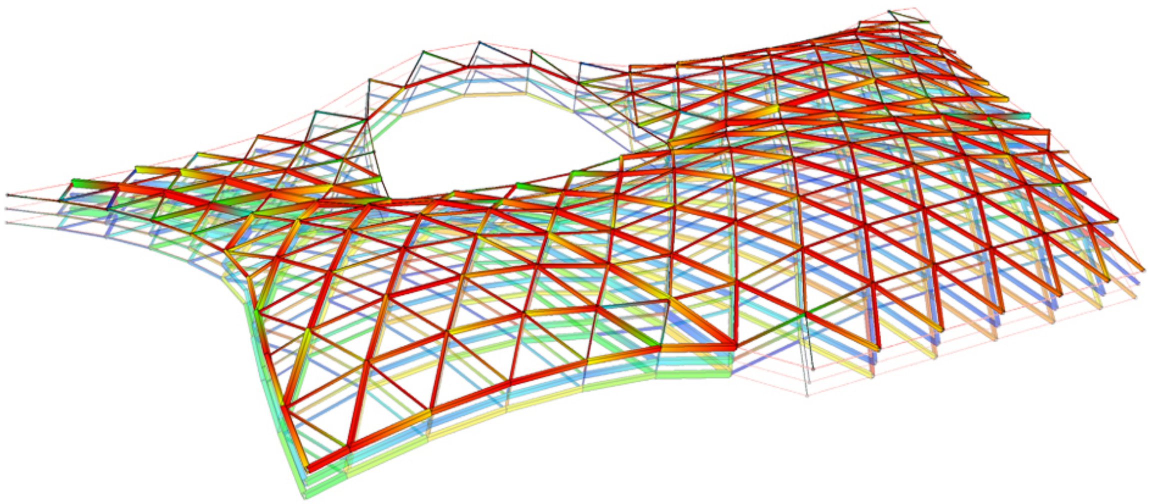
CHALMERS



Form finding and size optimization

Implementation of beam elements and size optimization in real time form finding using dynamic relaxation

Master's thesis in Architectural Engineering

JENS OLSSON

Department of Applied Mechanics
Division of Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2012
Master's thesis 2012:46

MASTER'S THESIS IN ARCHITECTURAL ENGINEERING

Form finding and size optimization

Implementation of beam elements and size optimization in real time form finding using dynamic relaxation

JENS OLSSON

Department of Applied Mechanics
Division of Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2012

Form finding and size optimization
Implementation of beam elements and size optimization in real time form finding using dynamic relaxation
JENS OLSSON

Master's thesis 2012:46
ISSN 1652-8557
Department of Applied Mechanics
Division of Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
Shows the results from form-finding and sizing of the roof at Exeter University with SMART Form. The three overlaid images show the total, bending and axial utilization where red is 70 %.

Chalmers Reproservice
Gothenburg, Sweden 2012

Form finding and size optimization
Implementation of beam elements and size optimization in real time form finding using dynamic relaxation
Master's thesis in Architectural Engineering
JENS OLSSON
Department of Applied Mechanics
Division of Dynamics
Chalmers University of Technology

ABSTRACT

In the pursuit of performance-based architecture there is an increased demand for integration between engineers and architects, between analysis and design. The key to make wise choices early in a design process, has previously been based on experience from prior work and experiments with physical models. With modern computational power those choices can be supplemented with real-time feedback from analysis tools. But few programs are designed for the type of light weight, real-time analysis that can be useful in the exploration of an architectural concept. This has been the target for a research and development team, SMART Solutions, at Buro Happold.

SMART Form is a tool for performance-based architecture as a plug-in for the 3D modeling software Rhinoceros. It combines analysis and rationalization of complex geometries with form finding and sculpting of structures, driven by a dynamic relaxation solver. The plugin transforms the 3D modeling software to a virtual physical environment, influenced by various forces and a gravitational field. It enables sketching and sculpting of efficient compression (shells and gridshells) and tension structures (cable nets and membranes).

The purpose for the thesis has been to advance SMART Form by implementing beam elements and explore how bending action can be used in the form finding process, allowing for a compromise between initial geometry and fully form found geometry. Automatic sizing of the beam elements is also implemented as a method to evaluate form. The thesis leads to an investigation of how to conduct conceptual structural design, and how this process can potentially be approached differently with the means of real-time feedback from analysis, form finding and sizing. In order to allow for real-time analysis an iterative technique called dynamic relaxation is used as the computational solver.

As a result of the exploration of form finding with bending action a method called *Limit control form finding* was developed. This method enables form optimization of the most critical parts of a structure by specifying a limit for the bending capability of the elements. It was found to work in a good way for 2D structures but sometimes resulting in a less appealing, wrinkled looking grids for 3D structures.

Dynamic relaxation is found to be a suitable method for real time analysis, although with some important issues regarding speed. The number of iterations needed to reach convergence is concluded to depend on the relation between local and global stiffness for each node in the structure. Hence the method works efficiently for some types of structural configuration (for example gridshells with a locally weak direction), while a conventional matrix solver would perhaps be more suitable for other cases (for example space truss structures).

The implemented functionality, such as form finding using cables, limit control and automatic sizing is demonstrated with a couple of test cases. One of which is the Exeter gridshell roof where engineers at Buro Happold have been involved from the concept design to the realized structure. The real design process for the project is shown and new approach based on real time feedback is demonstrated. With the new functionality implemented during the thesis, SMART Form has become a much more sophisticated tool, and a great improvement for the design process of the Exeter roof could be made. Due to the possibilities of a quick and efficient collaborative design process, that comes with the new tool, it seems to offer great potential in bridging between engineers and architects in structural concept design.

Keywords: Form finding, Dynamic relaxation, Size optimization, Real-time analysis

Preface

This Master's Thesis comprises 30 credits and has been carried out during the summer and autumn of 2012. The work has taken place at the office for Buro Happold in Bath, England in close collaboration with the research and development team SMART Solutions.

Main supervisor has been Dr. Al Fisher, analyst and head of research and development in the SMART Solutions team. From Chalmers University of Technology lecturer Håkan Johansson has assisted as the examinator for the thesis.

Acknowledgements

First and foremost I want to thank Dr. Al Fisher for his encouragement and guidance throughout his supervision of this thesis. I would also like to thank Shrikant Sharma, head of SMART Solutions, for giving me the opportunity to collaborate with Buro Happold for the thesis.

I would also like to thank Dr. Håkan Johansson who has been the examiner from my university for his support and much appreciated help with the report writing.

Thanks are also due to Prof. Chris Williams, Senior Engineers Mark Cunliff and Chris Shrubshall, whom have all contributed with important feedback to support the software development in the thesis.

At last I want to thank Erica Henrysson and Marks Stark for the many rewarding discussions we have had regarding of form finding and coding in SMART Form.

Nomenclature

General note

In general, scalars are written using normal roman letters, e.g. scalar a or scalar A, whilst vectors and matrices are written using bold roman letters and distinguished between using upper case letters, respectively, e.g. the vector $\boldsymbol{a}$ and the matrix $\boldsymbol{A}$. The indices m and i most often refers to element index and a node index respectively.

Roman upper case letters

A	Cross section area
A_m	Cross section area for element m
C	Viscous damping factor
D_{EA}	Axial stiffness
D_{EI}	Rotational stiffness
E	Young's modulus
E_m	Young's modulus for element m
F_{1i}	Global force applied to the node at end 1 (i takes the form x,y,z)
F_{2i}	Global force applied to the node at end 2 (i takes the form x,y,z)
G	Shear modulus
H_{iy}	Out-of-balance torque for node i around the y-axis
I_x	Second moment of area around the local x-axis
I_y	Second moment of area around the local y-axis
J_{iy}	Moment of inertia for node i around the local y-axis
K_{ix}	Axial stiffness for node i in the x-direction
K_v	Torsional constant
L	Element length
L_0	Element initial length
L_t	Element length at time t
L_{mx}	Length of element m in the x-direction
M_{x1}	Moment around local x-axis for end 1
M_{x2}	Moment around local x-axis for end 2
M_{y1}	Moment around local y-axis for end 1
M_{y2}	Moment around local y-axis for end 2
M_{ϕ}	Twisting moment around local z-axis
M_{1i}	Global moment applied to the node at end 1 (i takes the form x,y,z)
M_{2i}	Global moment applied to the node at end 2 (i takes the form x,y,z)
M_{ix}	Mass for node i in the x-direction
M_{ix}^t	Mass for node i in the x-direction at time t
$M_{x,Ed}$	Design values for moment around local x-axis
$M_{y,Ed}$	Design values for moment around local y-axis

N	Axial force
N_m	Axial force for element m
N_{Ed}	Design values for the compression force
P_{ix}	External force applied to node i in the x-direction
P_c	Buckling force for Euer buckling
R_{ix}	Out-of-balance force for node i in the x-direction
S	Rotational stiffness
S_{iy}	Rotational stiffness for node i in the y-direction
T_{iy}	External torque applied to node i around the y-axis
U	Strain energy

Roman lower case letters

a	Acceleration
a_{ix}	Acceleration for node i in the x-direction
a_{ix}^t	Acceleration for node i in the x-direction at time t
c	Cross section scale factor
e_a	Axial deformation due to nodal translations
e_b	Axial deformation due to nodal rotations
k	Effective length scale factor
k_{xx}	Interaction factor for axial force and moment
k_{xy}	Interaction factor for axial force and moment
k_{yx}	Interaction factor for axial force and moment
k_{yy}	Interaction factor for axial force and moment
u	Parameter for the shape function
u_{currnet}	Current utilization
u_{sought}	Sought level of utilization
q_y	Distributed load applied in the y-direction
v	Velocity
v_{ix}	Velocity for node i in the x-direction
v_{ix}^t	Velocity for node i in the x-direction at time t
w_p	Particular solution for the deflection
x	Positions along the x-axis in the the global coordinate system
x^t	Positions along the x-axis in the the global coordinate system at time t

Roman bold upper case letters

$\boldsymbol{A}$	Transformation matrix for rotations
$\boldsymbol{C}$	Coordinate system written on matrix form
$\boldsymbol{J}_i$	Moment of inertia matrix for node i
$\boldsymbol{K}$	Global stiffness matrix
$\boldsymbol{K}^e$	Element stiffness matrix

$\boldsymbol{M}_i$	Mass matrix for node i

Roman bold lower case letters

$\boldsymbol{a}^e$	Element displacement vector
$\boldsymbol{f}^e$	Element force vector
$\boldsymbol{f}^l$	Load vector
$\boldsymbol{r}$	Vector from origin to any point on an element
$\boldsymbol{r}_1$	Vector from origin to the start of an element
$\boldsymbol{r}_2$	Vector from origin to the end of an element
$\boldsymbol{p}$	Vector between the end nodes of an element
$\boldsymbol{y_1}$	Local coordinate axis defining the y-direction for end 1
$\boldsymbol{y_2}$	Local coordinate axis defining the y-direction for end 2
$\boldsymbol{x_1}$	Local coordinate axis defining the x-direction for end 1
$\boldsymbol{x_2}$	Local coordinate axis defining the x-direction for end 2
$\boldsymbol{z_1}$	Local coordinate axis defining the z-direction for end 1
$\boldsymbol{z_2}$	Local coordinate axis defining the z-direction for end 2

Greek lower case letters

α	Rotational acceleration
α_{iy}	Rotational acceleration for node i around the local y-axis
α_{iy}^t	Rotational acceleration for node i around the local y-axis at time t
δ_{ix}	Displacement of node i in the along the x-axis
ϵ_{ijk}	3D permutation tensor
θ	Rotation
θ_{iy}	Rotation of node i around the local y-axis
θ_{iy}^t	Rotation of node i around the local y-axis at time t
θ_{x1}	Rotation of end 1 around the local x-axis
θ_{x2}	Rotation of end 2 around the local x-axis
θ_{y1}	Rotation of end 1 around the local y-axis
θ_{y2}	Rotation of end 2 around the local y-axis
κ_x	Curvature around local x-axis
κ_y	Curvature around local y-axis
λ	Load scale factor
$\bar{\lambda}$	Non-dimensional slenderness
ρ	Material density
ϕ	Angle of twist
χ	Reduction factor for buckling
χ_{LT}	Reduction factor for lateral torsional buckling
ω	Rotational velocity
ω_{iy}	Rotational velocity for node i around the local y-axis

ω_{iy}^{t}	Rotational velocity for node i around the local y-axis at time t
ω_{ij}	Rotational velocity for node i in relation to adjacent node j

Greek upper case letters

Δt	Time step
$\Delta M_{x,Ed}$	Moment due to the shift of the centrodial x-axis
$\Delta M_{y,Ed}$	Moment due to the shift of the centrodial y-axis

Contents

1 Introduction

1.1 Background

In this chapter topics related to the function of SMART Form as a design tool for performance driven architecture are introduced. The aim is partly to place the tool in a historical context among computer tools in architecture and/or engineering and partly to attempt to point out some limitations with existing tools. Examples of historical and recent projects are presented and this section is summarized with an introduction to SMART Form.

1.1.1 Free form architecture

In the later half of the 20th century technology took a giant leap forward as computers entered the scene of science and engineering. Computer aided design allows for visualizing geometries and analyzing the structural behavior of the most complex systems. Modern 3D CAD software have become essential tools in any design practice. Whether conducting vehicle, product or architectural design, the aid of computers have changed not only the work flow but also the design process.

Computer based 3D modeling has historically been influenced from many different industries, although two of the main drivers have been the machine industry, developing cars and airplanes, and the film and entertainment industry pushing the development of animated illusions. The much different aims with 3D modeling from the two industries led to development of two separate 3D modeling techniques: surface modeling and polygon modeling. Surface modeling is accurate and precise whereas polygon modeling is more similar to sculpting with clay, the artistic freedom is prioritized over precision.

Sir Edmund (Ted) Happold was among the first engineers to ever use computers for modeling in structural engineering. In the late 1950's his team of engineers at Arup pioneered this new technic in building design with the Sydney Opera house. The complex design work for the shells was achieved through the use of computers in order to model the geometry of the roof and analyze its structure [2]. From the late 50's and until the 90's "free form" architecture was dominated by form-finding structures, where there is a distinct relationship between form and function. The development within this field was led by pioneers like Frei Otto, Heinz Isler, Felix Candela, Ted Happold and Horst Berger just to mention a few. These structures were probably amongst the most spectacular at the time (if not still), not only from a structural point of view, but also from a free form point of view. Some examples of these types of structures can be found in Figure 1.1.2 - Figure 1.1.5.

Computers entered the architectural scene with the PC explosion in the 1990's. To begin with, CAD was mainly a tool for 2D drawings, essentially just replacing hand drawing, but also introducing a new level of precision. It was a great step forward in productivity, but did not contribute much to the design process. Some resisted to adopt the computer modeling, arguing that it challenged the creativity, whilst other were keen to use this new technic. But it wasn't until 3D CAD was adopted that computers became a true game changer [7]. The first use of 3D modeling was mainly as a tool for renderings and presentations, however the significant change occurred when 3D CAD started to be used as a design tool. The influence of this new way of designing can be seen in projects from the last decade by architects like Zaha Hadid, Frank Gehry, Studio Morphosis, Asymptote and others [7]. In comparison to the form finding structures, this computer based sculptural architecture has seemingly lost the link between form and structural function. With software developed for the film industry, architectural forms can be sculpted in the most unimaginable ways, tending to not leave much room for the structural parameters. Two examples of which can be seen in Figure 1.1.6 and Figure 1.1.7.

Even though computers and 3D CAD have enabled a new type of free form architecture, some are skeptical to the way these tools are used. In the paper [5] professor Horst Berger raises the question of what we define as good architecture. Not only the way we conduct design has changed with the aid of computers, but also what is possible to fabricate and construct, anything seems to be possible. The almost unlimited availability of tools and technology makes the decision of what should and should not be designed and built difficult [5]. Berger points out that sensationalism in itself, often is enough to justify the excessive cost of structurally unreasonable architecture, as a result of sculptural free form design.

The adoption of CAD software in architecture practices was a slow process but today 2D and 3D CAD is widely spread in virtually all firms. The next step in the development of CAD systems is seemingly the exploration of 4th dimension; time [7]. Making time part of the design process, enables fly-through animations and real time feedback.

One can argue that despite the great availability of computer tools for structural analysis and architecture, there is still a gap to fill. Especially when it comes to combining architecture and engineering in conceptual performance based design.

1.1.2 Performance based design

Performance based is a term commonly phrased in contemporary architecture. It implies an urge to enhance some aspects of performance for a design, a building or more applicable for this thesis, a structure. The concept of driving design with some aspects of performance is not new, it has been around for a long time, it is just the label, the tools and the methods that changes.

In order to discuss performance for design it can be helpful to separate and categorize its functions and features. In the discussion of contemporary architecture the term *parameter* has become wildly used. If one thinks of a building as a set of parameters, the design process can be likened to a balance act, where parameters are weight and prioritized to fit the scheme and context for a project.

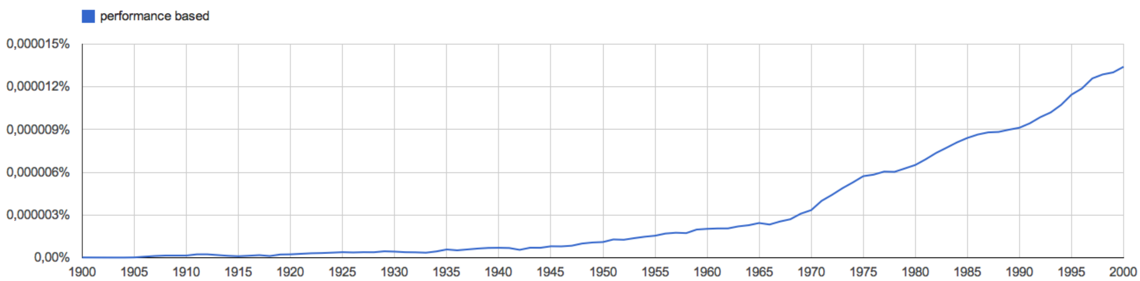


Figure 1.1.1: *A search at Google Ngram for the expression Performance based. The graph shows the frequency of use for the expression from year 1900 until 2000.*

The parameters dictating the performance of a building are many, intricately linked and impossible to completely separate. In an attempt, however, to simplify the complexity of building design, let us consider four categories of parameters: the social, cultural, environmental and economical category. Some of these can easily be measured and quantified with scientific methods, others require experience, deep knowledge and practical training to appreciate. The social and cultural aspects of architecture are categories of parameters that are difficult to measure. The incorporation into the design process has to be based on intuition, experience and knowledge from previous work. The environmental and economical category on the other hand, involves quantifiable and measurable parameters. These aspects are strongly linked to the choice and/or use of material, transportation and construction technics as well as the use of man-power and technical solutions.

In traditional architectural design the integration of measurable parameters often take place in a late phase of the design process. The concept design is rather focused on performance of less tangible nature (aesthetics, experience of spaces and plan functionality), meaning that measurable aspects of performance often get postponed [25]. Considering the great impact of choices made in conceptual design this approach is limiting the success of an integrated solution.

In the way the term *performance based* is used today, there is an almost implicit association with computer tools together with measurable parameters, and this is no coincides. A computer device can be programmed to carry out arithmetic and logical operations with a capability of performing billions of instructions per second. This vast computational power makes them ideal for repetitive calculations, for instance useful in optimization algorithms. But the target parameters for the optimization have to be simple enough to be formulated in an computer algorithm. This is typically the case with measurable and quantifiable parameters. In the application to structural engineering these could be weight, force, moment, cost or deflection. The parameter in focus for this thesis is the structural performance, dictated by the use of material for lightweight compression structures.

Historic examples of form finding based architecture

Frei Otto was one of the most influential and recognized architects in the field of lightweight structures [10]. After he finished his doctoral dissertation in suspended roofs he designed the Olympic Stadium in Munich in close collaboration with engineers Behnisch and Schlaich. He founded the Institute of Light Weight Structures in 1964 at the University of Stuttgart and taught the subject until 1990.

Figure 1.1.2: *The Olympic stadium in Munich from 1972. [15]*

Felix Candela was trained as an architect at Universidad Politecnica de Madrid, Spain. He got interested in how calculations could predict strength and collapse of structures. Candela pioneered innovations in double curved hyperbolic shell surfaces structures, and rather than shaping the structures arbitrarily, he insisted to design structures where the form corresponds to the flow of forces [10].

Figure 1.1.3: *The 'Los Manantiales' restaurant concrete shell structure from 1956. [22]*

Heinz Isler was a Swiss engineer and is often regarded as one of the great pioneers of shell structures. He gained renown for his experimental form finding methods using physical models. In 1959 at the first ever IASS (International association for Shell Structures) conference Heinz presented his idea of shaping shell structures as inverted hanging cloths [10].

Figure 1.1.4: *Heimberg Tennis Center, Switzerland*

Edmund (Ted) Happold studied Civil Engineering at Leeds University. After graduation he worked for architects Alvar Aalto and Basil Spence before joining Arup. He became the head of Structures 3 at Arup in 1967 and worked on landmark projects like the Sydney Opera and Centre Pompidou. Ted collaborated closely with Frei Otto, Ian Liddell and many other of the pioneers in lightweight structures. In 1976 when he became a professor in Architecture and Engineering at Bath university, he left Arup together with 7 other colleagues and started his own firm. The firm is called Buro Happold and is today a world renown engineering consultancy with over 1400 employees and 27 offices spread out with a global reach [6].

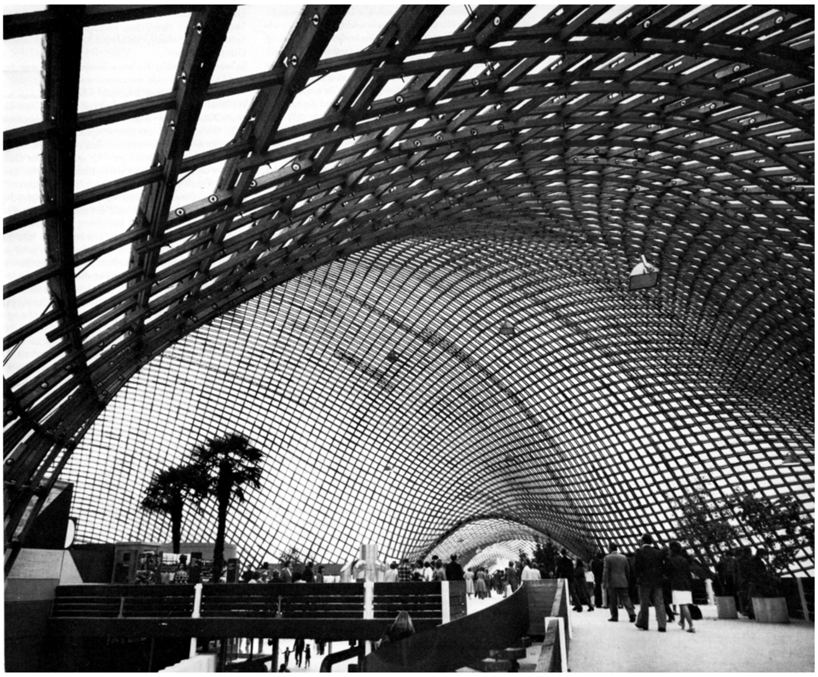


Figure 1.1.5: *An interior photograph of Mannheim Multihallen, designed and constructed by Frei Otto in collaboration with Buro Happold in 1975.*

Recent examples of free form architecture

Frank Garyh is a canadian-american architect based in Los Angeles. Garyhs work is said to be among the most important contemporary architecture and he himself is said to be one of our times most important architects [26]. Garyh dreamt about becoming a sculptor before choosing the route of architecture, something that is clearly reflected in his work as an architect.

Figure 1.1.6: *The Guggenheim museum in Bilbao which is among the first examples of computer based free form architecture [26].*

Zaha Hadid is an Iraq-British architect and one of the biggest names within the deconstructionism style. Already with a degree in mathematics, Zaha moved to London in 1972 to study architecture at Architectural Association School of Architecture. She joined Office for Metropolitan Architecture after her degree and opened her own office in 1979 [29]. Zaha architects have manifested themselves as masters of fluid form computer generated architecture.

Figure 1.1.7: *The alpine funicular railway designed by Zaha Hadid Architects and opened in 2007 [29]. Seemingly a computer generated sculptural structure, where the form is prioritized over the structure.*

1.1.3 Form finding

The nature of long span, shell and membrane structures relies on the integration of force, function, geometry and material. The design of this type of light weight structures is heavily constrained by the laws of nature. In the pursuit of buildings with high-level performance an understanding of spaces, people and culture, as well as knowledge of how to dictate the flow of forces has to be part of the early design process.

By minimizing the bending moment and shear force in a structure, less material is needed, leading to an economical and environmentally sustainable design. The lightness and slenderness of these structures also give them an aesthetically appealing appearance. However the shaping of such structure has to undergo a form finding process dictating the resulting shape. Unlike contemporary free-form architecture, defined purely mathematically (with the aid of computer based 3D CAD), form-finding shapes are defined by the structures initial geometry, the applied form finding load and the boundary conditions, making the design process much more involved.

Form finding is a process used to find the geometry for a load-carrying surface stressed structure. These types of structures are geometrically complex and have to be designed and fabricated with high precision, in order to work efficiently [16]. Form finding has traditionally been carried out using physical or computer based numerical modeling, often a combination of the two. However computer based modeling has today taken the over hand since it offers better control and precision. Surface stressed structures are usually separated into two categories: tensile and compression structures. Tensile structures referring to membranes or cable nets and compressive structures referring to shells and grid shells. The form finding process is similar for the two types, however with some important differences.

Tensile structures

A tensile structure is a configuration of structural elements that carry load through pure tension. An example of a tensile structure is membranes, usually constructed as a set of continuous surfaces, carrying load through membrane action. An other type of tensile structure is cable nets, similar to membranes in form and function but discretized into cable elements carrying the load through axial tension. Most tension structures however are combinations of cables and membranes, especially for larger venues. An example of a tensile structure is the Olympic Stadium in Munich which can be seen in Figure 1.1.2. Common materials for fabric membranes are PVC, coated polyesters or PTFE coated fibre glass. Cables on the other hand are usually made out of steel, stainless steel or some polyester. Tensile structures have the ability to span remarkably long distances with very little material, making them the most efficient of structures.

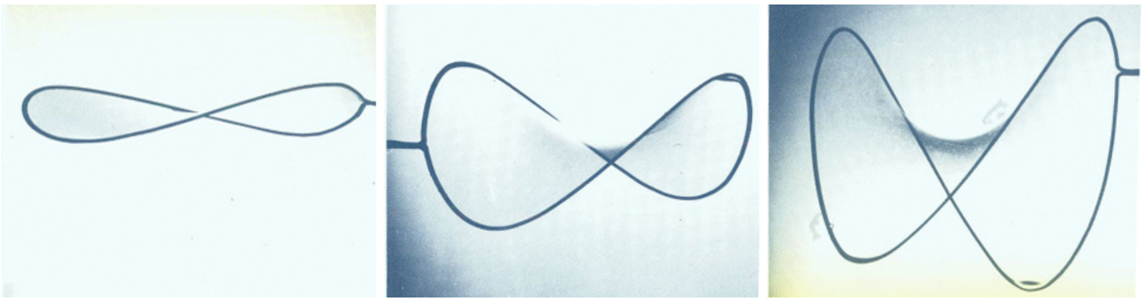


Figure 1.1.8: *A physical model to form find a membrane geometry using steel wire and soap film made by Frei Otto. The soap film takes the form of a minimal surface and changes as the boundary changes [23].*

The aim of form finding for a tensile structure is to determine the geometrical configuration where the loaded membrane does not slack nor crease. This is achieved through an iterative process where the nodes in the initial geometry are moved in the direction of their out-of-balance force to find static equilibrium. The force induced on the nodes comes from prestress applied to the adjacent elements. Membranes are usually constructed with orthotropic fabrics, with different properties for the two main directions (the warp and weft direction). Thus the prestress induced in the membrane elements can be set to one magnitude for the warp and another one for the weft direction. This prestress is held constant throughout the form finding process, and since no other loading is applied the prestress is the governing form driver. If the prestress is set to the same magnitude for the two directions (which is usually the case), the form found surface will be a true minimal surface.

With the aid of physical modeling, a similar result can be obtained by using soap film or elastic fabric [16]. By using soap film modeling, the resulting form is close to a minimal surface where the tension is equal in both directions. With elastic fabrics on the other hand it is the prestress induced in the two main directions that dictates the form. It can be beneficial with physical models for designers to experience the stiffness and flow of forces. But it is difficult to control the magnitude of the prestress, making the detailed form elaboration challenging. On top of that, physical models are difficult to measure and translate into a full scale structure. With the aid of computers this process can be controlled in more detail, enabling the designer to compromise the form to meet other constraints. The prestress can be set to different values in different directions and cables with a much higher level of stiffness can be modeled, enabling a more elaborate design process [16].

Once the form is defined, the structure is analyzed under various load cases by calculating elastic deformations and stress concentrations in the fabric and the cables. This analysis technic is usually referred to as elastic control and the technic for form finding by dictating the prestress is called *Force control*.

Compressive structures

A shell structure is a double curved continuos solid surface formed to work in pure compression, usually made from some composite material like concrete. A gridshell on the other hand has similarities to a shell in form and function but discretized into a grid of bars. These bars can be made up from any kind of material: steel, aluminum, wood or even cardboard tubes [9]. Generally, grid shells made from metallic materials are built up with straight members welded at the joints, whereas grid shells of timber can be made from continuous planks, bent into shape and pinned together with the crossing planks. Examples of gridshells and a shell structures can be found in Figure 1.1.5 and Figure 1.1.4, respectively.

