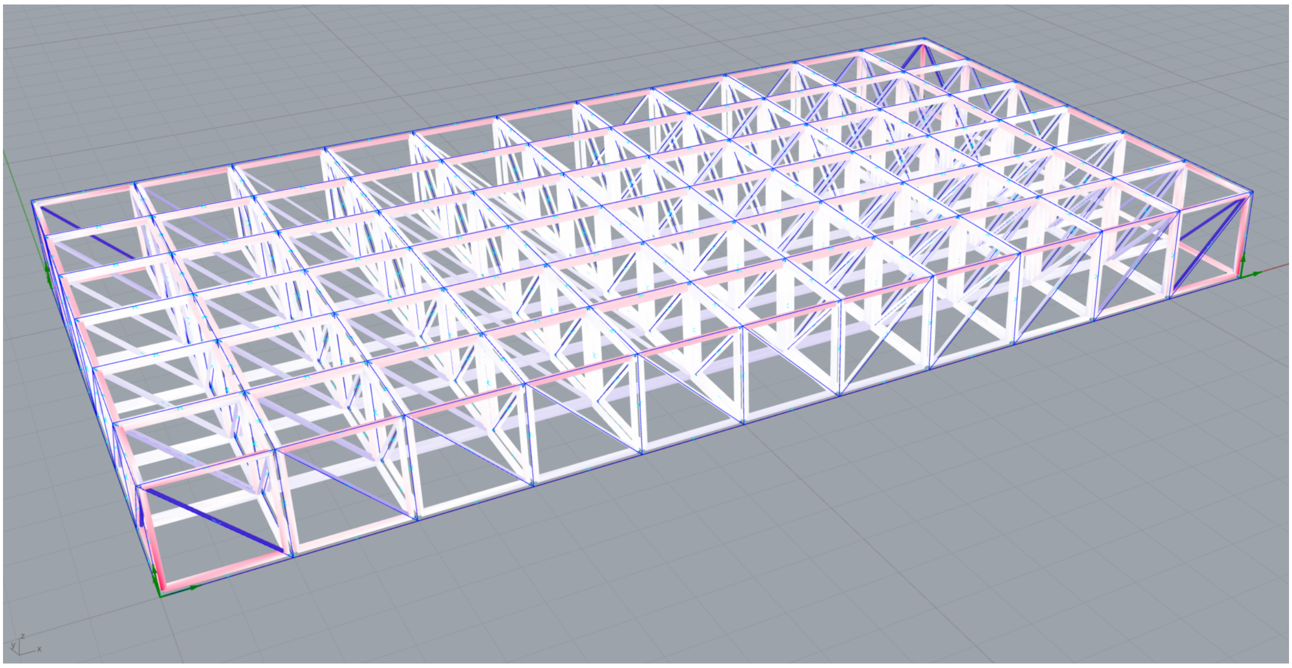
Structural design and optimisation of a topologically reconfigurable modular steel space frame system

Ruben Koopman

October 2024



TU Delft

Structural design and optimisation of a topologically reconfigurable modular steel space frame system

by

Ruben Koopman
(4893247)

in partial fulfilment of the requirements for the degree of
Master of Science
in Civil Engineering
Delft University of Technology

Thesis Committee:	Dr. R. Oval	TU Delft, Supervisor
	Dr. F. Kavoura	TU Delft, Supervisor

TUDelft

Preface

This thesis is the result of a two-year-long master track in structural engineering and marks the end of my time as a student at the Faculty of Civil Engineering at TU Delft where I have studied for more than six years. It also marks the beginning of my life as an engineer. The project was difficult but also fun and I learned a lot of valuable lessons. Sometimes this subject reminded me of my biology classes from high school due to all the terminology around genetic algorithms and the parametric software program "Grasshopper" that uses a lot of plugins with names of animals. I presume this does not get better. Computational design is something that I wanted to learn during my studies and I am happy that I have learned a lot about it with this thesis.

During my time at this university, I have gained a lot of experience and I made friends and connections who helped me arrive at the place where I am today. I would like to thank them all for their help and support. It has been a great time thanks to all of you. First I would like to thank the committee members who guided me during the process of writing this thesis. Robin Oval helped me a lot with structuring my work and also introduced me to the topic in the beginning. He was my daily advisor and I could always come by for questions. I enjoyed our conversations a lot and your knowledge on the subject brought the project to a much higher level. I am very grateful for the support and supervision. To Florentia Kavoura I am also very grateful. Without her input on the design of steel connections, the design of a complete "system" would not have been possible. Thank you for your time and expertise.

I also made a lot of friends at the faculty, which I would like to thank. Several of them I have known for years now and are very close to me. Thank you for all the times we had coffee during the breaks, and see you soon.

Last but not least I would like to thank my family who have always supported me. My brothers were always by my side and my parents have given me the best support during my entire academic career. I am looking forward to the future.

Ruben Koopman
Delft, October 2024

Abstract

In many structures, space frames are used as the main load-bearing system. Especially for structures that are designed to have large spans or free-form in design. The reason is that space frames have a very high structural performance. Besides, they are also material and cost-efficient. Challenges for space frames are that they often require complex designs and the elements used are unique. Joint design is also difficult for these irregular constructions. This thesis explores the possibility of structurally optimising space frame design by using topologically reconfigurable modules, taking into account circularity. The focus lies on planar, square on square double-layered grids. The research question to be answered is the following:

"What kind of topologically reconfigurable modular system enables the generation of efficient space frames that are suitable for circular construction?"

The first design step is the initial topological design of steel cubic modules. This forms the basis of the catalog. With Grasshopper and Karamba3D (parametric FEM software) single-span trusses are then designed to determine the required cross-sections of the elements. The beams and columns are given SHS cross-sections for different spans and the diagonals are given various CHS cross-sections. With the software IDEA StatiCa various intra- and intermodular joints are designed and the rotational and translational stifffnesses analysed for the different modules.

The former Grasshopper model is extended with a GA called Galapagos which is a plugin just like Karamba3D. This is used for topological optimisation of a given space frame structure. A GA is an evolutionary algorithm that is inspired by the natural selection process that Darwin described. The fittest genes in every population of solutions are used in further iterations. This process continues until the total number of iterations is reached or a threshold is met [1]. The algorithm reconfigures the different modules and minimises the weight of the structure while staying within the constraints of maximum deflection, maximum material utilisation, and avoiding buckling. Logarithmic barrier functions are used for these constraints and together with the minimisation of the weight a fitness function is made. The lower the value of the fitness function, the better the solution is. The joint stiffness for each of the joint types for different |M|/N ratios is known. In the model, an initial stiffness is assigned to the joints. Then, depending on the moments and forces found in the joints, a new stiffness is assigned corresponding with the relations found in the stiffness analysis. The resulting stiffness loop describes non-linear behavior.

A verification is performed for the joints in the model. It was observed that the model was sensitive to small differences in stiffness within the joints of a standard truss. There were torsional moments observed that also resulted in an uneven distribution of the forces in the truss. To overcome this problem the stiffness of all joint groups is changed at once instead of individually. This resulted in more logical results where the support reactions were symmetric again. The model is then verified with a simple structure to see if the performance of the algorithm matches the expectations. A simple 2 x 2 grid is constructed and the model is tested. The simulation could not be performed until 50 generations because the stiffness loop led to a large accumulation of memory on the computer. It was concluded that the loop was not needed and therefore omitted because mostly only 1 iteration was needed. The model then was tested again on the small-scale model and compared to a couple of intuitively good-performing structures. This resulted in the conclusion that the model performed well and converged towards a final solution. However, since there were 10^6 different configurations possible the best solution was not reached in the end. It is recommended to always perform a couple of simulations to get a good solution.

After verification, the model is validated with a different FEM software called RFEM 5. Single grasshopper modules are structurally analysed in Karamba3D and imported in RFEM as well. The results show similarity in the order of magnitude of the stresses and the way the stresses are distributed. Furthermore, the reaction forces are also similar. The percentual differences between both models for the minimum and maximum stress are all below 5% except for module type 4. The absolute differences are in the order of 0.01 kN/cm^2. Besides, a small frame is analysed in Karamba3D and RFEM as well to check if the deflection is in the right order of magnitude which is also the case. The percentual differences for the maximum and minimum stress are 6.40% and 7.85%, and for the deflection 12.15% and thus slightly larger than for the modules. However, for the deflection, the difference is just 1.5 mm for a structure with a span of 16 m which is a small difference. The Karamba3D model is valid and also conservative because the observed stresses are larger than in the RFEM model for the same load condition.

A literature case study is performed on a space frame located in India. It is compared to the topologically reconfigurable model to check if the order of magnitude of the deflections and stresses is normal for this type

of truss. First, a standard Pratt truss is modelled in Grasshopper to see if it is possible to design a feasible structure. This structure stayed within the limits of the constraints for the applied load which meant it worked. The case study truss is of a different size than the small structure used in the verification. With the catalog of 6 different modules and 66 possible locations for the modules as many as 10^{101} configurations would be possible. For this reason, the simulations are performed with 1 module type first (module type 6) and then with 2 (module types 3 and 4), greatly reducing the amount of possible configurations. These are respectively 10^{71} and 10^{19}. There are more solutions for the simulation with 1 module type because this module has a larger number of orientations than the other two modules combined. The outcome of the case study was that the algorithm did not find better or equal solutions than the truss structure. This was because the convergence of the model was very slow, even for a simulation of more than 200 generations (10050 solutions) the best solution had a fitness value much larger than the truss structures. The issue was mainly that the extreme utilisation of some of the members was too large which made them fail. The BLF and displacement were within the boundaries. Comparing it with the case study the maximum stresses were indeed larger but the deflection was in the same order of magnitude. The Pratt truss with reconfigurable members which resembled a regular truss was in the right order of magnitude for deflection as well. This truss does not have extra parallel beams since not entire cubes are joined together but single beams and columns.

An application is analysed to see how the model acts when the supports are irregular. From this setup, it was unclear what would be an optimal solution. The selected modules for this problem did not give the option to create a regular truss-like structure. The model found a slightly better solution than the one initially found using intuition and expert judgment. The value of the fitness function decreased from 0.84 to 0.80, which is a 5.5% decrease. The conclusion from this case study is that when the optimal solution is unclear and the design space is large the model can find a relatively good solution. Combining this expert judgment and computational design can improve a model even more.

Overall the model can be used to design planar steel space frames with varying support conditions. These space frames are circular in design because their parts can easily be deconstructed and reused in similar structures. This is a huge advantage because it can save a lot of material and construction time as well. The modules themselves have the extra advantage that they can also easily be changed to a different type by varying the diagonals. This makes the system flexible in design. However, the optimisation efficiency is not very good. The algorithm does not find optimal solutions that can be used in practice for a large design space. The type of structure is also much heavier in general compared to conventional space frames. This is because joining entire modular cubes together requires more steel. There are numerous parallel members that do not appear in regular space frames where beams, columns, and diagonals are individually placed in the structure. This makes stress on the members relatively high and therefore unfeasible. To use the system is therefore a consideration for the designer. When focussing on modular and circular construction this system is very useful but it is less structurally efficient. The model with "modular elements" could potentially be used for a more structurally efficient system if the convergence towards a solution can be improved (with a different algorithm) and the speed at which solutions are calculated as well.

Contents

List of Tables

List of Figures

List of abbreviations

API Application Programming Interface. 9, 29

BLF Buckling Load Factor. iii, viii, 42, 45, 46, 54, 55, 57, 60, 61, 63–65, 67, 68, 74, 76, 78, 80, 81, 83, 85, 87–89, 92, 93, 95

CAD Computer Aided Design. 4, 9, 10

CBFEM Component Based Finite Element Model. 29

CHS Circular Hollow Section. ii, vii, 16, 21, 29, 32, 33, 125, 131, 144

CMAES Covariance Matrix Adaptation Evolution Strategy. 100

FE Finite Element. 4, 5, 70

FEM Finite Elements Method. ii, 5, 9, 10, 16, 20, 29, 40, 49, 95, 99, 100, 124, 136

GA Genetic Algorithm. ii, 5, 9, 10, 19, 41, 42, 44, 50, 51, 55, 64, 84, 89, 95, 97, 98, 100

HSS Hollow Steel Section. viii, 109–111, 113

IMC Inter Modular Connection. 28, 100

IPCC Intergovernmental Panel on Climate Change. 8

IQR Interquartile Range. 37, 120

LC Load Combination. vi, 37, 74, 76, 89

LCA Life Cycle Assessment. 9

MILP Mixed Integer Linear Programming. 9, 99

NURBS Non-Uniform Rational B-Splines. 10

OWSJ Open Web Steel Joist. 15

RAM Random Access Memory. 56, 60, 61, 76, 77, 81

RBFOpt Radial Basis Function Optimisation. 100

SHS Square Hollow Section. ii, vii, ix, 16, 21, 27–29, 31, 32, 112, 125, 131, 133, 135, 138, 144

SLS Serviceability Limit State. 42, 73

SOS Square On Square. 12, 88, 96

ULS Ultimate Limit State. 42, 73

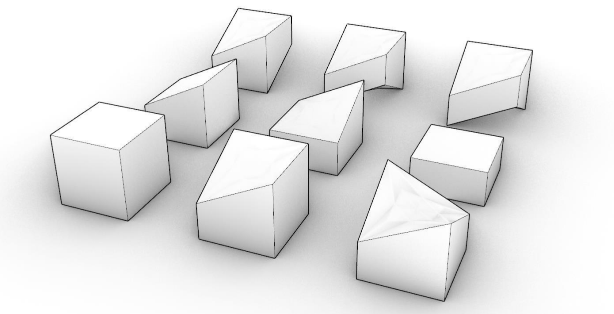
1 Introduction

Space grid structures originated in 1903 when Graham Bell assembled space structures from octahedral and tetrahedral units [2, 3]. After the introduction of the MERO system in 1943 space frame structures were commercialised [2, 3]. It was a method of construction that is still used today. It comprises tubular elements connected with ball-shaped joints. During the 1950's and 19060's space frame structures all over the world were introduced such as the space deck system, Triodetic system, Unistrut and Oktaplatte [2]. An overview of different worldwide used systems can be found in Appendix A. Many different systems and ways to design space frames are available today. Space frames are space structures, often made out of steel, that have bending stiff connections between members. This is unlike space trusses which have non-bending stiff connections such as pin joints [4]. Space structures can be very free-form in design as can be seen in Figure 1.1, which is an example of a single-layered grid. Furthermore, these structures are material and cost-efficient and have high structural performance [5]. They are often used to realise large spans and specific free-form structures. Furthermore, they are light because the material is distributed in such a way that most of the forces acting on the elements in the frame are axial forces and are distributed in three dimensions if designed properly [6]. This allows for an efficient load transfer. For this reason, space frames facilitate structural optimisation by material saving, however, they are complex in design. Due to the complex geometry, these types of structures cope with design challenges on design geometry, custom components, and difficulties during construction. The often unique custom components also limit the efficiency of the structure when it comes to the end-of-life stage. Reuse and circularity of such a custom design can be difficult because these parts might not be usable in other constructions. Engineers and architects wishing to reuse these special parts must take more time and be creative in design to integrate them. The reused material is not endless and provides the designer with a limited supply both in volume and in geometrical form [7]. Space frames can be designed by making use of modular design. They are built up from various modules with a certain configuration and repetition. These are the basic building blocks of the structure. This way of designing is becoming more and more common in today's world because of advancements in computational design tools which makes designing these complex structures easier. The development of quick tools for modular design is therefore a hot topic. Additionally, using parametric design the process of finding an efficient structure becomes more automated and speeds it up, additionally decreasing the costs. The research into modular design is an ongoing process worldwide. Quick tools like catalogs for 2D truss optimisation are being developed. Tyburec et al. developed a method to design modular trusses using Wang tiling encoding of the modules [8]. Tugilimana et al. introduces a continuous formulation of module rotation and topology optimisation also for 2D trusses. For 3D space frames these kinds of methods are not yet investigated in the literature.

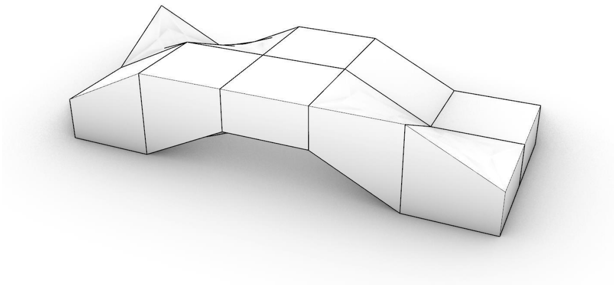
Figure 1.1: Heydar Aliyev Center (Source: Rethink The Future, 2022)

This thesis explores the potential of introducing modularity in the design and construction of space frames, in particular steel circular constructions. The aim is to develop a catalog of modules that can be reconfigured into different designs. Modularization increases their potential for the application of automated and circular construction. The focus will be on flat space frames, often used in roof designs. The aim is to design a complete system which includes joint design in the design phase. An example of how these modules could be reconfigured can be seen in Figure 1.2. The design of standard modules as well as the connections between these modules are important features. Both will be addressed in this thesis. This leads to the following research question:

"What kind of topologically reconfigurable modular system enables the generation of efficient space frames that are suitable for circular construction?"



(a) An initial catalog of modules for space frames as box models (Source: Robin Oval)



(b) Concept of module based assembly of space frames with varying curvature and thickness (Source: Robin Oval)

Figure 1.2: Initial catalog of modules for spaceframes and a concept of module based assembly of space frames with varying curvature and thickness

The main challenges to answering the research question lie in the development of the catalog, the design generation, and the detailing of the modules. Feasibility of the method is also something that must be investigated. Because of all these different aspects, it is not straightforward to answer the research question and it is therefore split into multiple sub-questions. Each of the sub-questions comprises a part of the answer. These sub-questions can be seen below.

1. What are the required parameters for the most optimal design of reconfigurable modular systems?
2. What is an efficient set of modulus/catalog for the modular design fabrication of steel circular constructions?
3. How can the modules of the catalog be connected, disconnected and reconnected efficiently in design?
4. What is a good computation method for (re)configuration exploration and structural optimisation of circular steel constructions?

Subquestion 1. is about the parameters needed for the most optimal design. This is important for the research because it provides the designer with a good overview of things to take into account. There are numerous parameters that have influence on a complex design and it is necessary to isolate the most important ones to keep the design straightforward. Besides, the importance of the different parameters for circularity or structural performance can be investigated. There are many of different design options for making modules. To make a design more circular and simple a small selection of smart options can be beneficial. Subquestion 2. is about this matter. Too many options for modules will make the design difficult. Subquestion 3. goes into more detail about the construction of the structure. The connections between the modules are very important. It will be one of the major tasks in this master thesis to design them correctly. Subquestion 4. is about the method that can be used to position the different modules in design space in an efficient way. Furthermore, the method also describes how the circularity of the structural components can be implemented in the design.

In the literature, numerous of different aspects of modularity and space frame design have been investigated already. For double layered gridshells often certain meshes are applied on both layers and then connected and optimised. Modules that are prefabricated such as the Unibat, ABBA Dekspace, Cubic Space Frame and Space Deck System exist but there is no literature on how these modules can be reconfigured and optimised for free-form structures [10]. Other modular systems can be found in Figure A.3 of Appendix A. This master thesis will contribute to the state of art by exploring this method for space frame design.

In the continuation of this report, a literature review is performed to find relevant information for the topic and to find the literature gap. More detailed literature can be found throughout this document and is not a separate section. Most sections are introduced with background information about the section's subject which comprises of state of art literature. In Section 2 the research methodology for this project is outlined. The focus of this thesis will be on planar space frames rather than free-form space frames. In Section 3 firstly a conceptual model is made. Then the design of the catalog of modules is performed. The different possible module sizes are determined. Furthermore, in Section 4 the design of joints is discussed. In Section 5 the optimisation process is discussed. In Section 6 the modelling workflow is outlined. After that, the model is verified in Section 7 and validated in Section 8. In Section 9 the model is compared to a literature case study to check if the model can make structures that are as good as or better than existing ones. In Section 10 a simple application is introduced to investigate the feasibility of the model for structures with irregular supports. In Section 11 the way of fabricating the modules and constructing the system on site are discussed. In Section 12, Section 13, and Section 14 the conclusion, discussion, and recommendations can be found.

2 Methodology

In this section, the research methodology is described. The ultimate goal of this research is to design and optimise a catalog of standard modules. These modules can be organised in certain configurations to design steel space frames with. To make a design a certain mesh tessellation has to be chosen. The options that are used in most practice are triangular and quadrilateral meshes in 2D. Hexagonal meshes are also often used. In 3D tetrahedra and hexahedra are therefore often used shapes [11]. For this research, the focus will be on designing standard modules in hexahedra because these are relatively simple building blocks which makes the design process also easier. It has to potential to be extended with other 3D shapes in later research. Making 3D structures with neighbouring hexahedra will result in two meshes above each other. A double-layered gridshell is made.

A big risk regarding the progress of the research is the complexity of the different steps involved in coming to a design. When initially a set of modules is designed and directly applied to a very complex free-form structure the risk is that some design issues are not addressed. Therefore, at the start of the research, the level of complexity will be kept low and will later be increased step by step. If complexity becomes too much a step back will be taken.

Step 1. literature review
The first step is to do a literature review. This is used to gain insight into the topic. It is necessary to make sure that the research is state-of-art and that a research gap is filled when the research is performed. This literature review will also help future research to learn about the state-of-art of the topic. The literature is reviewed at the beginning of the research but continually revisited and expanded as new literature is identified throughout the study. It can be seen as a process that acts in parallel with the other research steps. Therefore, the literature research is not a single section in this document. Literature studies are done for various sections.

Step 2. Catalog design
The second step is to design a variety of initial standard modules similar to the one shown in Figure 2.1. This is one of the steps that eventually leads to the module design which will form the final catalog. Cross-section and joint design will complete the catalog. At first, a conceptual model is presented that serves as a generic description of what is needed for the design in terms of parameters and relations. In this research, the combination of Grasshopper and Rhino3D will be used to make these designs. Rhino is a Computer Aided Design (CAD) software made available by TU Delft. Grasshopper is a plugin for Rhino and can be downloaded for free. When this is done a literature research is done to gain insight in what sizes are normally used for modules worldwide. With the obtained knowledge the catalog will be extended with information about different module sizes that can be used to design structures with different spans.

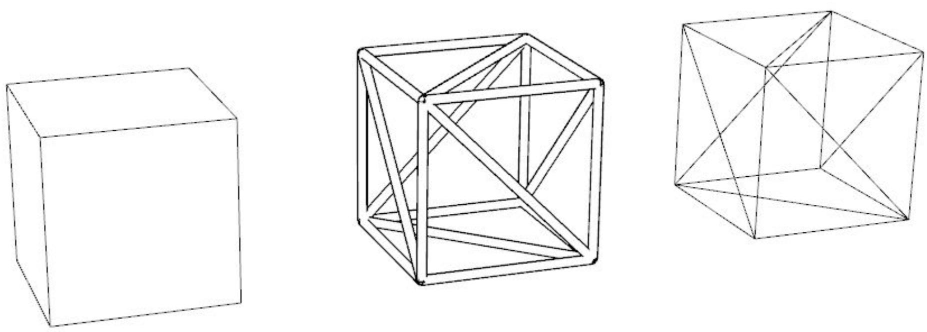


Figure 2.1: Different levels of modeling of a standard cube module: box, pipe, wireframe (Source: Robin Oval)

Step 3. Cross-section design
With a single module type from the catalog, a single-span structure will be designed in Grasshopper. For different module sizes that are found in step 2. there will be different spans covered by the structure. These spans correspond to the benchmark values that come with each module size. These structures will then be tested on their structural performance with Karamba3D which is a plugin for Grasshopper [12]. It is also made available by TU Delft. This software is used to perform Finite Element (FE) analysis on structures in Grasshopper and is known to be very robust [13]. Karamba3D has a function that can perform cross-section

optimisation with an algorithm. This will be used to select the cross-sections needed for the different models. The diagonals will get a different cross-section than the columns and beams. This results in eight cross-sections, two for each model. With Galapagos, a plugin for Grasshopper, which uses a Genetic Algorithm (GA) the needed stiffness of every configuration is determined [13]. This will be a requirement for the joint design later.

Step 4. Joint design
The joints are a very important part of this research. The behavior of a structure is dependent on the stiffnesses that are used for the joints. There are two types of joints, the intermodular and intramodular ones. The intermodular joints are the joints that connect the beams within a module. The intramodular joints connect the different modules to each other. The focus will mainly be on making the connections circular. Constructability is also an important part of the feasibility of the joints. The different joints are designed in the IDEA StatiCa software, also made available by TU Delft [14]. A joint classification is performed and the stiffness of the joints is determined for various load situations. This will be input for the model in Grasshopper. An iterative loop will be added to assign stiffness values to the joints depending on the distribution of the loads. The stiffnesses are important for the performance of structures that will be analysed using FE software.

Step 5. Topology reconfiguration model
The model in Grasshopper will then be extended. to include topology reconfiguration and optimisation. A GA will optimise the topology of the structure. The evolutionary solver Galapagos is a free plugin for Grasshopper that is used for that purpose. Given a certain grid, the algorithm selects modules from the catalog and assigns them to different locations in certain orientations. The structure is analyzed with the Finite Elements Method (FEM) software of Karamba3D. This will provide the algorithm with values for the performance of the structure. For optimisation a fitness function is made that minimizes the weight of the structure within the boundary conditions. The structure may not exceed the maximum allowed deformation, the utilization of the members is not allowed to be larger than 100% and the structure may not buckle. The algorithm will be designed such that it converges to an optimal solution. This will take several iterations.

Step 6. Model verification
The model verification is performed to see if the model provides reasonable results. It is the first test that can find mistakes in the model. The implementation of the boundary conditions and the GA are both checked. Furthermore, the FE-analysis is investigated to see if the stresses in the elements are reasonable and if the deflection is as well. For simplicity, a small structure is analysed and examined. Observations are made qualitatively.

Step 7. Model validation
The model verification does not show if the model performs well quantitatively. To check if the results are valid a different FEM software called RFEM will be used to compare with the output from Karamba3D. If the results for both softwares are similar then the model is correct. The modules will be analysed to see if the distribution of stresses are correct. A small truss structure will be analysed as well to check stress and deflection.

Step 8. Case study
The verification and validation show that the model works and the output is correct. However, a real test of the algorithm for a large-scale structure is not included in these tests. For this reason, the model will be compared to a structure in the literature to see if the accuracy is good enough and if the output is feasible. Deflection and stress levels will be compared to see if the results are in the right order of magnitude. Besides, it is checked if the exiting structure can be improved or if a similar structure can be made.

An application of the model for irregular structures is also part of the tests. This separate section will analyse a small scale model with irregular supports. The difference with the case study is that an optimised structure such as a truss is not known before the start of the simulation. A structure is designed with a selection of modules based on intuition. With a simulation, it is checked if the fitness of the intuitive structure can be improved with a simulation. This will show if the model can be used in combination with designer intuition.

During the process, the steps will sometimes be performed simultaneously and also iteratively. For instance, the catalog of modules will constantly be changed depending on which designs are feasible for the model and on the requirements. Also, the cross-section and joint design are not linear processes. This step-by-step approach is a guideline to keep the process on track and to see which steps still need to be performed. The overview of the methodology is schematized in Figure 2.2.

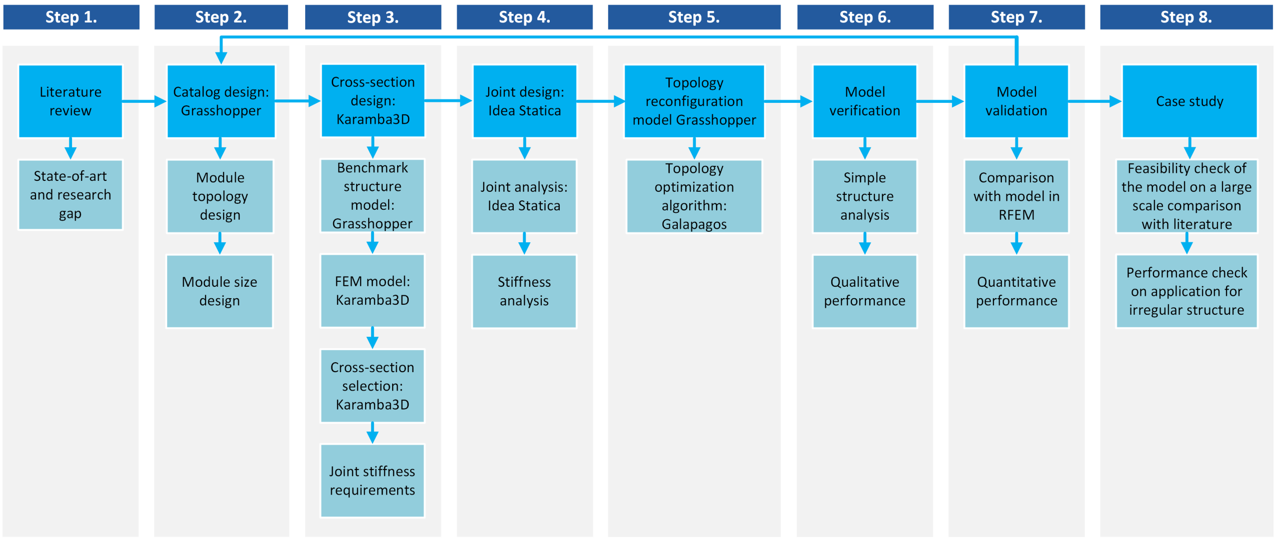


Figure 2.2: Methodology scheme

3 Module design

In this section, the design of the initial modules is made. This is done by looking into the literature about the needed model parameters and constructing a conceptual framework. This is needed because it gives a better overview of the need for certain configurations and argues why certain choices in design are made. Besides the design of the general shapes of the modules, a choice for element cross-sections is made. The need for specific joints is also outlined.

3.1 Conceptual model

The eventual goal is to reduce structural mass, and to maximise the element capacity and circularity potential of the structure. To achieve this multiple steps have to be taken. The first one is making a conceptual model. This is a more general description of the model that highlights all the various parameters that are included in the system and which relations they have to each other. Besides this, it shows the first key concepts and assumptions that are included in the design.

3.1.1 Model parameters

To make a design it is key to understand which parameters are important and how they are interrelated. The different parameters are derived from the literature review and can be seen below. Some of the parameters such as mesh density or offset are only important when creating a structure that comprises multiple modules.

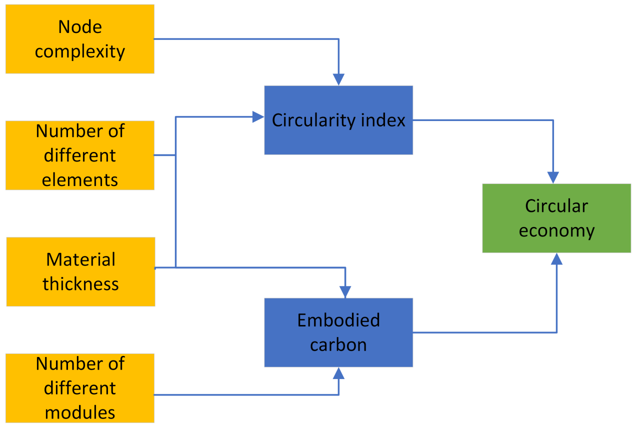
- Surface geometry
- Curvature
- Node complexity
- Node uniformity
- Support conditions
- Loads
- Cross sections used
- Truss height/Mesh offset
- Mesh density
- Pattern
- Imperfections

3.1.2 Conceptual framework

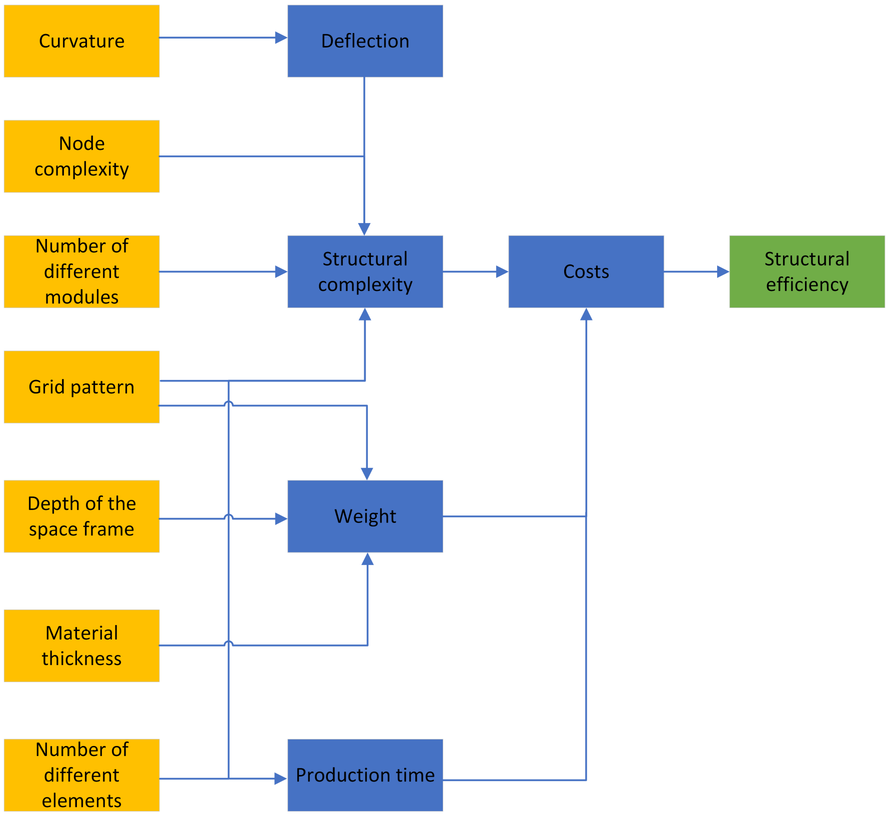
The variables and factors can be subdivided into three different groups. These are the independent, dependent and intervening variables. Their groups can be seen in Table 3.1. As can be seen from the table a couple of extra variables are added, the intervening variables. These variables do not influence the design in terms of technical perspective, but rather by considering feasibility, costs, and circularity. The dependent variables are influenced by the independent variables and intervening variables. They are important because they are used to answer the main research question. The dependencies are visualised in two frameworks, one for circular economy and one for structural efficiency. These can be seen in Figure 3.1a and Figure 3.1b. The reason that the frameworks are split is that the figure would become way to big and would not be a good overview. Besides, two of the most important objectives of this research can now be separately assessed. There are some overlapping parameters. Node complexity has an indirect influence on both circular economy and structural efficiency for instance.

Table 3.1: Independent, dependent and intervening variables

Independent variables	Dependent variables	Intervening variables
Curvature	Circular economy	Weight
Node uniformity	Structural efficiency	Costs
Node complexity		Production time
Singularities		Deflection
Number of vertices		Embodied carbon
grid pattern		Circularity index
Depth of space frame		



(a) Conceptual framework circular economy



(b) Conceptual framework structural efficiency

Figure 3.1: Conceptual frameworks for structural and circular economy; Independent variables are in yellow, dependent in green and intervening in blue

3.1.3 Circularity

The Intergovernmental Panel on Climate Change (IPCC) has stated that to reach the Climate Agreement the building sector needs to be "zero-carbon" by 2050. This is a huge task looking at the size of this industry. The construction sector uses 50% of all materials and emits up to 12% of global greenhouse gas emissions [7, 15]. Ways of reducing the environmental impact are using less material or using materials that have less impact [15, 16]. Furthermore, using components during their entire lifespan reduces the environmental impact [16]. The reuse of reclaimed steel structures also leads to the reduction of the carbon footprint of constructions. Partial disassembly of a structure into trusses instead of all structural members can further help this purpose. It gives less design freedom than when a structure is broken down into small elements [17]. Therefore, to design a structure such that certain parts can be reused can have large benefits. There are different ways to make an optimal design with reused material. It can be done with a GA [7], Mixed Integer Linear Programming (MILP) [18], the Hungarian Algorithm [7] or Best-Fit heuristics such as Greedy Search [15, 19]. The environmental impacts and savings by reusing elements can be quantified using Life Cycle Assessment (LCA) [18].

Minunno et al. outlines a framework that is applicable to prefabricated building modules or elements. The reduction, reusability, adaptability, and recyclability of components are assessed on the environmental advantages [20]. According to the design of disassembly principles, the nodes in wooden structures for example should be designed without glue, the number of parts and materials should be minimised and assembly processes should be reversible [21]. For steel elements glue is not used but in this case welding should be avoided to make it easier to disassemble a structure [22]. Environmental savings are possible with reuse but it can be costly monetary-wise because of the deconstruction and refurbishment [18].

3.1.4 Parametric design software

In literature, a great deal of research has been performed on space frames. One main finding is the way that different structures can be formed. For the design of complex structures, a classical way of designing is where the CAD and structural analysis programs are used back and forth. First, a geometric design is made with the CAD program. Then the structural analysis software is used to check the structural properties. This can take up much time due to the complexity. A quicker solution is the use of computational design. This allows for a more dynamic process in which the design, analysis and optimisation of a structure are coupled [23]. Often used software packages for parametric design are a combination of Rhino3D (CAD software) and Grasshopper (parametric modeling) or Dynamo Studio (parametric modeling) and Revit (CAD software) from Autodesk. These technologies are used for the geometric design of structures. FEM software is still needed to check structures made by these tools. In this thesis the combination of Rhino3D and Grasshopper is used because it is widely used in industry and the TU Delft has a license for it.

There are plugins available for Grasshopper such as Karamba3D which is a very robust option to explore the structural performance. Galapagos is a plugin that implements evolutionary optimisation algorithms. There are two options to choose from within Galapagos, these are the GA and the annealing algorithm [12]. Octopus is a plugin that is using the the SPEA-2 algorithm and can be used for multi-objective optimisation [24]. It also has extra functions such as Octopus loop which can be used for recursive looping in Grasshopper [24]. This basic looping which is widely used in computer programming is not possible with the basic Grasshopper components. This is due to the continuity of data flow on which Grasshopper is based. Looping can be implemented through plugins such as the earlier mentioned Octopus Loop, programming such as Python (GYPython), and data manipulation inside Grasshopper. Other plugins available for looping in Grasshopper are HoopSnake, Loop, and Anemone [25].

Plugins may lack some small functionalities that are sometimes needed for structural design. For joint design IDEA StatiCa will be used. Both for checking the feasibility and structural capacity of the joints it is useful. There exists a Karamba IDEA plugin that is a direct implementation of the IDEA StatiCa software in the Grasshopper and Rhino environment that is focused on connection design mainly. T that uses an Application Programming Interface (API) to link this software with Grasshopper. The problem is that it focuses on detail design and not global structure. Furthermore, the Koala plugin for Scia Engineer does not give the opportunity to directly load the data from the calculation model or the analysis results into Grasshopper [26]. Apellániz and Vierlinger describe another parametric FEM Toolbox plug-in that includes these functionalities [26]. There are numerous of other plugins available but a selection is made for this thesis. The software used in this thesis and what it is used for are listed below in Table 3.2. Most of the different plugins can be used for much more functionalities than what is shortly described here. An extensive literature review on the state of art in optimisation algorithms in Grasshopper is given by Vierlinger [24].

Table 3.2: Plugins for Rhino 8 used in the project

Plugin	Use
Grasshopper	Parametric visual programming software
Karamba3D	Structural design and FEM analysis
OctopusE loop	Looping in Grasshopper but normally used for multi-objective optimisation
Galapagos	GA optimisation
TTToolbox	Data management
Lunchbox	Data management
Metahopper	Controlling Grasshopper scripts

3.2 Design of the initial catalog of modules

Before designing the modules it is necessary to know what type of ground structure is used. A gridshell is a structure that due to its double curvature is very strong. They are single-layered and are constructed of a grid or lattice. Space frames can be seen as multilayered gridshells. For larger spans, gridshells are limited because of buckling in the elements and a double layer is necessary. The required resistance to bending moments can often not be reached with single-layered structures [27]. From spans of around 10 m, the beam elements in a single-layered grid are already not economic anymore. For larger spans open trusses or Vierendeel girders are often used for spans in one direction [2]. Free-form grid shells are made using numerical procedures. The reason is that there is no straightforward procedure to design these structures. In design methods for optimisation, both the topology and the geometry of the pattern are generated [28]. Topology optimisation is the optimisation of material distribution and density in a given domain. This is also called continuous topology optimisation. It can also be the optimisation of the element selection and cross-section in the ground structure which is called discrete topology optimisation [29]. Structural analysis is needed for the design but also the structural feasibility must always be assessed because a design has fabrication constraints [28].

Often Non-Uniform Rational B-Splines (NURBS) surfaces are used to design free-form structures [5, 30]. These are surfaces that can be formed into organic shapes with control points inside the CAD software. They make the transition of the surface volume smooth. It is a form of geometric generation. These are unlike analytical surfaces, such as a cylinder or sphere not directly described with fixed equations. Complex combinations of mathematical objects are required, like lines, curves and planes, formulae, and procedures [31]. The continuous NURBS surfaces are almost always discretised into a mesh. A mesh is a certain computational pattern with which an object is created in 2D or 3D. It consists of vertices, edges, and faces. The vertices are interconnected [32]. The options that are used in most practice are triangular and quadrilateral meshes in 2D. Hexagonal meshes are also often used. For each of these surfaces, the faces have the same number of vertices, edges have the same length, and vertices have the same number of neighbors [11]. In 3D tetrahedra and hexahedra are often used shapes that can be seen as two single-layered meshes that are vertically connected.

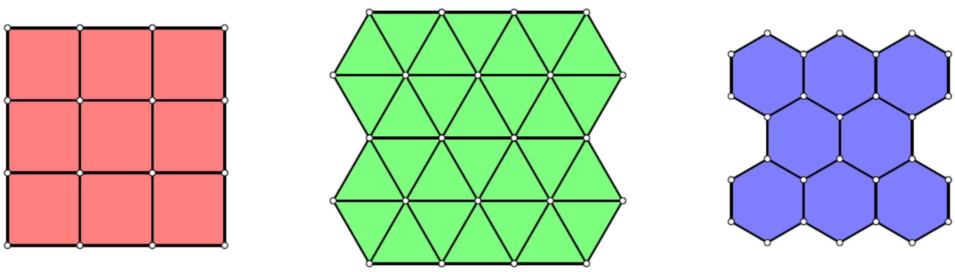


Figure 3.2: Basic regular tessellation of the plane. From left to right: quadrilateral, triangular, and hexagonal mesh (Source: Oval [11])

There are also more nonbasic patterns that can be used as a mesh for free-form structures. These can be seen in Figure 3.3. The difference with the formerly mentioned patterns is that these consist of two or multiple regular polygons. Each of the polygons is connected to the same other polygons in a repetitive manner. On curved or 3D spaces polygons of these tesalations have edges with varying lengths that may interrupt the existing pattern [11].

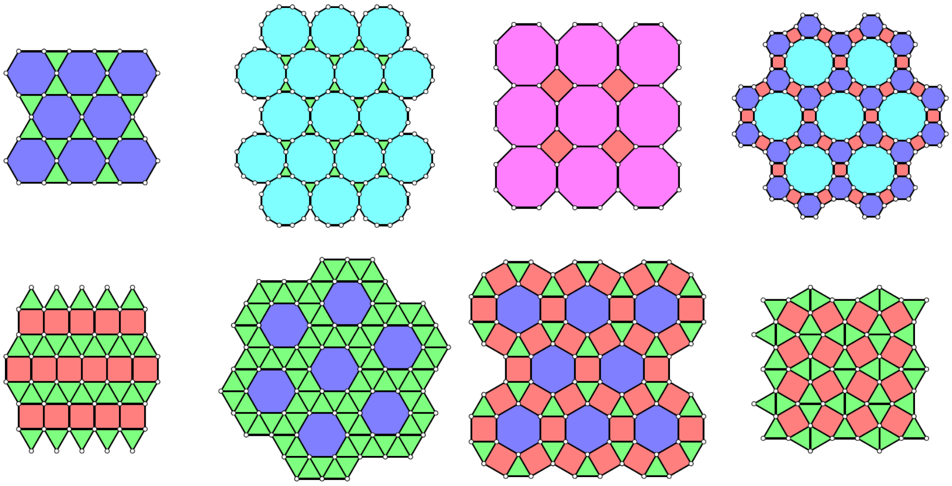


Figure 3.3: Semi regular tessalations of the plane (Source: Oval [11])

To create a double-layered structure the single layer could be copied and translated in a certain direction. An alternative method involves displacing the vertices of the second layer perpendicular to the first layer. This means each vertex would be shifted individually along the direction perpendicular to the surface at that specific point [27]. Also, mesh-based approaches are used [33]. Oval et al. introduces rule-based topology finding of quad-based mesh patterns [34]. This is a method that can be used for topological exploration and structural design. Mesnil et al. introduces a methodology to asses non-standard patterns for space frame design. The use of the marionette technique for the generation of non-standard patterns with planar facets is described [35]. In designing the catalog of modules different approaches can be taken. The mesh can be altered both for the base and for the top layer in the double-layered gridshell. For instance, a quadrilateral mesh can be used at the bottom and a triagonal mesh at the top layer. This would lead to exceptional shapes of the modules. Furthermore, the placements connecting elements between the two layers can be varied. This shows that there are many different possibilities to make double-layered grids. A couple of possibilities for simple structures can be distinguished in Figure 3.4. The four groups displayed are rectangular grids, diagonal grids, rectangular/diagonal grids and three-way grids [36].

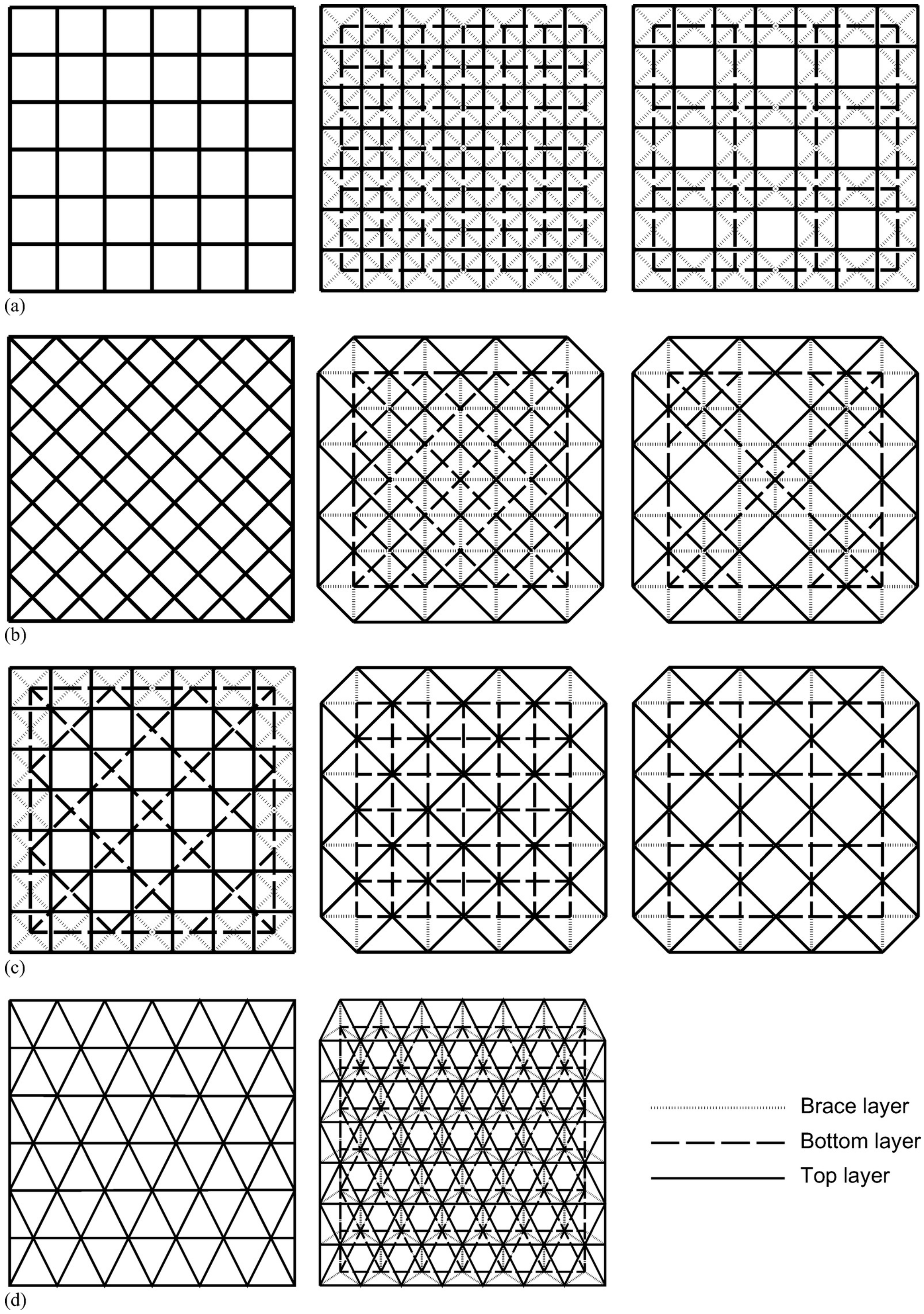
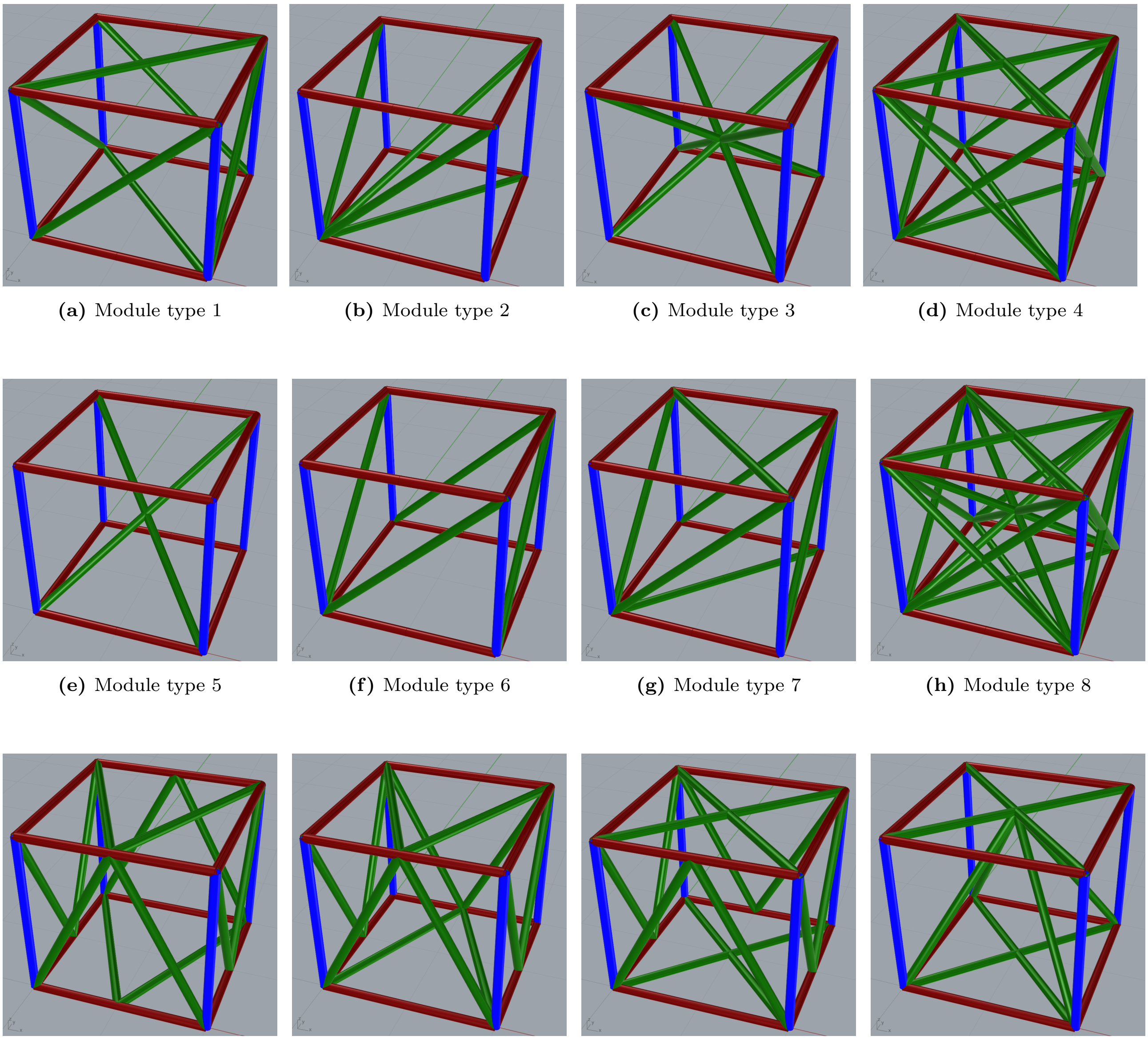


Figure 3.4: Various configurations of double-layered grid spatial structures: (a) Rectangular grids: Square On Square (SOS), SOS offset and square-on-larger-square offset; (b) Diagonal grids: diagonal-on-diagonal, diagonal-on-diagonal offset, and diagonal-on-larger diagonal offset; (c) Rectangular/diagonal grids: Square-on-diagonal offset, diagonal-on-square offset, and diagonal-on-larger square offset; (d) Three-way grids: Triangle-on-triangle and triangle-on-triangle offset. (Source: Setareh et al. [36])

As a first design step, the commonly used quadrilateral mesh is applied for both the top and bottom layer and the positions of the interconnecting elements are varied. This results in a SOS space frame. The idea behind this is that simple cubic module types can be used. Moreover, the system is generic and can easily be adapted to variable boundary conditions. Besides, it has the advantage that it can be seen through more easily than other space frame structures which is an aesthetic advantage in roof design [4]. A basic module can already be seen in Figure 2.1. This standard catalog of only one module is extended with various other concept modules that are compatible with the previous one. A preliminary design for these reconfigurable hexahedron modules is visualized in Figure 3.5. It shows different modules that all have different kinds of bracing for stability purposes. The distribution of forces is also different for various modules which potentially makes them useful in different design configurations. There is a large variation in topology of the modules to have a large range of possible stiffnesses and weights. For the design of the structure, the modules can all be placed in a certain configuration such that the material use is minimised and overall requirements are met. For instance requirements for maximum deflection.



(a) Module type 1 **(b)** Module type 2 **(c)** Module type 3 **(d)** Module type 4

(e) Module type 5 **(f)** Module type 6 **(g)** Module type 7 **(h)** Module type 8

(i) Module type 9 **(j)** Module type 10 **(k)** Module type 11 **(l)** Module type 12

Figure 3.5: Initial catalog of hexahedron modules; Beams are marked with red, diagonals with green and columns with blue color

3.3 Design of module size

The modules consist of different beam elements and their connections. For the modules to have a circular economy it is important to make the elements of which they consist standardised. The size of the modules may vary but standard cross-sections will contribute to the circularity. For optimal structural efficiency however, a variety of cross-sections could lead to a more lightweight and material-optimised structure. This is a trade-off that has to be made. Structural efficiency is also influenced by the number of different elements through production time and structural complexity. It is therefore decided to take only two cross-sections as the standard elements. There is one for the diagonals and one for the columns and beams. For this relationship, see also

Figure 3.1b. It is also preferable to only use one or just a few standard modules for the design of a structure. This is because it reduces the number of alterations to be made to the system which reduces the cost of fabrication [2].

In a conventional design approach, the size of the cross-section depends on the loads that are acting on it. The space frame modules however can be used for the design of a variety of structures. These include long-, mid- and short-span roofs, floors, exterior walls, and canopies [10]. Buildings such as sports arenas, workshops, terminals and warehouses are buildings that are often made with spaceframes [37]. Therefore, the required strength of the modules is different for each application. There are many different module sizes used in structures all over the world. The span-depth ratio of double-layered grids varies from 12.5 to up to 25 [10]. These numbers vary per source. According to Gasii for instance the span-depth ratio varies between 16 and 25. Makowski even mentions a ratio of 20 to 40, however, this must be reduced to somewhere between 15 and 20 for a grid supported at the corners [38]. Normally double layered gridshells are only used in construction of large spans of more than 24 m. In smaller structures often a single layer is sufficient enough. The upper and lower bounds of the span-to-depth ratio for different spans are visualised in Figure 3.6. It shows different ranges of different literature sources. The span-depth ratio should decrease with the span, however, it increases according to experience [10]. It was found by Gasii that module size generally increases with the span [22].

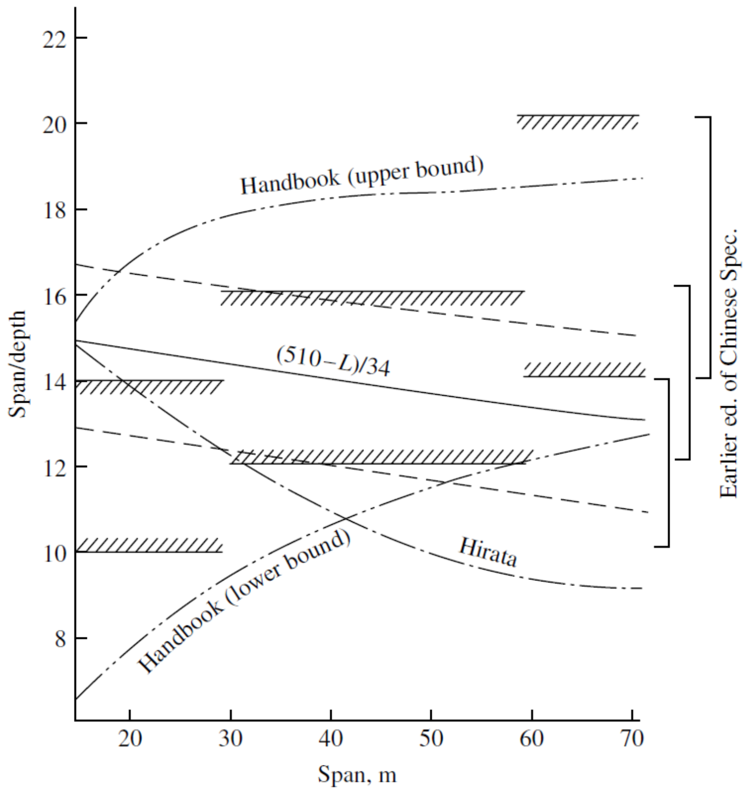


Figure 3.6: Relation between depth and span of double-layered grids. The equation describes a graph for a roofing system composed of steel purlins and metal decks. Here L is the short span. (Source: Lan [10])

The grid which influences the module size of a system has a large influence on the costs. For roofs in particular this has often got to do with the clear span of the cladding [39]. The module size fixes the number of nodes needed for the structure as well [37]. The Space Deck System consists of prefabricated pyramidal modules as can be seen in Figure 3.7 [4]. The standard modules have grid dimensions of 1200 mm x 1200 mm with depths of 750 mm or 1200 mm. Also, 1500 mm x 1500 mm with depths of 1200 mm or 1500 mm is possible or even 2000 mm x 2000 mm with a depth of 2000 mm [2].

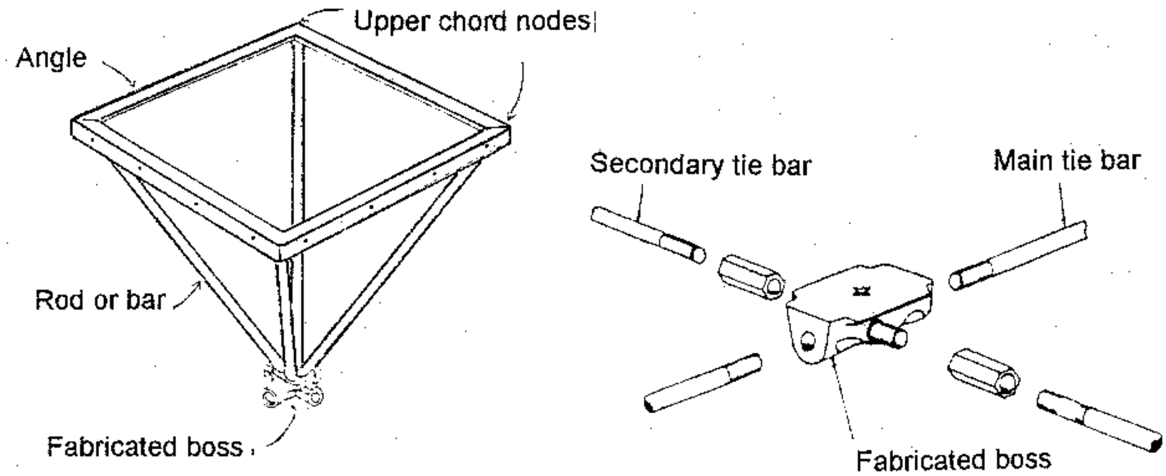


Figure 3.7: Space deck system. (Source: Lan [10])

Gerrits mentions that for prefabricated modules the size is often between 1.5 to 2.0 m. For the cubic systems such as the Cantarella system mentioned in Appendix A the systems are normally a bit larger, 2.0 to 3.5 m [39]. For different structure sizes the depth of the modules can also be different. Typical span-to-depth ratios for the space deck system are around 25 to 30 for roofs that are supported on all sides [2]. However, this ratio is reduced for roofs that are only supported at the corners. A span-depth ratio of 20 is chosen at first since this is in the middle of the range that Gasii suggests for double-layered gridshells [22]. Multiplying this with various depths between 2.0 and 3.5 m gives an approximation of the maximum span for every depth. The span range is the difference between these maxima. For example, at a depth of 2.0 meters, the maximum span range is 40 meters, while at a depth of 2.5 meters, it expands to 50 meters Therefore, the span range for a module depth of 2.5 meters falls between 40 and 50 meters. The ranges found in literature and computed ranges are both shown in Figure 3.8. The computed ranges for this model will be calculated further in this section. Since double-layered gridshells are usually used for spans larger than 24 m, this marks the minimum span within the range for the smallest module depth. Of course, smaller distances can also be covered by them but this will quickly become less economical than other structural elements such as Open Web Steel Joist (OWSJ)s. Also, the span range for single-layered grid is shown. For this structure type the maximum distance is only 10 m because for a larger distance the elements become less economical and open web trusses or Vierendeel girders should be substituted for the beam elements [2]. OWSJ girders can be used to span larger distances while still being economical. Depending on the size of the elements the spans can be tens of meters. For commonly used elements, the span range is between 3.66 and 30.5 m [40].

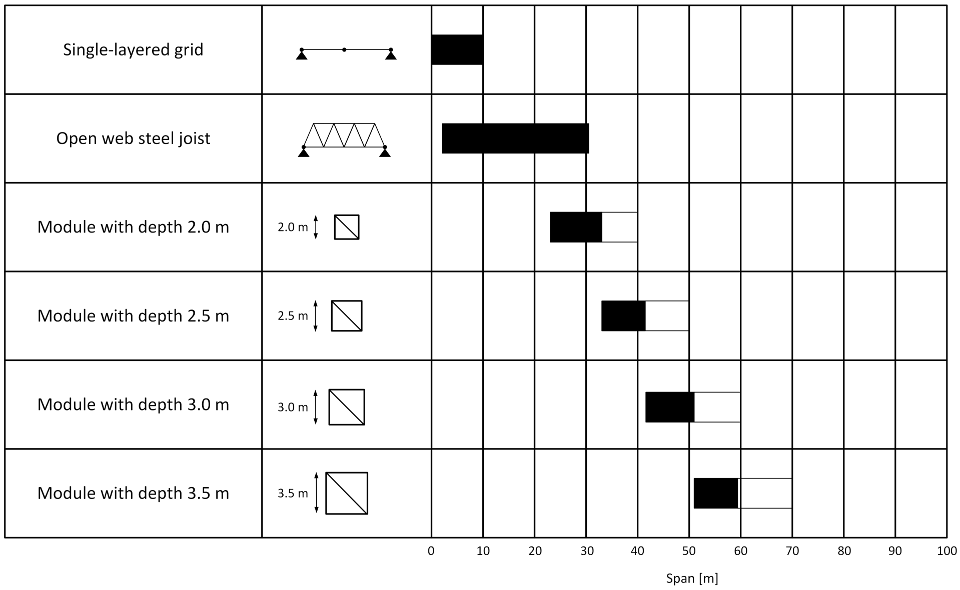


Figure 3.8: Span range for different module depths: The black beam shows the span width computed for this model and the white beam stretches to the maximum span as found in literature

3.4 Design of element cross-sections

The cross-sections that are used in modern space frames are often closed cross-sections such as pipe, square, or rectangular shape members. This can be seen in figures A.1, A.2 and A.3 from Appendix A Overview of connection types with a node, without a node and with prefabricated units. This appendix gives an overview of connection systems for space frames that are applied worldwide. A Circular Hollow Section (CHS) is often chosen for the design of space frames [22]. However, for the connections between complete modules often a Square Hollow Section (SHS) is chosen. This is because connection design is easier for these cross-sections. Easier connections will also be beneficial for the circularity in the end-of-life stage of the structure. Reuse of the modules or elements in other structures is therefore likely. Furthermore, a hollow structural section has the advantage that the material is evenly distributed. Such a cross-section is symmetrical to the load case independent of the direction of the vector [37]. These types of cross-sections are therefore well resistant to buckling about both principal axes. Furthermore, they have high torsional stiffness and a high strength-weight ratio [41].

The design optimisation will be done by introducing cross-section uniformity for all diagonals and for all columns and beams. This is a simplification that will set a baseline for the design. Later on, complexity can be increased for optimisation. However, it is strongly advised to keep it this way for the benefit of production and circularity. As mentioned before the space frame may be used for numerous different applications including for structures where large crowds can gather such as in sports hall roofs.

In Grasshopper and Rhino3D a parametric model is created. Module type 6 is used to make a structure with a large span that reaches up to the benchmark distances mentioned in Section 3.3. The reason that module type 6 is used is because when it is reconfigured correctly a regular truss such as a Howe of Pratt truss can easily be made with it. Each module size has a different span. With the model, the element cross-sections needed are checked. The modules are used to make a planar 3D truss in the Howe configuration. This is the name for a truss which has its diagonals oriented as shown in Figure 3.9c and Figure 3.9d. With the use of the Karamba3D plugin, the model is extended such that also FEM analysis can be performed on the structure. A combination of all loads with safety factors is applied to the structure. The distributed loads are transformed into point loads on the corners of the upper beams of the modules. These point loads resemble the forces caused by roof cladding plates laying on the structure.

There are a couple of selected loads used in this model. An initial variable imposed load of 0 to 1.00 kN/m^2 may be chosen for the design of the cross-section. This is a load that accounts for roof maintenance and represents people and equipment according to Eurocode 1 part 1-1 [42]. Also, point loads within the range of 0.9 kN to 1.5 kN may be selected. These act independently from the distributed load. The recommended values of 0.4 kN/m^2 and 1.0 kN are chosen for the analysis. The point loads are not placed at every column position in the structure but at every second column. Besides this, a snow load of 0.70 kN/m^2 is applied on the roof. This is a standard value in the Netherlands for snow load according to Eurocode 1 part 1-3 [43]. It also depends on the type of roof and the height and other coefficients. However, this is not relevant for this case since it is a flat structure. A dead load of 0.25 kN/m^2 is also adopted. This accounts for a lightweight roof with minimal installations and cladding, often the case for large halls. No horizontal loads are considered at this stage. Therefore, wind load is left out of consideration. The different applied loads can be seen in Table 3.3. The self-weight is variable and depending on the topology and therefore not valued in this table. The size of the modules can be varied. When this happens the point loads scale to the size of the modules. The self-weight also automatically changes.

Table 3.3: Loads used for cross-section design

Load	**Magnitude**	
Self weight	-	kN/m^2
Variable distributed load	0.40	kN/m^2
Variable point load	1.00	kN
Snow load	0.70	kN/m^2
Dead load	0.25	kN/m^2

The combinations of actions on a structure can be calculated according to Equation (3.1). The part between brackets, or the characteristic combination can be seen in Equation (3.2) [44]. It shows the combination of different loads acting on a structure multiplied with certain γ and ψ factors. With the γ factors taken from Table A1.2(A) and Table A1.2(B) of the Eurocode the load combinations are then calculated [44].

$$E_d = E\left\{G_{k,j}; P; Q_{k,1}; \psi_{0,i} Q_{k,i}\right\} \quad j \geq 1; i > 1 \tag{3.1}$$

$$\sum_{j\geq 1} \gamma_{G,j} G_{k,j} + \gamma_{Q,1} Q_{\mathrm{k,1}} + \sum_{i>1} \gamma_{Q,i} \psi_{0,\mathrm{i}} Q_{\mathrm{k,i}} \tag{3.2}$$

In which:

$G_{k,j}$ = Permanent loads [kN/m^2]

$Q_{k,1}$ = Leading variable load [kN/m^2]

$\psi_{0,i}$ = ψ factor for variable action [-]

$\gamma_{G,j}$ = Partial load factor for permanent load [-]

$\gamma_{Q,1}$ = Partial load factor for leading variable load [-]

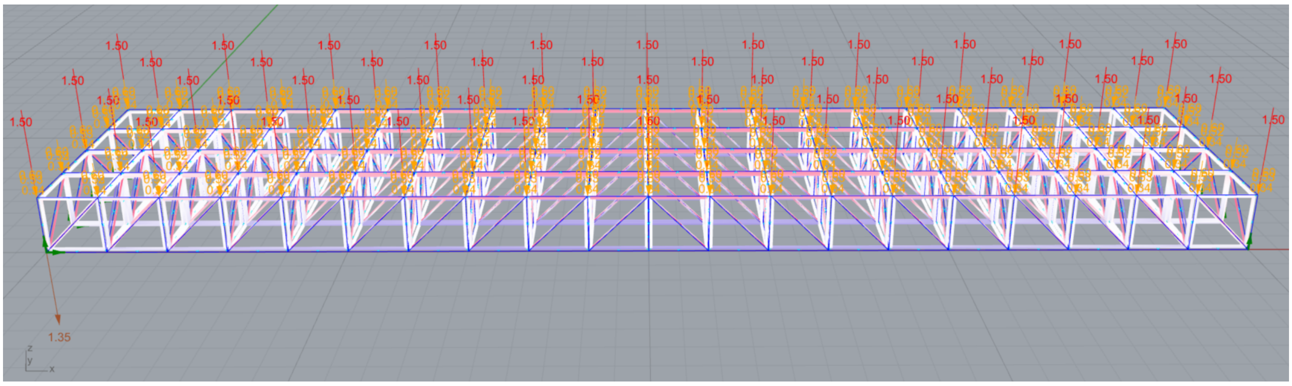
$\gamma_{Q,i}$ = Partial load factor for accompanying variable load [-]

For general loading situations where multiple variable loads play a role, the ψ_0 factor is assigned to the accompanying variable load. For different load types the ψ factors according to Table A1.1 of Eurocode 0 [44] are given in Table 3.4. It is found that the load combinations with variable impost load and partial snow load are the most critical.

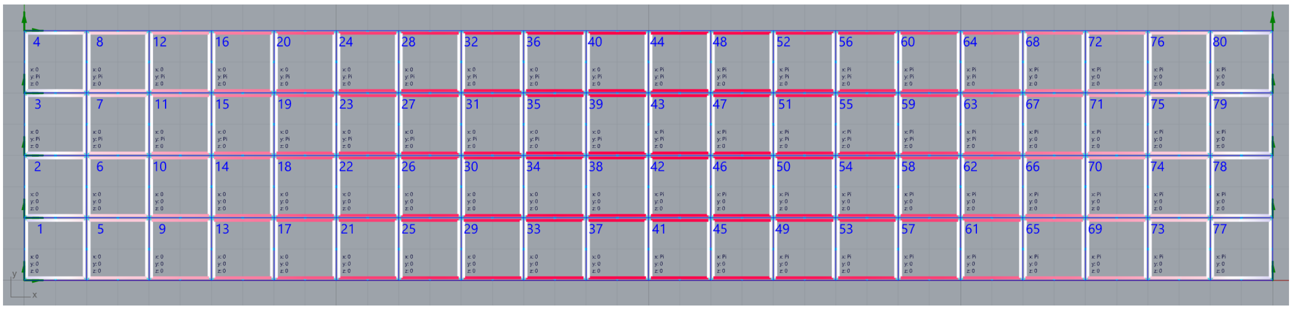
Table 3.4: ψ factors for building class C (Source: European Committee for Standardization (CEN) [44])

Action	ψ_0	ψ_1	ψ_2
Imposed loads in building category C	0.7	0.7	0.6
Snow loads for sites located at altitude H $\leq$ 1000 m a.s.l.	0.5	0.2	0.0

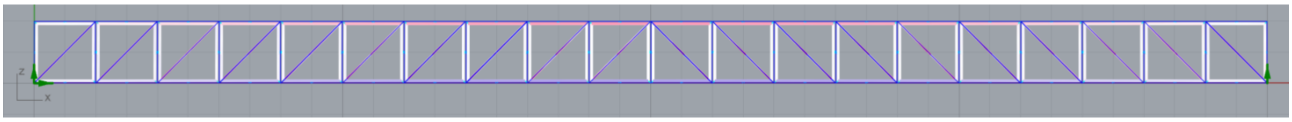
The structure configuration can be seen in Figure 3.9. The stresses in the model are also indicated with colors. Red color indicates high compressive stress and blue is high tensile stress. The size of the modules is 2 x 2 x 2 m. The total length of the structure is 40 m and its width is 8 m. On both ends of the structure, the modules are simply supported at every column which is every two meters. This is at the corners of the modules. The reason that this is done is to create a one-way span. The structure can be schematised as a large simply supported beam, which is a model that can easily be analysed. The point supports, which can be seen from the green arrows in Figure 3.9b can be seen as a line support along the two smaller edges of the structure. At one end the structure is free to move in the horizontal direction.



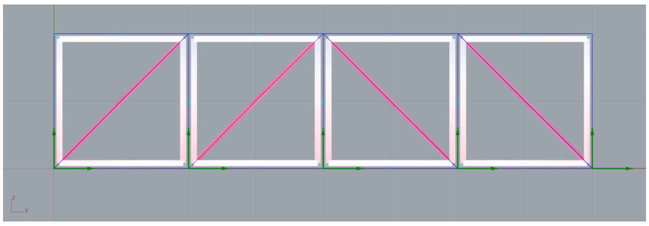
(a) 3D view of the cross-section optimisation model



(b) Top view of the cross-section optimisation model



(c) Front view of the cross-section optimisation model



(d) Side view of the cross-section optimisation model

Figure 3.9: Optimisation model setup. The numbers in blue indicate the position of the modules, the black numbers the rotation in x, y and z.

Each of the modules has a certain orientation in the coordinate system. All of the modules are numbered as can be seen in Figure 3.9b. Besides, their orientations are shown by the change in rotation in x, y, and z-direction from the original orientations shown in Figure 3.5. These are given in radians. It is important to keep track of this to be able to reproduce the results later.

The elements displayed in this model have an eccentricity from the lines where they are defined (thin light blue lines). For each element, this is a translation towards the center of the module it belongs to. This eccentricity is the distance between the intermodular connecting endplates and the centerline of corner points of the modules. Half of the beam plus this distance to the plate gives an eccentricity of roughly 80-100 mm depending on the element size. An eccentricity of 80 mm is taken here for the x-, y-, and z-direction. This is based on the joint design that is discussed in Section 4. This makes the module depths in reality a little bit smaller than what was mentioned before in Figure 3.8. In fabrication this has to be taken into account. The eccentricity of the elements leads also to eccentricity of the loads which can therefore introduce bending moments at the joints. The percentage of axial loading may therefore be lower than for conventional space frames.

The "Optimize Cross Section" function of Karamba3D adjusts the cross-sections based on material utilization, ensuring optimal use of the material. The algorithm used by the optimisation tool has three steps. First, the section forces at "nSamples" points along all beams are calculated using the initial cross-section. Then for each set, the first element type that suffices for the calculated stresses is chosen. This results in a cross-section from the family of CHS(EN10210-2) for the diagonals and a cross-section from the family of QRO(EN10219-2) for the beams and columns. Then the algorithm checks the stresses again. If there are no changes needed or the maximum number of iterations is reached the algorithm stops. If this is not the case the process will be done again. The simulation result is an initial design for the cross-sections that will later be tested on different structures as well. The cross-sections could therefore be adjusted later. The cross-section optimisation function is also supplied with a deflection constraint. According to the Dutch national annes of the Eurocode the deflection of roofs should not be larger than $\frac{1}{200}$ of the span [45] as shown in Equation (3.3). Furthermore, the stresses may not exceed the limits of the material.

$$d \leq \frac{1}{200} \cdot l_{roof} = d_{max} \tag{3.3}$$

In which:

d = Deflection [m]

l_{roof} = Length of the roof [m]

d_{max} = Maximum allowed deflection [m]

The joint stiffness has a large influence on the performance of the structure. There are different joints in the model depending on the type of module used. The joint design will be discussed in Section 4 but in this section the needed stiffnesses are determined. The translational stiffness and the rotational stiffness around the local y- and z-axis of the elements are needed for the model. There are different connections in the model. Each element has a connection to a corner point of a module. Besides, there are also splice connections that divide the elements in the middle. This is done to make the module elements smaller and easier to transport. A simplification that has been made for these splices is that the translational stiffness is fixed. With two end nodes and the splice in the middle, the structure has three aligned pins within a module. A local mechanism can form under loading because the two connected beams are able to move freely. Due to the deformation of this local mechanism, stress levels become very large locally. These high stress levels result in unreasonably large cross-sections which is undesirable and unrealistic. A simplification has been made in the model for these splices. The translational stiffness is fixed which gives more realistic results. This is reasonable because it is more in line with the actual behavior of the structure.

