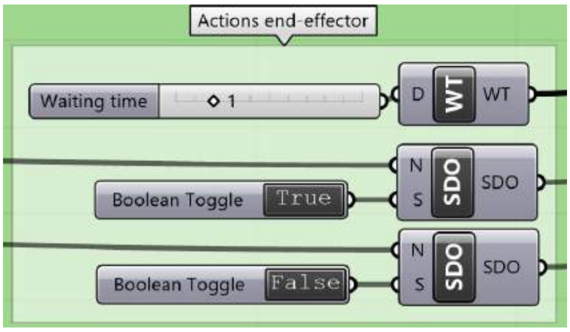


Figure A.45: Actions end-effector

When all the targets and actions are defined, these items can be placed in the correct order to construct the robot path. This is done with the weave component. Additionally a merge component is used, this component merges the total robot path with the earlier defined starting point from Figure A.42 and a linear configuration control component. The order of the robot path that is constructed with the weave component, so without the starting point, can be found below.

```
0.  Pick-up point high                  9.  Brick point high
1.  Pick-up point edge                  10. Brick point edge
2.  Pick-up point exact                 11. Brick point exact
3.  Waiting time                        12. Waiting time
4.  Set digital output to true          13. Set digital output to false
5.  Waiting time                        14. Waiting time
6.  Pick-up point exact                 15. Brick point exact
7.  Pick up point edge                  16. Brick point edge
8.  Pick up point high                  17. Brick point high
```

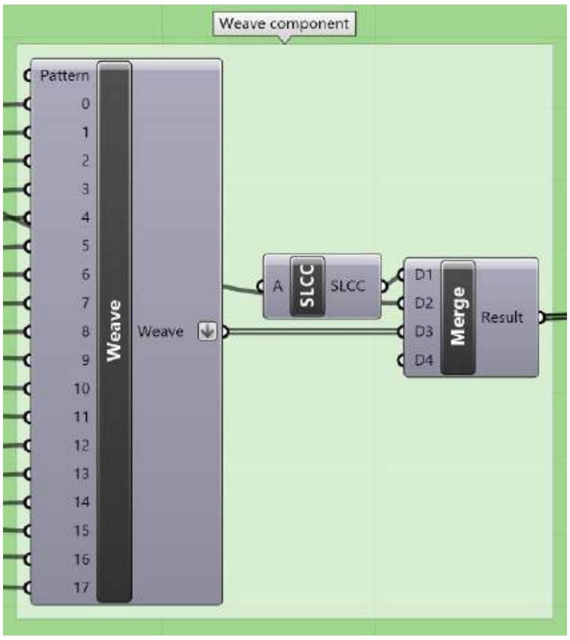


Figure A.46: Composition robot path

In order to get all the right information in the rapid code, that is made in Grasshopper and used in Robot studio, the robot end-effector is added into the Grasshopper script. In this case the vacuum tool is used and it is imported as a mesh.

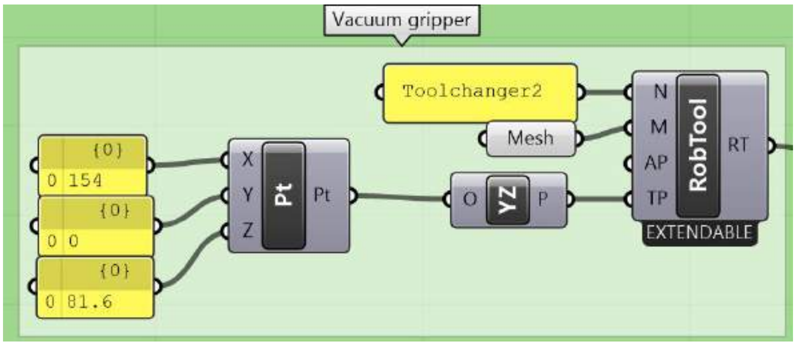


Figure A.47: Tool definition: vacuum gripper

The last part of this part of the framework is the part where the main and base code are generated. These codes are used in Robotstudio to actually control the robot. An example of both codes can be seen on the next pages. This last part also contains the robot info and a path generator. With this path generator the composed robot path is simulated and shown in Grasshopper. The robot path should also be checked within Robotstudio to account for possible errors but this path generator also filters out already some unwanted movements.