The form finding process for compression structures is somewhat different compared to form finding of tensile structures. There is usually no prestress in the elements and the governing driver of the form is the load (rather then the prestress). The applied load in the form finding process differs between projects, however, only self weight seems to often be the case. If the structure is a shell or a grid shell, it is discretized into membrane elements or cable elements. During the form finding process, the load is inverted (such that it is directed upwards, usually along the positive z-direction), resulting in a structure that is working in pure tension, since the elements can't take axial compression or bending. When the equilibrium geometry is obtained, the elements are converted into plates or beams and the load is inverted (directed downwards, usually along the negative z-direction). For this particular load case the structure is then working in pure compression and bending moments and shear forces are minimized. Finally an envelop of load cases accounting for both dead and live loads are applied, the structure is analyzed and the elements are then sized to cope with the worst case scenario.

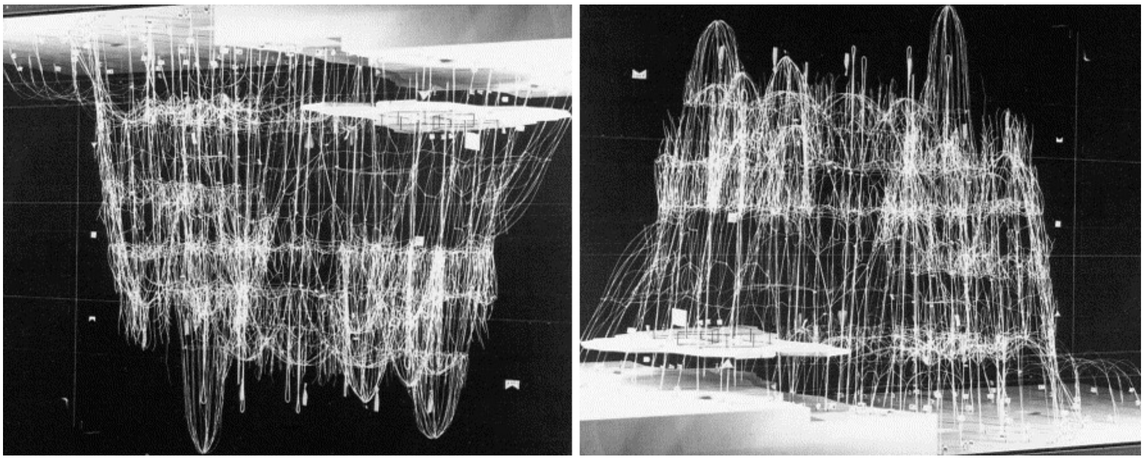


Figure 1.1.9: *A reconstruction of physical form finding model made by Antonio Gaudi. The idea is to distribute the load on the structure as weights hung in chains, the pure tensile hanging geometry is then inverted forming a compression structure optimized for that particular load case [32].*

Structural efficiency

Structural performance is an important aspect of free form architecture. A reduction in structural material often means reduced costs and environmental foot print. However, form finding applied on free form compression structures tends to strongly dominate the form, sometimes in a way where other parameters have to be compromised. It is therefore of interest to develop a process to enable a trade between structural efficiency and other architectural design parameters. This is elaborated in the results section below.

For the case of structures working predominantly in compression, it is a well known fact that axial force is a more material efficient way of carrying load compared to bending and shear force. Hence to enhance the efficiency of a structure the aim is to reduce the bending moments and shear forces. This is exactly what form found surface stress structures do, carrying load through axial/membrane action rather then bending action, making them slender and light weight. But in order to compromise the form for non-structural parameters, it might be of interest to allow parts of a structure working in bending whilst other parts are working in axial/membrane action. To be able to make this trade off, the form finding needs to be carried out using elements that have both bending and axial capability, i.e. beam elements.

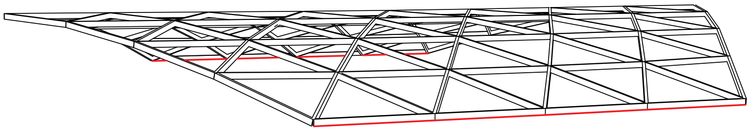


Figure 1.1.10: *The initial geometry of an unoptimized structure with two fixed boundaries which are colored red.*

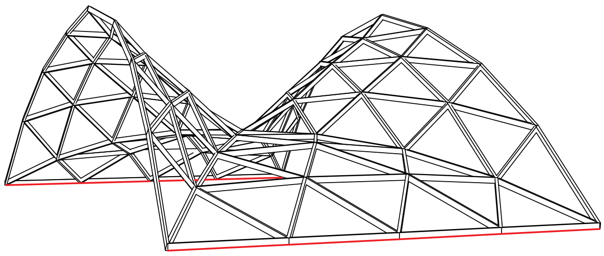


Figure 1.1.11: *The geometry for the structure which is form found with the classical hanging chain model approach. The form finding clearly dominates the resulting form.*

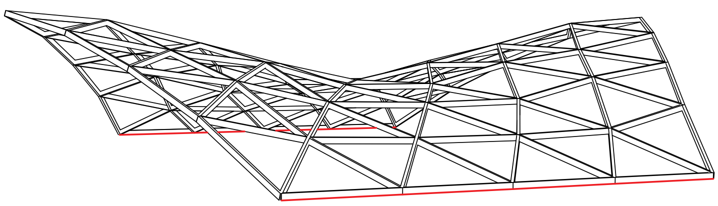


Figure 1.1.12: *The geometry for the structure which is form found using elements with bending capability. By controlling the bending capacity it is possible to trade off between initial and fully form found geometry, allowing for other design parameters to dictate the form.*

1.1.4 SMART Form

In the way we conduct the design of architecture or any other product or feature, the the most important design decisions are the ones made at an early stage [20]. The concept will set the path for the rest of the design, thus, it can improve the process or make it complicated, which might result in a lesser and/or more expensive product. Unlike the design industry where the final product is mass produced and sold in great volumes, each project in architecture tends to be bespoke to fit a different site and context. Whereas product designers can make great revenue by renewing and optimizing a product on a relative long term basis, an architect or engineer has to invent a unique building concept and explore its potential at a great speed. Putting the much different complexity of building design into the equation, together with an urge to obtain high level performance, it stands clear that the two related industries have different needs in terms of tools. However many of the tools used in building design were initially developed for other purposes. The analysis of a structural concept does not often require the accuracy and complexity of software developed for cars and airplanes. Likewise, modeling in architecture has other needs of precision compared to product design or film production. The use of computational power in the architectural design could potentially be better spent by enabling parametric modeling, combined with real time lightweight analysis, making it possible to quickly generate and evaluate many permutations of a concept.

The cap in parametric computer tools developed for architecture is something that avantgarde engineers and architects have worked to fill for the last decade or so. A number of software packages have been developed for this purpose, including *Generative Components* and *Grasshopper*. Today development is rather focused on integrating analysis into the parametric design environment. Buro Happold with their research team SMART Solutions are, with their extensive knowledge and experience in advanced structures, adding on to this development with in-house written analysis software for performance based design.

SMART Form is a software, written as a free plug-in developed by Buro Happold, for the 3D modeling software Rhinoceros. It combines analysis and rationalization of complex geometries, with form finding and sculpting of structures, driven by a dynamic relaxation solver. The plug-in transforms the 3D modeling software to a virtual physical environment under the influence of various forces and a gravitational field, enabling sketching and sculpting of efficient compression (gridshells, shells) and tension structures (cable nets, tents). Prior to this thesis, SMART Form had the capability of doing conceptual analysis with bar elements. Element properties were calibrated to be scale independent, rather than controlling material and section sizes the user could control stiffness and initial length. Hence, the software was doing proper structural analysis with bar elements, but did not output any quantifiable metrics like weight, forces and moments. Loading was controlled by the user setting the magnitude of the gravitational field.

In essence, SMART Form is a tool for for sculptural design with the foundation in material and structural principles. The development of the tool takes part of the exploration of 4D CAD, modelling and analysis based on real time feedback. More information about SMART Form and the latest free version can be found on [21].

1.2 Purpose

The purpose of this thesis is to advance Buro Happold's real time form finding tool SMART Form, by integrating beam elements and bending capability into the form finding process, allowing for a compromise between initial geometry and fully form found geometry. Automatic sizing is then implemented and the question of how form and size is related is raised and tackled. The thesis leads to an investigation of how to conduct conceptual structural design and how this process could potentially be improved or approached differently, with the means of real time feedback.

In order to conduct structural design in a playful, light weight and conceptual way, real time analysis is essential. SMART Form is based on dynamic relaxation as a solver, which is beneficial since intermediate steps between initial geometry and equilibrium geometry are calculated, thus can be visualized to provide real time feedback. The implementation of beam elements and automatic sizing will be built around the same dynamic relaxation solver.

1.3 Method

The thesis work started with a general literature review about form finding of structures and the application of dynamic relaxation to projects within the category of light weight structures. The focus was to get an understanding of the existing technics for form finding and how they differ between tensile and compression structure. Gradually, the literature review got more focused on the mathematics behind dynamic relaxation for bars and beam elements.

The development of SMART Form started with a couple of days of work restructuring the existing code to enable implementation of new functionality. Thereafter followed a time of software development, advancing SMART Form by implementing the new types of elements. SMART Form is written as a plug-in for the 3D modeling software Rhinoceros using Rhino DotNet SDK, based on the C# programming language. This was the platform for the development. At Buro Happold, Visual Studio 2005 is used as the programing environment for software development.

During the phase of software development, meetings were arranged with engineers at Buro Happold to get their feedback into the loop, making sure the tool was developed in a direction that would fit their design process.

A couple of test cases were developed to test the tool, partly during the development of functionality, and partly for guidance when designing the user interface. Models from real projects where Buro Happold have previously been involved, were also used for the same purpose. In order to validate the implemented elements, a convergence study was performed on three different test cases. The results from SMART Form were then compared with the analytical solutions (for the simplest case) or with analysis from Matlab (with the finite element toolbox CALFEM) and the commercial FEM software GSA.

Furthermore elaboration with form finding methods using beam elements was performed and three different approaches implemented. Automatic size optimization was implemented based on knowledge from discussions with engineers at Buro Happold and additional literature reviews.

Finally the work was consolidated into a finished plug-in with video demos demonstrating the new functionality in action.

1.4 Limitations

The thesis is focused on solving the statical equilibrium equations for structural analysis using the iterative technique dynamic relaxation. While geometrical nonlinearity is considered as a result of the iterative solver, so is not the case for material non-linearity. The implemented material models are generally simplified to isotropic materials.

The aim with the plug-in is to function as a tool for conceptual structural design rather than analysis for detailed design. Instead of giving information about the stress distribution in different structural parts, the intent is to give useful feedback on the overall form. Whether a structure is working in bending or membrane action, roughly what beam sections are needed for a structure exerted to a certain load, and how the sizes and weight varies when the form of the structure changes. To be able to give feedback on section sizes, automatic sizing is implemented based on the methodology for member capacity calculations presented in Eurocode 3. These equations are simplified to be more suitable for conceptual analysis and partial coefficients and safety factor for loading are neglected. The analysis of beam elements is also limited to closed hollow or solid cross sections, which furthermore helps to simplify these equations.

Even though programming has been a significant part of the thesis work, it is not presented in detail in the report, since it is considered as a tool for the development rather than the subject of interest.

1.5 Outline of the report

The first part of the report is an introduction to the topics performance driven design, use of computers in the design process, and form finding of light weight structures. An attempt is made to present these subjects from a historical point of view, but also to show their place in a more contemporary context. The software SMART Form, which is the subject for the development is then introduced in relation to these topics.

The introduction is followed by a theory chapter, which covers most of the physics and mathematics used in the structural analysis. The main part consists of an introduction to a vector based beam theory, showing how to calculate local forces and moments based on displacements and rotations. It is then shown how these forces and moments are converted to a global coordinate system and applied to the nodes in a structure. A theoretic foundation is presented for the application of dynamic relaxation to translational and rotational degrees of freedom. It is then shown how the static equilibrium can be obtained using dynamic relaxation for beam elements in a high level algorithm trying to give a holistic view of the process.

The next section in the theory chapter is focused on form finding using beam elements where three different approaches are presented. Finally two inherently different approaches to automatic sizing is touched upon and presented together with theory for calculations of stress based utilization and local and global buckling. In chapter following, speed and convergence of dynamic relaxation is treated.

A quick presentation of the user interface for SMART Form is given to bridge between the theory chapter and the case studies presented in the results chapter.

In the results chapter, it is shown how the new functionality can be used for conceptual structural design in three different case studies. The first one shows how to perform form finding for a simple grid shell using cable elements and how the height of the roof relates to the need of cross section sizes i.e. weight of the structure. It is also shown how to quickly evaluate the structural effects of changing cross sections, materials and orientations. In the second user story it is shown how beam elements can be used in form finding where a comparison is made between the initial form, a partly form optimized and a fully form found structure.

The last case study is based on a real project where Buro Happold have been involved in the design of a grid shell roof. The design process that the engineers and architects went through for the structural concept, is presented together with a proposal for a new approach based on the new implemented analysis with real time feedback.

Finally, a discussion about the thesis is given together with recommendations for further work.

2 Theory

This chapter covers most of the physics and mathematics used in the structural analysis. The first part consists of an introduction to a vector based beam theory, showing how to calculate local forces and moments based on displacements and rotations. It is then shown how these forces and moments are converted to a global coordinate system and applied to the nodes in a structure. A theoretic foundation is presented for the application of dynamic relaxation to translational and rotational degrees of freedom. Thereafter it is shown how the static equilibrium can be obtained using dynamic relaxation for beam elements in a high-level algorithm trying to give a holistic view of the process. Three different approaches to form finding using beam elements are presented. And finally, a method for automatic sizing is implemented and the theory is presented.

2.1 Beam theory

In this section a description of the equations involved in the analysis of beam elements is presented. A vector based beam theory particularly suitable for 3D dynamic relaxation is introduced. For a more detailed description of the theory and derivations of the equations presented in this chapter the reader is referred to Sigrid Adriaenssens PhD thesis *Stressed Spline Structures* [1].

2.1.1 Coordinate systems and definitions

Each node in the structure is given a coordinate system and 6 degrees of freedom (DOF) by allowing translations and rotations about the nodal x,y,z axes. Equally each beam is given two separate coordinate systems, one for each end, giving the element 12 DOF. The orientation of the beam coordinate systems is measured in reference to a vector $\boldsymbol{p}$ between the ends of the beam. Initially the z-axis for each of the two beam coordinate systems is aligned with this $\boldsymbol{p}$-vector. As the coordinate system of the adjacent nodes rotate, so does the beam end coordinate systems and the resulting angles, measured in reference to $\boldsymbol{p}$, causes bending and twisting moments in the element. When the node coordinate system translates, so does the adjacent beam end coordinate systems, causing elongation or compression of the members resulting in axial force.

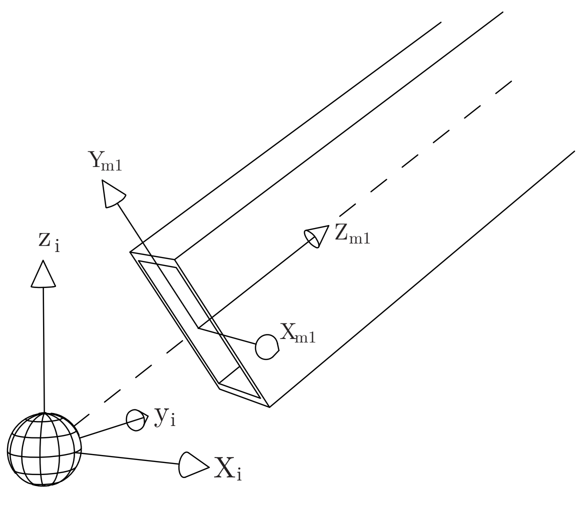


Figure 2.1.1: *The coordinate system of node i and the coordinate system of end* 1 *of adjacent beam* m.

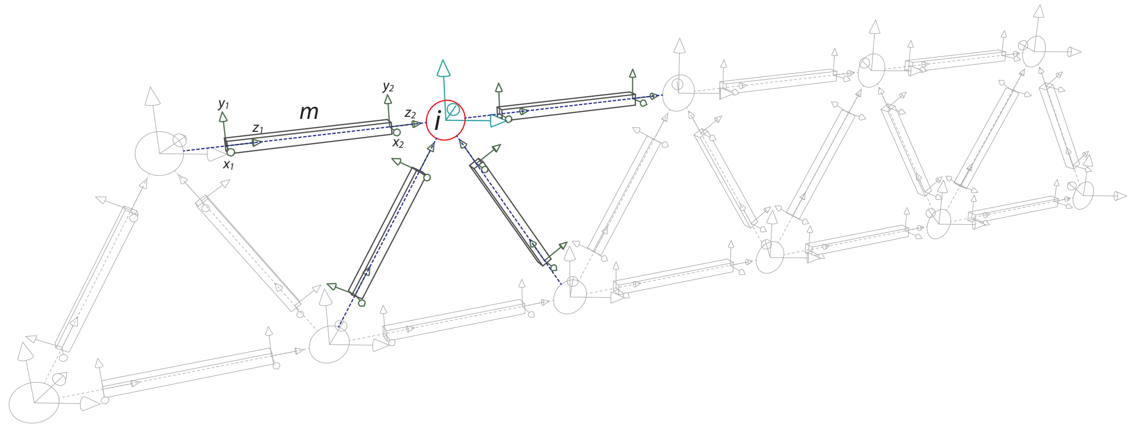


Figure 2.1.2: *An example of the arrangement of elements and coordinate systems of the truss modeled with beam elements. Each element adjacent to node i will contribute with force and moments at the node.*

The forces and moments generated in the elements are applied to the nodes and summed to give out-of-balance force and out-of-balance moment, which is driving the rotations and translations of the next iteration, resulting in new rotations and translations. This is repeated iteratively until the nodes have reached an equilibrium position where loading and element forces and moments balance each other out. However, the process is started from an initially unstressed geometry, where it is the sudden application of load that triggers the process by putting the nodes out of equilibrium. This way of finding the equilibrium geometry can be described in analogy with the more conventional matrix formulation based on the element force, stiffness and displacement equation

$$\boldsymbol{f}^e = \boldsymbol{K}^e \boldsymbol{a}^e, \tag{2.1.1}$$

where

$\boldsymbol{K}^e$ is the beam element stiffness matrix,

$\boldsymbol{a}^e$ is the displacement vector containing translations and rotations,

$\boldsymbol{f}^e$ is the load vector containing forces and moments.

For the detailed description of eq. (2.1.1) the reader is referred to [14].

The translations and rotations in vector $\boldsymbol{a}^e$ are multiplied with the element stiffness matrix $\boldsymbol{K}^e$ giving the element force vector $\boldsymbol{f}^e$. By summing $\boldsymbol{f}^e$ with the element load vector $\boldsymbol{f}^l$, the new out-of-balance force and out-of-balance moments are calculated. These forces and moments are then used in the dynamic relaxation process to move and rotate the nodes giving the new displacement vector $\boldsymbol{a}^e$ for the next iteration. Rather than assembling a global stiffness matrix and solving the system of equations in one go, dynamic relaxation can be seen as an node-wise iterative approach.

2.1.2 Displacements and rotations

Rotation X

The rotations, θ_{x1} and θ_{x2}, about the local x-axes of the beam element are calculated according to

$$\theta_{x1} = \frac{\boldsymbol{y}_1 \cdot \boldsymbol{p}}{L}, \qquad \theta_{x2} = \frac{\boldsymbol{y}_2 \cdot \boldsymbol{p}}{L}. \tag{2.1.2}$$

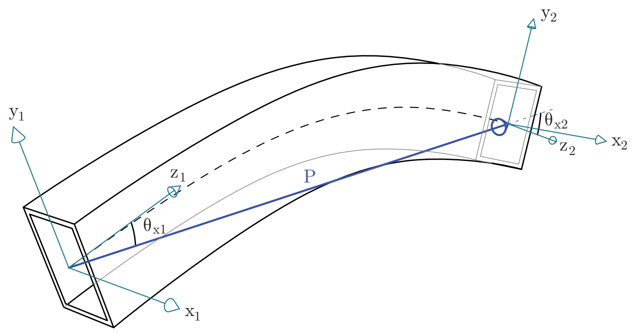


Figure 2.1.3: *The rotational displacement of a beam due to the rotation of its end coordinate systems around the local x-axes. The dashed line shows the centerline of the element and $\boldsymbol{p}$ is the element vector between the two ends.*

Rotation Y

The rotations, θ_{y1} and θ_{y2}, about the local y-axes of the beam element are calculated according to

$$\theta_{y1} = \frac{\boldsymbol{x}_1 \cdot \boldsymbol{p}}{L}, \qquad \theta_{y2} = \frac{\boldsymbol{x}_2 \cdot \boldsymbol{p}}{L}. \tag{2.1.3}$$

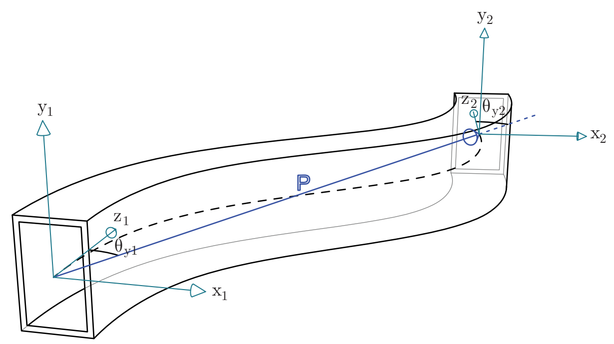


Figure 2.1.4: *The rotational displacement of a beam due to the rotation of its end coordinate systems around the local y-axes. The dashed line shows the centerline of the element and $\boldsymbol{p}$ is the element vector between the two ends.*

Rotation Z

The rotation, ϕ, about the local z-axes of the beam element is calculated according to

$$\phi = \frac{\boldsymbol{x}_1 \cdot \boldsymbol{y}_2 - \boldsymbol{x}_2 \cdot \boldsymbol{y}_1}{2}. \tag{2.1.4}$$

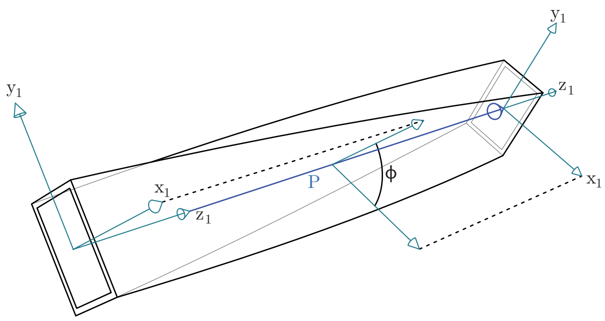


Figure 2.1.5: *The twist of beam m due to the rotation of its end coordinate systems around the local z-axes. The vector* $\boldsymbol{p}$ *is the element vector between the two ends.*

Axial displacements

The elongation or shortening of a beam element based on the translational displacements of the end nodes is calculated at time t according to

$$e_a = \frac{L_t^2 - L_0^2}{2L_0}. \tag{2.1.5}$$

Here

$$L_t = \sqrt{(x_2^t - x_1^t)^2 + (y_2^t - y_1^t)^2 + (z_2^t - z_1^t)^2}. \tag{2.1.6}$$

The total axial deformation of an element has a contribution from the elongation caused by the element bowing, which is calculated from

$$e_b = \frac{L_0^2[4(\theta_{x1}^2 + \theta_{y1}^2) - 2(\theta_{x1}\theta_{x2} + \theta_{y1}\theta_{y2}) + 4(\theta_{x1}^2 + \theta_{y1}^2)]}{2L_0}. \tag{2.1.7}$$

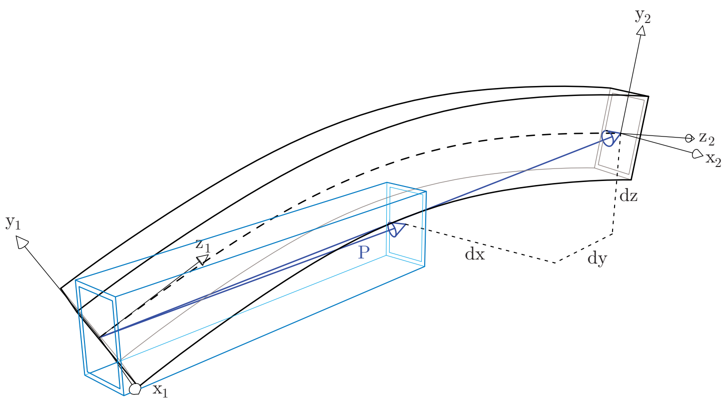


Figure 2.1.6: *The deformation of a beam due to translations and rotations of its end coordinate systems. The vector* $\boldsymbol{p}$ *is the element vector between the two ends.*

2.1.3 Local forces and moments

Subsequently, local forces and moments can be calculated form the rotations and displacements presented in the previous section. The equations are derived based on the relationship between stress, strain and material properties expressed in terms of energy. Since energy is a scalar quantity it is a convenient metric for the formulation of the governing equations for elastic bodies in solid mechanics. In the absence of energy losses due to friction and heat, the external work applied to an elastic beam causing it to distort, can be expressed in terms of strain energy, which is a form of potential energy.

The total strain energy in a beam element caused by its three deformation modes (axial deformation, bending and twisting) can be written as [1]

$$U = \frac{EA}{2L_0}(e_a + e_b)^2 + \frac{EI_x}{2}\int_L \kappa_x^2 dl + \frac{EI_y}{2}\int_L \kappa_y^2 dl + \frac{GJ}{2L_0}\phi^2, \tag{2.1.8}$$

where

E is the Young's modulus,

G is the shear modulus,

I_x, I_y is the second moment of area around x and y-axis respectively,

A is the cross section area,

κ_x, κ_y is the curvature around x and y-axis respectively,

J is the torsional constant,

L_0 is the length of the member,

ϕ is the twist angle.

Axial force caused by nodal translations and element bowing is calculated by differentiating eq. (2.1.8) with respect to e which yeilds

$$N = \frac{\partial U}{\partial e_a} = \frac{EA}{L_0}(e_a + e_b). \tag{2.1.9}$$

Equally, the moment equations are obtained by differentiating eq. (2.1.8) with respect to the angles θ_{x1} and θ_{x2}

$$\begin{bmatrix} M_{x1} \\ M_{x2} \end{bmatrix} = \begin{bmatrix} \frac{\partial U}{\partial \theta_{x1}} \\ \frac{\partial U}{\partial \theta_{x2}} \end{bmatrix} = \frac{NL_0}{30}\begin{bmatrix} 4 & -1 \\ -1 & 4 \end{bmatrix}\begin{bmatrix} \theta_{x1} \\ \theta_{x2} \end{bmatrix} + \frac{EI_x}{L_0}\begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}\begin{bmatrix} \theta_{x1} \\ \theta_{x2} \end{bmatrix}, \tag{2.1.10}$$

and for the moments around the y-axes,

$$\begin{bmatrix} M_{y1} \\ M_{y2} \end{bmatrix} = \begin{bmatrix} \frac{\partial U}{\partial \theta_{y1}} \\ \frac{\partial U}{\partial \theta_{y2}} \end{bmatrix} = \frac{TL_0}{30}\begin{bmatrix} 4 & -1 \\ -1 & 4 \end{bmatrix}\begin{bmatrix} \theta_{y1} \\ \theta_{y2} \end{bmatrix} + \frac{EI_y}{L_0}\begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}\begin{bmatrix} \theta_{y1} \\ \theta_{y2} \end{bmatrix}. \tag{2.1.11}$$

For the twisting moments, the derivative is taken with respect to the twist angle ϕ, yeilding

$$M_\phi = \frac{\partial U}{\partial \phi} = \frac{GJ}{L_0}(\phi). \tag{2.1.12}$$

2.1.4 Shape function

The shape of each beam element can be defined by a cubic shape function in terms of a parameter u. Consider an element m and let vectors r_1 and r_2 from the origin define the start and end nodes of the element respectively. Let the parameter at the start of the elements be $u = -0.5$ and the parameter at the end be $u = 0.5$ [1].