An iteration is performed to find stiffness values for the joints that result in a structure with the deflection and utilisation of the elements within limits. This is done with the GA Galapagos, a plugin for Grasshopper. A minimisation of the extreme utilisation of the structural elements is performed by varying the joint stiffness per joint group. The rotational and translational joint stiffness of each joint group are the variables in this optimisation. This is done for the structure with modules of depth 2 m. The resulting joint stiffnesses are shown in Table 3.5. The stiffness values are very high and can be considered as rigid. This initial rigidity of the joints is allowed in preliminary design to select sections for columns and beams for sway modular buildings as described by Farajian et al. [46]. This research focuses on modular design with inter-modular connections, and therefore, allowing a large initial stiffness of the joints is appropriate. Some of the connections are not tested in this setup. For instance, the bracings that connect four beams (connection E) or eight beams (connection F) can be present in module types 3, 4, 5, or 8 and are not present in this truss structure. For this reason, these are given the same values as the diagonal splices. These values are all the first design step in the process of designing joints. The location of the joints inside a module are shown with labels in Figure 3.10. In Section 4 the final design of the joints will be presented.

Table 3.5: Requirement for rotational stiffness ($S_{j,ini}$) and translational stiffness (S_t) values used in the model

Label	FEM model connections	$S_{j,ini}$ [MNm/rad]	S_t [MN/m]
A	Columns to corner points	3.9	976
B	Beams to corner points	3.9	976
C	Column splices	0.7	921
D	Beam splices	0.7	921
E	Diagonals bracing connecting 4 beams	4.7	651
F	Diagonals bracing connecting 8 beams	4.7	651
G	Diagonals to corner points	1.1	496
H	Diagonals splices	4.7	651

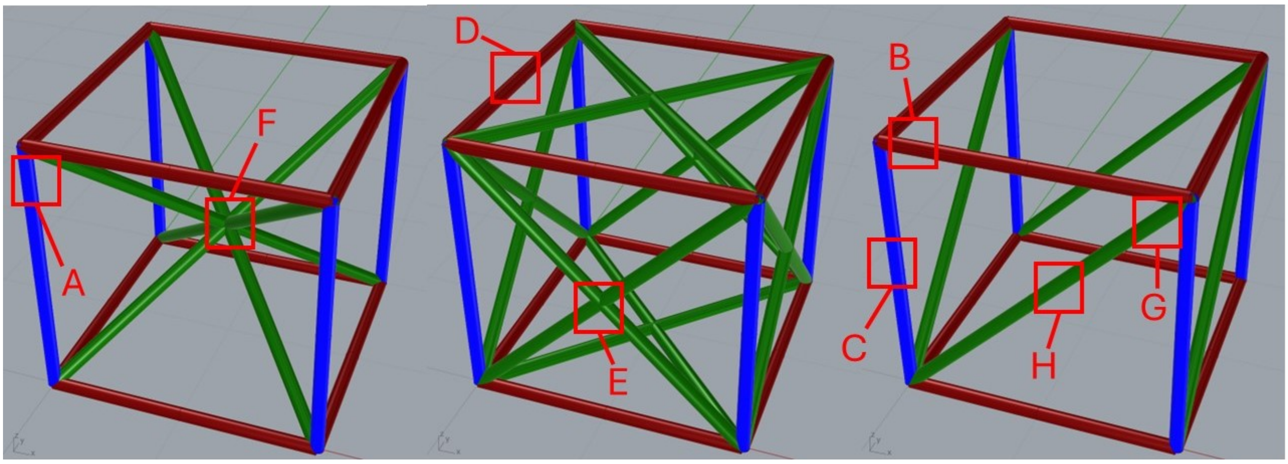


Figure 3.10: Location of the different joints in the modules

For different module sizes (2.0 to 3.5 m depth) structures with span benchmark values are designed. The same optimisation procedure is performed resulting in the different cross-sections shown in Table 3.6. Not all benchmark values for the distance could be reached with the values for the stiffness of the joints found earlier. Although the stiffness values are relatively high the benchmark values of 60 and 70 m could not be reached with the modules with a depth of 3.0 and 3.5 m respectively. The reason is that the structure modelled here is heavier than normal space frames because more elements are involved in the design. Furthermore, for the structures with a length of 40 and 50 m the value for the utilisation was already close to 100%. Therefore, the span-depth ratio is lowered from 20 to 17. This can also be seen in Figure 3.8. The new structure has a smaller length and can be seen in Figure 3.11. There are less modules used in the length. Due to the smaller length, the structure is in this example not entirely symmetrical anymore. This will cause a slight difference in how the loads are distributed.

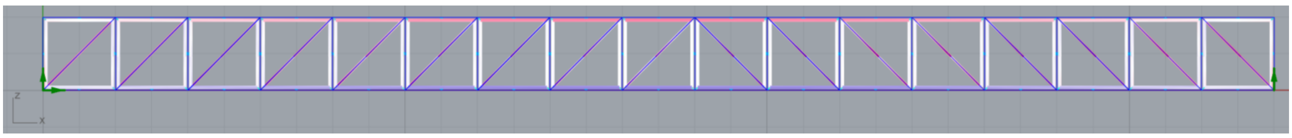


Figure 3.11: Truss structure with adjusted span-depth ratio front view

Besides, the cross-sections chosen for the modules with a depth of 1.0 and 2.5 m are slightly different than the cross-section optimisation algorithm had chosen. The cross-sections of the diagonals are larger. This is because the design of joints is easier with a little larger cross-sections. Also, making the type of cross-section the same benefits the circularity. In a later stage, the components can all be used in a new project.

Table 3.6: Cross-section types under varying module depth

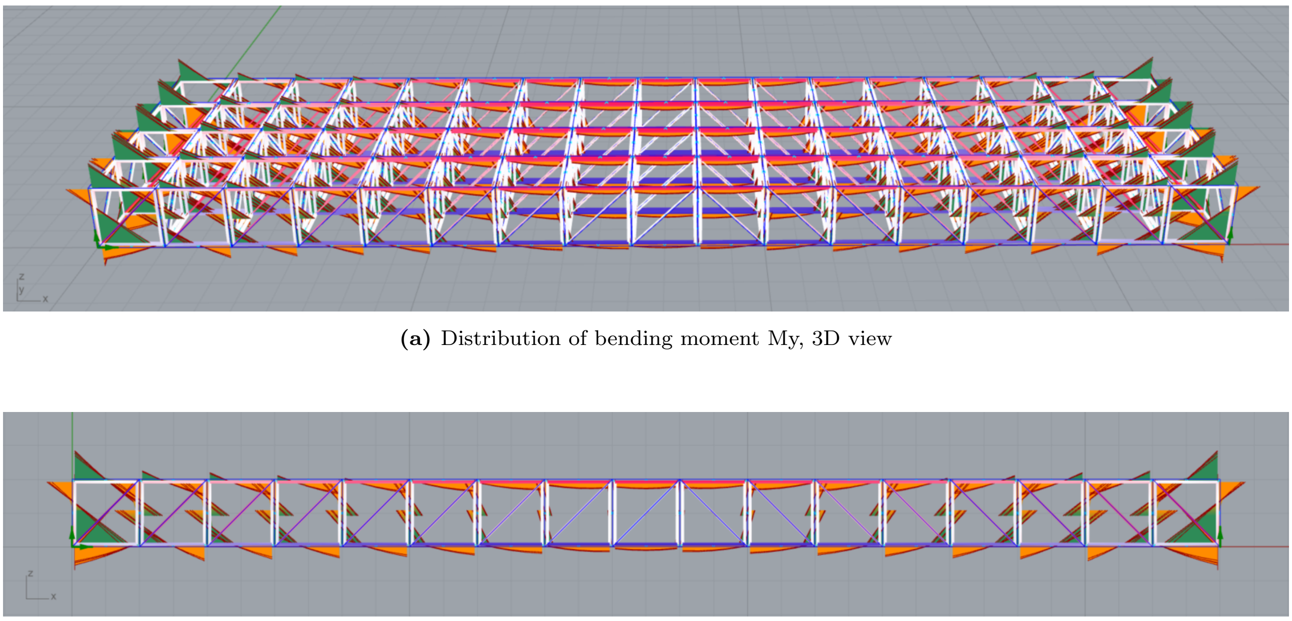
Module depth [m]	Model span [m]	Straight elements cross-section	Diagonal elements cross-section	Utilisation [%]	Deflection [mm]
2.0	34.0	SHS 90 x 6.0	CHS 60.3 x 3.2	69.1	82
2.5	42.5	SHS 100 x 6.0	CHS 60.3 x 3.2	95.1	128
3.0	51.0	SHS 120 x 12.0	CHS 76.1 x 5.0	99.6	185
3.5	59.5	SHS 140 x 8.0	CHS 88.9 x 6.3	97.9	246

There are no significantly lower stress levels when no splice connections exist in the beams and columns. The results for the structures with varying module depth can be seen in Table 3.7. There are large differences in deflection of the structures. The reason is that the elements behave more rigidly when no splices are added to the modules. If the splices were less rigid, also the stress levels could be much higher. The required stiffness of the splices was very high as well resulting in expensive joints. All other joints had to be stiff as well for the structure to be strong enough. It is a design choice to make the modules like this to make them more circular and easier to transport.

Table 3.7: Structure performance with and without splice connections

Module depth [m]	Utilisation with splices [%]	Utilisation without splices [%]	Deflection with splices [mm]	Deflection without splices [mm]
2.0	69.1	68.1	82	66
2.5	95.1	94.2	128	105
3.0	99.6	98.2	184	135
3.5	97.9	84.7	246	183

The distribution of the bending moment in the direction of the span can be seen in Figure 3.12. The truss is smaller in length than the one shown originally as mentioned earlier due to the reduced benchmark spans. Therefore, the structure is not entirely symmetrical. Due to the combination of columns and diagonals at the corner points there are some irregularities in the moment distribution.

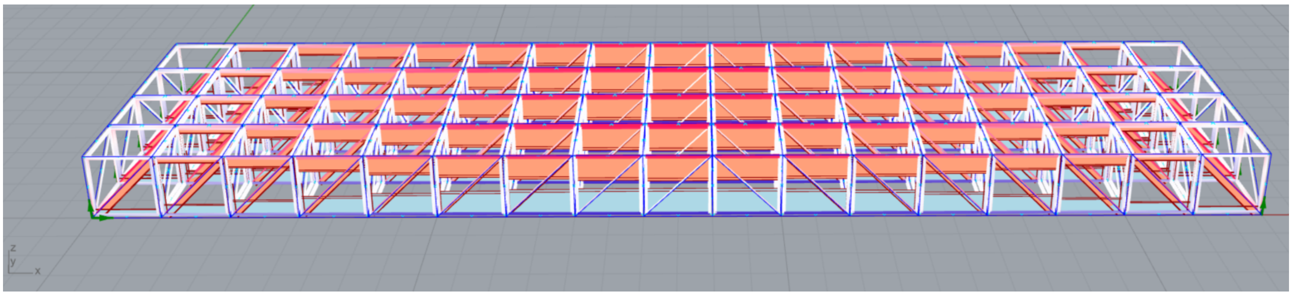


(a) Distribution of bending moment My, 3D view

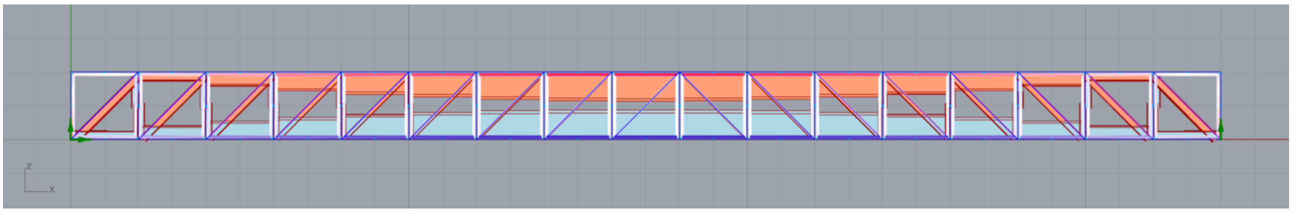
(b) Distribution of bending moment My, front view

Figure 3.12: Distribution of bending moment My

The normal force distribution in the truss can be seen in Figure 3.13. The top chord shows large compressional forces and the lower chord large tensional forces. The diagonals mostly experience compression, which is normal for a Howe truss configuration. The size of the normal forces is much larger in the chords than in the columns because of the moments occurring there.



(a) Distribution of normal force N, 3D view



(b) Distribution of normal force N, front view

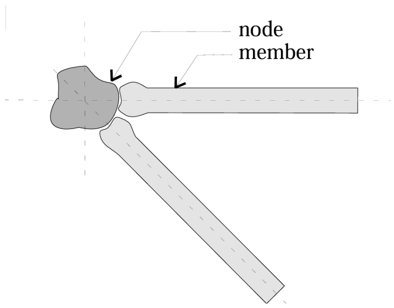
Figure 3.13: Distribution of normal force N

4 Design of joints

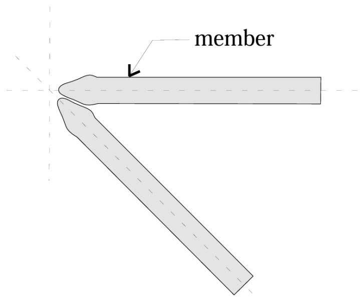
The design of the joints will be focused on connecting the types of elements for the different module types shown in Section 3.2. An overview of joint design for space frames is given with information from literature. Then the inter- and intramodular joints for the reconfigurable space frame model are designed.

4.1 Literature on joint design

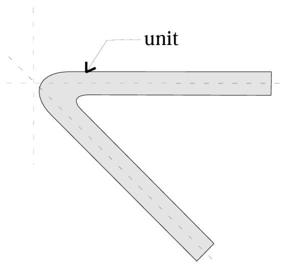
For the design of spaceframes, there are many different options for making connections. Depending on the design wishes of the engineer or architect a choice can be made for which type of components to use. A division can be made between three different kinds of connections. These are the connections with a node, without a node, and prefabricated units [39]. The different kinds of systems can be seen in Figure 4.1. Nodes are parts of the structure that connect different members. Connections without a node are realised by bolting or welding different elements without designing a specific part that connects them. Connections with prefabricated units are connections made in the factory with welds. They don't require any on-site assembly and are usually a part of modules [39]. The collection of these members is also called a unit. Another distinction that is often made between connections for steel structures is the subdivision in bolted, welded and combined nodal connections [22]. Combined nodal connections have both welded and bolted parts.



(a) Connection with a node



(b) Connection without a node



(c) Connection with prefabricated units

Figure 4.1: Classification of connections (Source: Gerrits [39])

There is a difference between a joint and a connection but they are often used interchangeably. The difference is that a joint consists of the connection and the other members attached to it. The connection is the part where two elements meet and the components used to connect them. The difference is shown in Figure 4.2.

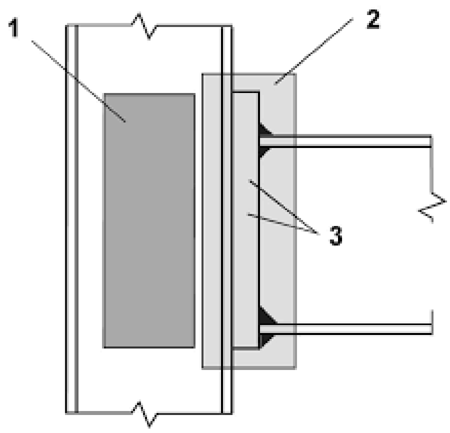


Figure 4.2: Joint, connection, component. The entire sketch is the joint. 1: Component web panel. 2: Connection 3: Components bolts, welds, end plate (Source: Kavoura [47])

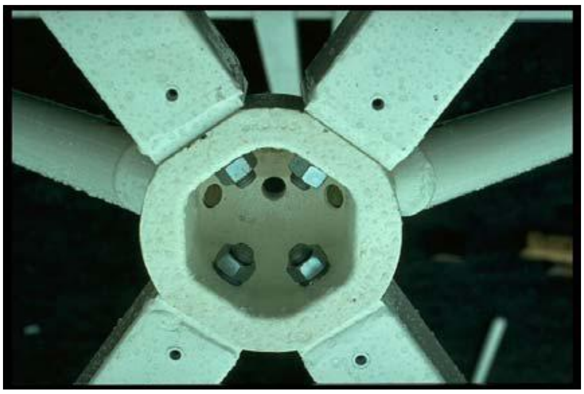
4.1.1 Connections with a node

The determining factor for costs in the design of a structure is the complexity of the connection node [22]. Besides, the design of nodes is often regarded as one of the most complex things to do in 3D modeling. Therefore, the design of nodes in gridshells is often a bit disregarded. A predefined catalog for connections between different members is often provided for design purposes. There are two common approaches for the design of nodes in gridshells. One is to avoid the node as a visible element and the other is to optimise the shape and topology to reduce the node angles. Ways to avoid nodes as visual objects are the lapped node concept, 'GoodKarma' connections, and Reciprocal frame lapping connections [21]. Furthermore, it is well known that node complexity will increase if there is torsion in the nodes. Therefore, in design this will be avoided as much as possible [48].

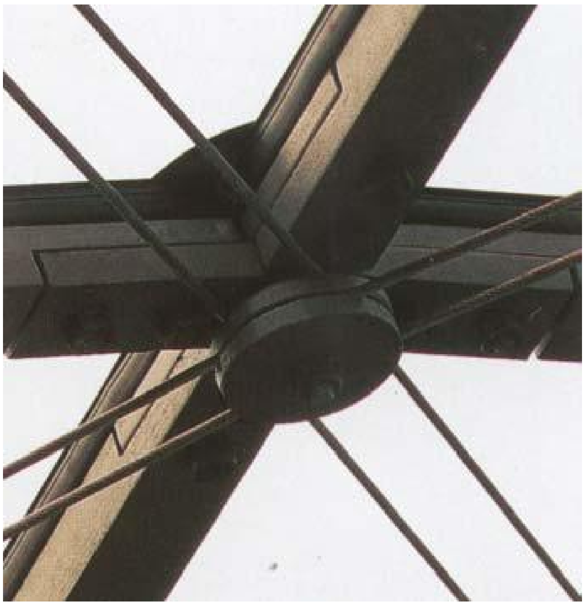
Koronaki et al. outlines a method for the optimization of space frames for joint uniformity. Because space frames and other surface structures are unique in their design and can have free-form shapes connections can be unique as well. This makes constructibility and production more difficult. The optimization of joint uniformity was assessed by investigating angles for different joints, grouping of joints, and geometrical optimization of spatial coordinates [5].

Types of nodes that are often used in double-layered structures are ball node connectors and bowl node connectors [31]. Other types of node connectors are disk nodes, cylinder nodes, and block nodes [10]. For single-layered structures, the types of nodes can be split into two groups. These are the splice connectors and end-face connectors [31, 37]. Splice connectors have a contact surface between the node and the connected structural member that runs along splice plates in the longitudinal axis of the member. These are joined together by welds or bolted splices with shear-stressed bolts. End-face connectors are connected to the different elements with end plates transverse to the longitudinal axis of the structural member. The connection is realised with tension-stressed bolts or welds [31]. These different types of nodes are visualised in Figure 4.3. An overview of different connections and with which cross-section they are compatible can be found in Appendix A.

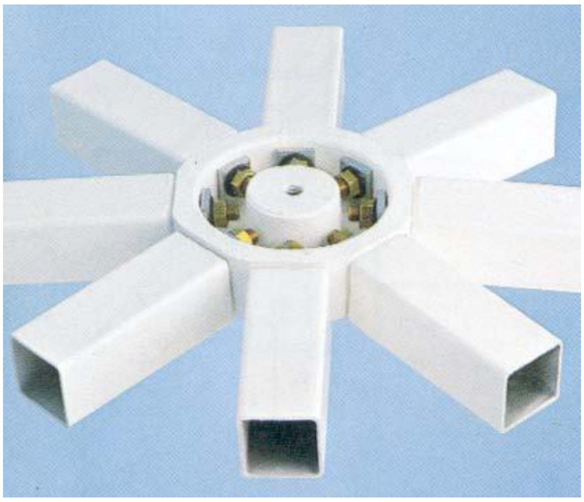
(a) MERO ball node connector (Source: Stephan et al. [31])



(b) MERO bowl node connector (Source: Stephan et al. [31])



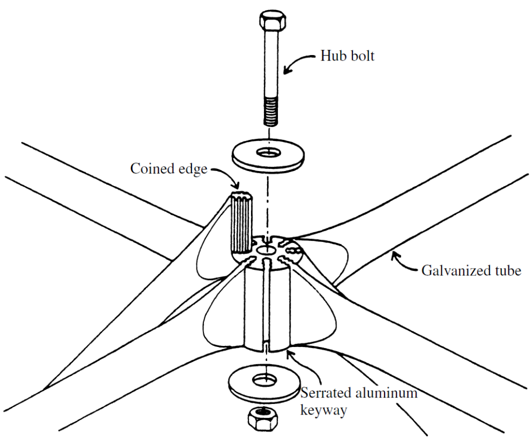
(c) Splice connector SBP-1 (Source: Stephan et al. [31])



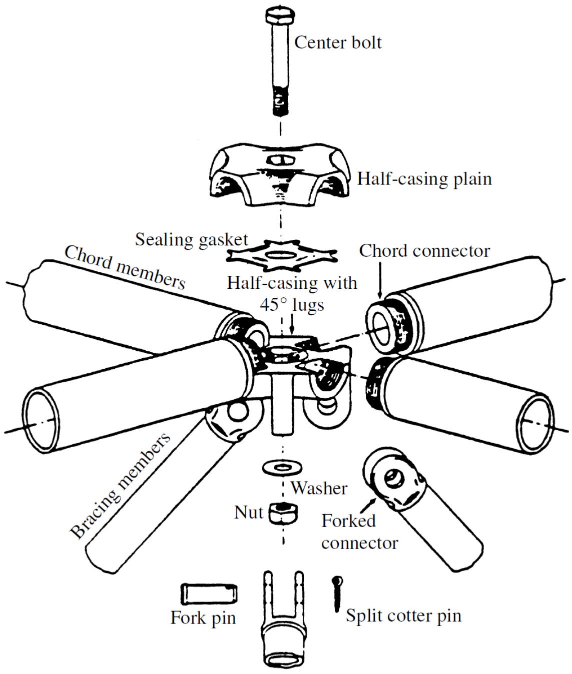
(d) End-face connector MERO-4 (Source: Stephan et al. [31])

Figure 4.3: Various node connectors. Figures 4.3a and 4.3b present nodes for double layered gridshells and figures 4.3c and 4.3d for single layered gridshells

Other type of node connections are the Triodecit and Nodus System. These connections are used in 3D spaceframes and can be seen in Figure 4.4. The connection in the Triodetic system, seen in Figure 4.4a, consists of an extruded aluminum connector hub with serrated keyways. A bolt with washers at each end completes the connection [10]. The Nodus system also connects circular tubes. As can be seen in Figure 4.4b the Nodus System consists of half-casing from cast steel with a hole in it. The chord connectors are from forged steel and are welded to the half-casings [10]. All parts are also interlocked. Most of the fabrication of this type of joint is done in the factory.



(a) Triodetic System (Source: Lan [10])



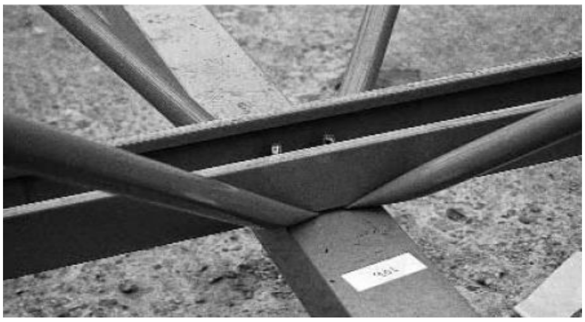
(b) Nodus system (Source: Lan [10])

Figure 4.4: Triodetic and Nodus System joints

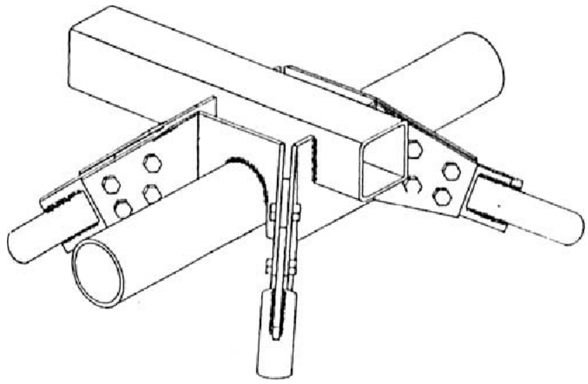
4.1.2 Nodeless connections

Connections without a node can be made with bolts and welds. Welded connections have the advantage that in almost every angle different rods can be connected to each other. However, welding is less time efficient and due to complexity is also less precise, leading to eccentricities. Also, welding causes stress in the material, and reassembly and disassembly are not possible [22]. Bolted connections have the possibility of reassembly disassembly making the connections more circular. Nodeless connections have the advantage that the connections can be made directly between the ends of grid members. This saves overall consts. However, there is the disadvantage that often these parts have standard angles between them, resulting in fewer possibilities in configuration [2].

Another type of system without nodes is a continuous chord system. These types of systems can be considered something between so-called 'piece-small' and modular systems. A piece-small system is a systems consisting of nodes with discrete beam elements between them. There is no spatial joint between the different members of the system. The beam elements are continuous through the joints. This has the advantage that no expensive joints have to be made. However, eccentricities may have a negative effect on the structure [2]. Examples of continuous chord systems are the Harley Space Truss System, Mai Sky System, and Catrus. Examples can be seen in Figure 4.5.



(a) Harley Type 80 node joint (Source: Chilton [2])



(b) Mai Sky System joint (Source: Courtesy Mai Sky inc.)

Figure 4.5: Harley Type 80 joint and Mai Sky System joint

4.1.3 Connections with prefabricated units

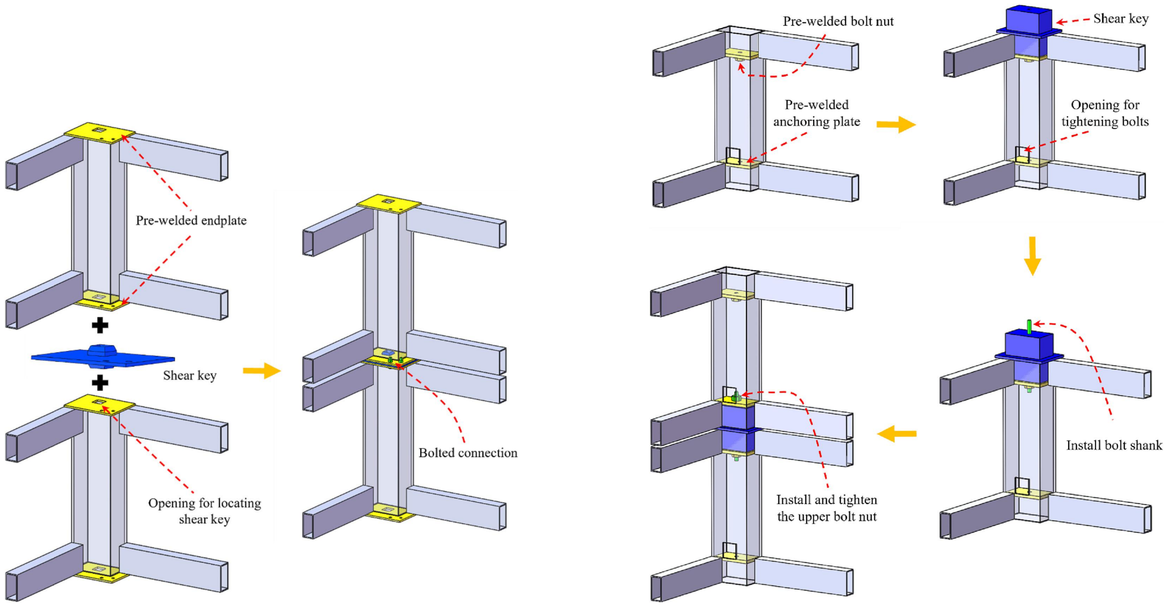
Connections that are prefabricated are already welded together in the factory. These elements form modules such as the ones shown in Figure A.3 in Appendix A [39]. A great advantage of this type of connection is that the speed at which structure can be made is improved. Less time is needed at the construction stage because complete modules can be lifted in position without the need to assemble them first. Transportation can be more difficult. When large modules are moved which can not efficiently be stacked on a transportation vehicle, it may result in inefficient transport. This is of course not an efficient way of bringing material to the construction site. Therefore, in design, this must be avoided as much as possible.

4.1.4 Inter modular connections

Besides connections that work between different elements, there are also existing connections for joining modules together. An overview of these different types of joints is given in papers written by Lacey et al. and Srisangeerthanan et al. [49, 50]. A short overview of their findings is also found in Appendix B. In this section the most important intermodular connections from literature for this thesis are presented.

Han et al. proposes two different connection types that give the modules high constructability through simplicity. The used connections apply to rectangular cross-sections. See also Figure 4.6. This type of connection in Figure 4.6a is made by pre-welding endplates to the cubic module. In between the two endplates of two different modules a shear key is and the plates are then connected with bolts. The bolt is responsible for taking the axial tensile forces. If there is compression this is taken by the hollow cross-sections that press against each other. The shear key is responsible for taking the shear force at the connection [52]. The on-site construction is simple because all pieces fit well into each other and the bolts are quickly installed. Because of this way of construction, the modules have parallel beams at the face of the connection. This results in a higher stiffness of the overall structure but also more material use. A similar thing can be observed from Figure 4.6b. This type of connection is realised with a tubular shear key that can be placed on the module. The shear key has a hole inside to allow a bolt to pass through. The module contains a pre-welded bolt in which the bolt can be fixed. After placing the shear key and the bolt in place a second module can be connected to the shear key and be tightened with another bolt nut. This bolt nut is accessed through a hole in the frame.

Han et al. tested the connections using a hydraulic jack that could apply a vertical load. It was found that for the connection in Figure 4.6a the main failure modes include bolt fracture and yielding of the endplate under tensile loading. Furthermore, due to lateral loading bending of the endplate around the bolts can occur leading to separation of the elements. For the module connection in Figure 4.6b it was found that the failure modes were bolt fracture and yielding of the anchoring plate. This type of connection has a greater deformation capacity. Also, the bolts experienced significant shear force. The initial rotational stiffness of the different specimens varied from 11 to 14 MNm/rad for the semi-rigid joints and 28.7 MNm/rad for the rigid joint specimen. [51].

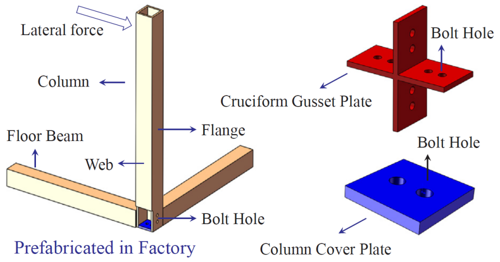


(a) Module connection with pre-welded endplate and shear key (Source: Han et al. [51])

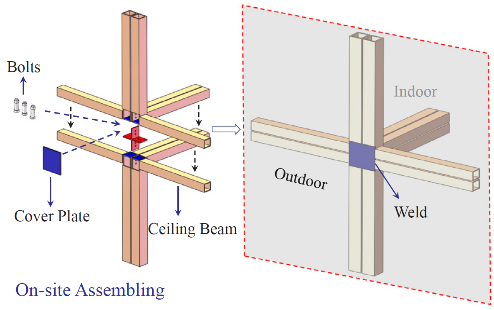
(b) Module connection with endplate and upper and lower tubes (Source: Han et al. [51])

Figure 4.6: Connections between modules with rectangular cross-section

Besides the intermodular joints that can be seen in Figure 4.6 there are also some other options. Locking devices are also used and instead of bolts, rod joints can be used [52]. It must be stressed that the connections that are displayed in Figure 4.6 can only be used at corner points. More connection types are needed to make sure that in every configuration of the modules they are all interconnected. There must be connections between 2 horizontally connected modules, similar to the connections shown in Figure 4.6. Furthermore, four modules can also meet with a connection in the middle of them. This is another challenge. For every module type in the catalog, it must be ensured that a connection exists that can connect all modules in a certain configuration. An example of the way this could be done is shown in Figure 4.6. In this research by Deng et al. a cruciform crusset plate with boltholes is used as a connecting element. It is placed in between four modules that are connected to the plate with bolts. To be able to connect the bolts the SHS elements have holes cut in them to be able to have access to the boltholes. This is similar to what is done with the module in Figure 4.6b. After bolting the modules together a cover plate is welded to cover the holes. The thickness of the gusset plate was 10 to 20 mm for each of the specimens that were tested. Grade 10.9S M24 high-strength bolts were used. The connection was tested on monotonic and cyclic loading. It was found that the connection could be used to form a moment-resistant frame. According to Eurocode 3 Part 1–8 the connection can be classified as semi-rigid. The initial rotational stiffness of various specimens varied from 15.8 to 19.1 MNm/rad. Failure was mainly due to local buckling followed by weld fracture of the cover plate welds [41].



(a) Module connection with cruciform gusset plate (Source: Deng et al. [41])



(b) Modular connection with cruciform gusset plate on-site construction (Source: Deng et al. [41])

Figure 4.7: Connections between more than two modules with cruciform gusset plate

Connections can also be made without the used of gusset plates. [46] makes a classification system for inter modular connections that can be used in modular steel buildings. A general joint in a sway corner-supported modular frame can be seen in Figure 4.8. four columns and eight beams can be connected to this joint.

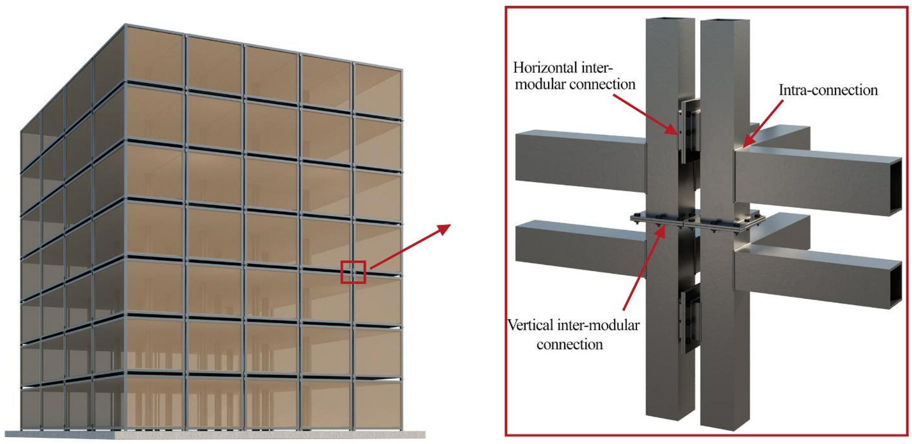


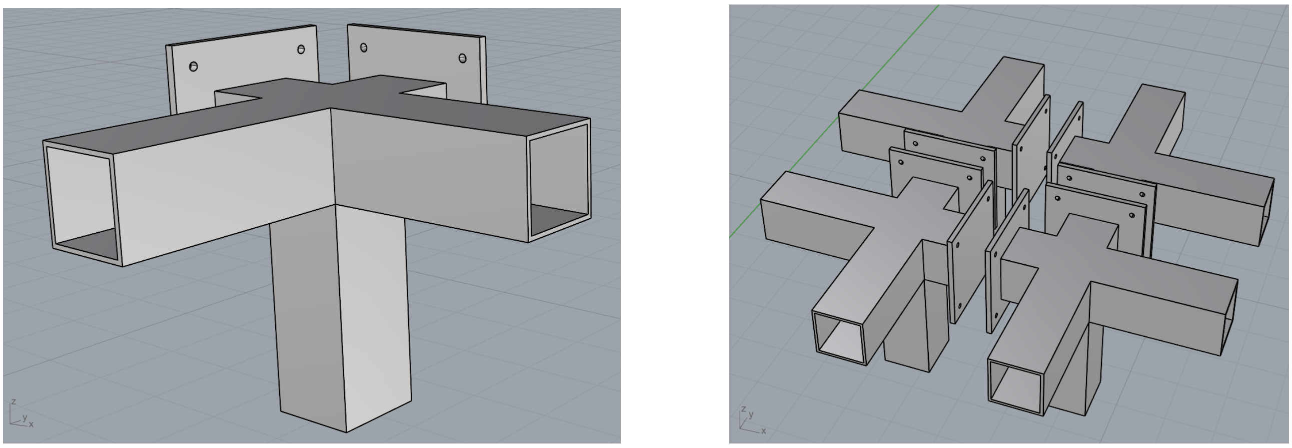
Figure 4.8: A typical joint in a perimeter frame (Source: Farajian et al. [46])

4.2 Design of intermodular connections

The method of jointing has to be considered carefully as mentioned before. The costs and time it takes to connect the modules are dependent on manufacturing and labour work for assembling [37]. The connections must be strong, stiff, simple, and easily manufactured [37]. Most of the connections mentioned in literature are mainly for connecting single elements. These are so-called intramodular connections. There are recent developments in connecting complete modules with connections, the intermodular connections. Some recently developed ones are mentioned in Section 4.1.4.

Intermodular connections can present challenges that differ from conventional space frame connections. Intermodular connections often require extra operation space and also the alignment of adjacent modules in a fast manner is regarded as a big issue. Also, lateral loading of interconnected modules is not well understood [51]. Most intermodular connections are made for SHS modules. Since there are no guidelines yet for structures with intermodular connections the Eurocode 3 is used for design considerations [53].

A combination of the connections in the previous sections 4.1.1 and 4.1.4 is used for the design of the new connection. The modules will have endplates welded to the corner points in both directions of the horizontal plane. This will allow them to be connected to one or two adjacent modules at each intersection point. The welding of the plates to the corner points can all be prefabricated in the factory. Figure 4.9 shows the different connection types in 3D. This connection shows similarities to a Inter Modular Connection (IMC) described in research by Yang which is a semi-rigid joint with an initial rotational stiffness of $S_{j,ini} = 739$ kNm/rad.



(a) 3D view of a standard corner with endplates

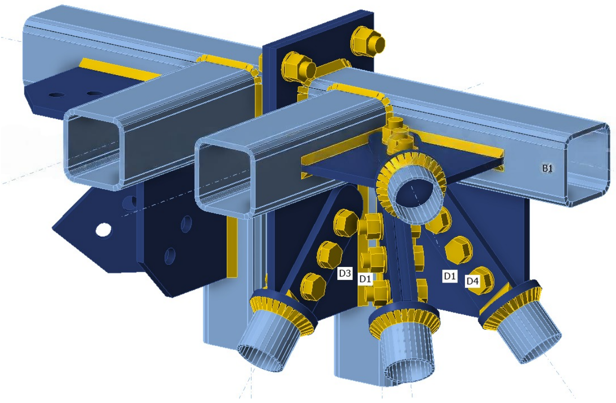
(b) 3D view of four modules connected with endplates

Figure 4.9: 3D views of the different intermodular connection types

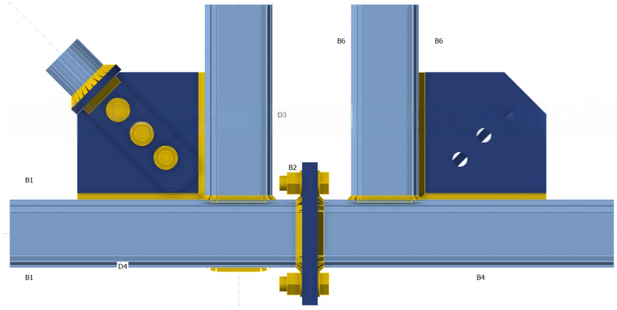
To see if the connections would be feasible they are modelled in the software IDEA StatiCa. This software can be used to design steel connections and do calculations with the Component Based Finite Element Model (CBFEM) method [55]. This is a method that is based on the component method used in Eurocode 3 for joint design in combination with FEM. The component method is a way of calculating the stiffness of a joint by modeling every component as a spring using various design formulae. The different stiffnesses are then combined to calculate the stiffness of the complete joint [56]. The CBFEM method uses this principle but for analysing the stress of individual components such as bolts or plates it uses FEM [55]. IDEA StatiCa is widely used in industry to design steel connections and is therefore a good check for both feasibility and capacity of the joints. The software can also be linked utilising API with Grasshopper and Rhino. This enables the communication between Grasshopper and IDEA StatiCa. However, since this option is still being developed it is not used in this study. The implementation of the parametricity of the joints in the model could however be a huge advantage in later research.

A connection between two modules modelled in IDEA StatiCa can be seen in Figure 4.10. The modules are connected at the corners through bolted endplates. This can be further extended to a connection between four different modules as well. Each of the corner points will then have an extra plate in the direction perpendicular to the one shown in the figure. The endplates have a thickness of 12mm and the bolts are of type M22 8.8. There are four of these large bolts connecting the modules per plate. The end plates that are welded to the modules are not placed directly on the corner but stick out a little, similar to the joints in Figure 4.8. The forces acting on the structure are therefore eccentric and moments are introduced. The reason that this is done is to provide space for the installation of the bolts. If the corners of the modules are placed too close to each other the diagonals would be in the way of placing the bolts of the end plate connection. To prevent this problem the distance between the edge of the corner of a module and the endplates must be 1.2 times the length of the bolt. The endplates have a thickness of 12 mm each so at least bolts of 40 mm must be used to account for the bolt head and the nut. This would result in a minimal distance between the endplate and the corner of the module of 48 mm.

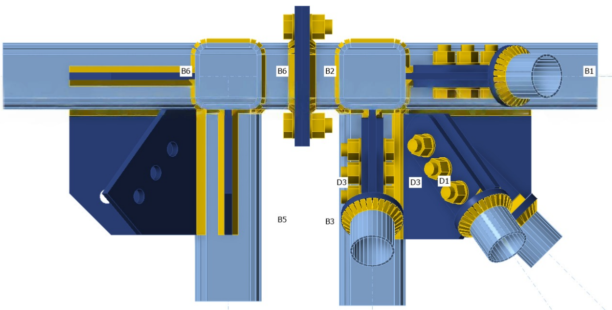
The beams and columns are made from of SHS elements and the diagonals are CHS elements. The element sizes can differ depending on the depth of the module, see also Table 3.6. The SHS elements are connected to each other with welds. This can be prefabricated in a factory. The diagonals are bolted to the main frame with the use of gusset plates. These are the plates that are welded to the frame, often connected to two beams. The only exception is for the diagonal that is directed in the space diagonal and connects two opposite vertices that are not on the same face. This diagonal is connected with a gusset plate which has a welded connection to one beam. The number of diagonals in each corner point can be varied. Depending on the module type zero to four different diagonals can end up at a corner point of a module. As an example, a corner with four different diagonals is shown in Figure 4.10.



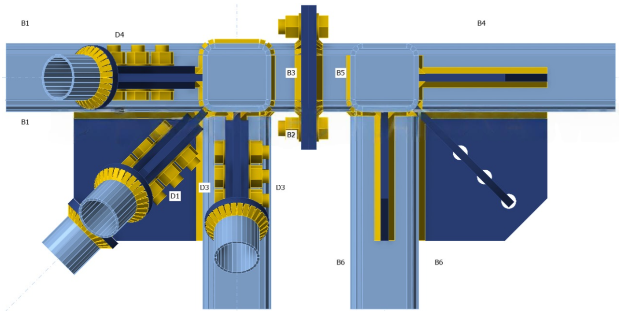
(a) 3D view



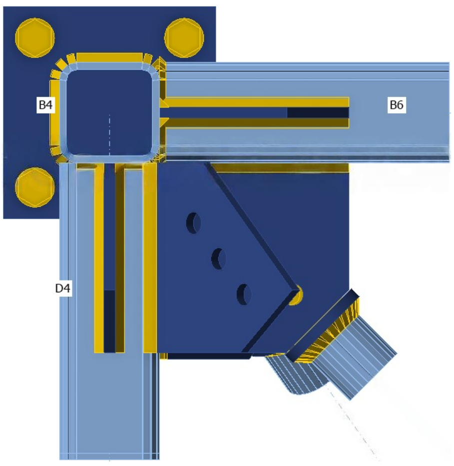
(b) Top view



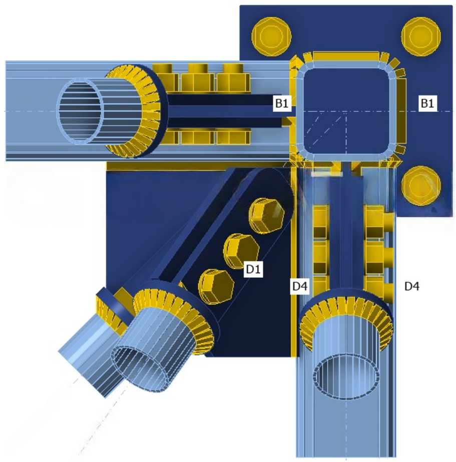
(c) Front view



(d) Bottom view



(e) Side view 1



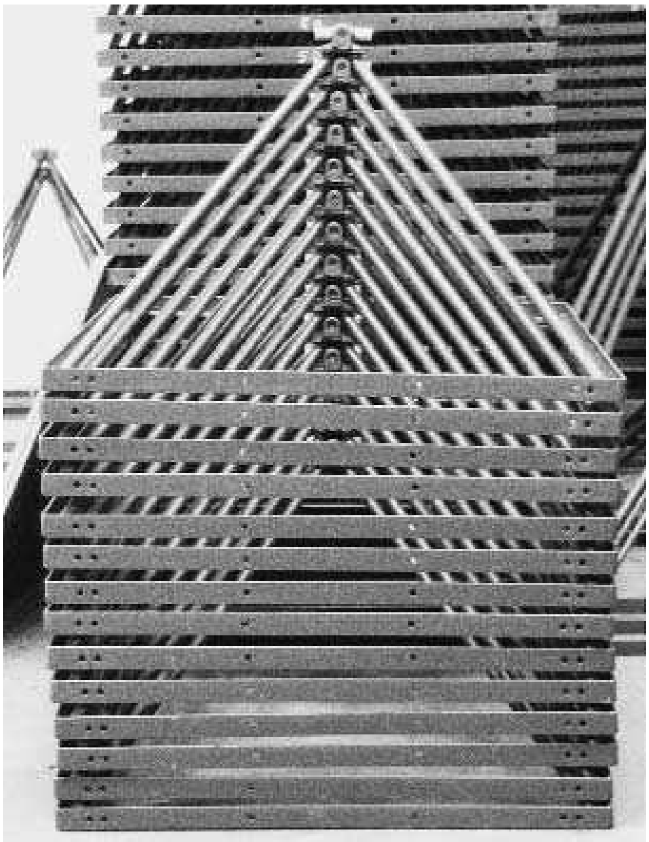
(f) Side view 2

Figure 4.10: Design of the intermodular connection showing two modules

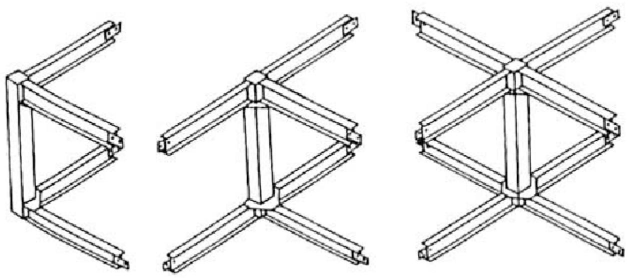
4.3 Design of intra modular connections

Transportation of entire modules can be cumbersome unless it is possible to stack them on a transportation vehicle efficiently. Otherwise, the amount of air transported can be very high. This leads to an increase in transportation costs as well as an increase in greenhouse gas emissions. Square-based pyramids or half-octahedra of Space Deck are examples of modules that can be easily stacked, see also Figure 3.7. Cubic space frames often require more space [2].

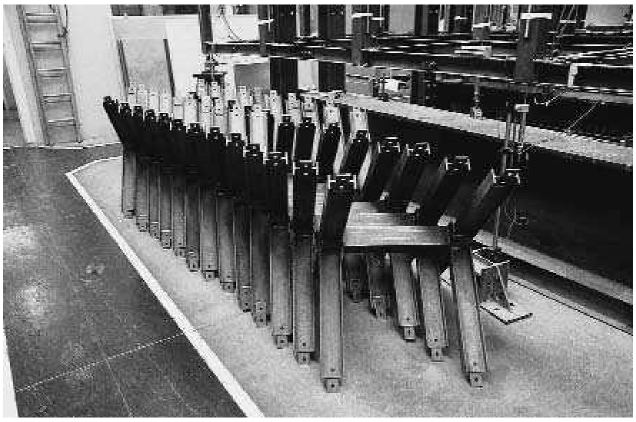
The CUBIC Space Frame developed in the late 1970s is a modular system that consists purely of cubes without any diagonals. Therefore, the loads are resisted by frame action and bending moments and shear forces are present besides the axial forces. It is based on the so-called "Vierendeel" girder. In fact the system can be seen as a 3D structure of intersecting Vierendeel girders. The chords or beams consist of open sections that are connected to SHS columns. These connections are considered to be rigid. At the midway between the intersections of the chords, pin joints are inserted. This is done both at the bottom and the top of the modules and allows them to be broken down into smaller modules with an "X", "T" or "L" shape [2], see Figure 4.11b. The different smaller modules can be nested. This allows for easier transportation of the modules to the construction site. However, it can sometimes still be difficult if a large grid size is chosen in a structure. The connections are realised with mid-chord splice joints with overlapping plates as can also be seen in Figure 4.11b. These plates are welded to the "X", "T" and "L" modules in the factory. On-site assembly is performed by connecting the modules with high-strength bolts [2]. The modules forming the CUBIC system described here were used to design the space frame roof of a maintenance hangar at Stansted Airport. The module size used in this structure was 2.0 x 3.5 m approximately and had a overall depth of 4.0 m. The vertical members that were used for these modules were of SHS with dimensions 200 x 200 or 300 x 300 mm. The roof that was covered by the space frame had an area of approximately 170 x 98 m [57]. This gives a good indication of what the element size of the modules for the topologically reconfigurable modules could be. The weight of the aforementioned space frame has a total weight of 920 tons which comes down to roughly 55 kg/m^2.



(a) Efficient stacking of Space Deck modules (Source: Chilton [2])



(b) CUBIC Space frame modules of "L", "T" and "X" shape (Source: Chilton [2])



(c) CUBIC Space frame modules nested for transportation (Source: Chilton [2])

Figure 4.11: Transportable modules: Space deck system and CUBIC system

To overcome the problem of moving too much air and no material with the modules designed in Section 3 solutions are sought that are similar to the ones mentioned in this chapter. As starting point for the design the CUBIC system is chosen since it comprises a double quadrilateral mesh. This system consists however of open steel sections such as IPE or HE-sections. For this reason, the different smaller modules shown in Figure 4.11b can easily be connected with steel plates overlapping plates. Hollow steel sections cannot be connected that way. Different solutions are sought. An overview of where which joints are used in a structure is given in Figure 4.12. Both intermodular and intramodular connections are shown here.

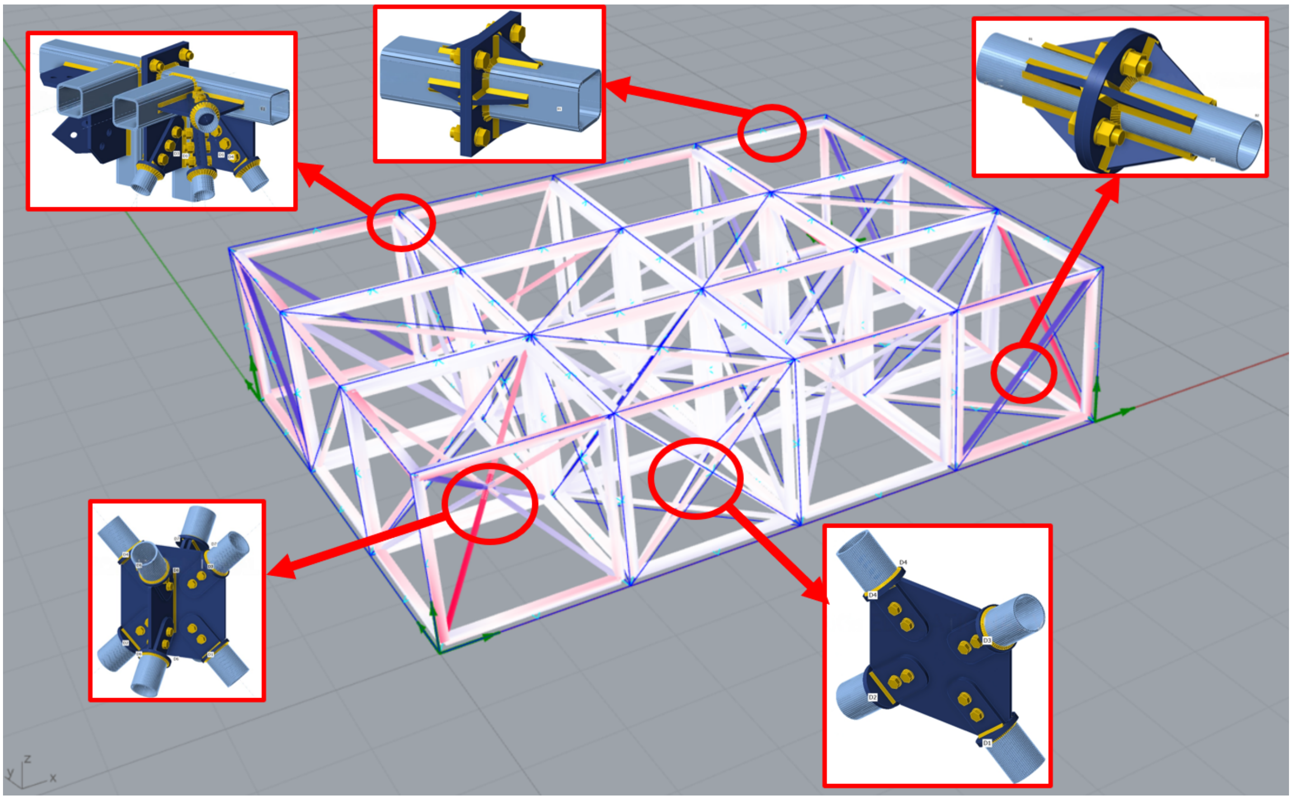
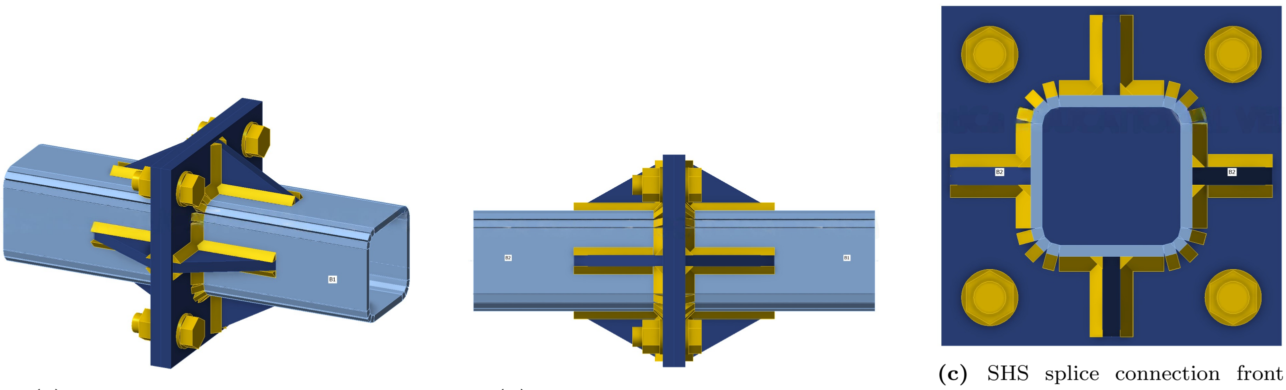


Figure 4.12: Overview of the use of different joints in a structure

SHS and CHS splice connections that are used are shown in Figure 4.13 and Figure 4.14. These bolted connections allow for quick construction and are feasible for hollow cross-sections. Also, the resulting forces are transferred axially leading to no eccentricity of forces in the members. Both splices are made by welding endplates to the cross-sections and bolt those endplates together. The bolt type that is used for the SHS splice connections is M18 x 8.8 and there are four of those, one for every corner. There are plate stiffeners that are used to bring down stress levels in the endplates. Each side of the element has a stiffener connected to it, four in total. The SHS splice connection has also got four bolts of type M18 8.8. It also has four plate stiffeners attached to the elements to reduce stress levels in the endplates. These plate stiffeners, also called ribs also make the joints behave more rigidly. This is needed because the stiffnesses that are needed are high. Some of the diagonals can experience large compressional forces.

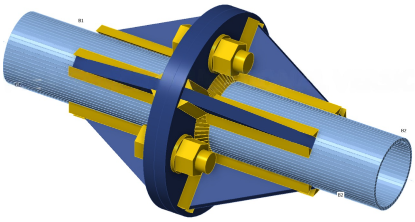


(a) SHS splice connection 3D view

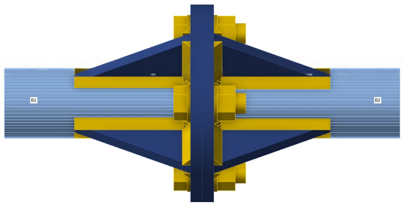
(b) SHS splice connection side view

(c) SHS splice connection front view

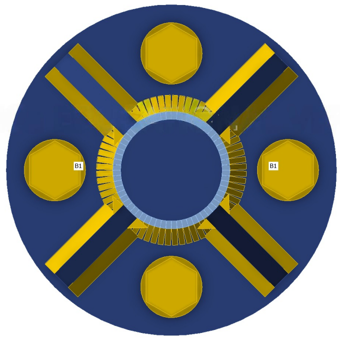
Figure 4.13: SHS splice connection (connection types C and D)



(a) CHS splice connection 3D view



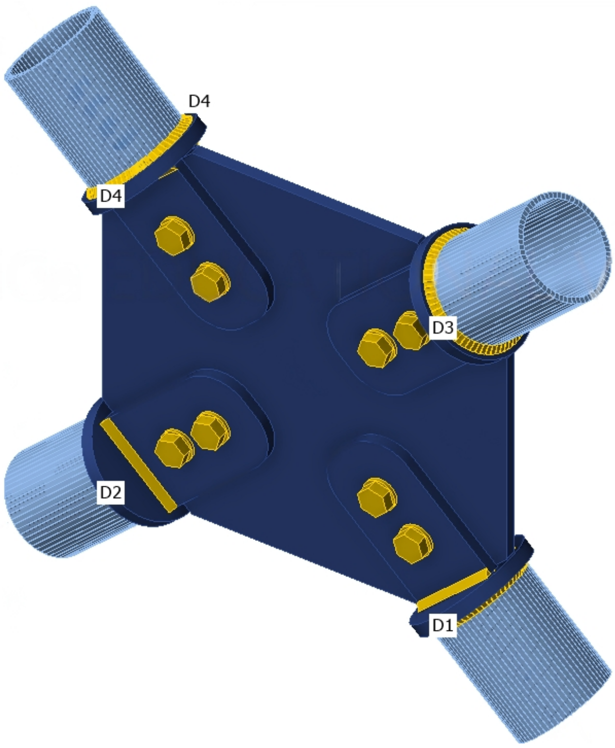
(b) CHS splice connection side view



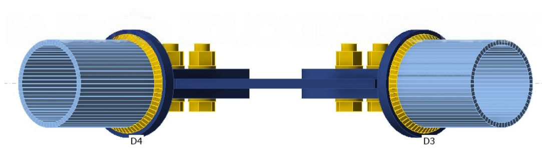
(c) CHS splice connection front view

Figure 4.14: CHS splice connection (connection type H)

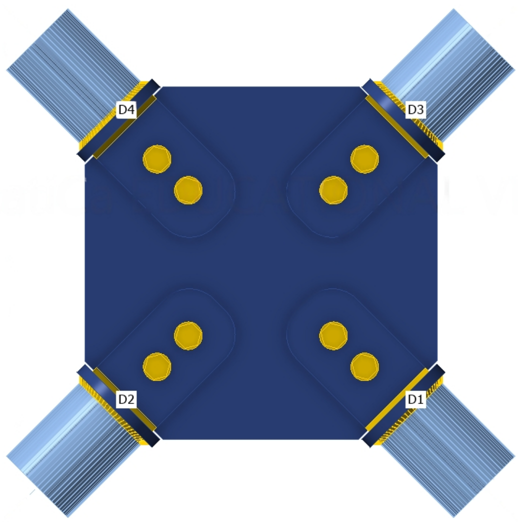
In some of the modules, the diagonals cross each other. At the intersections, it is required that they are bolted together such that the modules can be taken apart and assembled again as described earlier in this section. This results in complex nodes where multiple diagonals meet. There can be up to eight beams in total joining at one node. The beams are connected in the middle to steel plates with bolts. Figure 4.15 shows the joint for connecting four beams that form a cross-shaped bracing. A 3D view, side view, and top view are shown. The beams all have circular cap plates welded to the outside of their cross-section. These endplates in turn have two tongue plates welded to them. These can be bolted onto the large connecting plate in the middle of the node. The bolts are of type M16 x 8.8. The connection of the cap plate to the tongue plates is done with welds of thickness 8 mm. The cap plates are welded to the diagonals with welds of 6 mm thickness.



(a) 3D view for four beam node



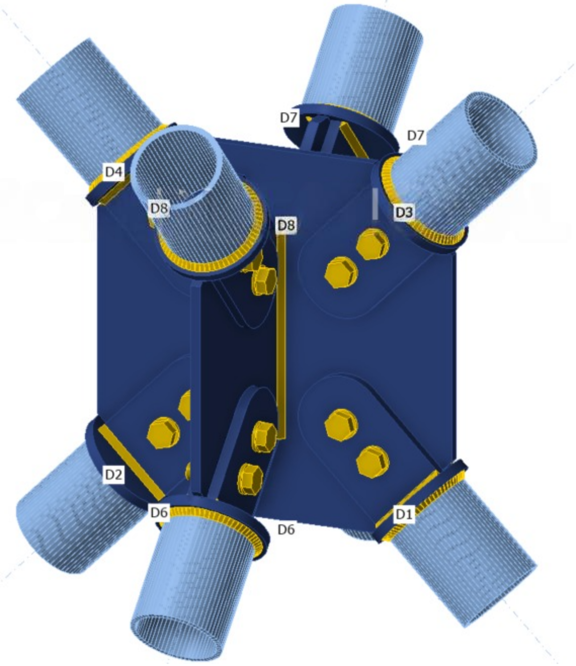
(b) Top view for four beam node



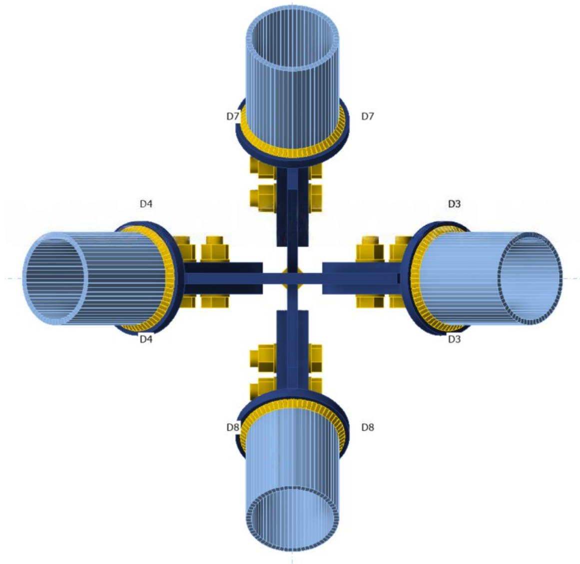
(c) Side view for four beam node

Figure 4.15: Intramodular connection for connecting four diagonals (connection type E)

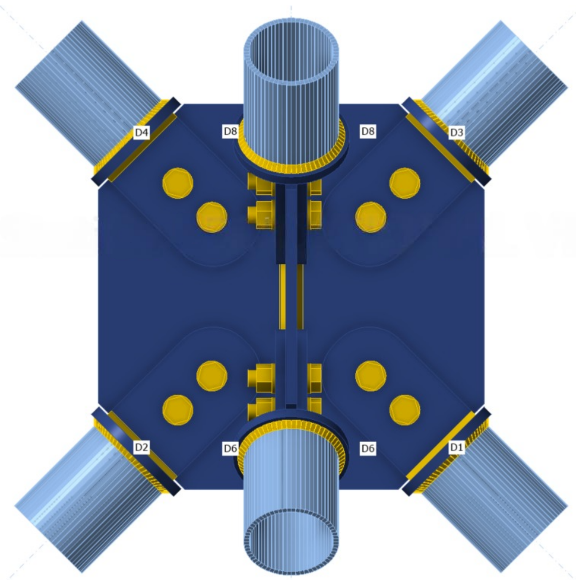
For the joint connecting eight diagonals, which can be seen in Figure 4.16, two extra plates are welded to the middle one. These two extra plates can connect another four different beams. The type of bolts used, plate thickness and throat thickness of the welds are all similar to the joint for connecting 4 diagonals.



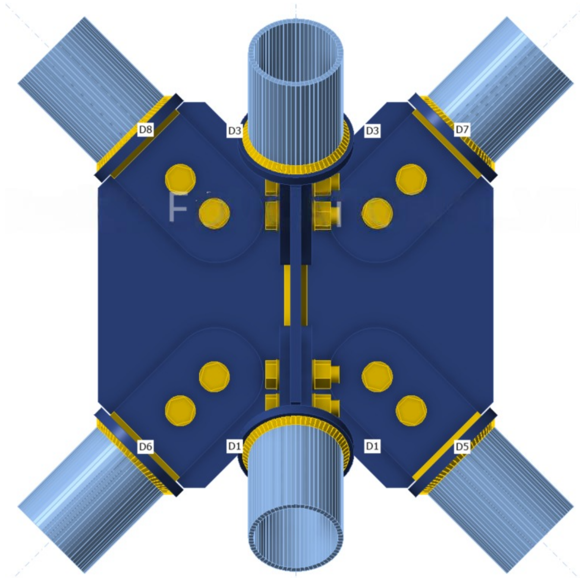
(a) 3D view for eight beam node



(b) Top view for eight beam node



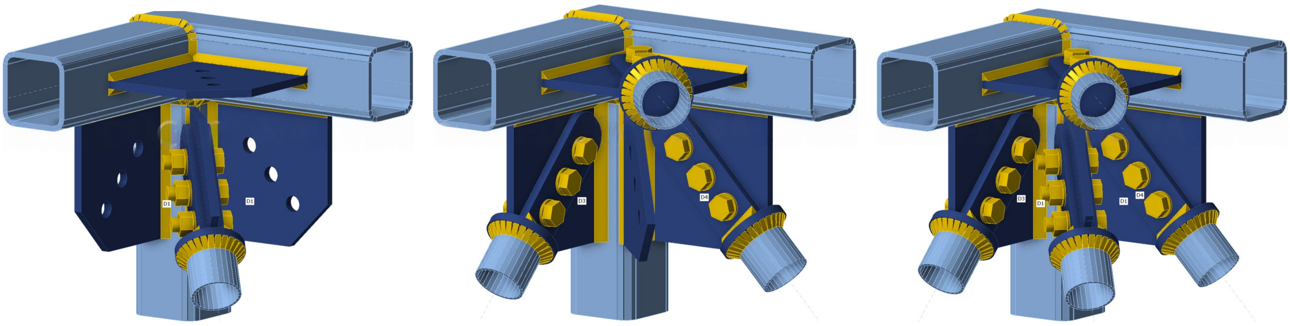
(c) Side view 1 for eight beam node



(d) Side view 2 for eight beam node

Figure 4.16: Intramodular connection views for connecting eight diagonals (connection type F)

As could have already been seen in Section 4.2, the diagonals are connected to the corners of the modules with gusset plates. These connections are established with 3 bolts per diagonal. The gusset plates are welded to the frame. The number of diagonals varies per module type as mentioned before. Regardless of the presence of diagonals, the gusset plates are connected to the corners as can be seen in Figure 4.17. In Figure 4.17a there is only one diagonal present but at every intersection of beams a gusset plate is added for example. These gusset plates act as stiffeners, providing extra stiffness to the joint. Besides, there is the possibility to add diagonals at a later stage in construction or when the structure is reused without having to weld new plates to the structure. This contributes to the overall circularity of the design, which is one of the goals. The options that are shown in Figure 4.17 are not the only three options. Depending on the module type there is more variation possible.



(a) Corner connection with 1 diagonal **(b)** Corner connection with 3 diagonals **(c)** Corner connection with 4 diagonals

Figure 4.17: Corner connection with gusset plates and varying number of diagonals (connection types A, B, G)

For some of the modules, it is difficult to transport them even if they are split up in parts. Stacking is cumbersome for module 8 in particular. It is only possible to transport it in a single piece. For modules 9 to 12, the splitting into submodules results in a large number of unique parts which is not wanted for the goal of circularity. Therefore, modules 8 to 12 are dropped from the initial catalog. The modules left are module types 1 to 7.

The modules are designed to be circular as mentioned before. First, the different module parts are made. Then these parts are stacked and transported with trucks for instance. This depends on the module size. Most containers have a width of 2.35m and varying lengths. The modules with a depth of 2.0 m can be stacked in these containers but for larger modules, trailers must be used. Then the cubic modules are constructed on sight. Subsequently, these modules are placed in position in a structure. At the end of the life of the structure, the modules can be reclaimed. Finally, they can be reused in a different structure, keeping their topology or changing it by disconnecting and reconnecting different diagonals resulting in slightly different module types. The life cycle of the modules is visualised in Figure 4.18.

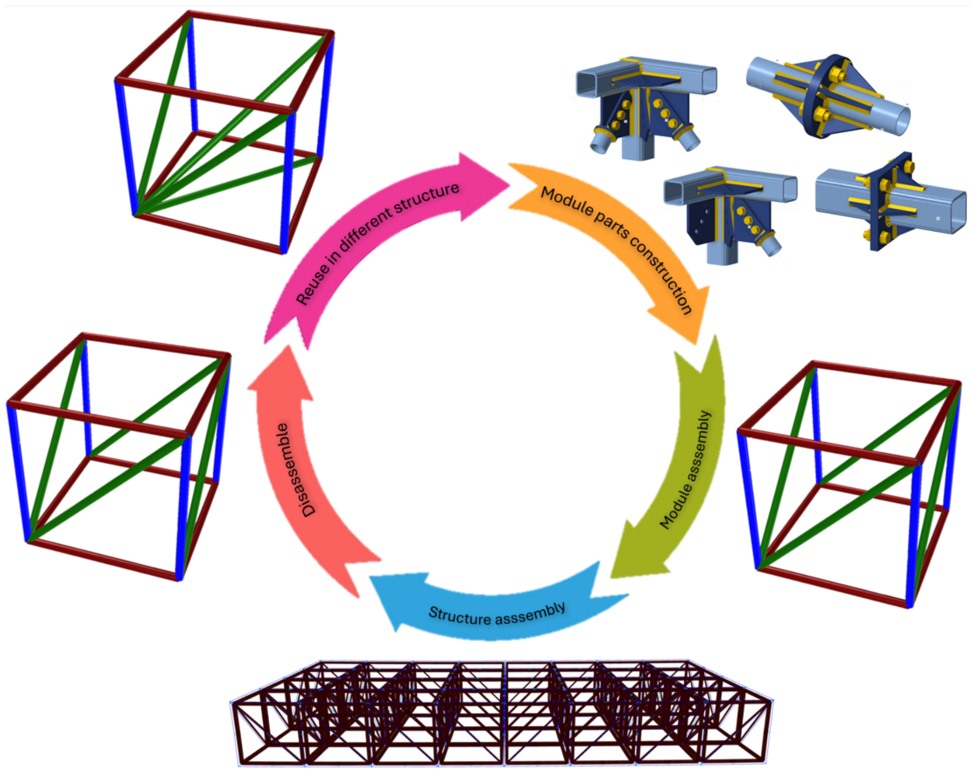


Figure 4.18: Life cycle of the modules

4.4 Mass of the joints

The mass of the joints can be found in Table 4.1. This is the added mass of the steel components, welds, and bolts and it is taken directly from IDEA StatiCa. The intermodular joint is split into its different connections. This makes it easier to later calculate the mass of the joints in a complex structure where the number of diagonals per corner varies.

Joint type	**Mass [kg]**
CHS splice (H)	6.59
SHS splice (C, D)	8.30
Four beam node (E)	17.76
Eight beam node (F)	36.16
Corner joint with 0 diagonals (A, B, G)	0.27
Corner joint with 1 diagonal (A, B, G)	5.55
Corner joint with 2 diagonals (A, B, G)	10.84
Corner joint 3 diagonals (A, B, G)	16.12
Corner joint 4 diagonals (A, B, G)	21.39

Table 4.1: Mass of the joints

With the mass of the different types of joints known the mass per module can be approximated. The mass of the elements and the mass of the joints is added to come to the total mass of each module type. The mass of the joints without splices is also calculated. For modules with joint types F and E, these joints are preserved because welding is not feasible for them. As can be seen from Table 4.2 the modules without splices have significantly lower mass than the modules with splices. The mean mass of the modules with splices is 697.0 kg and without splices 549.6 kg. The difference in the percentage of mass contributed by the joints is large. This percentage is twice as large in the case that splice joints are added to the design.

Table 4.2: Mass of the modules including joints

Module type	**Mass elements [kg]**	**Mass joints with splices [kg]**	**Mass joints without splices [kg]**	**Total mass with splices [kg]**	**Total mass without splices [kg]**	**Percentage of mass joints with splices**	**Percentage of mass joints without splices**
1	438.8	204.7	65.6	643.5	504.3	32%	13%
2	416.2	165.1	39.1	581.3	455.3	28%	9%
3	424.8	180.2	80.6	604.9	505.3	30%	16%
4	515.1	335.1	129.0	850.3	644.1	39%	20%
5	393.6	140.6	41.0	534.2	434.6	26%	9%
6	413.3	170.4	44.4	583.7	457.7	29%	10%
7	438.8	204.7	65.6	643.5	504.3	32%	13%
8	577.5	413.4	313.8	990.9	891.3	42%	35%
Mean	**452.3**	**226.8**	**97.4**	**679.0**	**549.6**	**32%**	**16%**

4.5 Joint strength

The joints are subjected to various forces and moments and have to withstand those. To see in what range of values the forces and moments occur, again the structure of Section 3.4 is analysed. This is done for every module size between 2.0 and 3.5 m. In Appendix C the distribution of normal forces, and moments can be found. Extreme values are also given there in tables. Furthermore, the spread in moment/normal force (|M|/N) ratios acting on the different elements can be found in Appendix C.7. The absolute value of the moment is used because the elements are symmetric so there is no difference in the behaviour of the elements between positive and negative moments. By being consistent with this there is also a direct insight if an element is in tension or compression. Compression forces are negative and tension forces are positive. The values are explicitly calculated for the modules with a depth of 2.0m. These ratios are used for the analysis of the different joints in IDEA StatiCa. There are many small forces in the model that go with large moments resulting in large |M|/N ratios. These values will not all be used in analysis because they are not representative for the behaviour of the structure. To overcome this problem outliers are omitted in Appendix C.7 with the use of boxplots. |M|/N ratios which are inside the Interquartile Range (IQR) are used for further analysis. These ranges will be used to do stiffness calculations with in IDEA StatiCa.

4.6 Sensitivity analysis

The |M|/N ratios of the beams are low. Most of the forces in space frames are axial as mentioned before. It is expected that the rotational stiffnesses of the joints have little effect on the outcome of the simulations because of this. Therefore, a sensitivity analysis is performed on the different stiffnesses used for joints in the model.