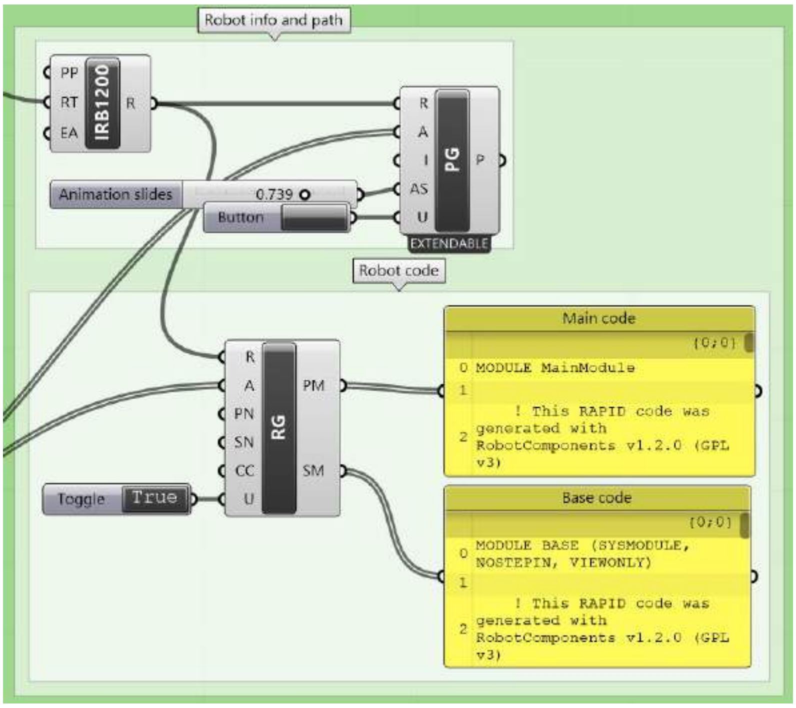


Figure A.48: Robot code and robot path

```
MODULE BASE (SYSMODULE, NOSTEPIN, VIEWONLY)

    ! This RAPID code was generated with RobotComponents v1.2.0 (GPL v3)
    ! Visit www.github.com/RobotComponents for more information

    ! System module with basic predefined system data
    ! ************************************************

    ! System data tool0, wobj0 and load0
    ! Do not translate or delete tool0, wobj0, load0
    PERS tooldata tool0 := [TRUE, [[0, 0, 0], [1, 0, 0, 0]], [0.001, [0, 0, 0.001],
        [1, 0, 0, 0], 0, 0, 0]];
    PERS wobjdata wobj0 := [FALSE, TRUE, "", [[0, 0, 0], [1, 0, 0, 0]], [[0, 0,
        0], [1, 0, 0, 0]]];
    PERS loaddata load0 := [0.001, [0, 0, 0.001], [1, 0, 0, 0], 0, 0, 0];

    ! User defined tooldata
    PERS tooldata Toolchanger2 := [TRUE, [[154, 0, 81.6], [0.5, 0.5, 0.5, 0.5]],
        [0.001, [0, 0, 0.001], [1, 0, 0, 0], 0, 0, 0]];

ENDMODULE
```

```
MODULE MainModule

    ! This RAPID code was generated with RobotComponents v1.2.0 (GPL v3)
    ! Visit www.github.com/RobotComponents for more information

    ! Declarations generated by Robot Components
    CONST robjoint init_joints2 := [0, 0, 0, 50, 25, 0];
    VAR jointtarget init_pos2 := [init_joints2, [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Brick_point_edge2 := [[395.4, 36.61, 91], [0, 0, 1, 0],
        [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Brick_point_exact2 := [[395.4, 36.61, 64.5], [0, 0, 1, 0],
        [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Brick_point_high2 := [[395.4, 36.61, 300], [0, 0, 1, 0],
        [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Pick_up_point_edge2 := [[400, 400, 40], [0, 0.707107, 0.707107,
        0], [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Pick_up_point_high2 := [[400, 400, 315], [0, 0.707107, 0.707107,
        0], [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
    VAR robtarget Pick_up_point2 := [[400, 400, 15], [0, 0.707107, 0.707107, 0],
        [0,0,0,0], [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];

    PROC main()
        ConfL\off;
        MoveAbsJ init_pos2, v200, z10, Toolchanger2\WObj:=wobj0;
        MoveL Pick_up_point_high2, v200, z10, Toolchanger2\WObj:=wobj0;
        MoveL Pick_up_point_edge2, v200, z0, Toolchanger2\WObj:=wobj0;
        MoveL Pick_up_point2, v20, z0, Toolchanger2\WObj:=wobj0;
        WaitTime 1;
        SetDO DO_01, 1;
        WaitTime 1;
        MoveL Pick_up_point2, v20, z0, Toolchanger2\WObj:=wobj0;
        MoveL Pick_up_point_edge2, v200, z0, Toolchanger2\WObj:=wobj0;
        MoveL Pick_up_point_high2, v200, z10, Toolchanger2\WObj:=wobj0;
        MoveL Brick_point_high2, v200, z10, Toolchanger2\WObj:=wobj0;
        MoveL Brick_point_edge2, v200, z0, Toolchanger2\WObj:=wobj0;
        MoveL Brick_point_exact2, v20, z0, Toolchanger2\WObj:=wobj0;
        WaitTime 1;
        SetDO DO_01, 0;
        WaitTime 1;
        MoveL Brick_point_exact2, v20, z0, Toolchanger2\WObj:=wobj0;
        MoveL Brick_point_edge2, v200, z0, Toolchanger2\WObj:=wobj0;
        MoveL Brick_point_high2, v200, z10, Toolchanger2\WObj:=wobj0;
    ENDPROC

ENDMODULE
```