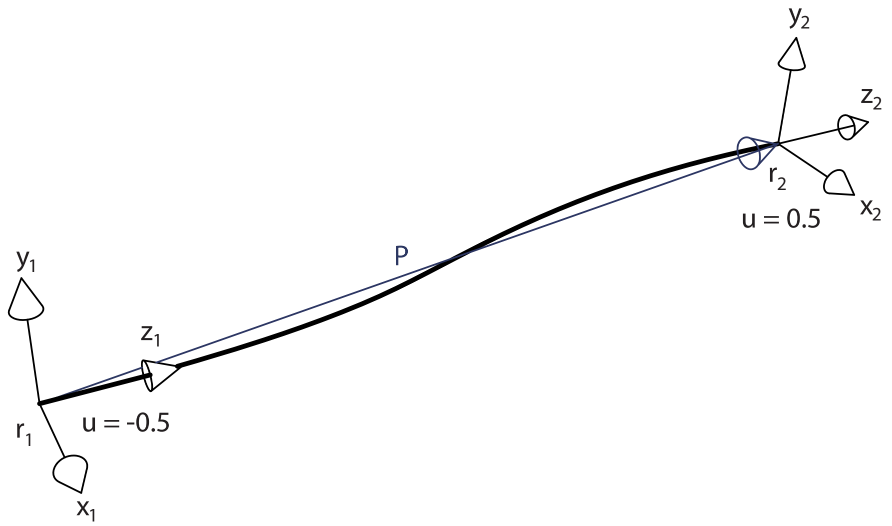


Figure 2.1.7: *Beam element defined by a cubic shape function.*

Any vector $\boldsymbol{r}$ from the origin to any point on the element can then be described by the function

$$\boldsymbol{r} = \frac{\boldsymbol{r}_1 + \boldsymbol{r}_2}{2} + \boldsymbol{p}u + \left[\frac{L^2\boldsymbol{z}_1}{\boldsymbol{z}_1 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](u^3 + \frac{1}{2}u^2 - \frac{1}{4} - \frac{1}{8}) + \left[\frac{L^2\boldsymbol{z}_2}{\boldsymbol{z}_2 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](u^3 - \frac{1}{2}u^2 - \frac{1}{4} + \frac{1}{8}). \quad (2.1.13)$$

Differentiating eq. (2.1.13) with respect to u twice yeilds

$$\frac{d^2\boldsymbol{r}}{du^2} = \left[\frac{L^2\boldsymbol{z}_1}{\boldsymbol{z}_1 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](6u+1) + \left[\frac{L^2\boldsymbol{z}_2}{\boldsymbol{z}_2 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](6u-1). \quad (2.1.14)$$

And the curvature around the x-axis

$$\kappa_x = \frac{\boldsymbol{y}_1 \cdot \left[\frac{L^2\boldsymbol{z}_1}{\boldsymbol{z}_1 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](6u+1) + \boldsymbol{y}_2 \cdot \left[\frac{L^2\boldsymbol{z}_2}{\boldsymbol{z}_2 \cdot \boldsymbol{p}} - \boldsymbol{p}\right](6u-1)}{L^2}, \quad (2.1.15)$$

$$\kappa_x = \frac{\theta_{x1}(6u+1) + \theta_{x2}(6u-1)}{L}. \quad (2.1.16)$$

Equally for the curvature around the y-axis

$$\kappa_y = \frac{\theta_{y1}(6u+1) + \theta_{y2}(6u-1)}{L}. \quad (2.1.17)$$

The moment and distribution due to bending around the x-axis can be calculated based on the curvature as

$$M(u)_x = \kappa(u)_x EI_x, \quad (2.1.18)$$

$$(2.1.19)$$

Equally for the moment and shear force around the y-axis

$$M(u)_y = \kappa(u)_y EI_y, \quad (2.1.20)$$

$$(2.1.21)$$

2.1.5 Global forces and moments

In the previous sections it has been shown how to calculate the rotations and translations of the end nodes for a beam element and how to calculate local forces and moments based on these displacements. However in order to apply local forces and moments to the nodes a transformation from to global coordinates has to be carried out.

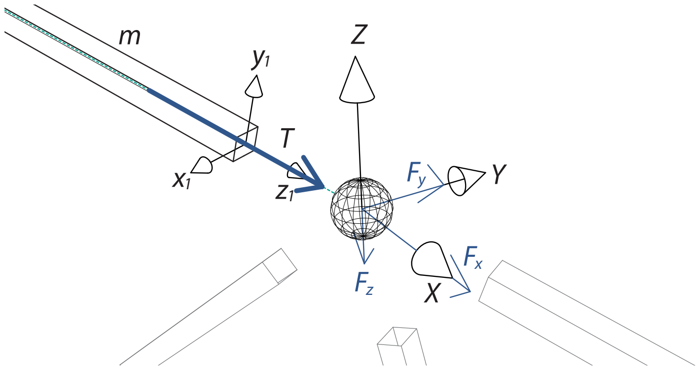


Figure 2.1.8: *The local force T at element m transformed into global forces at a node.*

The local axial force is transformed to global coordinates using the expression in eq. (2.1.23). The first term in the equation is basically a decomposition of the force into x, y and z coordinates, whereas the other terms are the force contribution from the moments.

The force applied to the first end is calculated from

$$F_{1i} = \frac{1}{L_0}(Np_i + M_{x1}y_{1i} - M_{y1}x_{1i} + M_{x2}y_{2i} - M_{y2}x_{2i}). \tag{2.1.22}$$

And the force applied to the second end is calculated from

$$F_{2i} = -\frac{1}{L_0}(Np_i + M_{x1}y_{1i} - M_{y1}x_{1i} + M_{x2}y_{2i} - M_{y2}x_{2i}). \tag{2.1.23}$$

Here i takes the form x, y and z.

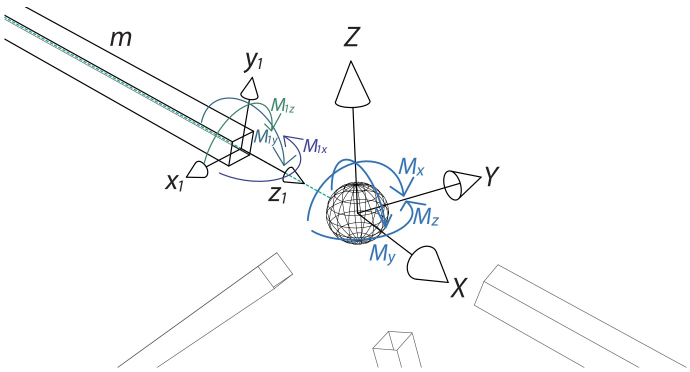


Figure 2.1.9: *Local moments at element m transformed to global moments at a node.*

The moments applied to the first end are calculated from

$$M_{1i} = -\epsilon_{ijk}\left(M_{x1}\frac{p_k y_{1j}}{L_0} - M_{y1}\frac{p_k x_{1j}}{L_0} + M_\theta \frac{(x_{1j}y_{2k} - y_{1j}x_{2k})}{2}\right). \tag{2.1.24}$$

And for the second end the applied moments are calculated based on

$$M_{2i} = -\epsilon_{ijk}\left(M_{x2}\frac{p_k y_{2j}}{L_0} - M_{y2}\frac{p_k x_{2j}}{L_0} - M_\theta\frac{(x_{1j}y_{2k} - y_{1j}x_{2k})}{2}\right). \tag{2.1.25}$$

Here

i, j, k takes the form of 1, 2, 3 (representing x,y,z).

ϵ_{ijk} is the 3D permutation tensor.

ϵ_{ijk} = 1, if $i = 1$, $j = 2$, $k = 3$ or a permutation of these values.

ϵ_{ijk} = -1, if $i = 3$, $j = 2$, $k = 1$ or a permutation of these values.

2.1.6 Rotations

Rotation vector

Any rotation can be defined as a scalar β for the angle and a vector $\boldsymbol{b}$ as the axis of rotation. A vector $\boldsymbol{x_0}$ can be transformed from its initial position (x_0, y_0, z_0) to its current position $\boldsymbol{x_t}$ at (x_t, y_t, z_t) using a rotation vector $\boldsymbol{b}$, where the relation between axis of rotation and angle is

$$|\boldsymbol{b}| = \tan\left(\frac{\beta}{2}\right). \tag{2.1.26}$$

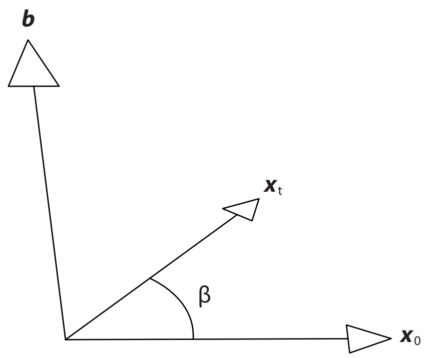


Figure 2.1.10: *The rotation of a vector $\boldsymbol{x}_0$ to $\boldsymbol{x}_t$ through rotation vector $\boldsymbol{b}$.*

Net result of two rotations

In the application of rotation vectors to the beam end coordinate systems, the rotation axes are the global x, y,$zaxesandtheanglesarecalculatedfromeq.$ (2.1.2), $eq.$ (2.1.3), $eq.$ (2.1.4)$.Whenrotatingthenodesaroundthegl o$ z or z, y, x). To get round this issue the concept of nonlinear vector rotations is implemented [1].

$$\boldsymbol{c} = \frac{(\boldsymbol{a} + \boldsymbol{b} + \boldsymbol{b} \times \boldsymbol{a})}{(1 - \boldsymbol{a} \cdot \boldsymbol{b})} \tag{2.1.27}$$

Application in DR

From eq. (2.1.27) the resulting rotation vector from two rotations is calculated. In order to calculate the rotation of a node in three dimensions the nonlinear vector rotation is used twice.

$$\begin{array}{cccccc} \boldsymbol{r}_x & \& & \boldsymbol{r}_y & \rightarrow & \boldsymbol{c} \\ \boldsymbol{r}_z & \& & \boldsymbol{c} & \rightarrow & \boldsymbol{d} \end{array}$$

Here

$\boldsymbol{r}_x$, $\boldsymbol{r}_x$, $\boldsymbol{r}_x$ are rotation vectors for rotation around each of the global coordinate axes x, y, z.

The resulting rotation vector $\boldsymbol{d}$ is then used to calculate a transformation matrix A according to

$$\boldsymbol{A} = \cos(\beta) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \frac{1-\cos(\beta)}{|\boldsymbol{d}|^2} \begin{bmatrix} d_1 d_1 & d_1 d_2 & d_1 d_3 \\ d_2 d_1 & d_2 d_2 & d_2 d_3 \\ d_3 d_1 & d_3 d_2 & d_3 d_1 \end{bmatrix} + \frac{\sin(\beta)}{|\boldsymbol{d}|} \begin{bmatrix} 0 & -d_3 & d_2 \\ d_3 & 0 & -d_1 \\ -d_2 & d_1 & 0 \end{bmatrix}, \quad (2.1.28)$$

where β is obtained from eq. (2.1.26). If the coordinate system axes are defined as

$$\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}, \quad (2.1.29)$$

$$\boldsymbol{y} = \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix}, \quad (2.1.30)$$

$$\boldsymbol{z} = \begin{bmatrix} z_1 & z_2 & z_3 \end{bmatrix}, \quad (2.1.31)$$

the whole coordinate system can be written as

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \boldsymbol{z} \end{bmatrix}, \quad (2.1.32)$$

the transformation of $\boldsymbol{C}$ is done by multiplication with matrix $\boldsymbol{A}$, effectively executing the rotation.

$$\boldsymbol{C'} = \boldsymbol{A}\boldsymbol{C}^{\boldsymbol{T}}, \quad (2.1.33)$$

$$\begin{bmatrix} x'_1 & x'_2 & x'_3 \\ y'_1 & y'_2 & y'_3 \\ z'_1 & z'_2 & z'_3 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}, \quad (2.1.34)$$

where the left hand side are the transformed coordinate vectors.

2.2 Dynamic relaxation

2.2.1 Translational degrees of freedom

Dynamic relaxation (DR) is a numerical method for solving the differential equation of motion by iteratively stepping through time with small increments Δt. The method can be applied to dynamic structural analysis as well as static structural analysis, the later being the case in this thesis. Historically the use of DR as an explicit solution technique for static analysis was introduced by A.S. Day [3]. The technique had previously been applied to tidal flow computations. Day and Otter however reformulated the equations by replacing fluid motion and continuity with damped structural motion and elasticity. The static solution of structural problems can then be seen as the limiting equilibrium position of heavily damped structural vibrations. DR is especially well suited for structural analysis of highly non-linear problems, therefore often applied to analysis of tensile structures.

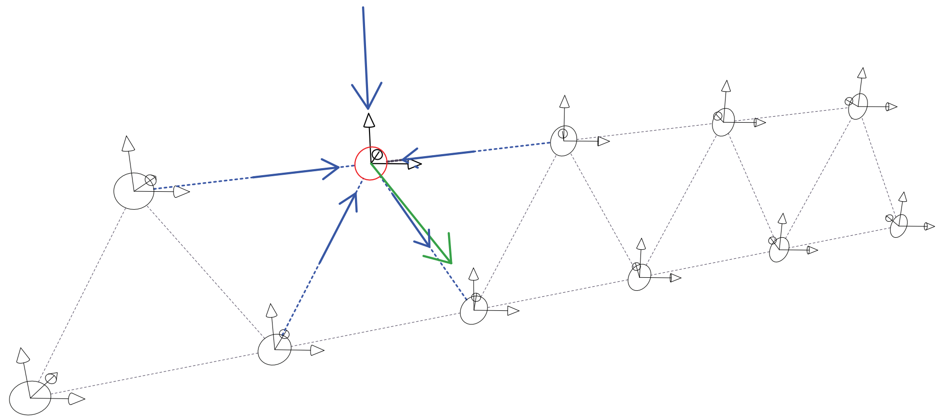


Figure 2.2.1: *A principle sketch of a truss with node i colored in red and its adjacent elements colored in blue. The out-of-balance force at node i is the summation of the external load P_i and the element forces.*

The basis of the method is to iteratively calculate the movement of all nodes in a structure from their initial positions, through time steps Δt, until their equilibrium positions are found and vibrations have died out due to artificial damping. The movement of each node is caused by an out-of-balance force calculated as the sum of the forces acting in the elements adjacent to the node and the applied load. Newtons second law states that the acceleration of a body is directly proportional to the body's net force which lay the foundation for dynamic relaxation. From the out-of balance-force and the nodal mass the acceleration can be obtained, and by stepping through time the velocity and movement of each node can consequently be calculated. Each node is moved in order to reduce the out-of-balance force, and when the total force have reached below a certain limit value, equilibrium is assumed to be reached. A time-stepping method called Euler forward is used throughout the thesis.

The basis for dynamic relaxation is Newtons second law of motion stating

$$F = ma. \tag{2.2.1}$$

This equation can then be rewritten for the displacement of any node i in the x-direction as

$$P_{ix} - K_{ix}\delta_{ix} = M_{ix}a_{ix}. \tag{2.2.2}$$

Where

P_{ix} is the applied force at node i in the x-direction,

K_{ix} is the stiffness at node i in the x-direction,

δ_{ix} is the displacement of node i in the x-direction,

C is the viscous damping constant at node i,

v_{ix} is the velocity of node i in the x-direction,

M_{ix} is the mass at node i in the x-direction,

a_{ix} is the acceleration of node i in the x-direction.

The relations given in eq. (2.2.2) apply equally for displacements in x, y and z direction. Introducing the term R, as the out-of-balance force acting on the node, the left hand side in eq. (2.2.2) can be written

$$R_{ix} = P_{ix} - K_{ix}\delta_{ix}. \tag{2.2.3}$$

Inserted into eq. (2.2.2) yeilds

$$R_{ix} = M_{ix}a_{ix}. \tag{2.2.4}$$

Consider the case in figure (2.2.1) where more than one element contributes with force at node i, eq. (2.2.3) can then be formulated as

$$R_{ix} = P_{ix} + \sum_{m=1}^{n} \frac{E_m A_m}{L_{mx}}(x_i - x_j). \tag{2.2.5}$$

Where

E_m is the Young's modulus for element m,

A_m is the cross section area for element m,

L_{mx} is the length of element m in the x-direction,

$(x_i - x_j)$ is the x-distance between the start and end node of element m.

Inserting eq. (2.2.5) into eq. (2.2.2)

$$P_{ix} + \sum_{m=1}^{n} \frac{E_m A_m}{L_{mx}}(x_i - x_j) = M_{ix}a_{ix}. \tag{2.2.6}$$

Clearly eq. (2.5.11) is a component in a system of second order differential equations, here with the dependent variable x_i, where x_j is treated as a constant. In order to solve such ODE, integration has to be carried out twice with sufficient initial conditions. For the application in structural analysis the initial geometry is known, hence the initial positions x is given at time $t = 0$, and the initial velocity is known to be $v(t = 0) = 0$. This is because the analysis starts from an initial equilibrium position where no loads are applied to the structure.

The integration is carried out using a time stepping method. First the acceleration for node i is calculated according to Newton's second law of motion.

$$a_{ix}^{(t+\Delta t)} = \frac{P_{ix} + \sum_{m=1}^{n} \frac{E_m A_m}{L_{\mathrm{mx}}}(x_i^t - x_j^t)}{M_{ix}^t}. \tag{2.2.7}$$

From the acceleration, the velocity can be obtained by multiplying the acceleration with a time step Δt. The viscous damping scalar C is introduced by multiplication with the velocity from the previous time step

$$v_{ix}^{(t+\Delta t)} = Cv_{ix}^t + a_{ix}^{(t+\Delta t)}\Delta t. \tag{2.2.8}$$

Finally, the positions are updated based on the new velocities

$$x_i^{(t+\Delta t)} = x_i^t + v_{ix}^{(t+\Delta t)} \Delta t. \tag{2.2.9}$$

The equations are nodally decoupled in the sense that the calculation of position $x_i^{(t+\Delta t)}$ for node i only depends on its own residual force and velocity from the previous iteration [3]. From the new nodal positions the forces in the links are recalculated and summarized to the nodes, giving the out-of-balance force driving the movement for the next iteration. If the out-of-balance force for all unconstrained nodes is sufficiently close to zero, convergence is reached and the static equilibrium found.

This method can be applied on a system of bars, springs or cable elements where the main difference is how the element stiffness is calculated. For a cable, however, the force is set to zero if the member is compressed since it cannot work in compression. In order to incorporate more complex elements with bending capability, an extra set of degrees of freedom has to be introduced. This is discussed in the next section.

2.2.2 Rotational degrees of freedom

In this section the implementation of rotations in dynamic relaxation will be presented. Previously it has been shown how the equilibrium equations for bars, springs or cable elements with 3 DOF are formulated and solved using dynamic relaxation. Beam elements however have an additional 3 rotational DOF:s for each node giving the capability to work in bending action. By extending the bar element formulation with the rotational DOF:s a beam element is obtained. To finalize the formulation, the coupling effect between the translations and the rotations, i.e. forces and moments, has to be considered. This is treated more in the next section.

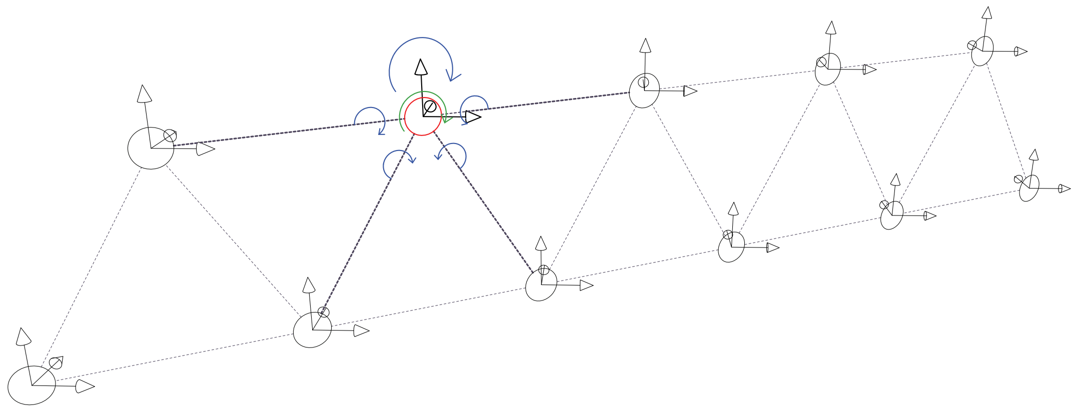


Figure 2.2.2: *Principle sketch of a truss with node i colored in red and adjacent elements colored in blue. The out-of-balance moment (the green arrow) at node i is calculated as the sum of the external torque T_i and the element moments.*

Recalling eq. (2.2.1), which is Newtons second law of motion and the foundation of dynamic relaxation for the translational DOF:s

$$F = ma. \tag{2.2.10}$$

This equation can be rewritten to apply for rotations as

$$T = J\alpha. \tag{2.2.11}$$

Furthermore, for the rotation of any node i around the z-axis, this equation can be expanded to

$$T_{iy} - S_{iy}\theta_{iy} = J_{iy}\alpha_{iy}. \tag{2.2.12}$$

Where

T_{iy} is the torque at node i around the y-axis,

J_{iy} is the moment of inertia of node i around the y-axis,

θ_{iy} is the angular displacement at node i around the y-axis,

α_{iy} is the angular acceleration of node i around the y-axis,

ω_{iy} is the angular velocity of node i around the y-axis,

S_{iy} is the rotational stiffness of node i around the y-axis.

The relation presented in eq. (2.2.12) applies equally well to rotations around the x and z-axes. Introducing H for the out-of-balance torque, the left hand side in eq. (2.2.12) can be written

$$H_{iy} = T_{iy} - S_{iy}\theta_{iy}. \tag{2.2.13}$$

Inserting eq. (2.2.13) into eq. (2.2.12) gives

$$H_{iy} = J_{iy}\alpha_{iy}. \tag{2.2.14}$$

Consider the case shown in figure (2.2.2) where more than one element is connected to node i. The out-of-balance moment is then

$$H_{iy} = T_{iy} - \sum_{m=1}^{n} \left(\frac{E_m I_{my}}{L_m} (4\theta_{iy} + 2\theta_{jy}) \right). \tag{2.2.15}$$

The summation regards the elements adjacent to node i, whereas node j refers to the other end of beam m. For a derivation of this relation the reader is referred to [1].

Where

E_m is the Young's modulus for the material of beam m,

L_m is the initial length of beam m connected to node i,

I_{my} is the second moment of area of beam m in bending around the y-axis,

n is the number of elements connected to node i,

θ_{iy}, θ_{jy} is the rotation in radians around the y-axis for node i and j respectively.

Inserting eq. (2.2.15) into eq. (2.2.14) yeilds

$$T_{iy} - \sum_{m=1}^{n} \left(\frac{E_m I_{my}}{L_m} (4\theta_{iy} + 2\theta_{jy}) \right) = J_{iy}\alpha_{iy}. \tag{2.2.16}$$

In analogy with the case of translational degrees of freedom the integration is carried out using a numerical time stepping method. First the angular acceleration is calculated from the out-of-balance moment

$$\alpha_{iy}^{(t+\Delta t)} = \frac{T_{iy} - \sum_{m=1}^{n} \left(\frac{E_m I_{my}}{L_m} (4\theta_{iy}^t + 2\theta_{jy}^t) \right)}{J_{iy}}. \tag{2.2.17}$$

From the rotational acceleration the rotational velocity can be obtained. Viscous damping is introduced as a scale factor C (usually set to ~ 0.95) multiplied with the velocity from the previous iteration,

$$\omega_{iy}^{(t+\Delta t)} = C\omega_{iy}^t + \alpha_{iy}^{(t+\Delta t)} \Delta t. \tag{2.2.18}$$

Finally the new rotation is calculated from

$$\theta_{iy}^{(t+\Delta t)} = \theta_{iy}^t + \omega_{iy}^{(t+\Delta t)} \Delta t. \tag{2.2.19}$$

2.2.3 Coupling between rotations and translations

In the two previous sections the implementation of translational and rotational degrees of freedom have been covered without considering the coupling between the two. In order to formulate the equations of a 12 DOF beam element solved by dynamic relaxation, the coupling effect has to be considered. In other words, the effect of moments for the translations and the effect of axial force for the rotations, has to be accounted for.

Recall equations eq. (2.2.5) and eq. (2.2.15) which describe the out-of-balance force and out-of-balance moments for the translations and the rotations. In order to introduce the coupling effect these equations have to be reformulated.

The axial force in a member is caused by elongation. Although in the previous formulation in eq. (2.2.5) elongation only accounts for the translations of the end nodes. However there is contribution caused by node rotations which makes the element bow. Consider the case with displacements along the x-axis and rotations around the y-axis, perpendicular to the x-axis. The axial force is then calculated as

$$R_{ix} = P_{ix} + \sum_{m=1}^{n} \frac{E_m A_m}{L_{mx}} (e_a + e_b) . \tag{2.2.20}$$

Here e_a and e_b are elongations due to axial deformation and bowing respectively from eq. (2.1.5) and eq. (2.1.7).

Similarly an axial force in a bent member will introduce moments at its two nodes. This is accounted for by replacing eq. (2.2.15) with the expression

$$H_{iy} = T_{iy} - \sum_{m=1}^{n} \left(\frac{E_m I_{my}}{L_m} (4\theta_{iz} + 2\theta_{jy}) + \frac{N_m L_m}{30} (4\theta_{iy} - \theta_{jy}) \right) . \tag{2.2.21}$$

The moment contribution from the axial force can be spotted in the second term including the axial force N_m. T_{iz} is the applied torque at node i around the z-axis, the sum regards the moment contribution from each beam element (from 1 to n) connected to node i, and j denotes the index for the other end of element m.

By comparison with the more conventional matrix formulation for beam elements, the different parts of eq. (2.2.20) can be recognized as elements in the stiffness matrix. For the derivation of the matrix formulation for geometrically non-linear beam elements the reader is referred to [14].

$$\boldsymbol{K}^e = \boldsymbol{K}_0^e + \boldsymbol{K}_\sigma^e . \tag{2.2.22}$$

In the matrix formulation below a $2D$ beam element with 6 DOF is considered, this is illustrated in Figure 2.2.3.

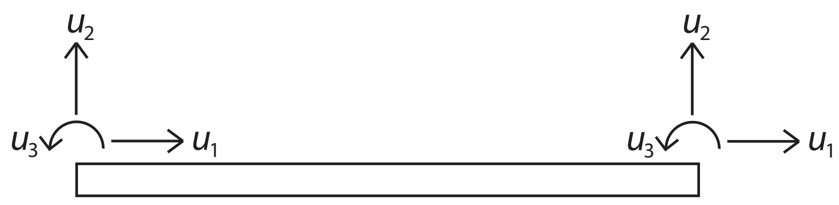


Figure 2.2.3: *2D beam elemnt with 6 DOF.*

Where $\boldsymbol{K}_0^e$ is the beam element stiffness matrix and $\boldsymbol{K}_\sigma^e$ is the change in stiffness due to the influence of an axial force Q_x.

Written in matrix form

$$\boldsymbol{K_0^e} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix}. \tag{2.2.23}$$

$$\boldsymbol{K_\sigma^e} = Q_x \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{6}{5L} & \frac{1}{10} & 0 & -\frac{6}{5L} & \frac{1}{10} \\ 0 & \frac{1}{10} & \frac{4L}{30} & 0 & -\frac{1}{10} & -\frac{L}{30} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{6}{5L} & -\frac{1}{10} & 0 & \frac{6}{5L} & -\frac{1}{10} \\ 0 & \frac{1}{10} & -\frac{L}{30} & 0 & -\frac{1}{10} & \frac{4L}{30} \end{bmatrix}. \tag{2.2.24}$$

The blue highlighted terms in matrices K_0^e and K_σ^e can be identified from eq. (2.2.21) as

$$H_{iy} = T_{iy} - \sum_{m=1}^{n} \left(\frac{E_m I_{my}}{L_m}(4\theta_{iz} + 2\theta_{jy}) + \frac{N_m L_m}{30}(4\theta_{iy} - \theta_{jy}) \right). \tag{2.2.25}$$

Written out for each of the four components in the matrix

$$\frac{4E_m I_{my}}{L_m} = \frac{4EI}{L}, \tag{2.2.26}$$

$$\frac{2E_m I_{my}}{L_m} = \frac{4EI}{L}, \tag{2.2.27}$$

$$\frac{4F_m L_m}{30} = \frac{4Q_x L}{30}, \tag{2.2.28}$$

$$-\frac{F_m L_m}{30} = -\frac{Q_x L}{30}. \tag{2.2.29}$$

Equally the green highlighted terms in matrix K_0^e can be identified from eq. (2.2.20) as

$$R_{ix} = P_{ix} + \sum_{m=1}^{n} \frac{E_m A_m}{L_{mx}}(e_a + e_b). \tag{2.2.30}$$

Written out for one of the four components (they are all the same)

$$\frac{E_m A_m}{L} = \frac{EA}{L}. \tag{2.2.31}$$

The matrix components regarding DOF:s 2 and 4 for the beam element in Figure 2.2.3 are not considered in the comparison since it was found rather complicated, thus outside of the scope for the thesis.

2.2.4 Algorithm for dynamic relaxation with beam elements

Following from the description of dynamic relaxation for translations and rotations together with the vector based beam theory, here is an attempt to give a holistic view of the solution process by putting the pieces together into one algorithm. Items 1 - 4 describe the setting up of the analysis model. Item 5 is the relaxation loop which is active until the solution has converged. Item (a) is a loop around each of the nodes in the structure where items (i) - (x) describe the procedure of how forces and moments are calculated causing the node to move towards their equilibrium position.