Again the simple space frame from Section 3.4 is used. It is not about the topology optimisation in this analysis and for good understanding a structure with a regular layout is chosen. The only module type used here is of type 6. The disadvantage is, however, that this module does not include joint types E and F. Therefore, the sensitivity to variation in joint stiffness of these joints is not analysed in this analysis. These joint types are of course relevant but this study is mainly to see if translational and rotational stiffness are both useful in design or if they can be neglected. For the outcome of this answer, the joint types E and F do not necessarily have to be analysed.

The structure is tested with the same Load Combination (LC)s as used before in Section 3.4. First, the translational stiffness is altered. The stiffnesses that are chosen are taken from Table 3.5. These values are multiplied with factors that make them vary in order of magnitude. The multiplication varies from 0.1 to 10^4 for the translational stiffness. Values closer to 0 than this range are not meaningful. To get a better understanding of what happens the stiffnesses of the joint types are changed individually instead of all at once and the deflection and elastic energy are measured. These are both measures for the performance of the structure under loading. It can be seen from Figure 4.19 that the deflection and the elastic energy are most affected by the joint stiffness of joints D and G which are splice joints of the beams and the joints of the diagonals to the corner points. The most influential after that is the stiffness of the diagonal splices (H). The other joint types have less influence on the behavior of the structure. The reason that the diagonals are so important is that they can transfer the forces to the supports of the structure when they are stiff enough.

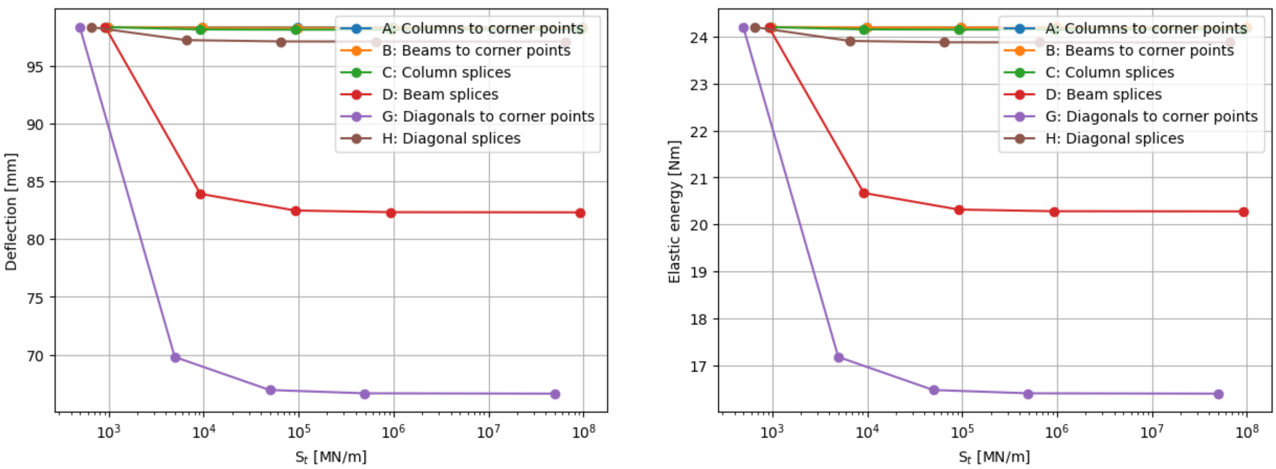


Figure 4.19: Sensitivity for varying S_t values in terms of deflection and elastic energy

The effect of changes in rotational stiffness can be observed in Figure 4.20. The range in orders of magnitude is larger for the rotational stiffness because a quasi-null rotational stiffness can be regarded as pinned. Therefore, these low values are meaningful in this case. The deflection and change in elastic energy are extreme for the diagonals connected to the corner points. The variance in deflection and elastic energy caused by the change in rotational stiffness is much lower than the change in translational stiffness. Besides, from this analysis it seems that the way the diagonals are connected to the corner points of the modules (joint G) seems relevant to the translational stiffness. The reason is that all joints are very stiff. The way the diagonals are connected to the frame greatly impacts the force distribution in the system. This is because of the large differences in moment distribution it causes.

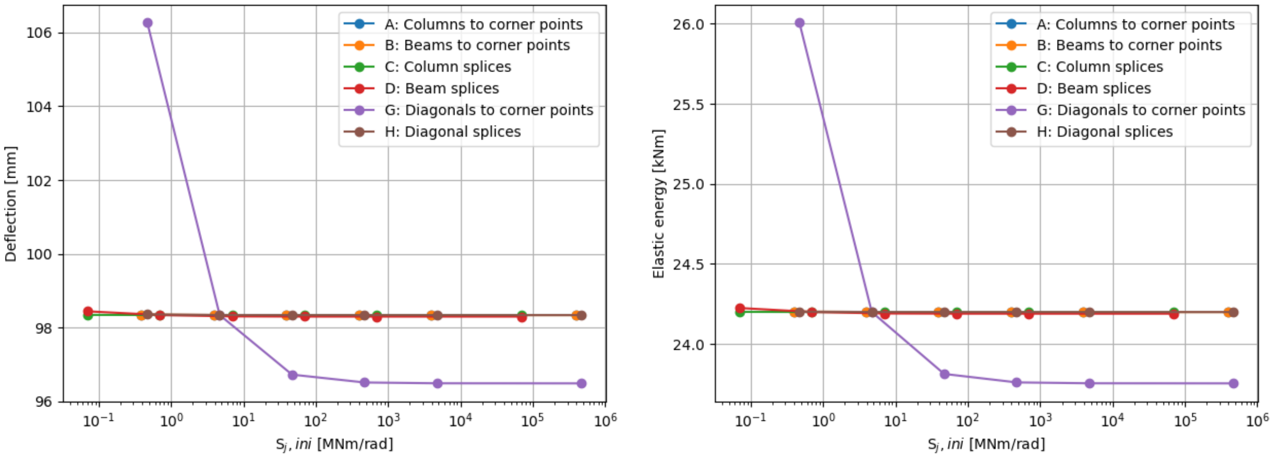


Figure 4.20: Sensitivity for varying $S_{j,ini}$ values in terms of deflection and elastic energy

There are significant changes in deflection and elastic energy because of variance in the stiffness of the joints. This is the case for translational and rotational stiffness. This analysis shows that these stiffnesses are relevant to the model and must both be further analysed. The chosen stiffness values are not final. In the upcoming sections the actual joints will be analysed in detail.

4.7 Stiffness calculation model IDEA StatiCa

It is important to calculate the stiffness of the different joints. This is needed for the Grasshopper model to be precise in its calculations. Stiffnesses of the joints have a large influence on the stress distribution in the system and on the deformation of the structure. This was the conclusion of Section 4.6. IDEA StatiCa software is used for the determination of the stiffnesses. For every member that is connected to the node, there is an individual joint stiffness. The rotational and translational stiffnesses can be obtained from the slopes of the M - ϕ and N - δ graphs that can be computed in the software. The δ is the local cross-section deformation in meters and ϕ is the rotation in radians.

Both rotational and translational stiffness are important for the model and will be determined for the different joints. IDEA StatiCa calculates the total rotation of a joint which consists of rotation caused by bending and rotation of the joint. To find only the rotation of the joint the software subtracts the rotation of a second model which has a completely rigid joint. See for an illustration Figure 4.21. It is summarised in Equation (4.1).

$$k_{joint} = \frac{M}{\theta_2} = \frac{M}{\theta_{tot} - \theta_1} \tag{4.1}$$

In which:

M = Applied moment [kNm]

θ_{tot} = Overall rotation [rad]

θ_1 = Rotation of the member [rad]

θ_1 = Rotation of the joint [rad]

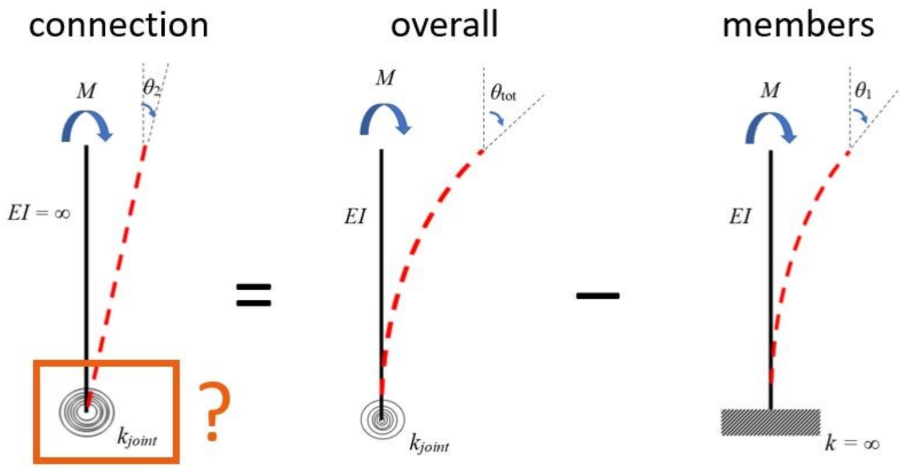


Figure 4.21: Stiffness analysis calculation in IDEA StatiCa (Source: Idea Statica [55])

4.8 Joint stiffness variability in the model

The stiffnesses are the result of the strength of the joint and the combination of applied moment and axial load on the joint. For a different load combination the stiffness values can change. The model made in Grasshopper makes a different configuration in the structure for every genome. Every time there are changes in forces and moments in the members. The loads on the joints are therefore not predictable. To see what happens the joints are repeatedly analysed in IDEA StatiCa for different |M|/N ratios. Here M is the applied moment at the end of the member and N is the applied axial force. The range in axial forces for the truss model can be obtained from Appendix C. Moreover, the spread in moments M_y and M_z can be found here. These ranges are used for further calculations of stiffness dependencies. The value for the moment M is varied between the maximum and minimum |M|/N values without exceeding the maximum moment or normal force. These ratios are used to test each joint type for different loads in IDEA StatiCa. For each joint type, a stiffness analysis report can be found in Appendix H. For simplicity, only the joints for modules with a depth of 2.0 m are examined. It is expected that similar joints with larger element cross-sections will experience the same stiffness values.

As mentioned before, the stiffness is dependent on the applied force and moment. This causes the force and moment distribution in the structure to change again, leading to a new stiffness in the joints. This process goes on until it converges. To make the stiffness of the joints vary like this an iteration loop is added to the model. This is done with the use of another plugin for Grasshopper called OctopusE in which the E stands for 'Explicit'. This is a version of the widely used Octopus developed by Vierlinger. This software was initially an extension designed to complement Galapagos and has the ability to optimise multiple objectives. It provides the user with multiple solutions with different trade-offs. During the development of this plugin also a loop component was developed. The loop component is based on Anemone which was a plugin made for looping in Grasshopper. The main difference to Anemone or the similar Hoopsnake is that the components are evaluated in a separate context in the loop component. This makes it work with Galapagos and Octopus [24]. The recursive loop has been tried at first with Anemone but it was found that the Galapagos solver did not wait for the iteration to finish. This has to do with the fact that Grasshopper is designed to follow data flow going from left to right in the script. This makes a programming loop that does not follow the normal concept of Grasshopper. Therefore, the Galapagos solver did not wait for Anemone to complete its iteration. This resulted in Galapagos analysing every stiffness iteration which was incorrect. For this reason, the explicit loop component of Octopus is used. A drawback of using this component is that a boolean threshold can not easily be implemented. Therefore, all data of the different iterations is stored and the correct iteration numbers where a threshold value is reached are later accessed. This method requires a longer computation time because more solutions are calculated than would have been necessary with the option of stopping the algorithm at a threshold value.

The different joints are given an initial stiffness that is presented in Table 3.5. The model is then analysed and the forces and moments at the different nodes are calculated. For each node the moment-to-force ratios $|M|$/N in the connected elements are determined. From the stiffness analysis performed with IDEA StatiCa the stiffness values for different |M|/N values are known. The values obtained from IDEA StatiCa are in summary displayed in Appendix H and analysed in Appendix D. If the newly calculated |M|/N value of a joint falls within an interval between two known values from the stiffness analysis, a new stiffness value corresponding to that interval is assigned to the joint. These values for the stiffness are new inputs for the beginning of the loop. The procedure will start again with calculating the model, obtaining forces and moments, and calculating new stiffness values. Some of the rotational stiffness values calculated by IDEA StatiCa have a large peak. This occurs when the |M|/N ratio is close to 0. van Spengler observed similar peaks in his research on semi-rigid connections for grid shells. The peak is not caused by inconsistencies in the input data into the IDEA StatiCa model. It is for low load ratios representative for the behavior of the joint. However, for small values the results are unreliable. This is especially the case for the decrease after the peak that is observed [58]. For this reason, the peak values are chosen as cap values for these small |M|/N ratios.

It is observed that these very high values for the stiffness can cause numeric instability. This leads to rigid body modes and highly localised stresses that do not reflect the actual behavior of the structure. Since no stiffness is in reality infinite they are all capped. This capping of the graphs is visualised in Appendix D.3. The rotational stiffness has a minimum value of 0.1 MNm/rad. This iterative process will require more computation time for the model. Every calculation of an entire model with new stiffness values can take up to various seconds depending on the size of the analysed structure. To make sure that the computations do not continue for too long a threshold value is set. The stiffness values of the previous run inside the iteration are compared with the new ones. Both are transformed to log scale because the stiffness values can get large. Then the difference between the summed values is taken. If this difference is smaller than the threshold value of 1% of the previous summed stiffness the iteration is stopped. Equation (4.2) shows the calculation of the threshold value. Besides a maximum of 10 iterations is performed. Figure 4.22 shows the condition at which this iterative loop is ended.

$$\sum_i (|\log(x_{i,t-1}) - \log(x_{i,t})|) \leq 0.01 \cdot \sum_i \log(x_{i,t-1}) \quad \text{or} \quad t \geq 10 \tag{4.2}$$

In which:

$x_{i,t-1}$ = List of stiffness values of the previous iteration [-]

$x_{i,t}$ = List of stiffness values of the current iteration [-]

t = Current iteration number [-]

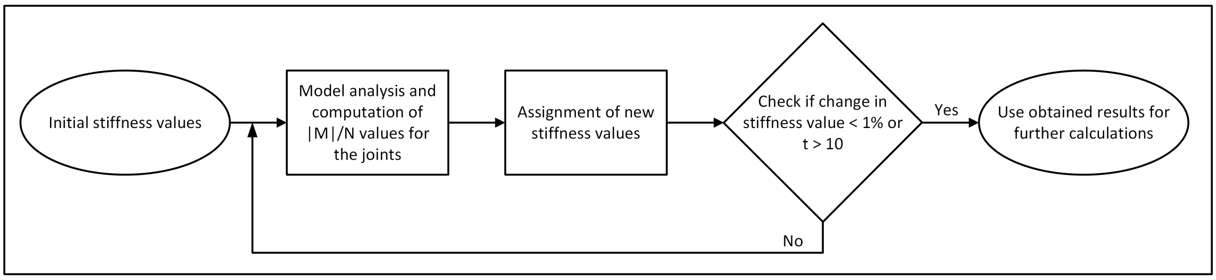


Figure 4.22: Scheme of the iterative loop for joint stiffness calculation

In the FEM model in Grasshopper it is possible to assign stiffnesses to joints using the Karamba3D joint-agent function. The stiffnesses are assigned to certain nodes. Within this model, it is not possible to create an intermodular joint as shown in Figure 4.10. This joint contains multiple nodes at a single joint location but only one node can be modelled. A simplification of this joint in the model is therefore unavoidable. The translational and rotational stiffness of the beams and columns to the corner of the module are taken as the minimum values of the different beams. For the members of the intermodular connection, the actual stiffness values are possibly higher than what is used in the model. This is because only part of the connection can be analysed at once. Modeling an entire intermodular connection in IDEA StatiCa is not possible because of the multiple nodes that are present.

5 Optimisation process

In optimisation of structures, the volume of the used materials must be minimised while maintaining structural equilibrium and strength. This can be achieved by altering material properties, member sizes, or geometry. Different types of optimisation are size, shape, and topology optimisation. Size optimisation is the alteration of the members' cross sections to make them fit for their purpose. Shape optimisation refers to making changes in the coordinates of the nodes [32]. For space frames, topology optimisation is often applied [27]. This is the spatial optimisation of material in a certain domain. For space frames, it means that a set of vertices is connected using a minimum volume of connecting members [27].

In this thesis, simple structures will be optimized with the use of Galapagos. This plugin for Grasshopper provides the user with a GA with which difficult problems can be solved. This is a subdivision of the evolutionary algorithms. A GA is inspired by the natural selection process that Darwin described. It is a population-based stochastic algorithm where the fittest genes of a population are simulated [1]. From a wide possibility of different solutions firstly an initial population is taken. This is done at random. Each population has a genome that encodes the solution. In this case, the genome is the solution chosen for each module location. This comprises the following genes: module type chosen, rotation in x, rotation in y, rotation in z. These rotations can be the following values: 0, $\frac{1}{2}\pi$, π and $\frac{3}{2}\pi$ radians. Modules will therefore always have right angles with each other. The population is then evaluated with a fitness function. After evaluation of the population, the parents for the next generation are selected. These selected parents are solutions with a relatively high fitness compared to the other solutions. These solutions are crossed over to create new solutions to be evaluated again in the next population. New solutions receive genes with the characteristics of the parents. The genes of part of the crossed offspring's chromosomes are randomly changed to generate new solutions that otherwise would be missed. The next generation consists of some of the best from the current generation, the crossover solutions, and the mutated crossover solutions. There are two ways in which the algorithm stops. One is when the results are converged and the other is when the specified maximum amount of iterations is reached [1]. The different steps of a standard GA can be seen in Figure 5.1.

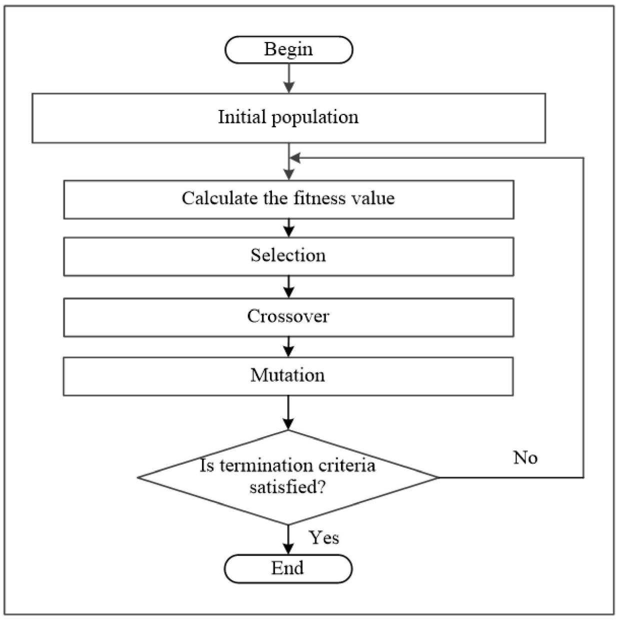


Figure 5.1: Flow chart of a genetic algorithm (Source: Albadr et al. [1])

The GA searches a parameter space to minimise or maximize a certain fitness function. The algorithm in this case minimises the weight of the total structure. The formulation for the minimization function can be seen below in Equation (5.1).

$$\min_{\rho_i A_i L_i} \quad W = \sum_{i=1}^{Ne} \rho_i A_i L_i \tag{5.1}$$

In which:

W = Weight of the structure [kg]

ρ_i = Material density [kg/m^3]

A_i = Cross-sectional area [m^2]

L_i = Length of the element [m]

N_e = Element number in the model [-]

In this equation, the weight of the connections is neglected. The weight of all the different elements that form the modules is summed which results in the total weight. The weight of the joints is significant as is shown in Section 4.4. For every module type the added mass of the joints is always around 30%, meaning that the mutual differences between the configurations with or without the weight of the joints are very similar. For this reason, the weight of the joints is neglected in the model for the stress constraint. Since the main goal of the optimisation is minimising the weight, it is taken into account here. The downside is that the extra weight of the joints causes extra loads. These loads are now neglected in the model. When designing a structure, the designer must be aware of this simplification and check it after the simulation.

Besides this, there are also a couple of constraints that define the solution space. These are related to Eurocode 3 constrictions for the Ultimate Limit State (ULS). The stress in the elements may not exceed the critical stress, see Equation (5.2). If this happens the material would fail and the structure could collapse. This is the case for every load combination. The utilisation of the material in each member can be measured with Karamba3D. A percentage of 100% utilisation in a single will serve as a constraint for the optimisation.

$$\frac{\sigma}{\sigma_{crit}} \leq 1.0 \tag{5.2}$$

In which:

σ = Stress in the member [N/mm^2]

σ_{crit} = Critical stress of the member [N/mm^2]

Also, the deflection of the structure may not exceed certain limits and is related to the Serviceability Limit State (SLS). The deflection of roofs should be limited to $\frac{1}{200}$ of the length of the roof for flat roofs according to the Dutch national annex of the Eurocode [45]. This applies to roofs that are not extensively used by people who walk on them. See Equation (3.3) for the formulation of the limit. This rule was also already used for the cross-section optimisation in Section 3.4.

The Buckling Load Factor (BLF) of the columns and beams may not be smaller than 1.0 which is critical. The BLF is the factor with which the load on a structure or element must be multiplied to make it buckle. If the values is smaller than 1.0 the existing load is therefore already large enough to make a structure buckle. The constraint is shown in Equation (5.3).

$$BLF \geq 1.0 \tag{5.3}$$

In which:

BLF = Buckling Load Factor [-]

To incorporate the boundary conditions in Equation (5.1), the constraints are included as barrier functions. These functions are used in optimisation to make sure that the solution doesn't go outside of the solution space and gives unfeasible solutions. A barrier function gives a large increase in the value of the fitness function when a certain limit is reached. Because this particular optimisation problem minimises the objective function's value, the solutions with relatively high values are recognized as unfeasible by the GA. There are multiple constraints and thus it is important to normalise them to try to make them all equally important. If one of the constraints yields significantly higher values compared to the others, it would disproportionately influence the results. The implementation of other constraints would therefore be redundant. Firstly, the value for the weight is normalised with Equation (5.4). This is a unity-based normalisation formula which is used to bring the value of the weight in the range of [0,1]. The minimum weight is determined by creating a structure with the same layout as the structure analysed. This is done automatically by the Grasshopper script. The module with the least cumulative length of elements is automatically the module with the smallest weight. This is because it has the least amount of diagonals while the number of beams and columns is the same for each module type. The difference weight of the structure that consists only of modules of the lightest module type is measured and used as the value for W_{min}. A similar computation is performed to determine the maximum weight. The script searches for the largest cumulative length of members to find the heaviest module type in the set. Then the maximum weight W_{max} is calculated by constructing a separate model with only this module type and measuring its weight.

$$W_{norm} = \frac{W - W_{min}}{W_{max} - W_{min}} \tag{5.4}$$

In which:

W_{norm} = Normalised weight [-]

W = Weight of the structure [kg]

W_{max} = Maximum weight of the structure [kg]

W_{min} = Minimum weight of the structure [kg]

For the deflection also a unity-based normalisation is performed. The range of the deflection is between 0 and the maximum allowed deflection d_{max}.

$$d_{norm} = \frac{d - 0}{d_{max} - 0} \tag{5.5}$$

In which:

d_{norm} = Normalised deflection [-]

d = Deflection of the structure [m]

d_{max} = Maximum allowed deflection [m]

In this model, the type of barrier function used for the constraints is the log-barrier method. The log-barrier method is used for inequality-constrained optimisation and is a type of interior-point technique [59]. It is a continuous function in which the value of the solution goes to infinity when it is closer to the boundary of the feasible region and almost starts to be unfeasible. An example of what log barriers can look like with different scaling parameters can be seen in Figure 5.2.

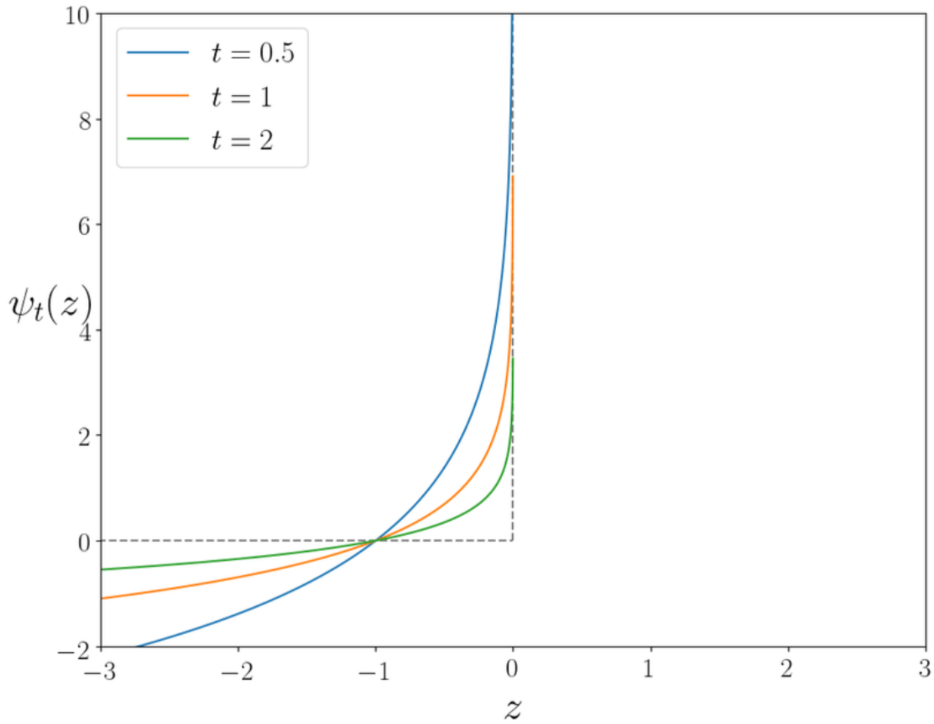


Figure 5.2: Example of a log barrier function for different scaling factors t (Source: Kervadec et al. [59])

In this function, it can be predefined how close to the constraint the solution will be found. The overall formula for the minimisation problem with log-barrier functions can be seen in Equation (5.6a). In this formula, f_1, f_2, and f_3 are the log-barrier functions for the earlier described constraints for deflection, utilisation, and buckling. The variables μ_1, μ_2 and μ_3 are scaling parameters. These are used to normalise the values of the different contributions of the constraint functions. However, these are already normalised in earlier calculations and therefore their values is initially taken as 1.0. They could later be used to alter the log-barrier functions based on their importance. The reason that a logarithmic barrier function is used is because it is continuous and therefore allows for a smoother transition between feasible and unfeasible regions. This helps to prevent numerical instability and also makes the convergence of the GA more robust. Besides, discrete functions would not give much freedom in finding solutions. Logarithmic barrier functions offer soft handling of the constraints which allows for more flexibility.

For the deflection, the value may be larger than the maximum deflection defined in Equation (3.3). Therefore, when this value is exceeded there will be a large penalty given to the optimisation function. This penalty increases quadratically. Normally the value inside the logarithmic function would go to 0 and the outcome is undefined. This will make the solution unfeasible. However, the solution with a deflection slightly larger than the maximum allowed would not receive a different penalty than a solution with a large deflection. This means the algorithm will recognize the solution with the least weight optimal. If all initial solutions are unfeasible the final solution can become a low weight structure but with very high deflection. To make sure that the algorithm learns, a smoother penalty function is added to values that go beyond the definition of the logarithmic barrier function. This will give a deflection that is larger than the maximum allowed deflection a penalty that becomes larger when the deflection becomes larger.

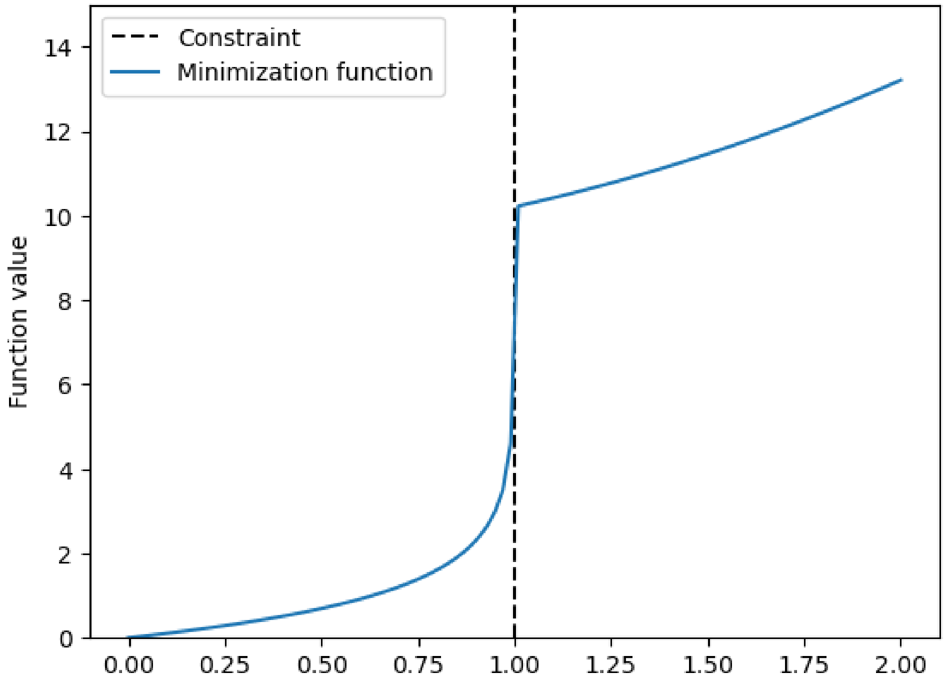


Figure 5.3: Log barrier function with quadratic penalty function

The logarithmic barrier function for the utilisation should be lower than a value of 1 to be feasible. This corresponds to a utilisation of the material equal to or less than 100%. Again, a value larger than 1 in this case will not give any result because the logarithm of a negative number is undefined. In the algorithm, values larger than 1.0 will therefore be given a quadratic penalty. This is similar to the deflection penalty.

The BLF is a value that is taken from the Karamba3D output for second-order analysis. As said before it is the factor that has to be multiplied by the load to achieve buckling of the structure. If this value is below 1.0 the structure will be unstable. So it follows that: BLF $\geq$ 1.0. This is the same as stating -BLF $\leq$ -1.0 and applying the same log function to it as for the other constraints. This results in the log barrier function in Equation (5.6d). The log barrier function is taken negative here as well because the logarithmic values for numbers larger than 1.0 are positive. Taking a negative here will result in a more minimal solution which is preferable for large BLFs. For values smaller than 1.0 the quadratic penalty function does not work the way it should. In this case, a value just below but close to 1.0 would lead to a more preferable value than a value close to 0. Therefore, a different penalty function is chosen that makes smaller values less optimal. In this case that is an exponential function. The difference between the two barriers for this constraint can be seen in Figure 5.4. The exponential function is less optimal the closer it gets to 0 which is how the constraint should be handled.

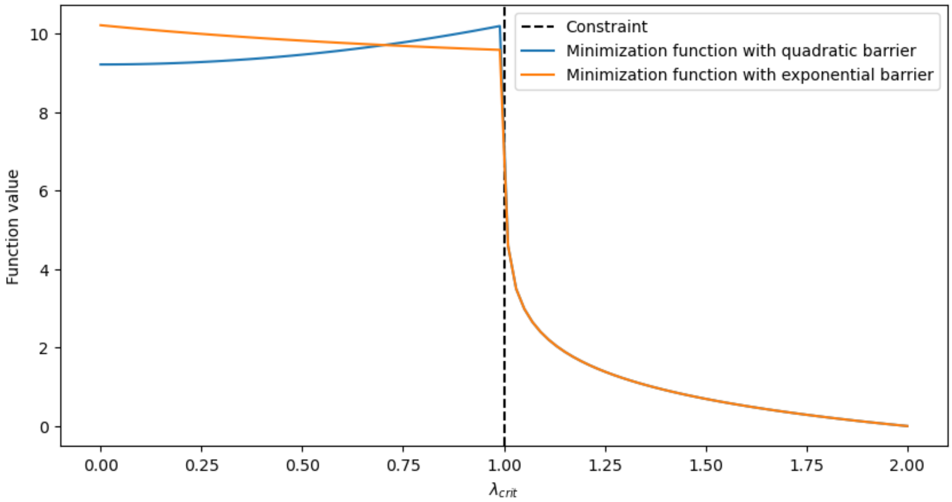


Figure 5.4: Log barrier function for buckling with quadratic and exponential penalty function

The BLF is an already normalised value that can have relatively high values. It is not unitised to a value between [0,1]. Several simulations must be performed to determine the maximum value of the BLF. This could vary for different structures of different sizes and support conditions. Therefore, the model is first tested to see if buckling is critical for the system. If this is not the case its contribution could be disregarded in the formula. Then it would only include a log-barrier function for the deflection and utilisation.

$$\min\ W_{\text{norm}} + \sum_{i=1}^{3} \mu_i f_i \tag{5.6a}$$

s.t.

$$f_1 = \begin{cases} -\ln(1 - d_{\text{norm}} + \epsilon) & \text{for } d_{norm} \leq 1.0 \\ -\ln(0.0001) + d_{\text{norm}}^2 & \text{otherwise} \end{cases} \tag{5.6b}$$

$$f_2 = \begin{cases} -\ln(1 - u + \epsilon) & \text{for } u \leq 1.0 \\ -\ln(0.0001) + u^2 & \text{otherwise} \end{cases} \tag{5.6c}$$

$$f_3 = \begin{cases} -\ln(BLF - 1 + \epsilon) & \text{for } BLF \geq 1.0 \\ -\ln(0.0001) + \exp(-BLF) & \text{otherwise} \end{cases} \tag{5.6d}$$

In which:

W_{norm} = Normalised weight of the structure [-]

μ_1, μ_2, μ_3 = Scaling parameters [-]

d_{norm} = Normalised deflection of the structure [-]

d = Deflection [m]

d_{max} = Maximum allowed deflection [m]

u = Maximum utilisation of a member in the model [-]

BLF = Buckling Load Factor [-]

ϵ = Small number to prevent numerical errors ($1e^{-6}$) [-]

The variables for the topology are purely geometric. Grasshopper has a so-called 'gene pools' component which consists of multiple sliders. These sliders control for each location in the specified grid which module type is assigned to each location and in which orientation. In the initial setup, there are four gene pools. One for the module type, one for the rotation around the local z-axis, one for the rotation around the local x-axis, and one for the rotation around the local y-axis. This turned out to be inefficient since there are $4 \cdot 4 \cdot 4 = 64$ rotations possible for every module type at every location. for a grid of 2 x 2 and with 8 possible modules this would result in a total of $(8 \cdot 64)^4 = 6.87 \cdot 10^{10}$ possible unique orientations. To bring the number of orientations down the unique module configurations are computed. Rotating the modules in every direction in different combinations leads to duplicate shapes. For instance, for symmetrical modules, there is only 1 orientation needed and not 64 identical ones. Table 5.1 shows the module types and how many unique orientations they can be rotated. The mean number of different modules per module size is 7.375. Module types 6 and 7 have a very asymmetric shape, resulting in many different possible configurations.

Table 5.1: Number of possible different orientations per module type

Module type	Nr. of different unique orientations [-]
1	6
2	8
3	1
4	1
5	6
6	12
7	24
8	1
Total	**59**

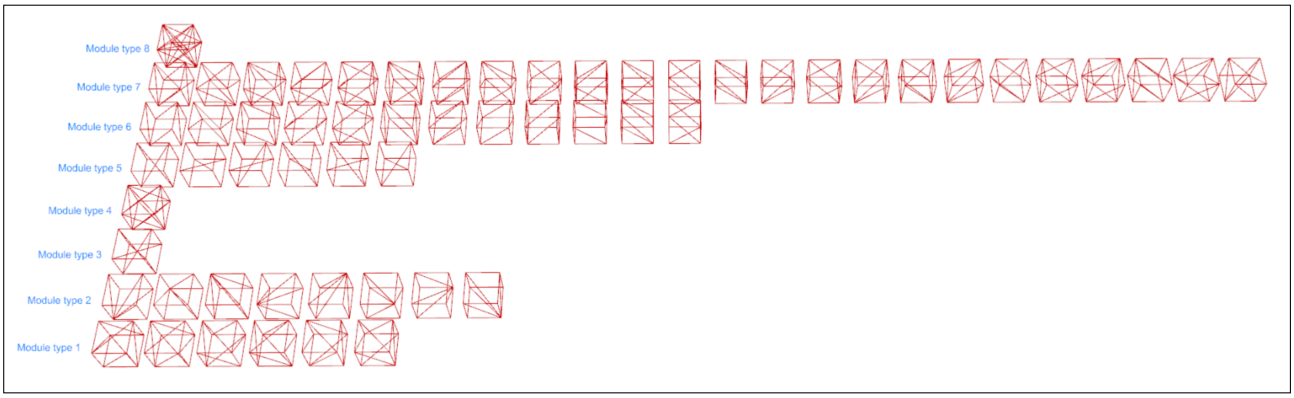


Figure 5.5: Catalog with 8 modules having unique orientations

By introducing only a single gene pool component that has access to all different uniquely oriented modules the number of possible configurations is reduced. There are now for the 2 x 2 square grid $59^4 = 1.21 \cdot 10^7$ possible configurations This is a reduction of 3 orders of magnitude which is a large reduction. However, this is still a too large number to calculate all possibilities with brute force. Since module type 7 has more than 3 times the mean number of possible orientations this module type will not be used for further computations. As mentioned before module type 8 is too complex for construction and is therefore also omitted from the catalog. Module type 6 also has many different possible orientations but simple trusses can be constructed with this module easily. Therefore, it is decided to keep this module in the catalog. The renewed catalog can be seen in Figure 5.6. There are now $59 - 24 - 1 = 34$ options for each cell. This will bring the total number of possible configurations on the same grid down to $(34)^4 = 1.34 \cdot 10^6$.

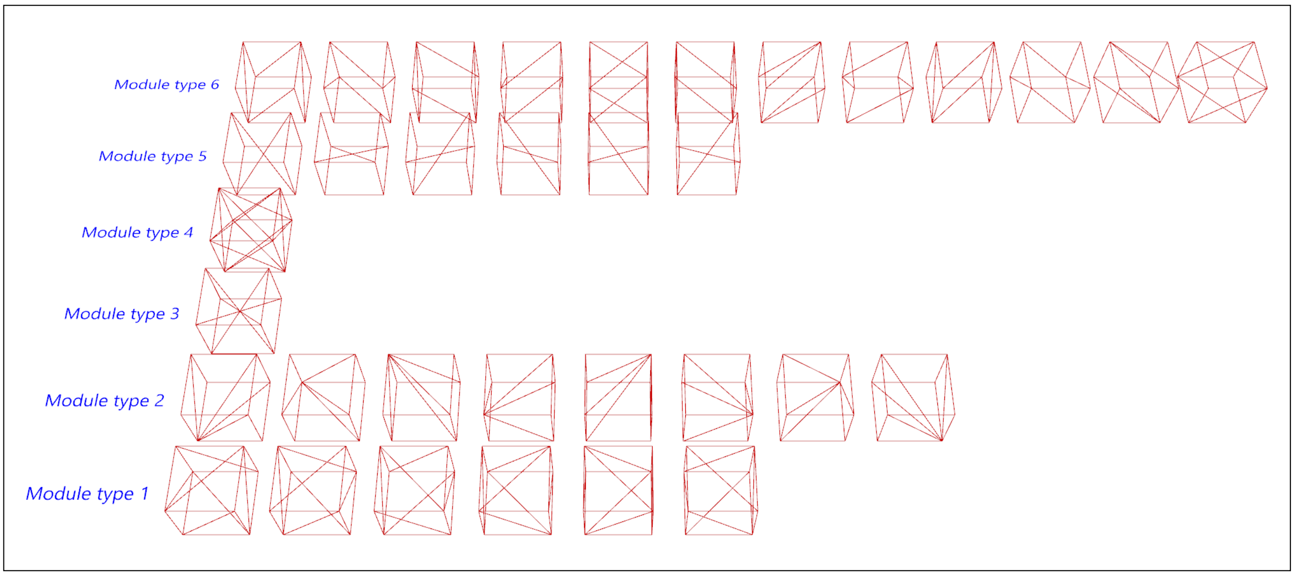


Figure 5.6: Catalog of module types 1 to 6 with modules having unique orientations

With this set of module orientations a bitmap is created. In this bitmap, each module type has been assigned a different color. These are the colors yellow, green, black, purple, blue and red for module types 1 to 6. The different orientations of each module type are indicated with shades of this color. For a certain configuration, the type of module used at each location and the orientation can be traced back to this bitmap. The bitmap is visualised in Figure 5.7.

Figure 5.7: Bitmap that shows the catalog of modules with unique orientations. The rotations in x, y, and z are given in subscripts for each module type.

In Galapagos it is possible to set a threshold value and a limit for the model's runtime. The value for the optimisation function depends on the value of the different constraints and thus indirectly on the loading. A lower value for the load results in lower stresses and therefore also a lower value for the utilisation. This would give a faster configuration to a low value for the objective function. This can result in only a small number of different combinations being addressed before the algorithm stops. To overcome this problem there will be no threshold value used for the different configurations. The runtime also varies with model size. Small models of for instance 2 x 2 take only a couple of hours to converge to a good solution. When using module types 1 to 6 the design space is $(34)^4 = 10^6$ different possibilities. Large models may take more than a day. No runtime limit is implemented. The solver has a maximum stagnant value. This is the maximum amount of generations that do not lead to a more optimised solution before the solver should stop. This value is set to 50 generations. Each population except for the first one consists of 50 solutions. There is a boost of 2 for the first generation which means that the number of individuals in this generation is doubled. This increases the chance of finding a good solution at the start and pushing the algorithm in the right direction. This means a total of 2550 solutions are processed by the algorithm. The percentage of solutions that are maintained for the next generation is set at a value of 5%. The inbreeding factor is at +75%. This factor defines the freedom of the algorithm to use similar or different genes to breed with. A high positive factor that is chosen here tells the algorithm to use genes with very similar results. The time it takes the model to calculate a structure is different per individual design. This is because the FEM analysis takes more time when a structure is more complex, even if the grid of the structure is the same. For a small scale model of 2 x 2, most designs take around 3 seconds to calculate. For larger structures, for instance, spans of 12 x 22 this can increase to more than 10 seconds per calculation.

6 Modelling workflow

In this section, the modelling workflow is described. This is done on the basis of Figure 6.1. The first step is the initial design of the structure. At first, the structural grid is determined. This includes grid size and the length and width of the structure. Then an initial geometry is chosen, which consists of randomly oriented modules within the grid. Furthermore, cross-sections are chosen based on Table 3.6. The joint design depends on the size of the cross-sections and initial geometry. Then the stiffness ranges of the joints are determined.

After this first stage, the optimisation procedure is started. The GA updates the geometry in every iteration until an optimum solution is found. This is the entire green block in the scheme. Within this process a stiffness iteration loop takes place, shown in the red block. Based on the |M|/N ratios found in the structure the joint stiffness is adjusted. The new joint stiffnesses lead to new |M|/N ratios in the structure. This process is repeated until convergence is reached or the maximum number of iterations. When 50 generations of iterations have been performed by the algorithm, the best found solution is obtained. This solution is then structurally analysed.

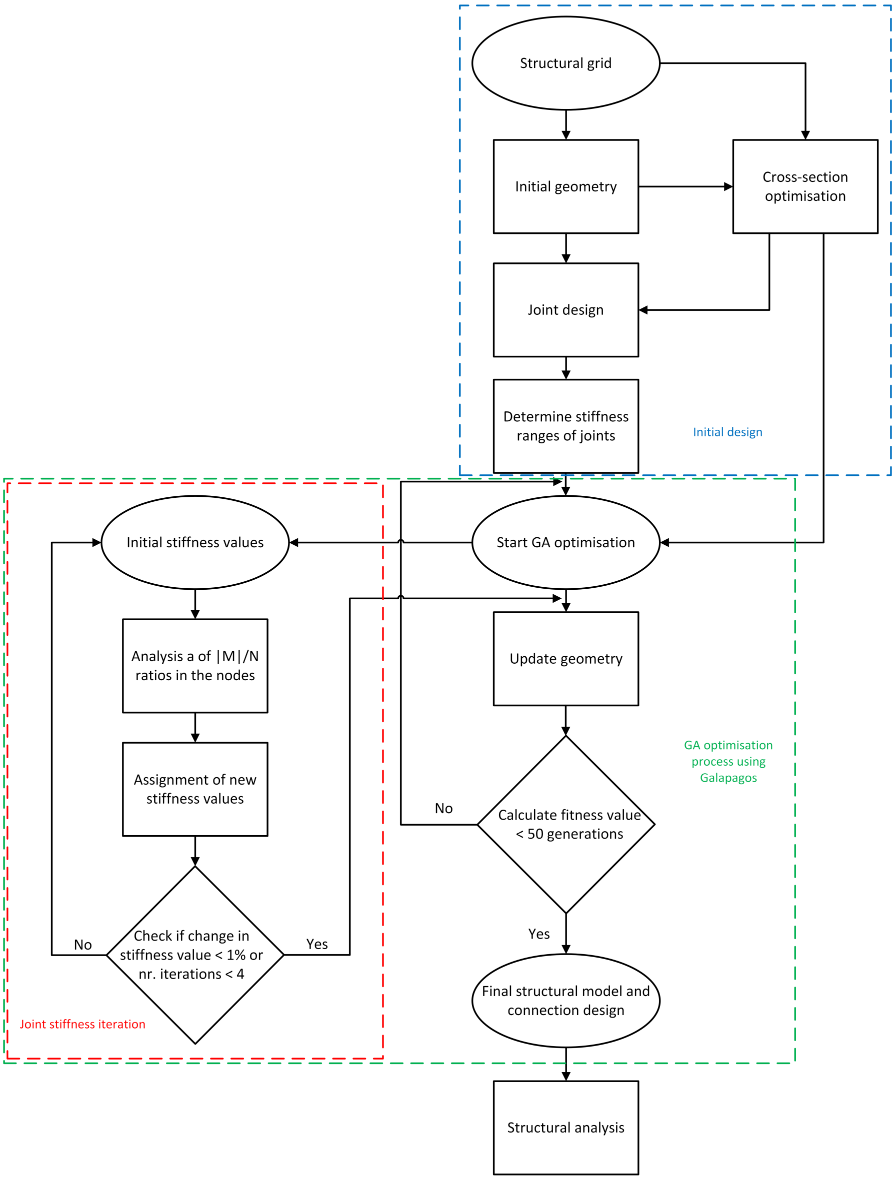


Figure 6.1: Modelling workflow

7 Model verification

There are many different aspects that make this model complex. The assignment of modules in different orientations to various locations, the eccentricity of the beams towards the center of the modules, the iteration loop for stiffness, and the GA that optimises the model all contribute to this complexity. First, it is checked if these aspects all work properly, especially the assignment of the joints. To do this a symmetrical model is made with module type 6 creating a truss. This truss has a grid of 4 x 16 to form a symmetrical one-way truss. The depth of the modules is 2.0 m. The supports are all pinned. They are constrained in the vertical direction and the support at (0,0,0) also has horizontal constraints.

7.1 Verification of the joints in the model

At first, it was observed that the truss experienced a large torsional moment. This was caused by internal forces because the deflection was not very large, only 11.46 cm. The moment can be observed from the asymmetrically and large reaction forces displayed in Figure 7.1. Also, there is a large horizontal force at the only support which translation in the x direction is fixed. To prevent rigid body rotation of the structure, there is a rotational constraint around the z-axis placed in the middle of the structure. The combination of active translational stiffness in the splices and the eccentricity of the elements towards the center of the modules causes the large internal forces. The partially overlapping elements experience high stresses due to the irregular local stiffness values at the splices. After increasing the stiffness to a large value ($1 \cdot 10^{12} MN/m$) for all splice connections this problem still existed. For that reason, the translational stiffness of the splices is turned off resulting in a fixed degree of freedom for this connection type. Although a rotational support was added to the model the first buckling mode was still a rigid body rotation around (0,0,0) because there are no lateral supports at the rotational support. Fixing the horizontal translation in all supports also did not have the right effect. The support reactions were still not symmetric with this setup. This was thus not the issue.

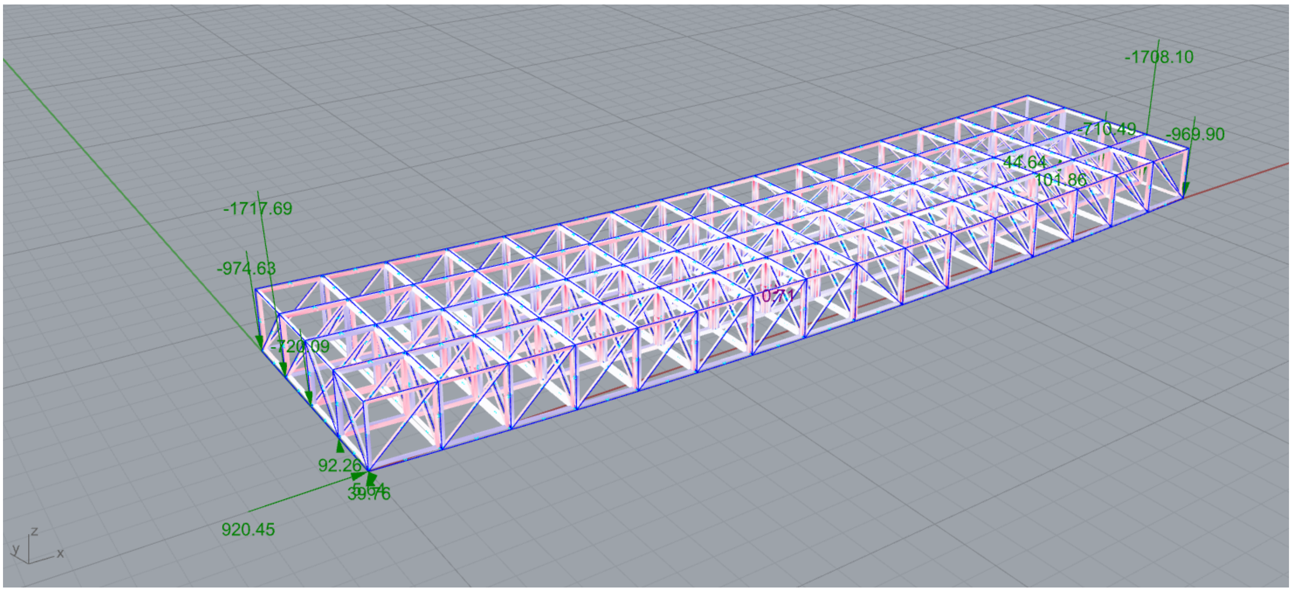


Figure 7.1: Model with translational stiffness of the splices turned on

Turning off the translational stiffness, resulting in completely fixed translation resulted in a symmetric response of the structure with support reactions that were more consistent with expected magnitudes. See also Figure 7.2. The results shown in this figure are before the iteration of the stiffness. The red colors indicate compressional stresses and the blue colors tensional stresses.

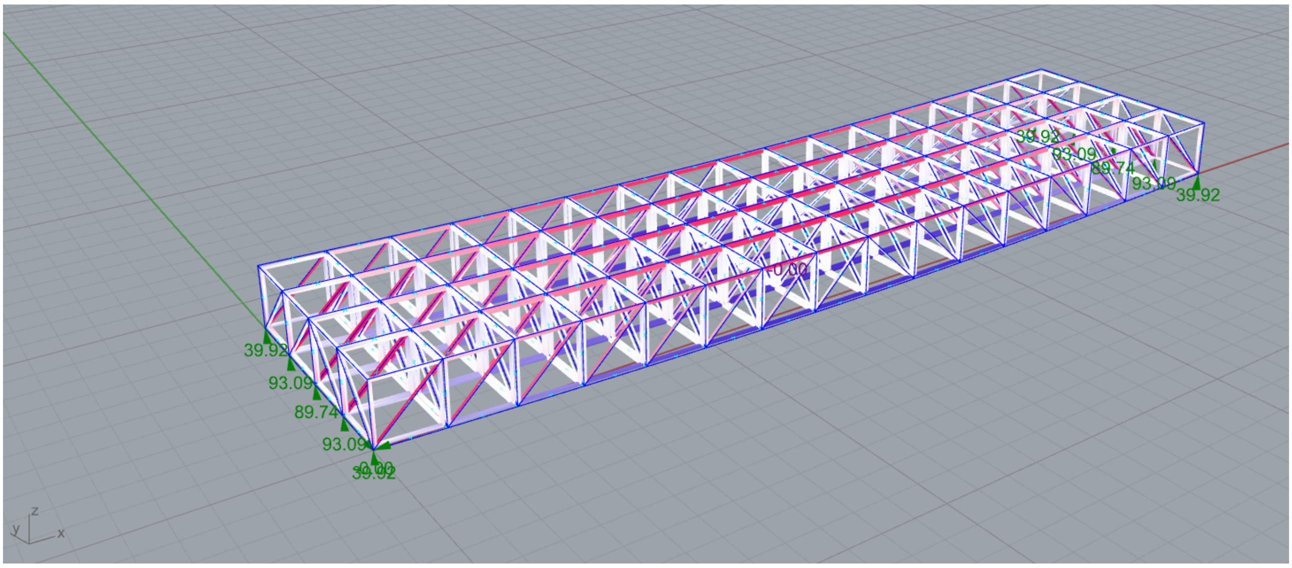
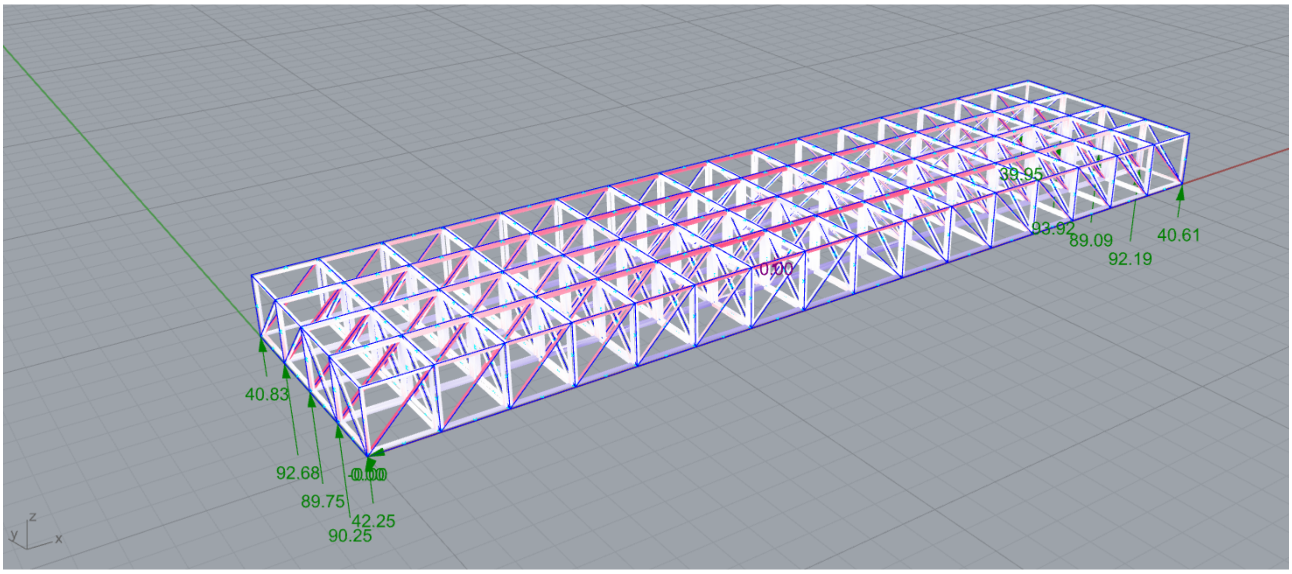
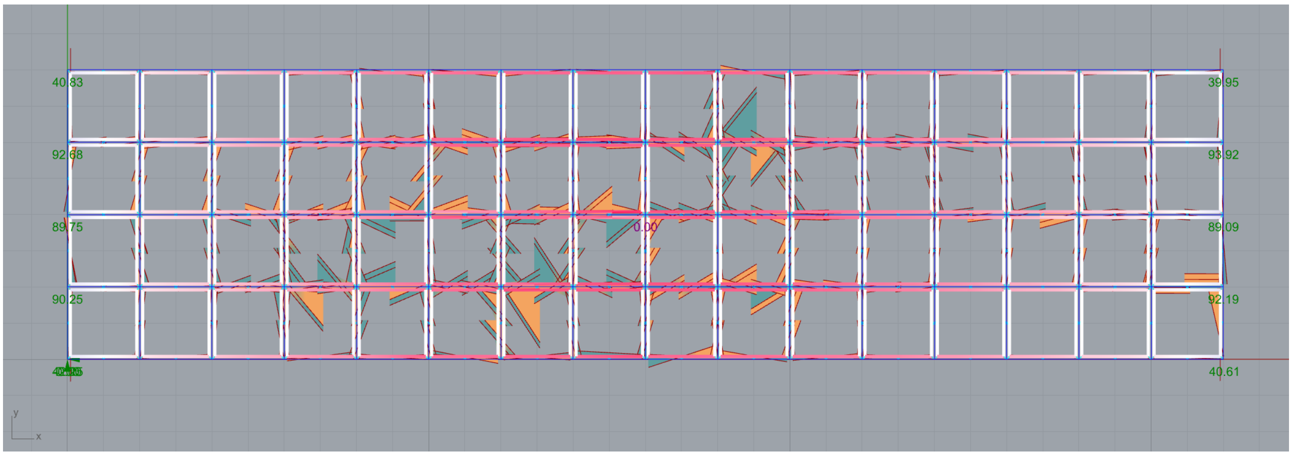


Figure 7.2: Model with translational stiffness of the splices turned off

To see if the structure still responds correctly after iteration of the stiffness, the loop described in Section 4.8 is added. This loop updates the stiffness of all joints individually depending on the |M|/N ratio from the previous run. As can be seen from Figure 7.3 the results are again asymmetrical but there is now only a small deviation in the size of the support reactions. There is a change in moment M_z and M_y present in the structure that causes this deviation. This can clearly be seen from Figure 7.3b where the moment M_z is shown in top view. The structure before the iteration loop has symmetrically distributed normal forces and moments. Therefore, the stiffness iteration loop should also result in symmetrical stiffness values for all joints. This is not the case and it is caused by numerical instability. Some joints therefore do not receive the same stiffness value as the equivalent joints in other locations in the structure. Therefore, redistribution of the forces and moments takes place and this causes these extra moments.



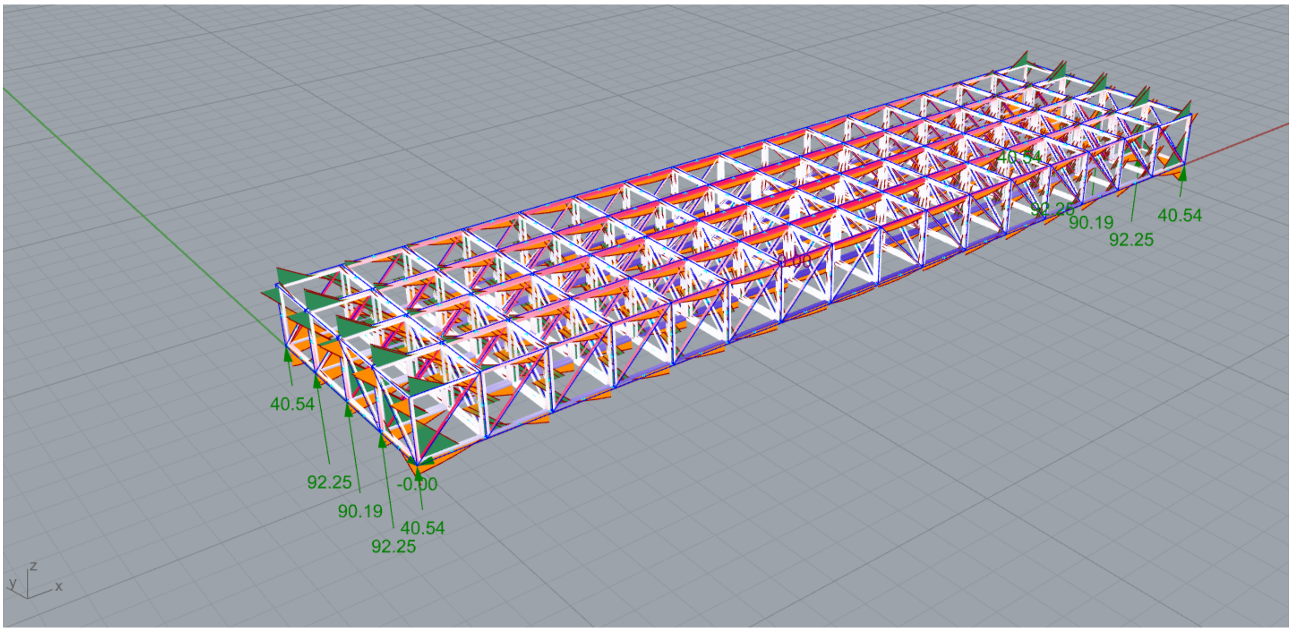
(a) Model with stiffness iteration loop for individual joints: 3D view



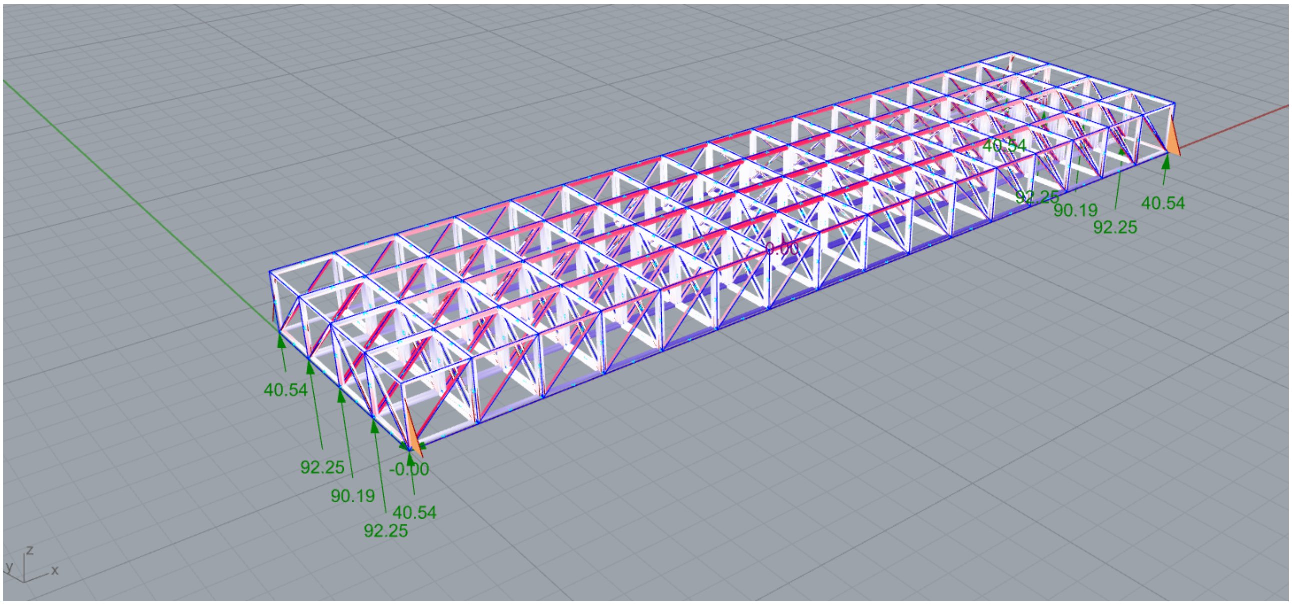
(b) Model with stiffness iteration loop for individual joints: moment M_z

Figure 7.3: Model with stiffness iteration loop for individual joints

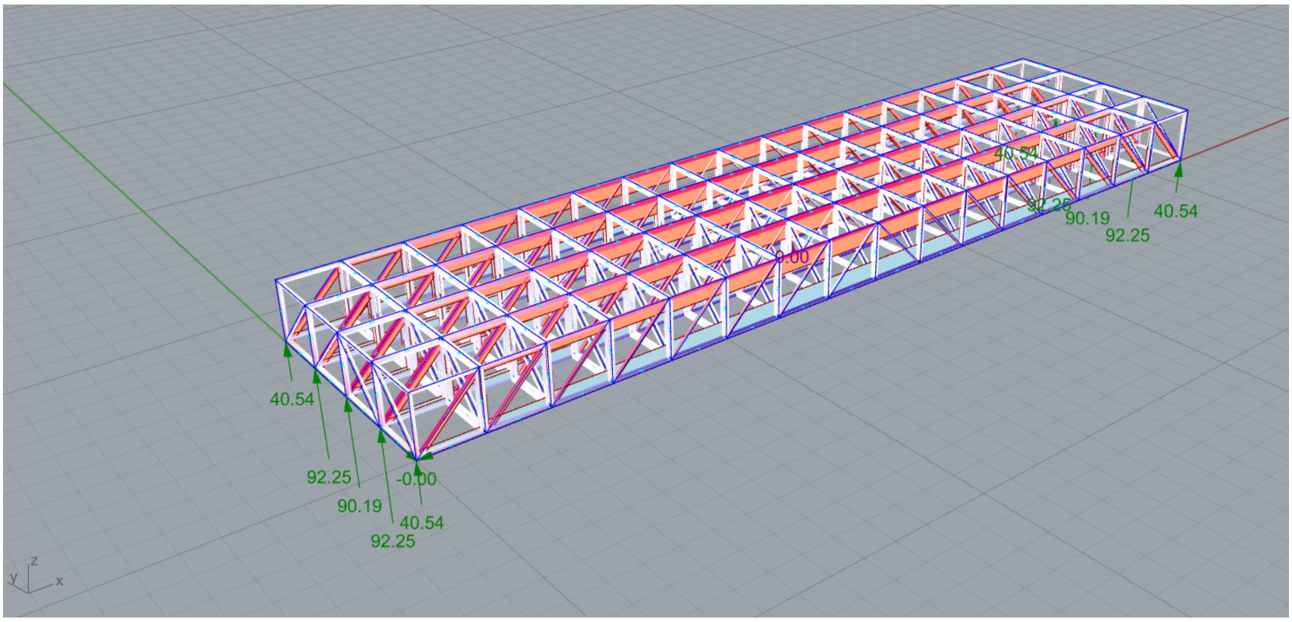
The model is sensitive to small differences in the stiffness values. This then leads to asymmetries and imperfections, due to the numerical errors. For this reason, the stiffness is updated per joint group. The individual stiffness values for the joints are calculated as normal. After that, the mean value is taken and assigned to all of the joints in the same joint group. This gives symmetrical results, which can be seen in Figure 7.4. The problem with irregular moments M_z in the model is completely gone with this approach.



(a) Model with stiffness iteration loop for joint groups: moment M_y



(b) Model with stiffness iteration loop for joint groups: moment M_z



(c) Model with stiffness iteration loop for joint groups: normal force N

Figure 7.4: Model with stiffness iteration loop for joint groups

The results of the different joint setups can be seen in Table 7.1. The deflections are all below the maximum allowed deflection of 16 cm. The extreme stress levels except for the first model as well. A problem with this model was that the truss could still rotate around the support at coordinates (0,0,0) despite the restriction of the rotation around the z-axis in the middle of the structure. The reason is that there is no lateral or vertical restriction on that same position. Therefore, the first buckling mode as shown by Karamba3D was a rotation of the entire structure around the corner support with lateral support conditions. This was not a real first buckling mode of the structure but a rigid body rotation. The BLF was in the order of 0.1 which was for these cases unrealistic. For this reason, the value for the BLF of the second buckling mode was selected which is close to the theoretical first buckling mode. The second buckling mode and higher were actual buckling modes and not rigid body rotations that showed global buckling with sinusoidal shapes. The values of all BLFs for the different models are above a value of 1.0, making them fulfill the constraint. For the first two models there is no stiffness iteration as mentioned before. For the last three models, the translational stiffness of the splices is fixed resulting in rigid translational stiffness.

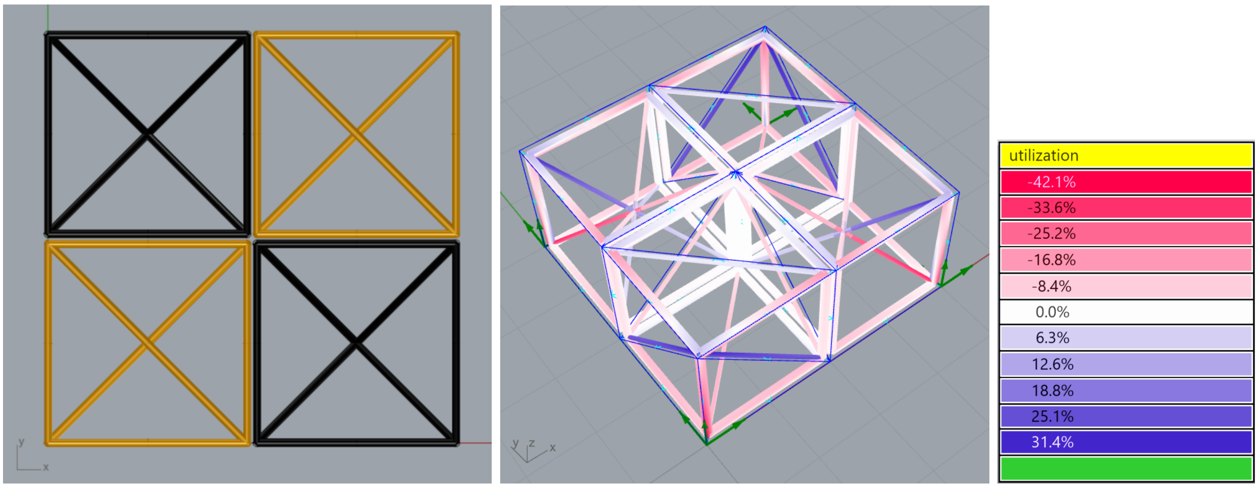
Table 7.1: Results of the truss model under different joint definitions

Model	Deflection [cm]	Extreme utilisation [%]	BLF [-]
Model with translational stiffness of the splices turned on and no stiffness iteration	11.5	294.4	1.1
Model with translational stiffness of the splices turned off and no stiffness iteration	8.9	50.5	2.4
Model with stiffness iteration loop for individual joints	13.1	95.3	3.8
Model with stiffness iteration loop for joint groups	12.0	49.2	3.9

7.2 Verification of the the Genetic Algorithm

To see if the model works as it was designed to, it is tested on a small scale model, this time also including the GA for topology optimisation. A grid of 2 m x 2 m is used and again the same load as for the calculation of the cross-section in Section 3.4. The modules each have a size of 2 x 2 x 2 m. This results in a space frame of 4 x 4 m with a 2 m depth. The cross-sections used for this module size are taken from Table 3.6. Only a vertical load is present in this situation consisting of selfweight, and permanent load. Since the structure is small, the size of the load is larger than the loads used in larger structures to make the stresses and deflections more significant. A dead load of 10 kN/m^2 is used besides self-weight. Furthermore, a partial load factor of 1.35 is added. The scaling parameter for buckling μ_3 is taken as 0.6 here. Earlier simulations showed that starting with an initial value of 1.0 resulted in a greater emphasis on the BLF compared to other barrier functions. This is unwanted and therefore this value is lowered. The values for μ_1 and μ_2 are still 1.0. The structure is supported at the four corners with point supports. Each of these supports restricts vertical translation and also both horizontal translations. Moments are not restricted. The module types 1 to 6 are used for this simulation. This results in $34^4 = 10^6$ possibilities for different configurations. It is not possible to find the best solution by brute force because each solution takes 2 seconds at minimum. This would mean that with this design space this would at least take a month.

The topology of the structure after 30 generations can be seen in Figure 7.5. There are two modules of type 3 and two of type 1 present in the structure. This can be seen from the bitmap of this structure in Figure 7.5a that can be compared to the catalog of Figure 5.7. They do not form an entirely symmetric shape but the stresses seem to be well distributed to the corners of the structure through mostly diagonals in tension. However, with module type 3 there are also a couple of diagonals in compression which could lead to higher risk of buckling. A single module structure has the potential to be more efficient since the distribution of forces is more evenly spread, leading to less high local stress in the members, a lower BLF, and therefore a better fitness value. Besides, it is known from Table 4.2 that module type 1 is one of the heavier module types. Module types 2, 3, 5 and 6 are all more lightweight. The optimisation is about minimising weight and for this structure and it does not find the most lightweight solution. This would be four modules of module type 5. This module type has worse structural performance than the other module types. Because of the logarithmic barrier functions, this would unlikely be the final chosen module on each location. Therefore, it is assumed that the model has not converged to the most optimal solution.



(a) Verification model: Bitmap

(b) Verification model: 3D view

(c) Verification model: Utilisation of the elements

Figure 7.5: Verification model structure

The structure's performance can be measured based on a performance metric table. This is a way of checking the structural performance and the feasibility of the structure. In Table 7.2 the performance metric for this small structure can be seen. It can be used to compare this solution with others found by the algorithm. For this reason, only the variables that directly influence the fitness function are displayed in this table. By comparing these different things a final decision on the design can be formulated.

Table 7.2: Performance metrics of the verification model

Performance metric	Value	
Weight	2323.2	[kg]
BLF	24.7	[-]
Displacement	3.58	[mm]
Extreme utilisation	50.4	[%]
Fitness	-0.6	[-]

7.3 Observations

What already directly can be seen from the model with the 2 x 2 grid is that the most critical positions in every simulation are near the supports of the structure. The largest stresses occur in the elements that are close to these locations. Buckling is most likely to happen here. The reason for this is that the number of parallel elements here is minimal and support reactions are some of the largest forces in the structure. At other intersections, multiple modules have their columns close together, which results in larger combined cross-sections and more resistance to the stresses. They can be more easily distributed. Local buckling is an important consideration for the space frame design. Failure of one of the axially compressed members may cause others to fail as well. This is because these receive extra loading that was formerly transmitted by the already failed member [2]. However, for this small model, local buckling is not critical.

The model converges to a final solution very well. The Galapagos software shows the progression of the optimisation process with a convergence graph. The convergence graph for this simulation is visualised in Figure 7.6. The values on the horizontal axis are the generations 0 to n. The vertical axis shows the variety of fitness values obtained in every generation. This variety for every generation is shown in yellow. The orange area is the standard deviation around the mean and the mean is shown with the thick red line in the middle. Every new best solution is shown in a white box with a '+' sign, so there are three of those in this simulation. As can be observed from the figure the number of generations calculated by the model reached 30 and not 50. The reason behind this is that the amount of data stored by the model became very large. Grasshopper eventually used up all available Random Access Memory (RAM) which was 32 GB. The main reason for the usage of this large amount of data was the stiffness iteration loop and the second-order analysis in combination with the eccentricities in the model.