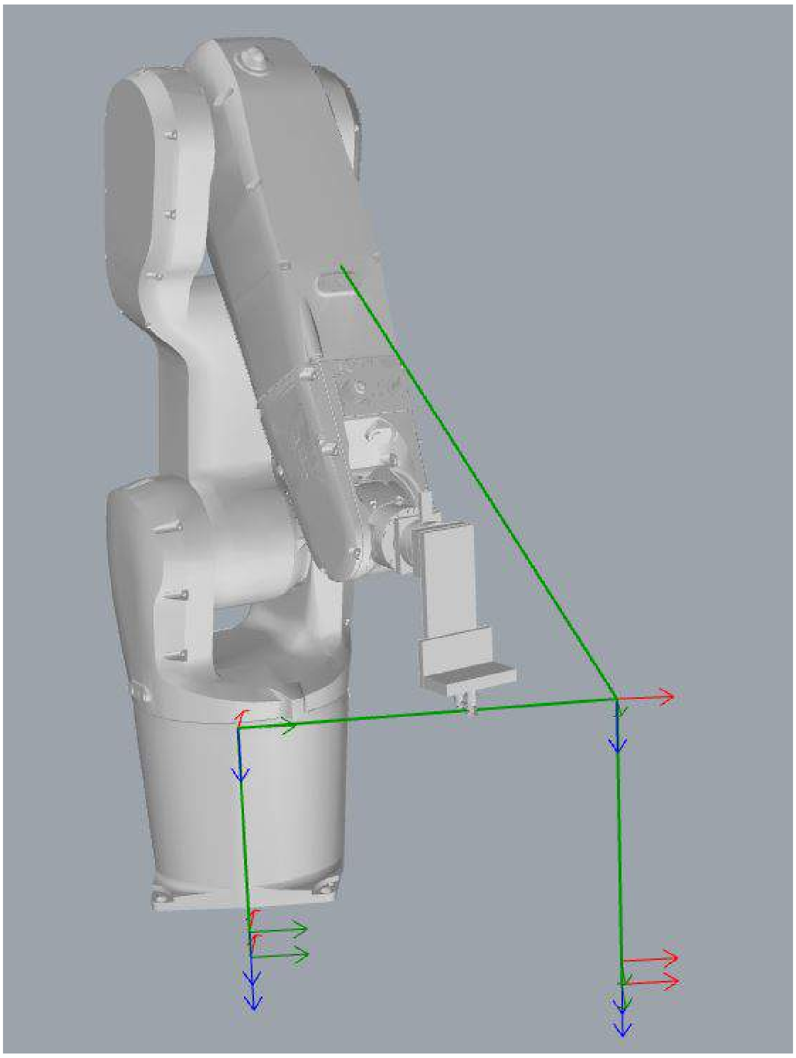


Figure A.49: Robot path with targets

Appendix B

Margin component calculation

As mentioned in Chapter 6 the margin component is a point of discussion. During the research it was the idea to calculate the margin component from gathered data of the final tests. This data is once again added in this appendix for the total overview but was already introduced in Section 4.2. The data that was to be used are the differences between the digital and the built model is both X and Y direction.

	Differences in mm X-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
1,00	0,32	-0,39	-0,76	-0,98	-0,32	0,43
1,25	0,32	-0,35	-0,78	-1,21	-0,35	0,51
1,50	0,17	-0,25	-0,77	-1,25	-0,20	0,67
1,75	-0,40	-0,22	-0,80	-1,52	-	-
2,00	-0,41	-0,23	-0,91	-	-	-

Table B.1: Differences final test and digital model data in mm (X-direction)

	Differences in mm Y-direction					
Amplitude	**Brick 1**	**Brick 2**	**Brick 3**	**Brick 4**	**Brick 5**	**Brick 6**
0,50	-0,07	-1,53	-1,27	-0,65	-0,18	0,28
0,75	0,03	-1,95	-1,44	-0,53	-0,36	0,086
1,00	-0,12	-2,50	-1,57	-0,24	-0,22	0,08
1,25	0,12	-2,86	-1,86	-0,53	-	-
1,50	0,05	-2,81	-1,94	0,03	-	-

Table B.2: Differences final test and digital model data in mm (Y-direction)

The idea was to use this data in combination with formula's to calculate a security of 95%.

$$Lowerlimit = \chi - 1,96 * \frac{\sigma}{\sqrt{n}} \tag{B.1}$$

$$Upperlimit = \chi + 1,96 * \frac{\sigma}{\sqrt{n}} \tag{B.2}$$

Within these equation χ is the average of the data sample, σ is the standard deviation of the data sample and n is number of samples.
However, when using this data to calculate the lower and upper limits of the margins it became clear that is was not as easy as first thought to get to one value for this margin component. When, for example, the lower limit for the X-amplitudes is calculated, the margin component would come to a number of -0,38875 millimeters. The accompanying upper limit would come to a value of