1. Create a center line geometry for the structure at hand.
2. Create nodes and sort the topology such that each element knows of its adjacent nodes and vice versa.
3. Introduce boundary conditions for the nodes.
4. Set element properties (i.e. cross sections, material and orientations).
5. While not converged:
   (a) For each node:
      i. Calculate mass and moment of inertia for the 6 DOF according to eq. (2.5.6) and eq. (2.5.10).
      ii. Set node forces and moments to zero.
      iii. Calculate force and moments based on the displacements and rotations from previous iteration by looping around each beam adjacent to the current node:
         - Calculate the member end rotations from eq. (2.1.2), eq. (2.1.3) and eq. (2.1.4).
         - Calculate the elongation due to axial displacement and bowing from eq. (2.1.5) and eq. (2.1.7).
         - Based on the new displacements calculate the axial force from eq. (2.1.12).
         - Equally from the rotations calculate the beam end moments, see eq. (2.1.10) and eq. (2.1.11).
         - Transform the axial force and the moments to global coordinates and apply to the node, according to eq. (2.1.23) and eq. (2.1.25).
      iv. Apply loading such as self weight and superimposed load cases on to the node.
      v. Move the node in the direction of the out-of-balance force:
         - Calculate the node acceleration from eq. (2.2.1).
         - Calculate the node velocity from eq. (2.2.8).
         - Update the new node position using eq. (2.2.9).
      vi. Calculate the node rotation angle based on the out-of-balance moment:
         - Calculate the angular acceleration of the node according to eq. (2.2.17).
         - Calculate the angular velocity of the node according to eq. (2.2.18).
         - Calculate the rotation angle according to eq. (2.2.19).
      vii. Based on the global x, y and z axes and the obtained rotation angles, calculate the nonlinear rotation vector from eq. (2.1.27).
      viii. Calculate the transformation matrix from eq. (2.1.28).
      ix. Rotate the node by transforming its coordinate system:
         - Calculate the result of the three rotations using eq. (2.1.27).
         - Calculate the transformation matrix based on eq. (2.1.28).
         - Execute the rotation by transforming the coordinate system from eq. (2.1.34).
      x. Equally transform the end coordinate systems for the adjacent beam ends.
   (b) Summarize the out-of-balance forces for all the unconstrained nodes and compare with the convergence criteria. If the force is smaller than the convergence limit, the analysis is done and the while loop condition is set to false.

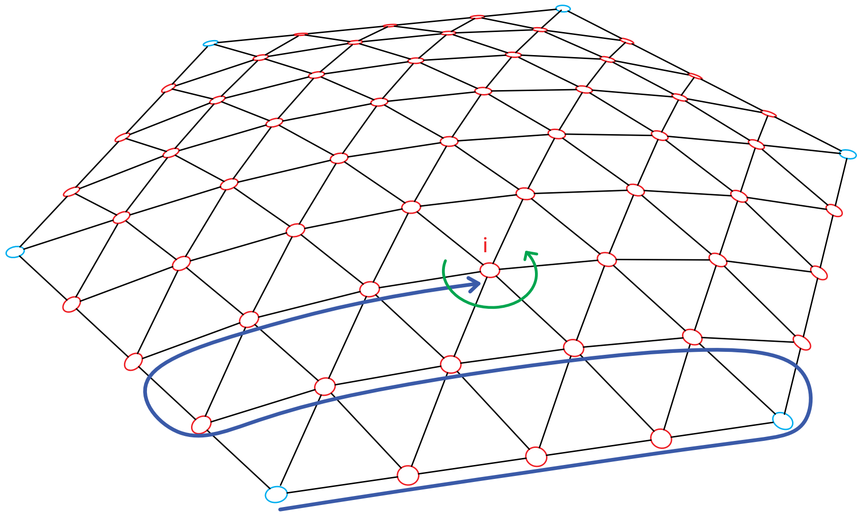


Figure 2.2.4: *Principle sketch for loop logic presented in the algorithm above. The blue arrow represents the loop around each of the nodes in item (a), and the green arrow represents the loop around each element adjacent to node i described in (iii) in the algorithm.*

2.3 Form finding

There are essentially two inherently different approaches to form finding of structures; elastic control and force control. Both methods aim to find the equilibrium geometry for a structure under a certain load and/or prestress, but the way the forces are calculated in the elements differs. A tensile structure is usually form found with the pretension in the membrane or cables as the governing form driver and the effect of self weight and loading is neglected. In the analysis of the form found structure when loading is accounted for the prestress is still the main source of stress in the structure, hence chosen as the driver in the form finding process. For a compression structure on the other hand, loading rather than prestress, is the main source of stress in the structure and is used as the governing form driver.

Form finding of compression structures

Elastic control form finding is basically non-linear structural analysis with the purpose of finding the static equilibrium geometry for a structure under a certain load. The classic approach for a compressive structure would be to convert all the bar and/or beam elements to cables and invert the load (see Figure 2.3.1) such that all elements work in pure tension. The initial geometry together with the section and material properties of the cables will dictate the resulting equilibrium geometry.

This process can be formulated as a system of equilibrium equations set up and solved using a matrix formulation. However, the analysis with cable elements is heavily non-linear, meaning that the matrix solution has to be carried out iteratively. Another approach to find equilibrium is to use dynamic relaxation which by its iterative nature inherently accounts for geometric non-linearity. This is the method adopted in this thesis. Dynamic relaxation is based on the differential equation for damped motion which describes a relation between acceleration, velocity and distance. The sudden application of load on the initially resting nodes put them out of equilibrium. Using dynamic relaxation the acceleration is calculated based on Newton's second law in eq. (2.2.1) where f is the out-of-balance force. By stepping through time the velocity and distance to move the nodes is then obtained. As the nodes move, the bars connected to the nodes start to stretch introducing forces pushing and pulling at the nodes. Each node is moved in the direction of its out-of-balance force in an iterative process until the applied force from the loading is compensated by bar forces, meaning that the node has found equilibrium. The mathematics behind the process is presented more thoroughly in chapter *2.2 Dynamic relaxation*.

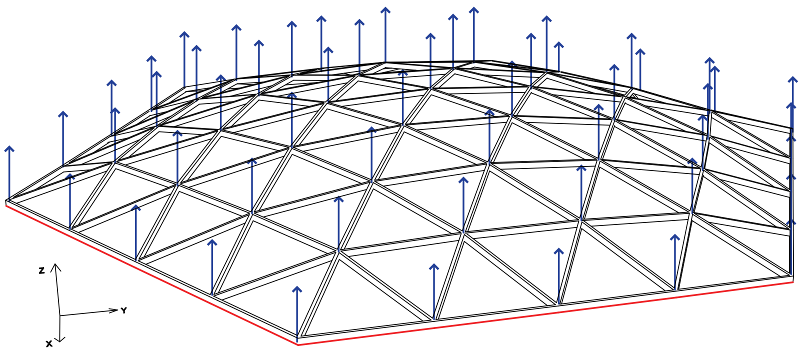


Figure 2.3.1: *Reversed loading (the blue arrows) applied during form finding of a grid shell structure. The elements are set to cables and a pure tension structure is obtained.*

Form finding of tensile structures

Force control form finding is used for tensile structures and uses prestress rather than loading as the governing form driver. The prestress in the membrane and/or cable elements is preset and these forces are summed on the nodes giving the out-of-balance force. The nodes are moved in the direction of the out-of-balance force using the equation of motion solved with dynamic relaxation. As the nodes move, the bars connected to the nodes start to stretch, but instead of calculating the force in the bars based on deformations, the force is preset and kept constant throughout the whole process. For a tensile structure the only driver of the form is this prestress, no external loads are applied. Fabric membranes usually have different stiffness in the warp and weft direction based on the orthotropic properties of the weave. Therefore the prestress can be set differently for the two directions and the relationship between these two prestresses is what dictates the resulting equilibrium geometry. In practice this means that the resulting form is independent of the magnitude of the prestress but rather dependent on the ratio between the prestress in the two directions. The process is also independent of material and section properties for the elements.

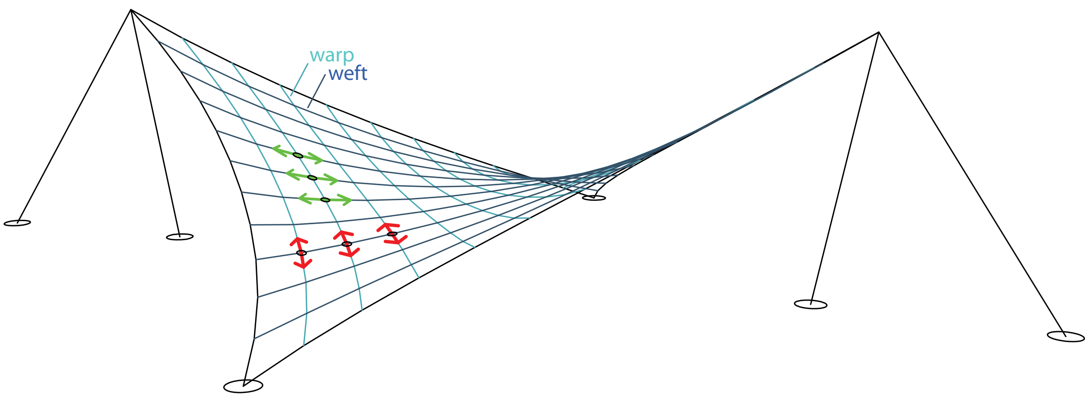


Figure 2.3.2: *Prestress in the weft and warp direction pulling at the nodes. It is the ratio between these stresses that dictates the result of the force control form finding.*

2.4 Beams in form finding

In this thesis form finding with bar, cable and beam elements has been implemented partly based on the traditional, pure tension hanging chain model approach, and partly in a more experimental way, influenced by the force control technic used for tensile structures. The aim has been to develop a method to conduct form finding which allows for a trade-off between initial geometry and fully form found geometry using elements with bending capability.

Three different approaches to form finding with beams are presented below, each of which is based on a reversed load case as the governing form driver where the aim is to find the equilibrium geometry of the structure.

2.4.1 Property control form finding

Property control form finding is implemented as a method of controlling axial and bending stiffness for the elements in a structure during the form finding process. Scale factors for axial (EA) and bending (EI) stiffness is exposed in the user interface as slider bars, allowing for the user to elaborate with the form. By dropping the bending stiffness to zero the beams are effectively converted into bar elements and the form adapts to find equilibrium without bending. By increasing the bending stiffness again the form adapts to work partly in bending and partly in membrane action where this ratio is controlled by the user. The different form options can be saved and evaluated using automatic sizing. With this approach each element in the structure is effected equally by the property scaling, whether or not it is heavily utilized and regardless if it is working in axial or bending or action.

2.4.2 Force/moment control form finding

This type of form finding is inspired by the method used for tensile structures. Instead of calculating the axial force and the moments based on translations and rotations, these values are set based on the capacity of the elements. The user controls the force in the elements with a slider bar by specifying the value as a percentage of the axial capacity. The bending moments are controlled in a similar fashion, based on the percentage of the bending capacity. In contrast to force control used for tensile structures the governing form driver is a load case. Thus it becomes a tricky balance act to ensure that the forces set in the elements can balance the applied loading. If the load is large compared to the element force, equilibrium might never be found. This is because the force does not change with the elongation of the members, it is just the direction of the force that changes when the geometry changes.

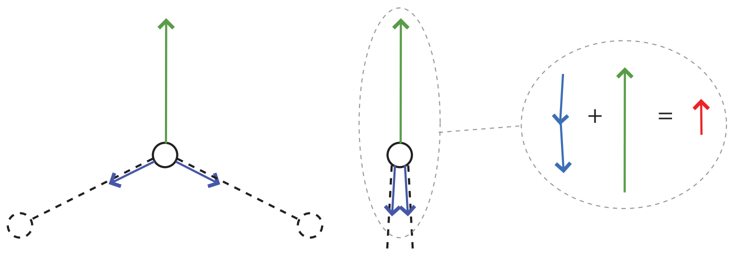


Figure 2.4.1: *Load (green) applied on a node and the specified forces (blue) in the adjacent elements. Even if the geometric configuration of the elements changes, equilibrium might not be possible for this relation between load and force.*

If was found a bit unintuitive to set both the x and y-moments of an elements to the same value, preferably these would be controlled separately. But because of the difficulty to balance the forces this method was not found to be particularly useful. However one great benefit with this method is that it allows the elements to more easily change from their initial length, since the change of length does not affect the axial force. This was found to be a problem with other methods, causing the structure to wrinkle in an undesirable way. The investigation however led to the implementation of limit control form finding presented in the next section.

2.4.3 Limit control form finding

Limit control form finding has similarities to the two methods presented above. It allows for elaboration with the ratio of bending and membrane action like in property control, but rather than scaling the EA and EI values for the elements the user specifies limit values for the axial and bending utilization in a way more similar to setting the values in force/moment control. These two limits are percentages of the member capacity, meaning that two beam elements with different cross sections but the same bending utilization limit are allowed to reach different maximum moments. If a member is loaded such that the axial force or the moment exceeds the specified limits the exceeded quantities are scaled down and the form will have to adapt. This approach makes it possible to enhance the structural performance in the most critical parts of a structure, where for example the bending action is dominating, without necessarily changing the parts that are not as critical.

To demonstrate how this works, consider the arc like structure in Figure 2.4.2. The load is reversed to act as a form finding load and the elements have circular hollow sections with a radius of 0.2 m.

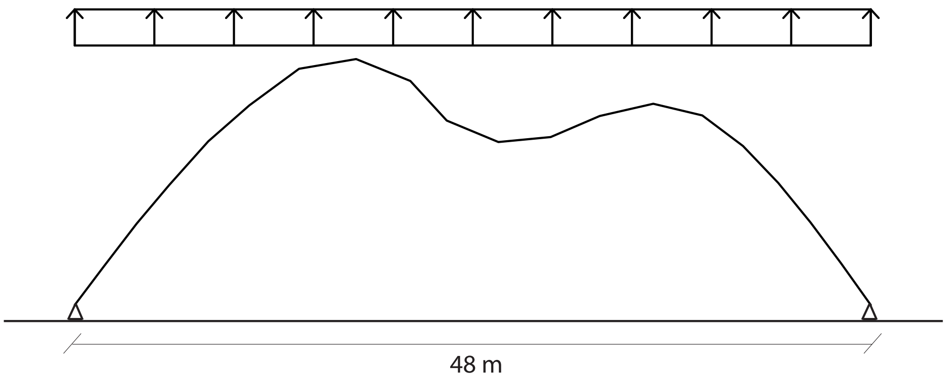


Figure 2.4.2: *Boundary condition, loading and initial center line geometry for a bent arc like structure.*

First of all elastic analysis is carried out showing that the mid part of the structure is far over utilized (see Figure 2.4.4). Secondly limit control form finding is carried out with the axial capacity limit kept constant at 50 % and the bending capacity varied from 75% - 0% resulting in 4 different geometries.

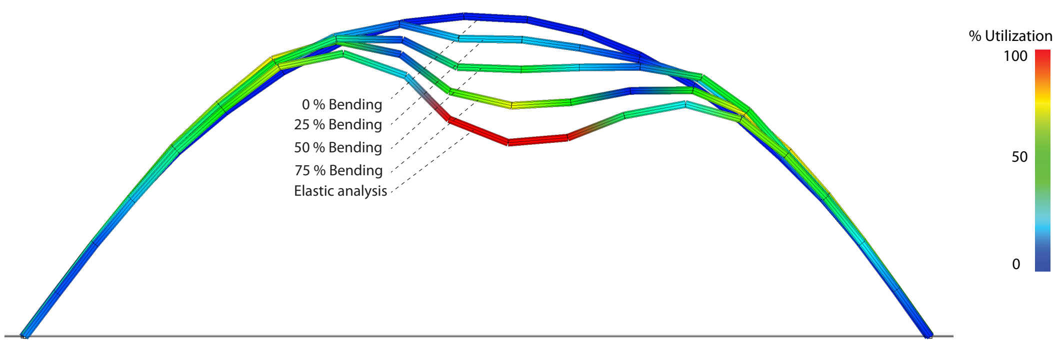


Figure 2.4.3: *Equilibrium geometry as a result of elastic analysis and 4 cases of limit control form finding with various settings. The colors display the utilization of the members where red is 100 % and blue is 0 %*

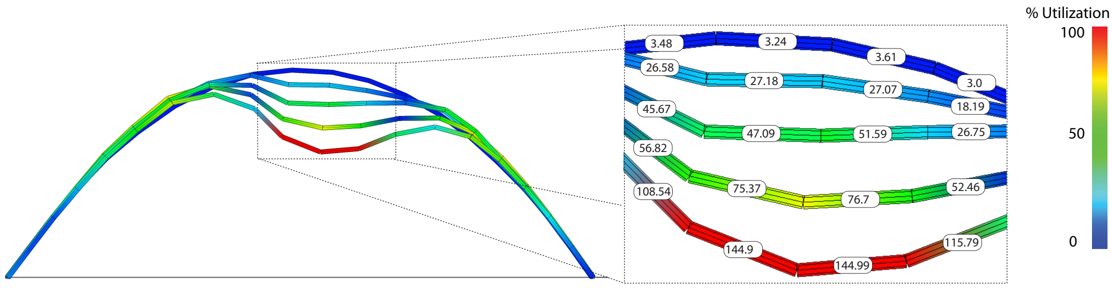


Figure 2.4.4: *Utilization values for the critical part of the structure.*

The initial structure in the example is due to its form predominantly working in bending which becomes clear by studying the values in the utilization plot in figure (2.4.4). The most critically stressed element in the case with 75 % as bending limit is utilizing 76.7 % of its total capacity, implying a 1.7 % utilization in axial action. Since the axial utilization is not significant in the example the limit is kept constant at 50 %. When the bending limit is dropped to 0 % the structure takes the form of a perfect cantenary and by observing the utilization values, now ranging from about 3.0 - 5.2 %, a remarkable difference structural efficiency is clear. On the other hand the form has undergone a drastic change under the form finding process potentially hindering other design parameters.

High level algorithm for the limit control form finding:

1. The dynamic relaxation process runs one iteration, calculating force F and moments M in the elements.
2. The axial and bending utilization U_F and U_M are calculated respectively.
3. The utilization limits are sent across from the UI i.e. If the limits are 40% axial and 60% bending, then $U_F^{40\%}$, and $U_M^{60\%}$ are calculated.
4. If $|U_F| > U_F^{40\%}$, then set the force such that $|U_F| = U_F^{40\%}$ is obtained.
5. A similar check is made for the moments, if $|U_M| > U_M^{60\%}$, then the moments M_x and M_y (giving U_M), for bending around the x, and y axis respectively, are both scaled equally such that $|U_M| = U_M^{60\%}$.
6. The node positions and rotations are now updated according to the calculated and scaled forces and moments.

Based on the levels of utilization the form adapts to find its equilibrium position for the applied load. This process of form finding allows for an interesting trade off between bending and membrane action.

2.5 Speed and Convergence

To achieve the objective of making SMART Form a real time and interactive analysis tool, speed is essential. The aim is to reduce the time it takes to reach convergence while making sure the analysis is stable. The obvious approach to increase speed of a dynamic relaxation solution would be to increase the time step. A greater time step means that each node is moved a greater distance for each iteration, resulting in lesser iterations needed to reach the equilibrium position, improving speed, assuming that instability does not occur. Two other parameters that dictate speed is the mass and moment of inertia used in eq. (2.2.7) and eq. (2.2.17). Let's first consider the mass and recall the use of Newton's second law which states

$$F = ma. \tag{2.5.1}$$

Which is used in dynamic relaxation to calculate the acceleration from eq. (2.2.7), in a simplified matter this can be written

$$a = \frac{F}{m}. \tag{2.5.2}$$

For a certain out-of-balance force the magnitude of the acceleration is completely dictated by the choice of the mass, a smaller mass results in a greater acceleration. If the aim is to trace the real dynamic behavior of the structure, the masses would be lumped based on the real structural weight [3]. In the case of static analysis however, the interest is solely to reach convergence as quick as possible, and fictitious masses calibrated for this purpose can be used. In the paper [3] by Michael Barnes, a derivation is presented for the relation between time step, mass and stiffness for a node in a cable net structure. Barnes proposes to keep the time step as a global constant and calibrating individual masses for the nodes based on their local stiffness. This approach is applied in the thesis, but rather than calculating one mass per node, one mass for each degree of freedom is calculated and formulated here as a mass matrix.

The relation between time step, mass and stiffness that will ensure a stable solution is formulated as [3],

$$\Delta t = \sqrt{\frac{2m}{S}}. \tag{2.5.3}$$

For a constant Δt the mass is calculated as

$$m = \frac{\Delta t^2 S}{2}. \tag{2.5.4}$$

Where the mass matrix $\boldsymbol{M}_i$ of node i can be expressed as a scalar multiplied with the node stiffness matrix

$$\boldsymbol{M}_i = \frac{\Delta t^2}{2} \boldsymbol{S}_i. \tag{2.5.5}$$

Written on matrix form

$$\boldsymbol{M}_i = \frac{\Delta t^2}{2} \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix}. \tag{2.5.6}$$

The components in the stiffness matrix S are calculated from

$$S_j = \sum_{m=1}^{n} \left(\frac{E_m A_m}{L_m} |\boldsymbol{p}_m \cdot \boldsymbol{c}_j| \right). \tag{2.5.7}$$

Here j takes the form x, y, z, the summation over m regards the elements 1 to n adjacent to node i, $\boldsymbol{c}$ is the unit vector taking the form of the global coordinate axes $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ and $\boldsymbol{p}_m$ is the vector between the end nodes of element m.

Equally a relation between time step, moment of inertia and rotational stiffness that ensures speed and stability is needed for the rotational degrees of freedom in eq. (2.2.17). In Appendix A an attempt to derive that relationship based on Barnes approach for the translational degrees of freedom is presented. Let us consider the relation between time step, moment of inertia and rotational stiffness as

$$\Delta t = \sqrt{\frac{2J}{S}}. \tag{2.5.8}$$

Where

S is the rotational stiffness,

J is the moment of inertia,

Δt is the time step.

In analogy with the masses eq. (A.1.14) is used to compute the components of the moment of inertia matrix $\boldsymbol{J}_i$ for node i as

$$\boldsymbol{J}_i = \frac{\Delta t^2}{2} \boldsymbol{S}_i. \tag{2.5.9}$$

On matrix form

$$\boldsymbol{J}_i = \frac{\Delta t^2}{2} \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix}. \tag{2.5.10}$$

Where the components in the stiffness matrix S are calculated from

$$S_j = \sum_{m=1}^{n} \left(\frac{1}{L} (E_m I_{mx} \boldsymbol{X}_j + E_m I_{my} \boldsymbol{Y}_j + G_m K_{vm} \boldsymbol{Z}_j) \right). \tag{2.5.11}$$

Here j takes the form x, y, z; the summation over m regards the elements 1 to n adjacent to node i; $\boldsymbol{X}_j$, $\boldsymbol{Y}_j$, $\boldsymbol{Z}_j$ are the normalized local coordinate axes and their component in j-direction; I_x, I_y are the second moment of area around local x, y axes respectively and K_v is the torsional stiffness.

Convergence criterion

The result form numerical calculations applied to a scientific problem will usually be an approximation. In order to formulate the convergence criterion it is therefor of interest to find a value where the approximation gives an error that is acceptably small. The level of acceptance may vary depending on the purpose of the analysis. In the implementation of convergence criteria in SMART Form, one criterion is used for the sizing whilst another one is used for the general analysis. To improve the speed of the automatic sizing, the sizing criterion is allowed to be more easily reached. This is because the static solution has to converge before the sizing is executed, and around 5-10 sizing iterations has to be performed for convergence.

Time stepping

As shown in the theory in section *2.2 Dynamic relaxation* the time stepping algorithm implemented for the analysis is the Euler Forward method. This is a general method that can be applied to solve ordinary differential equations using the formula

$$y_{n+1} = y_n + hf(x_n, y_n) \tag{2.5.12}$$

The method uses information from the beginning of the interval only, to increment the solution. It is not considered as being very accurate nor very stable in comparison to more involved methods, but has the advantage of being simple to implement [30].

2.6 Size optimization

The theory of two different approaches to automatic member cross section sizing is introduced in this chapter. The choice of method depends on what parameters are being optimized for. From a structural point of view there are two limit states that has to be satisfied in order for a structure to be classified as safe; the ultimate limit state and the serviceability state. A structure should be designed such that the probability of reaching one of these limit states, at which it would become unfit for its intended use, is acceptably low [18]. A stress based approach is used when sizing for ultimate limit state, essentially making sure that no members in a structure are over/under-utilized due to stress. Sizing for serviceability state on the other hand is about minimizing deflection at a critical point in the structure, and a virtual work based method is more suitable in this case. The stress based sizing is implemented during the thesis, while virtual work based sizing was found less useful for conceptual analysis.

2.6.1 Stress based sizing

The ultimate limit state concerns strength and stability requirements of a structure and can be seen as a collapse criterion. None of the members in the structure are allowed to be utilized over its capacity causing the risk of a collapse. To perform sizing for ultimate limit state, each member utilization is checked and the cross sections are scaled such that the sought utilization is reached [18].

A high level algorithm used for stress based size optimization.

1. While (any element is over/under-utilized, i.e. sizing not completed)

   Running the analysis:

   (a) While (not converged, i.e. analysis not completed)
   (b) Run the dynamic relaxation one iteration.
   (c) Check the out-of-balance force for convergence.

2. Calculate the utilization of the elements due to axial force and bending moments.
3. Scale the beam to reach sought utilization.
4. Update section based properties like mass and stiffness.

2.6.2 Virtual work based sizing

The serviceability limit state concerns a deflection requirement for a structure. If a structure fails to meet this criterion it might still be strong enough to stand up but it will be unfit for its intended use, for example in terms of durability or fire resistance. Automatic sizing, to fulfill a deflection criterion, starts by identifying a critical degree of freedom (DOF) for deflection. A unit load is applied in that DOF and a virtual work based method is used to calculate which elements that contributes to stiffness. These elements are sized until the deflection criterion is met.

Below is a high level algorithm used for virtual work based size optimization. The main differences to the stress based sizing is the convergence criterion and the approach used to find which elements to size.

1. While (deflection criteria is not fulfilled, i.e. sizing not completed)

   Running the analysis:

   (a) While (not converged, i.e. analysis not completed)
   (b) Run the dynamic relaxation one iteration.

(c) Check the out-of-balance force for convergence.

2. Apply a unit load in the degree of freedom for the critical deflection.
3. Find each element's contribution to stiffness in the critical DOF.
4. Scale the element sections in relation to their stiffness contribution.
5. Update section based properties like mass and stiffness.

2.6.3 Stress based utilization

The stress based utilization of a member is calculated according to Eurocode 3 specifications for uniform members in bending and axial compression (Ch. 6.3.3). It states that members subjected to bending and axial compression should satisfy [13]

$$\frac{N_{Ed}}{\chi_x \frac{N_{Rk}}{\gamma_{M1}}} + k_{xx}\frac{M_{x,Ed}+\Delta M_{x,Ed}}{\chi_{LT}\frac{M_{x,Rk}}{\gamma_{M1}}} + k_{xy}\frac{M_{y,Ed}+\Delta M_{y,Ed}}{\frac{M_{y,Rk}}{\gamma_{M1}}} \leq 1, \tag{2.6.1}$$

$$\frac{N_{Ed}}{\chi_y \frac{N_{Rk}}{\gamma_{M1}}} + k_{yx}\frac{M_{x,Ed}+\Delta M_{x,Ed}}{\chi_{LT}\frac{M_{x,Rk}}{\gamma_{M1}}} + k_{yy}\frac{M_{y,Ed}+\Delta M_{y,Ed}}{\frac{M_{y,Rk}}{\gamma_{M1}}} \leq 1. \tag{2.6.2}$$

Here

$N_{Ed}, M_{x,Ed}, M_{y,Ed}$ are the design values for the compression force and maximum moments about the x-x and y-y along the member respectively.

$\Delta M_{x,Ed}, \Delta M_{y,Ed}$ are the moment due to the shift of the centrodial axis according to [13] (Ch. 6.2.9.3).

χ_x, χ_y are the reduction factors due to flexural buckling [13] (Ch. 6.3.1).

χ_{LT} is the reduction factor due to lateral torsional buckling [13] (Ch. 6.3.2).

$k_{xx}, k_{xy}, k_{yx}, k_{yy}$ are the interaction factors.

In the implementation in SMART Form the two equations above are simplified, partly to not over complicate the otherwise lightweight and conceptual analysis and partly to enable a separation of axial and bending utilization.

First and foremost SMART Form is limited to beam elements with closed cross sections. This limitation simplifies the calculations of torsional capacity and means that lateral torsional buckling can be neglected [13] (s. 3-48). The increased loading effect due to a shift in the centrodial axis for members with closed sections has a minor impact and is therefor also neglected. Furthermore, to simplify the separation of bending and axial utilization the interaction factors k_{xx}, k_{xy}, k_{yx}, k_{yy} are set to 1. This is a conservative assumption effectively reducing the capacity for a member. The material based reduction factor γ_{M1} is set to 1 which is the correct value for steel. Since the material models implemented for timber and concrete are anyway very crude (and should not be relied on) the effect of γ_{M1} was chosen to be neglected.

The obtained simplified relation is

$$\frac{N_{Ed}}{\chi_x N_{Rk}} + \frac{M_{x,Ed}}{M_{x,Rk}} + \frac{M_{y,Ed}}{M_{y,Rk}} \leq 1, \tag{2.6.3}$$

$$\frac{N_{Ed}}{\chi_y N_{Rk}} + \frac{M_{x,Ed}}{M_{x,Rk}} + \frac{M_{y,Ed}}{M_{y,Rk}} \leq 1. \tag{2.6.4}$$

For the calculations of N_{Rk}, $M_{x,Ed}$, $M_{y,Ed}$ the reader is referred to *Eurocode 3: Design of steel structures* [13]. Extra attention is given to the reduction factor for buckling, χ, which is treated more in detail in the next section.