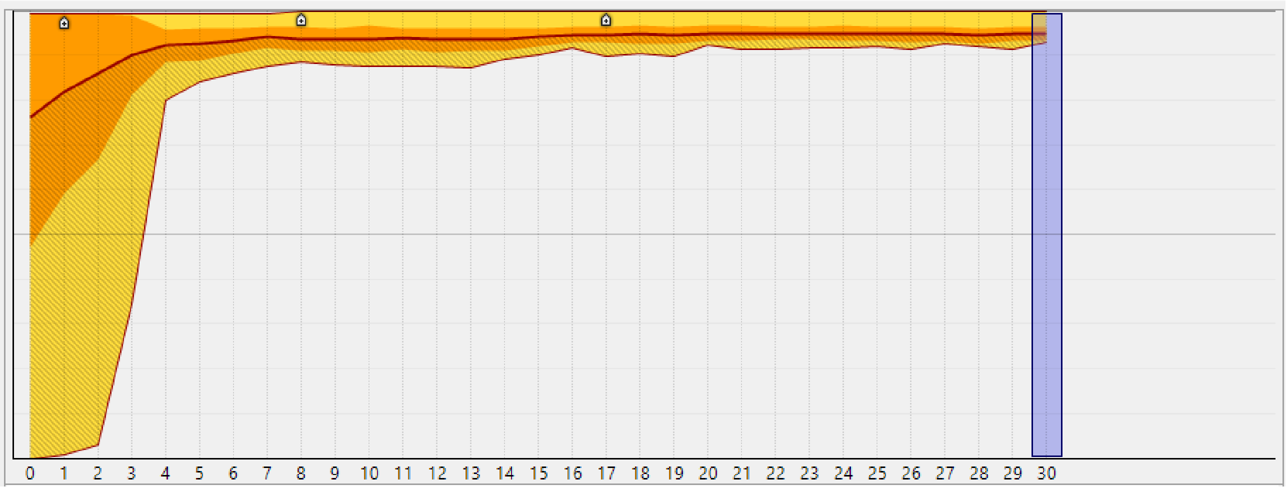
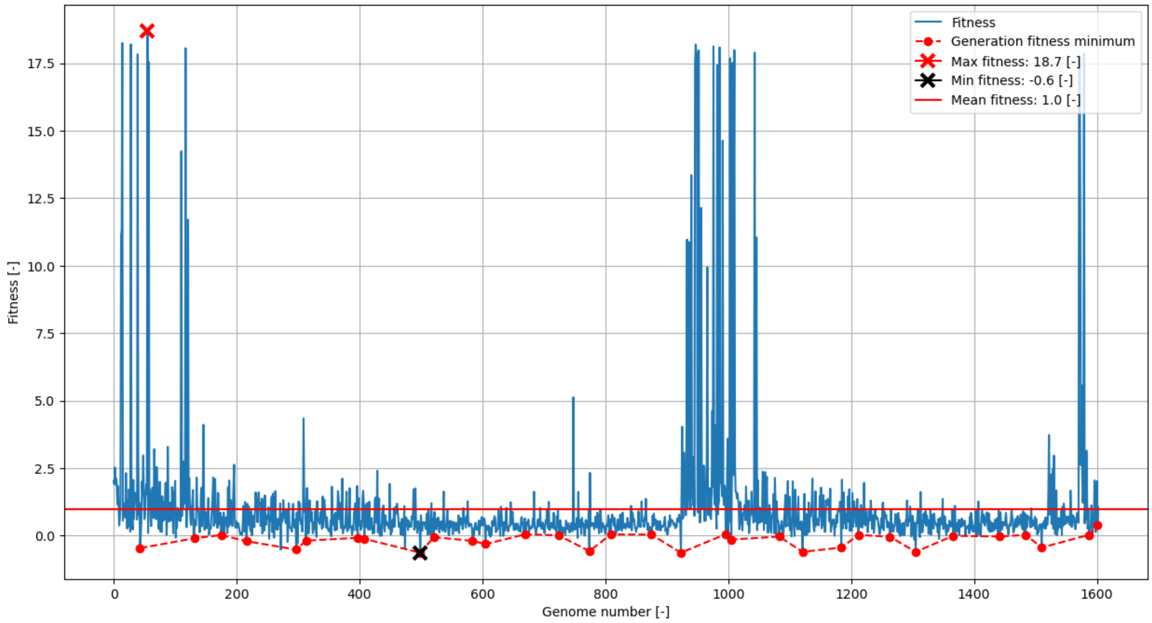


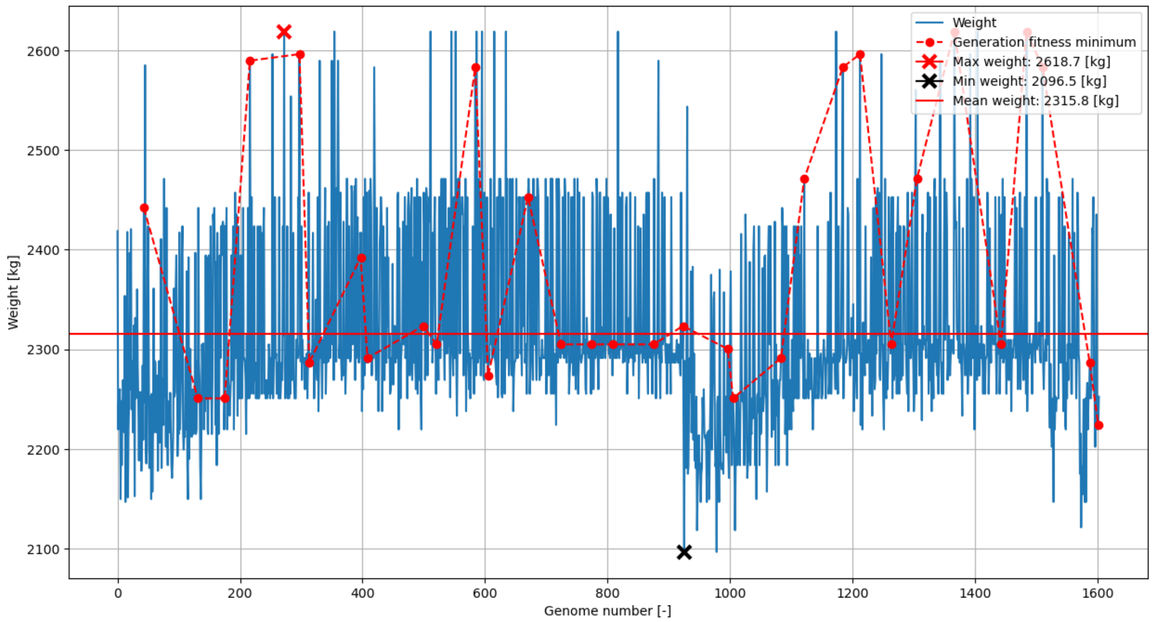
Figure 7.6: Conversion graph Galapagos for the verification model. The horizontal axis shows the generation number, the vertical axis the fitness value. Large values for the fitness are close to the origin

The change in Fitness, weight, BLF, displacement and extreme utilisation per genome number are also observed. The extreme utilisation is the highest recorded utilisation in any of the members present in the model. These variables can be seen in Figure 7.7. Minimum, maximum, and mean values are also shown. In every figure, the generational best fit is plotted. This is the genome that performs the best for the fitness function. In the other figures for for instance the weight it indicates which weight corresponds to the best generational solution.

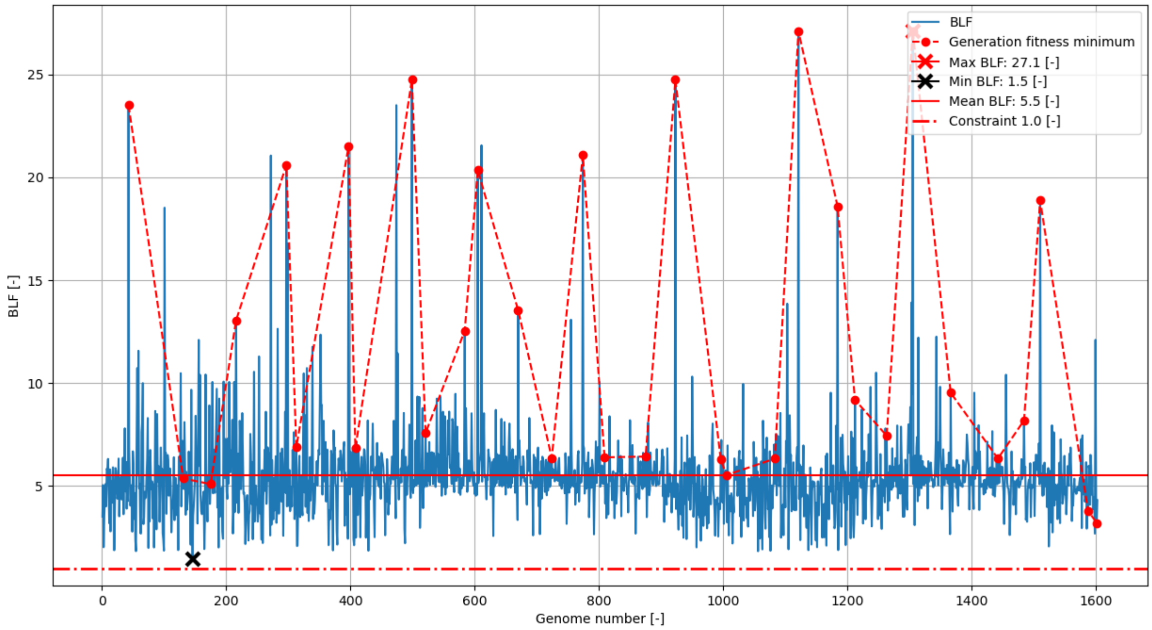
There is a slightly downward trend in the fitness graph. This shows that the model is slowly converging towards a final solution. Looking at the weight, this slightly increases from the starting value and seems to flatten out except for between the genome numbers 900 and 1100. There is a small dip in weight that gives the solutions a better fitness value. It can be seen however from Figure 7.7d, that the displacement of the structure is also larger which negatively influences the fitness function. The generational best value for the weight is also very different every time. This seems to be a bit different for the BLF. When a solution has a high BLF it is more fit in general. There is no clear trend observed for the BLF. There is a slightly downward trend observed in the displacement per genome number as well as for the maximum utilisation of the elements. The small differences have to do with the size of the structure. For a small grid of only 2 x 2, there are no large differences in weight and other factors such as BLF for different configurations. Therefore, the weight and BLF are very constant during the simulation. There is expected to be a larger spread in these variables when larger structures are modeled.



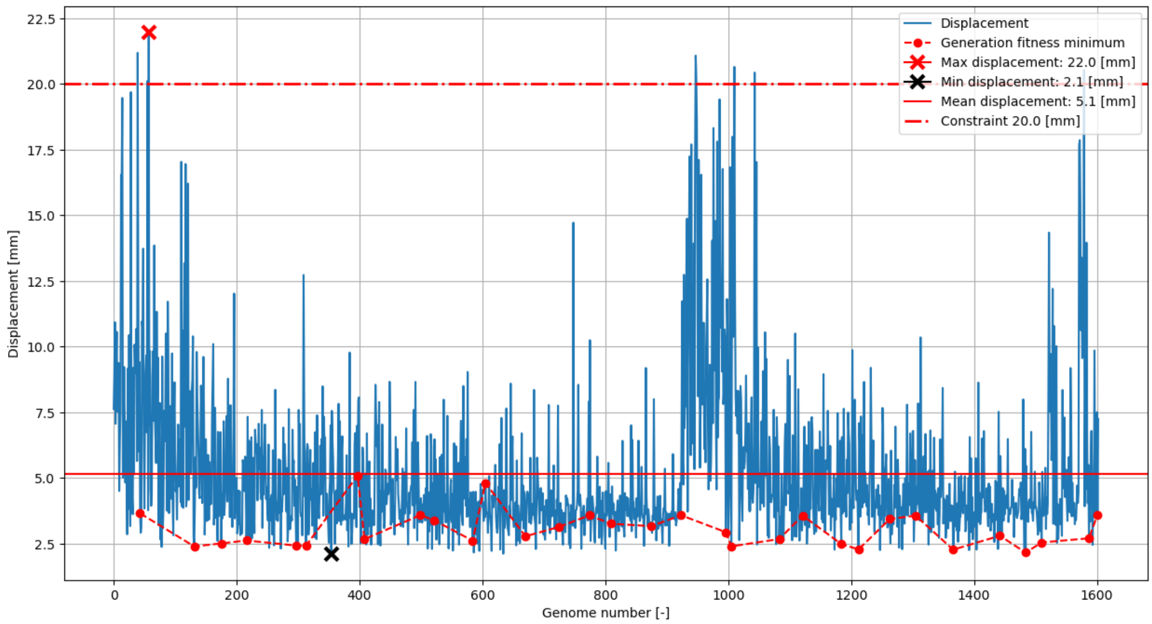
(a) Fitness per genome number



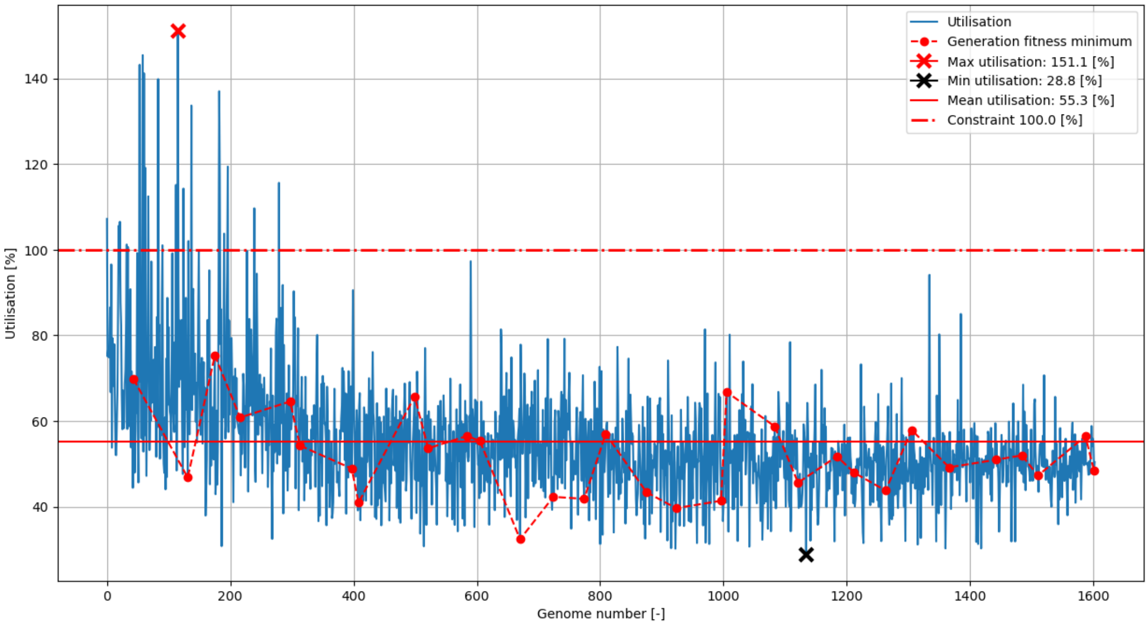
(b) Weight per genome number



(c) BLF per genome number



(d) Displacement per genome number



(e) Extreme utilisation in the members per genome number

Figure 7.7: Fitness, weight, BLF, displacement and extreme utilisation per genome number

The number of stiffness iterations needed per simulation can be seen in Figure 7.8. It becomes clear from this plot that a total number of 10 iterations is never reached. The maximum number of iterations was 7. The largest share of the solutions only needs 1 iteration. This is for 57.6% of the cases. It always happens that the loop is entered at the start because the initial stiffness is taken from Table 3.5 and therefore much different from the values of the stiffness analysis. The change in stiffness values is at the start always larger than 1.0%. It could be argued that the first alteration of the stiffness is already enough and entering the loop is not needed since so many genomes only use the loop once. The initially updated stiffness for these genomes is already good. In 42.4% of the solutions, a second iteration is needed. In 5.1% a third iteration is needed. In only 0.31% of the genomes, the number of iterations is equal to 5 or larger. In the model, all iterations are computed up to the threshold. Subsequently, the solution that satisfies the condition specified in Equation (4.2) is selected. By lowering the maximum number of iterations, less stiffness iterations will be performed. This will make the model faster. A short simulation of 635 different solutions had a mean time for the loop of 3.7 s. Without loop only $\frac{1}{10}$ of this time is needed so 0.4 s resulting in a time saving per solution of 3.3 s which is an 89.2% reduction. This is the time saving for a small model per solution. For a larger model, this improvement might even be larger since the analysis of large structures takes longer for the Karmamba3D components used in the model. Since the loop is not needed in most of the cases it is omitted in further calculations. It will also require less RAM from the computer. This will minimise the risk that the memory of the device will be full and Grasshopper stalls.

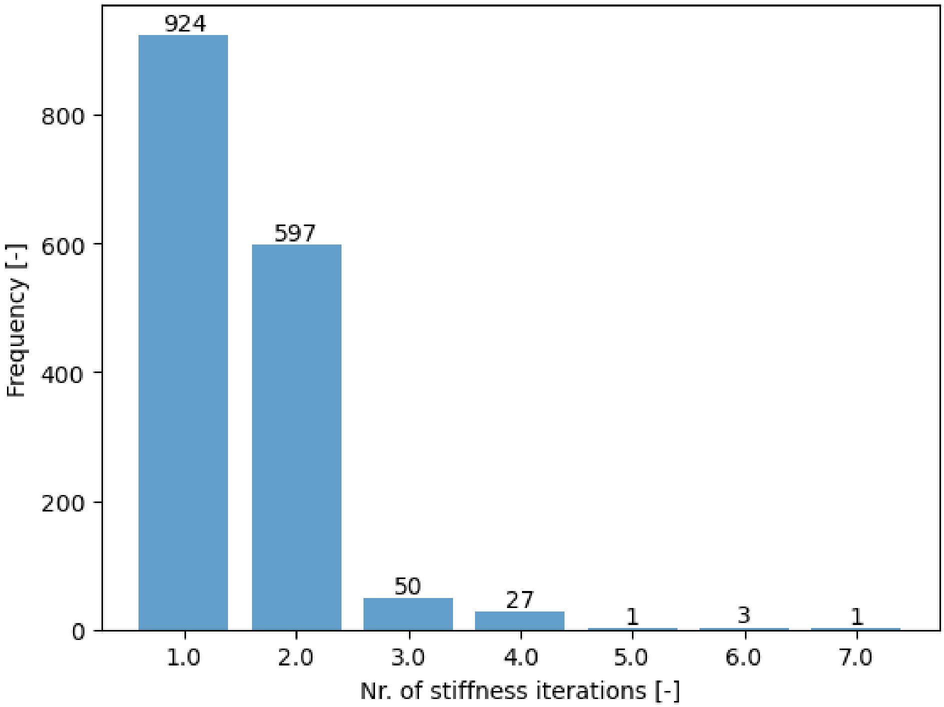
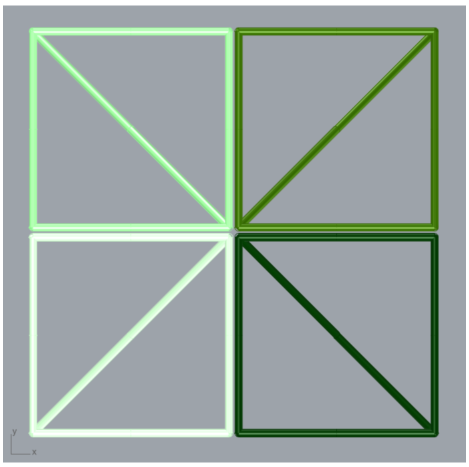


Figure 7.8: Barplot of number of needed stiffness iterations for every genome

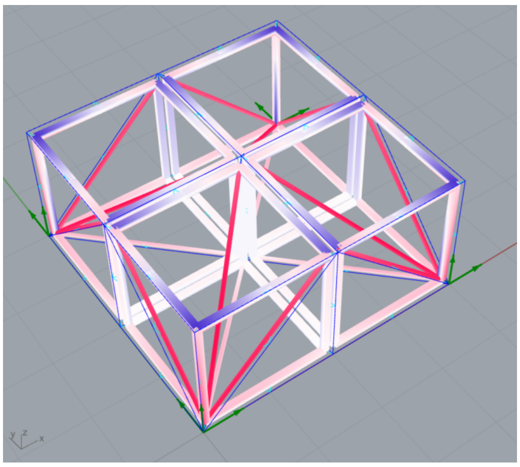
7.4 Simplified verification model in comparison with optimised structure

The goal of the verification is to determine if the model can select the optimal design. It is not possible to do this when the model is not able to complete due to memory storage problems. For this reason in the next simulation, the iterative stiffness loop is omitted. The data stored with this loop is one of the main reasons that the RAM used by Grasshopper becomes large besides the second-order analysis used for determining the BLF.

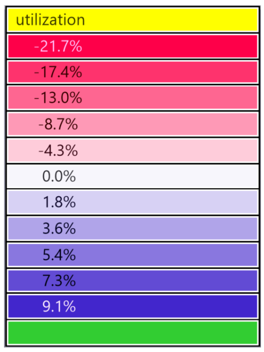
First, using intuition a couple of standard symmetric configurations are made. These structures are the most, or close to the optimal design for this small structure with the existing modules. There are six different configurations. Each of these has different values for the weight, BLF, displacement, and maximum utilisation. Their configurations are shown in Figure 7.9. Each module has a color code. The various color codes from Figure 5.7, which displays the catalog of modules with their unique orientations, are also provided. This catalog allows for quick identification of the module type and its orientation.



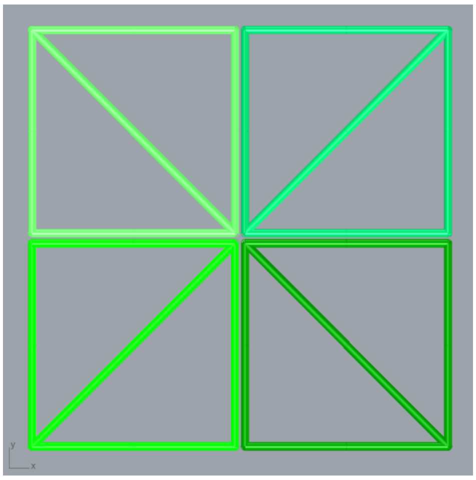
(a) Configuration 1: Bitmap



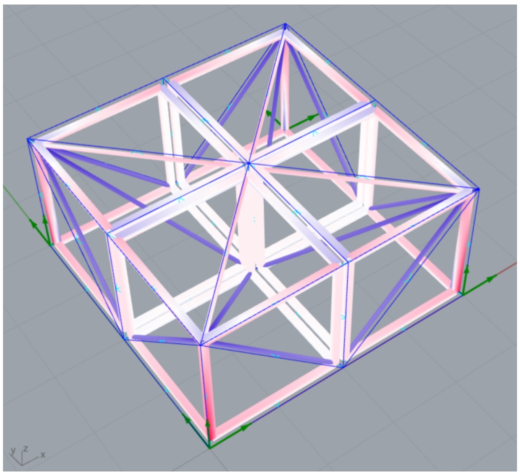
(b) Configuration 1: 3D view



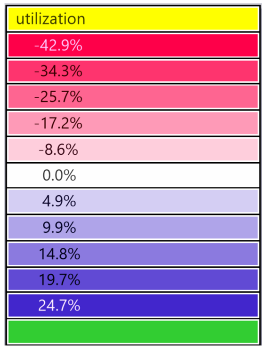
(c) Configuration 1: Utilisation of the elements



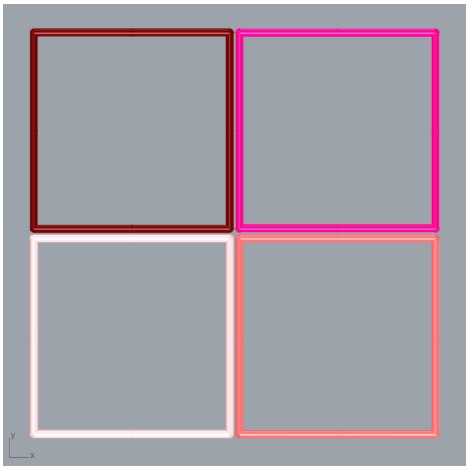
(**d**) Configuration 2: Bitmap



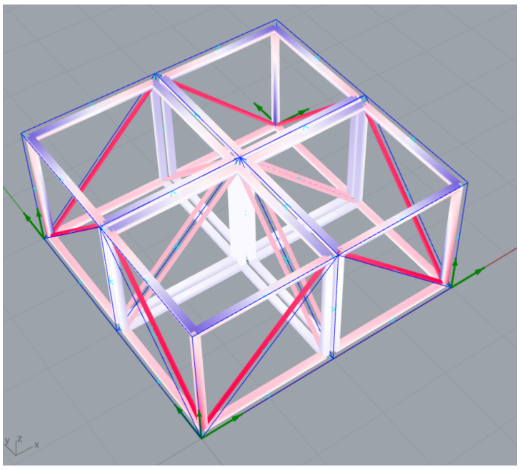
(**e**) Configuration 2: 3D view



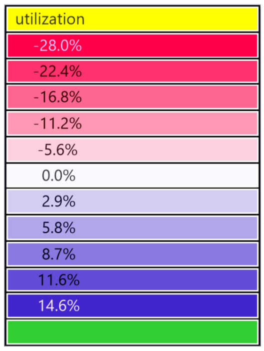
(**f**) Configuration 2: Utilisation of the elements



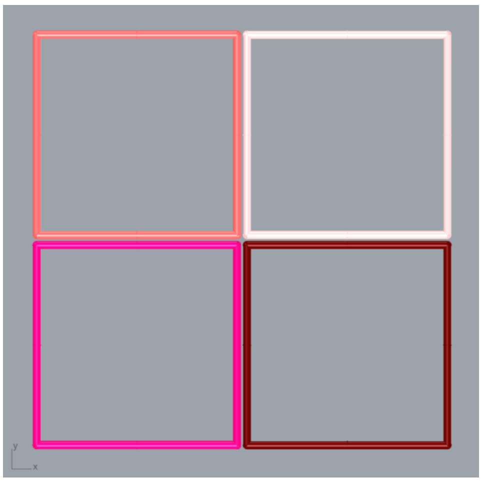
(**g**) Configuration 3: Bitmap



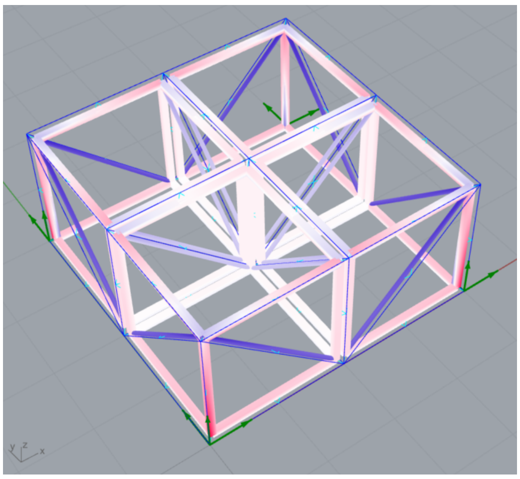
(**h**) Configuration 3: 3D view



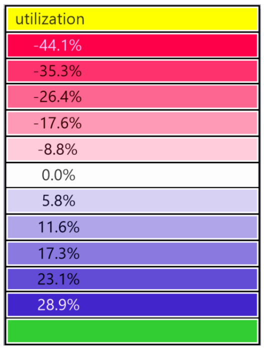
(**i**) Configuration 3: Utilisation of the elements



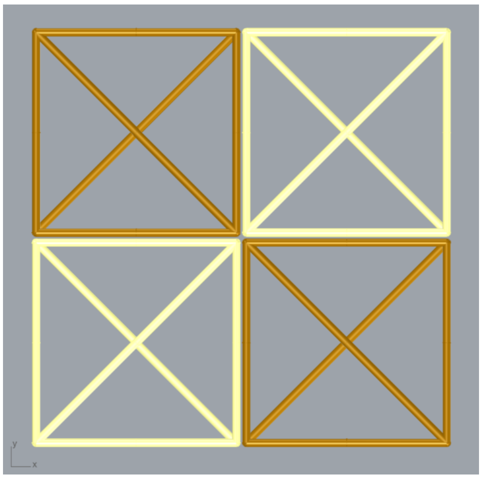
(**j**) Configuration 4: Bitmap



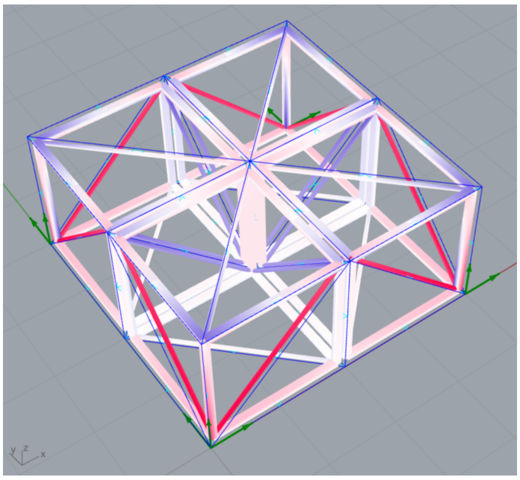
(**k**) Configuration 4: 3D view



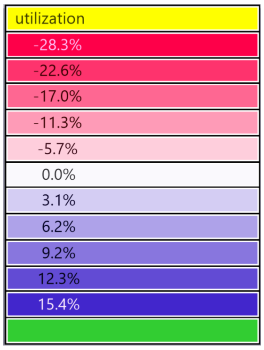
(**l**) Configuration 4: Utilisation of the elements



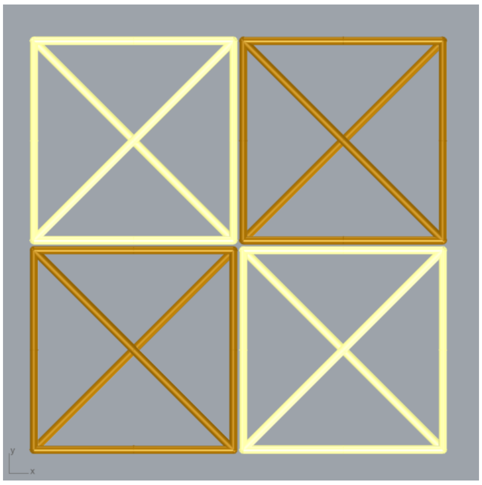
(m) Configuration 5: Bitmap



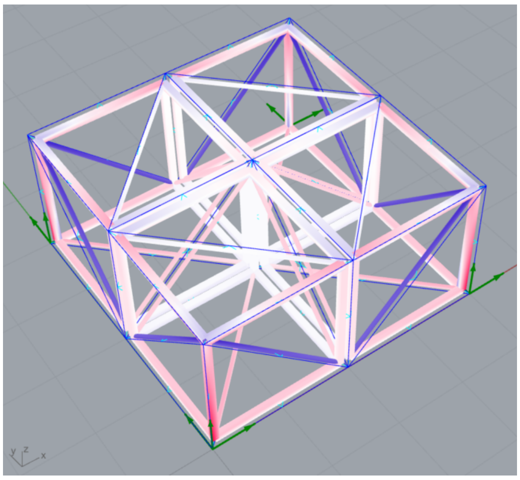
(n) Configuration 5: 3D view



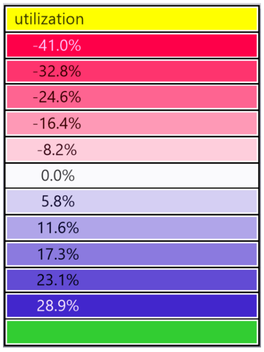
(o) Configuration 5: Utilisation of the elements



(p) Configuration 6: Bitmap



(q) Configuration 6: 3D view



(r) Configuration 6: Utilisation of the elements

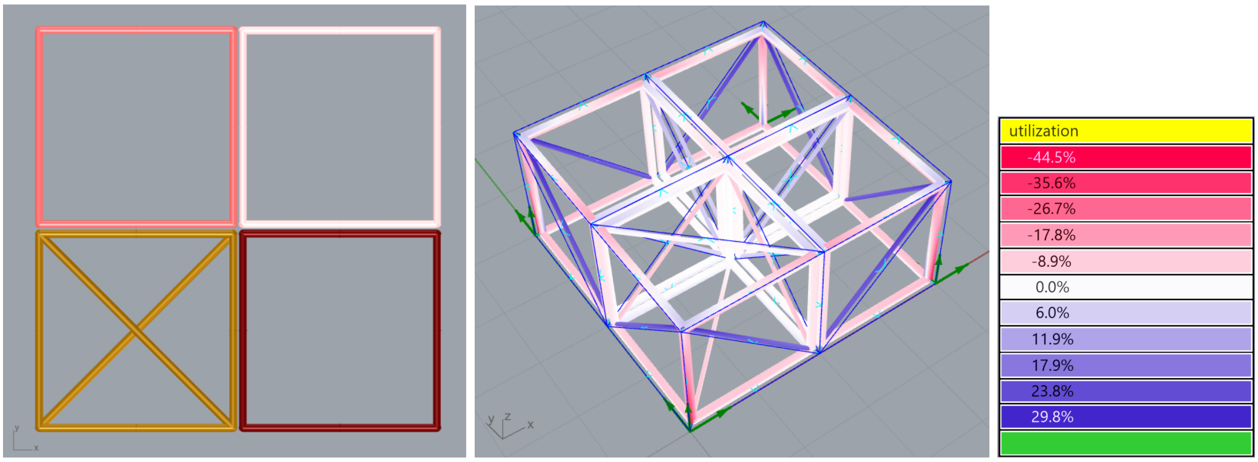
Figure 7.9: Different intuitively configured structures

Configurations 1 and 2 consist of modules from type 2, configurations 3 and 4 of modules from type 6, and configurations 5 and 6 from modules of type 1. Table 7.3 shows the performance of the different configurations for various metrics. Configuration 4 has the best fitness score. The reason is that it is a lightweight structure, has the largest BLF of all structures, and has the lowest deflection. The structure has an analogy with a flat Pratt truss. The diagonals that transfer forces are all loaded in tension. These elements therefore can not buckle. The columns and beams are loaded in compression but these have a larger cross-section than the diagonals and are therefore less prone to buckling. This results in the highest BLF for this structure. For all other configurations buckling is a bigger issue because some diagonals are loaded in compression.

Table 7.3: Comparison of symmetric configurations; The best performing configurations per metric are highlighted in green

Performance metric	Conf. 1	Conf. 2	Conf. 3	Conf. 4	Conf. 5	Conf. 6
Weight [kg]	2214.8	2214.8	2203.4	2203.4	2358.1	2358.1
BLF [-]	6.5	22.9	7.3	32.0	7.3	23.1
Displacement [mm]	2.0	3.1	2.1	3.5	2.2	3.4
Extreme Utilisation [%]	23.1	51.1	32.4	52.5	32.7	48.9
Fitness [-]	-0.5	-0.8	-0.6	-1.1	0.4	0.1

A new simulation is performed with only module types 1, 2, and 6. The reason is that these module types are used to create the configurations shown in Figure 7.9. This results in a total number of $6 + 8 + 12 = 26$ different options per location. There are thus $26^4 = 10^5$ different configurations possible. As mentioned before, the iterative loop will be taken out of the model. The initial stiffness will only be updated once during the calculation of a solution. It is expected that the results will be slightly different for the configurations in Figure 7.9 but that these differences are minor. The final result can be seen in Figure 7.10. It is a symmetrical model which looks similar to configuration 4 in Figure 7.9k and Figure 7.9j. The only difference with this structure is that configuration 4 has a symmetrical layout with only module type 6. This structure is found already at the twenty-fourth iteration. Unexpectedly, configuration 4 was not found eventually. This genome with high would have been taken to the next generation and adjusted for the other 26 generations that were still to come. The reason is that the number of possible configurations with a random change is still large. This is due to the 26 options for a single location. Therefore, even if the solution is close to an almost perfect one, it still takes numerous iterations for the GA to find this perfect solution eventually.



(a) Verification model without stiffness loop: Bitmap

(b) Verification model without stiffness loop: 3D view

(c) Verification model without stiffness loop: Utilisation of the elements

Figure 7.10: Verification model without stiffness loop

The solution of the model after 50 iterations is not a configuration that was expected from the start since it is asymmetrical. However, it scores very well. This is mainly due to the large buckling load factor. The score for the fitness value is almost as good as that of configuration 4 because it is almost similar. This can be seen in Table 7.4. The score is lower because the weight is larger which is the most important factor because the minimisation is about the weight. Besides, the BLF is slightly lower. These factors have slightly more influence than the small difference in displacement and maximum utilisation. As can also be seen from Table 7.4 the model with loop, which can be seen in Figure 7.5 has a worse score because it is heavier and especially the BLF is much lower. This simulation without a loop shows that it can converge to a good solution, however, it could take even more iterations to reach an optimal one. Initiating a second simulation with the best results from the last one is advised to reach an even better solution.

Table 7.4: Comparison of model output and intuitive best configurations

Performance metric	Conf. 4	Model without loop	Model with loop
Weight [kg]	2203.4	2251.2	2323.2
BLF [-]	32.0	31.9	24.7
Displacement [mm]	3.5	3.6	3.6
Extreme Utilisation [%]	52.5	53.0	50.4
Fitness [-]	-1.1	-0.8	-0.6

The conversion graph is visualised in Figure 7.6. It shows the fitness on the y-axis and the generation number on the x-axis. The orange region is the standard deviation from the average fitness. The yellow region shows the spread in the generation's weakest and strongest individual solutions. The graph shows convergence from the beginning since the spread in fitness values becomes smaller and the solutions improve. The simulation finds an optimal solution two times (seen from the "+" signs in the graph). After the twenty-fourth generation, no better solutions are found. This figure has been made with Python using the fitness value data from Grasshopper. It is validated visually using the graph that is made with Galapagos. Validation of the graph is performed in Appendix G.

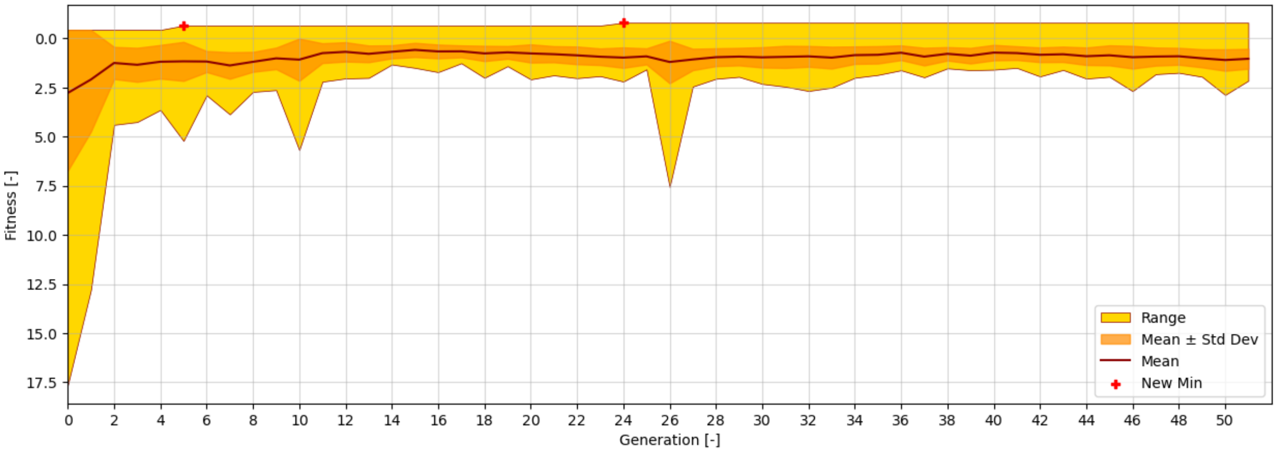


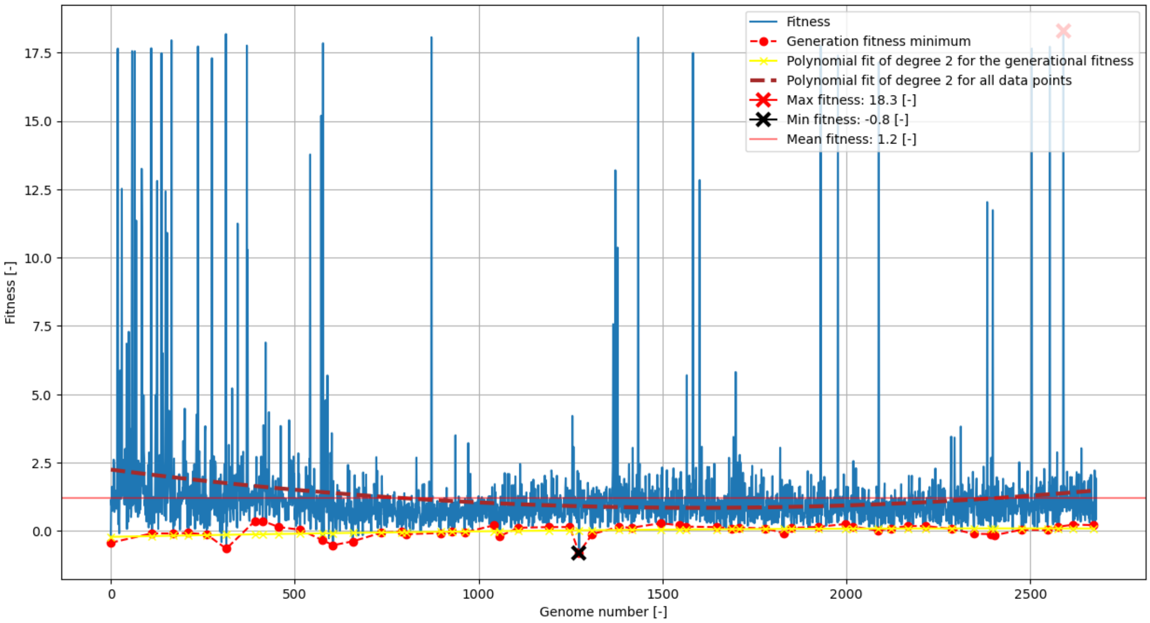
Figure 7.11: Conversion graph of the verification model without stiffness loop

In Figure 7.12 the fitness, weight, BLF, displacement, and extreme utilisation per genome number for this simulation are shown. A good convergence of fitness cannot be seen from the generational minimum values. A polynomial of degree 2 is fitted to all data and to the generational best data. This polynomial regression is done using the method of least squares. For the fitness, these graphs are almost flat and there is no clear trend. Most generational best values are around a value of 0.0 but this value fluctuates. The number of high fitness values seems to decline a little bit per genome number. At the end of Figure 7.12a a solution can be seen that comes close to the best solution found in the ninth iteration. If the simulation had continued better solutions could have been found from the genome of this one. However, as mentioned before, due to the large number of possible configurations this better solution was not found in this simulation.

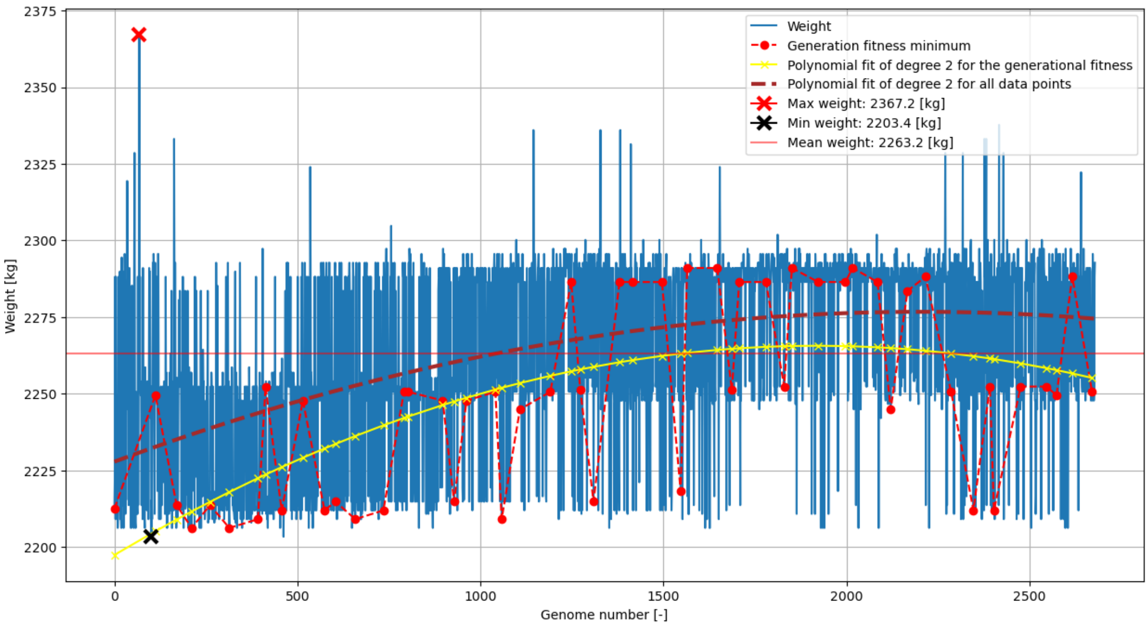
The spread in weight is not that large since the weight of the different module types used does not vary much and the structure is small. However, it can be seen that the weight of the structure at the start of the simulation is lower in the first 500 simulations than what is later observed. The minimum weight of the simulation is the same as found for configuration 4 which is logical. The overall performance of this solution was not very good and the algorithm prefers modules that are slightly heavier but perform better on other aspects such as extreme utilisation and weight.

From Figure 7.12c it can be seen that the value for the BLF is very constant per genome number. There is a peak in the graph that corresponds to the optimum found in the twenty-fourth generation. There are no other solutions found that come close to this BLF of 31.9. The other configurations have a mean value 4.8 and fluctuate around this value. Only one other value scores just above 20.0.

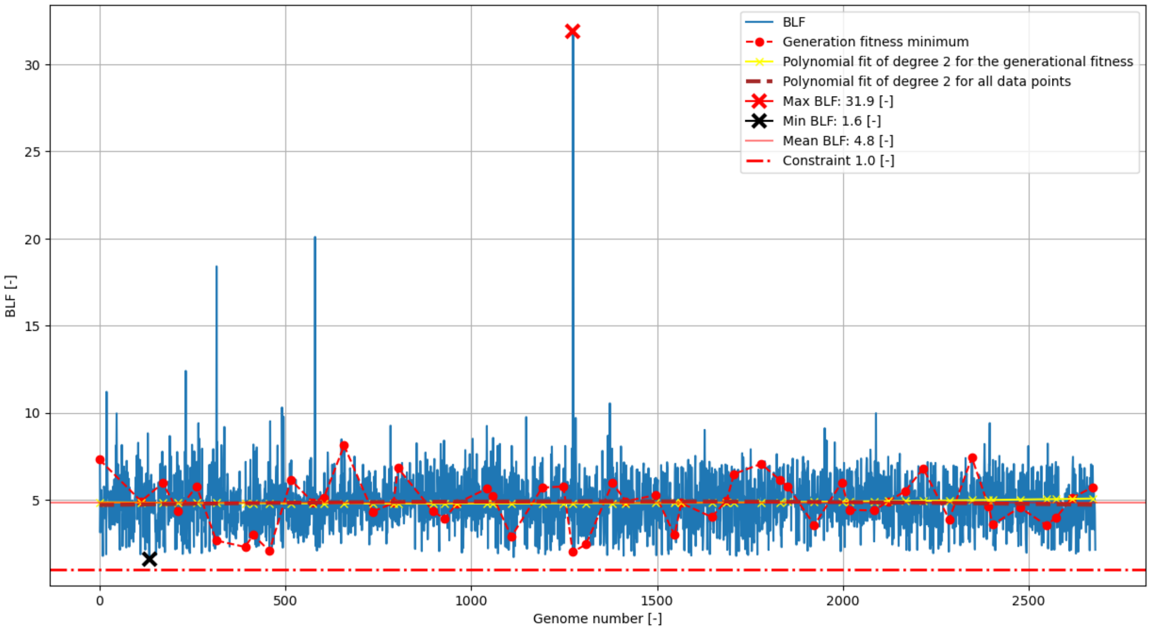
Displacements have a downward trend looking at all the different simulations. From the generational best solutions this is not observed. All of the generational best solutions however are close to the lowest displacements of that generation. They are all below the average displacement of all solutions as well. This is a good sign because this shows that the algorithm prefers solutions with low displacements. The same can be said for the best solutions looking at the extreme utilisation per genome number. All the generational best-fit solutions have a low value for extreme utilisation and only two are above the average. This is because these solutions are relatively at the beginning of the simulation where the extreme utilisations show larger values.



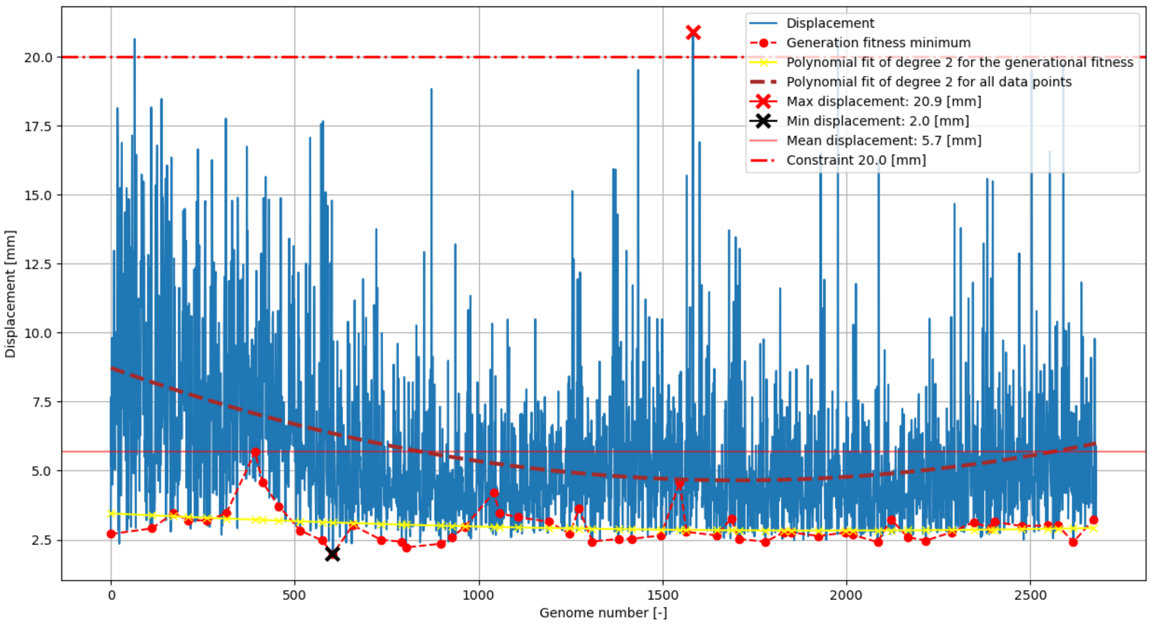
(a) Verification model without stiffness loop: Fitness per genome number



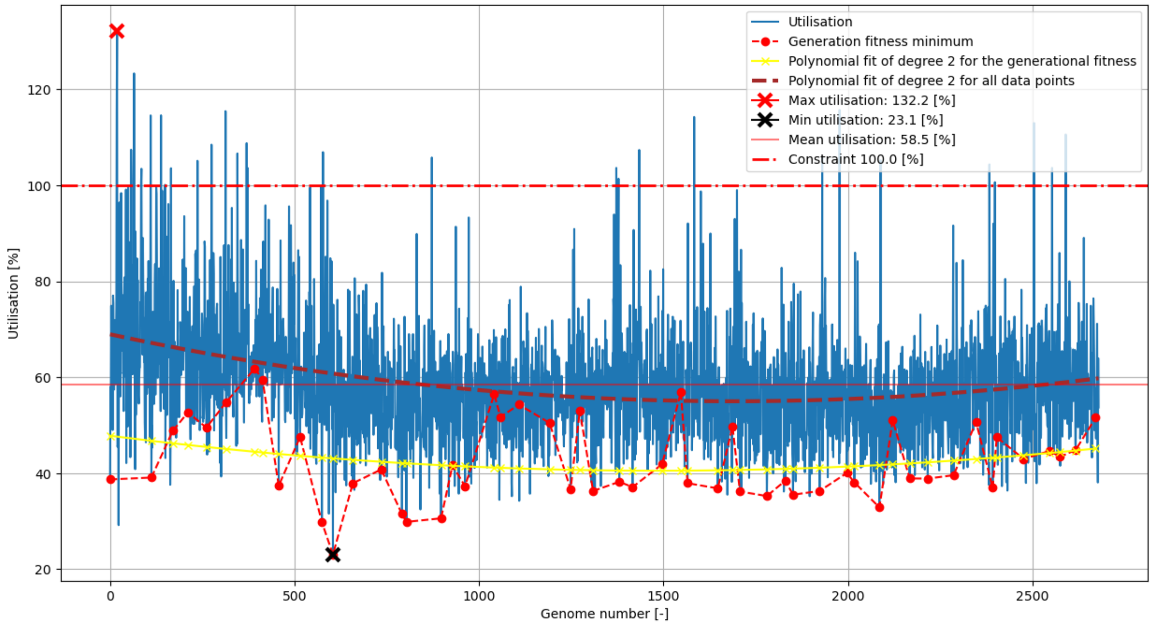
(b) Verification model without stiffness loop: Weight with joints per genome number



(c) Verification model without stiffness loop: BLF per genome number



(d) Verification model without stiffness loop: Displacement per genome number



(e) Verification model without stiffness loop: Extreme utilisation per genome number

Figure 7.12: Verification model without stiffness loop: Fitness, weight, BLF, displacement and extreme utilisation per genome number

Lastly, the percentage of axial deformation energy is checked. For normal trusses, this percentage is much larger than that of the bending energy. This percentage is also checked during the entire simulation. A histogram of the percentage of axial deformation energy can be seen in Figure 7.13. It clearly shows that in most simulations the percentage of axial deformation energy is much larger than for bending energy. In 98.7% of the cases, the axial deformation energy is larger than 50%. In most cases, it is around the mean value of 79.6%. It is concluded that the truss model shows deformation energy values normal for truss structures.

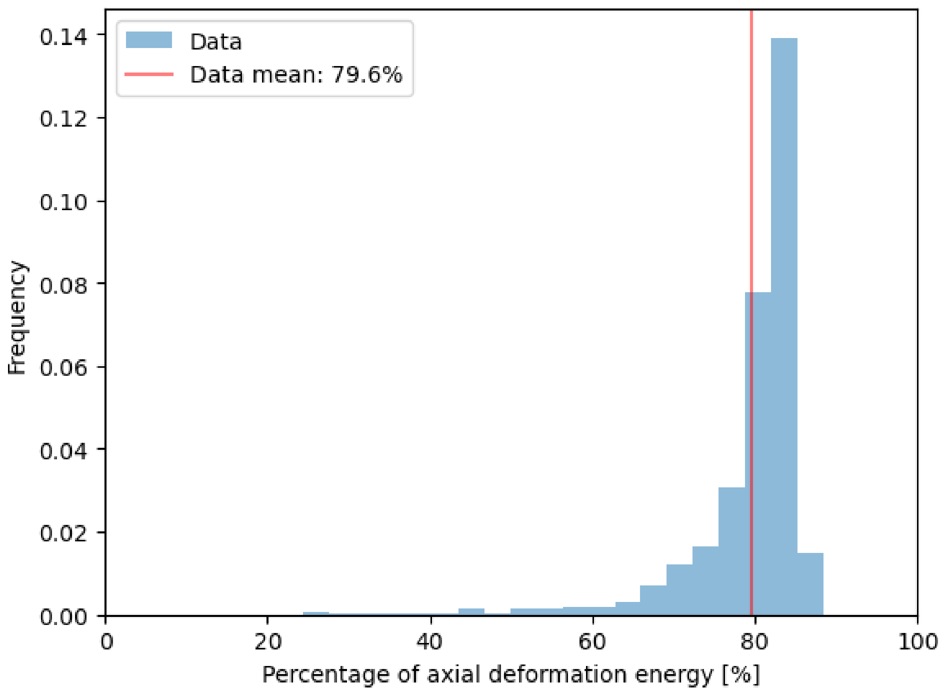


Figure 7.13: Histogram of axial deformation energy verification

7.5 Conclusion on the verification

From the verification of the joints, a couple of things become clear. The first one is that the splice connections caused a torsional moment in a regular space truss. This symmetric model suddenly had asymmetric internal forces. Because the translational deformation of the splice connections can be regarded as small, this stiffness was turned off in the model, meaning that it is fixed. This resulted in realistic symmetric internal forces. Furthermore, the model is sensitive to minor changes in the stiffness of the joints. The individual alteration of all joint stiffnesses leads to strange force and moment distributions and unrealistic results. For this reason, the stiffness iteration for joint groups was introduced. As a last step, the iterative loop for the joint stiffness was omitted because this saved computation time while the precision was just a little less than before. The simplifications in the model resulted in a more robust system while still providing an accurate representation of the joints' real behavior.

From the results regarding the small-scale simulations, it becomes clear that the model can find solutions that are close to the optimal one after a single run. Thus the verification process yielded positive results. Because there are many different configurations possible, even on a small scale, the optimum is not always reached. Therefore, it is advised to look carefully at the requirements of a structure and beforehand select a small number of modules to be used. This reduces the number of possible solutions. Furthermore, it is advised to do multiple simulations preferably with the best genomes of a former simulation to improve the result.

8 Model validation

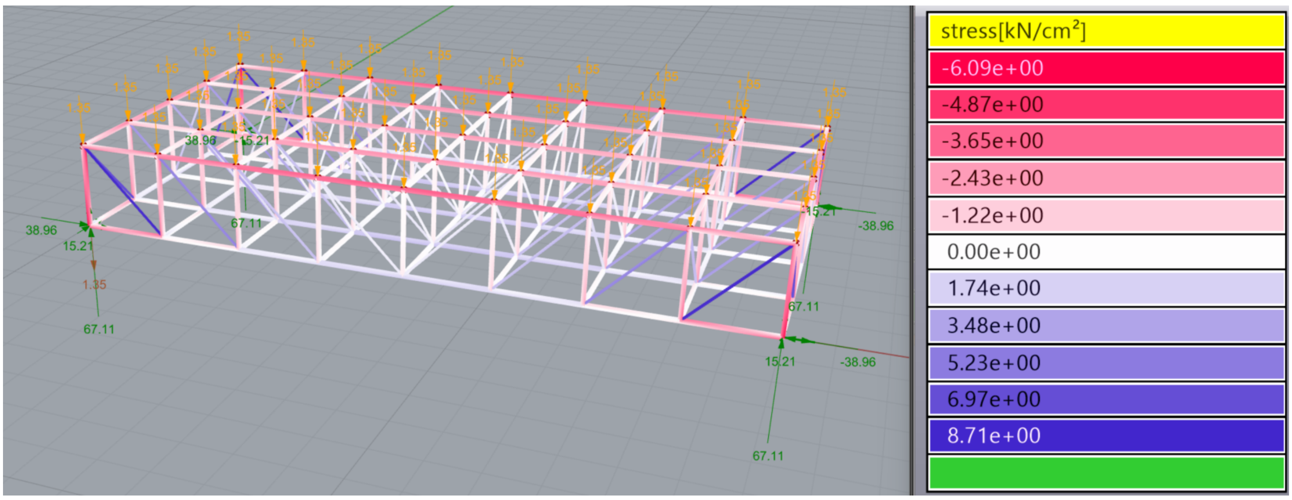
To check if the results from the model are accurate a validation is performed in RFEM 5. This is a FE software in which the models from Karamba3D can be directly imported. Module types 1 to 6 from the catalog are all tested. As can be seen in the figures in Appendix E the distribution of the stresses looks very similar between the two models. Besides, the reaction forces also indicate that the structures behave the same. The dead load of 10 kN/m^2 is used again in combination with self-weight just like for the verification. The same partial load factor of 1.35 is used. The eccentricity of the elements is included in the analyses. The cross-sections, material properties, and joint stiffness values are automatically imported from Karamba3D to RFEM.

The results of the analysis of the different modules are visualised in Table 8.1. The absolute percentual difference between the minimum and maximum stresses in the two software programs is indicated with the $|\Delta|$ sign. The differences are small, most are below 5% difference except for the maximum stress in module type 4. The stresses in the Karamba3D model are slightly larger in general than those in RFEM.

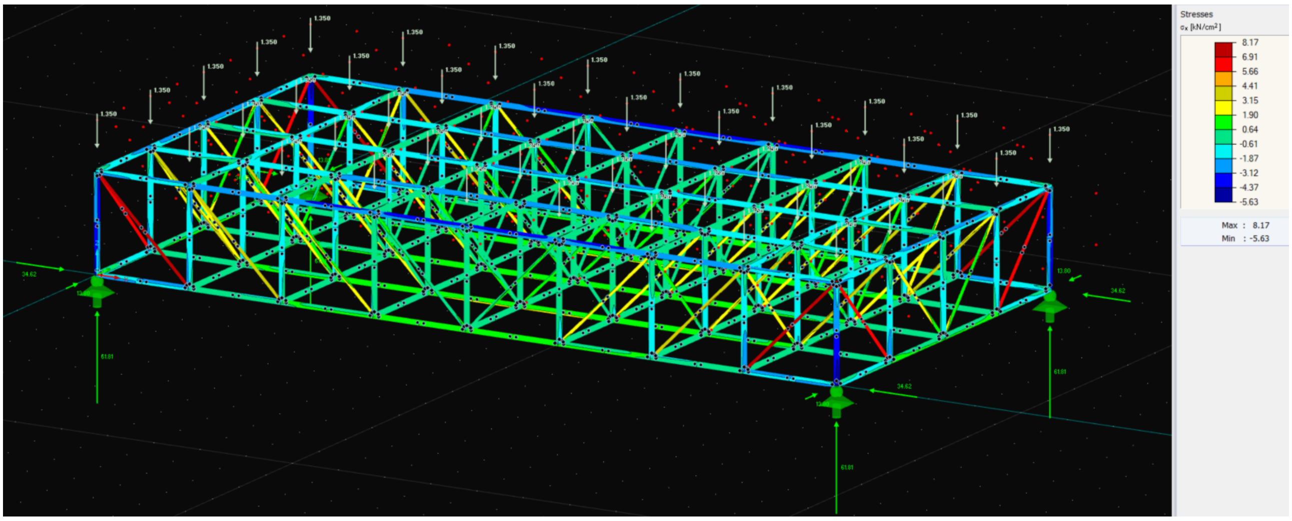
Table 8.1: Comparison of structural results between Karamba3D and RFEM for module types 1 to 6

	Module type 1			**Module type 2**			**Module type 3**		
Stress	Karamba3D	RFEM	$\|\Delta\|$	Karamba3D	RFEM	$\|\Delta\|$	Karamba3D	RFEM	$\|\Delta\|$
σ_{max} [kN/cm^2]	1.56	1.53	1.94	2.03	1.94	4.53	1.75	1.81	3.37
σ_{min} [kN/cm^2]	-2.76	-2.77	0.36	-3.14	-3.12	0.64	-2.63	-2.62	0.38
	Module type 4			**Module type 5**			**Module type 6**		
Stress	Karamba3D	RFEM	$\|\Delta\|$	Karamba3D	RFEM	$\|\Delta\|$	Karamba3D	RFEM	$\|\Delta\|$
σ_{max} [kN/cm^2]	1.84	1.23	39.74	2.11	2.03	3.86	1.67	1.62	3.04
σ_{min} [kN/cm^2]	-2.41	-2.30	4.67	-3.01	-3.00	0.33	-3.08	-3.06	0.65

For a small truss with a grid of 8 x 4 and module depth of 2 m, there is also a structural comparison performed. The same load combination is used on this frame as on the small modules, but the value of the dead load is reduced from 10 kN/m^2 to 1 kN/m^2. The distribution of stresses in the Karamba3D and RFEM model are visualised in Figure 8.1. The distribution of stresses is similar. There is tension in the diagonals, especially close to the supports, and compression in the top chords of the truss. The size of the support reactions is also in the same order of magnitude. For instance, the vertical support reactions in Karamba3D have a value of 67.1 kN and in RFEM a value of 61.8. This is a difference of just 5.3 kN and a percentual difference percentual difference of 8.2%. It differs only a couple of kN in force. A truss with modular elements is analysed here for simplification. The reason is that the duplicate elements for a large number of cubic modules could not all be loaded in the RFEM software. The eccentricities are thus not present for this frame. The simplified truss is still a good validation because the same model is used but without these extra beams.



(a) Truss analysis in Karamba3D



(b) Truss analysis in RFEM

Figure 8.1: Truss analysis in Karamba3D and RFEM

The results can be seen in Table 8.2. The percentual difference between the stresses for the two models is larger than for the single cubes. Also, deflection is measured because this is important for such a truss. This percentual difference is also bigger. The absolute difference between the deflection of the two models of 1.5 mm is very small for a truss of this size. For the stresses the absolute values are also in the same order of magnitude and relatively similar. Besides, both the stresses and the deflection are larger in the Karamba3D model which makes it conservative. This results in the conclusion for the validation that the model in Karamba3D is valid and conservative.

Table 8.2: Comparison of structural results between Karamba3D and RFEM for a small truss

	Karamba3D	RFEM	$\|\Delta\|$
σ_{max} [kN/cm^2]	8.71	8.17	6.40
σ_{min} [kN/cm^2]	-6.09	-5.63	7.85
Deflection [mm]	13.1	11.6	12.15

9 Case study simply supported truss

In this section the model is compared to a case study. Also, a generic truss will be constructed with a similar layout as the case study to check beforehand if a good solution is possible with the existing model.

9.1 Introduction to the case study

In their study, Irfan et al. analysed a space frame measuring 11 x 22 meters, examining various load combinations. They evaluated multiple types of frames for this grid, comparing different configurations. Specifically, they looked at steel space frames with two orthogonal latticed grids (both square configurations) and orthogonal square pyramid space grids with square-on-square offsets. These analyses included two-layer and three-layer grid structures [60]. The gridsize is 1.1 x 1.1 m and the space frame has a depth of 1.2 m for the double-layered grid and 2.4 m for the triple-layered grid. This example is used to see if the order of magnitude of the deflection is reasonable. The layout of the plan of the structure analysed is shown in Figure 9.1.

For the comparison to the model modules with a depth of 2 m are used. The structure itself has a width of 11 m instead of a width of 12 m. This has to do with the depth of the modules. The height is also different (2.0 m instead of 1.2 m or 2.4 m). This might lead to differences in deflection but it is expected that it has the same order of magnitude. The structure is large, there are $6 \times 11 = 66$ locations for where modules can be placed. This results in $34^{66} = 10^{101}$ different solutions when module types 1 to 6 are all used at once. It is not possible to calculate this with brute force.

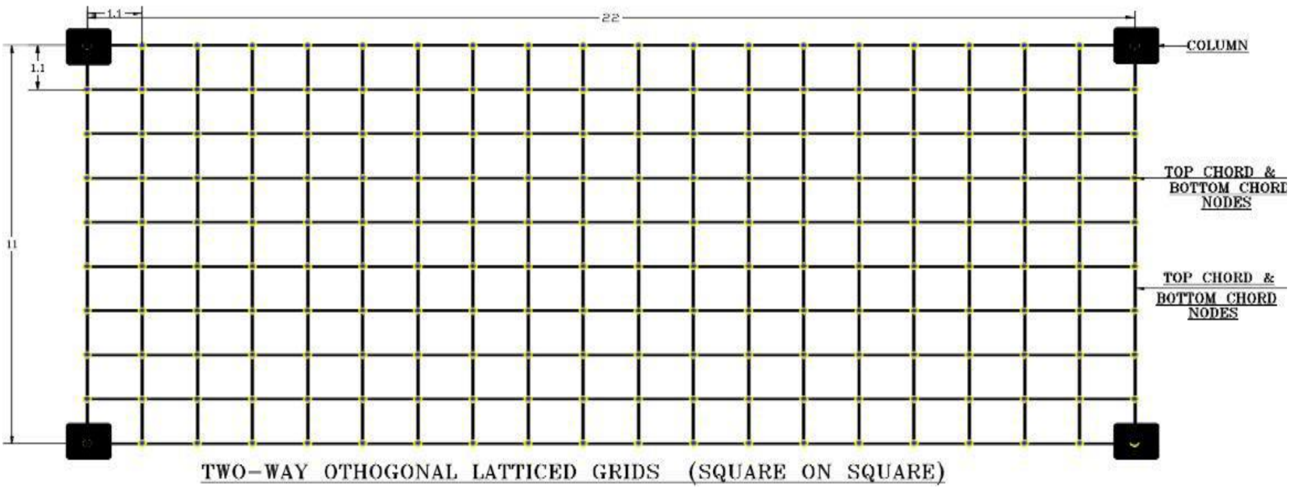


Figure 9.1: Plan of the double layered grid space frame (Source: Irfan et al. [60])

Different loads are acting on the structure: Dead load, live load, wind load, earthquake load, and temperature. For simplicity, the earthquake load is not analysed in this comparison. The values for dead load and live load can be seen in Table 9.1 and Table 9.2.

Table 9.1: Dead load considered (Source: Irfan et al. [60])

Weight	Load
Self weight factor	1.0
0.47mm Thick Galvalume Sheet	0.005 kN/m^2
Accessories, Eg- Bolts, Node, Etc.	0.004 kN/m^2
Purlin (RHS 96x48) & Stub	0.010 kN/m^2
Lighting - Point Load	0.1 kN

Table 9.2: Live load considered (Source: Irfan et al. [60])

Weight	Load
Non-Accessible Roof	0.75 kN/m^2

The building considered is located in India in Delhi. The wind load is calculated according to IS 875 part 3-2015 which is the Indian code for wind load design [61]. Constants used for the calculation of the wind loads are derived from Irfan et al. [60].

Table 9.3: Wind load parameters (Source: Irfan et al. [60])

Parameter	Value	Source
Regional Basic Wind Speed, V_b	47 m/s	Appendix 1
Risk Coefficient Factor, K_1	1.0	Table -1, design for 50 Years
Terrain Factor, K_2	0.934	Table -2, Category -3, Height -12m
Topography Factor, K_3	1.0	Clause 6.3.3
Important Factor Cyclonic Region, K_4	1.0	Clause 6.3.4
Wind Directionality Factor, K_d	0.90	Clause 7.2.1
Area Averaging Factor, k_a	0.80	Clause 7.2.2 - Table-4
Combination Factor, K_c	0.90	Clause 7.3.3.13
Solidarity Factor, φ	0	Clause 7.3.3.3

The design wind speed V_z is calculated using Equation (9.1). Plugging this into Equation (9.2) and Equation (9.3) gives the design wind pressure for the structure in this particular area. The parameters (K-values) used are all taken from Table 9.3. Their meaning is also given there. The design wind load p_d may not be lower than 70% of the wind pressure at height z p_z.

$$V_z = V_b \cdot K_1 \cdot K_2 \cdot K_3 \cdot K_4 \tag{9.1}$$

$$p_z = 0.6 \cdot V_z^2 \tag{9.2}$$

$$p_d = max(K_d \cdot K_a \cdot K_c \cdot p_z \ ; \ 0.70 \cdot p_z) \tag{9.3}$$

In which:

V_z = Design wind speed [m/s]

p_z = Wind pressure at height z [N/m^2]

p_d = Design wind pressure [N/m^2]

Both on the windward and leeward side of the structure wind forces are applied. The values of the pressure coefficients is taken from the IS 875 part 3-2015. The table for pressure coefficients on walls of rectangular buildings can be found in Appendix F. These values multiplied with the design wind pressure p_d result in the wind pressure load on each side of the structure for two horizontal perpendicular wind directions. An explanation of the different surfaces A, B, C, and D is given in Appendix F. If the wind is acting on the greater horizontal dimension of the structure surfaces A and B are loaded and the wind angle is 0°. If the wind is acting on the lesser horizontal dimension of the building the surfaces C and D are loaded and the wind angle is 90°.

Table 9.4: Wind pressure on different surfaces for two horizontal wind directions

Wind angle	$c_{pe} \cdot p_d$ **[kN/m^2] for surface**			
θ [°]	A	B	C	D
0	0.57	-0.24	-0.57	-0.57
90	-0.40	-0.40	0.57	-0.08

Besides, temperature loads have been added to the load conditions. A variation of temperature from 10 °C to 40 °C is used. The Karamba3D component for loads can be used to apply a temperature load on the structure. This can be significant for steel structures. All of the different loads have been combined into load combinations. These are shown in Table 9.5. A division is made between strength design loads ULS and serviceability design loads SLS. The load combinations for ULS are numbered from 1 to 4 and for the SLS from 5 to 8.

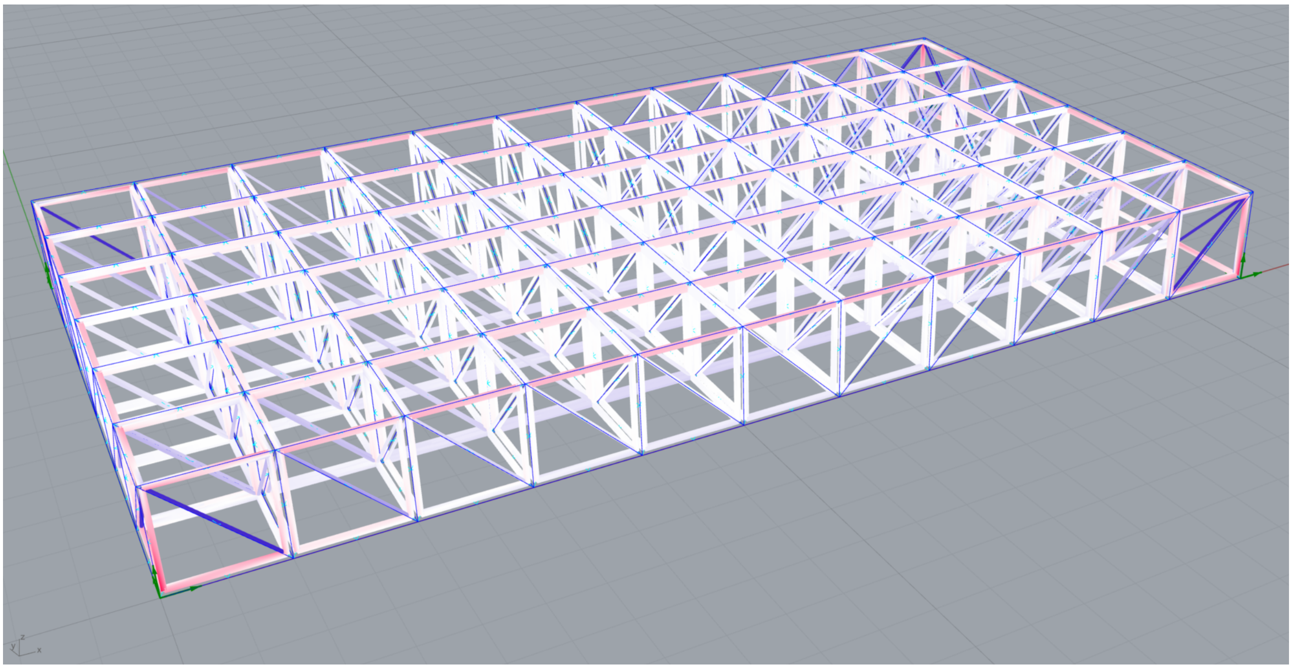
Table 9.5: Load combinations for strength and serviceability design as per IS 800-2007 (Source: Irfan et al. [60]). DL = Death Load, LL = Live Load, WL = Wind Load, TEMP = Temperature Load

LC	Strength Design	LC	Serviceability Design
1	1.5 DL + 1.5 LL	5	1.0 DL + 1.0 LL
2	1.5 DL + 1.5 WL	6	1.0 DL + 1.0 WL
3	1.2 DL + 1.2 LL + 1.2 WL	7	1.0 DL + 0.8 LL + 0.8 WL
4	0.9 DL + 1.5 WL	8	1.0 DL + 0.8 LL + 0.8 TEMP

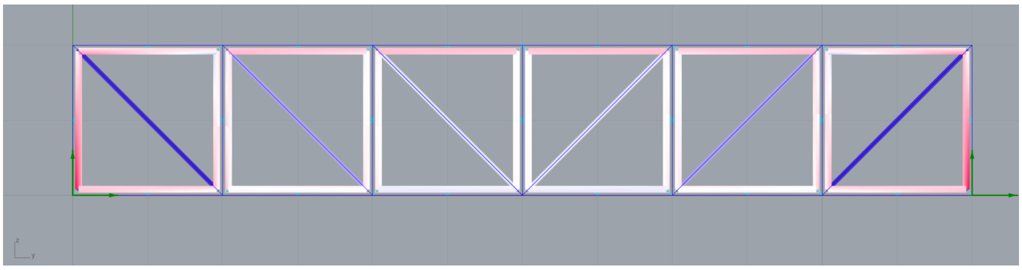
During the optimisation, only a single LC can be used. The reason is that different load cases have different optima. The most significant one, LC1 is chosen as LC to start the optimisation with. When the analysis is finished, the performance for the other load cases will be analysed as well.

9.2 Generic truss in Grasshopper

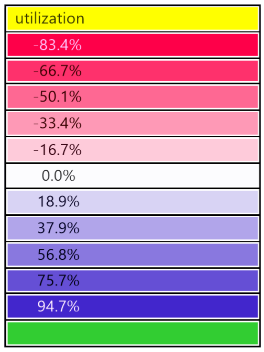
At first, a structure is made in Grasshopper which is close to the topology of the various structures designed by Irfan et al. [60]. This is done to see if the model is capable of staying within all constraints before running complete simulations. A truss model slightly different from the one in Section 7 is made. t was observed that the Howe truss had many diagonals in compression that led to a low value for the BLF. With the Pratt truss configuration, the diagonals will all be in tension and the columns will take the highest compressional forces. Therefore, this configuration is chosen for the structure. The load path will be slightly longer and deflection is also expected to be a little larger. Since this was not critical in the earlier simulations this will have a minor influence on the fitness function. Again module type 6 is used for creating the structure. The only elements that are prone to buckling are the columns and the upper chord beams. These have larger cross-sections so they need a much larger load before buckling occurs. The supports at the corner can all resist lateral loading since there are lateral load cases used in the case study. In the model the supports therefore all have restrained translational motion in the x-, y- and z-direction. The structure can be seen in Figure 9.2.