-0.38373 millimeters both resulting in around a margin component of 99,5%. This means that the safe boundaries used for the structural analysis are scaled to being 99,5% of the original boundaries for the X-direction. When doing the same for the Y-direction, the numbers deviate a lot as from the numbers mentioned for the X-direction. The lower limit would then be -0.84844 millimeters and the upper limit -0.83979 millimeters (bringing the margin component to 97,73 %). The X and the Y direction having a different margin component would not be a limitation since this could be made in Grasshopper, that the X side of the safe boundaries are scaled differently than the Y side. However, the biggest disadvantage is the fact that when this margin component is added, a lot of the results of Tables B.1 and B.2 are outside of this margin and will therefore still cause a collapse. Simply said, the differences between the two models differ quite a lot in one table resulting into a margin component value that might be too conservative for some amplitudes or bricks and too risky for other. Also, considering the causes to the differences between the models explained in Chapter 6, it could be the case that this data set is not representative for the actual differences between the two models cause by the robot. This means that the differences shown here could be caused by the robot or a lot of other causes, making it unclear what part of the differences between the models should be used for the margin component. In conclusion, using this data for determining the margin component was decided not to do since it would still arise a lot of questions. Therefore, it is recommended that more research can be conducted to determine such a margin component where is should also be kept in mind that a margin component per direction, per brick or per region commpared to the camera is an option.