2.6.4 Buckling

The axial capacity for a member is reduced in eq. (2.6.3) and eq. (2.6.4) with the reduction factors χ_x and χ_y, which are calculated based on the methodology presented in Eurocode 3 Chapter 6.3. The method and the assumptions made in the implementation to SMART Form can be described briefly from equation

$$\chi = \frac{1}{\phi + \sqrt{\phi^2 - \bar{\lambda}^2}}, \tag{2.6.5}$$

where

$$\phi = 0.5\left[1 + \alpha(\bar{\lambda} - 0.2)\right], \tag{2.6.6}$$

$$\bar{\lambda} = \sqrt{\frac{Af_y}{P_c}}, \tag{2.6.7}$$

and

$$\alpha = \text{an imperfection factor}, \tag{2.6.8}$$

$$P_c = \text{the buckling force according to Euler.} \tag{2.6.9}$$

Here the imperfection factor α is set to 0.49 which is a conservative assumption. The non-dimensional slenderness $\bar{\lambda}$ is calculated based on the whole section area which is a valid approach for cross sections in class 2 and 3.

One of the most significant parameters in the capacity calculations is the effective length. Doubling the effective length of a member effectively reduces its axial capacity to one forth. The relationship between the effective length scale factor k and the axial capacity for a member according to Euler is

$$P_c = \frac{\pi^2 EI}{(kL)^2}. \tag{2.6.10}$$

Here P_c is the critical axial force, E is the Young's modulus, I is the second moment of area of the member, k is the effective length scale factor and L is the length of the member.

The challenge with the implementation of utilization calculations in SMART Form was the estimation of effective length of the elements. The tool is aimed to work for (more or less) an arbitrary geometry so it was brought into discussion whether or not it was possible to automatically calculate the correct effective length of the elements. The conclusion was to expose the effective length scale factor k for the user to specify the value. Since the elements in SMART Form can be put into different layers and the value of k is set per layer, the user can specify the effective length with the accuracy needed for the problem at hand.

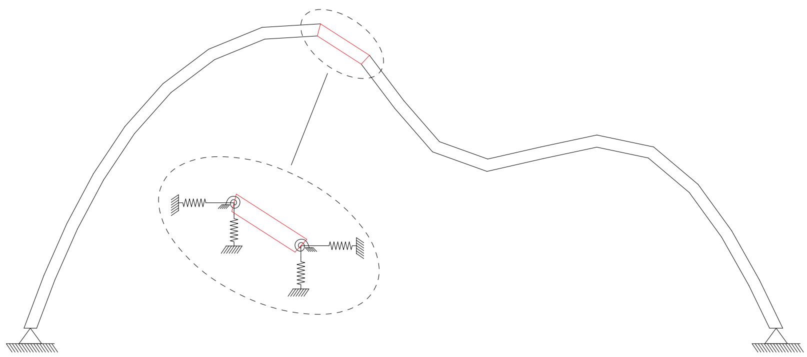


Figure 2.6.1: *Arc type of structure divided into finite elements.*

In order to calculate the proper effective length of an element in an arbitrary structure consider the red colored element in the 2D case in 2.6.1. The parts of the structure to the left and to the right of the element can be substituted with springs of equivalent stiffness using static condensation [14]. When the spring stiffness is determined an effective length could potentially be extrapolated from the Euler buckling cases. This method however would require heavy computationally power, hence would not be suitable for the real time analysis in SMART Form [19].

2.6.5 Heuristic approach to global buckling

In order to validate the results from the form finding and the assumptions made for the sizing (such as effective length and sought utilization level), a global buckling check is performed. This is implemented in SMART Form as a load scale factor slider bar. The idea is that the user form finds and sizes the structure under a so called form finding load case. Then applies a potentially different load case, for example an asymmetrical loading to analyze the sensitivity to elastic global buckling. The scale factor is slowly ramped up until the structure fails. An indication that buckling is progressing is when the convergence path suddenly changes direction which is illustrated in Figure 2.6.3.

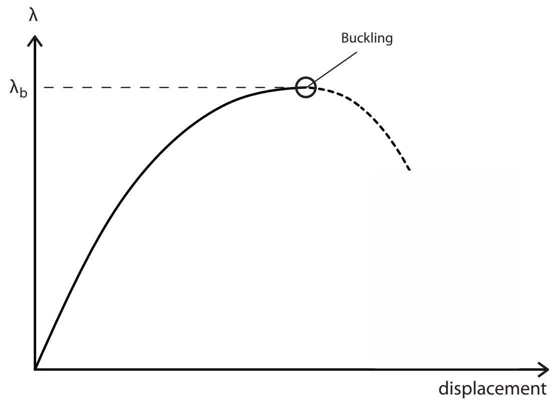


Figure 2.6.2: *Relation between load scale factor λ and displacement. When buckling occurs the structure can't take more loading.*

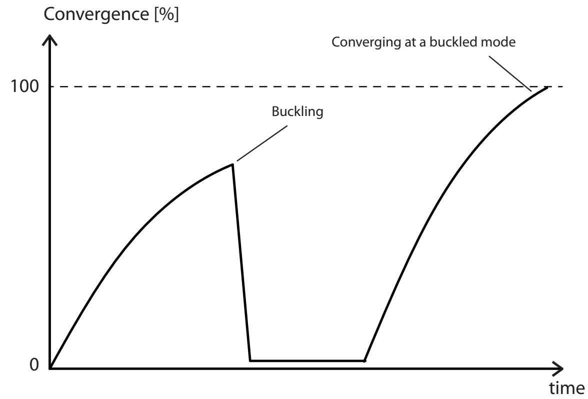


Figure 2.6.3: *Illustrative graph of the progress of the dynamic relaxation solution when buckling suddenly occurs before convergence is reached (which can be seen at the peak on the left hand side in the graph). The solution finally converges in a most likely grossly deformed state.*

2.6.6 Sizing sections

When the utilization of a member is obtained from eq. (2.6.3) and eq. (2.6.4) the member is sized to not over/under-utilize the material due to stress induced from loading. For this purpose a cross section scale factor is calculated and the section is scaled uniformly. Two different approaches to perform sizing were found useful.

Incremental sizing

The first approach is called incremental sizing and means that a member is scaled to increase its capacity incrementally, 5% up or down independent of how much it is over/under-utilized. This approach was found fairly stable and working for most types of structures. The downside with the method is speed, it is quite slow especially if the sections are much over/under-utilized to begin with. On top of that the solution will finally converge on members having approximately 5% more capacity than intended.

Direct sizing

The second approach is called direct sizing and means that the cross section is directly scaled to meet the exact sought level of utilization. This type of sizing was found quick but sometimes causing instability problems. It seemed inappropriate for space truss type of structures where the forces easily redistribute to different paths, seemingly happening when suddenly drastically changing the section sizes. However this method is quick and works very well for gridshell type of structures, where the force distribution is less variable.

The relationship between utilization and section size in eq. (2.6.3) and eq. (2.6.4) is quite involved so the scale factor is calculated iteratively based on the assumption that the relation is close to

$$c = \sqrt[3]{1 + (u_{\text{current}} - u_{\text{sought}})}. \tag{2.6.11}$$

Where c is the cross section scale factor, u_{current} is the utilization based on the current forces and moments, and u_{sought} is the utilization level aimed to reach. A recursive function was used to perform an iterative search for the right section scale factor based on eq. (2.6.11). It was found that less than four iterations usually results in a section size, giving a utilization less than 1% from the sought value. This process can be formulated in an algorithm as

For each beam in the structure:

1. Get sought utilization level from the user interface.
2. Calculate current utilization.
3. Calculate a value for the scale factor c based on eq. (2.6.11).
4. Recursive loop with condition; if($|(u_{\text{current}} - u_{\text{sought}})| > 0.01$)
   (a) Increment size with the c scale factor.
   (b) Calculate section properties.
   (c) Calculate the current utilization u_{current} based on present forces and moments.
   (d) From eq. (2.6.11) calculate the new scale factor c.
   (e) Go back to 3. and perform the check again. If true continue, else break.

The whole section is then scaled uniformly by multiplying the width, height and thickness by c. The types of sections implemented in SMART Form during the thesis is shown in Figure 2.6.4 and Figure 2.6.5.

Sections

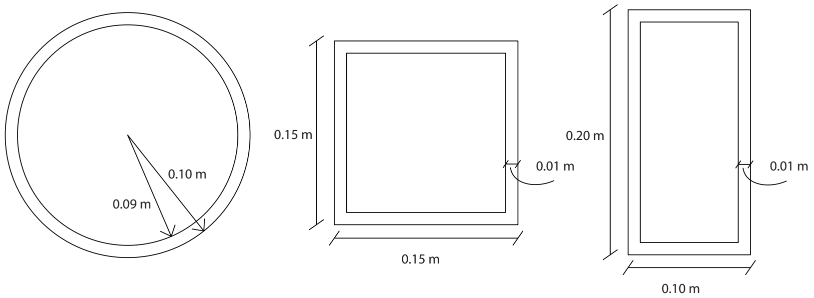


Figure 2.6.4: *Types of default hollow sections implemented in SMART Form for the thesis.*

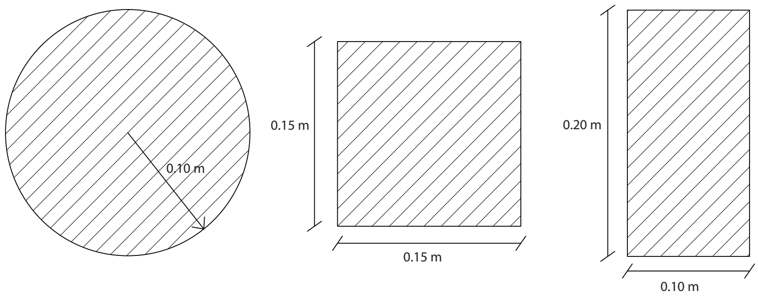


Figure 2.6.5: *Types of default solid sections implemented in SMART Form for the thesis.*

2.7 Load

2.7.1 Distributed load

The loads on the beam elements are applied as either point loads at the nodes or as distributed line loads. In the later case the load has to be reformulated into forces and moments applied to the ends. This reformulation is based on the theory in [14] (Ch.4).

Here follows a derivation of moment and section force distribution for a beam element with clamped ends, length L, and is loaded with an uniformly distributed load q_y. The particular solution to the beam differential equation is sought in order to determine the end shear forces $V_p(0)$, $V_p(L)$, and the end moments $M_p(0)$, $M_p(L)$.

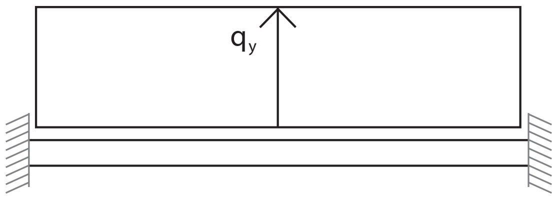


Figure 2.7.1: *Loading and boundary conditions for a beam with two clamped ends.*

Starting from the differential equation of the beam element

$$EI\frac{d^4w_p}{dx^4} - q_y = 0. \tag{2.7.1}$$

$$\tag{2.7.2}$$

The clamped beam has solution

$$w_p(x) = \frac{q_y}{EI}\left(\frac{x^4}{24} - \frac{Lx^3}{12} + \frac{L^2x^2}{24}\right). \tag{2.7.3}$$

From the relationship between shear force V, moments M and deflection in Euler beam theory

$$M_p(x) = EI\frac{d^2w_p}{dx^2}, \tag{2.7.4}$$

$$V_p(x) = -EI\frac{d^3w_p}{dx^3}. \tag{2.7.5}$$

Making use of eq. (2.7.3) together with eq. (2.7.4) and eq. (2.7.5) yeilds

$$M_p(x) = q_y\left(\frac{x^2}{2} - \frac{Lx}{2} + \frac{L^2}{12}\right), \tag{2.7.6}$$

$$V_p(x) = -q_y\left(x - \frac{L}{2}\right), \tag{2.7.7}$$

which is the moment and shear force distribution for the element, caused by the load q_y. In order to translate the distributed load into forces and moments at the end nodes eq. (2.7.6) and eq. (2.7.7) are applied for $x = 0$ and $x = L$ and the resulting forces and moments are locally added to the element.

Global distributed loads are divided into components based on the orientation of the local coordinate system. Meaning that moments and forces from eq. (2.7.6) and eq. (2.7.7) have to be calculated for each of the local coordinate axes, x and y.

2.7.2 Surface load

In order to apply loading to a structure quick and easy, a concept called surface loading was invented. The idea is that any surface selected, together with the center line geometry of the elements and the boundary condition points, is interpreted as a load surface. The load direction is dictated with the surface normal and the load magnitude is controlled with a slider bar on the user interface. A specific load surface only effects the nodes within its reach. Meaning that if a node projected along the surface normal ends up on the surface, it is within the reach, if the projection on the other hand fails it is not.

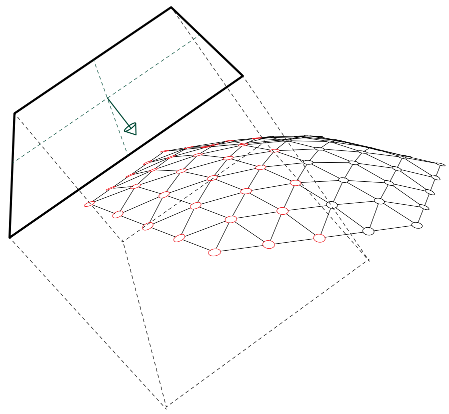


Figure 2.7.2: *Direction of the load surface and the nodes within its reach colored red.*

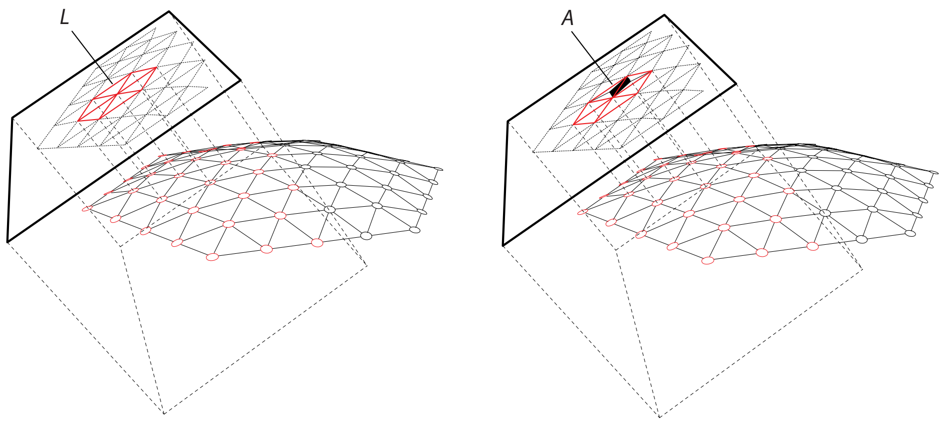


Figure 2.7.3: *Projected elements on the surface where the length or area is calculated if the loading is applied as line or area load.*

The load from the surface can be applied as point, line or area load. When the point load option is chosen the value from the slider bar on the user interface is interpreted as force (N), and applied as point loads on every node within the surface reach. For a line load on the other hand the value from the slider bar is interpreted as a N/m, which is then multiplied with the projection length of the element on the load surface, to give the distributed load. Equally for an area load the slider bar value is set to N/m^2 and the projected area is calculated and translated to an equivalent line load. The direction of the load is specified with the surface normal. The load surface can be rotated and moved in real time and the effected nodes, elements, projections and loads is updated. Line and area loads are applied on the elements as forces and moments based on the theory presented in the previous section.

2.8 Elements

The main focus in the thesis has been the implementation of beam elements. However bars, cables and another type of bespoke form finding element have been implemented to allow analysis of more interesting structural configurations. One of the great advantages of dynamic relaxation for conceptual analysis is the decoupling between equilibrium and compatibility equations, meaning that the velocity and movement of the nodes in eq. (2.2.8) and eq. (2.2.9) are evaluated for all the nodes before the residuals are calculated [3]. That makes it possible to incorporate complex material properties, on-off non-linearities, releases and interactive controls.

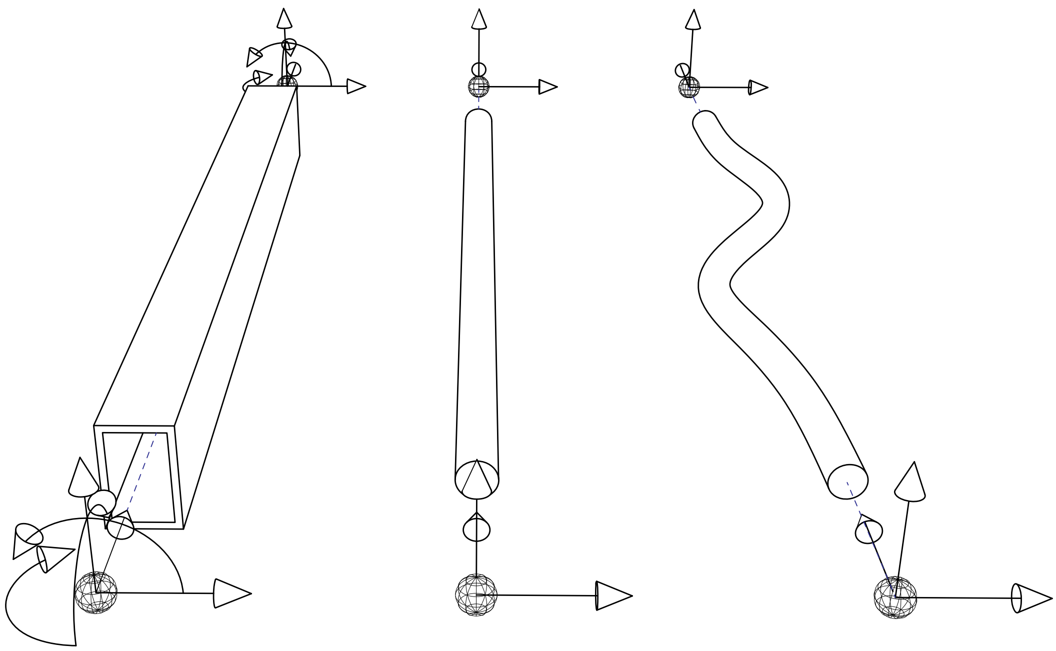


Figure 2.8.1: *Three element types implemented in SMART Form shown as conceptual sketches. From the left; a 12 DOF beam element, a 6 DOF bar element and a 6 DOF cable element.*

From a computational point of view the bar and the cable can be seen as simplifications of the beam element where the rotational DOF:s are neglected. A bar however takes both compression and tension whilst a cable only takes tension.

During the exploration of form finding methods it was found that the use of cable elements (pure tension) rather than bar elements (compression or tension), often resulted in smoother and more applicable forms. In the beam based form finding presented in chapter *2.4 Beams in form finding* it is shown how to conduct form finding while trading between bending and axial utilization, essentially blending between bars and beam elements. With the smoothness problem in mind a cable element with bending stiffness was implemented to allow form finding where it is possible to blend between cables and beams.

2.9 Testing and benchmarking

In order to verify the results from the implementation of beam elements in SMART Form a couple of test cases were produced and analyzed using different software. The approach was to test different deformation modes separately at first and then in combinations to demonstrate accuracy, stability and convergence. Analytical calculations were performed for simple test cases and the results were compared to the numerical analysis.

The tools used for the benchmarking was Matlab with the FEM toolbox Calfem and the commercial structural analysis software Oasys GSA. It was found beneficial to use Calfem at first, which enables complete control and insight into the calculations, in order to gain confidence for further testing with GSA. The way loads were applied in SMART Form at the time for benchmarking (half way through the thesis), was slightly different compared to GSA, hence only simple load cases could be studied. This is explained more in depth for each test case.

The following test cases were performed:

1. Elastica to test the bending capability. Results are compared with the same study performed by Sigrid Adriaenssens for her Phd dissertation, Stressed Spline Structures [1].

2. L-shaped cantilever beam exerted to self weight and point load to capture combined twisting and bending moment.

3. Simple gridshell analysed in SMART Form, Matlab and GSA.

2.9.1 Bending test

The non-linear behavior of a buckled pinned strut i.e. elastica, was studied by Euler over 250 years ago. Seemingly the first non-linear analysis of the elastic buckling phenomena [1].

An elastica is a straight member, in this case orientated along the global x-axis, with a constant circular cross section and two pinned joints at the ends constrained to the x-axis, which means that the end nodes are free to slide along that axis.

Point loads P and $-P$ are applied to the two ends directed along the x-axis introducing a compressive force in the member. A small transversal point load is applied in the middle of the elastica causing it to buckle. The test case is run with three different element discretizations to test convergence.

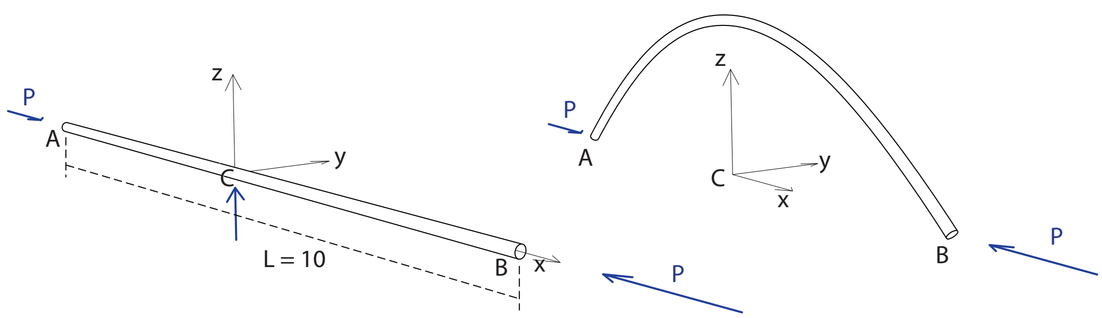


Figure 2.9.1: *Set up of the elastica test case.*

The following properties were used:

E is set to $210GPa$,

I is set to $4.7619e-7$, which multiplied with E gives the EI-value 100 kNm^2,

A is set to $4.7619e-4$, which multiplied with E gives the EA-value 100 MN,

L is the total length of the elastica and is chosen to 10 m.

Results from the bending test:

Buckled states	1		2		3		4	
Load P	10.48 kN		12.67 kN		18.46 kN		39.48 kN	
Displacement/Length	x/L	y/L	x/L	y/L	x/L	y/L	x/L	y/L
Analytical	0.4405	0.2110	0.2800	0.3595	0.0615	0.4015	-0.1700	0.3125
Numerical (16 links)	0.4380	0.2151	0.2716	0.3641	0.0406	0.4015	-0.2078	0.2978
Numerical (32 links)	0.4412	0.2100	0.2820	0.3586	0.0590	0.4018	-0.1772	0.3099
Numerical (64 links)	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	-0.1711	0.3123

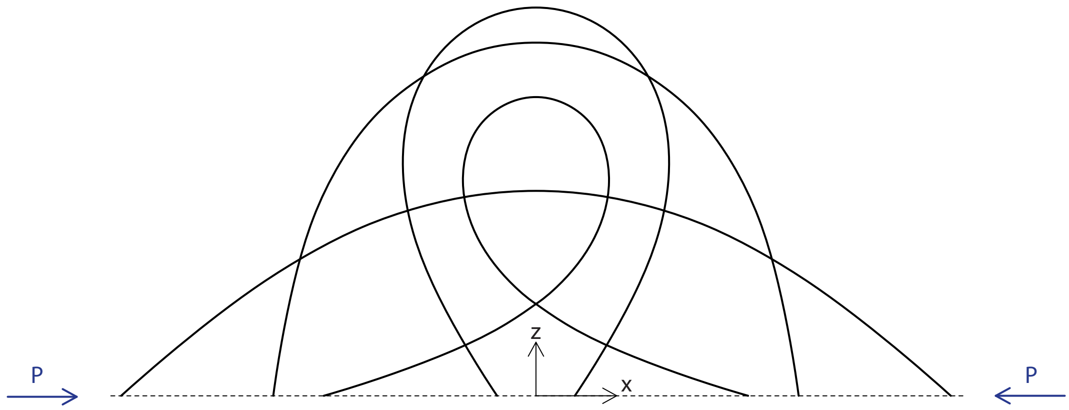


Figure 2.9.2: *Resulting deflected shapes for a 32-link elastica exerted to four different load magnitudes.*

As expected the deflection is getting closer to the analytical solution as the number of elements increases. The deflected shape is compared to a plot of the same analysis performed by Sigrid Adriaenssens for her Phd dissertation, and the results looks very similar. From the tabular values and the visual comparison the conclusion is that the bending test has been successful both in terms of accuracy, convergence and stability.

2.9.2 Twisting test

To validate the analysis for combined bending and twisting an L-shaped cantilever beam was modeled in SMART Form and compared with a model created in Matlab with the FEM toolbox CALFEM. The cantilever was modeled as a single symmetric L, each straight part 5 meters long. Three different element discretizations were used to test for convergence and the models were analyzed with both circular and rectangular hollow cross sections. A point load of magnitude 4325 kN is applied at C which is shown in Figure 2.9.3.

The following settings were used for the box section test:

E is set to $210GPa$

b is the width of the cross section, set to 0.1 m.

h is the width of the cross section, set to 0.2 m.

t is the cross section thickness, set to 0.01 m.

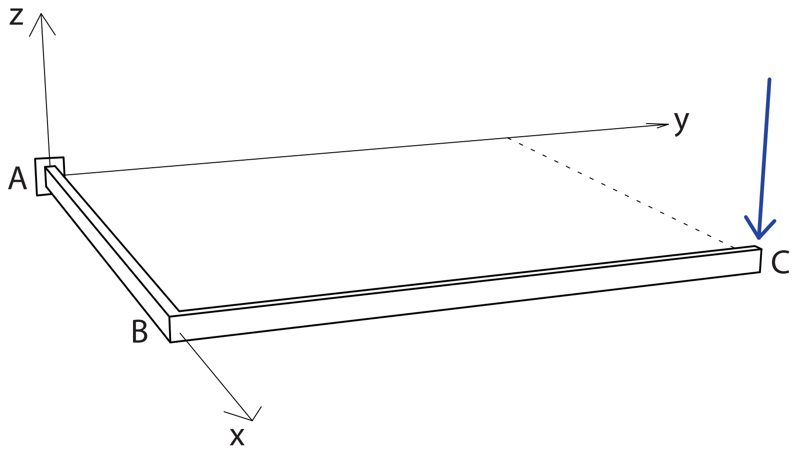


Figure 2.9.3: *Setup for the twisting test case. The cantilever beam is clamped at A and a point load is applied in C.*

I is calculated to $2.7787 * 10^{-05}$ m^4.

A is calculated to $0.0056m^4$

Results from the twisting test with a 4325 kN point load at C:

Quantity	Tip deflection $[m]$	Moment at A $[Nm]$	Twist at A $[Nm]$
Matlab (2 links)	-0.3899	21425.00	21425.00
Matlab (6 links)	-0.3899	21425.00	21425.00
Matlab (18 links)	-0.3899	21425.00	21425.00
SMARTForm (2 links)	-0.3897	21425.98	21415.76
SMARTForm (6 links)	-0.3888	21415.55	21375.71
SMARTForm (18 links)	-0.3887	21412.99	21372.03

Comparing the results from the SMART Form and Matlab analysis one can wonder why the tip deflection does not close in on the same solution as the number of elements increases. A likely explanation could be that the dynamic relaxation analysis in SMART Form, by its iterative nature, captures geometric nonlinearities which is not considered in the way the Matlab analysis are carried out.

2.9.3 Gridshell test case

The purpose with the last test case is to compare the SMART Form analysis for a complex structure with the commercial FEM software package GSA. The following cross section properties are used in SMART Form:

E is the Young's modulus and is set to $210GPa$,

G is the shear modulus and is set to $78GPa$,

ρ is the density and is set to $7800kg/m^3$,

r is the outer section radius and is set to 0.1 m,

t is the section thickness and is set to 0.01 m.

The grid shell is a triangulated dome shaped structure with a hexagonal base clamped at each of its six corners. The only load applied in the analysis is the self weight of the structure which is lumped

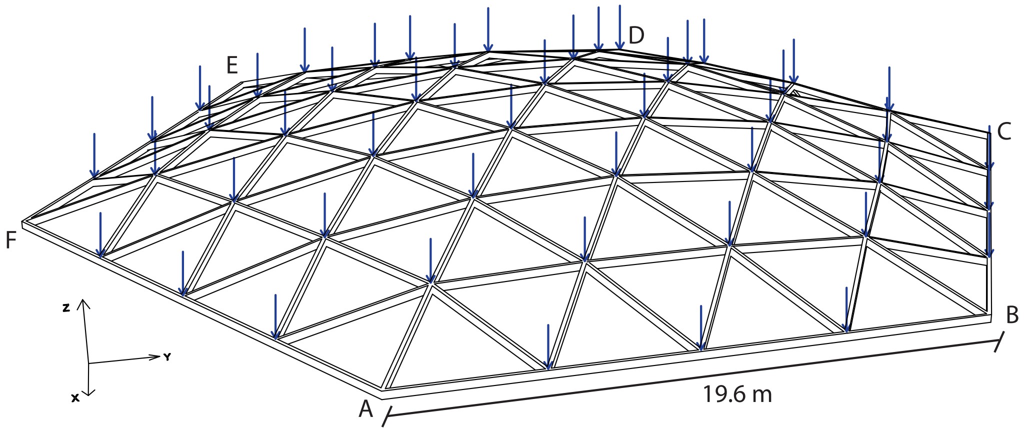


Figure 2.9.4: *Setup of the geometry for the grid shell test case. The structure is clamped at each of the six corners.*

onto the nodes solely as vertical forces. The implementation of distributed line loads was done at a later stage of the development and is not included in the benchmarking.