(a) Pratt truss: 3D view



(b) Pratt truss: side view

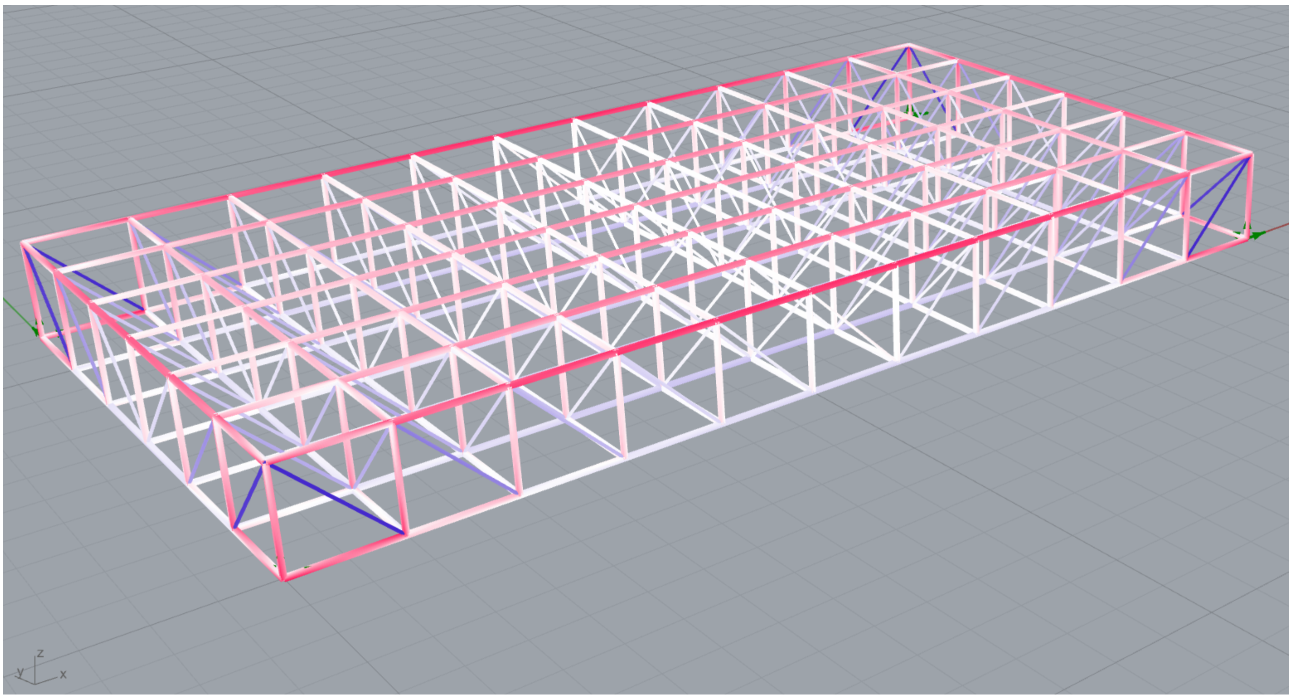


(c) Pratt truss: Utilisation of the elements

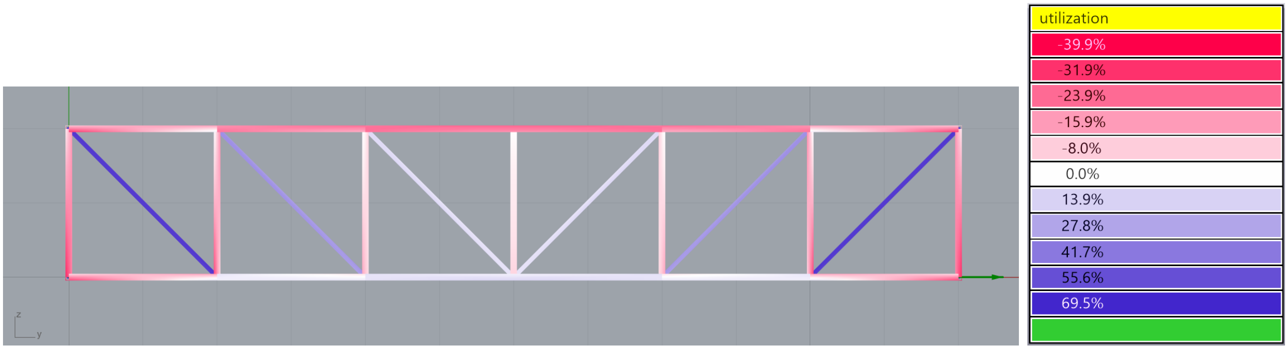
Figure 9.2: Pratt truss configuration

This truss model experienced large compressional stresses in the columns at the supports. These stresses were slightly larger than the maximum allowed level. Increasing the cross-section of all elements does not help much since the weight of the structure is also heavily increased, leading to an increase in stress in the columns. For this reason, the columns are reinforced locally. In the model, the steel strength of these elements is increased from S235 to S355. This change led to an extreme utilisation of 95.7%, located at the diagonals that lead the stresses to the corner column. In Table 9.6 the performance of the truss can be seen. It can be seen that the mean utilisation of the elements is just 13.0% indicating that most of the elements can take the forces easily. Furthermore, the percentage of axial deformation energy is larger in this truss than was the case for the small verification model in Section 7. This shows that for a real large-size truss structure most of the stresses are axial which is how a normal truss works.

Besides a truss that works with the cube modules, the model can also be simplified to a more regular space frame. This can be done by deleting the parallel beams in the model and by giving the elements no eccentricities. The model now is transformed into a regular Pratt structure. The same cross-sections are used as in the former model. Also, the corner columns are strengthened with a higher steel strength for good comparison. This structure is shown in Figure 9.3. The Grasshopper model can still be used in the same way as the original model with adjacent building blocks, but the topology optimisation of the cubes is then on the element level. The inspiration for this model comes from the CUBIC system discussed in Section 4. The difference is that this system has diagonals, which is different from the original [2].



(a) Pratt truss with modular elements: 3D view



(b) Pratt truss with modular elements: side view

(c) Pratt truss with modular elements: Utilisation of the elements

Figure 9.3: Pratt truss with modular elements configuration

The results for this truss can be seen in Table 9.6 as well. Space grids are lightweight and can weigh only 20 kg/m^2 [62]. There is a great variance in the weight of course. For instance, Stadium Australia has a large space frame roof with a weight of 88 kg/m^2 [2]. However, this is considered a relatively heavy space frame. The Pratt truss with modular elements has a weight that is comparable with that of Stadium Australia. The original structure is much heavier than normal space frames. Besides the weight, other factors make the modular elements' space frame score better on fitness. It also has a lower extreme utilisation and lower displacement. The BLF is comparable.

The number of joints and the number of elements for the trusses are also shown. The number of elements is the number of complete beams and columns, disregarding the splice connections that split them in two. This means it is the total number of elements that connect two vertices of the cubes. The number of elements in the Pratt truss with modular elements is almost twice as low as the Pratt truss with modular cubes. This explains the large difference in weight.

Table 9.6: Pratt truss performance metric for LC1

Performance metric	**Pratt truss with modular cubes**	**Pratt truss with modular elements**
Weight [t]	28.0	13.8
Weight with joints [t]	36.3	22.1
BLF [-]	10.0	10.3
Displacement [mm]	50.1	39.5
Extreme utilisation [%]	95.6	69.5
Mean utilisation [%]	13.5	15.3
Percentage of axial deformation energy [%]	94.9	97.9
Fitness [-]	2.7	0.4
Nr. joints [-]	2355	1593
Weight/Area [kg/m^2]	137.4	83.6
Nr. elements [-]	1056	531

9.3 Model simulation

As a first test of the reconfiguration model, the model was used with the integrated stiffness iteration loop. This structure is much larger than the one in Section 7 and therefore quickly used up all available RAM. Only 5 generations were calculated before the program had to be stopped. The convergence graph showed good convergence, however, no clear conclusions could be drawn from the distribution of modules in the structure, their orientation, or any changes that happened per genome number. The problem with large amounts of memory being used by the model was already known from Section 7 so this was expected. To overcome this the stiffness iteration loop has been omitted for this large-scale model as well. After providing the initial stiffness values to the joints, based on the |M|/N ratio in the joints and the stiffness analysis new stiffness values are assigned. After this step no iteration is performed anymore, leading to a great reduction of computation time and saving of RAM.

As mentioned earlier when using module types 1 to 6 of the catalog results in 10^{101} different possible solutions. The more module types and possible locations there are, the larger the design space. This is an exponential relationship which can be seen from Equation (9.4).

$$S = \left(\sum O_m\right)^N \tag{9.4}$$

S = Design space [-]

O_m = Number of module orientations [-]

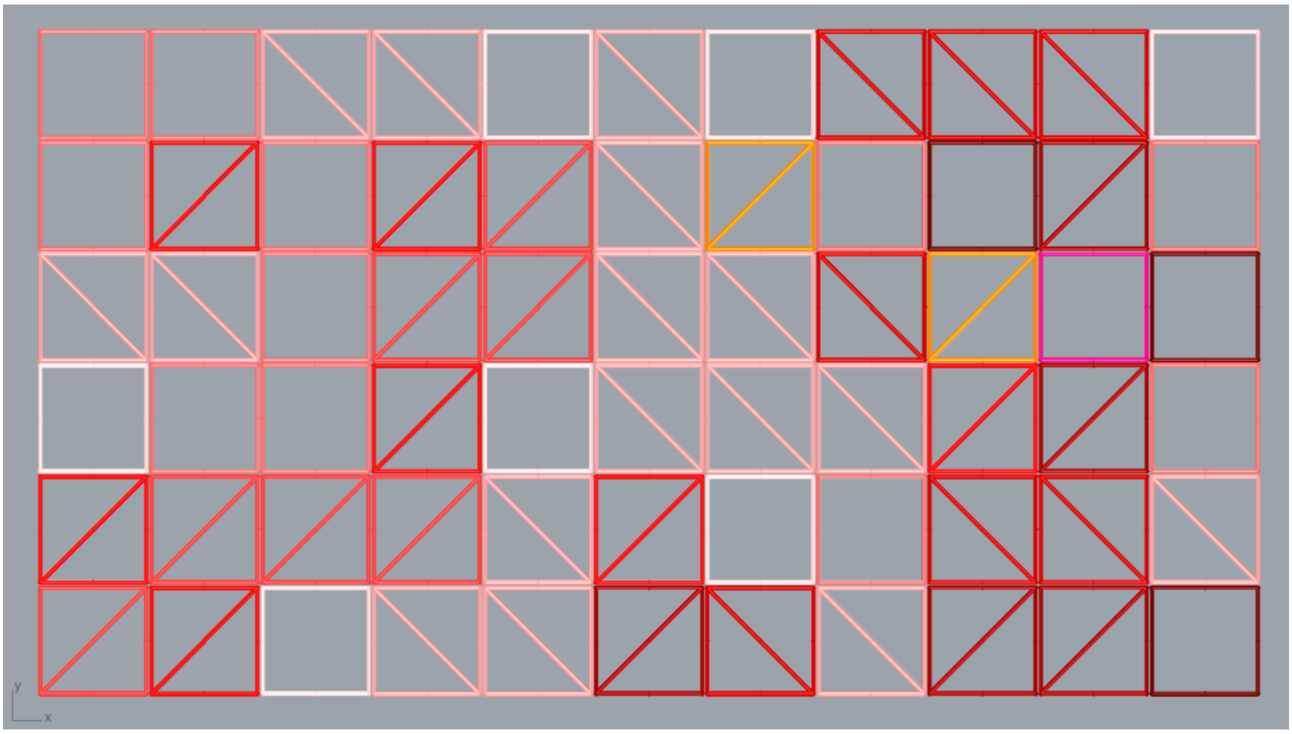
N = Number of possible locations for the modules [-]

For this reason, the model is first tested with fewer module types at once. Since the Pratt truss is built up from solely module type six a simulation will be performed with only this module type. Furthermore, a simulation with two module types will be performed as well. These are modules of type 3 and 4 because there is a large weight difference between the two modules and they are both perfectly symmetrical. Therefore, they each have only 1 possible configuration, leading to 2 possible configurations per location. As can be seen from Table 9.7 there is a large difference in the number of different configurations between the two runs. However, both simulations cannot be computed with brute force because of the time it takes to calculate a design. A simulation of such a large model takes longer than for the small one in Section 7. For this model, one configuration takes about 10 seconds per solution.

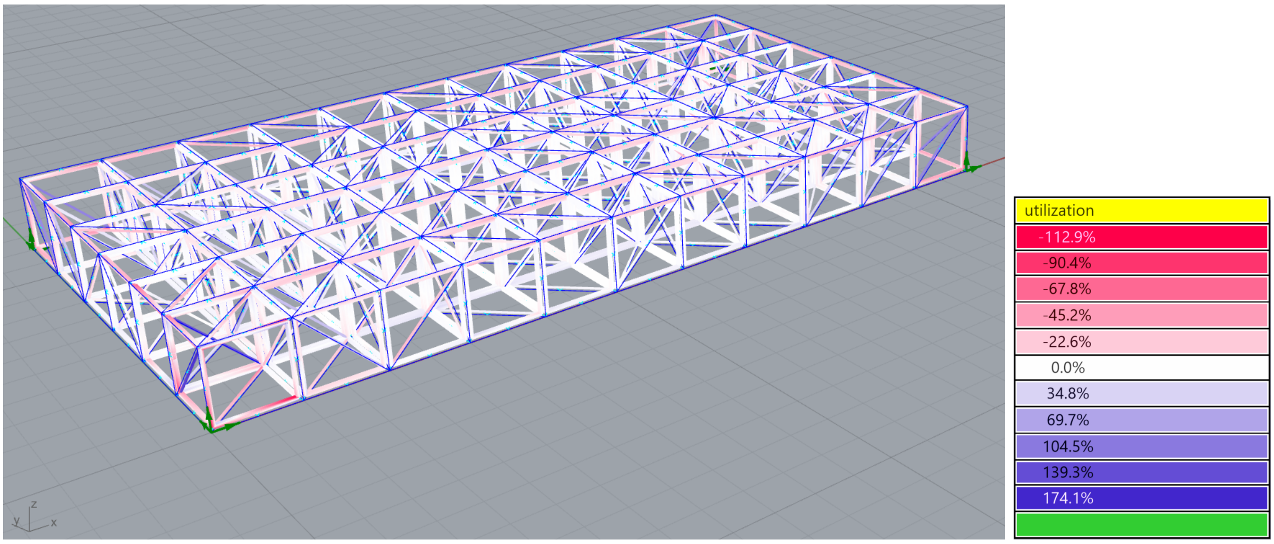
Table 9.7: Simulations and design space

Simulation	Module types	Nr. of possible solutions
1	6	10^{71}
2	3, 4	10^{19}

The resulting structure for a simulation with module type 6 can be seen in Figure 9.4. It can be seen that there is no clear repetitive structure in this solution.



(a) Structure model with module type 6: Bitmap



(b) Structure model with module type 6: 3D view

(c) Structure model with module type 6: Utilisation of the elements

Figure 9.4: Structure model with module type 6

The convergence graph of the model with only module type 6 can be seen in Figure 9.5. It shows clear convergence towards a final solution with many improvements of the best solution during the simulation. It improves the best solution 19 times which shows that the model works for finding better solutions over time. The percentual decrease from the first best solution is small, only 5.5% after 50 generations.

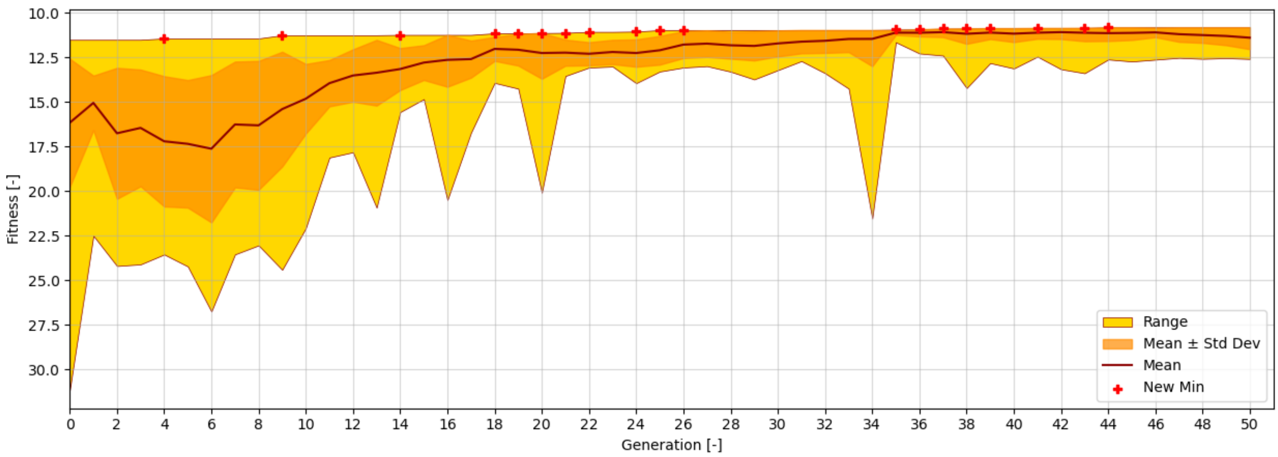
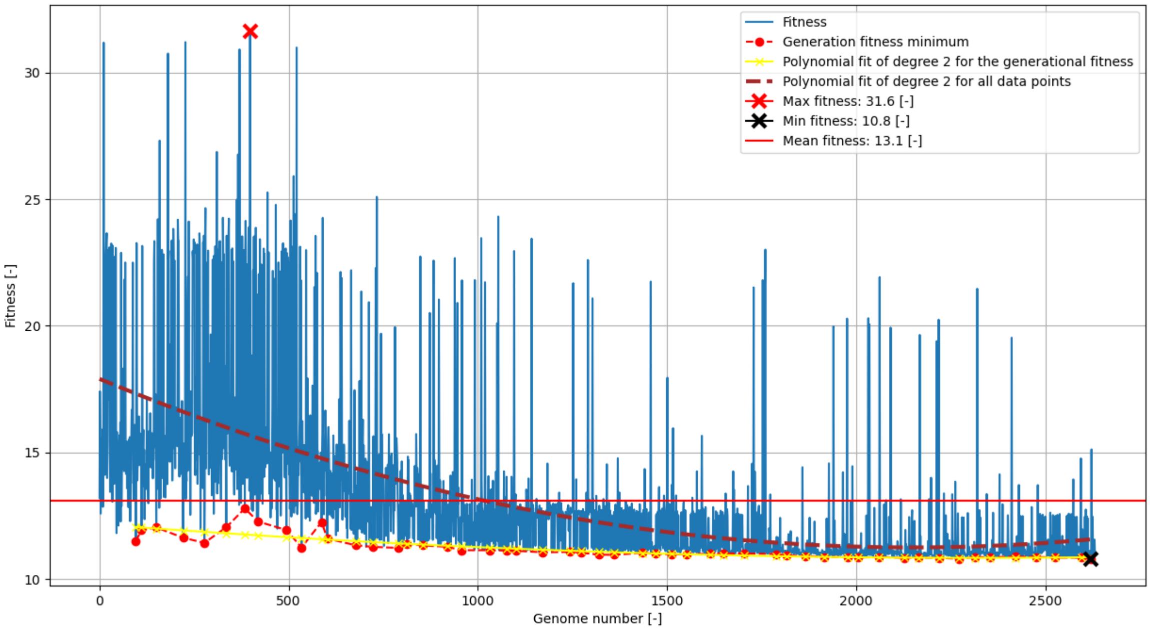


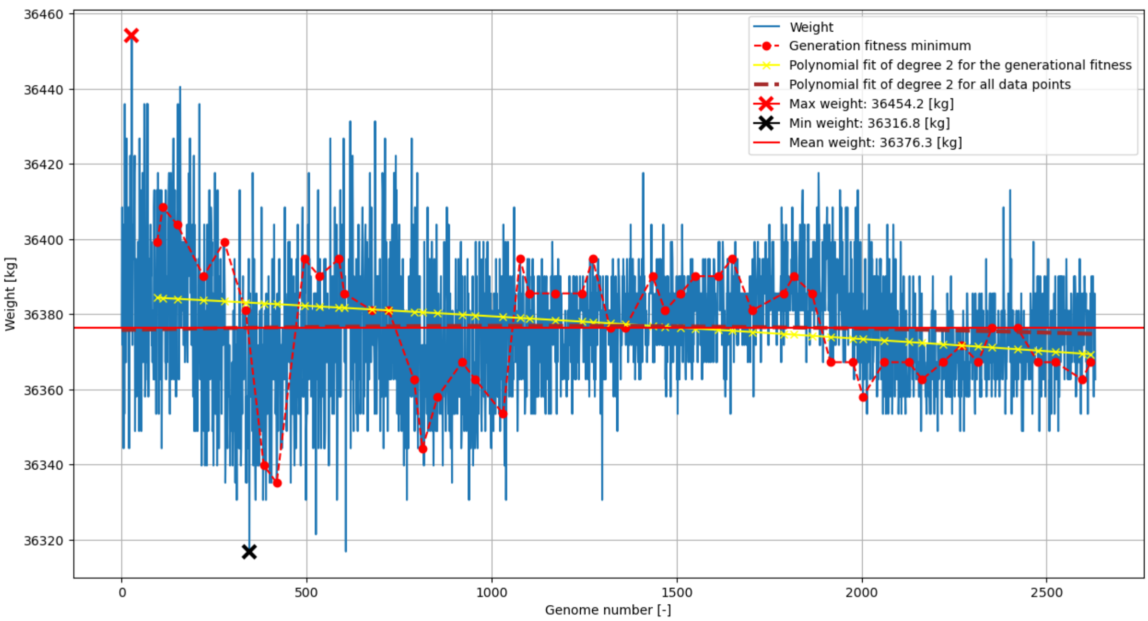
Figure 9.5: Convergence graph for the model with module type 6

Figure 9.6 shows the fitness, weight, BLF, displacement, and utilisation per genome number. The final solution after 50 generations is not close to the fitness value of the intuitively chosen truss structure. This is because the model converges slowly towards a final solution since there are so many different options. It could have taken many more configurations to find a global optimum. The convergence is slow but good. The fitness value has a clear downward trend for both the generational fitness minima and all the data points. This is better observed in these figures than in the small verification model. This was expected beforehand because, on a small scale, the effects of choice of topology are less important for strength and stability. That is why constructions such as trusses are made for large structures.

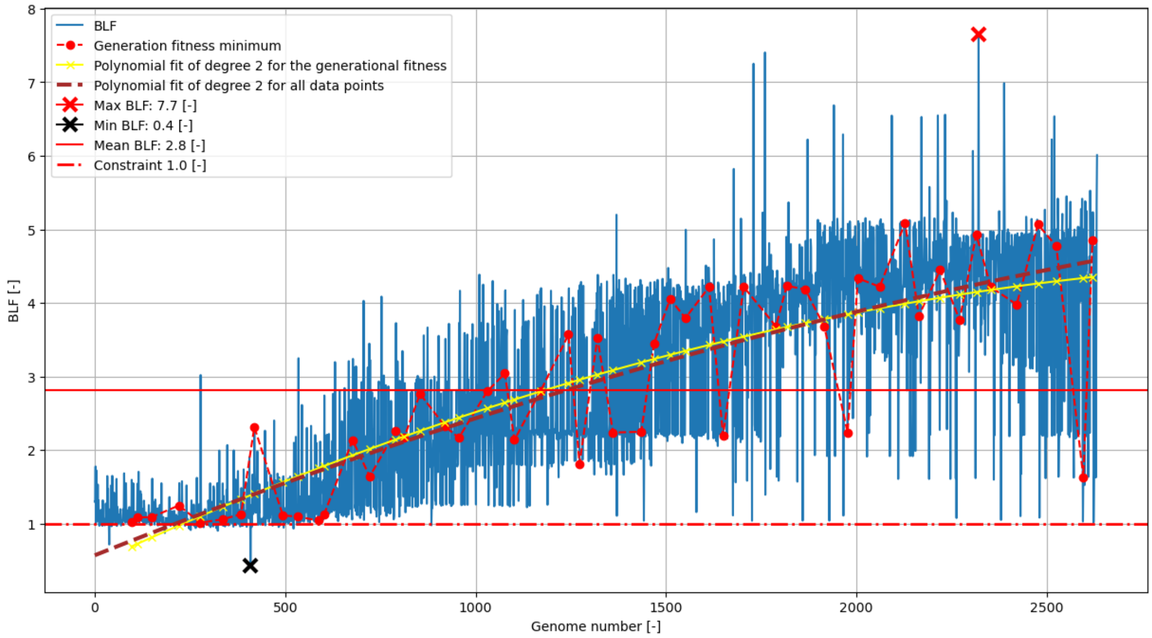
A polynomial of degree 2 is fitted to all the generational fitness minima that show these trends clearly. This polynomial regression is done using the method of least squares. The weight does not show much because only a single module type is used. The only reason that there is fluctuation is because the joints change in different configurations. Furthermore, the BLF has a clear upward trend, and the deflection clear downward one. Looking at the fitness minima for utilisation there is no clear downward or upward trend. At the start of the simulation, it goes up and near the end, it goes down. None of the utilisation values is below 100%. Looking at all of the data points for utilisation there is a clear downward trend. All these trends are exactly what is wanted with the logarithmic barrier functions used for these constraints.



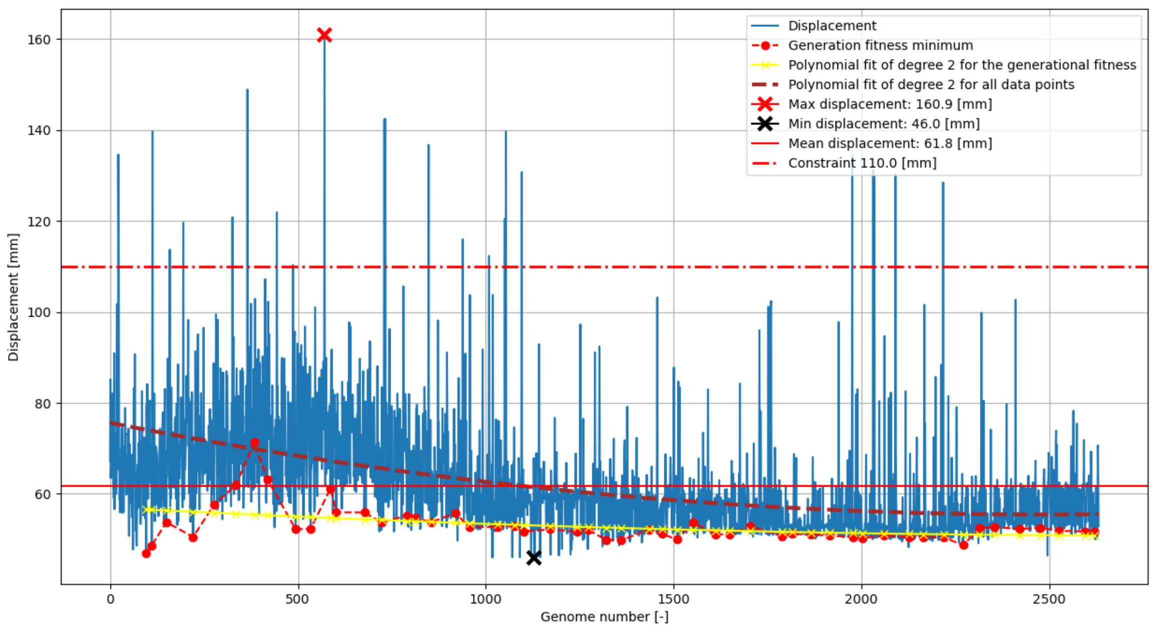
(a) Model with module type 6: Fitness per genome number



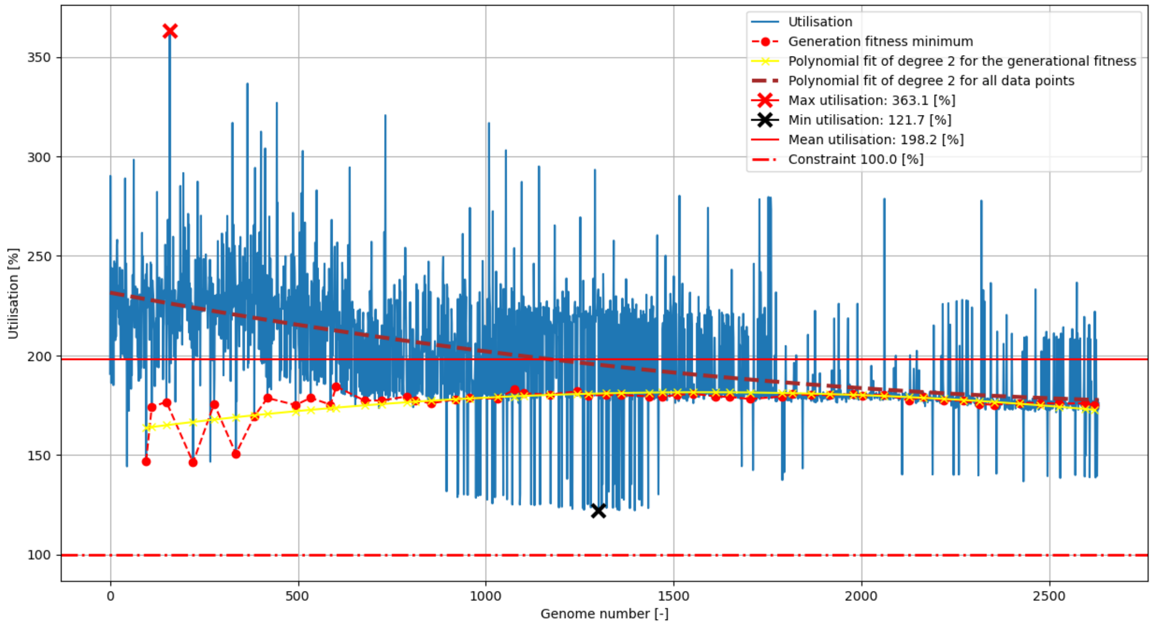
(b) Model with module type 6: Weight with joints per genome number



(c) Model with module type 6: BLF per genome number



(d) Model with module type 6: Displacement per genome number



(e) Model with module type 6: Extreme utilisation per genome number

Figure 9.6: Model with module type 6: Fitness, weight, BLF, displacement and extreme utilisation per genome number

Even more so than for the simulation for the small verification model, the percentage of axial deformation energy is high. The mean is 81.1% which is similar to the 79.6% found for verification. There are no outliers anymore as was the case for the verification. Values below 50% are not found. This shows that the behavior of the structure is that of a truss.

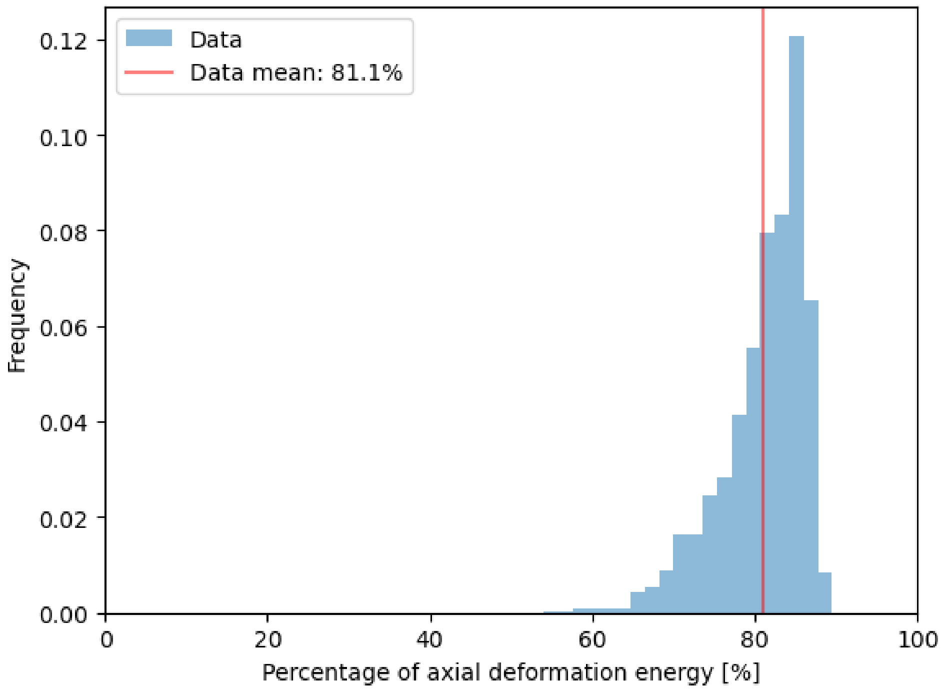
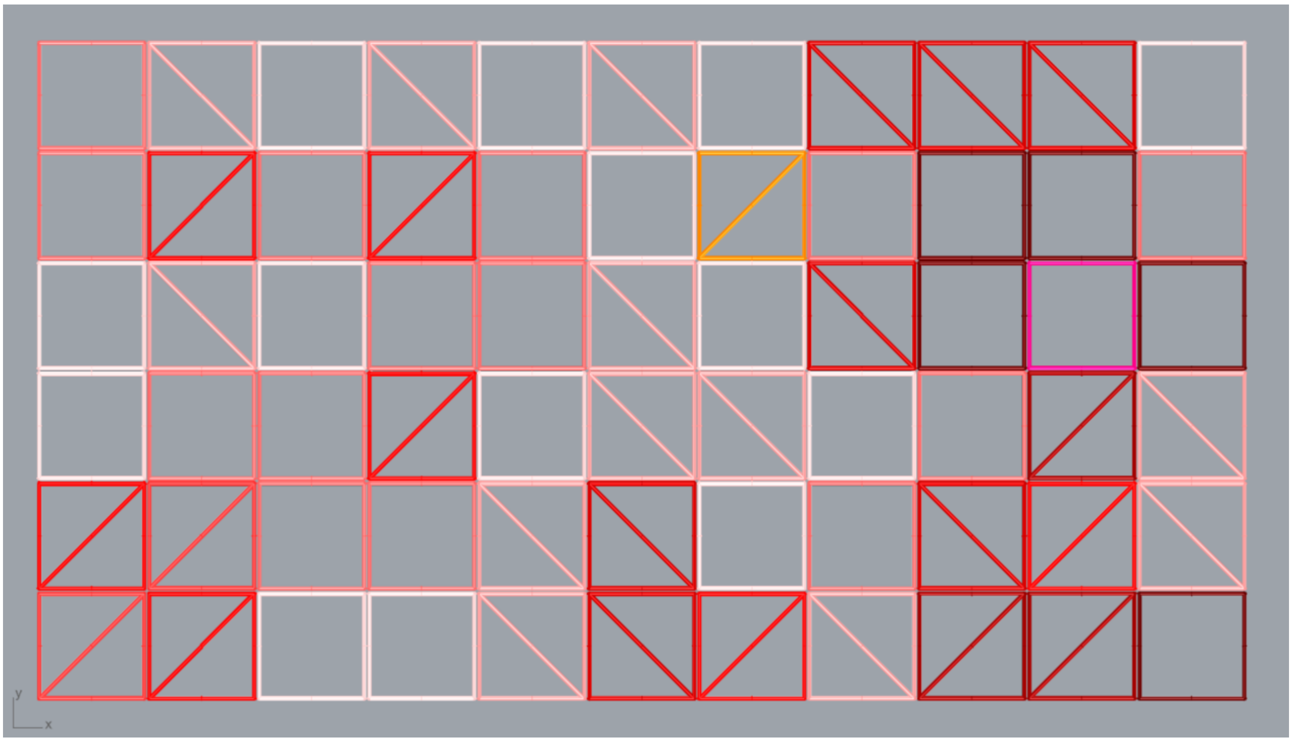
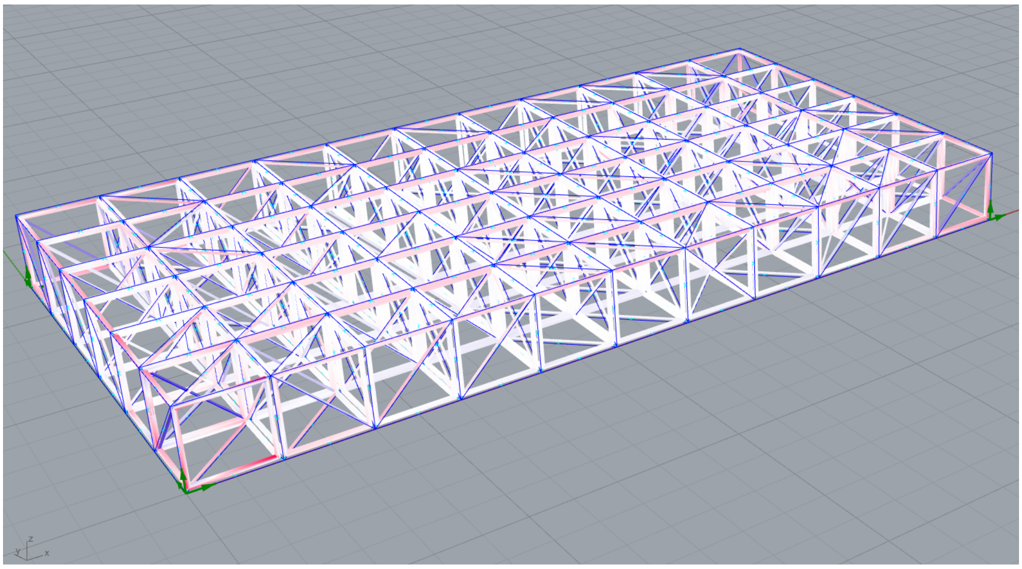


Figure 9.7: Histogram of axial deformation energy

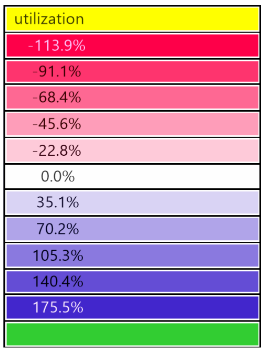
A long run on the same model has also been performed on a computer of TU Delft with more RAM memory. The final solution for this simulation can be seen in Figure 9.8. There is still no repetition in the structure. In most of the corners, there is a preference for the same module orientation as for the Pratt truss. This is however the only thing that shows resemblance. Some more modules have similar orientations but there is no coherence.



(a) Validation model with module type 6, long run: Bitmap



(b) Model with module type 6, long run: 3D view



(c) Model with module type 6, long run: Utilisation of the elements

Figure 9.8: Model with module type 6

The conversion graph is shown in Figure 9.9. New minimal solutions are found but not as frequently as in the former simulation. The improvement is also just 1.0% over 205 generations. This graph has been made with python and shows almost exactly the same figure as the output of the Galapagos plugin. The only difference is that the bottom part of the "Range" is a little spikier for the graph made with Python. This could be due to the difference in scale of the y-axis. It has not been found what scale is used by Galapagos. The reason that this graph is used is that Galapagos only shows a maximum of 55 generations in a single plot. Not everything could have been shown otherwise. The most important feature of the graph is the appearance of new best solutions. Within the first 50 iterations, it happens the most. Just like before, it converges quickly at the beginning and then the graph flattens.

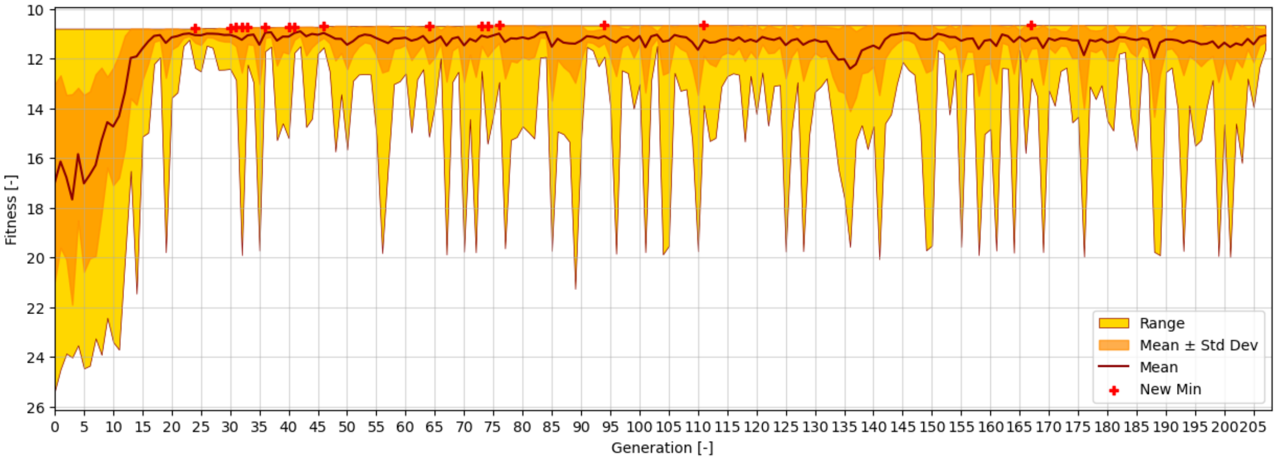


Figure 9.9: Conversion graph of the model with module type 6, long run

The fitness of this simulation shows a decline at the start just like before. After that, the values for the fitness function for the different solutions are steady. Similar observations were made for the BLF, deflection, and extreme utilisation. These are not shown because it would be similar to the results of Figure 9.6 but with more data points.

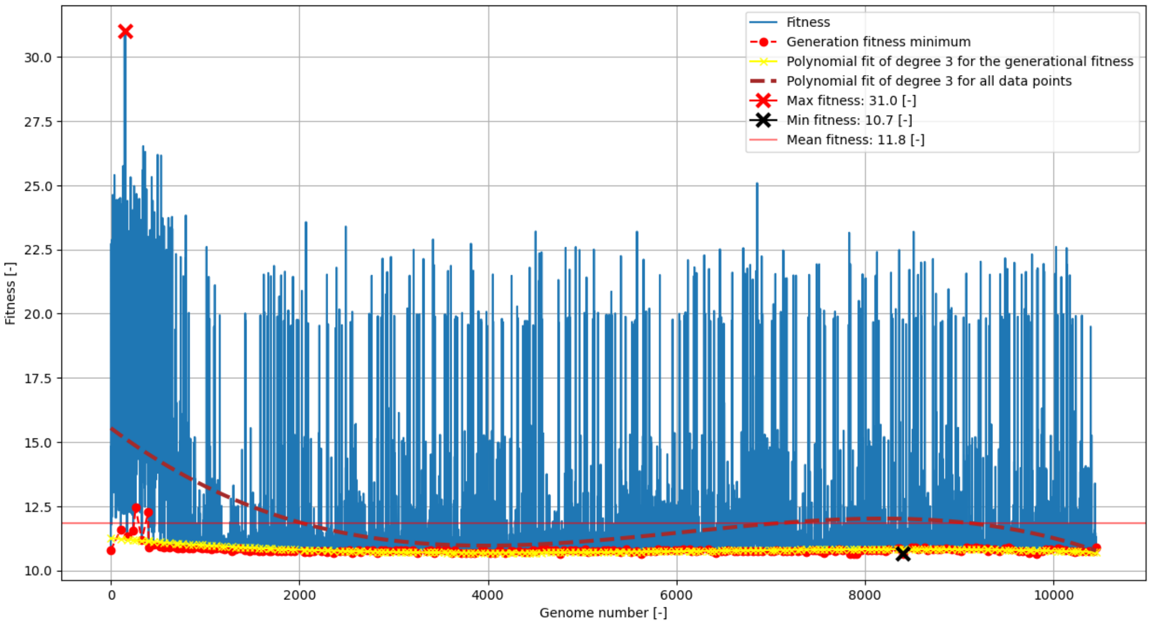
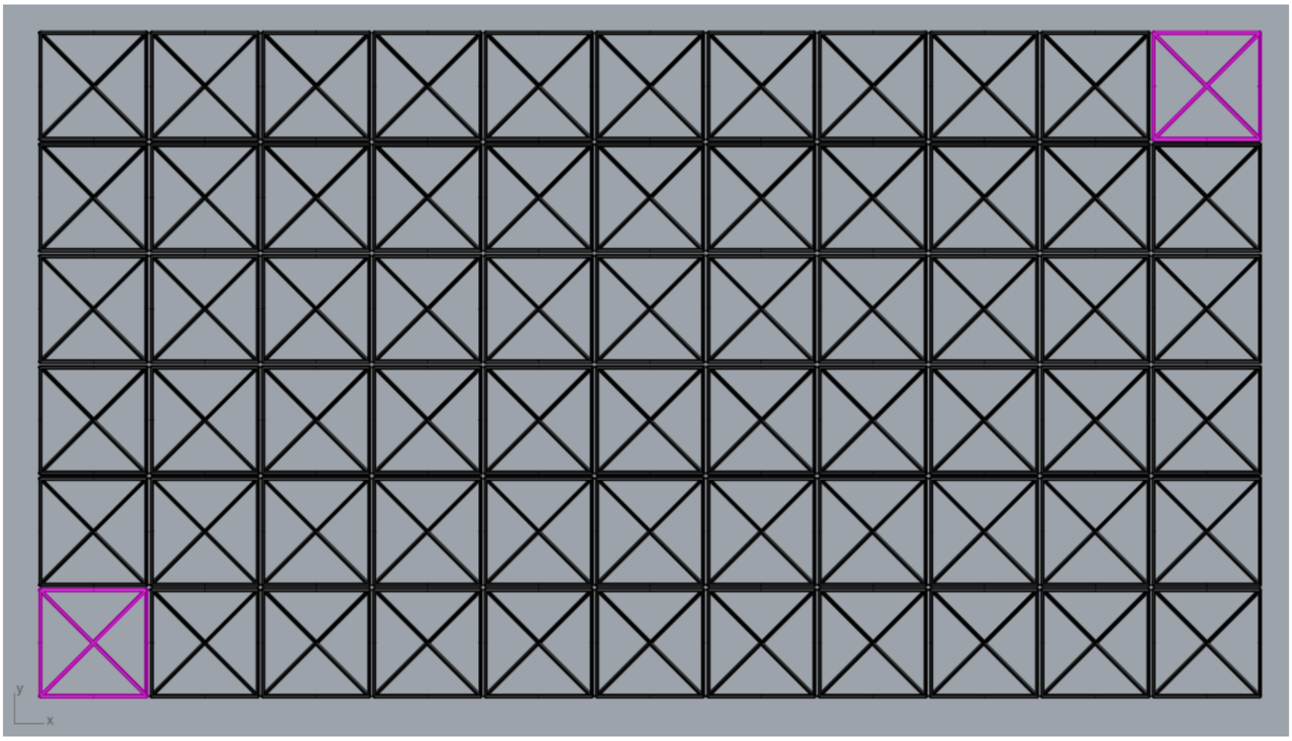


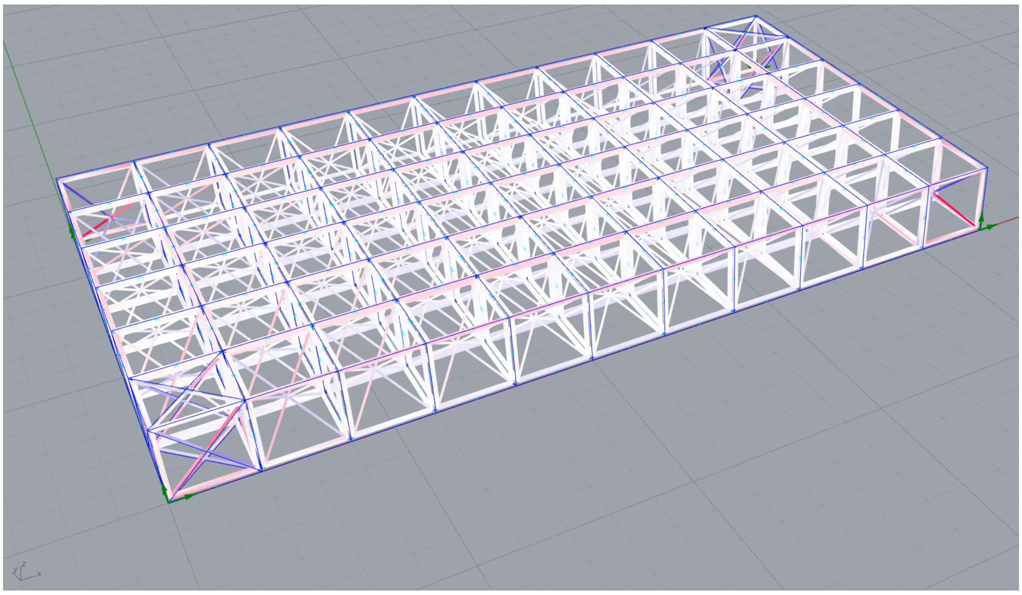
Figure 9.10: Fitness of the model with module type 6, long run

9.4 Model with 2 module types

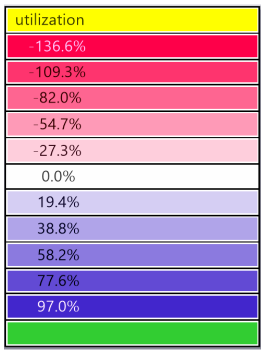
This simulation is performed with two module types. These are module types 3 and 4 with a combined number of $7.39 \cdot 10^{19}$ possible configurations. They are both completely symmetric modules which is why there are so much less possible configurations than for the simulation with module type 6. The same support conditions and loads are applied. The final result can be seen in Figure 9.11. It has a fitness value of 9.288. As can be seen, the model does show a clear preference for module type 3 in most positions. The only two locations where module type 4 is chosen are at the supports. If at all locations module type 3 was chosen the fitness would be lower (9.360) than in the final solution. Putting module type 4 at every corner results in a much better solution with a fitness value of 1.955. This is mainly because the extreme utilisation is now below the treshold (89.4%) which was not the case before. Module type 4 is better at directing forces to the supports. It is also checked what happens when module type 4 is placed in every location. This results in a fitness value of 3.176. The constraints are not exceeded but the structure is much heavier with this type of module than with module type 3. It is therefore likely that the situation where module type 4 is put at the corners and module type 3 everywhere else is the optimal solution for this design space. The GA came very close to this solution. This simulation shows, therefore, that for a large model, it is possible to get to a good solution when the design space is not too large. If modules have only 1 possible configuration this is an advantage.



(a) Model with module type 3 and 4: Bitmap



(b) Model with module type 3 and 4: 3D view



(c) Model with module type 3 and 4: Utilisation of the elements

Figure 9.11: Model with module type 3 and 4

As can be seen from Figure 9.12 the best configuration is already found at generation 9. This is quick, also seeing that there are 91 generations in the simulation. It is not surprising that this happens because there are only two options for modules to choose from for each location. Module type 3 is preferred since module type 3 weighs 604.9 kg and module 4 weighs 850.3 kg. They both have a good structural performance in terms of stiffness. The best solutions will therefore from the start have a large amount of module type 3. A smaller design space thus leads to quicker convergence.

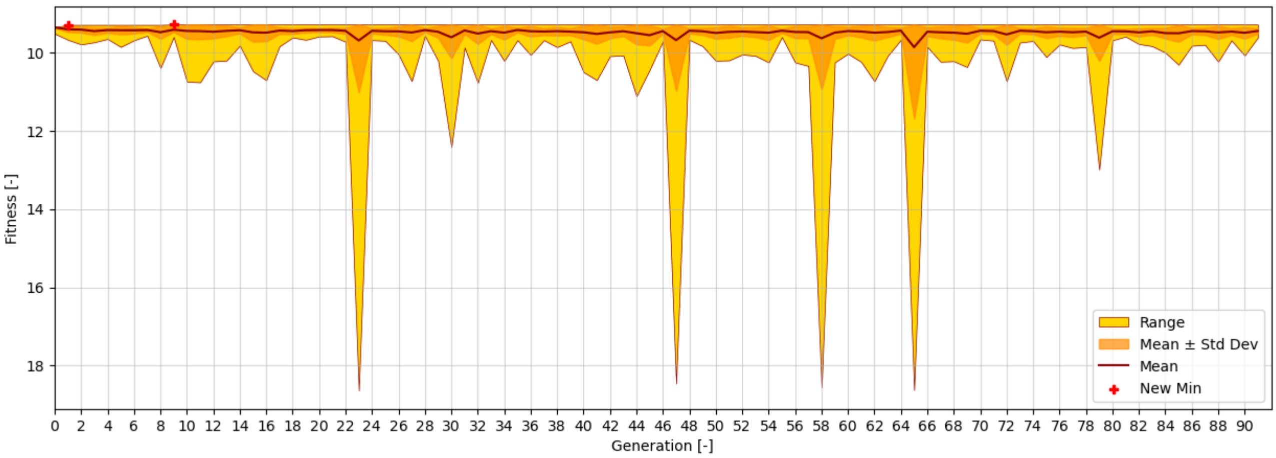


Figure 9.12: Convergence graph for the simulation with module type 3 and 4

The fitness over the generations can be seen in Figure 9.13. There is a variance in the values but the mean value is much lower for this simulation. It is also close to the minimum value. There is no real observable trend in the figure. This shows also that good solutions are found relatively quickly. The BLF for the solutions is within limits and the same is true for the displacement. However, the extreme utilistation is too large for this structure as well.

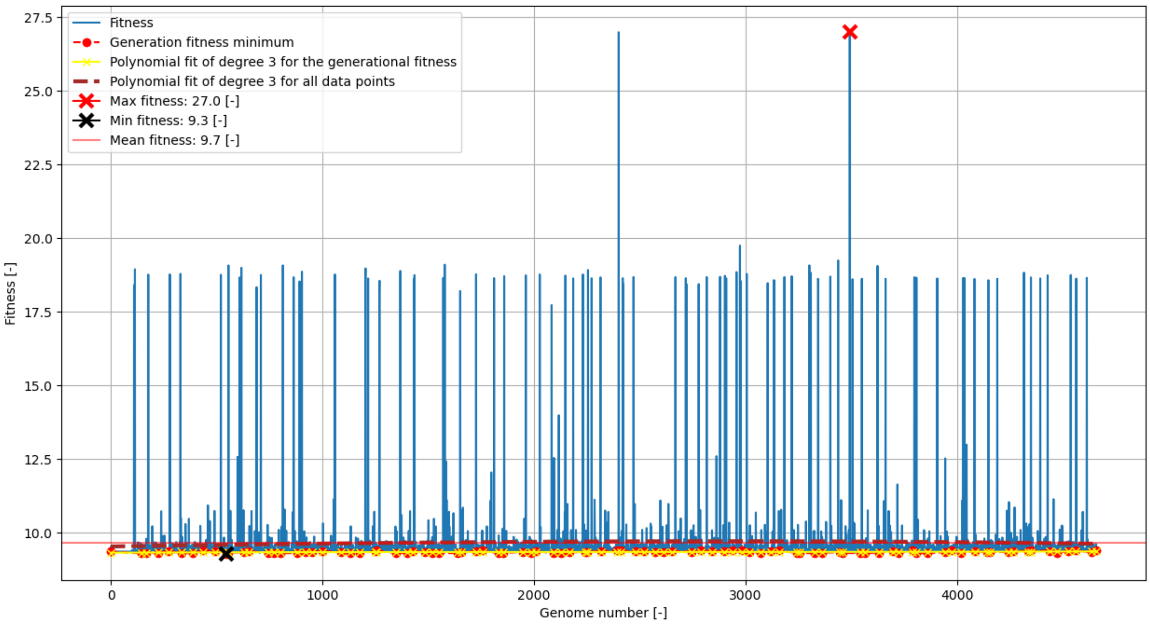
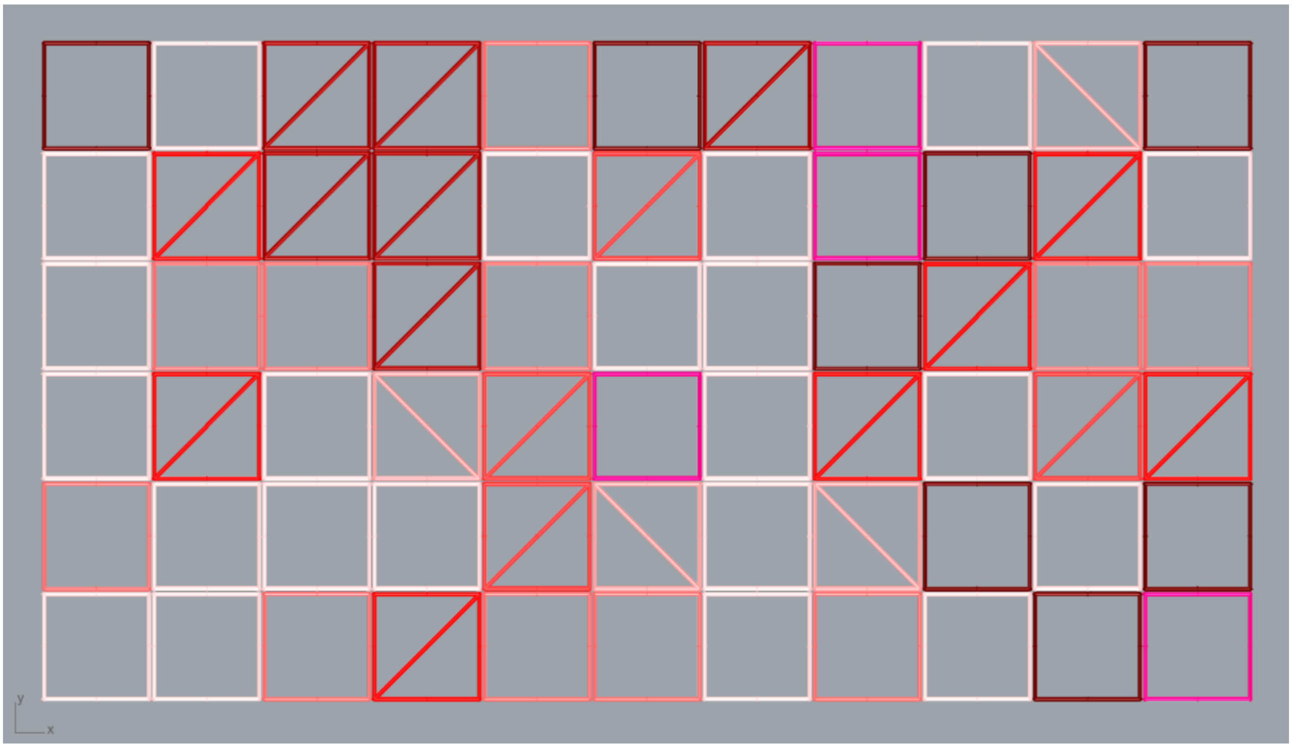


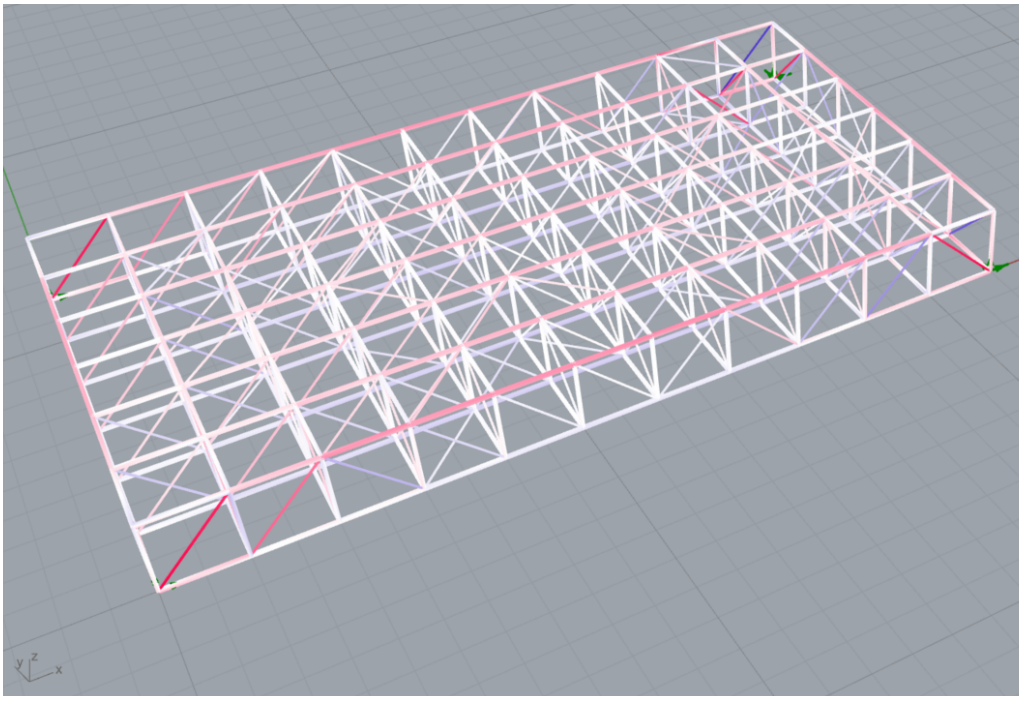
Figure 9.13: Fitness of the model with module type 3 and 4

9.5 Model with modular elements

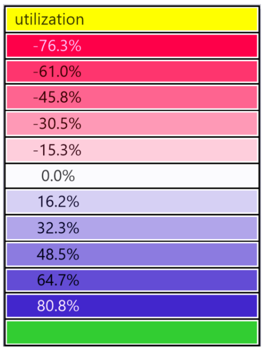
The model is also able to run a simulation where the parallel beams of adjacent cubes are formed into one. The number of elements is lower and so is the weight. This simulation has no modular cubes but modular elements and is performed with only module type 6 to see if a truss can be created. Since the model has to analyse fewer elements it runs faster. Almost the same number of generations is reached within a day that was reached for the modular cube elements within two days. the final result can be seen in Figure 9.14. It can be seen from this figure that there are more modules with their diagonals in the vertical plane and not the horizontal. This is better for the distribution of forces in the truss. There are 21 modules with their diagonals in the horizontal plane while there were 32 (almost half) for the simulation with modular cubes, see Figure 9.9.



(a) Model with modular elements and module type 6: Bitmap



(b) Model with modular elements and module type 6: 3D view



(c) Model with modular elements and module type 6: Utilisation of the elements

Figure 9.14: Model with modular elements and module type 6

The convergence graph for this simulation can be seen in Figure 9.15. As indicated by the red crosses a new best value for the fitness function is found 30 times. This indicates that the convergence is quicker for this simulation than it was for the long-run simulation with modular cubes. This one, lasting for more than 200 generations had only 16 improvements.

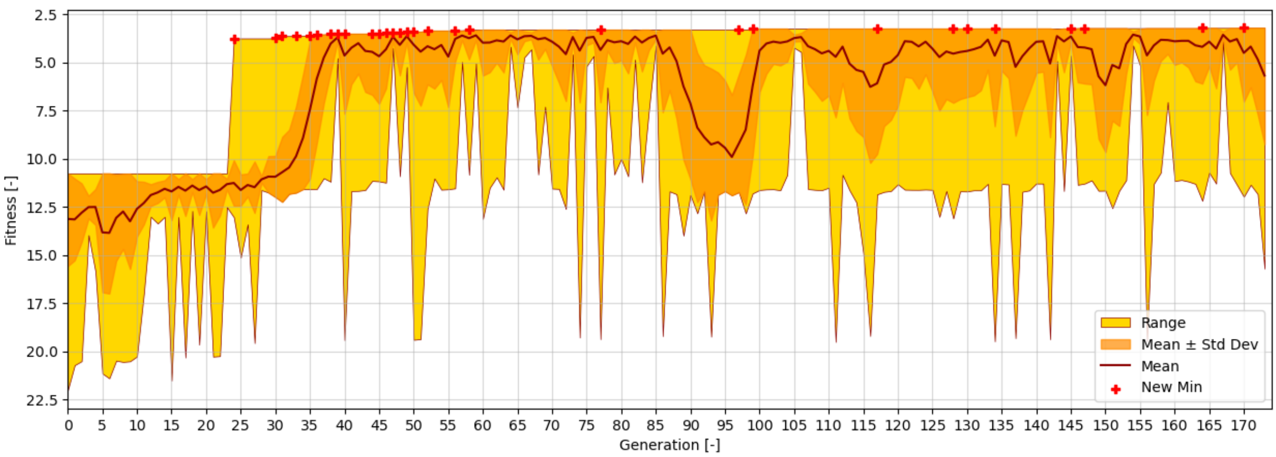
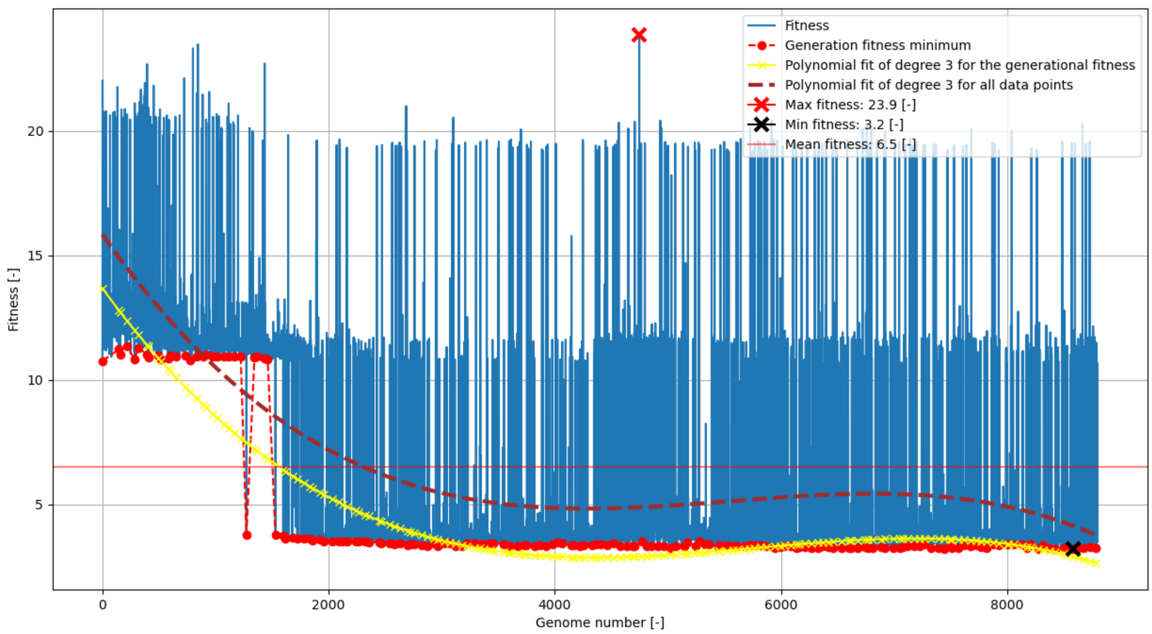
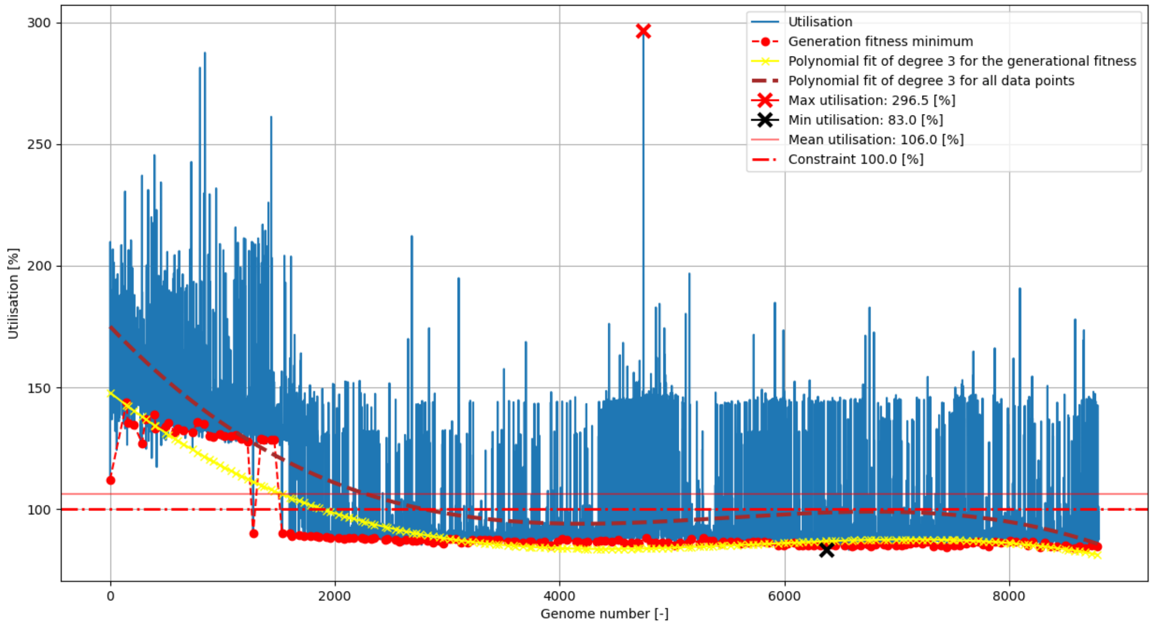


Figure 9.15: Convergence graph for the simulation with modular elements and module type 6

There is a smooth increase in BLF and a decrease in deflection. This is similar to what was seen in other simulations. The extreme utilisation is different. It is well above the threshold value of 100% at the start but after 24 generations there is a jump. This is likely caused by the change in the orientation of the modules located at the supports. These experience the highest stress levels and when their diagonals are orientated in the horizontal plane the forces cannot be directed to the supports which makes the elements fail. Orientating them correctly reduces extreme stress levels significantly. After the jump, the extreme utilisation goes below the threshold of 100% for the generational best fits.



(a) Model with modular elements and module type 6: Fitness per genome number



(b) Model with modular elements and module type 6: Extreme utilisation per genome number

Figure 9.16: Model with modular elements and module type 6: Fitness and extreme utilisation per genome number

9.6 Comparison of the results

There are two basic frames analysed by Irfan et al.. These are the SOS offset and SOS with two-layered grids. This is a widely used system where the top chord grid is offset to the lower chord grid [2]. In this case, the offset is half a grid square. Their deflections and weight can be seen in Table 9.8. The deflection and weight of the truss models and the topologically reconfigurable space frame are also listed in this table. This allows for quick comparison between the different structures. The Pratt trusses have also got the weight of the connections added to their total weight. This is not included in the total weight of the frame structures calculated by Irfan et al. [60]. Without this, the weight of the Pratt truss with modular elements is 52.1 kg/m^2 which corresponds to the weight of the case study frames. The deflection is also in the correct order of magnitude. The increase in weight going from the modular elements model to modular cubes is 64.4% looking at the structure with the weight of the joints included. The increase in weight between the case study SOS truss and the pratt truss with modular cubes is 127.7%. Between the modular elements truss and the case study SOS truss the percentual difference is only 12.2%.

Table 9.8: Vertical deflection and weight of basic frame structures and topologically reconfigurable space frame

	Space frame types	**Vertical deflection [mm]**	**Weight [kg/m^2]**	**Weight without joints [kg/m^2]**
Case study	SOS Offset - Two Grid Layer	55.0	-	27.4
	SOS - Two Grid Layer	61.6	-	46.1
Grasshopper models	Pratt truss with modular cubes	50.1	137.4	105.9
	Pratt truss with modular elements	39.5	83.6	52.1
	Simulation with 1 module type	48.0	138.0	105.9
	Simulation with 2 module types	42.5	143.9	109.4
	Simulation with modular elements	37.9	87.7	55.5

Besides the comparison with the known data from the space frame from Irfan et al. the Grasshopper models can be compared on more aspects. The results can be seen in Table 9.9. It is clear that although the deflection and weight are about the same for the simulations and pratt trusses, this is not the case for the extreme utilisation and the BLF. The BLF is not critical but the extreme utilisation is. It exceeds the threshold of 100% by more than 75% which is very much. The model has taken a whole weekend in the long run and more than 10,000 solutions have been calculated but still no good solution has been found. The fitness function indicates that the found solution is far from a regular good one (Pratt truss).

The simulation with two module types has a much smaller design space due to their unique geometries. The best solution that is possible within this design space is not found. Just like for the other simulations, the extreme utilisation is still too large but a better solution is found by looking at the fitness function. The weight is also much larger than the Pratt truss with modular elements. The simulation that did not use the modular cubes but modular elements had a much better result eventually. This can be explained by the relatively low self-weight of the structure because fewer elements are included in the design. It differs per module in how many diagonals are used which are a significant factor in the total number of elements. As a rule of thumb, the number of elements in structures with modular elements is roughly half that of the modular cubes model. The load on the structure is much lower because of this. It resulted in stresses that were below the threshold of extreme utilisation and this greatly reduced the value of the fitness function.

Table 9.9: Performance metrics of various models for LC1

Performance metric	**Pratt truss with modular cubes**	**Pratt truss with modular elements**	**Simulation with 1 module type**	**Simulation with 1 module type long run**	**Simulation with 2 Module Types**	**Simulation with modular elements**
Weight [t]	28.0	13.8	30.0	28.0	28.9	14.6
Weight with joints [t]	36.3	22.1	36.4	36.4	38.0	23.1
BLF [-]	10.0	10.3	5.2	5.7	5.1	2.4
Displacement [mm]	50.1	39.5	51.7	48.0	42.5	37.9
Extreme utilisation [%]	95.6	69.5	175.2	176.5	149.0	83.9
Mean Utilisation [%]	13.5	15.3	16.5	16.5	12.5	14.0
Percentage of axial deformation energy [%]	94.9	97.9	84.2	78.7	93.9	97.0
Fitness [-]	2.7	0.5	10.8	10.7	9.3	3.2
Nr. Joints [-]	2355	1593	2420	2402	2282	1594
Weight/Area [kg/m^2]	137.4	83.6	137.8	138.0	143.9	87.7
Nr. of elements [-]	1056	531	1056	1056	1072	602

9.7 Conclusion on the case study

Concluding this section, the case study did not yield the expected results. It is possible to make a structure with the described system that fulfills the different constraints. This was checked at the beginning with the pratt truss configuration. However, this structure will be much heavier than if a more regular method is applied where there is no need to place entire cubes directly next to each other. The modular element model which contains no modular cubes but is constructed as a normal truss is much lighter. Furthermore, where the GA could find good solutions, close to optimal, during the verification in Section 7, this was not the case for the case study when the design space was very large. For large structures with many possible solutions, there was some convergence but it was slow. Even after a simulation of two days a solution that fulfilled all the constraints was not found. One simulation was performed with two symmetric module types, namely type 3 and 4. Because of the symmetry, the number of possible solutions was much lower for this simulation. However, an optimal value could not be found by the model. With small alterations of the best solution, a better one was quickly found. This shows that the GA might not be the best possibility for this type of model because of the complexity.

10 Application

Standard trusses for one-way span structures have been optimised for many years. Therefore, It is difficult to make significant topological improvements for these kind of structures with the method described in this thesis. Of course, the streamlining of the design can be improved with the model which helps to be more efficient in the design phase. Besides, the new system allows for a different modular approach to construction. For structures with more irregular designs also the topological advantages of a parametric system can potentially be better leveraged. For that reason, a grid with irregular support conditions is made. It will give insight into which modules are preferred in certain locations.

To keep the number of different possible configurations relatively low the grid of the structure will not be too large. The size of the grid is 2 x 5 and the module size chosen is that of 2 x 2 x 2 m. The structure will thus have a width of 6 m and a length of 10 m. The supports of the structure are placed at the corners again but also at random supports in the middle. These supports are indicated in Figure 10.1 with the green arrows with red dots in the middle. The red dots are the coordinates of the support and the green arrows indicate the lateral constraint in that direction. All supports are pinned so there is rotation possible but in the x-, y-, and z-direction the supports can not move. Besides that there are supports now in the middle, also an overhang is created. This overhang creates bending moment inversions in the frame and it is interesting to see what happens to the distribution of forces.

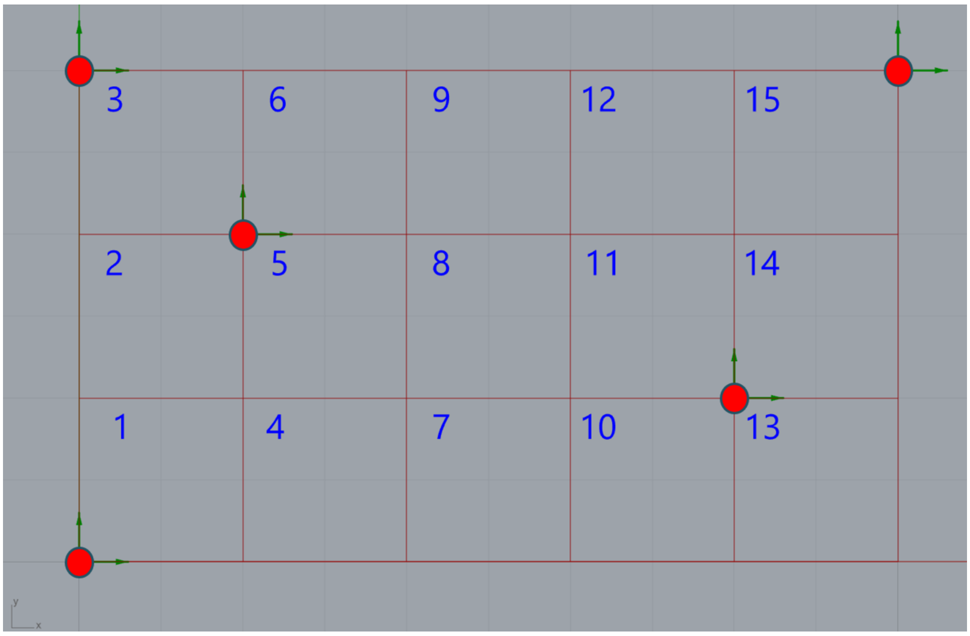
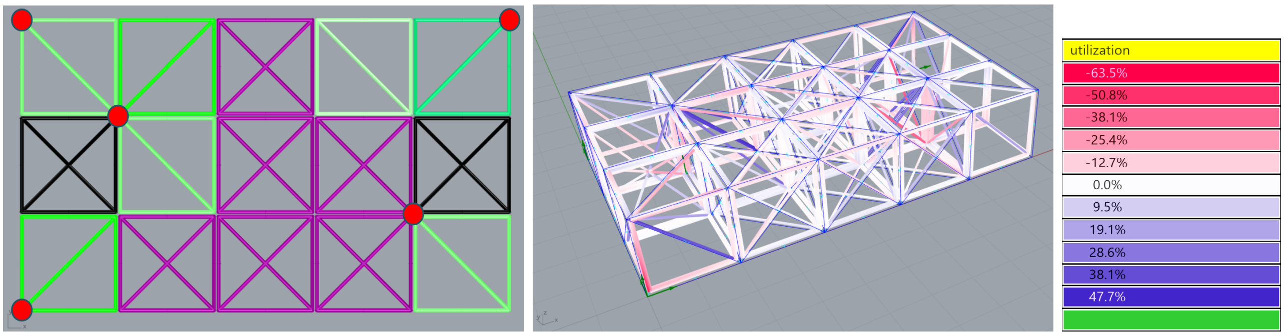


Figure 10.1: Model with irregular supports setup. The red dots indicate the location of the supports. The blue numbers are the location labels.

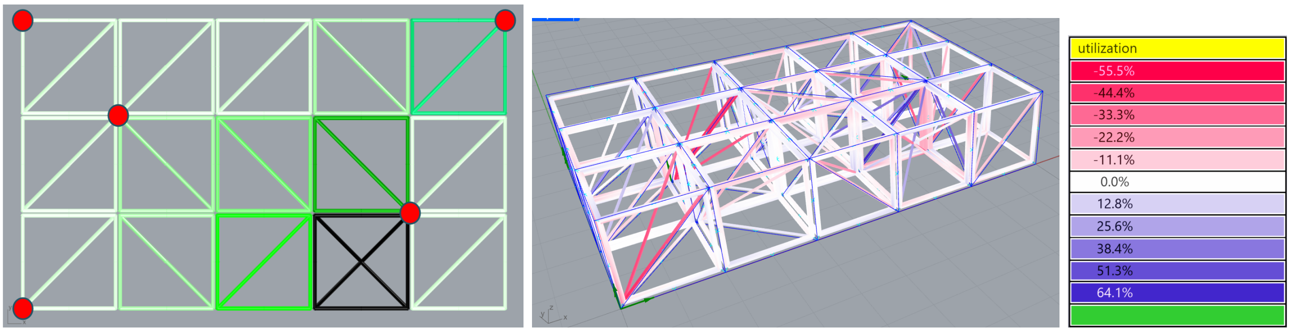
The forces acting on the structure are the same as for the verification structure because of its relatively small size. Only a couple of modules are chosen to be used in this simulation. This is also done to keep the number of possible configurations relatively low. The module types chosen are 2, 3, and 4 which have a total of 10 possible configurations. This results in 10^{15} different possible configurations. This is still not possible to calculate with brute force but is compared to using all modules relatively on the low side. The reason that these modules are chosen is different for each module type. Module type 2 works potentially very well at supports. Furthermore, module type 3 has a completely different layout than the other 2 and they are both lightweight. Module type 4 is stiffer and weighs more than the other two modules. It could be used to stiffen the structure at certain locations where needed. It is much more difficult to find a suitable structure with these three modules based on intuition. A basic truss cannot be made so easily since module type 6 is not selected now. Structural engineering experience is useful here but it might not be enough to find an optimal solution like it was for the verification or case study where standard structures were easily designed. The result of an intuitive structure can be seen in Figure 10.2. The structure is made by varying the orientation of the modules while monitoring what happens with the fitness function and the variables that are in the constraints. In the display of the model, there is an option to show the deflection which is used to determine where stiff modules (type 4) are needed. These are mostly placed in the middle to make sure the structure does not bend too much there. At the supports often module type 2 is used to focus the forces on that position. Module type 3 is used at locations where there is some more stiffness needed than module type 2 can give while keeping the structure lightweight. Module type 4 is much heavier than the other two modules.



(a) Intuitive model with irregular supports: Bitmap **(b)** Intuitive model with irregular supports: 3D view **(c)** Utilisation of the elements

Figure 10.2: Intuitive model with irregular supports. The red dots indicate the location of the supports.

A simulation is performed to find a better solution for this problem. The layout of the final structure for this simulation can be seen in Figure 10.3. Module 2 is used almost everywhere and module 3 only once. Module 4 is not used, probably because it is too heavy and the other constraints are met with the other module types.



(a) Model with irregular supports: Bitmap **(b)** Model with irregular supports: 3D view **(c)** Utilisation of the elements

Figure 10.3: Model with irregular supports after simulation. The red dots indicate the location of the supports.

The convergence graph shows that the best-performing solution is updated many times. Especially in the beginning, this conversion happens. After the 41st generation this stagnates. In the second half of the simulation, only a single structure is found with a better fitness value than any of the solutions found earlier.

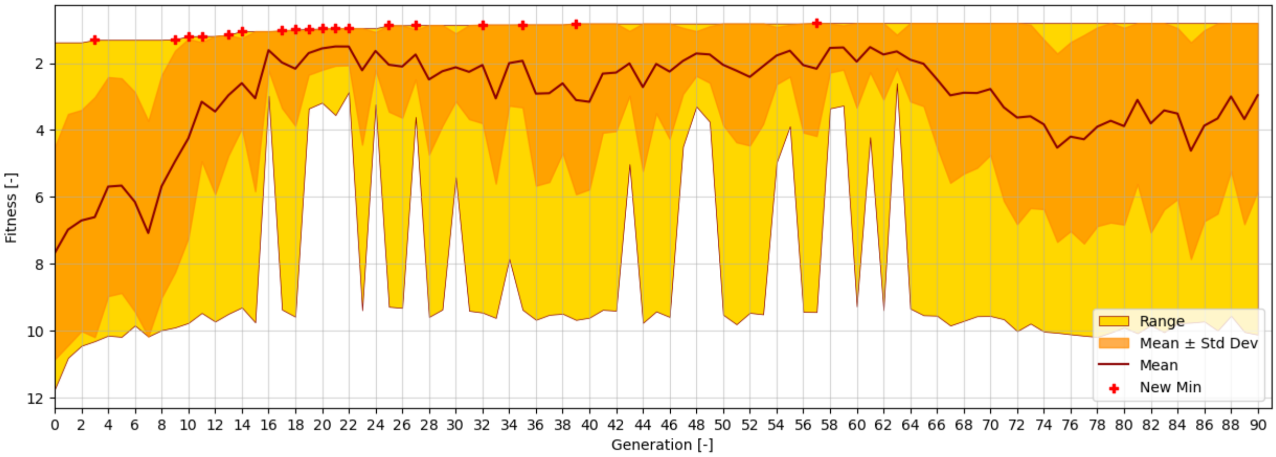


Figure 10.4: Convergence graph for the model with irregular supports

The fitness graph in Figure 10.5 the progress of the slow conversion can also be observed from the generational best solutions. There is a clear downward trend. The spread in fitness values is large for all of the different solutions.

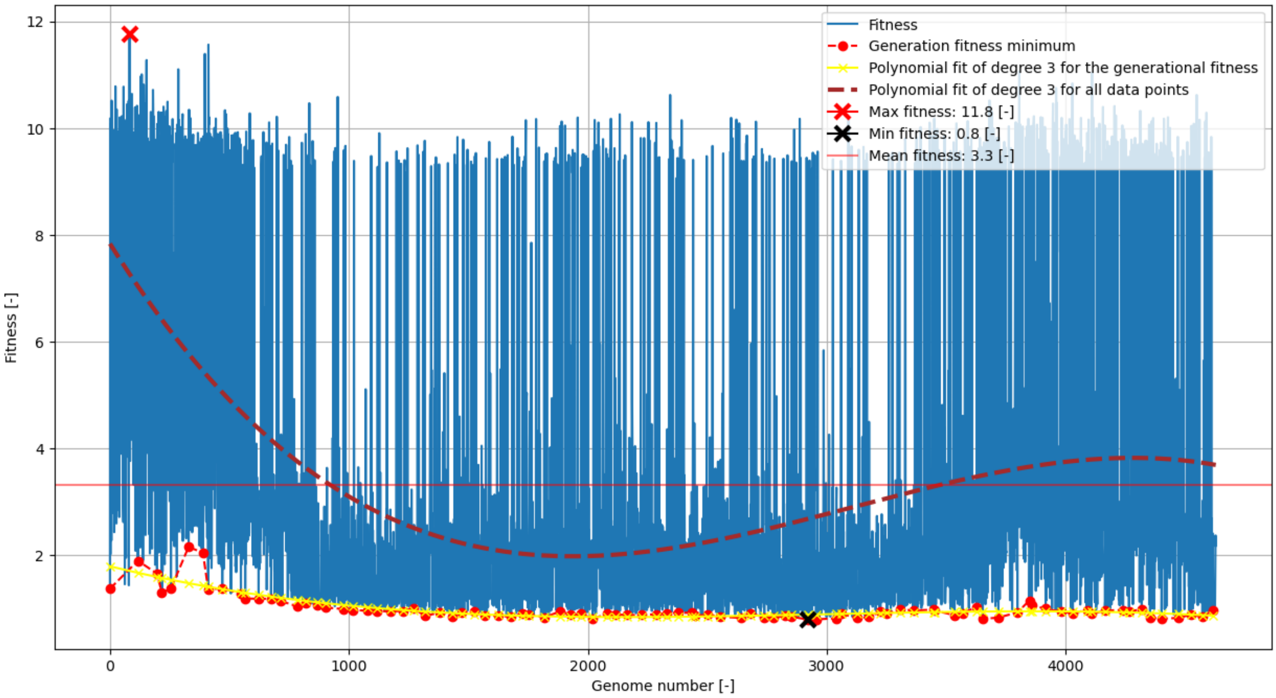


Figure 10.5: Model with irregular supports: Fitness per genome number

In Table 10.1 the performance of both structures can be observed. The simulation found a structure that was slightly better in the end. This has to do with the fact that mostly module type 2 is used in the final output of the simulation which makes the structure much lighter than the intuitive structure. The extreme utilisation is also lower. This probably has to do with the fact that a lower force is acting on the structure due to the 10% lesser weight and the topology is slightly better. The BLF of the intuitive structure is much higher, meaning that it will not buckle as quickly as the structure that came out of the simulation. However, this model also needs more than three times the current load to buckle so it is also safe. Displacements are very comparable.

Table 10.1: Comparison of the simulation and intuitive structure for the model with irregular supports

Performance metric	Simulation	Intuitive structure
Weight [t]	6.4	7.0
Weight with joints [t]	8.3	9.3
BLF [-]	3.1	8.5
Displacement [mm]	6.7	6.2
Extreme Utilisation [%]	65.4	77.1
Mean utilisation [%]	13.8	12.3
Percentage of axial deformation energy [%]	89.3	89.6
Fitness [-]	0.80	0.84
Nr. joints [-]	610	613
Weight/Area [kg/m^2]	138.6	154.7
Nr. of elements [-]	288	240

This example is not exact since the optimal solution is still unknown. The optimal fitness value could be much lower than the two values given in Table 10.1. It might be possible that with some simple changes by hand, the intuitive structure could be further optimised and perform much better than the structure found by the simulation. However, this is not obvious. This application shows that the model can be used when the support conditions are irregular and an ideal structure is not clear from the start. It gives great insight into which modules to use as well since there was a clear preference for one type of module in this structure. Even though module type 4 is much stiffer, module type 2 is preferred everywhere because of its much lower weight and relatively good structural performance. Both intuitive or heuristic methods and computational design can be combined as well. The final result of a simulation could again be tweaked with expert vision and intuition to find an even better solution.

11 Fabrication and construction

The different submodules can be welded together in the factory. This allows for precise welding of the joints and thus for an optimised strength of the pieces. The diagonals can also be welded to the endplates already in the factory. By doing this at the beginning, only bolted connections will have to be made at the construction site.

Regarding the construction of the system, it is a big advantage that all the modules can be quickly assembled and they do not consist of numerous smaller nodes and bars. There are many ways in which the modules can be assembled on-site. The most commonly used techniques according to papers written by Chilton and Lan are the following [2, 10]:

1. Assembly of the modules on a temporary staging or scaffolding [2]. This is a good method when there is no space for large cranes for example. It requires a considerable amount of work on site because of the large amount of scaffolding needed.
2. Assembly of the modules in the air. This can be done from a cantilevering part of the roof or with cranes [2, 10]. Often full scaffolding is used when no cantilevering part is present. It is suitable for space frames with bolted connections [10].
3. Assembly of the modules into larger panels on the ground before connecting them in the air. This then has to be done with cranes [2]. When the space frame consists of several prefabricated units that can be assembled on the ground this is a good method. It is often used for double-layered grids that have no large changes in mechanic behavior when it is divided into smaller parts [10].
4. Assembly of the entire structure on the ground before lifting it into position with cranes [2, 10]. This will increase the quality of the work and also the efficiency [10].
5. Assembly of the entire structure on the ground before jacking or winching it into position [2, 10]. This can be done for double-layered grids with supports along the edge of the structure or at numerous point supports. The jacks are placed at the position of the supports [10].
6. Assembly by sliding elements in the air. At the roof level of a structure, the modules slide into position by rails. The rails are placed on each side of the building and the modules or module strips span between these rails. When they are in position they are assembled or during the sliding. The benefits of this are that other construction work can be carried out simultaneously such that time is saved and also less scaffolding is needed. The technique requires an orthogonal grid system like the one used in the reconfiguration model [10].

A factor that plays a large role in the choice of assembly method is the size of the area available for construction. It is often preferred to lift an entire structure with cranes or jacks. However, this is not always possible. There has to be enough space for the cranes for the structure to be lifted. Usually method 1. is only used when it is not possible to do any of the other methods. This is because, besides space, another important factor is the cost of scaffolding [2]. Another factor is the weight of the modules. Smaller modules can be lifted by smaller cranes that use less space as well. Therefore, the smaller the module size, the smaller the crane but the longer the assembly takes.

All methods are possible for the structures discussed in this thesis. For methods that use large strips or panels of modules such as methods 3 and 6 an extra stability analysis has to be performed on these smaller parts of the structure. It is advised when there is much space available to use method 4 or 5 and otherwise method 3 or 6. Methods 1 and 2 often require a great amount of scaffolding which is not preferred.

Methods 3 and 4 are expected to become quicker because larger parts of the structure can be assembled at once. The modules are much bigger than single beam or column elements. The same goes for method 2 for the assembly of the structure in the air. However, heavier components also require more heavy cranes.

12 Conclusion

This research proposes a systematic approach to the design of space frames. This includes the design of joints and standard modules that form the structure that can be optimised using computational design methods. A parametric model is made in Grasshopper that integrates all different aspects. With the GA Galapagos an optimisation of the topology within a certain design space (Minimisation function, size of the structure, and modules used) can be performed in the model. The eventual goal is to design efficient, circular structures with minimum weight while staying within the constraints of maximum deflection and utilisation. The structure is also not allowed to buckle, which is the third constraint. The research question to be answered is the following:

"What kind of topologically reconfigurable modular system enables the generation of efficient space frames that are suitable for circular construction?"

A final catalog of 6 different modules is proposed. This catalog can be used by the designer or engineer to make structural models. The modules all have cubic shapes which makes it easy to put them together but with different topologies due to a variety of different diagonals. Therefore, they all have different behavior under loading and different weights. Every module type has its unique orientations. The more modules are combined, the larger the design space.

A complete system is considered, meaning that the connections for each module type are also designed. It is decided that the modules should have splice connections and the cubes should be connected with intermodular joints. This increases complexity but improves the stackability of the modules and circular construction. The cubes without splice connections would typically be closed structures and transporting these would result in the transportation of much air and not material. With the use of Eurocode hand calculations, and IDEA StatiCa software the different connections are analysed. The stiffness of the connections partially depends on the moment-to-force ratio (|M|/N) in the joints. These relations are also added to the model to make it more precise. For automated topology optimisation a GA is used. This algorithm minimises the weight of the structure. Besides, with logarithmic barrier functions, it tries to stay within the boundaries of maximum deflection, extreme utilisation, and the minimal value for the BLF.

The model is created in Grasshopper with a variety of plugins such as Karamba3D for FEM analysis of structures. The model is used to make topologically reconfigurable flat space frames. After verification and validation of the model with a couple of benchmark structures and also performing a small application study the following conclusions can be drawn to answer the research question:

- Joint stiffness can be integrated into a parametric design model of topologically reconfigurable steel space frames. This allows for faster design because the joints will not have to be designed at a later stage. The process is more streamlined. The global design of the joints is also already decided so this limits the number of unique connections. Iterative looping over the stiffness of all individual joints can potentially make the model more precise. However, the model is sensitive to these small differences in stiffness and the iterative procedure does not add much to the precision of the model while the computation time increases because of it. The verification of a small model showed that in 42.4% of the solutions, a second iteration is needed and in 5.1% a third iteration is needed. The time per solution by omitting the loop is reduced from 3.7 to 0.4 s. Therefore, it is better to alter the stiffness of the joints per joint group and only update the stiffness once, removing the iterative loop. This decreases computation time and needed memory storage as well.
- The model can find solutions close to optimal when the design space is relatively small. In the verification, a solution very similar to a well performing truss structure was found with just 1 module placed differently. The difference in weight, BLF, displacement, and extreme utilisation were all below 3%. For large models which have a large design space such as roof structures, the model is less useful because it takes a very long time to find good solutions. The validation showed that for a structure with a grid of 6 x 11 a single solution takes 10 s and the design space with the use of only module type 6 is 10^{71}. Brute force on this problem would take $2.2 \cdot 10^{62}$ times the age of the earth. This shows that finding a good solution also requires a very fast convergence. However, the GA which is an evolutionary algorithm takes a long time to converge. New best solutions are only found once in a couple of generations. Using it for a small model and with a limited selection of the catalog is therefore advised.
- For irregular structures that do not have a clear optimal solution the model can be of much help. The application case study showed that it can find solutions suitable for constructions that are close to or even better than intuitive designs looking at the outcome fitness function. Furthermore, a combination of simulation and intuition can potentially produce an even better design.

- The structures that can be designed with the modular reconfiguration model are heavier than normal space frames in general. In comparison with a case study truss the weight is more than double. The case study showed that the percentual difference in weight between a SOS truss and the pratt truss with modular cubes is 127.7%. For this reason in comparison with regular space frames smaller spans can be reached with this model. The initial idea was that because of the increased number of elements smaller cross-sections were needed. Because of the large stresses at the support, smaller cross-sections were not an option. Local strengthening of elements at the support is needed for large span structures such as the one in Section 9. In this section, an increase in the steel strength of the supports from S235 to S355 was needed. Other measures such as adding steel reinforcing plates or choosing a different cross-section near the supports would also be possible.
- The model can also be used to design structures closer to regular space frames by focusing on modular elements instead of modular cubes. Duplicate beams are dropped resulting in standard structures. This requires less material and the structure is more lightweight. More optimal solutions can be reached with this type of structure in terms of structural performance. For a Pratt truss, a decrease of 64.4% of the weight can be realised. In comparison to a case study frame the modular elements model is only 12.2% heavier. The extreme utilisation of the elements for the Pratt truss is also 26% less. The deflection of a modular cubes Pratt truss is 50.1 mm and for a modular elements Pratt truss 39.5 so lower by 21.2%.

Summarising, the final model can be used to design flat steel space frames that are suitable for circular construction. With the computational method, it takes long to find a solution that can be used in practice for large structures with a large design space. The structural efficiency of the system is lower than for conventional space frames because it is heavier which also leads to larger internal stresses. A switch can be made with the model towards the model with "modular elements" instead of "modular cubes". This is a more lightweight version of the system where duplicate beams are not used anymore and the idea of "Modular cubes" is not used. Structures made with this model are more similar to conventional space frames. This extension of the model could potentially be used in practice if the convergence towards a solution can be improved and the speed at which solutions are calculated. The great advantage of this system lies in its circularity and quick construction. The system allows the construction of flat space frames with fewer design steps, and fewer on-site construction work, and also allows reusing the modules at the end-of-life stage of a structure. The structure can easily be disassembled into modules or module parts if needed because of the bolted connections. Modules can be changed to a different module type by disassembling or assembling diagonals to the main frame allowing for flexibility in the system as well. This research contributes to the development of practical space frames for circular and modular designs that can be used around the world.

13 Discussion

There are multiple things that could have been done differently during this research. There are a couple of important assumptions and simplifications for the settings of the model. Furthermore, the GA has some drawbacks that need to be clear to the user of the model. The limitations of the model also have to be clarified. Just like in the conclusion, an overview is created using bullet points.

13.1 Model assumptions and simplifications

- In the stiffness calculations, the intermodular joint is not analysed as a whole but is split in different parts. Each joint consists of multiple nodes and connections. Where modules are connected, intra- and intermodular connections form a complex joint. Idea StatiCa can only analyse a joint correctly if a single node is present in the model. For this reason, the simplified parts are analysed separately. This analysis results in values for the stiffness of the joints where the presence of various other members is disregarded. These values are therefore less precise. The calculations performed with the component method are also done on simplified versions of the joint. These calculations lead to in general lower values for the stiffness of the joints which are therefore conservative. Lower stiffness of the joints might lead to larger deflection under loading.
- The stiffness values are assigned per joint group instead of for individual joints. Besides, the stiffness is only updated once depending on the present |M|/N ratio. The verification showed that the time per solution was reduced by 89.2% in comparison with the model where a stiffness loop was included. In many cases, this removal of the stiffness loop is close to reality but it does not always give precise results. In 42.4% of the solutions, a second iteration is needed and in just 5.1% a third. The advantage of faster computation however more important than higher precision of the stiffness of the joints.
- The weight of the joints has only been taken into account for the fitness function. The additional force on the structure that is caused by the weight of the connections is neglected in strength calculations. Therefore, the calculated stresses in the members are different in reality. This also influences the buckling behavior of the structure. The difference can be large since the weight of the joints is on average 32% of the weight of the structure. for the different modules. The designer would have to check the outcome of the model with the additional weight of the joints in the end. Otherwise, deflections and stresses are not conservative.

13.2 Model limitations

- The model is only able to design planar structures. Adding curvatures would make the number of applications of the model much larger but would also increase complexity. Camber, which is only a small curvature of the structure is also not taken into account. This is normally done for the rainwater run-off of roofs [2]. The implementation of camber will be much easier than implementing curves that are used for free-form structures.
- The model can now only be used to make square on square double layered space frames. Only cubic modules are now part of the catalog. Other dimensions of the space frame modules could have been used. In structures around the world, different grid configurations are used. Triangle on hexagon or triangle on triangle are examples grids [2]. This would increase the number of applications of the model. A drawback would be that new joints would be designed and the models' complexity would increase. Furthermore, the size of the modules is the same in the x-, y-, and z-direction. Most of the systems worldwide are not as regular as the system described here. For instance, the CUBIC system which has been used as inspiration for the design has edges with slightly different lengths. The module size used for building a maintenance hangar at Stansted Airport was 2.0 x 3.5 with a depth of 4.0 m [57]. However, this would not allow for rotations of the modules in all directions. For tetrahedra, this would have been possible. The use of a triangular grid could come with some advantages such as better redistribution of forces and better stability. Moreover, it could lead to a more flexible design because free-form shapes are easier to make with tetrahedra. Curves and irregular surfaces can easier be followed with these modules.
- Due to the eccentricity implemented in the model, there is an overestimation of the lengths of the elements used in the model. The length of the beam elements in the model is equal to the original length of the line that goes from one node to another. However, there is an eccentricity of the elements of 8 mm towards the center of the modules. Beams are twice this eccentricity shorter so 16 mm. This leads to a slightly wrong calculation which can also be seen in the display of the model.

13.3 Genetic Algorithm drawbacks

- The optimisation method utilises a GA which can provide the user with good results but also has its flaws. The first problem is that the algorithm is slow. When the analysed structure becomes as large as the model used for validation for example it can take 10 s or more per solution to calcualte. When 50 generations are needed consisting of 50 individual solutions the computation time will be a day or even more. As seen in the simulations even after more than 50 iterations an optimal solution may not be reached because of the size of the design space. Besides that the iterations are slow, the convergence is as well. The model shows improvements in the fittest solution during the simulations but the increase is limited. One run showed an improvement of just 5.5% after 50 generations. For the long run of over 205 generations, only an improvement of 1.0% was observed.
- The model can not tell if there is a solution that fulfills all the different constraints. The algorithm converges towards a certain value but feasibility may never be reached. The user of the model must take this into account when running the model. After putting the algorithm to a standstill the different constraints will need to be checked again manually to see which ones are met and which ones aren't.

14 Recommendations

There are a couple of recommendations that result from the conclusion and discussion. These sections showed that the model is limited and there are many possible improvements. It must be kept in mind, that every change might bring more complexity to the model and the system. The recommendations are listed below. There is a subdivision in model improvements, an extension of the analysis and the use of the model.

14.1 Model improvements

- The model is limited to specific structure types. Only flat space frames with rectangular-sized structures can be generated. Besides this, the model only uses a square on square double layered grid structure. Other mesh types as described in Section 3.2 could be used for the model such as a triangular mesh. Instead of using modules with uniform dimensions in every direction also different depths could be used. This would lead to a different load distribution in the system. A larger depth for the same grid could lead to better overall resistance to bending moments in the structure.
- The stiffness of the joints has been determined using the IDEA StatiCa software and the component method. The component method has been used to verify the results obtained from IDEA StatiCa. However, the values of both methods were of different orders of magnitude. This was mainly because for the rotational stiffness IDEA StatiCa did not give feasible results, as explained in Appendix D. Besides, simplifications in the joint calculations have been made to be able to assess the intermodular joints. The behaviour is thus not very precisely modelled. To determine the real value of the stiffness values of the joints and also other joint behaviour it is strongly advised to perform lab tests on steel prototypes. Incremental load tests can be used to determine the translational and rotational stiffness of the joints.
- To streamline the workflow it is recommended to integrate Idea Statica with the Grasshopper model. There is an Idea Statica plugin for Grasshopper and Rhino available that makes analysis of the joints in the model quicker and easier. It would make the iterative loop for the stiffness of the joints redundant, decreasing the computation time of the model. The problem however is that this software is relatively new and still under development. It can be used already but it is difficult to integrate it into this project. However, for future research, it is advisable to use it because the components of the joints can also be parametrised. For different module depths, the stiffness values of the joints can then be computed as well.

  Momentarily the joints are only analysed for elements used in modules with a depth of 2.0 m. The stiffness of the joints is assumed to be invariant of the size of the elements. If the module depth increases also larger elements are used in the model. The same joint type is used but the steel components where it consists of are larger. It should be tested if the stiffness values for larger joints change. Besides IDEA StatiCa calculations this can additionally be done with lab tests. To integrate all these aspects into the model, the set-up in Grasshopper will have to be changed. At the moment the model is made with the use of beam offsets. This has the drawback that for the intermodular joint all elements are connected to a single node. In reality, there are multiple nodes in this single joint. There are intramodular corner nodes and intermodular nodes that connect different modules with bolted endplates. In the revised model these nodes will also have to be modelled. This will drop the need to introduce an element offset. That will also make integration of Idea Statica in these nodes easier. Stiffness values can then be applied to different parts of the intermodular joint which is more realistic.
- There is the possibility to run the model for "modular elements". This is a more efficient way of distributing material and does not follow the principle of interconnecting cubes. Also, the structural performance of this system is better. Different joints are therefore needed. Currently, the model uses joint stiffness values for the "Modular cubes" system. A new joint analysis has to be performed on the type of joints to use in this other system and what their stiffness is under varying load scenarios.

  For the types of structures that can be made with this model optimisation based on a ground structure could be used as well. This is a way of optimising where the beams have continuous values that describe the beam cross-section. For a certain design problem, the members of the frame all receive a cross-section ranging from quasi-null to a maximum The minimisation of the structure volume within certain constraints can be performed with for instance MILP which is a discrete approach [63]. However, MILP may struggle to handle the continuous variation of beam cross-sections unless discretized. Furthermore, this method does not allow for modularity or reconfigurability of modules.
- The weight of the joints must be added to the model for the FEM calculation. For simplicity, it is momentarily added after the structural analysis as input for the weight of the fitness function. Adding

it to the FEM analysis would increase model complexity but also would give a more precise outcome on the deflection and stresses of a structure. All are eventually used to determine the value of the fitness function.

- There are many options for software and algorithms for optimisation available. A GA like Galapagos works but is very slow. Other algorithms might be faster such as Opossum. This plugin for Grasshopper uses Radial Basis Function Optimisation (RBFOpt), which is a global optimisation method that uses machine learning techniques and converges faster than evolutionary solvers. It also has an evolutionary solver called Covariance Matrix Adaptation Evolution Strategy (CMAES) which can be used for comparison. Opossum can be used to solve single- and multi-objective functions. It is also able to handle complex problems efficiently [24]. These solutions are often less good than the final solutions found by a GA. Additionally, machine learning could be integrated into the model. After training this could speed up the process of finding optimal solutions.
- The model simulates structures in 3D. To speed up the model it could be transformed back to a 2D tiling problem. This is done in other research as well such as in the research performed by Tyburec et al. or research by Tugilimana et al. [8, 9]. This simplification will lead to less accuracy since the real structure is in 3D but simplifications are sometimes needed to make a model faster and more useful.

14.2 Model analysis extension

- The efficiency of the structures in terms of circular economy or feasibility is not measured. A cost analysis of the used system has not been made. This research has mainly been on the structural feasibility and application. Since the joints are of a new type it is important to know how much a joint costs in comparison to conventional ones. Besides, the effect of the use of splices on the entire cost is also something that can be useful to know. These different aspects determine if a system like this can eventually be implemented in projects.
- The model has not been tested on dynamic load situations. It would be interesting to see how the structure reacts to earthquakes. Research on the seismic performance of IMCs has not been widely performed [41] but the performance of space frames has. A comparison between the performances of a regular spaceframe and this model could lead to interesting results. Especially for countries where these dynamic loads often occur. Besides, wind loads would be interesting for similar reasons. Horizontal loads can have a large effect on how the model places certain modules. In windy regions it could be useful for design.
- Another thing that could be done to improve the model is to add a logarithmic barrier function to minimise the number of joints used. This could lead to a more efficient structure in terms of construction. Less time is needed to join all components when there are fewer joints.
- Furthermore, other factors that are important to take into account in design and have not been investigated are the financial costs and embodied carbon in the structure. As Figure 3.1 clearly describes these factors influence the circular economy of the design and the structural efficiency. These are both under investigation in this thesis and therefore are a good addition to the model.

14.3 Use of the model

- It is recommended for the use of the model to always run a couple of simulations to check the validity of the model. It was found that even for small-scale models, when using numerous different modules the optimal solution may not be reached. To come close to this solution the best genomes of a simulation may be reused in the next one for optimisation. Using expert judgement it is recommended to make a structure based on heuristics first. Then, it can be checked if this structure can be improved with the model.

15 Bibliography

[1] Musatafa Abbas Albadr, Sabrina Tiun, Masri Ayob, and Fahad Al-Dhief. Genetic Algorithm Based on Natural Selection Theory for Optimization Problems. *Symmetry 2020, Vol. 12, Page 1758*, 12(11):1758, 10 2020. ISSN 2073-8994. doi: 10.3390/SYM12111758. URL `https://www.mdpi.com/2073-8994/12/11/1758/htmhttps://www.mdpi.com/2073-8994/12/11/1758`.

[2] John Chilton. *Space grid structures.*, volume 1. Architectural Press, Oxford, 2000. ISBN 0750632755.

[3] Yaozhi Luo and Yu Xue. Recent development and engineering practices of space grid structures in China. *International Journal of Space Structures*, 3 2024. ISSN 20598033. doi: 10.1177/09560599241228934/FORMAT/EPUB. URL `https://doi.org/10.1177/09560599241228934`.

[4] Jaime Sanchez-Alvarez and Herbert Klimke. The Mero Legacy: Two engineers recall ... Technical report, 1 2020. URL `https://www.academia.edu/41974404/The_Mero_Legacy_Two_engineers_recall_`.

[5] Antiopi Koronaki, Paul Shepherd, and Mark Evernden. Geometry optimization of space frame structures for joint modularity. Technical report, MIT, Boston, 7 2018.

[6] G. S. Ramaswamy, Mick Eekhout, and G. R. Suresh. *Analysis, design and construction of steel space frames*. Thomas Telford, London :, 2002. ISBN 0727730142. URL `https://tudelft.on.worldcat.org/search/detail/49240236?queryString=ti%3D%20%22Analysis%2C%20design%20and%20construction%20of%20steel%20space%20frames%20%22&clusterResults=false&groupVariantRecords=false`.

[7] Yijiang Huang, Latifa Alkhayat, Catherine De Wolf, and Caitlin Mueller. Algorithmic circular design with reused structural elements: Method and Tool. *Conceptual Design of Structures*, 9 2021. doi: 10.3929/ethz-b-000515183. URL `https://doi.org/10.3929/ethz-b-000515183`.

[8] Marek Tyburec, Jan Zeman, Martin Doškář, Martin Kružík, and Matěj Lepš. Modular-topology optimization with Wang tilings: an application to truss structures. *Structural and Multidisciplinary Optimization*, 63(3):1099–1117, 3 2021. ISSN 16151488. doi: 10.1007/S00158-020-02744-8/FIGURES/16. URL `https://link.springer.com/article/10.1007/s00158-020-02744-8`.

[9] Alexis Tugilimana, Ashley P. Thrall, Benoît Descamps, and Rajan Filomeno Coelho. Spatial orientation and topology optimization of modular trusses. *Structural and Multidisciplinary Optimization*, 55(2):459–476, 2 2017. ISSN 16151488. doi: 10.1007/S00158-016-1501-7/FIGURES/14. URL `https://link.springer.com/article/10.1007/s00158-016-1501-7`.

[10] Tien T. Lan. Space Frame Structures. Technical report, Institute of Building Structures, Chinese Academy of Building Research, Beijing, 2005.

[11] Robin Oval. Topology Finding of Patterns for Structural Design. Technical report, École des Ponts ParisTech, Paris, 12 2019.

[12] Clemens Preisinger. Linking structure and parametric geometry. *Architectural Design*, 83(2):110–113, 3 2013. ISSN 00038504. doi: 10.1002/AD.1564.

[13] David Rutten. Galapagos: On the logic and limitations of generic solvers. *Architectural Design*, 83(2):132–135, 3 2013. ISSN 00038504. doi: 10.1002/AD.1568.

[14] Idea Statica. Structural design of steel connections, 2024. URL `https://www.ideastatica.com/support-center/general-theoretical-background`.

[15] Jan Brütting, Patrick Ole Ohlbrock, Julian Hofer, and Pierluigi D'Acunto. Stock-constrained truss design exploration through combinatorial equilibrium modeling. *International Journal of Space Structures*, 36(4):253 – 269, 12 2021. ISSN 20598033. doi: 10.1177/09560599211064100.

[16] Jan Brütting, Camille Vandervaeren, Gennaro Senatore, Niels De Temmerman, and Corentin Fivet. Environmental impact minimization of reticular structures made of reused and new elements through Life Cycle Assessment and Mixed-Integer Linear Programming. *Energy and Buildings*, 215, 5 2020. ISSN 03787788. doi: 10.1016/J.ENBUILD.2020.109827.

[17] Albertine Van Marcke, Vittoria Laghi, and Josephine Voigt Carstensen. Automated planar truss design with reclaimed partially disassembled steel truss components. *Journal of Building Engineering*, 84:108458, 5 2024. ISSN 2352-7102. doi: 10.1016/J.JOBE.2024.108458.

[18] Jan Brütting, Joseph Desruelle, Gennaro Senatore, and Corentin Fivet. Design of Truss Structures Through Reuse. *Structures*, 18:128–137, 4 2019. ISSN 23520124. doi: 10.1016/j.istruc.2018.11.006.

[19] Jan Brütting, Gennaro Senatore, and Corentin Fivet. Form Follows Availability - Designing Structures Through Reuse. *Journal of the International Association for Shell and Spatial Structures*, 60(4):257–265, 12 2019. ISSN 19969015. doi: 10.20898/J.IASS.2019.202.033.

[20] Roberto Minunno, Timothy O'Grady, Gregory M. Morrison, Richard L. Gruner, and Michael Colling. Strategies for Applying the Circular Economy to Prefabricated Buildings. *Buildings 2018, Vol. 8, Page 125*, 8(9):125, 9 2018. ISSN 2075-5309. doi: 10.3390/BUILDINGS8090125. URL `https://www.mdpi.com/2075-5309/8/9/125/htmhttps://www.mdpi.com/2075-5309/8/9/125`.

[21] Steinar Hillersøy Dyvik, Sverre Magnus Haakonsen, Marcin Luczkowski, and Yosuke Komiyama. Aluminium nodes for timber gridshells: Parametric detailing of node principles. *International Journal of Space Structures*, 12 2023. ISSN 20598033. doi: 10.1177/09560599231180155/ASSET/IMAGES/LARGE/10.1177{_}09560599231180155-FIG20.JPEG. URL `https://journals.sagepub.com/doi/full/10.1177/09560599231180155`.

[22] G. M. Gasii. Structural and Design Specifics of Space Grid Systems. *Science & Technique*, 16(6):475–484, 11 2017. ISSN 2227-1031. doi: 10.21122/2227-1031-2017-16-6-475-484.

[23] Marie Paschke, Stefan Neuhäuser, Klaas De Rycke, and Christoph Gengnagel. Digital Prototyping – Modelling Concepts for a Generative Computer Based-Design of Complex Tensile Structures. *Impact: Design With All Senses*, pages 159–172, 2020. doi: 10.1007/978-3-030-29829-6{_}13.

[24] Robert Vierlinger. *Octopus Generative Design in Architectural Engineering.* PhD thesis, University of Applied Arts Vienna, Vienna, 1 2022.

[25] Wolfgang E. Lorenz. Loops in Grasshopper. In *Digitale Architektur und Raumplanung*, volume 1, chapter 3, pages 15–24. Technische Universität Wien, Vienna, 2019. ISBN 9783950446418.

[26] Diego Apellániz and Robert Vierlinger. Enhancing structural design with a parametric FEM toolbox. *Steel Construction*, 15:188–195, 11 2022. doi: 10.1002/stco.202200004.

[27] Paul Shepherd and Will Pearson. Topology Optimization of Algorithmically Generated Space Frames. *Proceedings of the IABSE-IASS Symposium 2013 - Beyond the limits of Man*, page 356, 2013.

[28] R. Oval, R. Mesnil, T. Van Mele, P. Block, and O. Baverel. A vector encoding for topology finding of structured quad-based patterns for surface structures. *International Journal of Space Structures*, 11 2023. ISSN 20598033. doi: 10.1177/09560599231207650/ASSET/IMAGES/LARGE/10.1177{_}09560599231207650-FIG20.JPEG. URL `https://journals.sagepub.com/doi/full/10.1177/09560599231207650`.

[29] Linwei He, Matthew Gilbert, and Xingyi Song. A Python script for adaptive layout optimization of trusses. *Structural and Multidisciplinary Optimization*, 60(2):835–847, 8 2019. ISSN 16151488. doi: 10.1007/S00158-019-02226-6/FIGURES/10. URL `https://link.springer.com/article/10.1007/s00158-019-02226-6`.

[30] Hans Schober. *Transparent Shells.* Wiley, 2015. ISBN 9783433606001. URL `https://tudelft.on.worldcat.org/search/detail/935251727?queryString=transparent%20shells`.

[31] Soeren Stephan, Jaime Sanchez-Alvarez, and Klaus Knebel. Reticulated Structures on Free-Form Surfaces. Technical report, MERO GmbH & Co, Würzburg, 2004. URL `https://www.researchgate.net/publication/228563882`.

[32] Juraj Salitrežić. Influence of Initial Mesh Topology on the Optimal Structural Design of Steel Space Frame Structures. Technical report, TU Delft, Delft, 2022. URL `http://repository.tudelft.nl/`.

[33] Romain Mesnil, Cyril Douthe, Olivier Baverel, and Bruno Léger. Generalised cyclidic nets for shape modelling in architecture. *International Journal of Architectural Computing*, 15:148–168, 11 2017. doi: 10.1177/1478077117714917.

[34] Robin Oval, Romain Mesnil, Tom Mele, Philippe Block, and Olivier Baverel. Rule-based topology finding of patterns towards multi-objective structural design. In *Proceedings of the IASS Annual Symposium*, Barcelona, 8 2019.

[35] Romain Mesnil, Cyril Douthe, and Olivier Baverel. Non-standard patterns for gridshells: fabrication and structural optimization. *Journal of the International Association for Shell and Spatial Structures*, 4:277–286, 11 2017. doi: 10.20898/j.iass.2017.194.893. URL `https://www.researchgate.net/publication/321429756_Non-standard_patterns_for_gridshells_fabrication_and_structural_optimization#fullTextFileContent`.

[36] M. Setareh, B. Jones, L. Ma, F. Bacim, and N. Polys. Application and Evaluation of Double-Layer Grid Spatial Structures for the Engineering Education of Architects. *Journal of Architectural Engineering*, 21(3):04015005, 9 2015. ISSN 1076-0431. doi: 10.1061/(ASCE)AE.1943-5568.0000179/ASSET/5DD879CA-D9AF-4956-B0C7-48E4F58068D1/ASSETS/IMAGES/LARGE/FIGURE7.JPG. URL `https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29AE.1943-5568.0000179https://ascelibrary.org/doi/10.1061/%28ASCE%29AE.1943-5568.0000179`.

[37] Ioannis Mirtsopoulos. Additive manufacturing of algorithmically form generated nodes for double layer free form structures. Technical report, Technical University of Delft, Delft, 1 2015.

[38] Z. S. Makowski. *Analysis, design and construction of double-layer grids*. Applied Science Publishers, London, 1981. ISBN 085334910X, 9780853349105.

[39] Joop M. Gerrits. An Architectonic Approach Of Choosing A Space Frame System. Technical report, Delft University of Technology, Delft, 1998.

[40] Steel Beam - Open-Web Joist Dimensions & Drawings — Dimensions.com, 2024. URL `https://www.dimensions.com/element/steel-beam-open-web-joist`.

[41] En Feng Deng, Liang Zong, Yang Ding, Xiao Meng Dai, Ni Lou, and Yang Chen. Monotonic and cyclic response of bolted connections with welded cover plate for modular steel construction. *Engineering Structures*, 167:407–419, 7 2018. ISSN 0141-0296. doi: 10.1016/J.ENGSTRUCT.2018.04.028.

[42] European Committee for Standardization (CEN). Eurocode 1: Actions on structures Part 1-1: General actions: Densities, self-weight, imposed loads for buildings. Technical report, European Committee for Standardization (CEN), Brussels, 2002.

[43] European Committee for Standardization (CEN). Eurocode 1: Actions on structures Part 1-3: General actions - Snow loads. Technical report, European Committee for Standardization (CEN), Brussels, 7 2003.

[44] European Committee for Standardization (CEN). Eurocode 0: Basis of structural design. Technical report, European Committee for Standardization (CEN), Brussels, 12 2002.

[45] European Committee for Standardization (CEN). National Annex to NEN-EN 1990+A1:2006+A1:2006/C2:2019 Eurocode: Basis of structural design. Technical report, Koninklijk Nederlands Normalisatie-instituut, Delft, 11 2019.

[46] M. Farajian, P. Sharafi, H. Eslamnia, K. Kildashti, and Y. Bai. Classification of inter-modular connections for stiffness and strength in sway corner-supported steel modular frames. *Journal of Constructional Steel Research*, 197:107458, 10 2022. ISSN 0143-974X. doi: 10.1016/J.JCSR.2022.107458.

[47] Florentia Kavoura. CIEM5000 SE Base Module-Sustainable Construction Members & Systems. Lecture 2.6.2 - Introduction to joints with steel sections I, 2022.

[48] Xavier Tellier, Romane Boutillier, Cyril Douthe, and Olivier Baverel. Patterns for gridshells with negligible geometrical torsion at nodes. *Curved and Layered Structures*, 8(1):147–156, 1 2021. ISSN 23537396. doi: 10.1515/CLS-2021-0014. URL `https://hal.science/hal-04309274https://hal.science/hal-04309274/document`.

[49] Andrew William Lacey, Wensu Chen, Hong Hao, and Kaiming Bi. Review of bolted inter-module connections in modular steel buildings. *Journal of Building Engineering*, 23:207–219, 5 2019. ISSN 2352-7102. doi: 10.1016/J.JOBE.2019.01.035.

[50] Sriskanthan Srisangeerthanan, M. Javad Hashemi, Pathmanathan Rajeev, Emad Gad, and Saman Fernando. Review of performance requirements for inter-module connections in multi-story modular buildings, 3 2020. ISSN 23527102.

[51] Xiao Zhou Han, Chen Wang, Xiao Huang Can He, Yi Fei Hu, Tak Ming Chan, and Kwok Fai Chung. Experimental investigation on bolted inter-module connections and subassemblies of steel modular frames. *Journal of Constructional Steel Research*, 211:108118, 12 2023. ISSN 0143-974X. doi: 10.1016/J.JCSR.2023.108118.

[52] Xiao Huang Can He and Tak Ming Chan. Classification system for inter- and intra-module joints in non-sway steel MiC structures. *Journal of Constructional Steel Research*, 206, 7 2023. ISSN 0143974X. doi: 10.1016/j.jcsr.2023.107913.

[53] Gohar Nadeem, Nor Azizi Safiee, Nabilah Abu Bakar, Izian Abd Karim, and Noor Azline Mohd Nasir. Connection design in modular steel construction: A review. *Structures*, 33:3239–3256, 10 2021. ISSN 2352-0124. doi: 10.1016/J.ISTRUC.2021.06.060.

[54] Haibo Yang. Performance analysis of semi-rigid connections in prefabricated high-rise steel structures. *Structures*, 28:837–846, 12 2020. ISSN 2352-0124. doi: 10.1016/J.ISTRUC.2020.09.036.

[55] Idea Statica. Stiffness analysis, 2024. URL `https://www.ideastatica.com/support-center/stiffness-analysis`.

[56] European Committee for Standardization (CEN). Eurocode 3: Design of steel structures Part 1-8: Design of joints. Technical report, European Committee for Standardization (CEN), Brussels, 2005.

[57] J.C. Chilton and L.A. Kubik. Design and Construction of the CUBIC Space Frame Roof, Maintenance Hangar, Stansted Airport. In Ture Wester, Stefan J. Mewadowski, and IB Mogensen, editors, *Proceedings of the international IASS symposium*, Copenhagen, 9 1991. unstakademiets Forlag Arkitektskolen. URL `https://www.researchgate.net/publication/321490172`.

[58] Friso van Spengler. *Parametric design of grid shell structures A design method considering semi-rigid joints and stiffness based connection design.* PhD thesis, Delft University of Technology, Delft, 7 2024.

[59] Hoel Kervadec, Jose Dolz, Jing Yuan, Christian Desrosiers, Eric Granger, and Ismail Ben Ayed. Constrained Deep Networks: Lagrangian Optimization via Log-Barrier Extensions. *30th European Signal Processing Conference (EUSIPCO)*, pages 962–966, 4 2019. doi: 10.23919/EUSIPCO55093.2022.9909927. URL `http://arxiv.org/abs/1904.04205http://dx.doi.org/10.23919/EUSIPCO55093.2022.9909927`.

[60] Danish Irfan, S. Varadharajan, Shahina Mateen, Ashish Kumar, and Bishnu Kant Shukla. Assessment of Behavior of Steel Space Frames With Two-Way Orthogonal Latticed Grids and Orthogonal Square Pyramid Space Grids. *AIP Conference Proceedings*, 2800(1), 9 2023. ISSN 15517616. doi: 10.1063/5.0163368.

[61] Bureau of Indian standards. Design loads (other than earthquake) for buildings and structures - Code of practice - Part 3 Wind loads. Technical report, Bureau of Indian standards, New Delhi, 4 2015. URL `www.bis.org.inwww.standardsbis.in`.

[62] S. R. Satish Kumar and A. R. Santha Kumar. Design of Steel Structures. Technical report, Indian Institute of Technology Madras, Chennai, 2014.

[63] Jan Brütting, Gennaro Senatore, and Corentin Fivet. MILP-based discrete sizing and topology optimization of truss structures: new formulation and benchmarking. *Structural and Multidisciplinary Optimization*, 65 (10), 10 2022. ISSN 16151488. doi: 10.1007/s00158-022-03325-7.

[64] Kyung-Suk Choi, Ho-Chan Lee, and Hyung-Joon Kim. Influence of Analytical Models on the Seismic Response of Modular Structures. *Journal of the Korea institute for structural maintenance and inspection*, 20(2):74–85, 3 2016. ISSN 2234-6937. doi: 10.11112/JKSMI.2016.20.2.074. URL `http://dx.`.

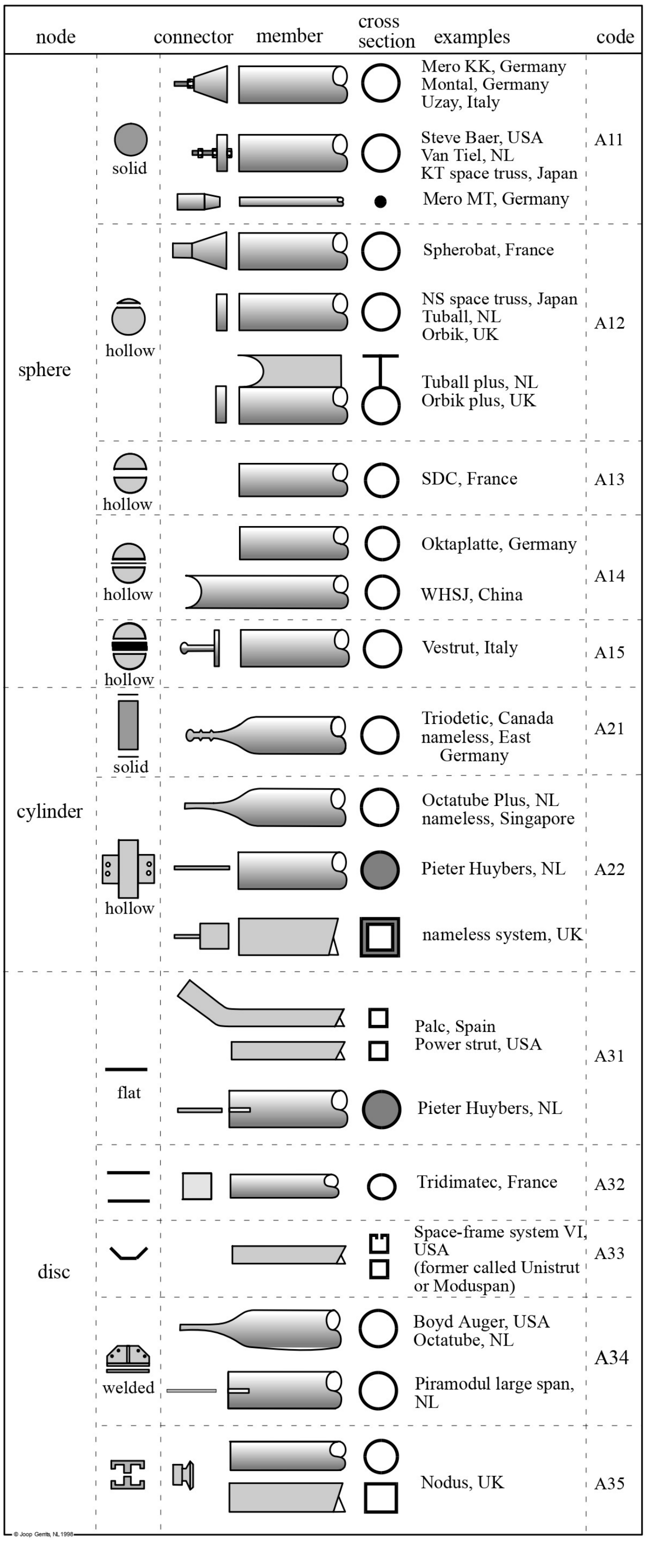
[65] Jeung Hwan Doh, Nhat Minh Ho, Dane Miller, Tim Peters, David Carlson, and Pasteur Lai. Steel Bracket Connection on Modular Buildings. *Journal of Steel Structures & Construction*, 02(02), 2017. doi: 10.4172/2472-0437.1000121.

[66] Seungjae Lee, Jaeseong Park, Euishin Kwak, Sudeok Shon, Changhoon Kang, and Hosoon Choi. Verification of the Seismic Performance of a Rigidly Connected Modular System Depending on the Shape and Size of the Ceiling Bracket. *Materials 2017, Vol. 10, Page 263*, 10(3):263, 3 2017. ISSN 1996-1944. doi: 10.3390/MA10030263. URL `https://www.mdpi.com/1996-1944/10/3/263/htmhttps://www.mdpi.com/1996-1944/10/3/263`.

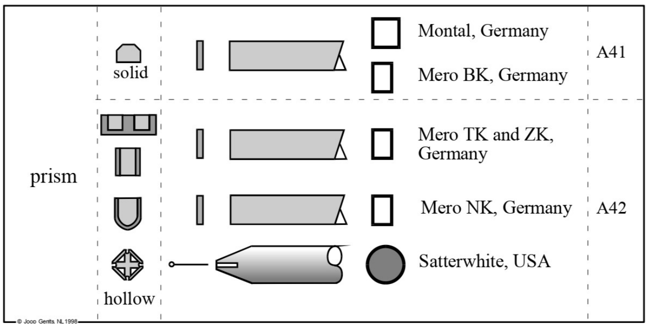
[67] En Feng Deng, Jia Bao Yan, Yang Ding, Liang Zong, Zhong Xian Li, and Xiao Meng Dai. Analytical and numerical studies on steel columns with novel connections in modular construction. *International Journal of Steel Structures*, 17(4):1613–1626, 12 2017. ISSN 20936311. doi: 10.1007/s13296-017-1226-5.

[68] Zhihua Chen, Yang Liu, Xu Zhong, and Jiadi Liu. Rotational stiffness of inter-module connection in mid-rise modular steel buildings. *Engineering Structures*, 196:109273, 10 2019. ISSN 0141-0296. doi: 10.1016/J.ENGSTRUCT.2019.06.009.

A Overview of connection types with a node, without a node and with prefabricated units



(a) Connection types with a node part 1 (Source: Gerrits [39])



(b) Connection types with a node part 2 (Source: Gerrits [39])

Figure A.1: Connection types with a node

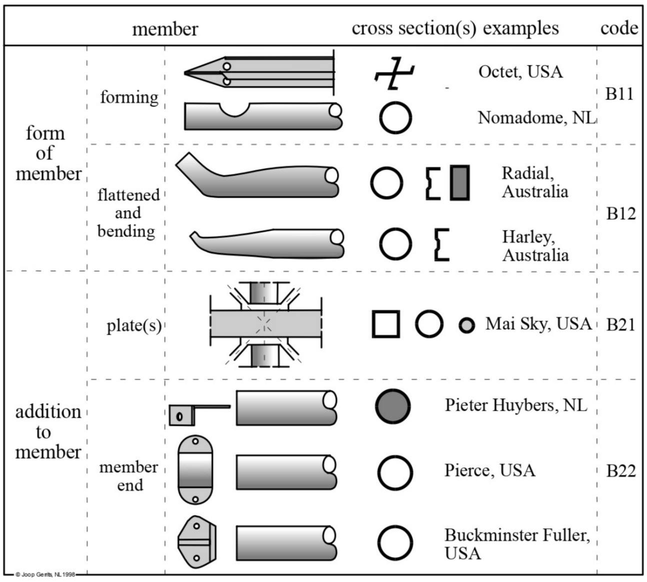


Figure A.2: Connection types without a node (Source: Gerrits [39])

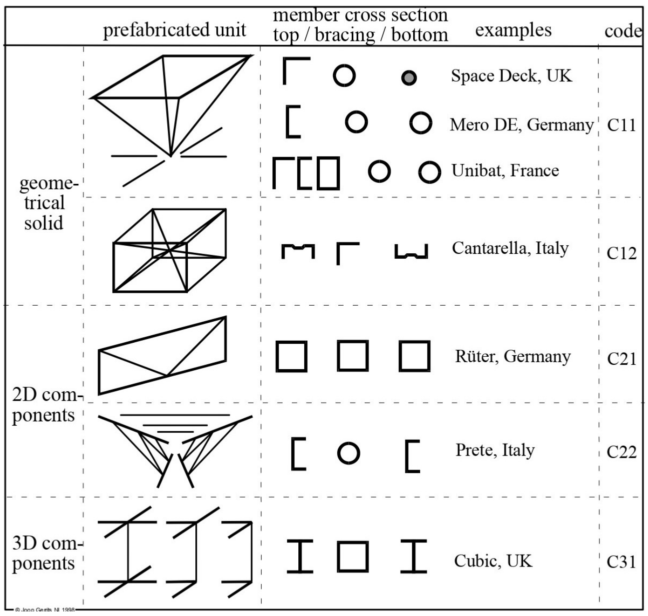


Figure A.3: Connection types with prefabricated units (Source: Gerrits [39])

Table A.1: Commonly used proprietary systems (Source: Lan [10])

Name	Country	Period of development	Material	Connecting method
MERO	Germany	1940–1950	Steel Aluminum	Bolting
Space Deck	United Kingdom	1950–1960	Steel	Bolting
Triodetic	Canada	1950–1960	Aluminum Steel	Inserting member ends into hub
Unistrut (Moduspan)	United States	1950–1960	Steel	Bolting
Oktaplatte	Germany	1950–1960	Steel	Welding
Unibat	France	1960–1970	Steel	Bolting
Nodus	United Kingdom	1960–1970	Steel	Bolting and using pins
NS	Japan	1970–1980	Steel	Bolting

B Overview of inter modular connection types

The figures below form an overview of existing inter-module connection types. The presented overview is collected from the overviews of bolted inter-module connections and the performance of inter-module connections in multi-story modular buildings by Lacey et al. and Srisangeerthanan et al.. [49, 50].

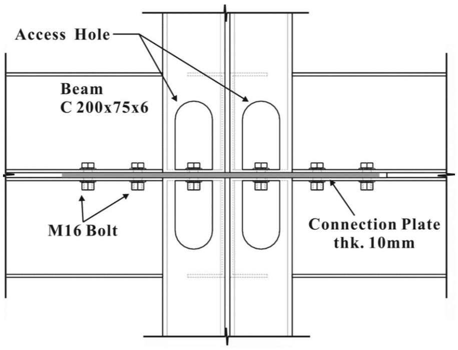


Figure B.1: Bolted connection plate, connecting Hollow Steel Section (HSS) (Source: Choi et al. [64])

A bolted connection plate is placed between two stacked modules. It is bolted together through the flanges of the upper and bottom beam. There is a hole in the columns which is therefore weakened [50]. This is done to be able to access the bolts that are situated at that location. Some localised strengthening might be needed near the holes.

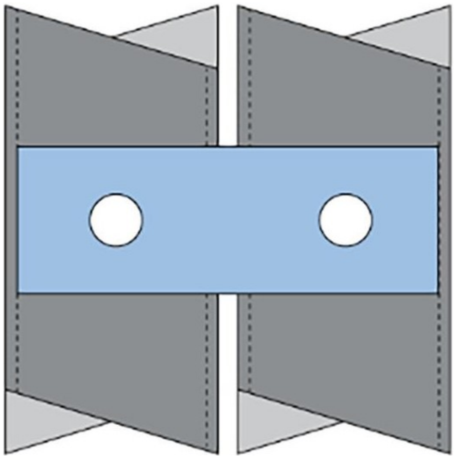


Figure B.2: Tie plate connecting hollow or open steel section column (Source: Lacey et al. [49])

A tie plate can connect two sections to each other. These can either be hollow or open. This is done by bolts. It can be used to connect modules as well but the connections are pinned so they cannot take moments unless multiple bolts are applied.

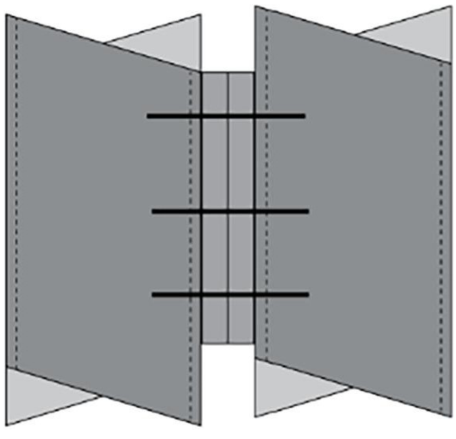


Figure B.3: Bolted side plate connecting HSS(Source: Lacey et al. [49])

This relatively simple bolted connection. Multiple bolts join the side plates of the elements together. It has a relatively high initial stiffness. When subjected to loading the failure mode of this connection is a combination of plate yielding and tension failure of the bolts [49].

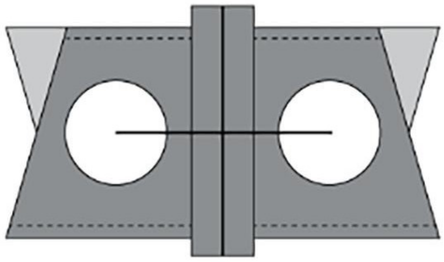


Figure B.4: Bolted end plate connecting HSS with access hole, or open-angle section columns (Source: Lacey et al. [49])

This end plate connection is used for HSS elements. Two columns in this example are joined together by a bolt that goes through the endplates of the elements. Holes are made in the columns so that the bolt nut can be reached and fastened.

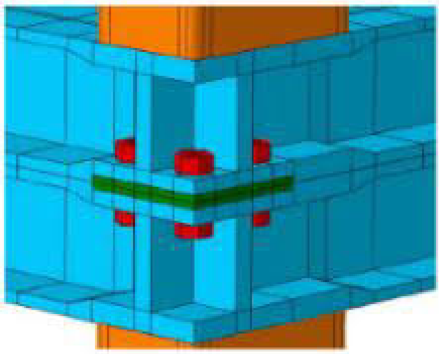


Figure B.5: Bolted connection, connecting open steel section beams (Source: Lacey et al. [49])

This is a relatively easy bolted connection between open steel section beam elements. Construction is easy but it is not applicable on HSS elements.

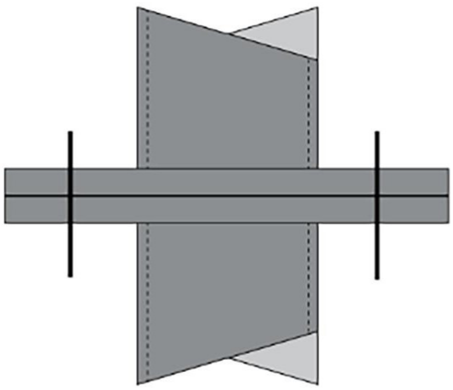


Figure B.6: Bolted end plate (bolts on two sides), connecting HSS (Source: Lacey et al. [49])

This bolted connection is between two elements in the direction of their centroidal axis. It is like in Figure B.3 done with plates with bolts running through them.

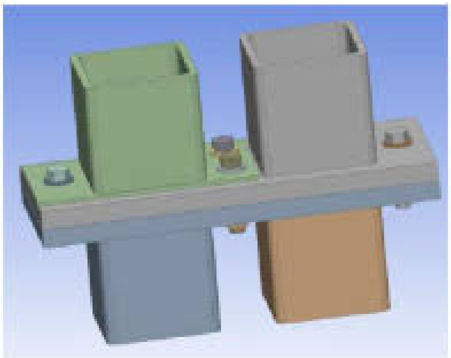


Figure B.7: Complex bolted endplate, connecting HSS (Source: Lacey et al. [49])

This is also an endplate connection like Figure B.4 and Figure B.6. With this connection, however, more than two modules are connected. Two of the endplates are elongated on one side. This allows connecting an extra two elements on the elongated parts of the endplates.

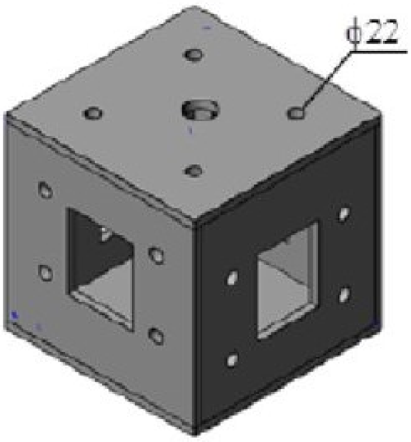


Figure B.8: Steel bracket welded to corner columns (Source: Lacey et al. [49])

This connection is node-like. It is a 370 × 370 × 370 mm hollow cube with a wall thickness of 15 mm. There are many different faces. One face is plain, two faces have rectangular cut-outs for the use of assembly tools. There are two faces with four 24-mm-diameter holes for bolts and a rectangular cut-out for access. Furthermore, there is one face with four 24-mm-diameter holes for bolts and a larger 48-mm-diameter hole in the center for ease of transport The diameter of the bolts is 22 mm (M22) with a steel grade of 8.8.[65]. The cubes can be used as corner points for modules. Different modules can then be bolted together.

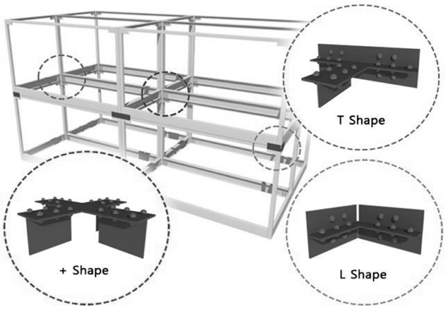


Figure B.9: Steel bracket bolted or welded to floor and ceiling beams (Source: Lee et al. [66])

This connection consists of metal plates that can be attached to webs and flanges of adjacent modules. The steel plates are connected with bolts both horizontally and vertically [50]. It allows for attaching multiple modules.

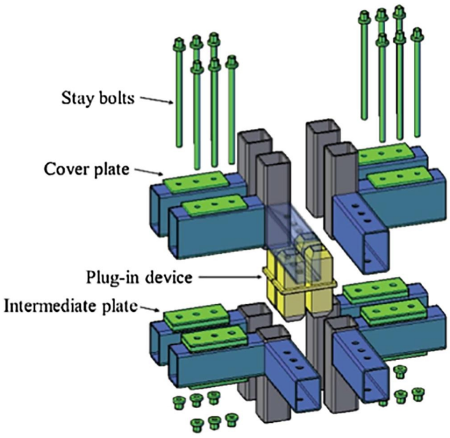


Figure B.10: Bolted connection with plug-in device (Source: Lacey et al. [49])

Eighth modules can be connected with this type of connection. The modules consisting of SHSs are vertically connected with long bolts that go through cover plates and beam elements that lay on top of each other. A plug-in device is placed in the vertically placed open cross-sections. It fits well into it in one piece and provides horizontal connectivity [50].

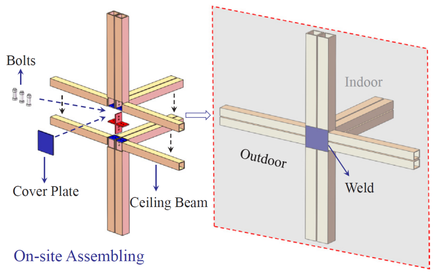


Figure B.11: Bolted connection with welded cover plate (Source: Deng et al. [41])

In this connection a cruciform plate with boltholes is used as a connecting element. It is placed in between four modules that are connected to the plate with bolts. To be able to connect the bolts the SHS elements have holes cut in them to be able to have access to the boltholes. After bolting the modules together a cover plate is welded to cover the holes. The thickness of the gusset plate is 10 to 20 mm. It was found that the connection could be used to form a moment-resistant frame. According to Eurocode 3 Part 1–8 the connection can be classified as semi-rigid. Failure was mainly due to local buckling followed by weld fracture of the cover plate welds [41].

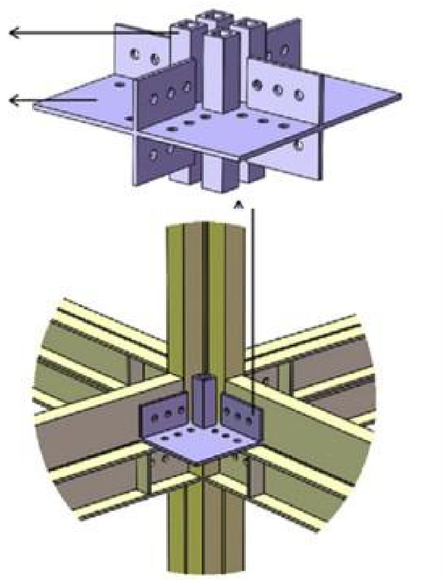


Figure B.12: Bolted connection with rocket-shaped tenon and gusset plates (Source: Deng et al. [67])

The shape of this connection looks a bit like a combination of the plug-in device of Figure B.10 and the cruciform shape of Figure B.11. The plug-in part can fit right into the columns of a module while the beams are bolted together with the gusset plates. The bolts run through the webs and the flanges of the connected beams [50].

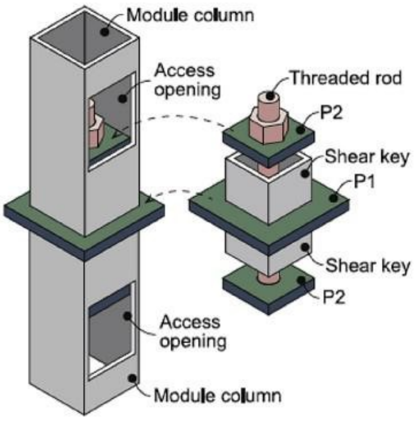


Figure B.13: Pre-tension bolt connection with plug-in shear key (Source: Lacey et al. [49])

A shear key with a threaded rod inside realises this connection between two module columns. The shear key fits into the columns to provide lateral strength. The rod is pre-tensioned and is accessible through access openings in the columns. Two plates provide vertical anchorage. The holes inside the columns may cause some weakening of the cross-sections and probably strengthening is needed [50].

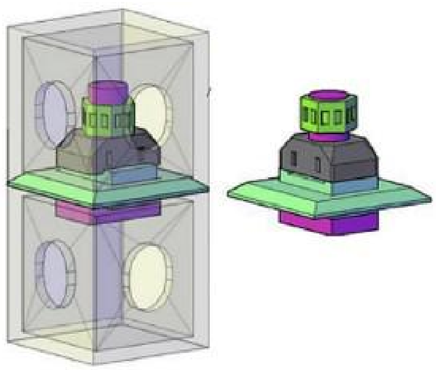


Figure B.14: Connection between HSS with corner fitting and connector (Source: Chen et al. [68])

In this figure, a special kind of connection between columns is shown. The lower column has a rectangular hole where the connector fits in. This connector has a nut on top which can be rotated with an instrument that fits through the hole in the column. By rotating the nut, the lower part of the connector rotates as well so that it doesn't fit in the rectangular hole anymore. After this final step the modules are connected [68]. Horizontal connectivity may later be provided by welding the plate elements of the connector [50].

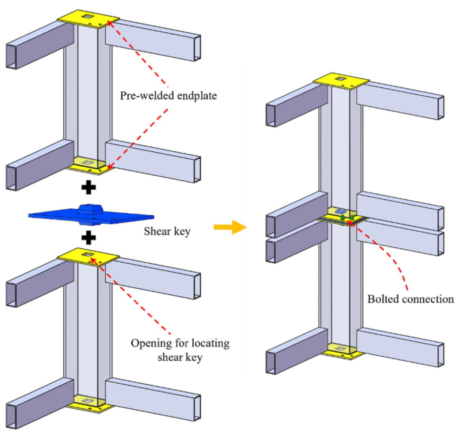


Figure B.15: Module connection with pre-welded endplate and shear key (Source: Han et al. [51])

This type of connection in Figure B.15 is made by pre-welding endplates to the cubic module. A shear key is placed between the modules and the endplates are bolted together. This shear key takes the shear force and the bolts and modules take the tension or compression. The modules fit well because of the shape of the shear key and construction is therefore easy [52]. The main failure modes are bolt fracture and yielding of the endplate under tensile loading. The endplate can also bend around the bolts resulting in separation of the endplates [52].

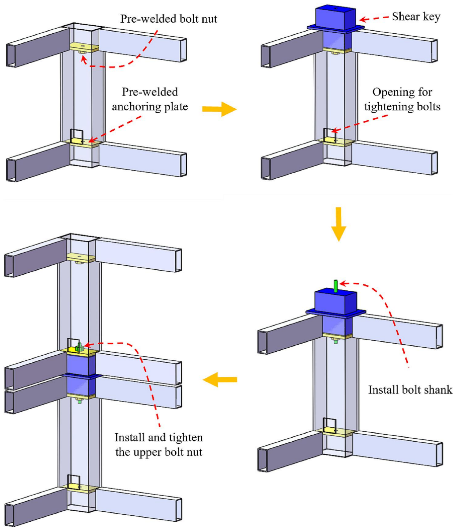


Figure B.16: Module connection with endplate and upper and lower tubes (Source: Han et al. [51])

This type of connection is made with a tubular shear key that can be placed on the module very similar to Figure B.13. The shear key has a hole inside to allow a bolt to pass through. The module contains a pre-welded bolt that fits through the shear key. After placing the shear key and the bolt in place a second module can be connected to the shear key and be tightened with another bolt nut. This bolt nut is accessed through a hole in the frame [51]. The failure modes are bolt fracture and yielding of the anchoring plate. Also, the bolts experienced significant shear force [51].

C Forces and moments in truss model

This section shows the occurring forces and moments in the truss model used for cross-section optimisation in Section 3.4. Also, the moment-to-force ratios in the structure are shown. These values are used in the joint design in Section 4. The results of four models with different module sizes are shown in Appendix C.1 to Appendix C.6. The M/N ratios that are used for the joint design can be seen in Appendix C.7. The nodal forces and moments that are displayed are in the local coordinate systems of the elements. In Karamba3D the x-axis in this coordinate system is the beam axis. Together with the y- and z-axis, they form a right-handed coordinate system. The normal force for all members is the most critical one because these forces are much higher than the forces in y and z. The moments around the beam's axis are small compared to the other moments. Therefore M_x will be neglected and M_y and M_z are analysed in the model.