Quantity	DispX $[m]$	DispY $[m]$	DispZ $[m]$	MomentX $[Nm]$	MomentY $[Nm]$	Twist $[Nm]$	Axial $[N]$
Matlab	0.0023	0.0020	0.0091	8.8988E3	1.5658E3	1.3328E3	119.14E3
SMARTForm	0.0023	0.0019	0.0087	8.9272E3	1.5766E3	1.2820E3	119.39E3
GSA	0.0021	0.0018	0.0085	9.0319E3	1.4271E3	1.2773E3	121.58E3

Table 2.9.1: Results from the analysis where the loading is the lumped self weight. The table shows the maximum absolute value of each metric.

A slight deviation in the results is expected since the three models doesn't have identical settings. The Matlab model and the SMART Form model has the same load applied, but SMART Form inherently accounts for geometric nonlinearity, which is not considered in the Matlab analysis. GSA on the other hand is set up to account for geometric nonlinearities, but the self weight is applied as line loads rather than point loads like in the SMART Form analysis.

Another analysis is carried out where the self weight is neglected and only 1 kN point loads are applied to each of the nodes. The following results were obtained:

Quantity	DispX $[m]$	DispY $[m]$	DispZ $[m]$	MomentX $[Nm]$	MomentY $[Nm]$	Twist $[Nm]$	Axial $[N]$
Matlab	0.0068	0.0058	0.0240	23.238E3	4.7996E3	3.4886E3	251.70E3
SMARTForm	0.0066	0.0056	0.0231	23.350E3	4.8420E3	3.3537E3	252.55E3
GSA	0.0067	0.0057	0.0236	23.210E3	4.8084E3	3.4432E3	253.04E3

Table 2.9.2: Results from the analysis where the loading is applied as 1 kN on each node. The table shows the maximum absolute value of each metric.

Again the results show a slight deviation between the SMART Form and GSA. This might have to do with the choice of convergence criterion, material model or element discretization. Properties like Young's modulus and shear modulus as well as density for the steel material used in GSA was found to deviate slightly from the material model used in SMART Form. A more likely source of error however could be the element discretization. The gridshell in SMART Form is modeled with one single beam element for each link between two nodes, whereas multiple elements per link might be used in GSA.

For conceptual analysis the conclusion of the testing is that the deviation is acceptably low. Although more throughly testing would be required in order to gain confidence that the results from the implemented analysis are reliable.

3 Software

3.1 Programming in Rhino

Rhino 4 offers the ability to create plug-ins using the .NET platform, by implementing the Rhino DOT Net SDK. This SDK offers access to a vast library of Rhino commands including the Open NURBS library with classes for the geometric objects. Plug-ins for Rhino can be written using VB.NET, C#, C++.NET, Delphi.NET languages just to mention a few. For this thesis C# was the language of choice for the development.

For further information about the Rhino .NET SDK, the reader is referred to `http://wiki.mcneel.com/developer/dotnetplugins` [17].

3.2 SMART Form User interface

In this chapter a quick overview is given of the user interface developed during the thesis. The focus in the thesis has not been to create a user-friendly working environment, but rather to push the development of new interesting functionality within the scope of the topic. Nevertheless a glimpse of the user interface could be good as a bridge between the theory section and the case studies presented in the Results chapter.

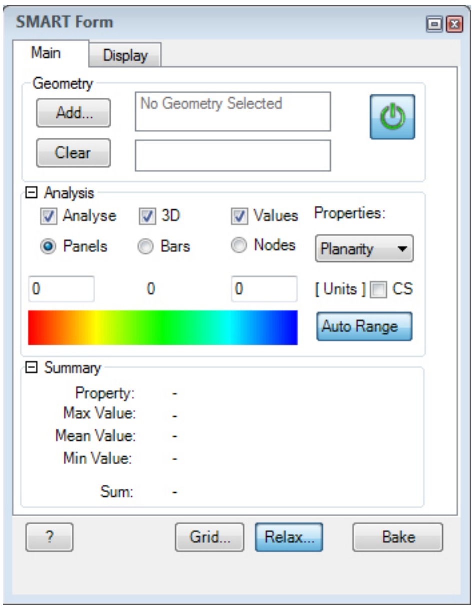


Figure 3.2.1: *Main dialog of SMART Form.*

The main dialog in SMART Form, shown in Figure 3.2.1, allows the user to control the visualization of the analysis. Most of the UI in this dialog was implemented prior to the thesis. However the *3D* checkbox enabling visualization of 3D beam elements and some extra options in the *Properties* combo box was added together with the options to show values and coordinate systems for the elements.

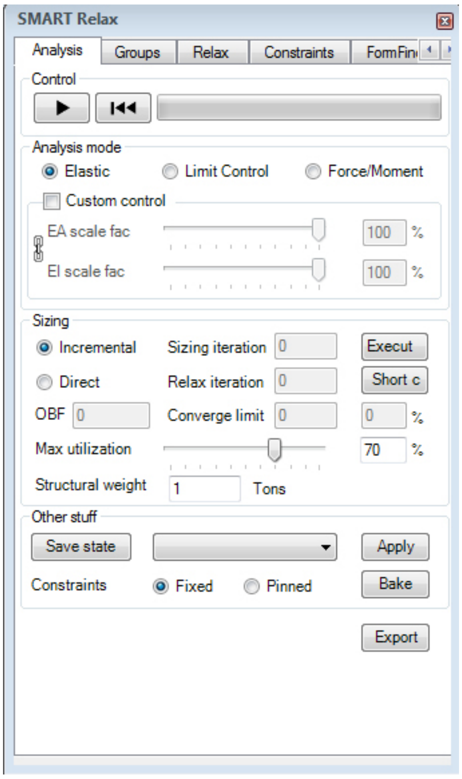


Figure 3.2.2: *Analysis tab of the user interface developed specifically for the thesis.*

The dialog in Figure 3.2.2 shows the analysis tab. At top the user can chose analysis mode which essentially means choosing between the three types of beam based form finding methods presented above. For regular analysis the default elastic analysis mode would be chosen. If the *Custom control* check box is checked the user have the possibility to scale the EA and EI values. If the mode is then switched to *Limit Control* or *Force/Moment control* the slider bars will change to control the levels of axial and bending utilization.

In the sizing group box the user can choose between *Incremental* and *Direct sizing* described in the sizing chapter. The slider bar with the label *Max utilization* controls the level of utilization to which the sizing is performed. The number of relaxation iterations, sizing iterations, the out-of-balance force, the convergence limit (as force) and a percentage of convergence is also displayed in this group box, giving the user information of the progress of the analysis.

In the last group box on the analysis tab there is a button called *Save state*. When the button is pressed the current geometry is saved as an option added to the neighboring combo box. Any amount of states can be saved and when a state is chosen and the *Apply* button is pressed the model jumps to that particular saved geometrical configuration, reseting all the initial lengths, rotations and coordinate systems. The *Bake* button adds the current geometry to the Rhino document.

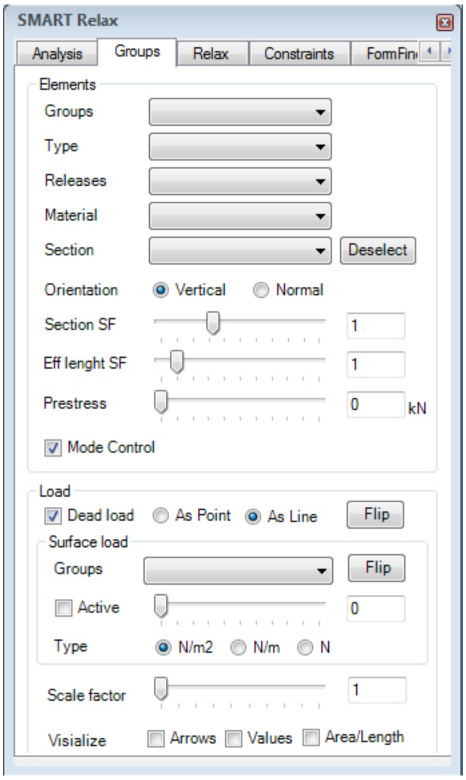


Figure 3.2.3: *Second tab of user interface developed specifically for the thesis.*

The *Groups* tab is where all the element properties and the loading is controlled. As a group is chosen from the top of the *Elements* combo box the other settings (in the same combo box) apply only to that group. The user can specify properties like element type (bar, beam cable, cable-beam), releases, material, section, orientation, cross section scale factor, effective length scale factor and prestress for the active group.

In the *Load* group box the user can chose to activate or deactivate the *Dead Load* with a check box. It can be chosen whether to apply the load as point load or line load and the direction can be changed using the *Flip* button.

The surfaces selected with the input geometry will end up as surface loads in the *Group* combo box in the *Surface Load* group box. The user can choose to active the load for a group of surfaces, control the magnitude using the slider bar and decide whether the value should be applied as N/m^2, N/m or just N.

The scale factor slider bar in the bottom of the tab is used for global buckling analysis. All the activated loads are scaled using the slider bar value. The user can also choose to visualize the active load with arrows, values, and even projections on the load surfaces.

4 Results

In this chapter the results from the software development are presented as a couple of case studies where the implemented functionality is demonstrated. The aim is to present a potentially new design process for conceptual structural design enabled by the means of real-time feedback for analysis, form finding and automatic sizing. The last case study concerns a real project where Buro Happold has been involved. These user stories have been important drivers during the development of functionality as well as user interface in SMART Form.

4.1 Simple grid shell

The first case study shows how to perform form finding for a simple gridshell using cable elements, and how the height of the roof relates to the need for cross section sizes i.e. structural material. It is also shown how to quickly evaluate the structural effects of changing cross section type and material.

The input geometry is created in Rhino, center lines for elements, points for fixities and surfaces for loading. The direction of the normals for each surface dictates in which direction the load is applied, this is illustrated in Figure 4.1.1.

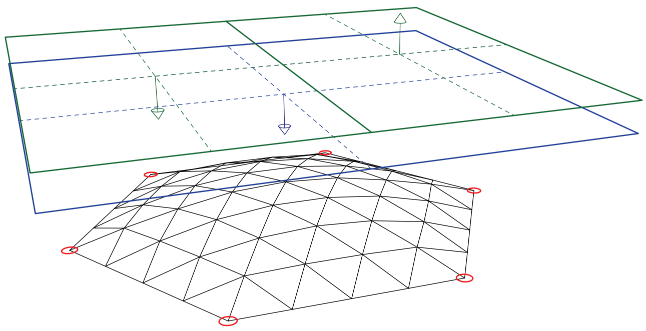


Figure 4.1.1: *Input geometry with two different load cases defined by the surfaces.*

The geometry is added to SMART Form and the analysis starts automatically with the self weight as applied loading. Initially, all the elements are set to beams with a default section (circular hollow $d = 0.2, t = 0.01$) and default material as structural steel. The only load applied on the structure is the self weight.

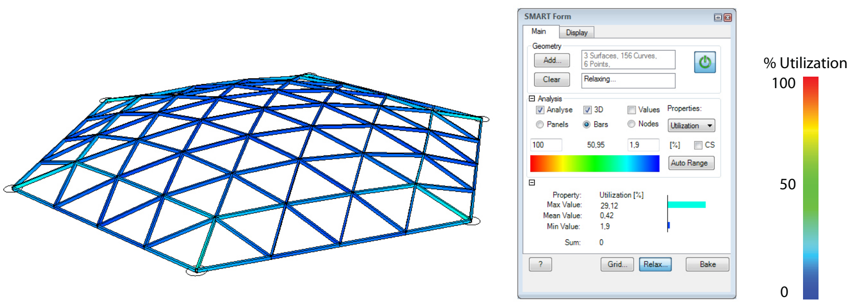


Figure 4.1.2: *Results from the initial analysis performed with default settings. The color plot shows the utilization of the beam elements based on the equations for combined axial force and moments found in [13] (0% utilization is blue, 100% is red).*

The cross sections are changed to rectangular hollow sections ($b = 0.1, h = 0.2, t = 0.01$) and the orientation of the elements is set to be normal to the surface. In addition to the self weight a line load of 3 kN/m is applied to the beams and all the loads are inverted to point in the positive z-direction.

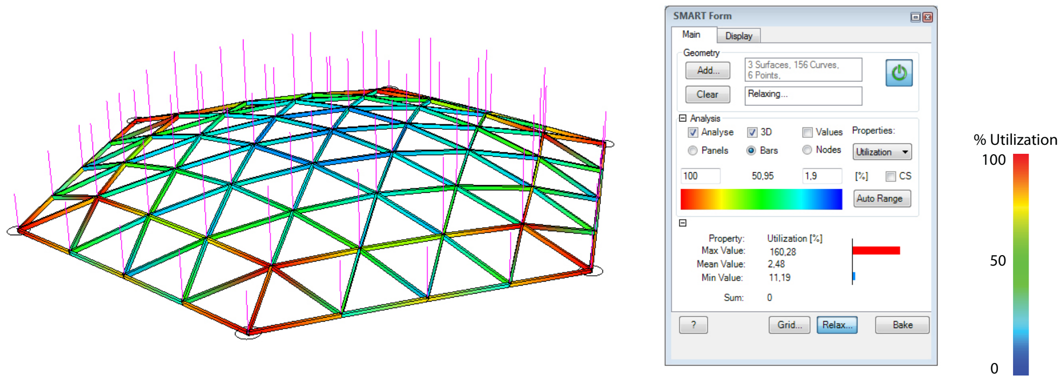


Figure 4.1.3: *Results from the analysis with the reversed loading (pointing in the positive z-direction). The color plot shows that some of the beams are over utilized due to the magnitude of the loading.*

The beam elements are changed into cables such that they work in pure tension. By elaborating with the section sizes under a constant load three different geometries are form found and saved in SMART Form. The load is inverted, now directed along in the negative z-axis, and the elements are set to beams again. The sought sizing utilization is set to 70%, and the sizing is carried out for the initial geometry together with the three different form found geometries.

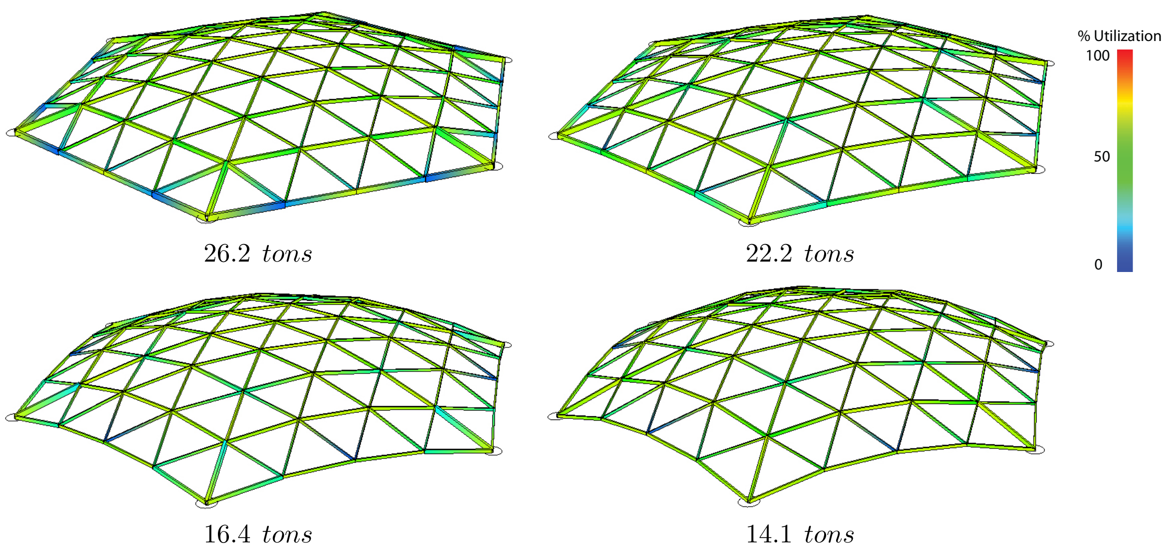
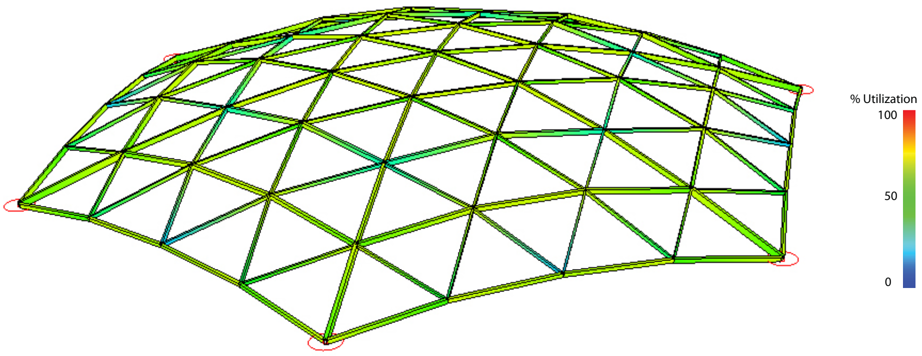


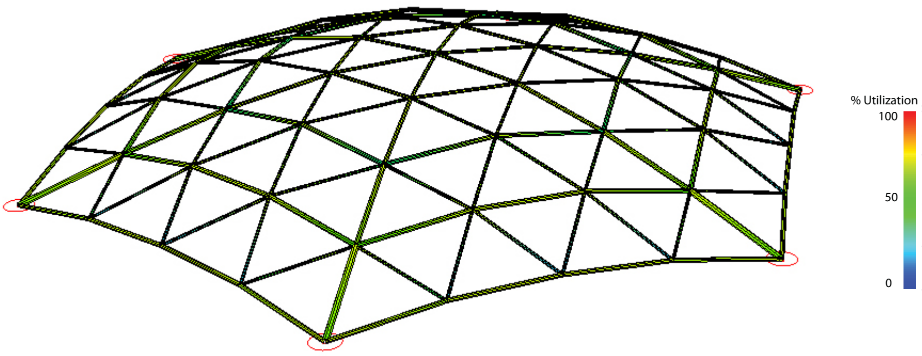
Figure 4.1.4: *The results from sizing of the four different geometries exerted to the same loading. The colors show the utilization which is close to 70 % (yellow) for each element.*

The third option is chosen for further studies since it is fairly similar to the initial geometry but with a 46% reduction in weight. It is of interest to compare the efficiency of different types of cross sections. Automatic sizing is preformed for the geometry with each of the different hollow section types (starting from the same default sections).



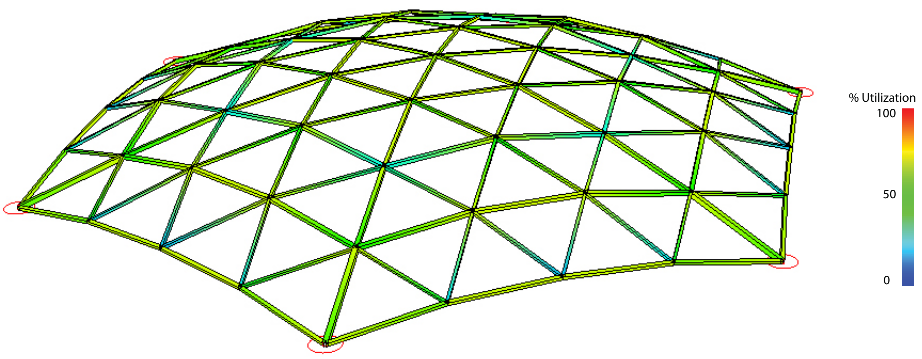
16.2 tons

Figure 4.1.5: *The results from the sizing with rectangular sections.*



10.6 tons

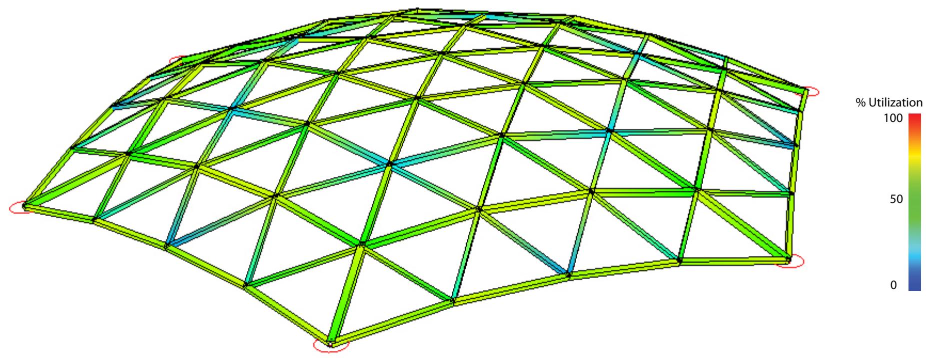
Figure 4.1.6: *The results from the sizing with circular sections.*



14.7 tons

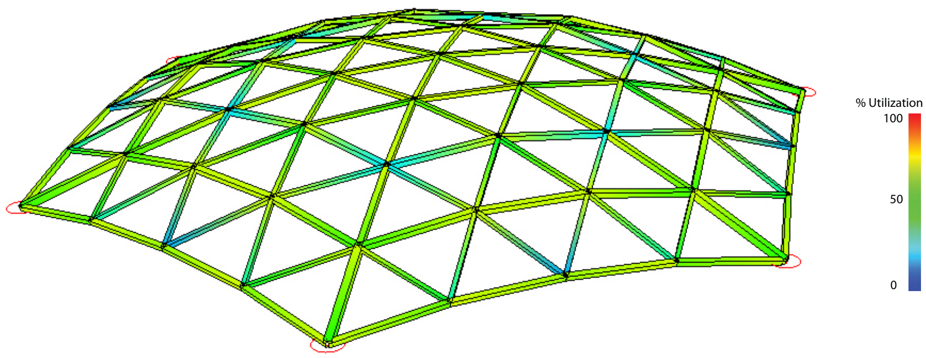
Figure 4.1.7: *The results from the sizing with square sections.*

The square section is chosen for further studies. Now we are interested in evaluating how the choice of material changes the structural performance. The material models implemented in SMART Form are very crude and do not capture a orthotropic behavior. However this is considered to be a fairly good approximation for conceptual analysis since the utilization of the elements (bars and beams) is mainly longitudinal.



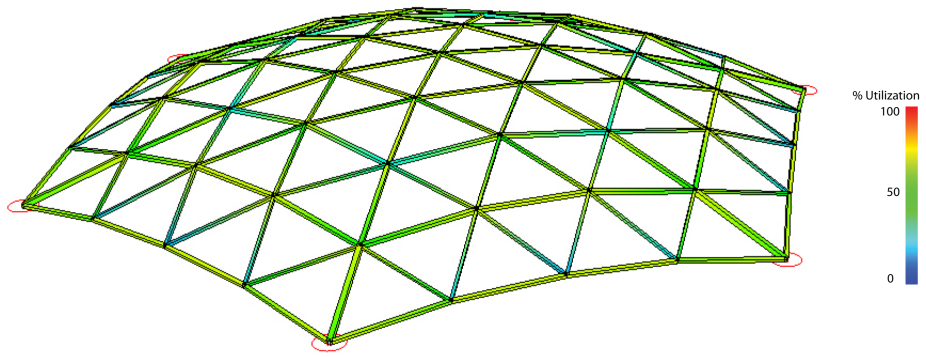
36.5 tons

Figure 4.1.8: *The results from the sizing with concrete beams.*



7.9 tons

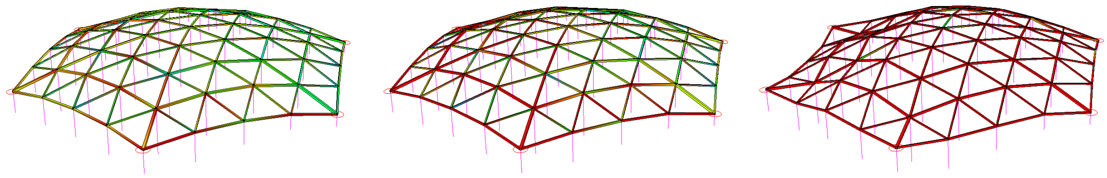
Figure 4.1.9: *The results from the sizing with timber beams.*



14.7 tons

Figure 4.1.10: *The results from the sizing with steel beams.*

Steel is chosen as the material and the structure is tested for sensitivity to global buckling. An additional asymmetric load is applied.



$Loadscalefactor$ 1.0 $\quad$ $Loadscalefactor$ 1.5 $\quad$ $Loadscalefactor$ 2.0

Figure 4.1.11: *Results from global buckling test.*

Conclusions

Since SMART Form is responding to changes in real time this study is very quick to perform and communicates the results in a clear way. The reliability of the results can be questioned especially with regards to material model and buckling behavior. To make the sizing reliable the choice of effective length for the members has to be carefully considered. The axial capacity is heavily dependent on this property and it was found difficult to automatically calculate for an arbitrary geometry.

4.2 Form improvement using limit control form finding

The second user story is about evaluating a free form structure and improving the form using limit control form finding. This method enables to improve the worst parts of the structure, by limiting the axial and bending capacity for the elements. The results are evaluated by comparing the total structural weight after sizing the members.

The input geometry is created in Rhino, center lines for elements and points for fixities and surfaces for loading. The points (marked as red circles in Figure 4.2.1) are fixed boundary conditions and the lines are separated into two layers, one for beams (black lines) and one for cables (blue lines).

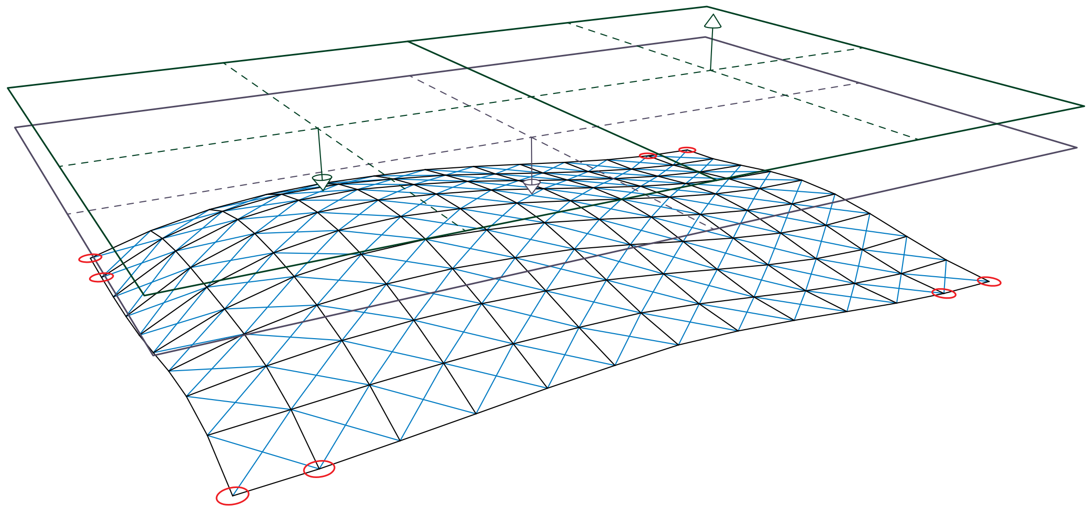


Figure 4.2.1: *Input geometry with two different load cases defined by the surfaces.*

The geometry is loaded into the plug-in and the analysis starts automatically with self weight as the applied loading. All the elements are initially set to beams with hollow circular sections ($d = 0.2, t = 0.01$) and with steel as material. The blue lines in Figure 4.2.1 are switched to cables and given a solid circular section with a smaller dimension. The supports are switched from initially being fixed to pinned and the only load applied on the structure is the self weight.

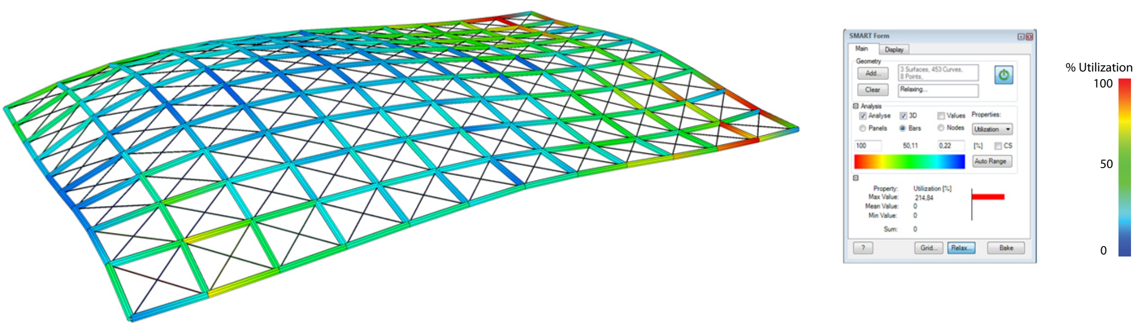


Figure 4.2.2: *Results from the initial analysis with self weight where the colors show the utilization of the elements.*

Additional loading (1 kN/m) is applied as line loads on the elements, and the analysis is updated.