C.1 Forces and moments in beam splices

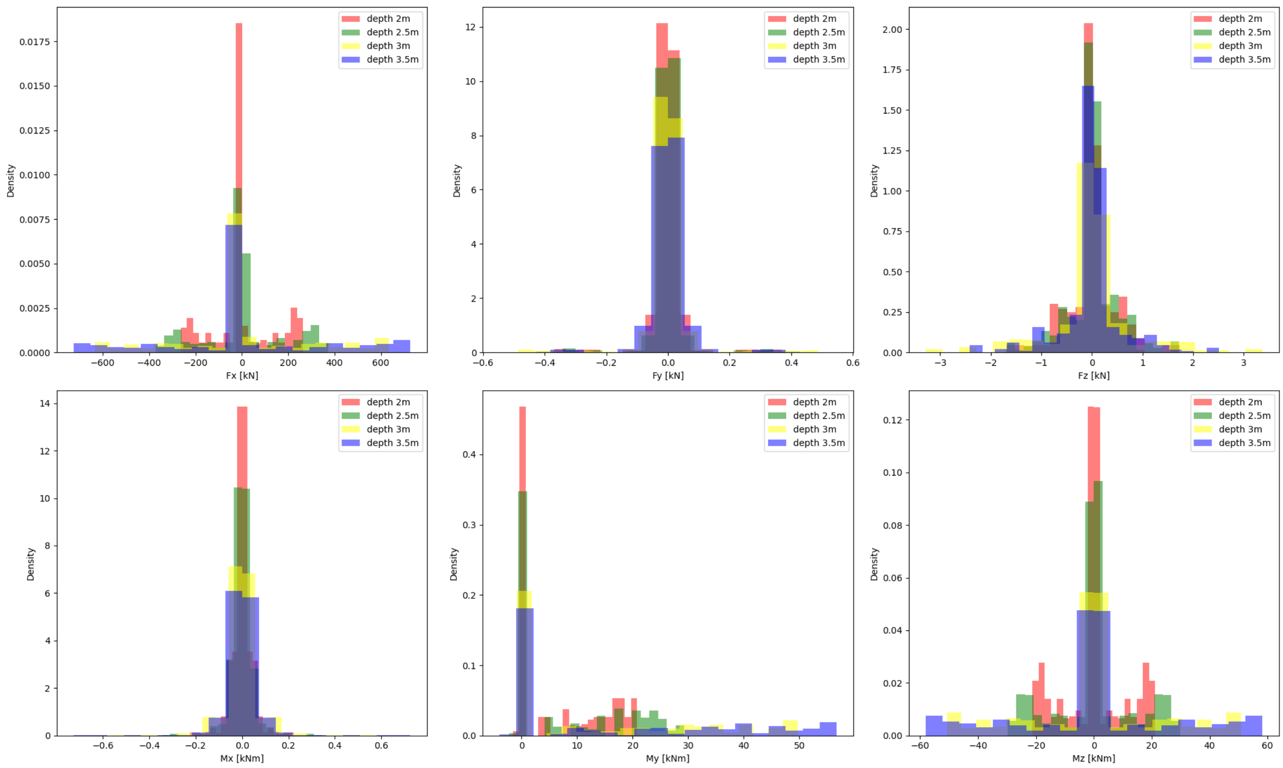


Figure C.1: Forces and moments in beam splices (connection type D)

Table C.1: Beam splices forces and moments (connection type D)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-264.273	264.803	-0.369	0.369	-1.687	1.728	-0.224	0.224	-1.518	20.804	-21.184	21.184
2.5	-371.707	372.393	-0.428	0.428	-1.574	1.613	-0.345	0.345	-2.211	29.355	-29.791	29.791
3.0	-496.813	497.590	-0.487	0.487	-1.679	1.546	-0.478	0.478	-3.026	39.310	-39.806	39.806
3.5	-639.521	640.377	-0.546	0.546	-1.860	1.861	-0.626	0.626	-3.962	50.666	-51.229	51.229

C.2 Forces and moments in column splices

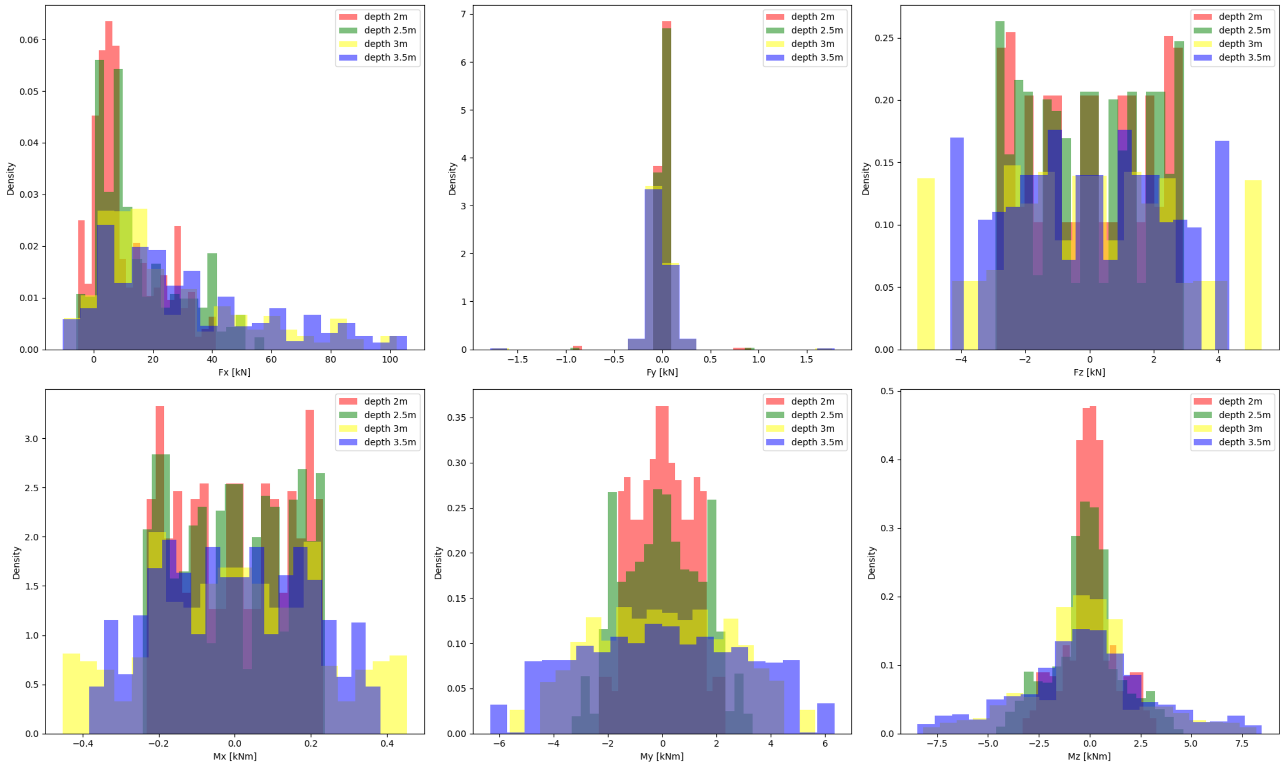


Figure C.2: Forces and moments in column splices (connection type C)

Table C.2: Column splices forces and moments (connection type C)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-5.220	41.157	-0.925	0.924	-2.889	2.892	-0.232	0.232	-2.331	2.331	-3.287	3.287
2.5	-5.867	57.494	-0.956	0.956	-2.936	2.940	-0.242	0.237	-3.337	3.337	-4.596	4.596
3.0	-9.870	102.289	-1.766	1.766	-5.353	5.358	-0.453	0.453	-5.633	5.633	-8.183	8.183
3.5	-10.407	105.720	-1.787	1.788	-4.336	4.343	-0.383	0.383	-6.347	6.347	-8.460	8.460

C.3 Forces and moments in diagonal splices

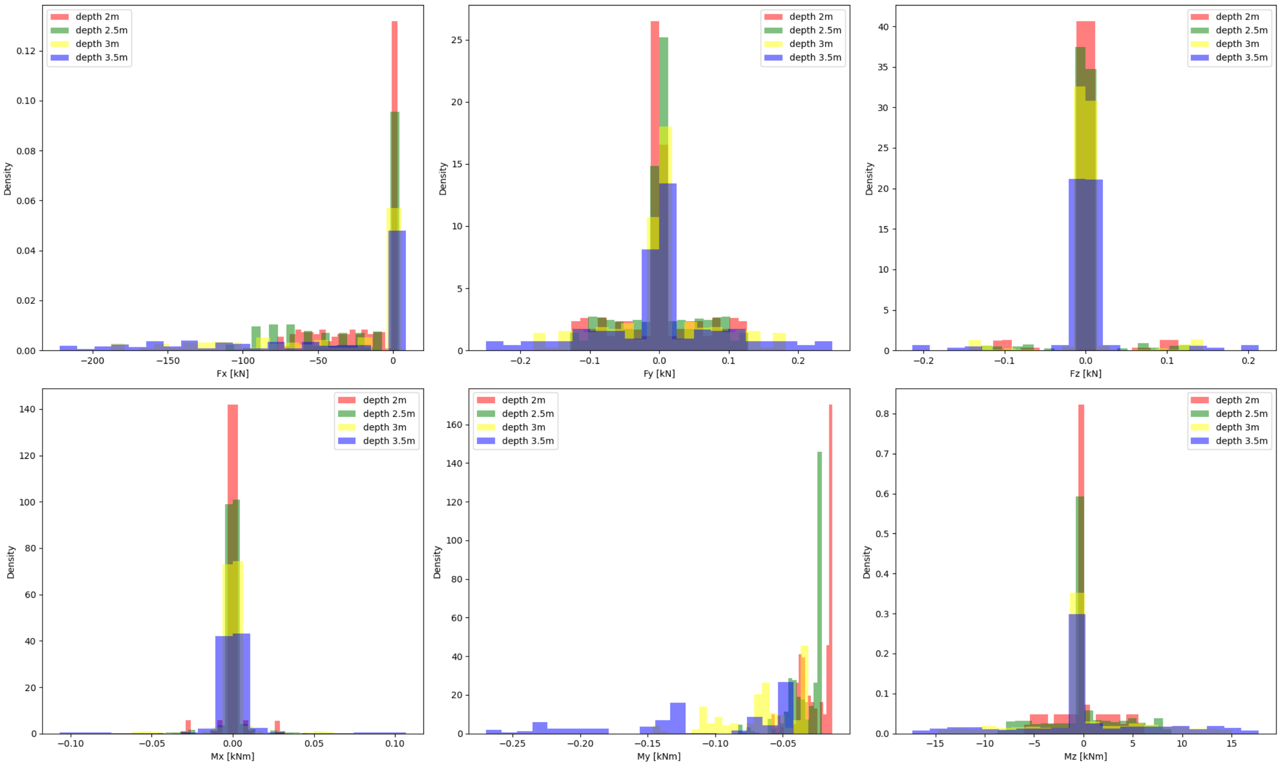


Figure C.3: Forces and moments in diagonal splices (connection type H)

Table C.3: Diagonal splices forces and moments (connection type H)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-76.883	3.048	-0.127	0.127	-0.114	0.114	-0.032	0.032	-0.058	0.013	-6.020	6.151
2.5	-111.394	4.161	-0.128	0.128	-0.129	0.129	-0.046	0.046	-0.083	0.021	-8.716	8.912
3.0	-188.161	5.473	-0.182	0.182	-0.144	0.144	-0.062	0.062	-0.147	0.031	-14.796	15.053
3.5	-221.917	8.525	-0.249	0.249	-0.213	0.213	-0.107	0.107	-0.270	0.042	-17.291	17.753

C.4 Forces and moments in columns connected to corner points

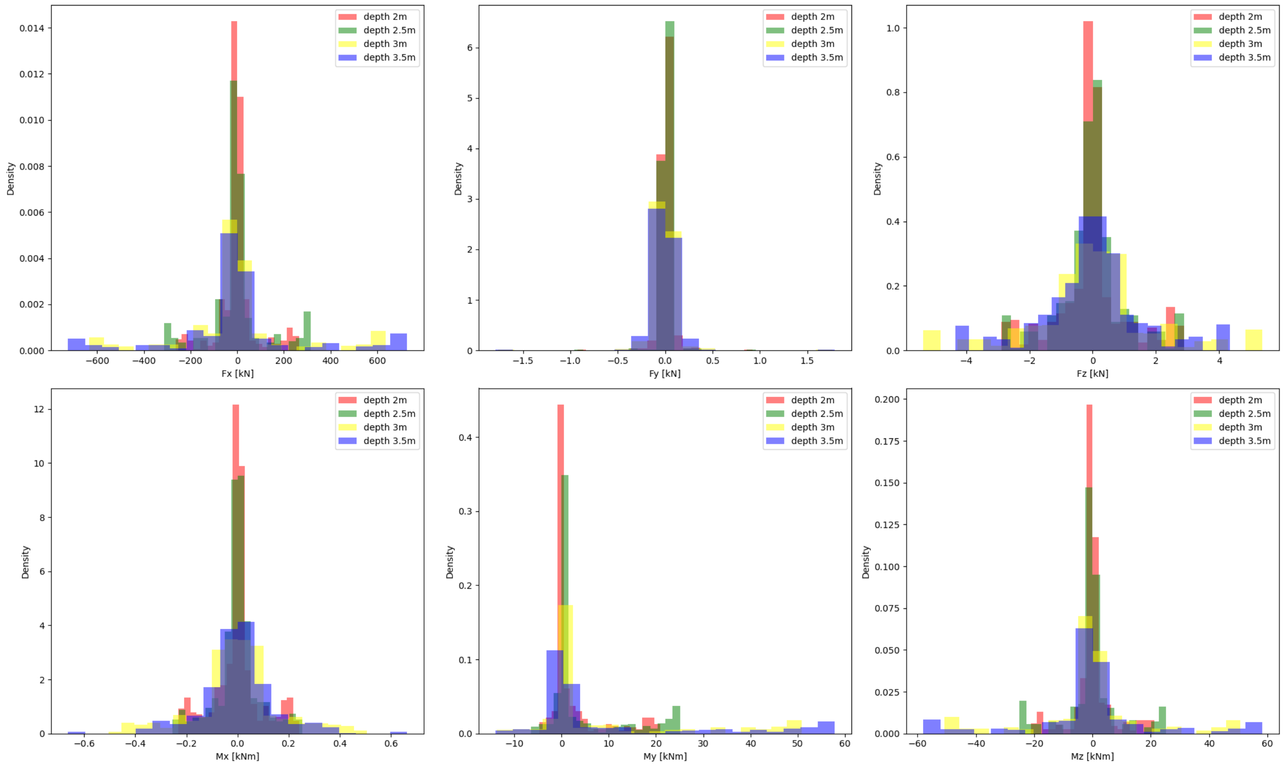


Figure C.4: Forces and moments in columns connected to corner points (connection type A)

Table C.4: Columns to corner points forces and moments (connection type A)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-264.202	264.803	-0.925	0.924	-2.889	2.892	-0.231	0.241	-4.636	20.927	-21.136	21.183
2.5	-314.098	314.770	-0.956	0.956	-2.878	2.881	-0.253	0.253	-6.384	25.105	-25.138	25.181
3.0	-633.760	634.903	-1.766	1.766	-5.353	5.358	-0.506	0.506	-13.669	50.865	-50.718	50.795
3.5	-724.752	726.998	-1.787	1.788	-4.336	4.343	-0.663	0.663	-13.916	57.787	-58.032	58.175

C.5 Forces and moments in beams connected to corner points

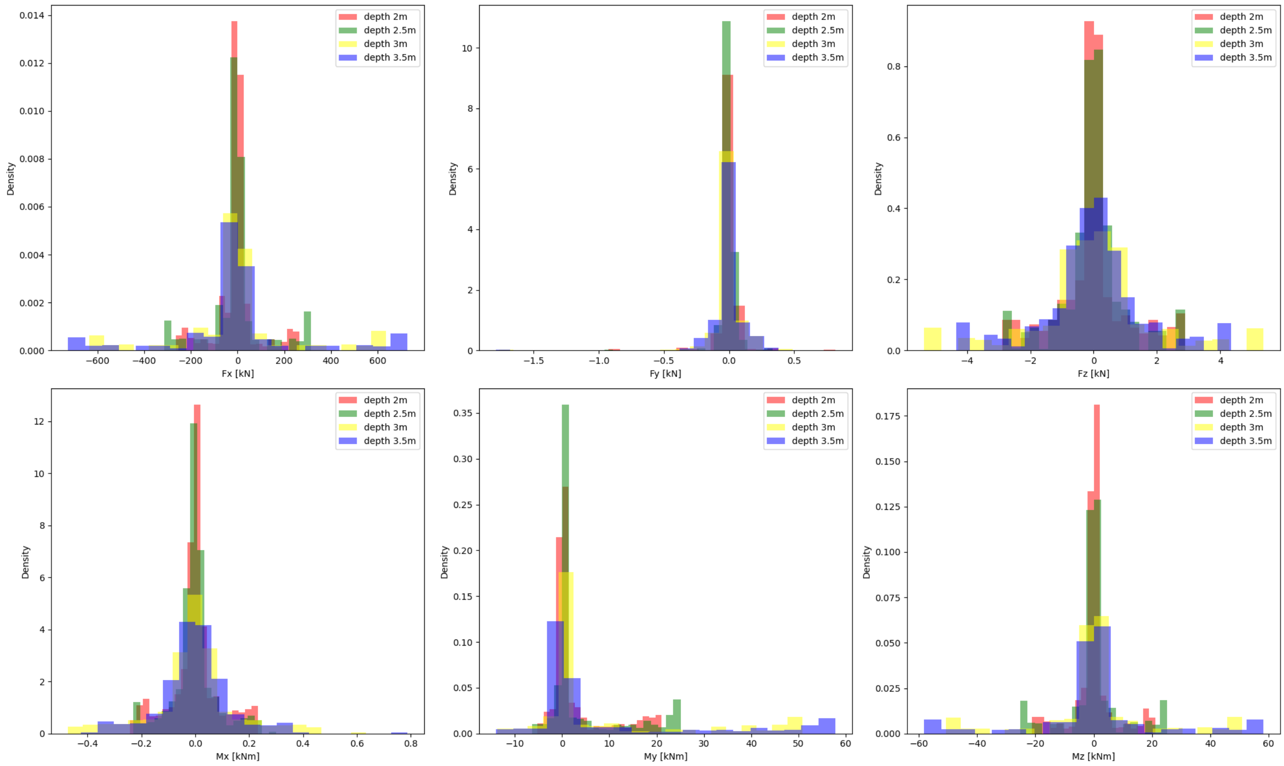


Figure C.5: Forces and moments in beams connected to corner points (connection type B)

Table C.5: Beams to corner points forces and moments (connection type B)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-264.272	264.803	-0.925	0.815	-2.889	2.892	-0.241	0.232	-5.223	21.055	-21.183	21.151
2.5	-314.256	314.770	-0.956	0.332	-2.878	2.881	-0.231	0.300	-6.406	25.105	-25.181	25.181
3.0	-633.990	634.903	-1.766	0.487	-5.353	5.358	-0.472	0.634	-13.693	50.865	-50.795	50.789
3.5	-725.499	726.998	-1.787	0.377	-4.336	4.343	-0.424	0.787	-13.950	57.787	-58.175	58.143

C.6 Forces and moments in diagonals connected to corner points

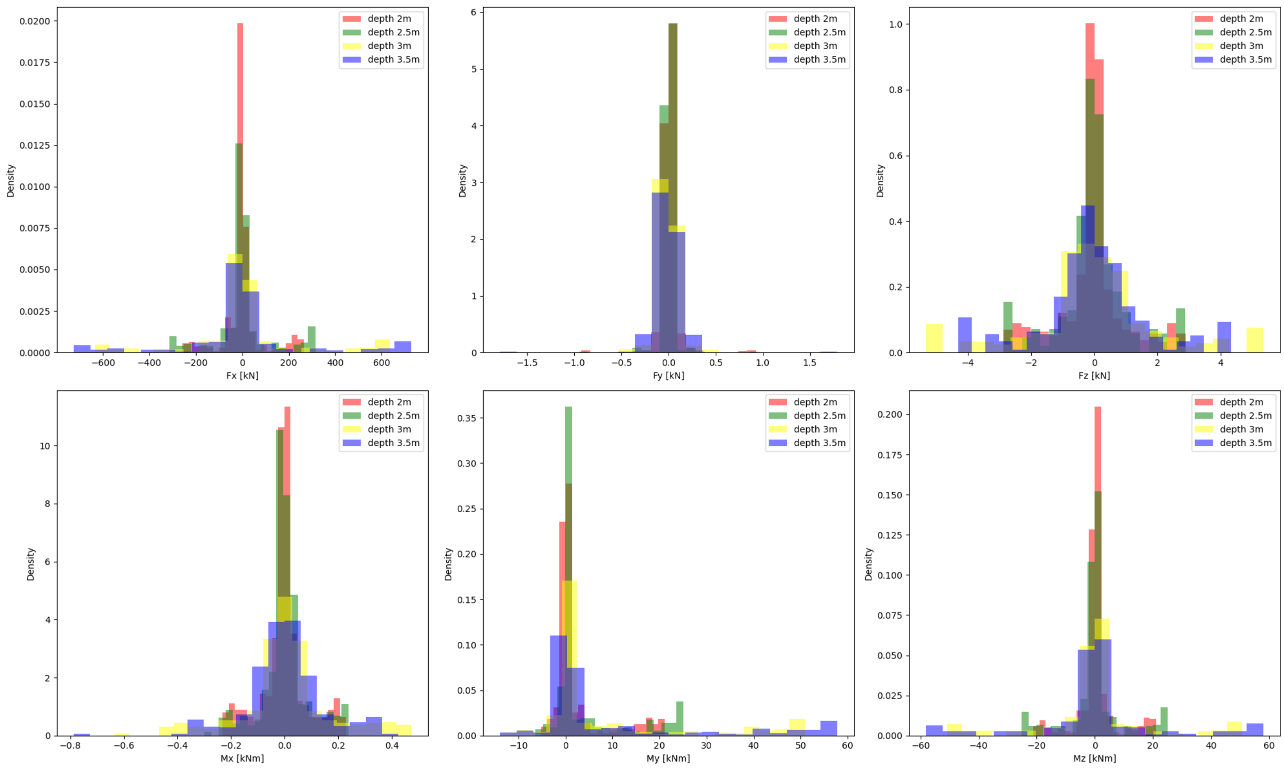


Figure C.6: Forces and moments in diagonals connected to corner points (connection type G)

Table C.6: Diagonals to corner points forces and moments (connection type G)

Depth [m]	Fx [kN]		Fy [kN]		Fz [kN]		Mx [kNm]		My [kNm]		Mz [kNm]	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
2.0	-257.302	264.372	-0.925	0.924	-2.879	2.892	-0.232	0.229	-5.223	21.055	-21.151	21.107
2.5	-314.256	314.770	-0.956	0.956	-2.866	2.881	-0.300	0.237	-6.411	25.103	-25.181	25.147
3.0	-633.990	634.903	-1.766	1.766	-5.330	5.358	-0.634	0.472	-13.702	50.859	-50.789	50.722
3.5	-725.499	726.998	-1.787	1.788	-4.312	4.343	-0.787	0.424	-13.966	57.777	-58.143	58.083

C.7 |M|/N ratios in the nodes

The ratios of the moments to the normal forces are important data for the design of the joints. Some of the normal forces are very small, leading to large extreme |M|/N ratios. Because of this, the outliers are large as well for these ratios. To identify the outliers the IQR is used. The IQR is a subset of a list of values that are situated between the median of the first quartile and the third quartile. The outliers are the values outside of the range that is given in Equation (C.1). These moment-to-force ratios will not be analysed in the joint design except when a large moment or force causes these outliers. The absolute value of all of the moments is taken in the calculations. This is done to be able to make a distinction between compressional and tensional forces present in the members. From Table C.7 it is clear that the normal forces in the columns and the diagonal splices are much larger than the occurring moments. This is expected since the point loads acting on the structure are placed directly on the columns. Furthermore, the diagonals are meant to transfer normal forces and not bending moments. The ratio of moments to normal forces in the corner joints and the beam splices are larger than in the other joints. For the corner joints, which is expected. The chords have an analogy with a beam on discrete supports, in this case, the columns. Large positive bending moments occur at these supports. The discontinuity in the structure can explain the relatively larger moment-to-normal force ratios in the splices because of these joints. This causes a different moment distribution leading to relatively larger bending moments in these joints.

$$Q_1 - 1.5 \cdot \mathrm{IQR} \leq x \leq Q_3 + 1.5 \cdot \mathrm{IQR} \tag{C.1}$$

In which:

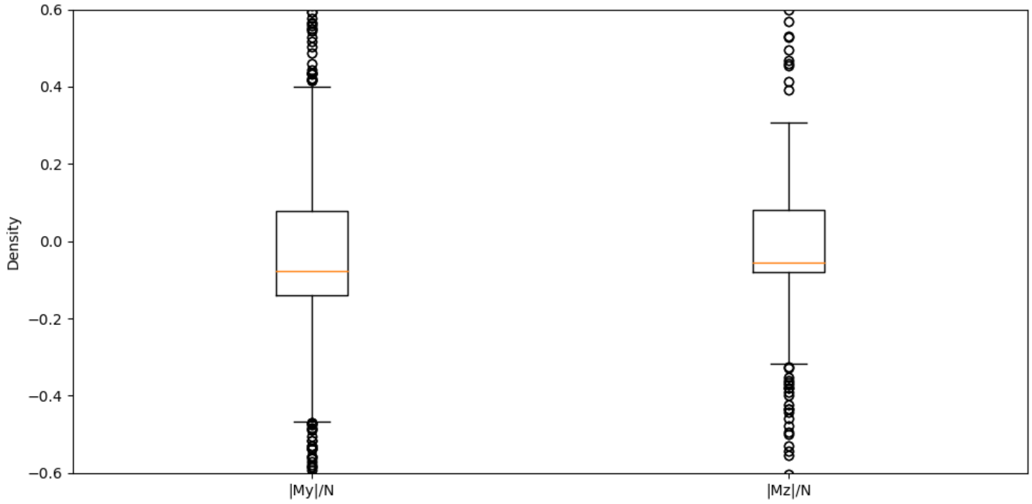
Q_1 = First quartile (25^{th} percentile) [-]

Q_3 = Third quartile (75^{th} percentile) [-]

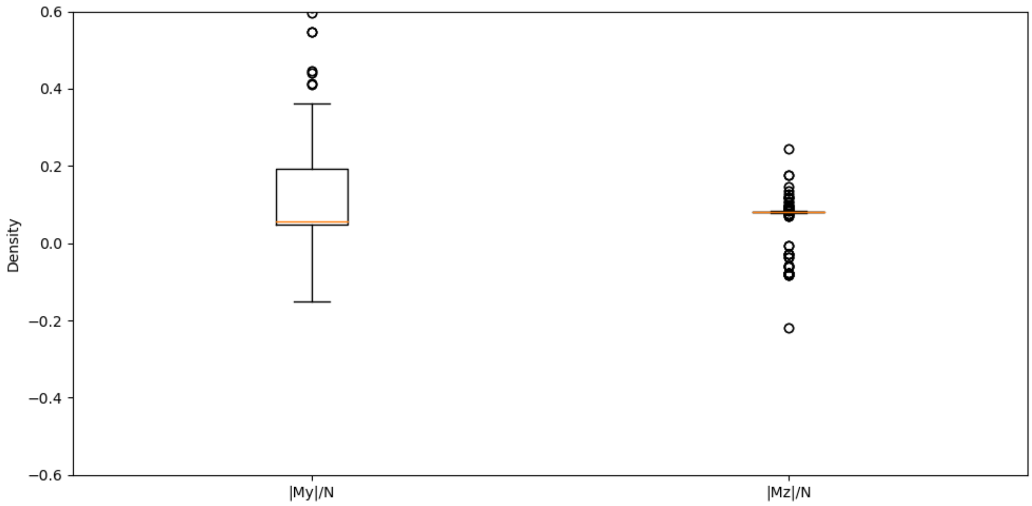
IQR = Interquartile Range (Q_3 - Q_1) [-]

Table C.7: Range in |My|/N and |Mz|/N values without outliers

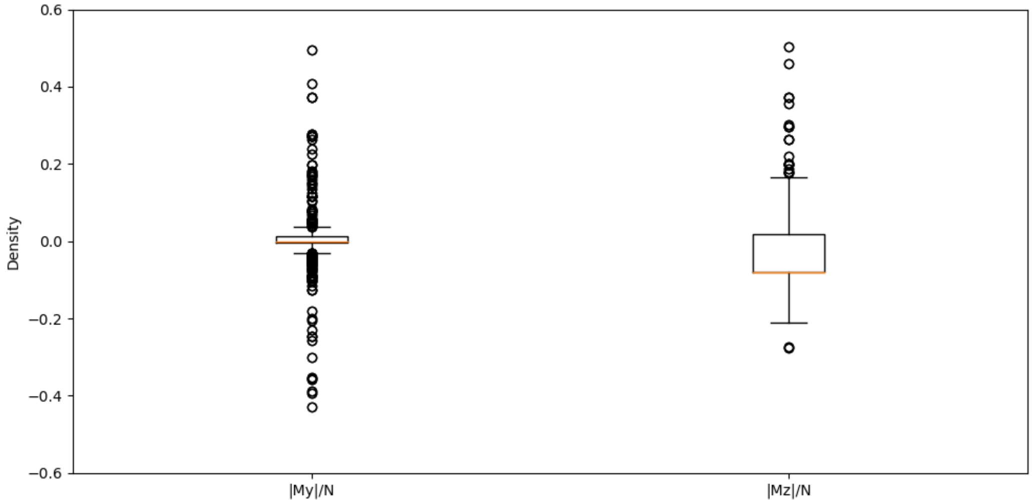
Type of joints	**|My|/N** range	**|Mz|/N range**
Beam splices (D)	-0.466 to 0.399	-0.318 to 0.307
Column splices (C)	-0.150 to 0.363	-0.077 to 0.085
Diagonal splices (H)	-0.030 to 0.036	-0.211 to 0.166
Columns to corner points (A)	-0.357 to 0.443	-0.308 to 0.320
Beams to corner points (B)	-0.388 to 0.450	-0.308 to 0.298
Diagonals to corner points (G)	-0.402 to 0.450	-0.304 to 0.320



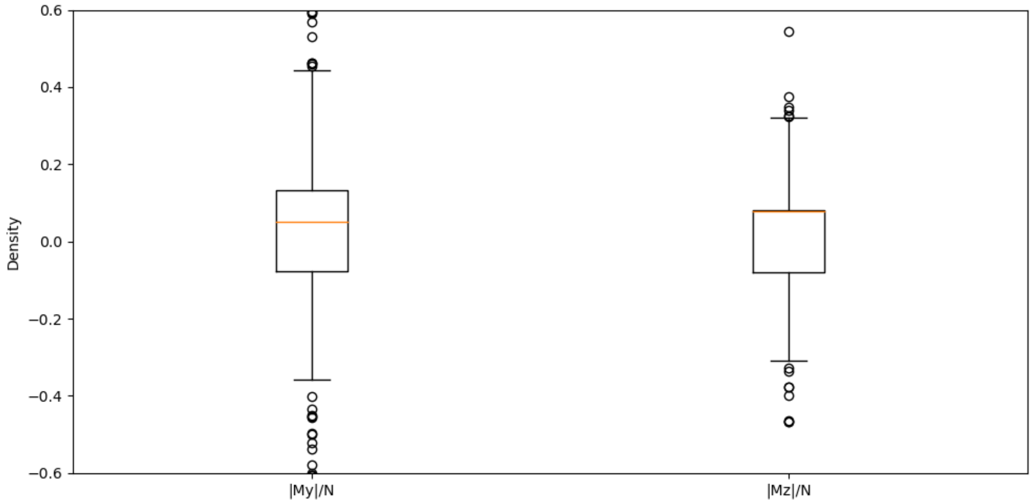
(a) |M|/N boxplot beam splices (connection type D)



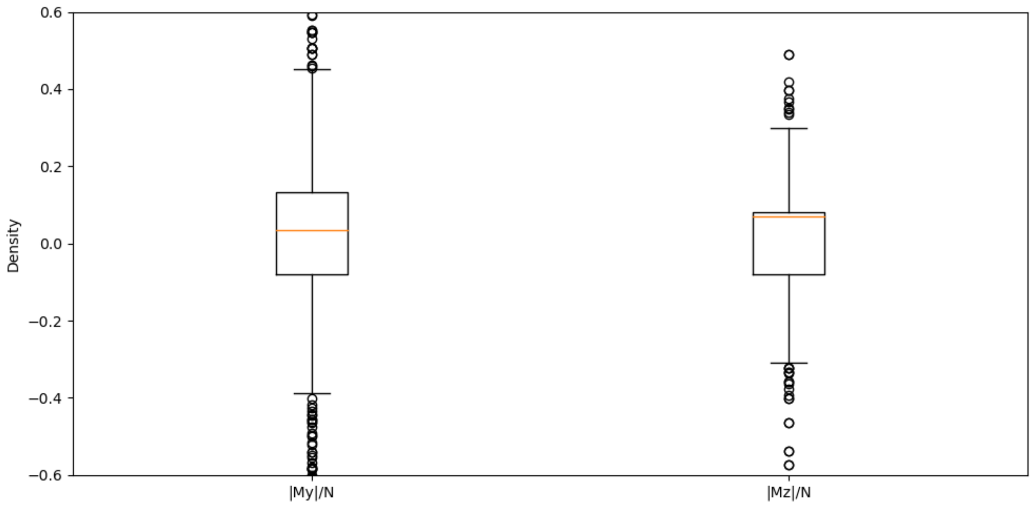
(b) |M|/N boxplot column splices (connection type C)



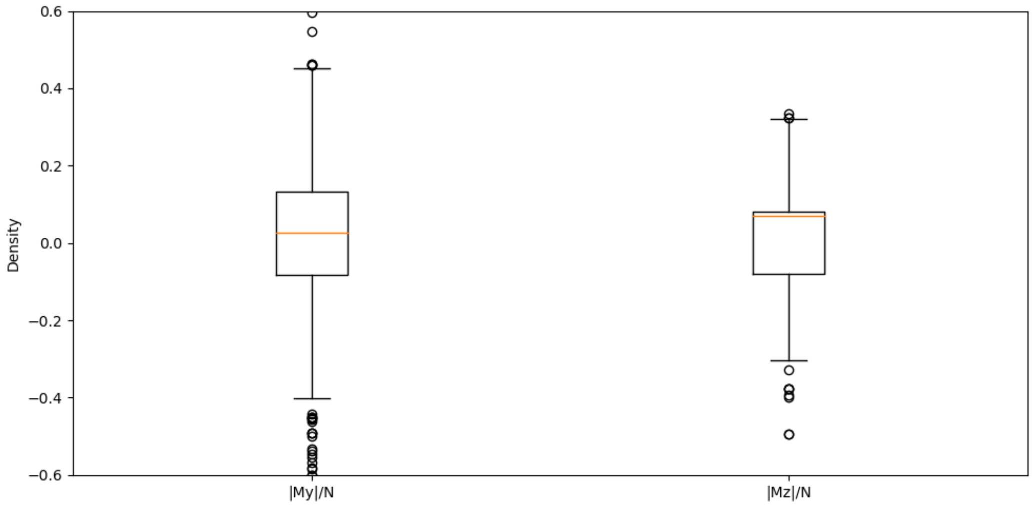
(c) |M|/N boxplot diagonal splices (connection type H



(d) |M|/N boxplot columns to corner points (connection type A)



(e) |M|/N boxplot beams to corner points (connection type B)



(f) |M|/N boxplot diagonals to corner points (connection type G)

Figure C.7: |M|/N boxplots for different joints in the model

D Stiffness analysis of the joints

In this appendix, the resulting stiffness values that are taken from IDEA StatiCa are displayed for varying M/N ratios. These ranges are input for the FEM model in Grasshopper. Depending on the forces and moments measured in the model new stiffness values are assigned to the joints in an iterative loop. The range in |M|/N values is larger than the ranges shown in the boxplots of Appendix C.7. The reason for this is that if the moments were not considered, there would be no values included where the moment exceeds the normal force. This would not give a clear overview of the behavior of the joints. It is observed that there is a peak close to an |M|/N ratio of 0 for all of the joints for both translational and rotational stiffness. This peak is well captured when the range of |M|/N values is increased from -2.0 to +2.0. For the splice joints, there is no distinction between the moment around the local z-axis (M_z) and around the local y-axis (M_y). For joints that form an asymmetric system with connected elements, there is a distinction between the moment around M_z and M_y.

D.1 Hand calculation of the translational stiffness

In this section, the translational stiffness of the various joints is calculated using a simple method. The axial stiffness of a member can be calculated with Equation (D.1). For the length of the member, L is the same as the length of the joint.

$$S_t = \frac{E \cdot A}{L} \tag{D.1}$$

In which:

S_t = Translational stiffness [N/mm]

E = Modulus of elasticity [N/mm^2]

A = Area of the cross-section [mm^2]

L = Length of the member [mm]

The results of the computation can be seen in Table D.1. The area A for each of the plates is the size of the endplate connected to the member. For the splice connections, the length L is the distance from one stiffener to the other. For the other connections, this length is the distance from the element to the node.

Table D.1: Joint types and their translational stiffness

Joint type	E [N/mm^2]	A [mm^2]	L [mm]	St [MN/m]
SHS member to corner point (A, B)	2.1E+09	36100	215	353
SHS splices (C, D)	2.1E+09	36100	180	421
Diagonals bracing 4 beams (E)	2.1E+09	20106	509	83
Diagonals bracing 8 beams (F)	2.1E+09	20106	509	83
Diagonals to corner points (G)	2.1E+09	22698	304	157
CHS splices (H)	2.1E+09	22698	220	217

D.2 Calculation of the initial rotational stiffness using Eurocode component method

In this section the initial rotational stiffness of various used joints is calculated using the Eurocode (NEN EN 1993-1-8 section 6.3) [56]. The component method is used to check the outcome of the stiffness analysis in IDEA StatiCa. The method breaks down a joint into its component parts. Then for each part, the stiffness is calculated. The contributions of all components are added to obtain the total rotational stiffness of the joint. The Eurocode only describes how to calculate the stiffness for open cross-sections such as H- or I-profiles but not for closed cross-sections. An approximation is made for the closed cross-sections used in this design. Relevant stiffness formulas normally used for closed cross-sections are used. Equation (D.2) shows the formula for the initial rotational stiffness of the joint. As can be seen from this formula the contribution of all different components is captured in the summation of all stiffness coefficients k_i for individual joint components.

$$S_{j,ini} = \frac{E \cdot z^2}{\mu \cdot \sum_i \frac{1}{k_i}} \tag{D.2}$$

In which:

$S_{j,ini}$ = Initial rotational stiffness [Nmm/rad]

E = Modulus of elasticity [N/mm^2]

z = Lever arm [mm]

k_i = Stiffness coefficient for basic joint component i [N/mm]

μ = Stiffness ratio [-]

For the value of μ Equation (D.3) is used. In most cases, the values for μ will be equal to 1.

$$\mu = \begin{cases} 1 & \text{if } M_{j,\mathrm{Ed}} \leq \frac{2}{3} M_{j,\mathrm{Rd}} \\ \frac{1.5\, M_{j,\mathrm{Ed}}}{M_{j,\mathrm{Rd}}} & \text{if } \frac{2}{3} M_{j,\mathrm{Rd}} < M_{j,\mathrm{Ed}} \leq M_{j,\mathrm{Rd}} \end{cases} \tag{D.3}$$

In which:

$M_{j,Ed}$ = Design moment at the joint [Nm]

$M_{j,Rd}$ = Design moment resistance of the joint [Nm]

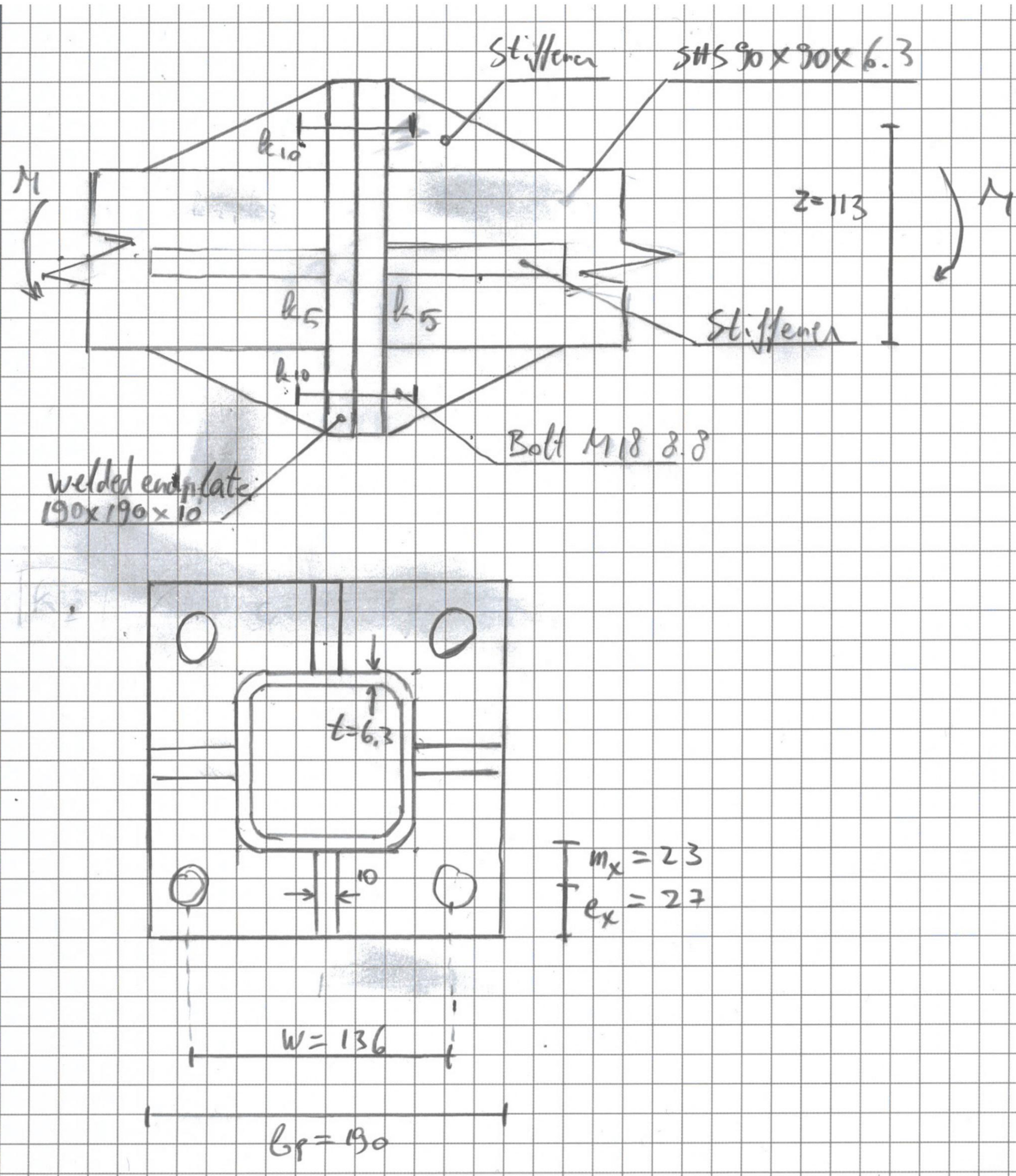
μ = Stiffness ratio [-]

Not all joints can be analysed with this method because of the complexity of the joints, for instance, the node connecting four or eight elements is not something that can be easily calculated with the Eurocode. The splice connections are well-suitable however. Different components play a role in this connection. These are the beam elements, the bolts, and the endplates. The resulting values for the joints that could be analysed with this method are displayed Table D.2. It is in the same order of magnitude as the initial rotational stiffness of intermodular joints found in literature, see Section 4.1.4.

Table D.2: Initial rotational stiffness of joints calculated with the component method

Joint type	$S_{j,ini}$ **[MNm/rad]**
SHS splice (C, D)	11.03
CHS splice (H)	7.71
Beam to corner connection (A, B)	3.22

D.2.1 Hand calculation component method SHS splice



Only k_5 and k_{10} play a role

$$k_5 = \frac{0.9 \cdot l_{eff} \cdot t_{fc}^3}{m^3}$$

$m = m_x$ for this case

From table 6.6

$$\left.\begin{aligned} a &= 4 m_x + 1.25 e_x \\ b &= e + 2 m_x + 0.625 e_x \\ c &= 0.5 b_p \\ d &= 0.5 w + 2 m_x + 0.625 e_x \end{aligned}\right\} l_{eff} = 85.875$$

$$\boxed{\rightarrow k_5 = 6.35}$$

$k_{10} = 1,6\, A_s / L_b$

$A_s = 192\ mm^2$

$L_b = 5 + 3 + \frac{1}{2}(12 + 15) = 21,5$

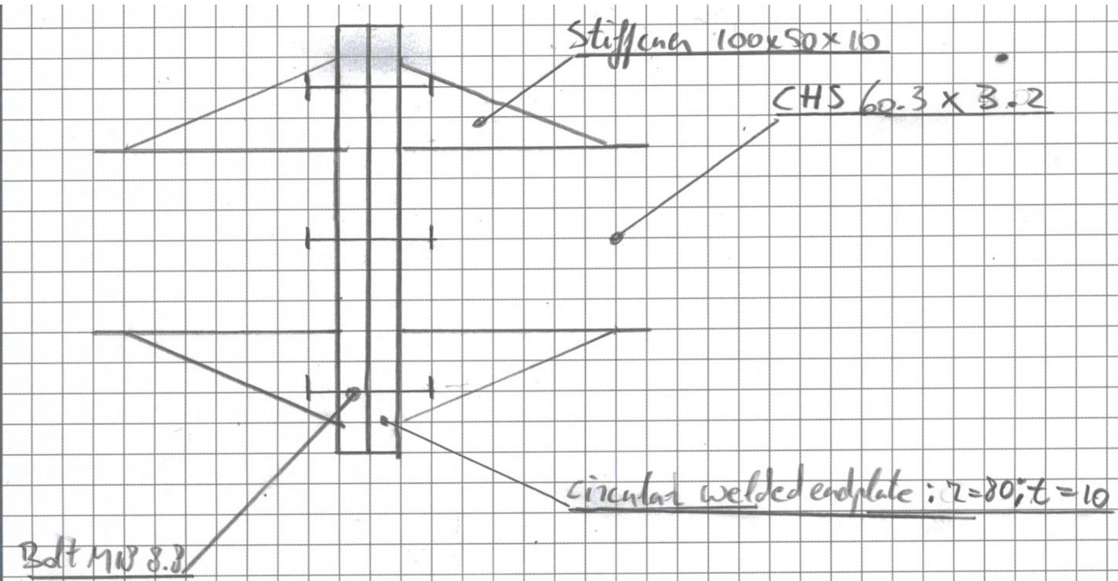
$\rightarrow k_{10} = 11.68$

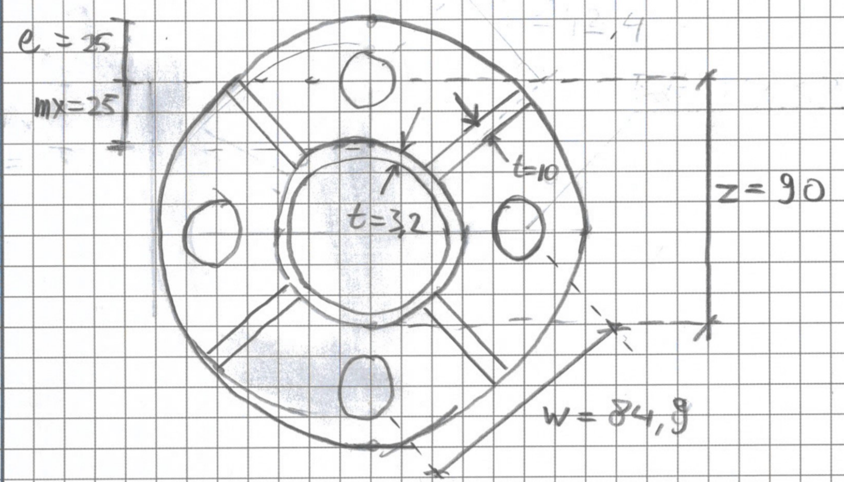
$\sum_i \frac{1}{k_i} = \frac{1}{6,35} + \frac{1}{11,68} = 0,243$

$\mu = 1,0 \quad , E = 210000\ N/mm^2$

$S_{j,ini} = 11,03\ MNm/rad$

D.2.2 Hand calculation component method CHS splice





k_5 and k_{10} play a role for this connection

$k_5 = \frac{0,9 \cdot l_{eff} \cdot t_{fc}^3}{m^3}$ where $m = m_x$ in this case

$l_{eff} = \min(2\pi m_x ; \pi m_x + w ; \pi m_x + 2e) = 128.5$

$\rightarrow k_5 = 7.40$

$k_{10} = 1,6 \frac{A_s}{L_b}$, $A_s = 192\ mm^2$

$L_b = 5 + 3 + \frac{1}{2}(12 + 15) = 21,5$

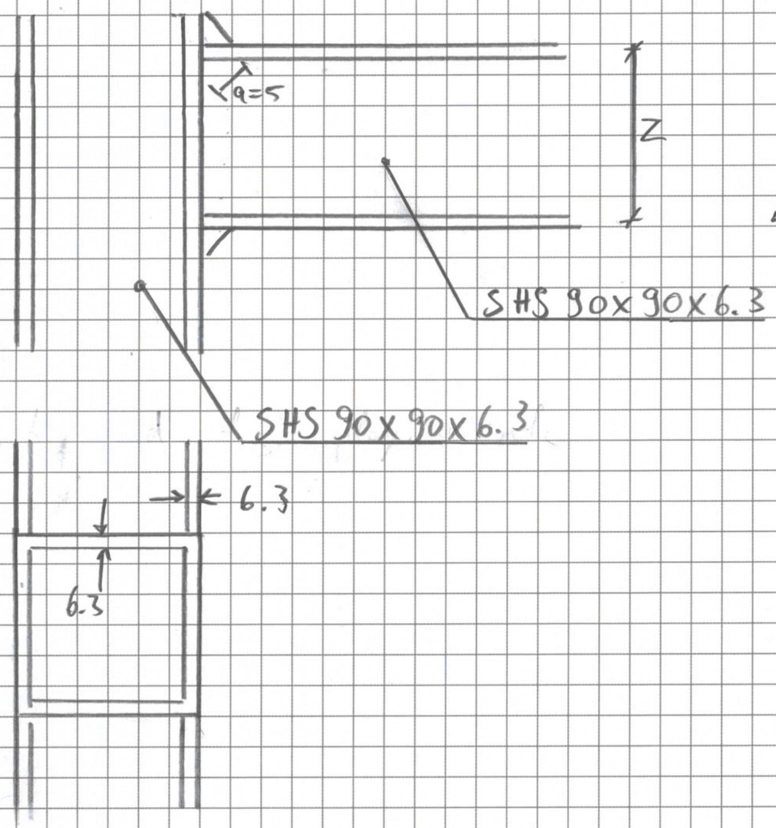
$\rightarrow k_{10} = 11,68$

$$\sum_i \frac{1}{k_i} = \frac{1}{1{,}56} + \frac{1}{11{,}68} = 0{,}22$$

$\mu = 1{,}0$, $E = 210000\ N/mm^2$

$S_{j,ini} = 7{,}71\ MNm/rad$

D.2.3 Hand calculation component method corner joint



k_1, k_2 and k_3 play a role

$k_1 = \dfrac{0.38\, A_{vc}}{\beta z}$ $A_{vc} = 1159\ mm^2$ (Table 6.4)

$z = h - t_f = 90 - 6.3 = 83.7$ (Fig 6.15)

$\beta \approx 1$ (5.3(7))

$k_1 = 5,26$

$k_2 = \dfrac{0.7\, b_{eff,t,wc} \cdot t_{wc}}{d_c}$, $d_c = h - 2t_f = 77.4\ mm$

$b_{eff,t,wc} = t_{fb} + 2\sqrt{2}\, a_b + 5(t_{fc} + s) = 51.9$ (6.16)

$k_2 = 2,96$

$k_3 = \infty$ for a stiffened welded connection
This is the case because the plates where the diagonals attach to act as stiffeners

$$\sum_i \frac{1}{k_i} = \frac{1}{5{,}26} + \frac{1}{2{,}96} + \cancel{\frac{1}{\infty}} = 0{,}53$$

$S_{j,ini} = 3.22$ MNm/rad

D.3 Joint stiffness values under various |M|/N ratios

The values for the stiffness under various |M|/N ratios are shown in the figures below. There are clear differences between the members in compression (negative values) and members in tension (positive values). It is observed that for certain joints the rotational stiffness is equal to 0 MNm/rad or ∞ MNm/rad. This ∞ MNm/rad value is denoted here as 10^{30} MNm/rad to be able to show it in a plot. The stiffness values within the analysed |M|/N values are computed for different sizes of the moment. The translational stiffness has a clear peak close to the value of 0.0 |M|/N for all joints. The relations are clear and have been tested for multiple load cases. For the rotational stiffness, there is no clear relation. In some cases, the rotational stiffness is infinitely large and in other cases, it is equal to 0.0 MNm/rad. These are unrealistic values. The rotation can not be infinitely large or equal to zero. In reality, the joint always allows for some rotation and a perfect hinge does not exist. There is always a small resisting force in a joint.

The results of the component method for the SHS and CHS splice and the corner joint from Appendix D.2 are also shown in the figures. These values differ from the results from IDEA StatiCa mainly due to the ∞ or 0 initial rotational stiffness computed with the software for the rotational stiffness. The values for translational stiffness are closer to the IDEA StatiCa results. The orders of magnitude are similar. This shows that the translational stiffness values computed by the software are reasonable.

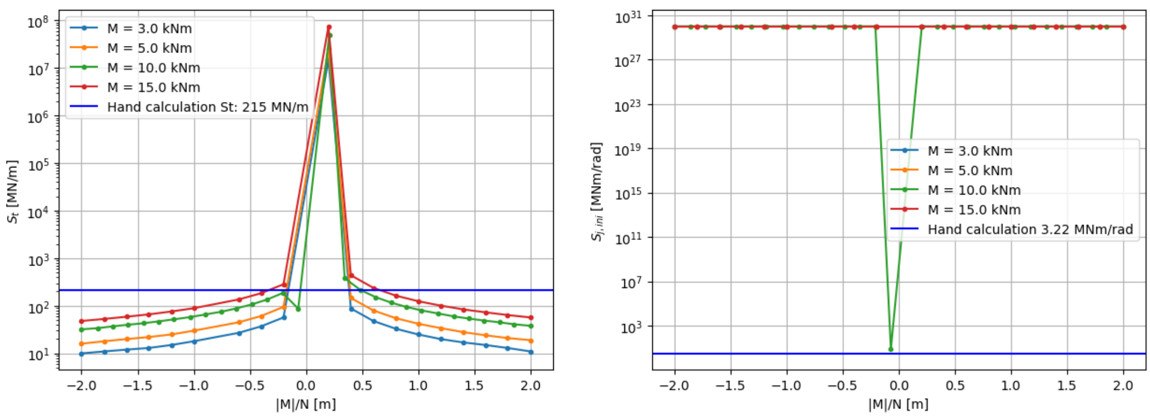


Figure D.1: SHS element to corner joint stiffness (connection type A, B)

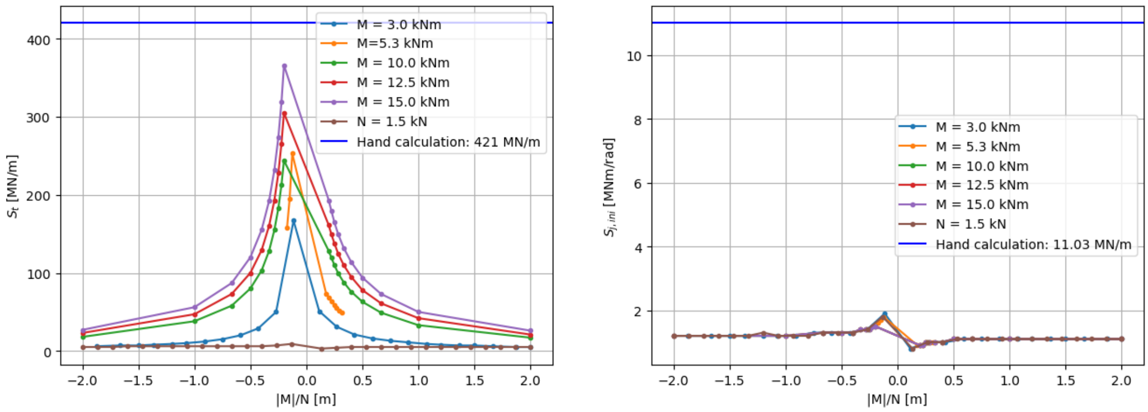


Figure D.2: Beam and column (SHS) splice stiffnesses (connection type C, D)

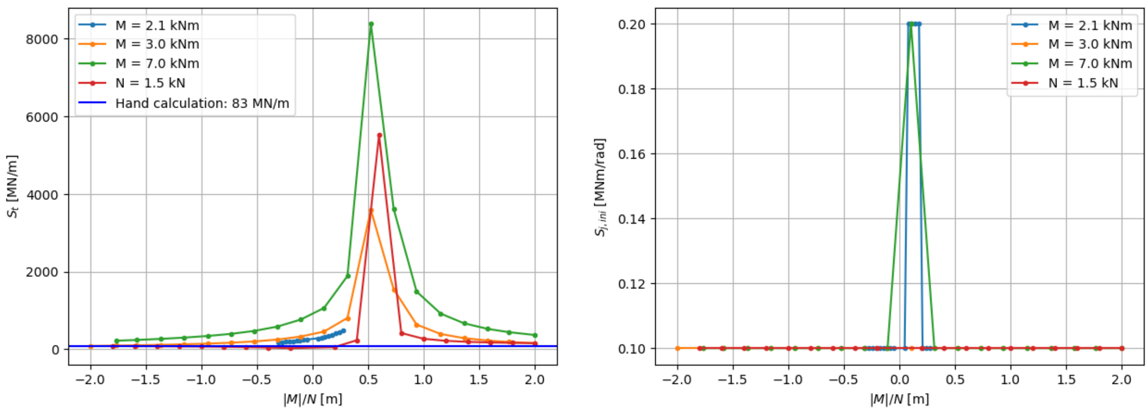


Figure D.3: Stiffness values for wind bracing connecting 4 diagonals (connection type E)

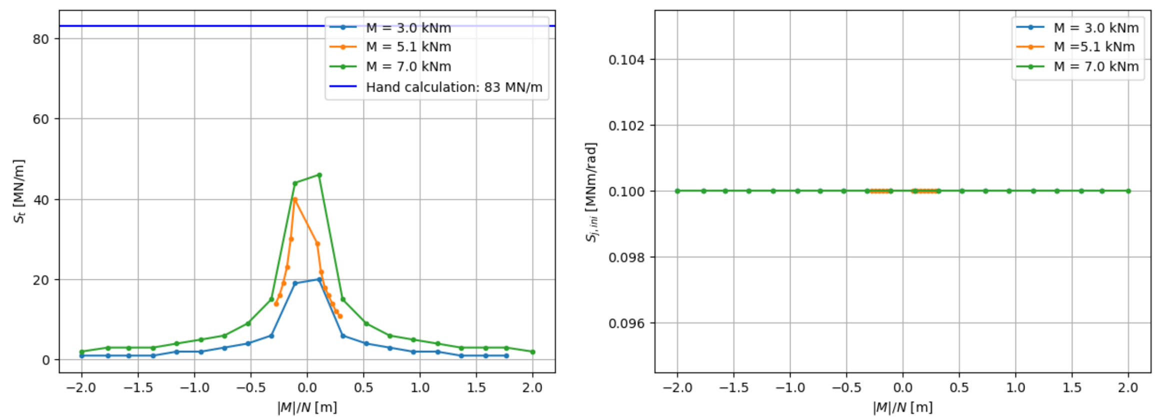


Figure D.4: Stiffness values for wind bracing connecting 8 diagonals (connection type F)

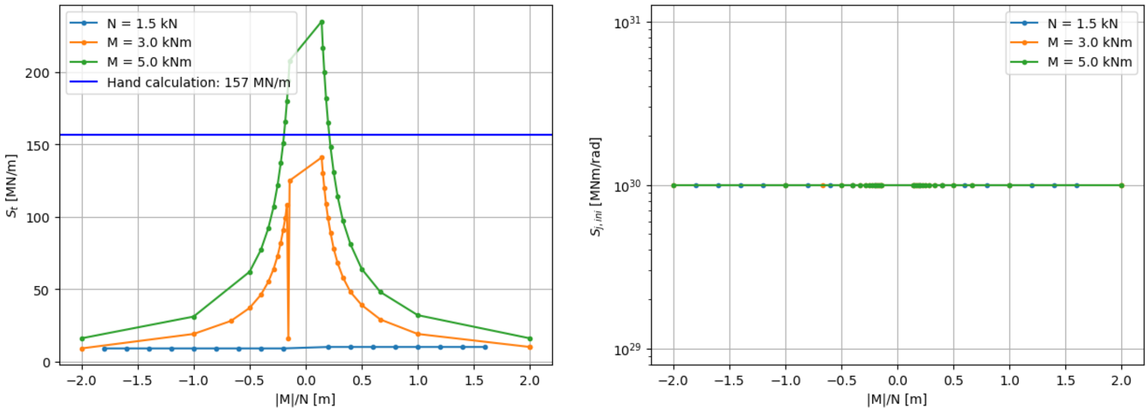


Figure D.5: Diagonal to corner point stiffnesses (connection type G)

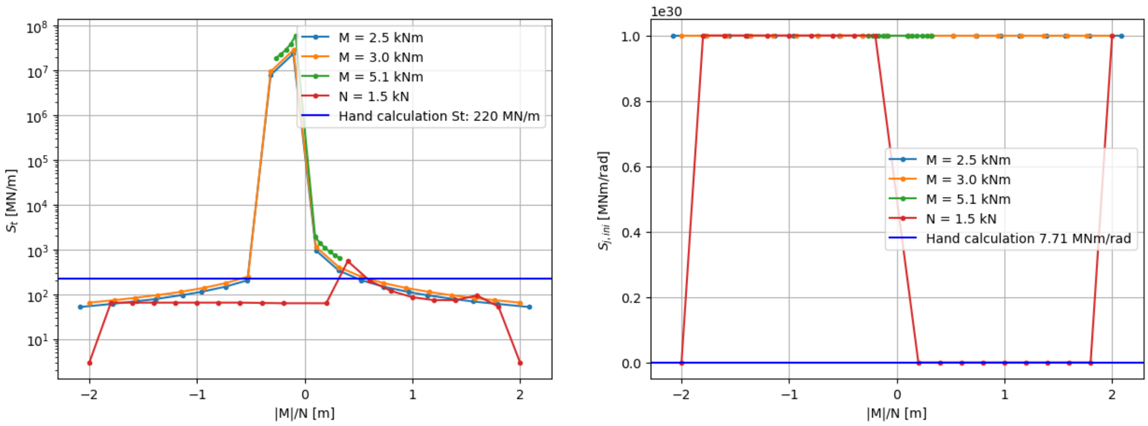


Figure D.6: Diagonal (CHS) splice stiffnesses (connection type H)

It is interesting to see what happens exactly for the initial rotational stiffness of the SHS splice connection. As can be seen from Figure D.7 there is a peak that shows some similarities with the peaks seen for translational stiffness. Around the |M|/N value of 0 MNm/rad there is a larger value for compression than for tension forces. Because of the compression, the rotation of the member is resisted more by the endplates of the connection which causes the rotational stiffness to be larger. In tension, the endplates are moving in opposite directions which increases the available space for the members to rotate. This leads to a lower translational stiffness. The peak is not caused by inconsistencies in the input data because it is observed in multiple load cases. The size of the loads influences the size of the peak but not its location. The decrease in stiffness for load ratios after the peaks does not represent structural behavior. It is not certain up to which point in the peak the results can still be trusted. According to research done by van Spengler the stiffness values up until the peak might still be reliable but there is uncertainty in the conclusion [58].

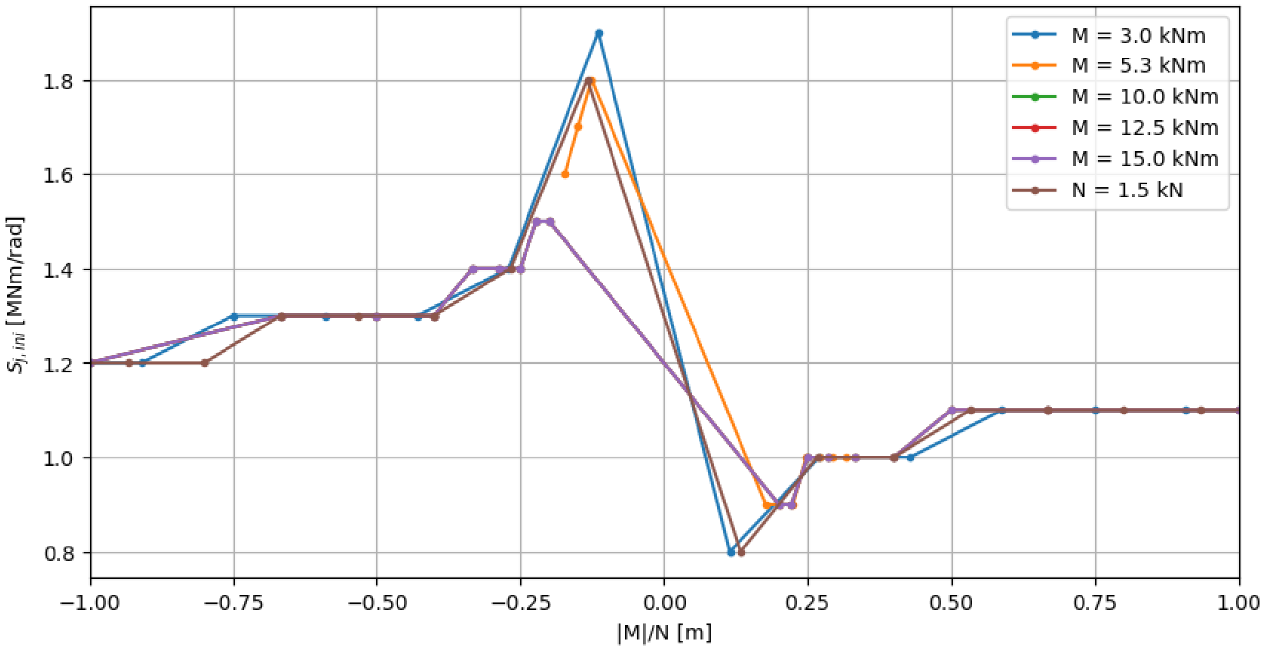


Figure D.7: Beam and column (SHS) splice stiffnesses zoomed in at the peak (connection type C, D)

The reason that IDEA StatiCa is not able to calculate rotational stiffness for the joints is because the joints are all very stiff. Besides, the |M|/N ratios are relatively small and IDEA StatiCa has trouble with calculating stiffness value for low internal forces. The connections can be classified as rigid. Because the slope of the moment-rotation diagram of the joints becomes too steep IDEA StatiCa cannot calculate this stiffness with precision anymore, see also Figure D.8. This can result in the values that go to infinity or zero for rotational stiffness. During conversations with an expert from IDEA StatiCa, this was the conclusion. The development team of IDEA StatiCa would like to improve the software on this but it is not the priority. Normally this is outside of the scope because joints that rigid do not have to be given a stiffness in the global model.

Instead of 3 bolts the joint can also be modeled with just 1 bolt. This results in a rotational stiffness of 52.6 MNm/rad which is still large. Without the diagonal and gusset plate the stiffness of the joint is 3.0 MNm/rad. This is close to the value of 3.2 MNm/rad calculated with the component method for this type of joint. To be safe the component method results are therefore used. These are conservative because the diagonals or gusset plates are not taken into account.

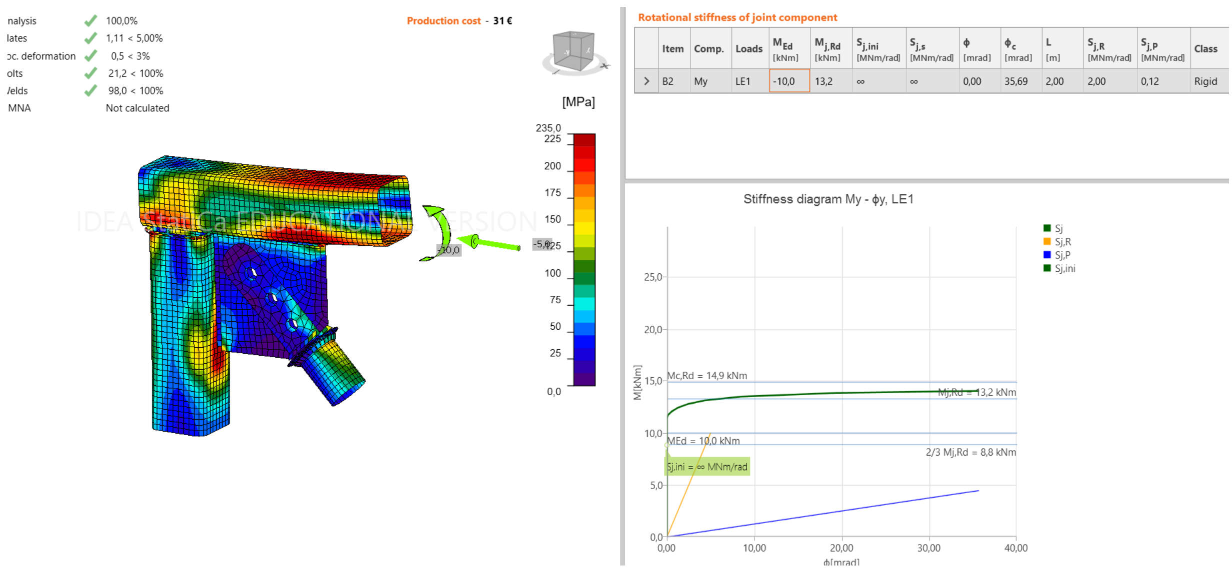


Figure D.8: Moment-rotation diagram for the diagonal to corner joint

D.4 Capped joint stiffness values under various |M|/N ratios

In this section, the joint stiffness values are capped. The peaks are lowered because they are not representative of the real joint behavior. The graphs for the translational joint stiffness are capped at certain values where the slope does not become too steep. The extreme values that do not follow the initial pattern of the graph are left out.

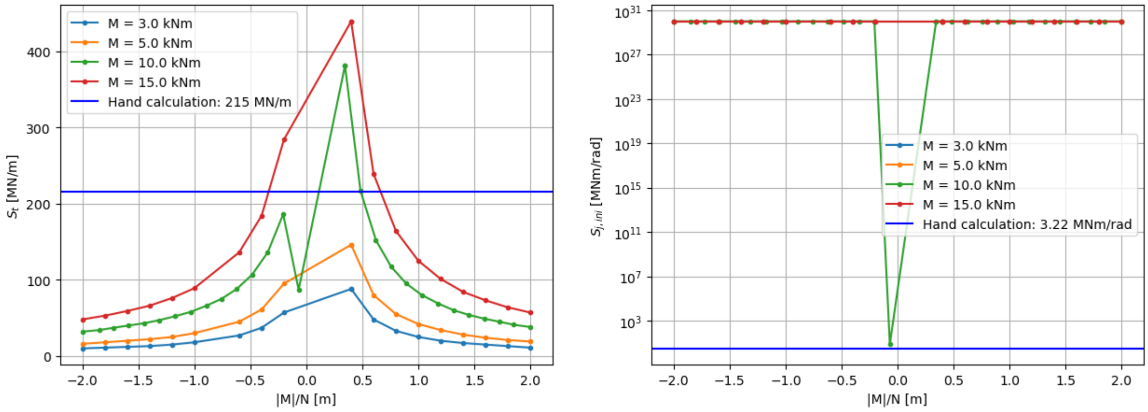


Figure D.9: SHS element to corner joint stiffness with capped values(connection type A, B)

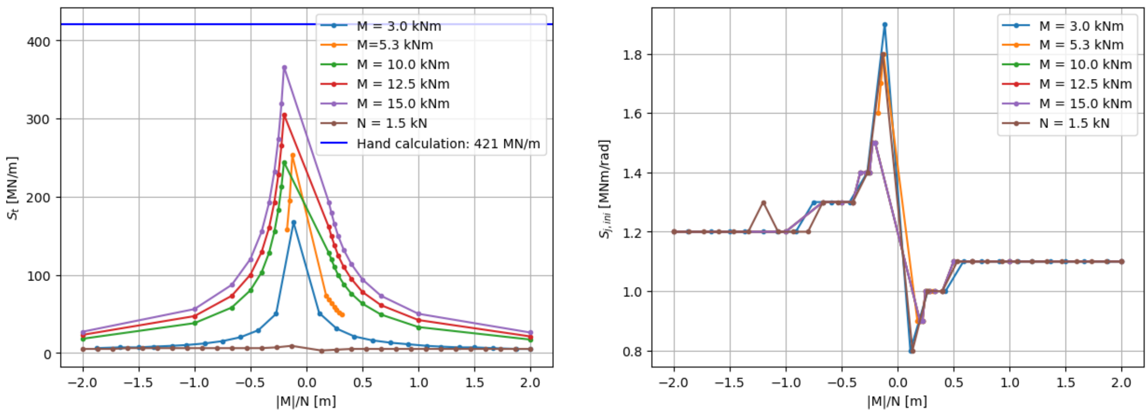


Figure D.10: Beam and column splice (SHS) stiffnesses with capped values (connection type C, D)

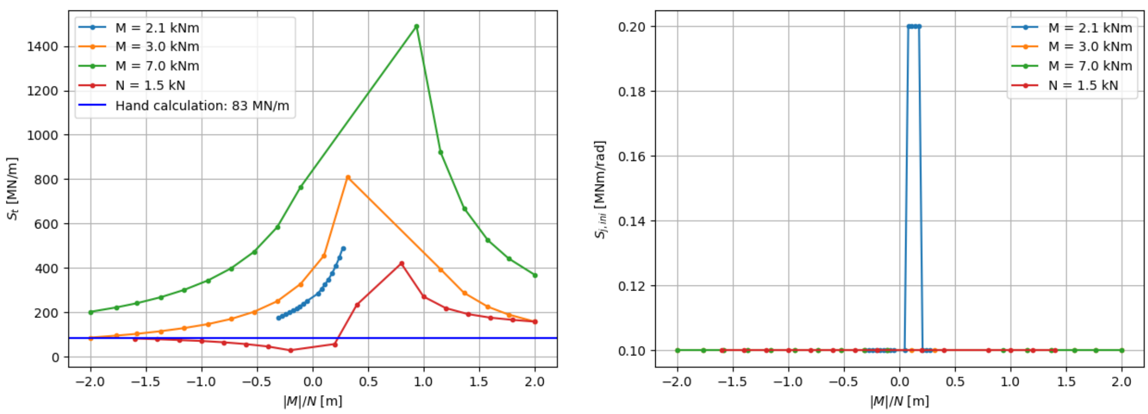


Figure D.11: Stiffness values for wind bracing connecting 4 diagonals with capped stiffness (connection type E)

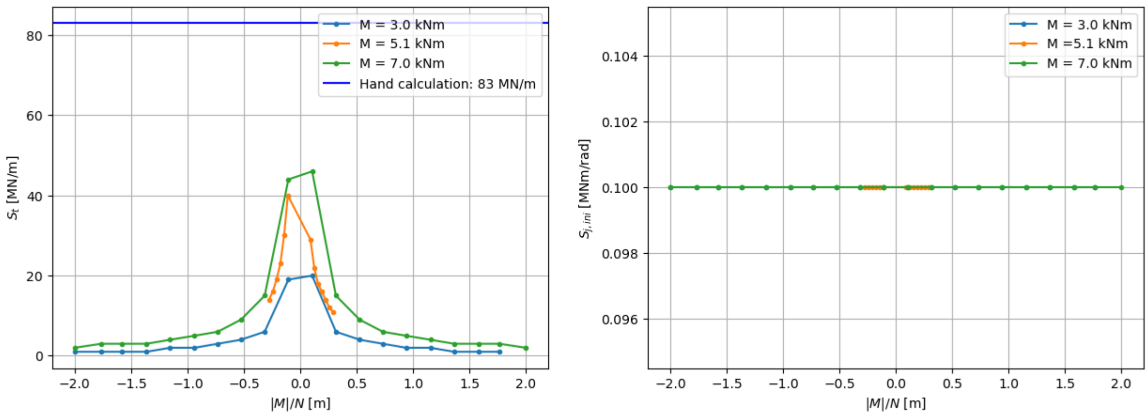


Figure D.12: Stiffness values for wind bracing connecting 8 diagonals with capped stiffness (connection type F)

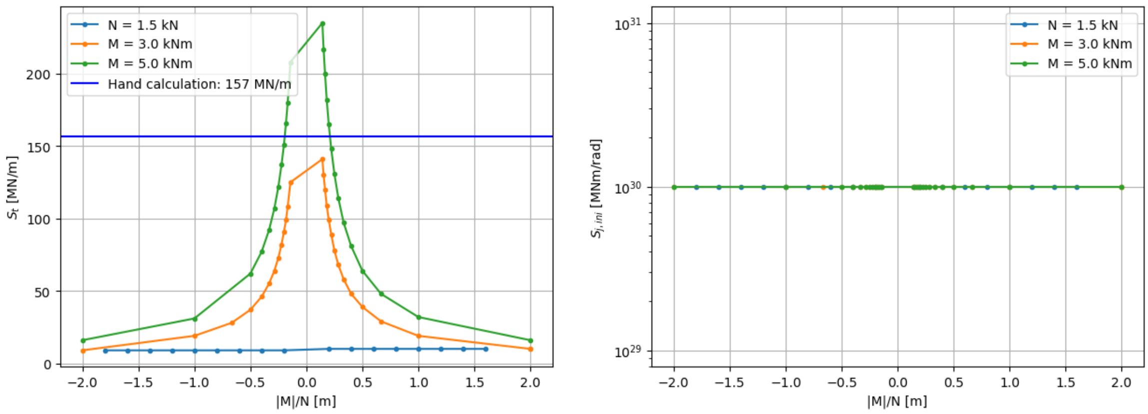


Figure D.13: Diagonal to corner point stiffnesses with capped values (connection type G)

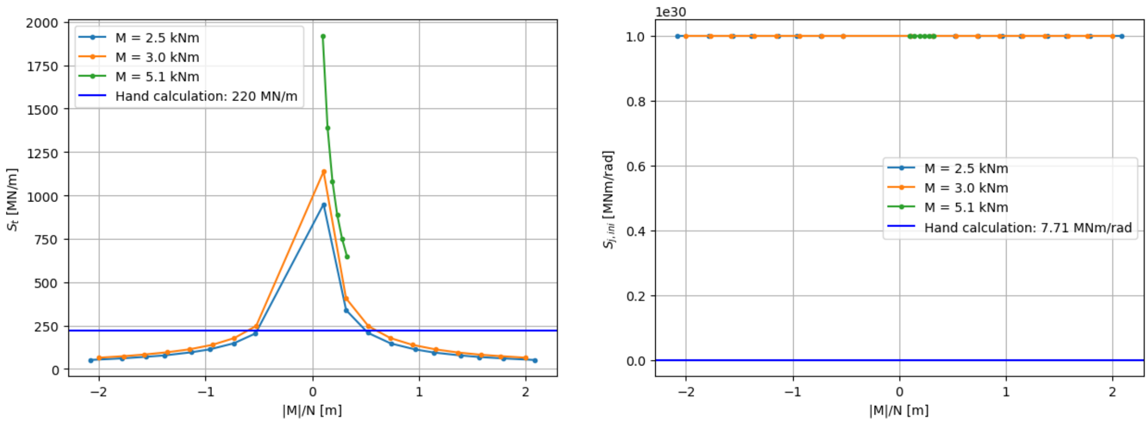


Figure D.14: Diagonal (CHS) splice stiffnesses (connection type H)

D.5 Joint stiffness values used in the FEM Model

In this section, the eventually used graphs for the joint stiffness are displayed. These graphs form an input for the FEM model in Grasshopper. The model selects a value on the graph based on the calculated |M|/N ratio during the stiffness iteration loop. If a calculated value of the ratio is located between two data points, the new stiffness will be the data point that has the lowest value of the two. This is visualised with a step function in the figures below. Doing it this way the assigned value for the stiffness will always be on or below the graph, resulting in conservative values. Values for |M|/N that are outside of the domain of the graph are assigned the value of the data point in the first or last position.

For the translational stiffness, the graphs that are in the middle of the different computed load scenarios shown in Appendix D.4 are used. These are conventional values where the joint does not fail. The values for the bending moment that is applied on the joint during the stiffness analysis are shown. The initial rotational stiffness of the joints is kept at the value computed with the component method. There are joints where the component method cannot be applied. These joints are very complex and numerous elements come together at these joints, making them stiff. Therefore, they are expected to be minimally as stiff as the splice connections with similar cross-sections. For this reason, they receive the same constant value for their rotational stiffness.

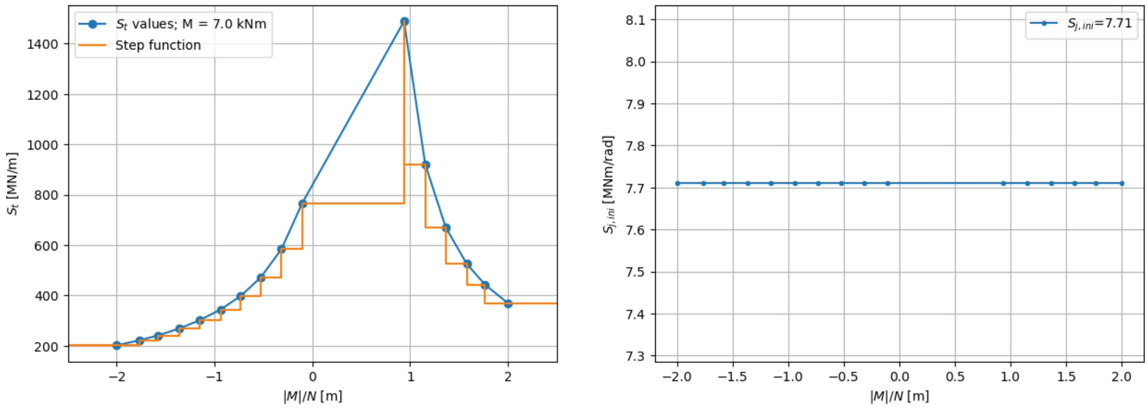


Figure D.15: Stiffness values for wind bracing connecting 4 diagonals stiffness used in model (connection type E)

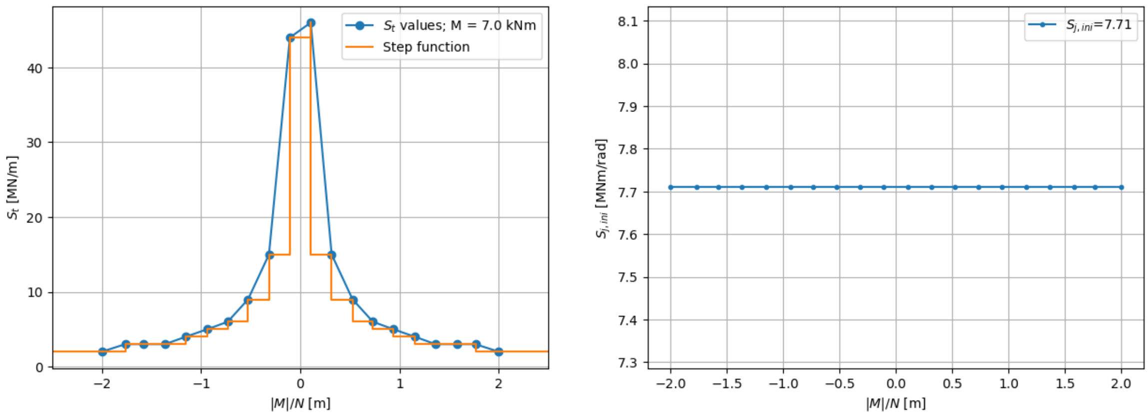


Figure D.16: Stiffness values for wind bracing connecting 8 diagonals used in model

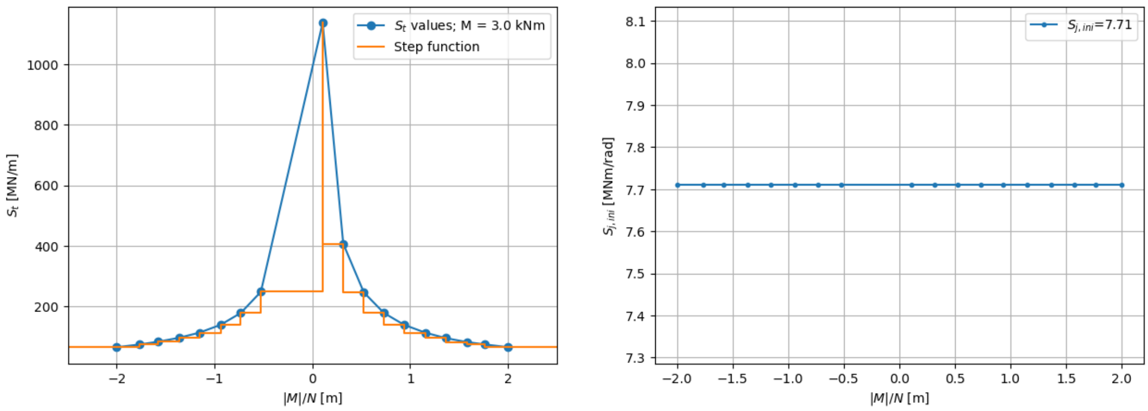


Figure D.17: Diagonal (CHS) splice stiffnesses used in model (connection type H)

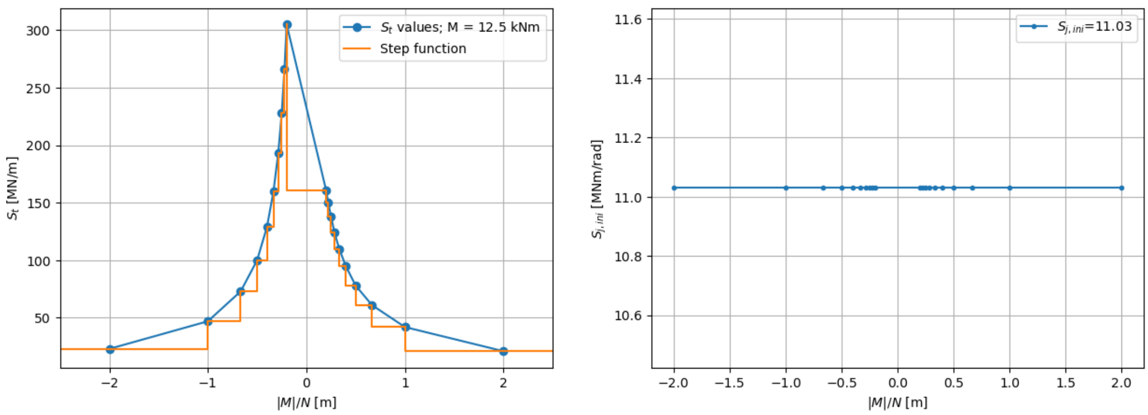


Figure D.18: Beam and column splice (SHS) stiffnesses used in model (connection type C, D)

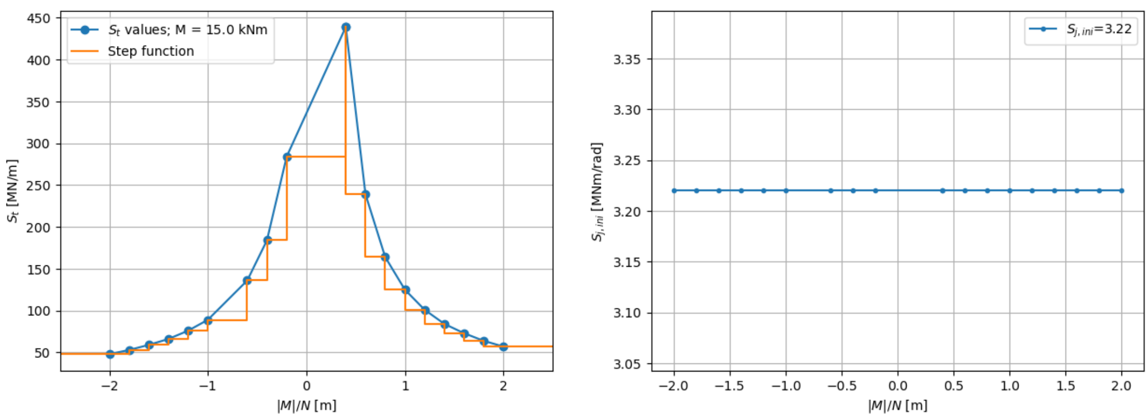


Figure D.19: SHS element to corner joint stiffness used in model (connection type A, B)

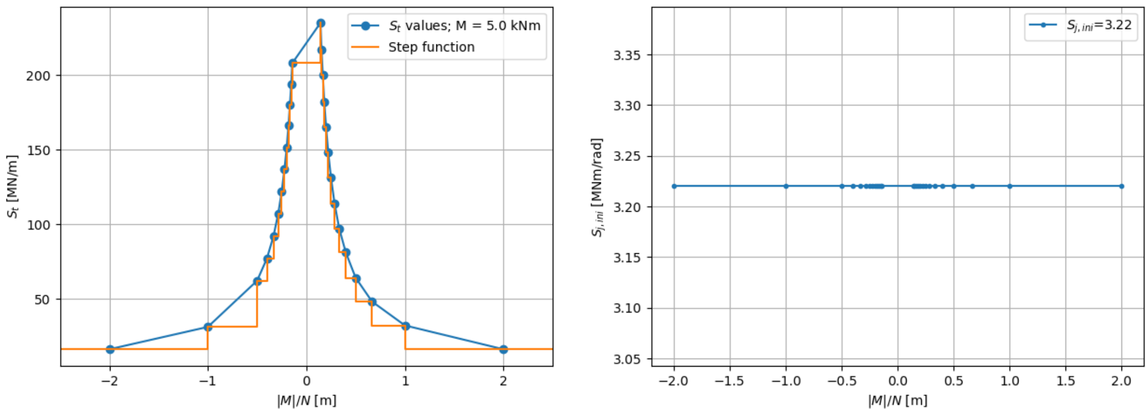
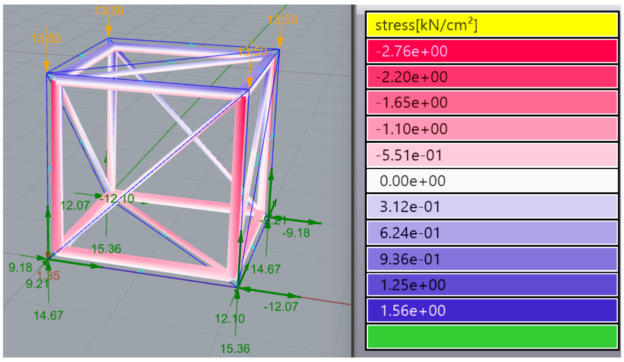


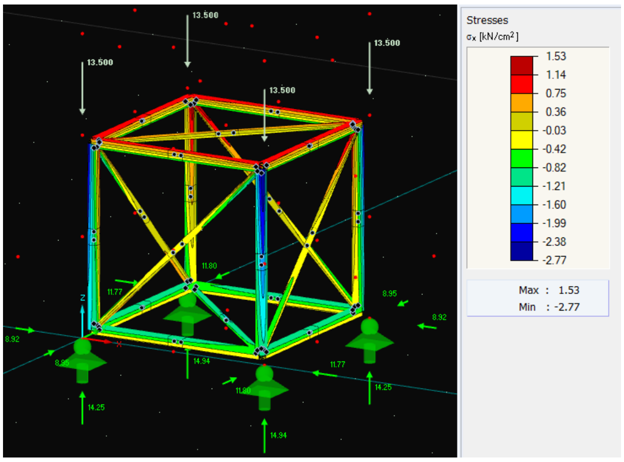
Figure D.20: Diagonal to corner point stiffnesses used in model (connection type G)

E Karamba3D and RFEM results

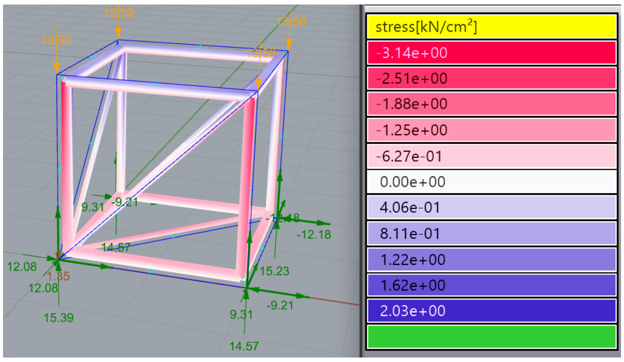
In this section, the comparison between the stress distributions of module types 1 to 6 in Karamba3D and RFEM 5 is shown. For all modules, the stress distributions are similar in both models, as can be seen in Figure E.1.



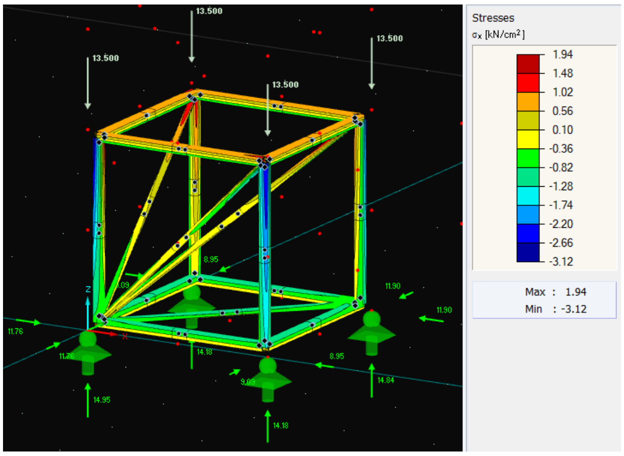
(a) Karamba3D model of module type 1



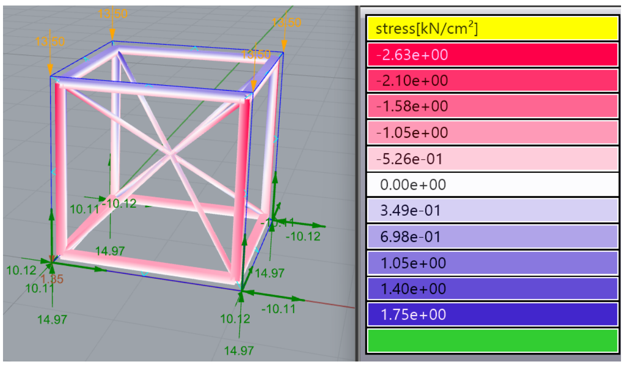
(b) RFEM model of module type 1



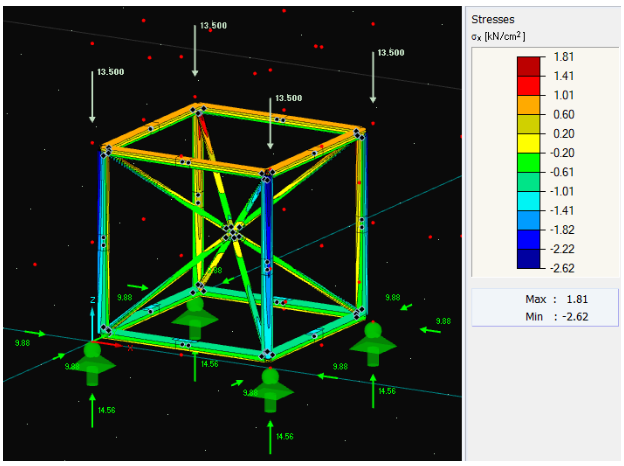
(c) Karamba3D model of module type 2



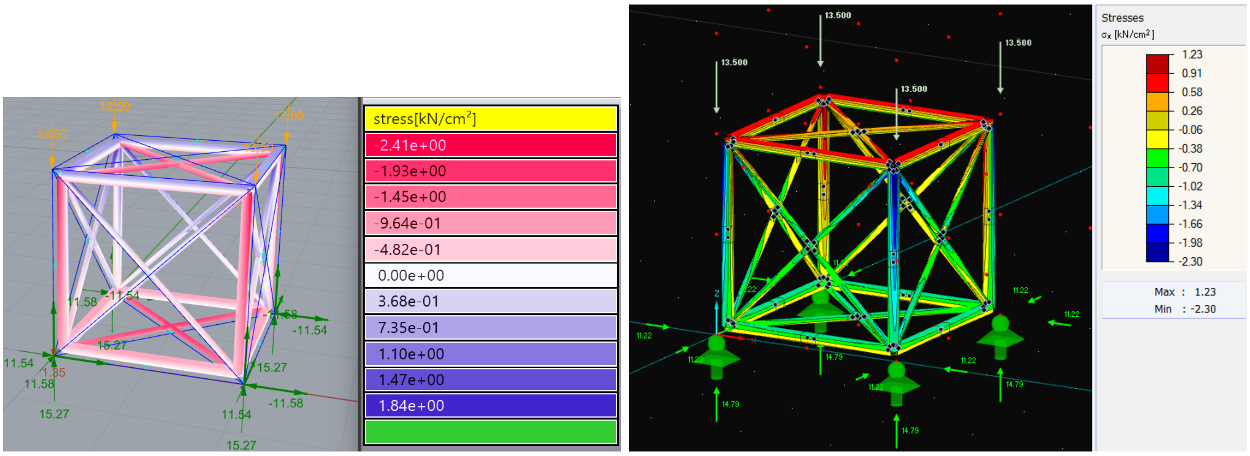
(d) RFEM model of module type 2



(e) Karamba3D model of module type 3

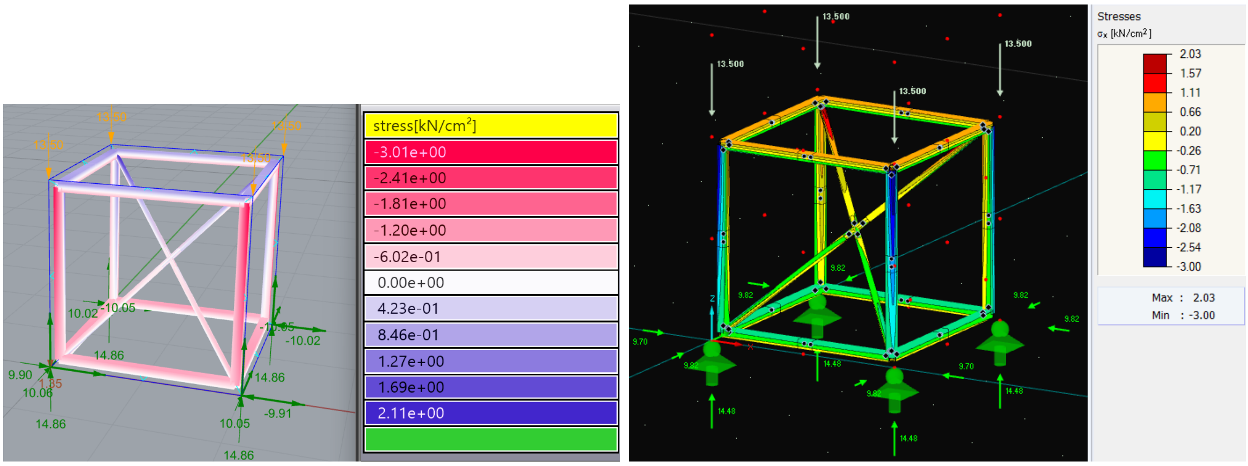


(f) RFEM model of module type 3



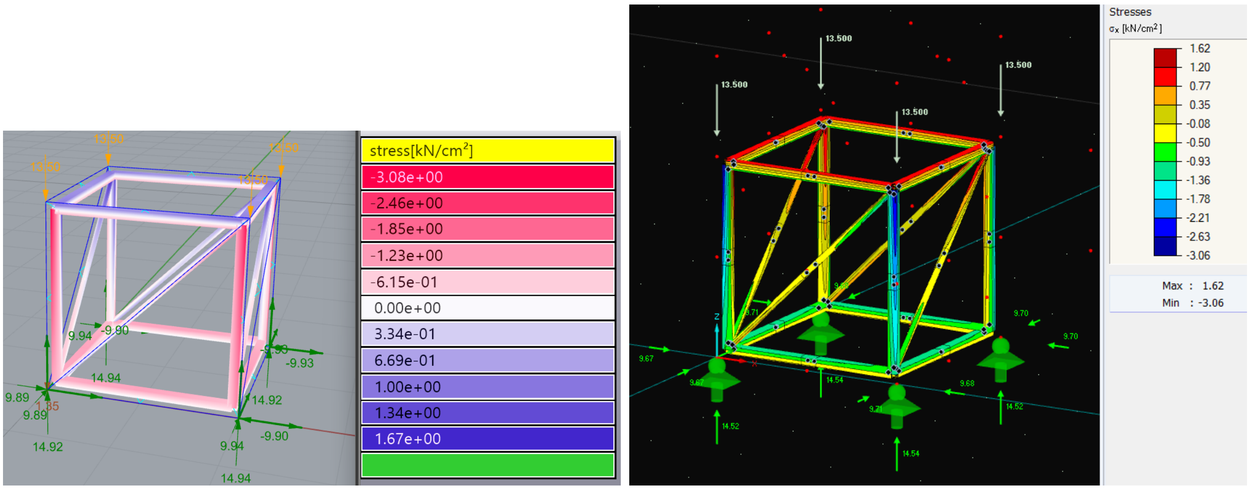
(g) Karamba3D model of module type 4

(h) RFEM model of module type 4



(i) Karamba3D model of module type 5

(j) RFEM model of module type 5



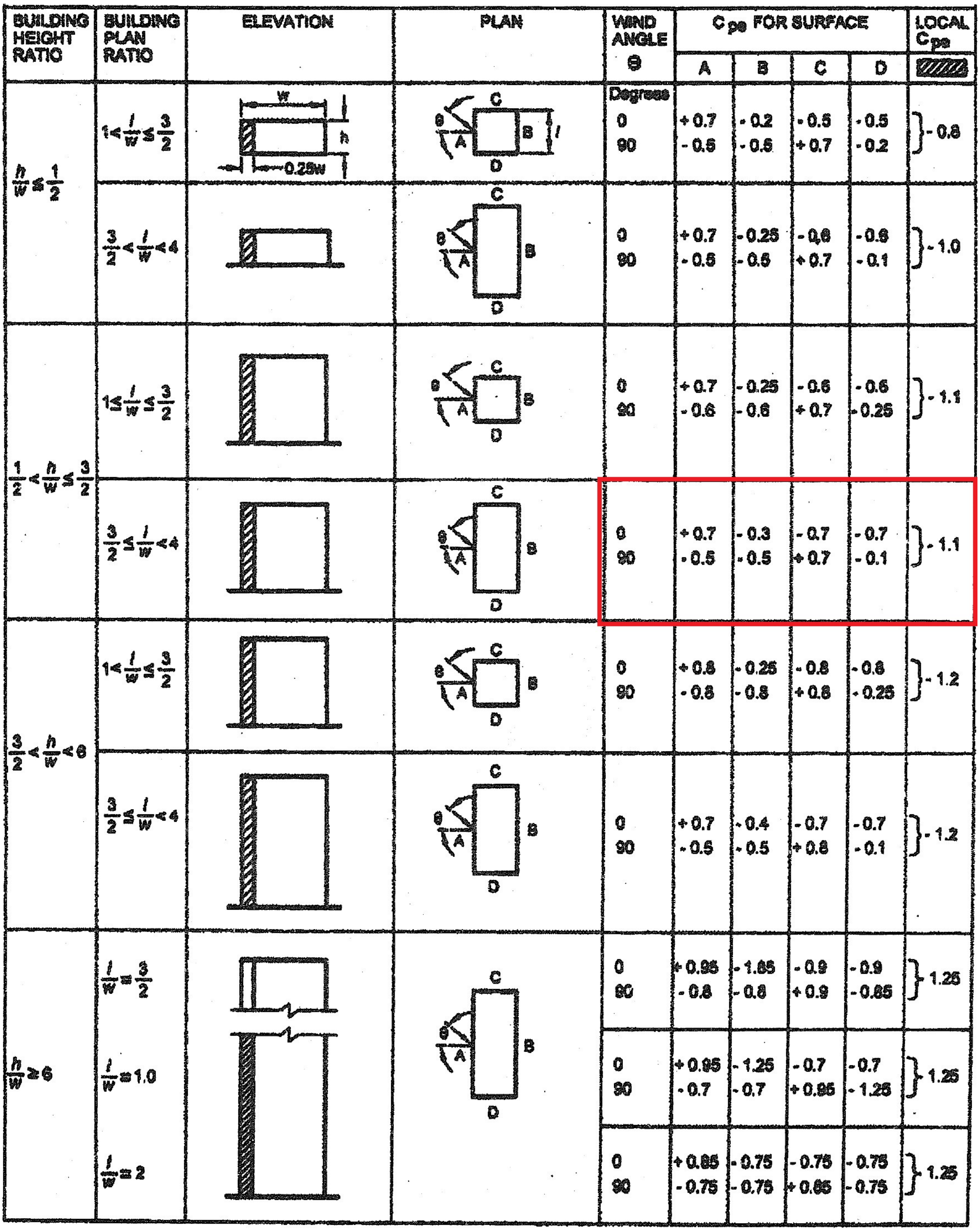
(k) Karamba3D model of module type 6

(l) RFEM model of module type 6

Figure E.1: Comparison of modules under loading in Karamba3D and RFEM

F External pressure coefficients for walls of rectangular grid buildings in according to Indian standards

The structure analysed by [60] has a height of 12 m, a width of 11 m and a length of 22 m. The ratio $\frac{h}{w}$ is 1.09 and the ratio $\frac{l}{w}$ is 2.00. Therefore, the values for the external pressure coefficients C_{pe} that are used for this building are the ones marked in red in Figure F.1. These values are multiplied with the design wind pressure to obtain the correct wind load values for the model.



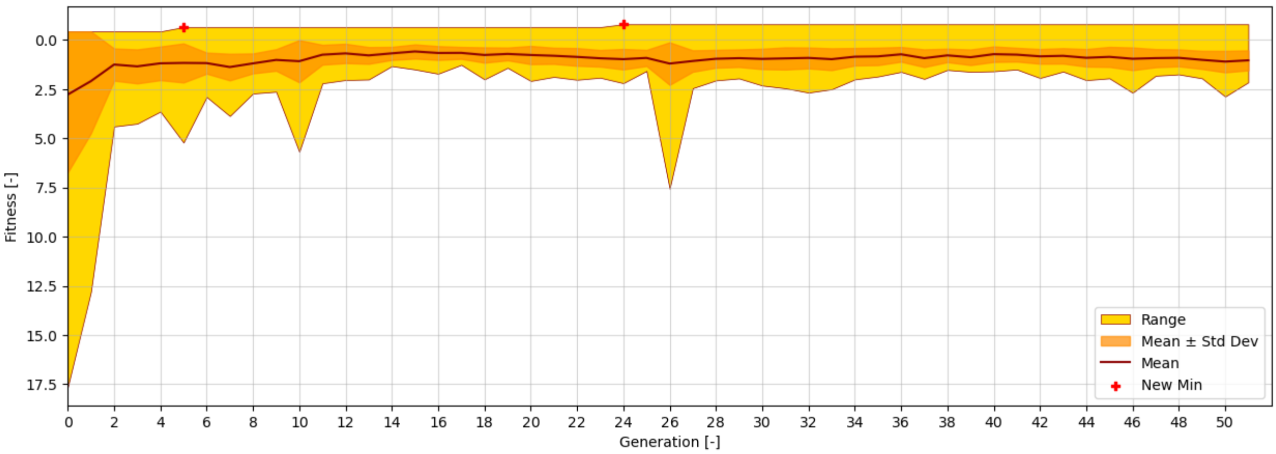
Building height ratio	Building plan ratio	Wind angle θ (degrees)	C_{pe} for surface A	B	C	D	Local C_{pe}
$\frac{h}{w} \leq \frac{1}{2}$	$1 < \frac{l}{w} \leq \frac{3}{2}$	0	+0.7	-0.2	-0.5	-0.5	-0.8
		90	-0.5	-0.5	+0.7	-0.2	
	$\frac{3}{2} < \frac{l}{w} < 4$	0	+0.7	-0.25	-0.6	-0.6	-1.0
		90	-0.5	-0.5	+0.7	-0.1	
$\frac{1}{2} < \frac{h}{w} \leq \frac{3}{2}$	$1 \leq \frac{l}{w} \leq \frac{3}{2}$	0	+0.7	-0.25	-0.6	-0.6	-1.1
		90	-0.6	-0.6	+0.7	-0.25	
	$\frac{3}{2} \leq \frac{l}{w} < 4$	0	+0.7	-0.3	-0.7	-0.7	-1.1
		90	-0.5	-0.5	+0.7	-0.1	
$\frac{3}{2} < \frac{h}{w} < 6$	$1 < \frac{l}{w} \leq \frac{3}{2}$	0	+0.8	-0.25	-0.8	-0.8	-1.2
		90	-0.8	-0.8	+0.8	-0.25	
	$\frac{3}{2} \leq \frac{l}{w} < 4$	0	+0.7	-0.4	-0.7	-0.7	-1.2
		90	-0.5	-0.5	+0.8	-0.1	
$\frac{h}{w} \geq 6$	$\frac{l}{w} = \frac{3}{2}$	0	+0.95	-1.85	-0.9	-0.9	1.25
		90	-0.8	-0.8	+0.9	-0.85	
	$\frac{l}{w} = 1.0$	0	+0.95	-1.25	-0.7	-0.7	1.25
		90	-0.7	-0.7	+0.95	-1.25	
	$\frac{l}{w} = 2$	0	+0.85	-0.75	-0.75	-0.75	1.25
		90	-0.75	-0.75	+0.85	-0.75	

Figure F.1: External pressure coefficients for walls of rectangular clad buildings (Source: Bureau of Indian standards [61])

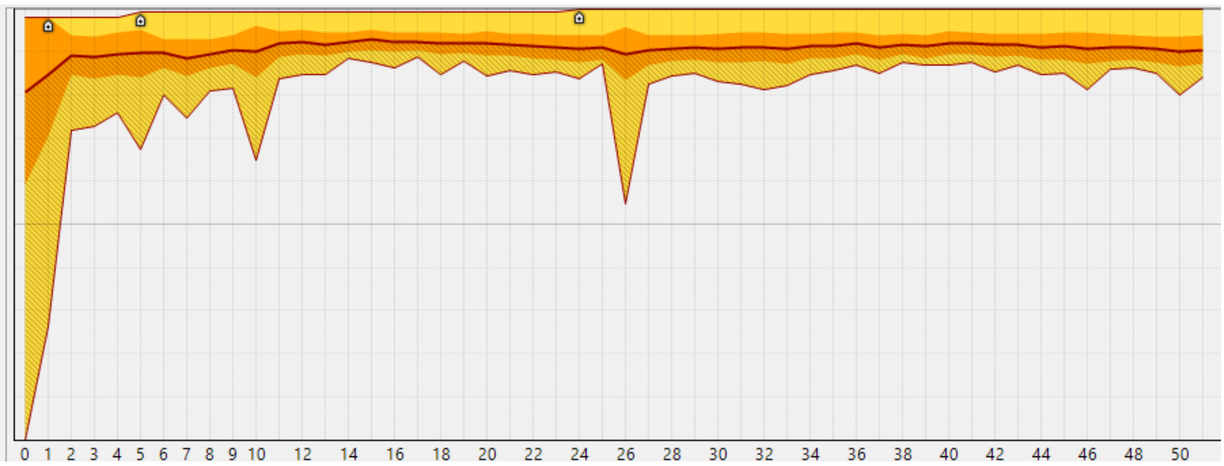
G Validation convergence graph made in Python

In this section the validation of the convergence graphs that are made with programming in Python is validated. Figure G.1 shows both a graph taken from a screenshot in Galapagos and a graph made with Python. The Python and Galapagos graphs clearly show the same shape for the mean, the standard deviation around the mean, and the range of extreme values for each generation. The new minimal fitness values also are the same. The total number of these improvements and their position on the graph are identical.

The graph in Python is made by grouping the data into bins that contain all the values of a generation. If there are outliers in the data these are trimmed. This is done by removing the last two values for each generation except for the first one. Then the mean and range in values are calculated as well as the standard deviation. These are all plotted along with the cumulative best fitness value. The results of most of the convergence graphs look similar to the ones produced by Galapagos. There are also minor differences for some of the figures, especially for the range of values. However, the trend is very clearly visualised and the new best fitness values correspond to the ones in Galapagos.



(a) Convergence graph Python



(b) Convergence graph Galapagos

Figure G.1: Comparison of the convergence graph for verification from Python and Grasshopper

H IDEA StatiCa stiffness reports

In this section, the different reports that are made in IDEA StatiCa are displayed. For every joint type calculations have been performed. These calculations were done for different combinations of moment and normal force. The ranges used for these ratios are taken from Appendix C. The normal force is varied and the moment is a single value in these ratios. The maximum combination of normal force and moment is not exceeded. In the grasshopper model, the maximum bending moments occur in the nodes. There is a gradient of the moments in the beams. However, in IDEA StatiCa, the maximum applied bending moments are applied at the beam ends in front of the nodes. When the obtained nodal moments are applied in an IDEA StatiCa model, these are larger than the actual moments on the beams in the Grasshopper model. Therefore, the maximum bending moment may be suddenly exceeded. It is checked for each cross-section that the applied moment in IDEA StatiCa is not larger than the limit for plastic bending moment.

$$M_{applied} \leq W_{pl} \cdot f_y \tag{H.1}$$

In which:

$M_{applied}$ = Applied bending moment [kNm]

W_{pl} = Plastic section modulus [m^3]

f_y = Yield strength [kN/m^2]

Table H.1: Plastic moment capacity of the cross-sections

Cross-section	W_{pl} **[mm^3]**	M_{pl} **[kNm]**
CHS 60.3 x 3.2	10400	2.44
SHS 90.0 x 6.0	59520	13.99

In IDEA StatiCa the cross-section SHS 90.0 x 6.0 is unavailable. Therefore, SHS 90.0 x 6.3 is used for calculations.

H.1 Report CHS diagonal splice

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	23-4-2024
Code	EN

Material

Steel	S 355, S 235

Results

Summary

Name	Parameters	Results		
CHS Splice connection final	CHS60.3/3.2 S 235 M18 8.8 Stiffness	Analysed member	B2	
		Sj,ini(My, LE1)	∞ MNm/rad	Rigid
		Sj,ini(My, LE2)	∞ MNm/rad	Rigid
		Sj,ini(My, LE3)	∞ MNm/rad	Rigid
		Sj,ini(My, LE4)	∞ MNm/rad	Rigid
		Sj,ini(My, LE5)	∞ MNm/rad	Rigid
		Sj,ini(My, LE6)	∞ MNm/rad	Rigid
		Sj,ini(My, LE7)	∞ MNm/rad	Rigid
		Sj,ini(My, LE8)	∞ MNm/rad	Rigid
		Sj,ini(My, LE9)	∞ MNm/rad	Rigid
		Sj,ini(My, LE10)	∞ MNm/rad	Rigid
		Sj,ini(My, LE11)	∞ MNm/rad	Rigid
		Sj,ini(My, LE12)	∞ MNm/rad	Rigid
		Sj,ini(My, LE13)	∞ MNm/rad	Rigid
		Sj,ini(My, LE14)	∞ MNm/rad	Rigid
		Sj,ini(My, LE15)	∞ MNm/rad	Rigid
		Sj,ini(My, LE16)	∞ MNm/rad	Rigid
		Sj,ini(My, LE17)	∞ MNm/rad	Rigid
		Sj,ini(My, LE18)	∞ MNm/rad	Rigid
		Sj,ini(My, LE19)	∞ MNm/rad	Rigid
		Sj,ini(My, LE20)	∞ MNm/rad	Rigid
		Sj,ini(My, LE21)	∞ MNm/rad	Rigid
		Sj,ini(My, LE22)	∞ MNm/rad	Rigid
		Sj,ini(My, LE23)	∞ MNm/rad	Rigid
		Sj,ini(My, LE24)	∞ MNm/rad	Rigid
		Sj,ini(My, LE25)	∞ MNm/rad	Rigid
		Sj,ini(My, LE26)	∞ MNm/rad	Rigid
		Sj,ini(My, LE27)	∞ MNm/rad	Rigid
		Sj,ini(My, LE28)	∞ MNm/rad	Rigid
		Sj,ini(My, LE29)	∞ MNm/rad	Rigid
		Sj,ini(My, LE30)	∞ MNm/rad	Rigid
		Sj,ini(My, LE31)	∞ MNm/rad	Rigid
		Sj,ini(My, LE32)	∞ MNm/rad	Rigid
		Sj,ini(My, LE33)	∞ MNm/rad	Rigid
		Sj,ini(My, LE34)	∞ MNm/rad	Rigid
		Sj,ini(My, LE35)	∞ MNm/rad	Rigid
		Sj,ini(My, LE36)	∞ MNm/rad	Rigid
		Sj,ini(My, LE37)	∞ MNm/rad	Rigid
		Sj,ini(My, LE38)	∞ MNm/rad	Rigid
		Sj,ini(My, LE39)	∞ MNm/rad	Rigid
		Sj,ini(My, LE40)	∞ MNm/rad	Rigid
		Sj,ini(My, LE41)	∞ MNm/rad	Rigid
		Sj,ini(My, LE42)	∞ MNm/rad	Rigid
		Sj,ini(My, LE43)	∞ MNm/rad	Rigid
		Sj,ini(My, LE44)	∞ MNm/rad	Rigid
		Sj,ini(My, LE45)	∞ MNm/rad	Rigid
		Sj,ini(My, LE46)	∞ MNm/rad	Rigid
		Sj,ini(My, LE47)	∞ MNm/rad	Rigid
		Sj,ini(My, LE48)	∞ MNm/rad	Rigid
		Sj,ini(My, LE49)	∞ MNm/rad	Rigid
		Sj,ini(My, LE50)	∞ MNm/rad	Rigid
		Sj,ini(My, LE51)	∞ MNm/rad	Rigid
		Sj,ini(My, LE52)	∞ MNm/rad	Rigid
		Sj,ini(My, LE53)	∞ MNm/rad	Rigid
		Sj,ini(My, LE54)	∞ MNm/rad	Rigid
		Sj,ini(My, LE55)	∞ MNm/rad	Rigid
		Sj,ini(My, LE56)	∞ MNm/rad	Rigid
		Sj,ini(My, LE57)	∞ MNm/rad	Rigid
		Sj,ini(My, LE58)	∞ MNm/rad	Rigid
		Sj,ini(My, LE59)	∞ MNm/rad	Rigid
		Sj,ini(My, LE60)	∞ MNm/rad	Rigid
		Sj,ini(My, LE61)	∞ MNm/rad	Rigid
		Sj,ini(My, LE62)	∞ MNm/rad	Rigid
		Sj,ini(My, LE63)	∞ MNm/rad	Rigid
		Sj,ini(My, LE64)	∞ MNm/rad	Rigid
		Sj,ini(My, LE65)	∞ MNm/rad	Rigid
		Sj,ini(My, LE66)	∞ MNm/rad	Rigid
		Sj,ini(My, LE67)	∞ MNm/rad	Rigid

Name	Parameters	Results		
		Sj,ini(My, LE68)	∞ MNm/rad	Rigid
		Sj,ini(My, LE69)	∞ MNm/rad	Rigid
		Sj,ini(My, LE70)	∞ MNm/rad	Rigid
		Sj,ini(My, LE71)	∞ MNm/rad	Rigid
		Sj,ini(My, LE72)	∞ MNm/rad	Rigid
		Sj,ini(My, LE73)	∞ MNm/rad	Rigid
		Sj(N, LE1)	78 MN/m	
		Sj(N, LE2)	78 MN/m	
		Sj(N, LE3)	78 MN/m	
		Sj(N, LE4)	78 MN/m	
		Sj(N, LE5)	78 MN/m	
		Sj(N, LE6)	78 MN/m	
		Sj(N, LE7)	78 MN/m	
		Sj(N, LE8)	78 MN/m	
		Sj(N, LE9)	77 MN/m	
		Sj(N, LE10)	75 MN/m	
		Sj(N, LE11)	75 MN/m	
		Sj(N, LE12)	74 MN/m	
		Sj(N, LE13)	77 MN/m	
		Sj(N, LE14)	77 MN/m	
		Sj(N, LE15)	77 MN/m	
		Sj(N, LE16)	78 MN/m	
		Sj(N, LE17)	78 MN/m	
		Sj(N, LE18)	78 MN/m	
		Sj(N, LE19)	78 MN/m	
		Sj(N, LE20)	78 MN/m	
		Sj(N, LE21)	78 MN/m	
		Sj(N, LE22)	63 MN/m	
		Sj(N, LE23)	73 MN/m	
		Sj(N, LE24)	84 MN/m	
		Sj(N, LE25)	94 MN/m	
		Sj(N, LE26)	115 MN/m	
		Sj(N, LE27)	136 MN/m	
		Sj(N, LE28)	179 MN/m	
		Sj(N, LE29)	247 MN/m	
		Sj(N, LE30)	7900000 MN/m	
		Sj(N, LE31)	23700001 MN/m	
		Sj(N, LE32)	1066 MN/m	
		Sj(N, LE33)	405 MN/m	
		Sj(N, LE34)	248 MN/m	
		Sj(N, LE35)	176 MN/m	
		Sj(N, LE36)	135 MN/m	
		Sj(N, LE37)	114 MN/m	
		Sj(N, LE38)	94 MN/m	
		Sj(N, LE39)	83 MN/m	
		Sj(N, LE40)	73 MN/m	
		Sj(N, LE41)	63 MN/m	
		Sj(N, LE42)	79 MN/m	
		Sj(N, LE43)	89 MN/m	
		Sj(N, LE44)	100 MN/m	
		Sj(N, LE45)	115 MN/m	
		Sj(N, LE46)	136 MN/m	
		Sj(N, LE47)	168 MN/m	
		Sj(N, LE48)	215 MN/m	
		Sj(N, LE49)	300 MN/m	
		Sj(N, LE50)	9500000 MN/m	
		Sj(N, LE51)	28500001 MN/m	
		Sj(N, LE52)	1278 MN/m	
		Sj(N, LE53)	486 MN/m	
		Sj(N, LE54)	295 MN/m	
		Sj(N, LE55)	213 MN/m	
		Sj(N, LE56)	166 MN/m	
		Sj(N, LE57)	135 MN/m	
		Sj(N, LE58)	114 MN/m	
		Sj(N, LE59)	99 MN/m	
		Sj(N, LE60)	89 MN/m	
		Sj(N, LE61)	78 MN/m	
		Sj(N, LE62)	16400000 MN/m	

Name	Parameters	Results		
		Sj(N, LE63)	19200001 MN/m	
		Sj(N, LE64)	23200001 MN/m	
		Sj(N, LE65)	29200001 MN/m	
		Sj(N, LE66)	39400001 MN/m	
		Sj(N, LE67)	60600002 MN/m	
		Sj(N, LE68)	2295 MN/m	
		Sj(N, LE69)	1701 MN/m	
		Sj(N, LE70)	1339 MN/m	
		Sj(N, LE71)	1101 MN/m	
		Sj(N, LE72)	930 MN/m	
		Sj(N, LE73)	804 MN/m	

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_{C}	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5

H.2 Report SHS beam splice

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	23-4-2024
Code	EN

Material

Steel	S 355, S 235, S 275

Results

Summary

1 / 6

Name	Parameters	Results		
SHS column splice connection final	SHS90/90/6.3 S 235 Edit of M18 8.8 Stiffness	Analysed member	B2	
		Sj,ini(My, LE1)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE2)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE3)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE4)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE5)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE6)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE7)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE8)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE9)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE10)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE11)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE12)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE13)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE14)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE15)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE16)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE17)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE18)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE19)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE20)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE21)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE22)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE23)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE24)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE25)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE26)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE27)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE28)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE29)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE30)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE31)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE32)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE33)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE34)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE35)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE36)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE37)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE38)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE39)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE40)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE41)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE42)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE43)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE44)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE45)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE46)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE47)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE48)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE49)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE50)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE51)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE52)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE53)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE54)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE55)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE56)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE57)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE58)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE59)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE60)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE61)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE62)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE63)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE64)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE65)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE66)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE67)	0.0 MNm/rad	Pinned

Name	Parameters	Results		
		Sj,ini(My, LE68)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE69)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE70)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE71)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE72)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE73)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE74)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE75)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE76)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE77)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE78)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE79)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE80)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE81)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE82)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE83)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE84)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE85)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE86)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE87)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE88)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE89)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE90)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE91)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE92)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE93)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE94)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE95)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE96)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE97)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE98)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE99)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE100)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE101)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE102)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE103)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE104)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE105)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE106)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE107)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE108)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE109)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE110)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE111)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE112)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE113)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE114)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE115)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE116)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE117)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE118)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE119)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE120)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE121)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE122)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE123)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE124)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE125)	0.0 MNm/rad	Pinned
		Sj,ini(My, LE126)	0.0 MNm/rad	Pinned
		Sj(N, LE1)	31 MN/m	
		Sj(N, LE2)	35 MN/m	
		Sj(N, LE3)	41 MN/m	
		Sj(N, LE4)	48 MN/m	
		Sj(N, LE5)	56 MN/m	
		Sj(N, LE6)	67 MN/m	
		Sj(N, LE7)	82 MN/m	
		Sj(N, LE8)	102 MN/m	
		Sj(N, LE9)	131 MN/m	

Name	Parameters	Results		
		Sj(N, LE10)	174 MN/m	
		Sj(N, LE11)	243 MN/m	
		Sj(N, LE12)	364 MN/m	
		Sj(N, LE13)	606 MN/m	
		Sj(N, LE14)	1205 MN/m	
		Sj(N, LE15)	3504 MN/m	
		Sj(N, LE16)	67692 MN/m	
		Sj(N, LE17)	3335 MN/m	
		Sj(N, LE18)	1162 MN/m	
		Sj(N, LE19)	590 MN/m	
		Sj(N, LE20)	357 MN/m	
		Sj(N, LE21)	240 MN/m	
		Sj(N, LE22)	172 MN/m	
		Sj(N, LE23)	129 MN/m	
		Sj(N, LE24)	592 MN/m	
		Sj(N, LE25)	81 MN/m	
		Sj(N, LE26)	67 MN/m	
		Sj(N, LE27)	56 MN/m	
		Sj(N, LE28)	47 MN/m	
		Sj(N, LE29)	41 MN/m	
		Sj(N, LE30)	35 MN/m	
		Sj(N, LE31)	10 MN/m	
		Sj(N, LE32)	18 MN/m	
		Sj(N, LE33)	25 MN/m	
		Sj(N, LE34)	31 MN/m	
		Sj(N, LE35)	36 MN/m	
		Sj(N, LE36)	40 MN/m	
		Sj(N, LE37)	44 MN/m	
		Sj(N, LE38)	47 MN/m	
		Sj(N, LE39)	49 MN/m	
		Sj(N, LE40)	50 MN/m	
		Sj(N, LE41)	345 MN/m	
		Sj(N, LE42)	254 MN/m	
		Sj(N, LE43)	190 MN/m	
		Sj(N, LE44)	144 MN/m	
		Sj(N, LE45)	109 MN/m	
		Sj(N, LE46)	81 MN/m	
		Sj(N, LE47)	58 MN/m	
		Sj(N, LE48)	40 MN/m	
		Sj(N, LE49)	24 MN/m	
		Sj(N, LE50)	11 MN/m	
		Sj(N, LE51)	4 MN/m	
		Sj(N, LE52)	7 MN/m	
		Sj(N, LE53)	9 MN/m	
		Sj(N, LE54)	10 MN/m	
		Sj(N, LE55)	11 MN/m	
		Sj(N, LE56)	11 MN/m	
		Sj(N, LE57)	12 MN/m	
		Sj(N, LE58)	12 MN/m	
		Sj(N, LE59)	12 MN/m	
		Sj(N, LE60)	12 MN/m	
		Sj(N, LE61)	154 MN/m	
		Sj(N, LE62)	157 MN/m	
		Sj(N, LE63)	174 MN/m	
		Sj(N, LE64)	201 MN/m	
		Sj(N, LE65)	283 MN/m	
		Sj(N, LE66)	38 MN/m	
		Sj(N, LE67)	39 MN/m	
		Sj(N, LE68)	179 MN/m	
		Sj(N, LE69)	27 MN/m	
		Sj(N, LE70)	8 MN/m	
		Sj(N, LE71)	6 MN/m	
		Sj(N, LE72)	11 MN/m	
		Sj(N, LE73)	15 MN/m	
		Sj(N, LE74)	18 MN/m	
		Sj(N, LE75)	20 MN/m	
		Sj(N, LE76)	21 MN/m	
		Sj(N, LE77)	22 MN/m	

Name	Parameters	Results		
		Sj(N, LE78)	22 MN/m	
		Sj(N, LE79)	22 MN/m	
		Sj(N, LE80)	21 MN/m	
		Sj(N, LE81)	2236 MN/m	
		Sj(N, LE82)	503 MN/m	
		Sj(N, LE83)	251 MN/m	
		Sj(N, LE84)	154 MN/m	
		Sj(N, LE85)	102 MN/m	
		Sj(N, LE86)	69 MN/m	
		Sj(N, LE87)	46 MN/m	
		Sj(N, LE88)	30 MN/m	
		Sj(N, LE89)	17 MN/m	
		Sj(N, LE90)	8 MN/m	
		Sj(N, LE91)	818 MN/m	
		Sj(N, LE92)	1565 MN/m	
		Sj(N, LE93)	16390 MN/m	
		Sj(N, LE94)	120 MN/m	
		Sj(N, LE95)	113 MN/m	
		Sj(N, LE96)	106 MN/m	
		Sj(N, LE97)	100 MN/m	
		Sj(N, LE98)	94 MN/m	
		Sj(N, LE99)	89 MN/m	
		Sj(N, LE100)	85 MN/m	
		Sj(N, LE101)	253 MN/m	
		Sj(N, LE102)	163 MN/m	
		Sj(N, LE103)	118 MN/m	
		Sj(N, LE104)	92 MN/m	
		Sj(N, LE105)	75 MN/m	
		Sj(N, LE106)	64 MN/m	
		Sj(N, LE107)	55 MN/m	
		Sj(N, LE108)	50 MN/m	
		Sj(N, LE109)	44 MN/m	
		Sj(N, LE110)	41 MN/m	
		Sj(N, LE111)	37 MN/m	
		Sj(N, LE112)	33 MN/m	
		Sj(N, LE113)	31 MN/m	
		Sj(N, LE114)	10897 MN/m	
		Sj(N, LE115)	455 MN/m	
		Sj(N, LE116)	219 MN/m	
		Sj(N, LE117)	143 MN/m	
		Sj(N, LE118)	106 MN/m	
		Sj(N, LE119)	85 MN/m	
		Sj(N, LE120)	70 MN/m	
		Sj(N, LE121)	62 MN/m	
		Sj(N, LE122)	53 MN/m	
		Sj(N, LE123)	48 MN/m	
		Sj(N, LE124)	43 MN/m	
		Sj(N, LE125)	38 MN/m	
		Sj(N, LE126)	35 MN/m	

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_{C}	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5

H.3 Report 4 beam node

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	24-4-2024
Code	EN

Material

Steel	S 355, S 235

Results

Summary

Name	Parameters	Results		
4 beam node stiffness final	CHS60.3/3.2 S 235 M16 8.8 Stiffness	Analysed member	D1	
		Sj,ini(My, LE1)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE2)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE3)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE4)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE5)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE6)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE7)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE8)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE9)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE10)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE11)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE12)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE13)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE14)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE15)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE16)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE17)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE18)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE19)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE20)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE21)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE22)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE23)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE24)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE25)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE26)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE27)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE28)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE29)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE30)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE31)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE32)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE33)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE34)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE35)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE36)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE37)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE38)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE39)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE40)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE41)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE42)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE43)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE44)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE45)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE46)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE47)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE48)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE49)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE50)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE51)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE52)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE53)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE54)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE55)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE56)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE57)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE58)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE59)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE60)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE61)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE62)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE63)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE64)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE65)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE66)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE67)	0.1 MNm/rad	Semi-rigid

Name	Parameters	Results		
		Sj,ini(My, LE68)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE69)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE70)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE71)	0.2 MNm/rad	Semi-rigid
		Sj,ini(My, LE72)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE73)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE74)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE75)	0.1 MNm/rad	Semi-rigid
		Sj(N, LE1)	86 MN/m	
		Sj(N, LE2)	95 MN/m	
		Sj(N, LE3)	103 MN/m	
		Sj(N, LE4)	115 MN/m	
		Sj(N, LE5)	129 MN/m	
		Sj(N, LE6)	147 MN/m	
		Sj(N, LE7)	170 MN/m	
		Sj(N, LE8)	202 MN/m	
		Sj(N, LE9)	250 MN/m	
		Sj(N, LE10)	328 MN/m	
		Sj(N, LE11)	456 MN/m	
		Sj(N, LE12)	809 MN/m	
		Sj(N, LE13)	3593 MN/m	
		Sj(N, LE14)	1546 MN/m	
		Sj(N, LE15)	638 MN/m	
		Sj(N, LE16)	394 MN/m	
		Sj(N, LE17)	288 MN/m	
		Sj(N, LE18)	225 MN/m	
		Sj(N, LE19)	190 MN/m	
		Sj(N, LE20)	158 MN/m	
		Sj(N, LE21)	166 MN/m	
		Sj(N, LE22)	176 MN/m	
		Sj(N, LE23)	192 MN/m	
		Sj(N, LE24)	219 MN/m	
		Sj(N, LE25)	271 MN/m	
		Sj(N, LE26)	420 MN/m	
		Sj(N, LE27)	5518 MN/m	
		Sj(N, LE28)	234 MN/m	
		Sj(N, LE29)	57 MN/m	
		Sj(N, LE30)	29 MN/m	
		Sj(N, LE31)	46 MN/m	
		Sj(N, LE32)	57 MN/m	
		Sj(N, LE33)	65 MN/m	
		Sj(N, LE34)	71 MN/m	
		Sj(N, LE35)	75 MN/m	
		Sj(N, LE36)	79 MN/m	
		Sj(N, LE37)	82 MN/m	
		Sj(N, LE38)	84 MN/m	
		Sj(N, LE39)	202 MN/m	
		Sj(N, LE40)	222 MN/m	
		Sj(N, LE41)	241 MN/m	
		Sj(N, LE42)	268 MN/m	
		Sj(N, LE43)	301 MN/m	
		Sj(N, LE44)	344 MN/m	
		Sj(N, LE45)	398 MN/m	
		Sj(N, LE46)	472 MN/m	
		Sj(N, LE47)	584 MN/m	
		Sj(N, LE48)	765 MN/m	
		Sj(N, LE49)	1065 MN/m	
		Sj(N, LE50)	1888 MN/m	
		Sj(N, LE51)	8384 MN/m	
		Sj(N, LE52)	3614 MN/m	
		Sj(N, LE53)	1489 MN/m	
		Sj(N, LE54)	921 MN/m	
		Sj(N, LE55)	670 MN/m	
		Sj(N, LE56)	525 MN/m	
		Sj(N, LE57)	443 MN/m	
		Sj(N, LE58)	369 MN/m	
		Sj(N, LE59)	177 MN/m	
		Sj(N, LE60)	184 MN/m	

Name	Parameters	Results		
		Sj(N, LE61)	191 MN/m	
		Sj(N, LE62)	199 MN/m	
		Sj(N, LE63)	208 MN/m	
		Sj(N, LE64)	217 MN/m	
		Sj(N, LE65)	227 MN/m	
		Sj(N, LE66)	239 MN/m	
		Sj(N, LE67)	251 MN/m	
		Sj(N, LE68)	284 MN/m	
		Sj(N, LE69)	304 MN/m	
		Sj(N, LE70)	325 MN/m	
		Sj(N, LE71)	348 MN/m	
		Sj(N, LE72)	376 MN/m	
		Sj(N, LE73)	408 MN/m	
		Sj(N, LE74)	445 MN/m	
		Sj(N, LE75)	489 MN/m	

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_{C}	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5

H.4 Report 8 beam node

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	24-4-2024
Code	EN

Material

Steel	S 355, S 235

Results

Summary

Name	Parameters	Results		
		Analysed member	D7	
		Sj,ini(My, LE1)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE2)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE3)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE4)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE5)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE6)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE8)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE9)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE10)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE11)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE12)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE13)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE14)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE15)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE16)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE17)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE18)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE19)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE20)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE21)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE22)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE23)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE24)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE25)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE26)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE27)	0.1 MNm/rad	Semi-rigid
		Sj,ini(My, LE28)	0.1 MNm/rad	Semi-rigid
8 beam node	CHS60.3/3.2 S 235 M16 8.8 Stiffness	Sj,ini(My, LE29)	0.1 MNm/rad	Semi-rigid
		Sj(N, LE1)	834 MN/m	
		Sj(N, LE2)	934 MN/m	
		Sj(N, LE3)	1060 MN/m	
		Sj(N, LE4)	1253 MN/m	
		Sj(N, LE5)	1280 MN/m	
		Sj(N, LE6)	1280 MN/m	
		Sj(N, LE8)	1231 MN/m	
		Sj(N, LE9)	1231 MN/m	
		Sj(N, LE10)	1231 MN/m	
		Sj(N, LE11)	1202 MN/m	
		Sj(N, LE12)	1021 MN/m	
		Sj(N, LE13)	901 MN/m	
		Sj(N, LE14)	809 MN/m	
		Sj(N, LE15)	11 MN/m	
		Sj(N, LE16)	12 MN/m	
		Sj(N, LE17)	14 MN/m	
		Sj(N, LE18)	16 MN/m	
		Sj(N, LE19)	19 MN/m	
		Sj(N, LE20)	23 MN/m	
		Sj(N, LE21)	30 MN/m	
		Sj(N, LE22)	40 MN/m	
		Sj(N, LE23)	29 MN/m	
		Sj(N, LE24)	22 MN/m	
		Sj(N, LE25)	18 MN/m	
		Sj(N, LE26)	16 MN/m	
		Sj(N, LE27)	14 MN/m	
		Sj(N, LE28)	12 MN/m	
		Sj(N, LE29)	11 MN/m	

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_{C}	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5

H.5 Report corner joint SHS

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	24-4-2024
Code	EN

Material

Steel	S 355, S 235

Results

Summary

Project:
Project no:
Author:

Name	Parameters	Results		
Beam to corner point 2	SHS90/90/6.3, CHS60.3/3.2 S 235 M18 8.8 Stiffness	Analysed member	B2	
		Sj,ini(My, LE1)	∞ MNm/rad	Rigid
		Sj,ini(My, LE2)	∞ MNm/rad	Rigid
		Sj,ini(My, LE3)	∞ MNm/rad	Rigid
		Sj,ini(My, LE4)	∞ MNm/rad	Rigid
		Sj,ini(My, LE5)	∞ MNm/rad	Rigid
		Sj,ini(My, LE6)	∞ MNm/rad	Rigid
		Sj,ini(My, LE7)	∞ MNm/rad	Rigid
		Sj,ini(My, LE8)	∞ MNm/rad	Rigid
		Sj,ini(My, LE9)	∞ MNm/rad	Rigid
		Sj,ini(My, LE10)	∞ MNm/rad	Rigid
		Sj,ini(My, LE11)	∞ MNm/rad	Rigid
		Sj,ini(My, LE12)	∞ MNm/rad	Rigid
		Sj,ini(My, LE13)	∞ MNm/rad	Rigid
		Sj,ini(My, LE14)	∞ MNm/rad	Rigid
		Sj,ini(My, LE15)	∞ MNm/rad	Rigid
		Sj,ini(My, LE16)	∞ MNm/rad	Rigid
		Sj,ini(My, LE17)	∞ MNm/rad	Rigid
		Sj,ini(My, LE18)	∞ MNm/rad	Rigid
		Sj,ini(My, LE19)	∞ MNm/rad	Rigid
		Sj,ini(My, LE20)	∞ MNm/rad	Rigid
		Sj,ini(My, LE21)	∞ MNm/rad	Rigid
		Sj,ini(My, LE22)	∞ MNm/rad	Rigid
		Sj,ini(My, LE23)	∞ MNm/rad	Rigid
		Sj,ini(My, LE24)	∞ MNm/rad	Rigid
		Sj,ini(My, LE25)	∞ MNm/rad	Rigid
		Sj,ini(My, LE26)	∞ MNm/rad	Rigid
		Sj,ini(My, LE27)	∞ MNm/rad	Rigid
		Sj,ini(My, LE28)	∞ MNm/rad	Rigid
		Sj,ini(My, LE29)	∞ MNm/rad	Rigid
		Sj,ini(My, LE30)	∞ MNm/rad	Rigid
		Sj,ini(My, LE31)	∞ MNm/rad	Rigid
		Sj,ini(My, LE32)	∞ MNm/rad	Rigid
		Sj,ini(My, LE33)	∞ MNm/rad	Rigid
		Sj,ini(My, LE34)	∞ MNm/rad	Rigid
		Sj,ini(My, LE35)	∞ MNm/rad	Rigid
		Sj,ini(My, LE36)	∞ MNm/rad	Rigid
		Sj,ini(My, LE37)	∞ MNm/rad	Rigid
		Sj,ini(My, LE38)	∞ MNm/rad	Rigid
		Sj,ini(My, LE39)	∞ MNm/rad	Rigid
		Sj,ini(My, LE40)	∞ MNm/rad	Rigid
		Sj,ini(My, LE41)	∞ MNm/rad	Rigid
		Sj,ini(My, LE42)	∞ MNm/rad	Rigid
		Sj,ini(My, LE43)	∞ MNm/rad	Rigid
		Sj,ini(My, LE44)	∞ MNm/rad	Rigid
		Sj,ini(My, LE45)	∞ MNm/rad	Rigid
		Sj,ini(My, LE46)	∞ MNm/rad	Rigid
		Sj,ini(My, LE47)	∞ MNm/rad	Rigid
		Sj,ini(My, LE48)	∞ MNm/rad	Rigid
		Sj,ini(My, LE49)	∞ MNm/rad	Rigid
		Sj,ini(My, LE50)	∞ MNm/rad	Rigid
		Sj,ini(My, LE51)	∞ MNm/rad	Rigid
		Sj,ini(My, LE52)	∞ MNm/rad	Rigid
		Sj,ini(My, LE53)	∞ MNm/rad	Rigid
		Sj,ini(My, LE54)	∞ MNm/rad	Rigid
		Sj,ini(My, LE55)	∞ MNm/rad	Rigid
		Sj,ini(My, LE56)	∞ MNm/rad	Rigid
		Sj,ini(My, LE57)	∞ MNm/rad	Rigid
		Sj,ini(My, LE58)	∞ MNm/rad	Rigid
		Sj,ini(My, LE59)	∞ MNm/rad	Rigid
		Sj,ini(My, LE60)	∞ MNm/rad	Rigid
		Sj,ini(My, LE61)	∞ MNm/rad	Rigid
		Sj,ini(My, LE62)	∞ MNm/rad	Rigid
		Sj,ini(My, LE63)	∞ MNm/rad	Rigid
		Sj,ini(My, LE64)	∞ MNm/rad	Rigid
		Sj,ini(My, LE65)	∞ MNm/rad	Rigid
		Sj,ini(My, LE66)	∞ MNm/rad	Rigid
		Sj,ini(My, LE67)	∞ MNm/rad	Rigid

Name	Parameters	Results		
		Sj,ini(My, LE68)	∞ MNm/rad	Rigid
		Sj,ini(My, LE69)	∞ MNm/rad	Rigid
		Sj,ini(My, LE70)	∞ MNm/rad	Rigid
		Sj,ini(My, LE71)	∞ MNm/rad	Rigid
		Sj,ini(My, LE72)	∞ MNm/rad	Rigid
		Sj,ini(My, LE73)	7.6 MNm/rad	Rigid
		Sj,ini(My, LE74)	∞ MNm/rad	Rigid
		Sj,ini(My, LE75)	∞ MNm/rad	Rigid
		Sj,ini(My, LE76)	∞ MNm/rad	Rigid
		Sj,ini(My, LE77)	∞ MNm/rad	Rigid
		Sj,ini(My, LE78)	∞ MNm/rad	Rigid
		Sj,ini(My, LE79)	∞ MNm/rad	Rigid
		Sj,ini(My, LE80)	∞ MNm/rad	Rigid
		Sj,ini(My, LE81)	∞ MNm/rad	Rigid
		Sj,ini(My, LE82)	∞ MNm/rad	Rigid
		Sj,ini(My, LE83)	∞ MNm/rad	Rigid
		Sj,ini(My, LE84)	∞ MNm/rad	Rigid
		Sj,ini(My, LE85)	∞ MNm/rad	Rigid
		Sj,ini(My, LE86)	∞ MNm/rad	Rigid
		Sj,ini(My, LE87)	∞ MNm/rad	Rigid
		Sj(N, LE1)	10 MN/m	
		Sj(N, LE2)	11 MN/m	
		Sj(N, LE3)	12 MN/m	
		Sj(N, LE4)	13 MN/m	
		Sj(N, LE5)	15 MN/m	
		Sj(N, LE6)	18 MN/m	
		Sj(N, LE7)	27 MN/m	
		Sj(N, LE8)	37 MN/m	
		Sj(N, LE9)	57 MN/m	
		Sj(N, LE10)	15000000 MN/m	
		Sj(N, LE11)	88 MN/m	
		Sj(N, LE12)	48 MN/m	
		Sj(N, LE13)	33 MN/m	
		Sj(N, LE14)	25 MN/m	
		Sj(N, LE15)	20 MN/m	
		Sj(N, LE16)	17 MN/m	
		Sj(N, LE17)	15 MN/m	
		Sj(N, LE18)	13 MN/m	
		Sj(N, LE19)	11 MN/m	
		Sj(N, LE20)	16 MN/m	
		Sj(N, LE21)	18 MN/m	
		Sj(N, LE22)	20 MN/m	
		Sj(N, LE23)	22 MN/m	
		Sj(N, LE24)	25 MN/m	
		Sj(N, LE25)	25 MN/m	
		Sj(N, LE26)	30 MN/m	
		Sj(N, LE27)	45 MN/m	
		Sj(N, LE28)	61 MN/m	
		Sj(N, LE29)	95 MN/m	
		Sj(N, LE30)	25000001 MN/m	
		Sj(N, LE31)	146 MN/m	
		Sj(N, LE32)	80 MN/m	
		Sj(N, LE33)	55 MN/m	
		Sj(N, LE34)	42 MN/m	
		Sj(N, LE35)	34 MN/m	
		Sj(N, LE36)	28 MN/m	
		Sj(N, LE37)	24 MN/m	
		Sj(N, LE38)	21 MN/m	
		Sj(N, LE39)	19 MN/m	
		Sj(N, LE40)	48 MN/m	
		Sj(N, LE41)	53 MN/m	
		Sj(N, LE42)	59 MN/m	
		Sj(N, LE43)	66 MN/m	
		Sj(N, LE44)	76 MN/m	
		Sj(N, LE45)	89 MN/m	
		Sj(N, LE46)	136 MN/m	
		Sj(N, LE47)	184 MN/m	
		Sj(N, LE48)	284 MN/m	

Name	Parameters	Results		
		Sj(N, LE49)	75000002 MN/m	
		Sj(N, LE50)	439 MN/m	
		Sj(N, LE51)	239 MN/m	
		Sj(N, LE52)	164 MN/m	
		Sj(N, LE53)	125 MN/m	
		Sj(N, LE54)	101 MN/m	
		Sj(N, LE55)	84 MN/m	
		Sj(N, LE56)	73 MN/m	
		Sj(N, LE57)	64 MN/m	
		Sj(N, LE58)	57 MN/m	
		Sj(N, LE59)	32 MN/m	
		Sj(N, LE60)	34 MN/m	
		Sj(N, LE61)	37 MN/m	
		Sj(N, LE62)	40 MN/m	
		Sj(N, LE63)	43 MN/m	
		Sj(N, LE64)	47 MN/m	
		Sj(N, LE65)	52 MN/m	
		Sj(N, LE66)	58 MN/m	
		Sj(N, LE67)	66 MN/m	
		Sj(N, LE68)	75 MN/m	
		Sj(N, LE69)	88 MN/m	
		Sj(N, LE70)	107 MN/m	
		Sj(N, LE71)	136 MN/m	
		Sj(N, LE72)	186 MN/m	
		Sj(N, LE73)	87 MN/m	
		Sj(N, LE74)	48300001 MN/m	
		Sj(N, LE75)	381 MN/m	
		Sj(N, LE76)	217 MN/m	
		Sj(N, LE77)	152 MN/m	
		Sj(N, LE78)	117 MN/m	
		Sj(N, LE79)	95 MN/m	
		Sj(N, LE80)	80 MN/m	
		Sj(N, LE81)	69 MN/m	
		Sj(N, LE82)	60 MN/m	
		Sj(N, LE83)	54 MN/m	
		Sj(N, LE84)	49 MN/m	
		Sj(N, LE85)	45 MN/m	
		Sj(N, LE86)	41 MN/m	
		Sj(N, LE87)	38 MN/m	

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_{C}	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5

H.6 Report corner joint CHS

Project:
Project no:
Author:

Project data

Project name	
Project number	
Author	
Description	
Date	24-4-2024
Code	EN

Material

Steel	S 355, S 235

Results

Summary

Name	Parameters	Results		
Diagonal to corner point 1	SHS90/90/6.3, CHS60.3/3.2 S 235 M18 8.8 Stiffness	Analysed member	D4	
		Sj,ini(My, LE49)	∞ MNm/rad	Rigid
		Sj,ini(My, LE50)	∞ MNm/rad	Rigid
		Sj,ini(My, LE51)	∞ MNm/rad	Rigid
		Sj,ini(My, LE52)	∞ MNm/rad	Rigid
		Sj,ini(My, LE53)	∞ MNm/rad	Rigid
		Sj,ini(My, LE54)	∞ MNm/rad	Rigid
		Sj,ini(My, LE55)	∞ MNm/rad	Rigid
		Sj,ini(My, LE56)	∞ MNm/rad	Rigid
		Sj,ini(My, LE57)	∞ MNm/rad	Rigid
		Sj,ini(My, LE58)	∞ MNm/rad	Rigid
		Sj,ini(My, LE59)	∞ MNm/rad	Rigid
		Sj,ini(My, LE60)	∞ MNm/rad	Rigid
		Sj,ini(My, LE61)	∞ MNm/rad	Rigid
		Sj,ini(My, LE62)	∞ MNm/rad	Rigid
		Sj,ini(My, LE63)	∞ MNm/rad	Rigid
		Sj,ini(My, LE64)	∞ MNm/rad	Rigid
		Sj,ini(My, LE65)	∞ MNm/rad	Rigid
		Sj,ini(My, LE66)	∞ MNm/rad	Rigid
		Sj,ini(My, LE67)	∞ MNm/rad	Rigid
		Sj,ini(My, LE68)	∞ MNm/rad	Rigid
		Sj,ini(My, LE69)	∞ MNm/rad	Rigid
		Sj,ini(My, LE70)	∞ MNm/rad	Rigid
		Sj,ini(My, LE71)	∞ MNm/rad	Rigid
		Sj,ini(My, LE72)	∞ MNm/rad	Rigid
		Sj,ini(My, LE73)	∞ MNm/rad	Rigid
		Sj,ini(My, LE74)	∞ MNm/rad	Rigid
		Sj,ini(My, LE75)	∞ MNm/rad	Rigid
		Sj,ini(My, LE76)	∞ MNm/rad	Rigid
		Sj(N, LE49)	16 MN/m	
		Sj(N, LE50)	31 MN/m	
		Sj(N, LE51)	62 MN/m	
		Sj(N, LE52)	77 MN/m	
		Sj(N, LE53)	92 MN/m	
		Sj(N, LE54)	107 MN/m	
		Sj(N, LE55)	122 MN/m	
		Sj(N, LE56)	137 MN/m	
		Sj(N, LE57)	151 MN/m	
		Sj(N, LE58)	166 MN/m	
		Sj(N, LE59)	180 MN/m	
		Sj(N, LE60)	194 MN/m	
		Sj(N, LE61)	208 MN/m	
		Sj(N, LE62)	16 MN/m	
		Sj(N, LE63)	32 MN/m	
		Sj(N, LE64)	48 MN/m	
		Sj(N, LE65)	64 MN/m	
		Sj(N, LE66)	81 MN/m	
		Sj(N, LE67)	97 MN/m	
		Sj(N, LE68)	114 MN/m	
		Sj(N, LE69)	131 MN/m	
		Sj(N, LE70)	148 MN/m	
		Sj(N, LE71)	165 MN/m	
		Sj(N, LE72)	182 MN/m	
		Sj(N, LE73)	200 MN/m	
		Sj(N, LE74)	217 MN/m	
		Sj(N, LE75)	235 MN/m	
		Sj(N, LE76)	16 MN/m	

Project:
Project no:
Author:

Code settings

Item	Value	Unit	Reference
Safety factor γ_{M0}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M1}	1.00	-	EN 1993-1-1: 6.1
Safety factor γ_{M2}	1.25	-	EN 1993-1-1: 6.1
Safety factor γ_{M3}	1.25	-	EN 1993-1-8: 2.2
Safety factor γ_C	1.50	-	EN 1992-1-1: 2.4.2.4
Safety factor γ_{Inst}	1.20	-	EN 1992-4: Table 4.1
Joint coefficient βj	0.67	-	EN 1993-1-8: 6.2.5
Effective area - influence of mesh size	0.10	-	
Friction coefficient - concrete	0.25	-	EN 1993-1-8
Friction coefficient in slip-resistance	0.30	-	EN 1993-1-8 tab 3.7
Limit plastic strain	0.05	-	EN 1993-1-5
Detailing	Yes		
Distance between bolts [d]	2.20	-	EN 1993-1-8: tab 3.3
Distance between bolts and edge [d]	1.20	-	EN 1993-1-8: tab 3.3
Concrete breakout resistance check	Both		EN 1992-4: 7.2.1.4 and 7.2.2.5
Use calculated αb in bearing check.	Yes		EN 1993-1-8: tab 3.4
Cracked concrete	Yes		EN 1992-4
Local deformation check	Yes		CIDECT DG 1, 3 - 1.1
Local deformation limit	0.03	-	CIDECT DG 1, 3 - 1.1
Geometrical nonlinearity (GMNA)	Yes		Analysis with large deformations for hollow section joints
Braced system	Yes		EN 1993-1-8: 5.2.2.5