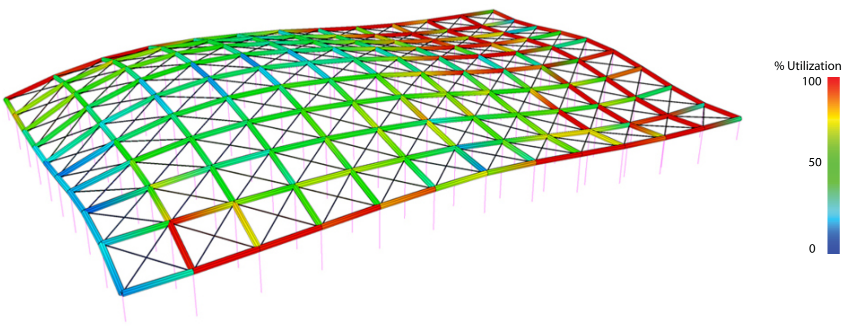


Figure 4.2.3: *Total utilization and the direction of the load.*

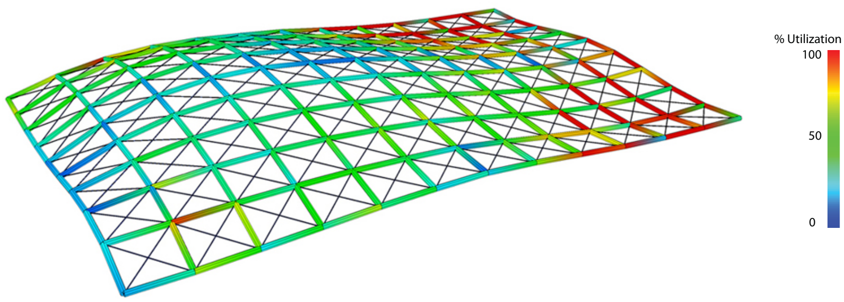


Figure 4.2.4: *Bending utilization.*

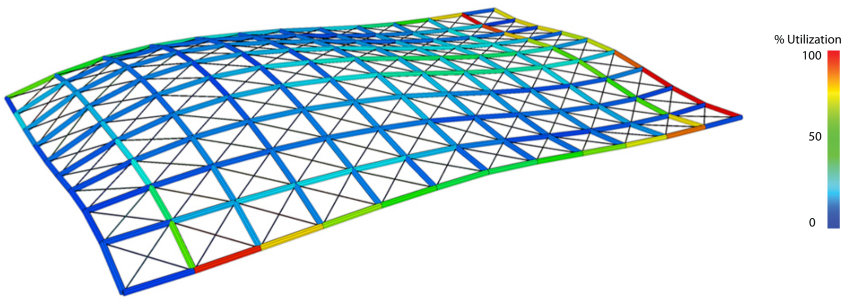


Figure 4.2.5: *Axial utilization.*

By studying the figures above it stands clear the structural action is dominated by bending. The lack of curvature on the right hand side leads to stress concentrations due to bending moments which will result in great section sizes and a heavy structure.

In order to reduce bending action but preserve some resemblance of the initial form, limit control form finding is carried out. The load is reversed to work as a form finding load and three geometries are form found with various settings. The resulting geometries are saved for sizing and comparison.

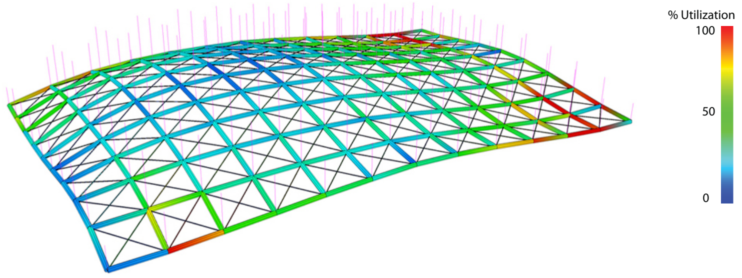


Figure 4.2.6: *Initial geometry exerted to the form finding load before the limit control form finding is executed. The colors show the utilization.*

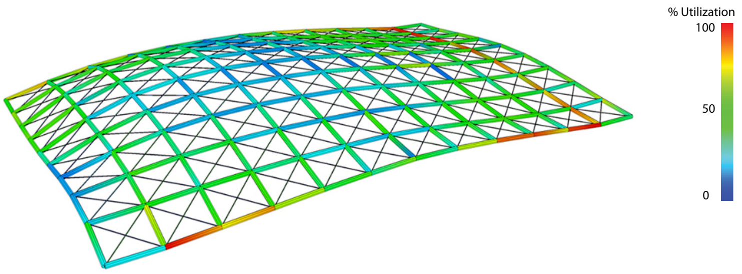


Figure 4.2.7: *50 % axial and 50 % bending capacity.*

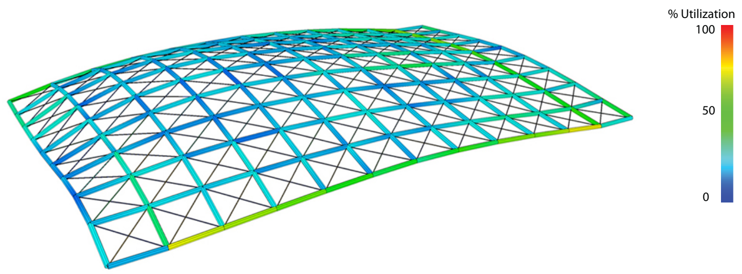


Figure 4.2.8: *80 % axial and 20 % bending capacity.*

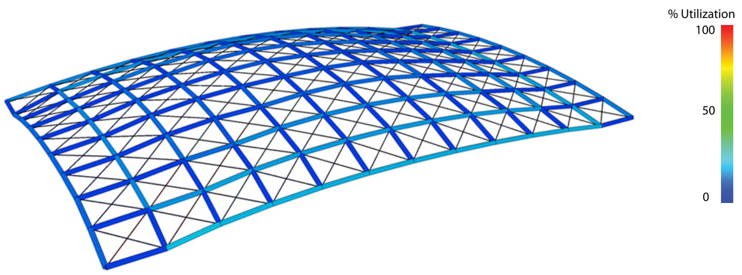
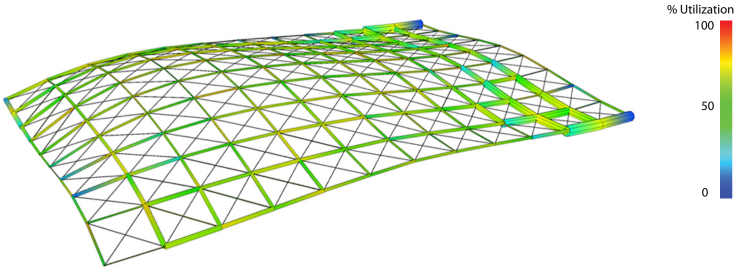


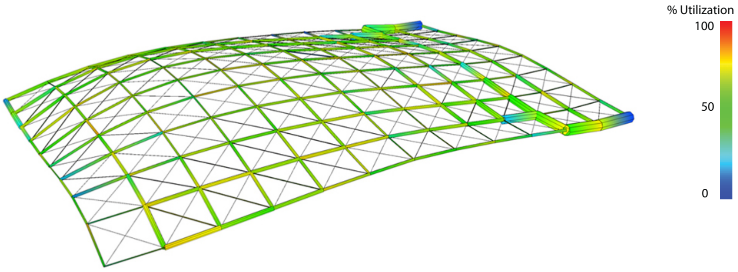
Figure 4.2.9: *100 % axial and 0 % bending capacity with cable-beam elements.*

The load is again reversed to point in the negative z-direction. The initial structure and each of the form found options are, based on the same loading, sized for stress, and the total tonnage is calculated. The effective length scale factor is set to 2.0.



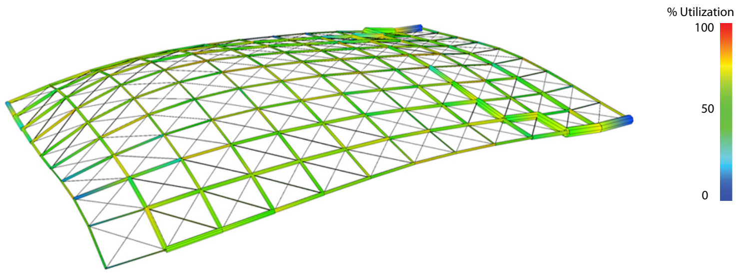
134.6 tons

Figure 4.2.10: *Sizing result for the Initial geometry.*



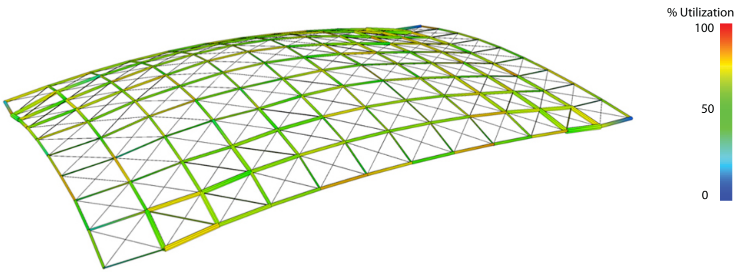
110.3 tons

Figure 4.2.11: *Sizing result for the option 50% axial and 50% bending.*



98.75 tons

Figure 4.2.12: *Sizing result for the option 80% axial 20% and bending.*

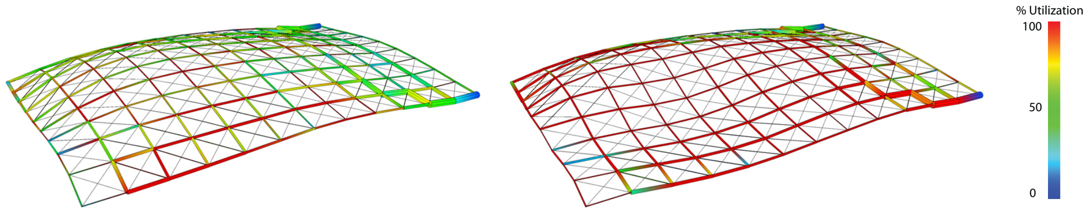


63.09 tons

Figure 4.2.13: *Sizing result for the option 100% axial and 0% bending.*

By studying the results from the sizing in Figure 4.2.10 - 4.2.13 it is clear that the lack of curvature at the far end of the structure is causing bending action, resulting in bulky sections. In the last option the form has changed enough for the structure to work predominately in axial action (rather then bending), hence the great drop in tonnage.

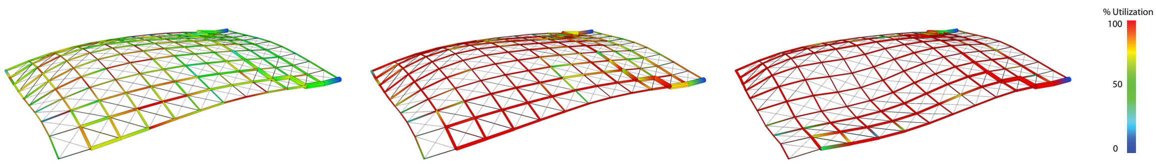
The third option is chosen for further analysis since the form is improved a fair bit, 28% reduction in weight, but still with resemblance of the initial form. After the sizing is performed it is of interest to study the sensitivity to global buckling. An additional asymmetric load (250 N/m) is applied to the structure as shown in figure 4.2.1.



Load scale factor 1.0 Load scale factor 1.5

Figure 4.2.14: *The results from the first buckling analysis.*

Clearly the structure is not strong enough to be regarded as safe from a global buckling point of view. Thus the structure has to be sized in a way that increases its capacity. This is done by changing the estimated effective length for the beam elements from being 2, to being 3 times their initial length, effectively reducing the capacity to carry axial load. The sizing is repeated, resulting in a slightly heavier structure (110 tons) and a new buckling study is performed.



Load scale factor 1.0 Load scale factor 1.5 Load scale factor 2.0

Figure 4.2.15: *The results from the second buckling analysis.*

Conclusions

From figure 4.2.15 the relation between reduction of bending moment and reduction of structural weight stands clear. The last option, form found with 0% bending capacity gives a 53 % reduction in weight compared to the initial structure, still capable of carrying the same load. On the other hand, in options 2 and 3 there are still similarities to the initial form, allowing for other design parameters, rather than the structural form finding, to dictate the form. Perhaps due to the free form nature and the shallowness of the structure, it is seemingly sensitive to global buckling even after the form improvement.

4.3 Exeter University roof

Architects Wilkinson Eyre together with engineers at Buro Happold were appointed to design the Campus for Exeter University Forum in 2008. They founded the concept for the design around the natural features of the site, creating a ”green corridor” to connect the campus buildings with the landscape [27]. The project includes new and old structures enclosed by an undulating gridshell roof. As well as unifying the buildings beneath and hosting a new reception, the roof is a shelter for a range of teaching and discussion spaces. It creates an enclosed street that links all of the facilities, while offering a good environment for learning, teaching and research [28]. A summary of the design process that engineers at Buro Happold went through for the concept design of the roof structure is presented, together with a proposal for a new way of approaching this type of design.

Figure 4.3.1: *A perspective view on the concepts for the Exeter University roof.*

Real design process

The concept design of the roof structure was an iterative process in close collaboration between engineers at Buro Happold and architects at Wilkinson Eyre. As the architects proposed changes to the geometry, due to the complexity of the structure, new analysis had to be performed in order for the engineers to point out the structural effects of the changes. One of the main drivers of the roof geometry was the view from the roof terrace. A flatter roof would give better views while inducing greater stress in the structural members. Thus the relation between roof height and section sizes was brought into discussion. A number of different analysis models with various heights were generated in the finite element software Robot. Analysis were then carried out, the most critical members sized and the results fed back to the architects. This procedure was repeated over and over again making the design work slow and inefficient [11].

Figure 4.3.2: *Pictures from the realized project.*

Proposal for new design approach

A proposal for how this design process could be approached, with the aid of real time structural analysis implemented into the designers environment, is presented here. Only the main part of the structure is considered in the example that follows. The area to cover with the roof structure is shown in Figure 4.3.3.

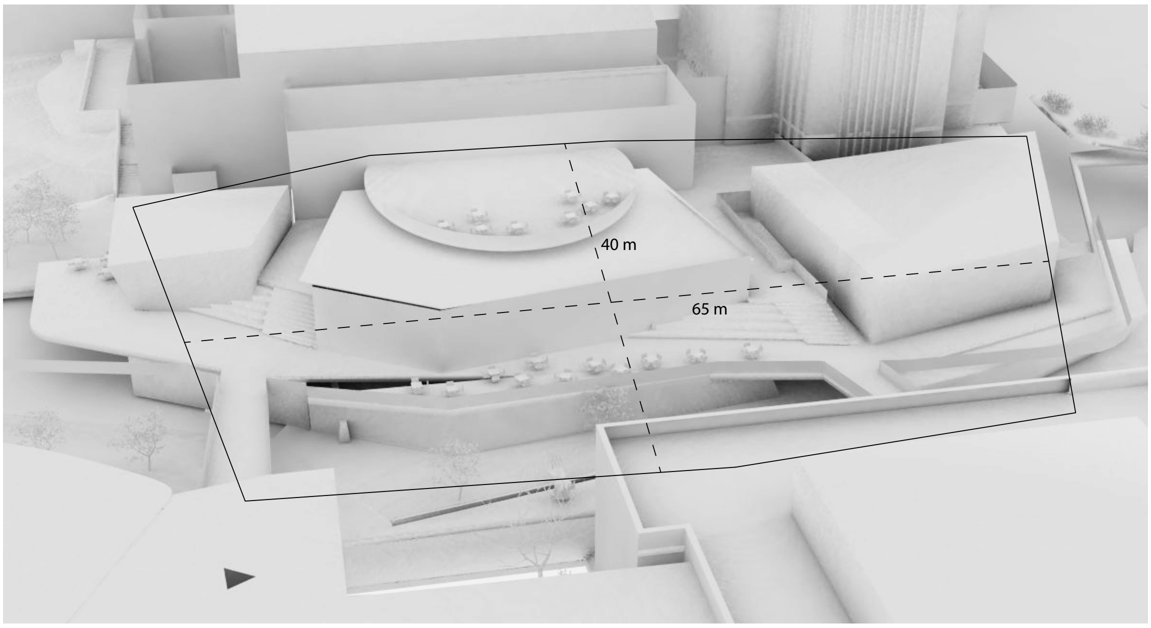


Figure 4.3.3: *Render from a 3D model of the Exeter University campus and the main area to be enclosed by the roof.*

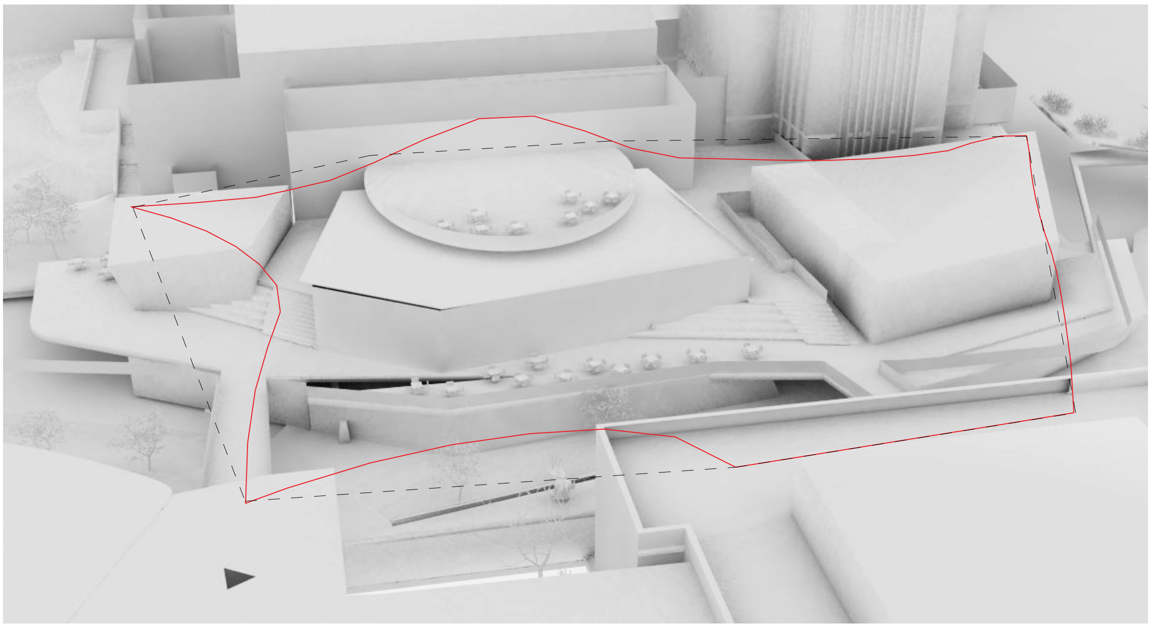


Figure 4.3.4: *Boundary for the roof with a red curve proposed by the architects.*

The area within the boundary is triangulated into a grid and a hole is made for the roof terrace. As mentioned an important driver of the roof design was the view from the terrace. Structural performance also played an important role, a slender structure would give a light and spacious expression from inside. To compromise between view and structural height form elaboration was carried out with focus on the sight lines, shown with the arrows in Figure (4.3.5).

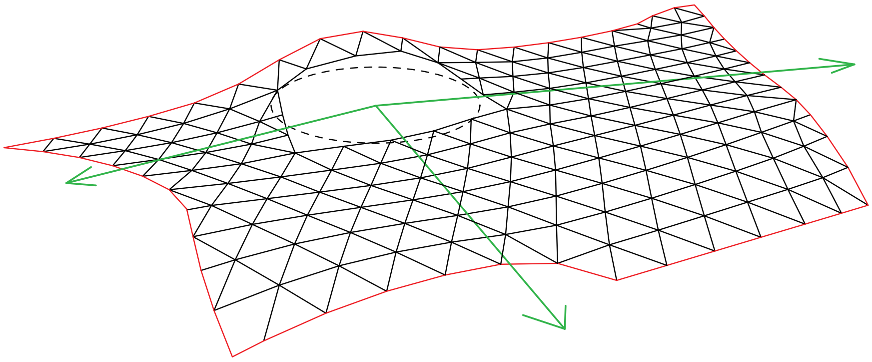


Figure 4.3.5: *The roof terrace is shown as a dashed circle and the desired sight lines as green arrows.*

The input geometry for SMART Form consisted of center lines for the beam elements, points for fixities and surfaces for loading. Three different groups of surfaces were used for the loading. The group called *Whole Load* is used equivalent to a snow load and acts as the governing load case for the form finding. The *Quarter Load* and *Half Load* are used to impose asymmetric loading for the global buckling studies. The element center lines are put into different layers, enabling separated control of the section properties for the elements during the form finding process.

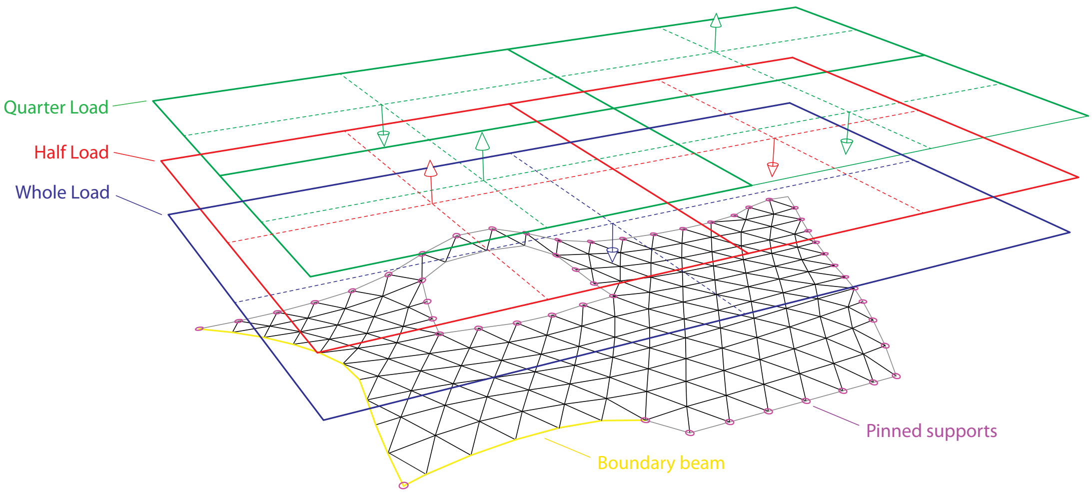


Figure 4.3.6: *Input geometry for SMART Form.*

The elements in SMART Form are converted into cables and a load of $10kN/m^2$ is applied as form finding load in the positive z-direction. The magnitude of the load is not important in this stage since it is only used to drive the shaping of the structure. The elements on the edge in the lower left corner are converted to beams and given a huge solid circular section, this is to prevent the boundary from changing during the form finding process. By keeping the load constant and varying the section sizes for the different element layers, four form options are generated.

The first option shown in Figure (4.3.7) is one where the view parameter is completely ignored and the structure is form found to a convenient height, comparable with the third option. The second option is a flat structure where good views are obtained solely by the flatness of the roof. In the third and forth option however, a compromise between view and structural efficiency is considered by forcing parts of the structure to be shallow whilst other parts are higher.

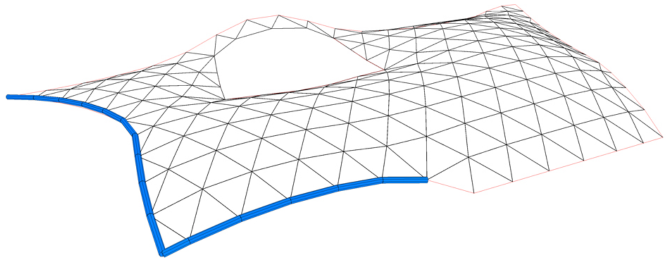


Figure 4.3.7: *First form found option when the view is not considered and the cable sections are changed to give a convenient roof height.*

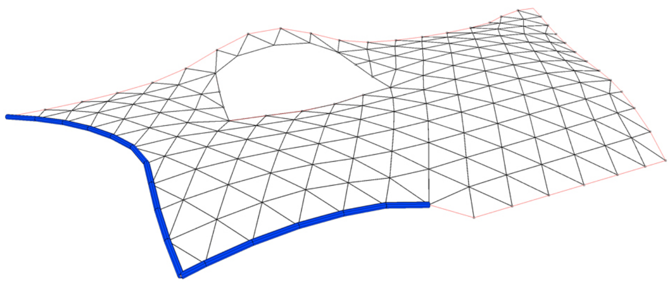


Figure 4.3.8: *Second form found option where the sizes of the cables are adjusted such that a flat structure is obtained giving views in all directions.*

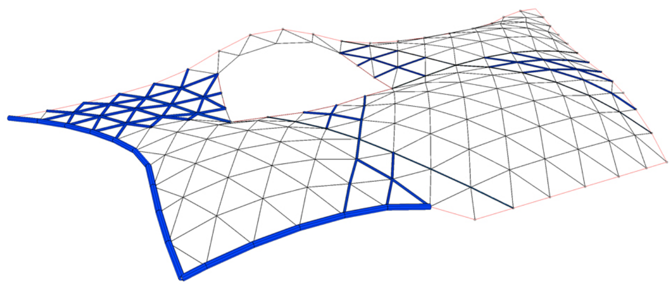


Figure 4.3.9: *Third form found option where the cable sizes are adjusted per layer giving a structure where specific sight lines are prioritized.*

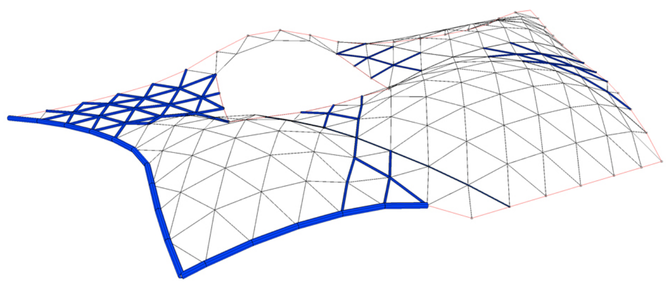
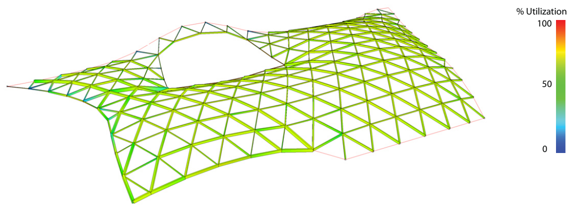


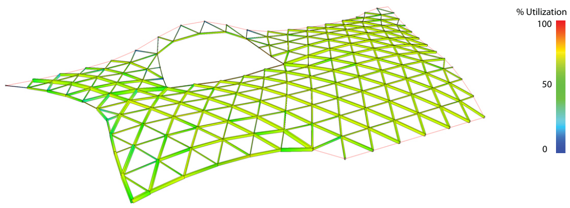
Figure 4.3.10: *Forth form found option where the cable sizes are adjusted per layer giving a structure where specific sight lines are prioritized. This option is similar to the third one but slightly higher.*

The load from the *Whole Load* surface is reversed and the magnitude set to $1000N/m^2$, the self weight is also activated as additional loading. The effective length scale factor from eq. (2.6.10) is set to 2.0, which is a conservative value in the sense that it will reduce the capacity for the members. The sizing utilization level is set to 70 %, meaning that each member will be sized to utilize approximately that amount of its capacity. The result of these assumptions will be verified with a heuristic global buckling check in the end of the design process. The beams are given solid rectangular sections orientated normal to the surface. Timber is chosen as material model (treated as an isotropic material) since glue lamb beams was the choice for the real project. Based on these settings each of the form found options are sized and compared.



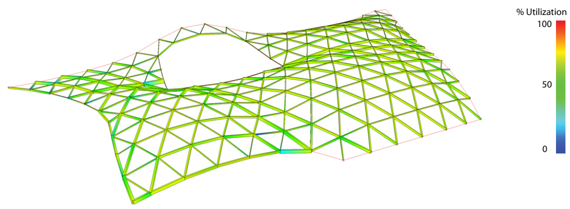
30.47 tons

Figure 4.3.11: *Result from sizing for option 1.*



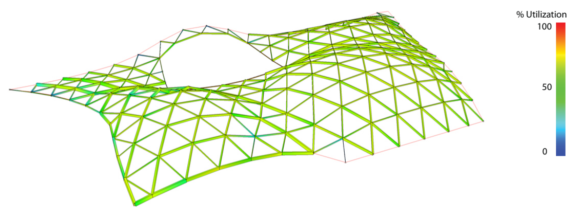
39.10 tons

Figure 4.3.12: *Result from sizing option 2, the flat structure.*



33.16 tons

Figure 4.3.13: *Result from sizing option 3, which is normal high with sight lines.*



30.01 tons

Figure 4.3.14: *Result from sizing option 4, which is the high version with sight lines.*

The axial force is calculated and shown in the figures below.

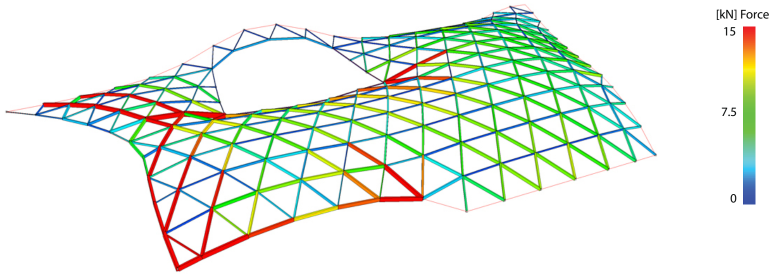


Figure 4.3.15: *Axial force for option 1.*

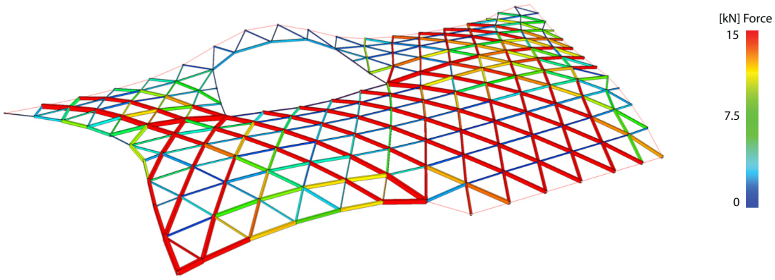


Figure 4.3.16: *Axial force for option 2.*

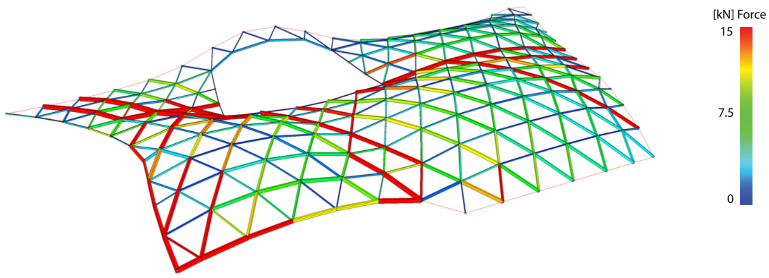


Figure 4.3.17: *Axial force for option 3.*

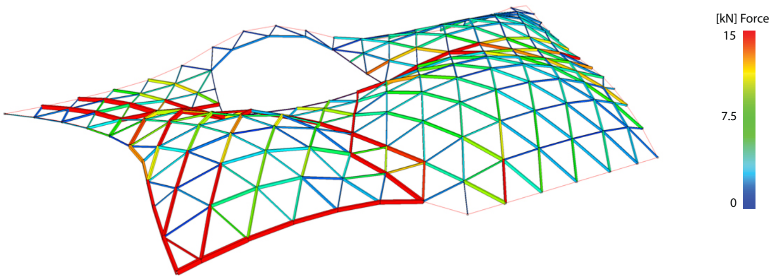


Figure 4.3.18: *Axial force for option 4.*

Equally the moment around the local x-axis (which is the stiffest bending axis the the rectangular sections) is calculated and shown in the figures below.

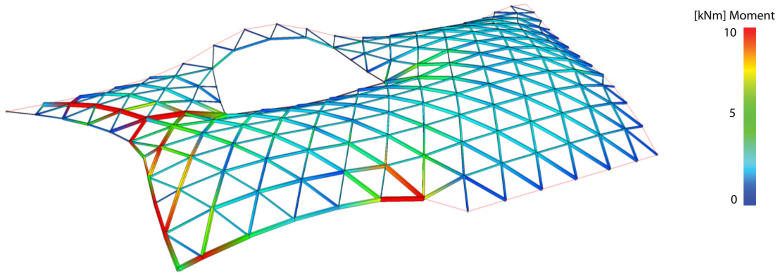


Figure 4.3.19: *Moment around the local x-axis force for option 1.*

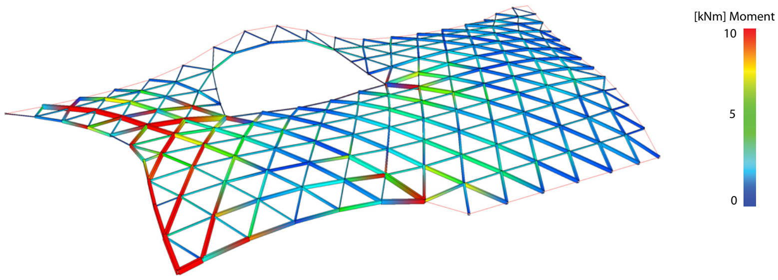


Figure 4.3.20: *Moment around the local x-axis force for option 2.*

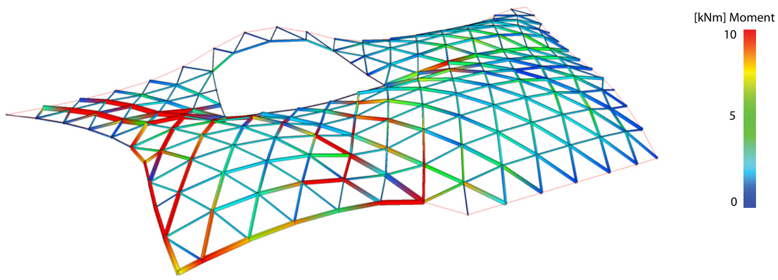


Figure 4.3.21: *Moment around the local x-axis force for option 3.*

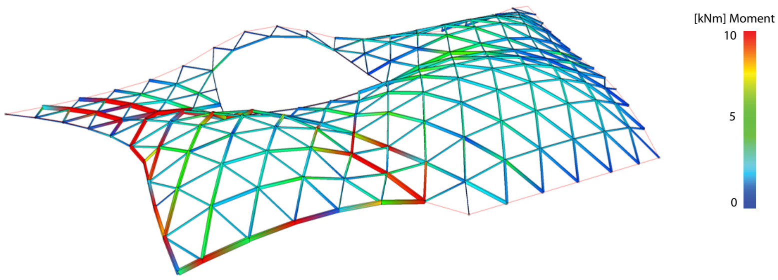


Figure 4.3.22: *Moment around the local x-axis force for option 4.*

Figure 4.3.15 - 4.3.18 and Figure 4.3.19-4.3.22 clearly show that the left hand side of the structure is working in bending action resulting in larger section sizes. It is also clear that some bending was introduced in options 3 and 4 where parts of the structure is pushed down to enable sight lines.

Option no. 3 is chosen for further studies since it is a relatively light structure where sight lines are prioritized without making the roof too high. The geometry is out put into Rhinoceros for the option to manually change the geometry.

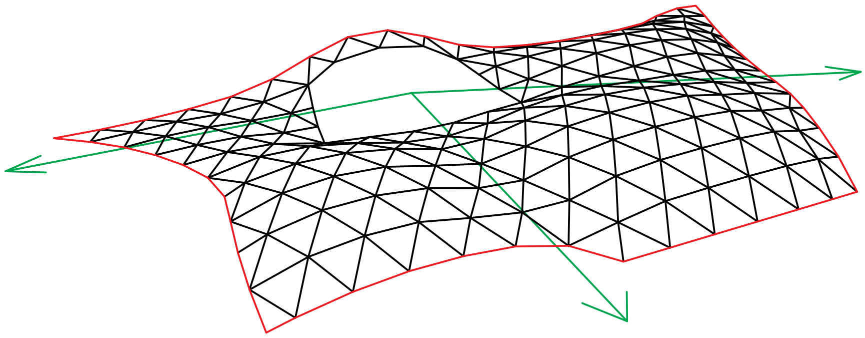


Figure 4.3.23: *Form found geometry out put into Rhinoceros for manual changes.*

After the manual changes (to smoothen out parts of the grid) the structure is relaxed again using cable elements, making sure that the changes did not cause unnecessary bending action. The real form finding load, self weight + $1000N/m^2$ is used for this purpose.

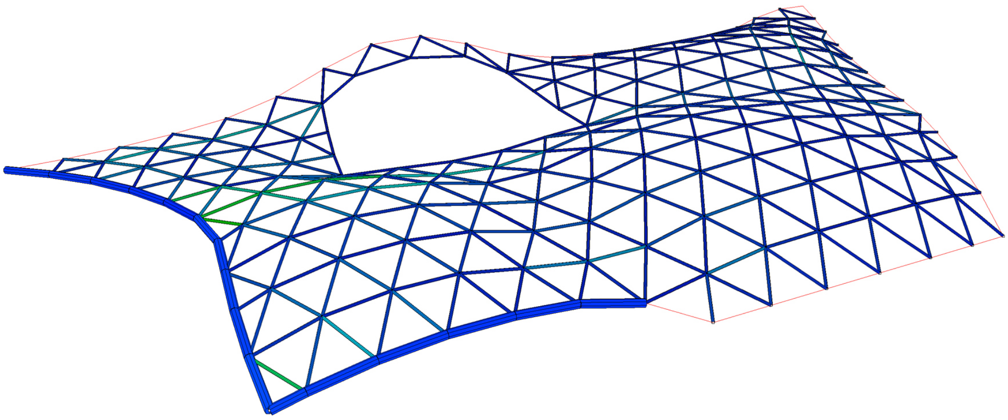


Figure 4.3.24: *Relaxed geometry, the border in the lower left corner is again modeled with massive beam elements to keep it from changing during the form finding process.*

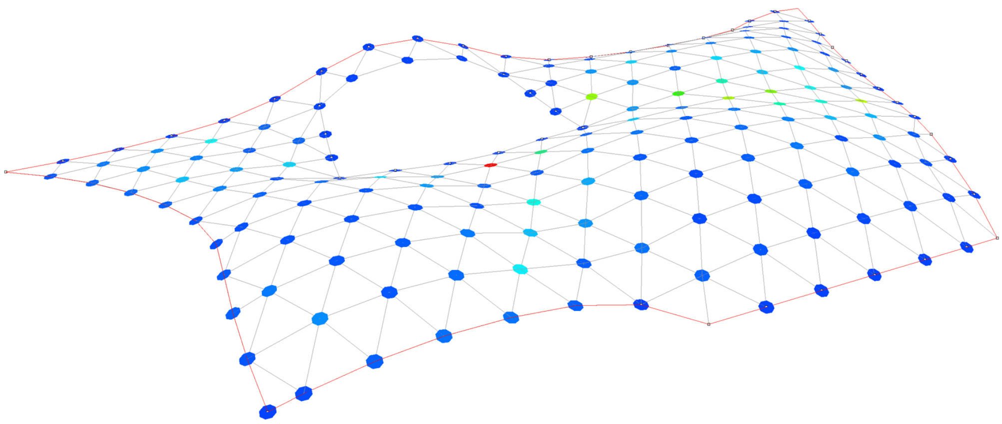


Figure 4.3.25: *Displacement of the nodes due to the final relaxation.*

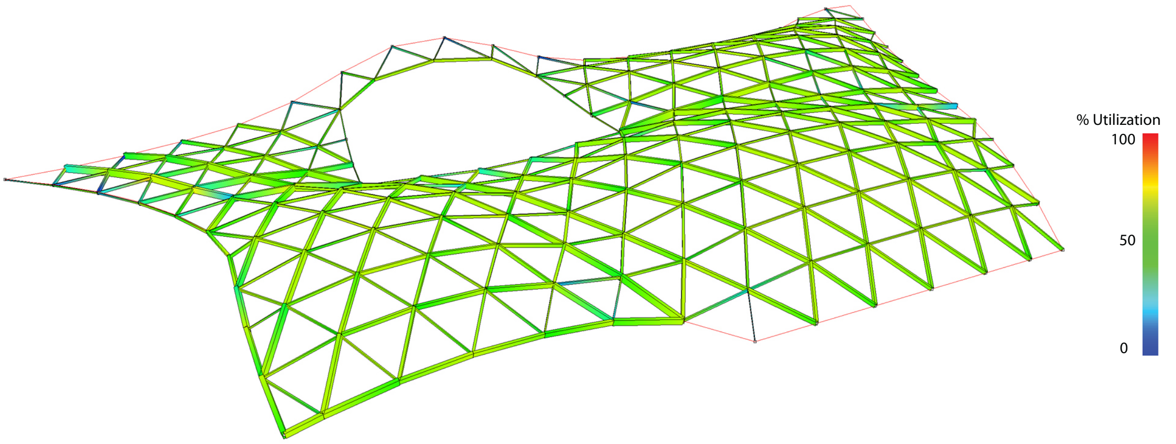


Figure 4.3.26: *Result from sizing the gridshell. Structural weight 33 tons.*

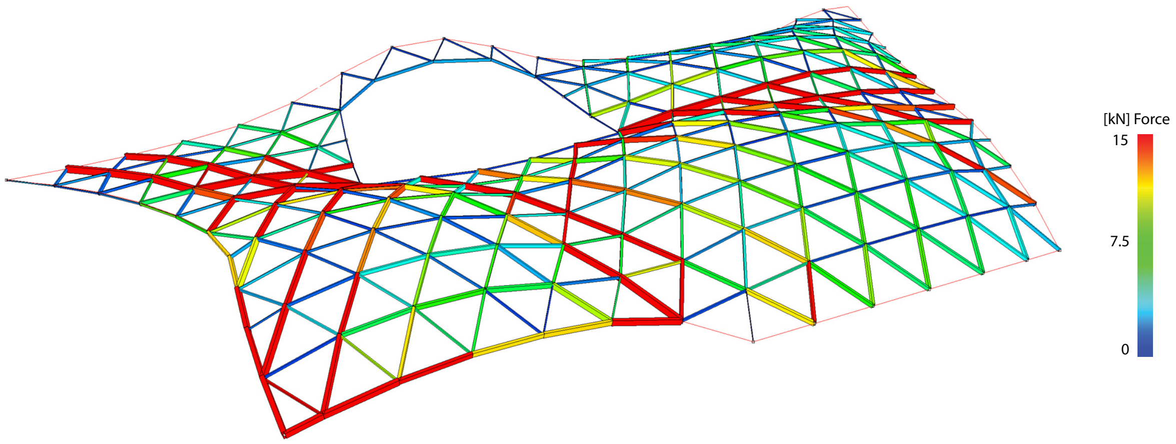


Figure 4.3.27: *Axial force in the members, red is 150 kN or more.*

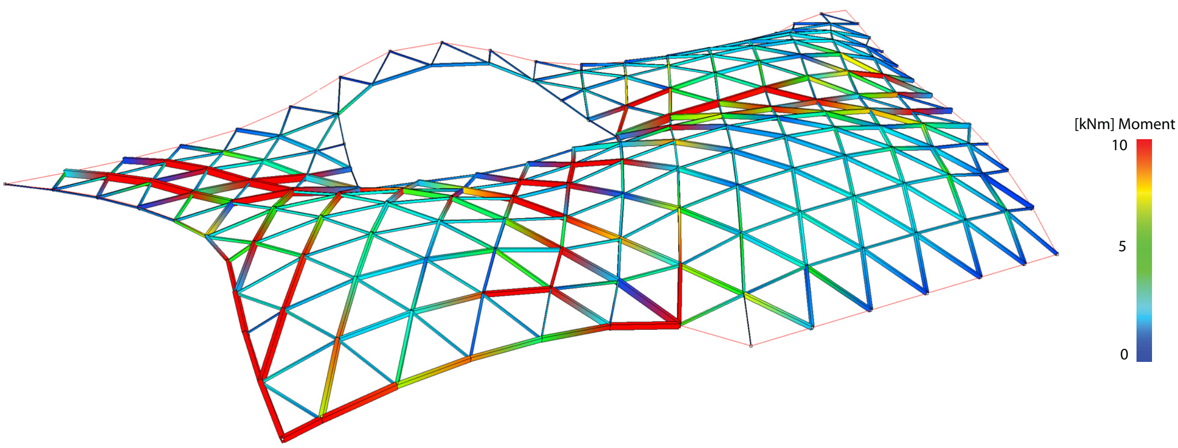


Figure 4.3.28: *Moment around the local x-axis for the members, red is 10 kNm or more.*

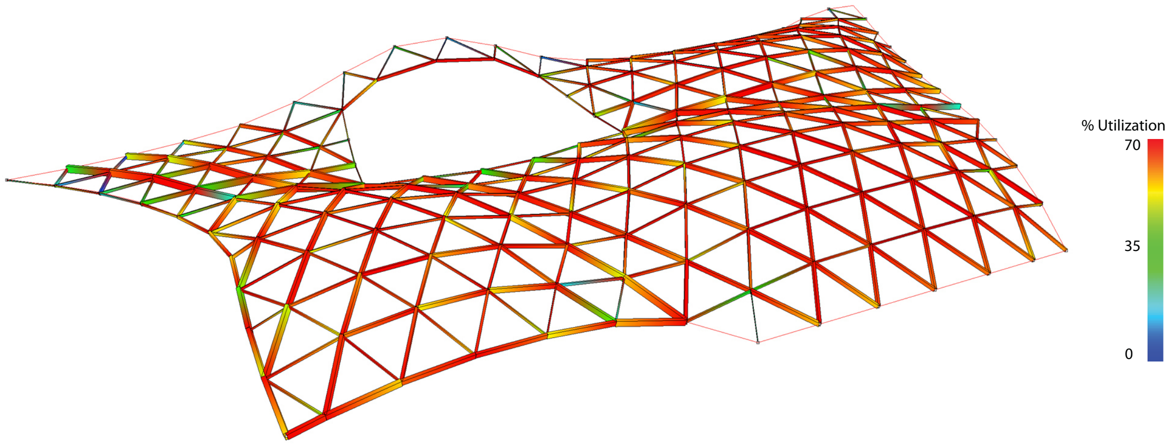


Figure 4.3.29: *Total utilization where red is set to 70 %.*

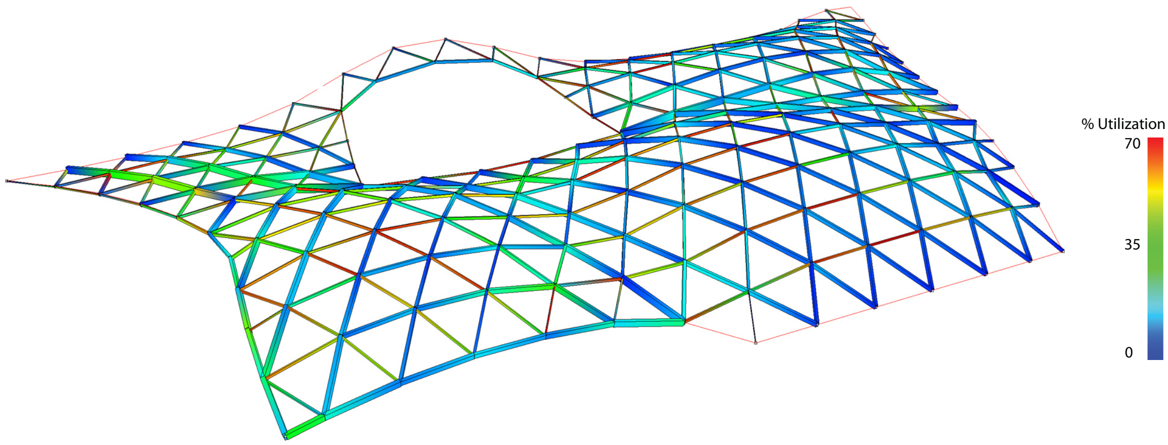


Figure 4.3.30: *Bending utilization where red is set to 70 %.*

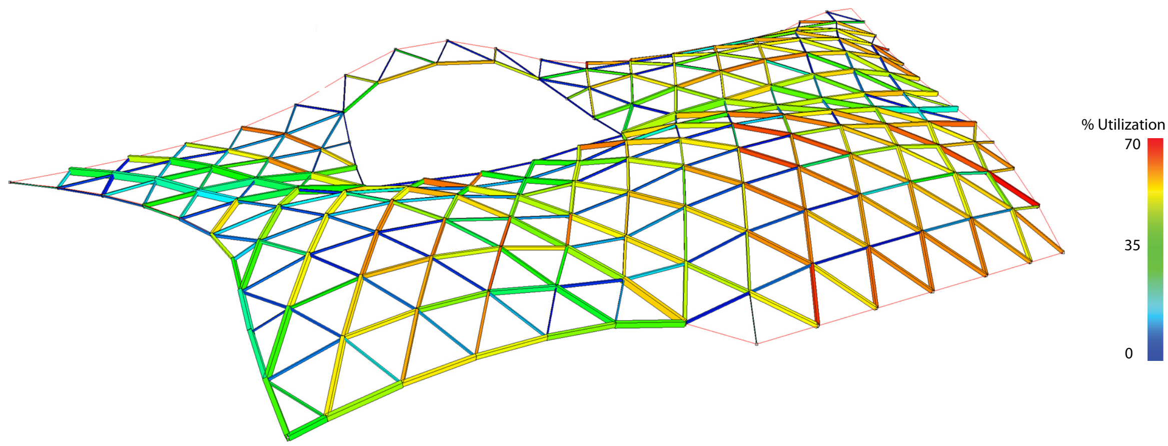


Figure 4.3.31: *Axial utilization where red is set to 70 %.*

Finally the structure is checked for global buckling. This is done by applying all three load cases shown in Figure 4.3.6 together with self weight and slowly increasing the load scale factor. Each load case is set to $500N/m^2$ and due to the direction of the load surfaces the total applied load is asymmetric.

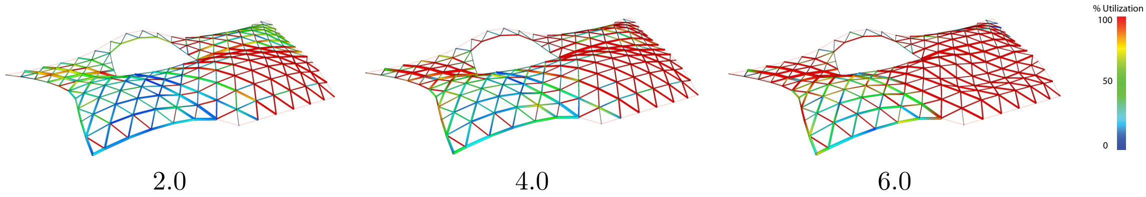


Figure 4.3.32: *Utilization of the elements during a global buckling study with the load defined as shown in Figure 4.3.6.*

The direction of the *Half Load* is reversed which changes the asymmetry of the loading, and another buckling study is performed.

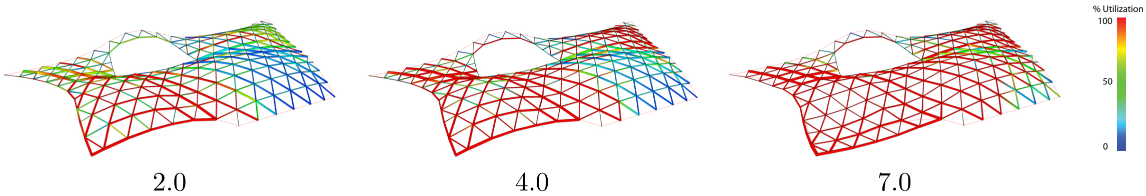


Figure 4.3.33: *Utilization of the elements during a global buckling study where the load direction is reversed for the Half Load.*

Finally a buckling study with a symmetric load case is performed. The *Whole Load* magnitude is set to $1000N/m^2$ and the load scale factor is slowly ramped up.

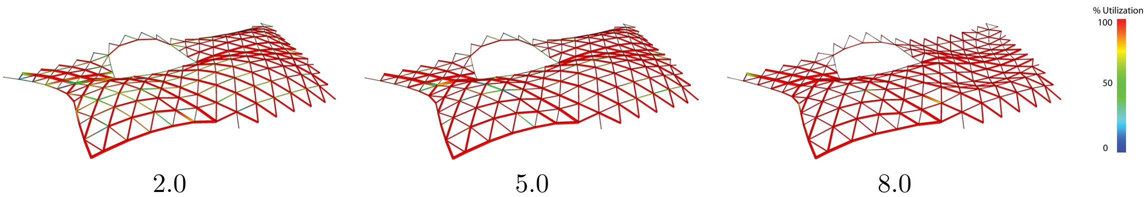


Figure 4.3.34: *Utilization of the members during a global buckling study with symmetric loading.*

The structure is seemingly well suited to resist global buckling, hence the assumption to set the effective length scale factor to 2.0 might be over conservative. By changing the scale factor to 1.0 and resizing the structure a total weight of 22 tons was obtained, to compare with the previous weight of 33 tons. By performing a new global buckling study it is found that the limiting load scale factor is around 3, which on the other hand might be a bit on the low side.

Conclusion

This study was very quick to perform and for this particular case SMART Form worked well as a design tool. It was sort of a three step process were the tool was used to form the structure from a more or less flat initial grid, the structure was then output for manual changes to finally be brought back into SMART Form for the last optimization and analysis. In the processes of forming the different options the architectural constraints (the edge beam and the view) could be accounted for and with the ability of sizing the members it was quick to evaluate the consequents of changing these constrains in terms of material use. However some sort of rationalization would have been useful in the sizing such that each cross section is not unique with arbitrary dimensions.

5 Disscusion

The implementation of beam elements in SMART Form has on the one hand taken the tool to a new level of sophistication, and on the other hand made it less intuitive and more complicated to use. In order to make the tool useful for the broader community the graphical user interface would need a revisit and more thoroughly testing would have to be carried out on the code and implemented functionality. Despite these inadequacies, the new development has resulted in a potentially very useful tool for performance driven architecture. In contrast to prior this thesis, SMART Form now deals with real dimensions and feeds back realistic metrics (forces, moments, weight etc) making it a possible driver for performance based design as discussed in the introduction. Architectural and structural parameters can be accounted for and compromised in a way that is usually a tedious and slow iterative process, one example of which is shown with the Exeter roof design process in the results section. But the usefulness of the tool heavily relies on the possibility to provide real time feedback.

Real time feedback not only enables the exploration a great amount of permutations of a concept in a short period of time, it can also inform a designer in an different way compared to regular analysis. One can study the redistribution of forces as the stiffness suddenly changes or compare the sensitivity and effect of changing a variety of parameters. For example, what happens to the force distribution in a structure if the bending stiffness suddenly drops or if prestress is introduced in the cable elements.

To make the analysis in SMART Form in real time for an arbitrary structure was found to be a great challenge. It was concluded that, in the way dynamic relaxation (DR) is implemented, it works efficiently for certain types of structures and is less suitable for other types. The first measure taken to improve speed was to reformulate the masses and moments of inertia from scalar values to matrices with separate magnitudes for each degree of freedom. This implementation made significant improvement but did not solve the whole problem. Consider the truss in Figure 5.0.1, under influence of a vertical load, the node i at the tip may have to move a significant distance in order for the structure to find static equilibrium. This movement is predominantly caused by movements of the other nodes in the structure, rather then deformation of the adjacent elements. But to ensure local stability the nodal mass and moment of inertia matrices (which for a fixed time step together with the applied force dictates how far the node can move or rotate in one iteration) are calculated based on the local stiffness. Meaning that, if the node is locally stiff in comparison to its global stiffness, it will have to move that long distance to the equilibrium position with tiny increments, causing convergence to be slow. This was found to be problematic in particular for space truss types of structures where the local stiffness often is great compared to the global stiffness. Gridshell structures, on the other hand, have a weak direction (normal to the surface) which seemingly makes them more pertinent for DR.

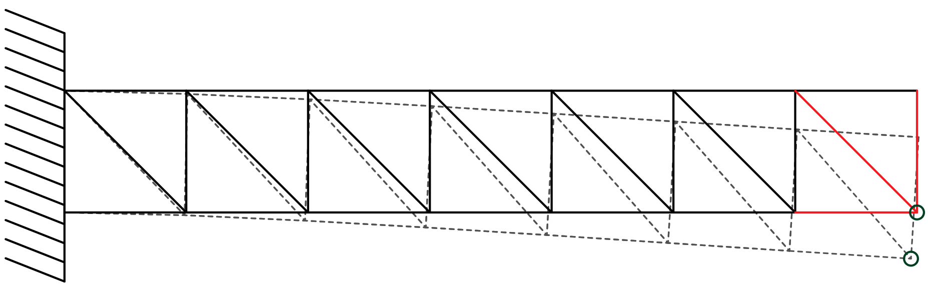


Figure 5.0.1: *A truss illustrating the local and global stiffness for node i. The local stiffness is based on the contribution from the adjacent elements, colored in red, while the global stiffness is dependent on the whole structure.*

In contrast to DR a matrix solver is independent of the relation between local and global stiffness. However, iterative matrix inversions have to be performed in order to capture geometrical nonlinearity, making the results comparable with DR analysis. A matrix solver does not calculate intermediate steps (between initial geometry and final displacement), thus is not ideal for real time feed back, but it has a great advantage in speed. One idea that came up during discussions regarding the speed problem related to the local/global stiffness ratio discussed above, was the implementation of an additional matrix solver in a separated from the DR analysis. When the matrix solution is calculated, the relaxation could then take on from the obtained displacements, either by jumping directly to the linear solution or by interpolation to mimic real time behavior. But before implementing this matrix approach it would make sense to further explore different choices of time stepping methods like Euler backward, Runge-Kutta or leap frog. It could also be beneficial, from a speed point of view, to separate the calculations into different threads, for instance by grouping the nodes and assigning each group its own thread.

Another interesting application of combined DR and a matrix method, could potentially be in automatic sizing. For the first iteration, when the static equilibrium is sought, based on initial geometry and the applied load, dynamic relaxation is quite slow and a matrix method would be preferred. But for the rest of the iterations, when the sizes are changing, the solution is most likely close, already to begin with. For this purpose dynamic relaxation could be quite efficient.

An important benefit with dynamic relaxation was found to be the numeric decoupling between the equilibrium condition in eq. (2.2.5) and the compatibility conditions in eq. (2.2.8) and eq. (2.2.9). The decoupling makes it easy to creatively elaborate with form finding methods and create bespoke elements (like the cable-beam element). This is for example used in the *limit control* form finding, where forces and moments are calculated and scaled for all elements in the same iteration before the nodal velocities and positions are updated. The decoupling also facilitates incorporation of interactive controls to allow real time modification of element stresses, loading and boundary conditions.

One of the main goals with the thesis was to develop a method for beam based form finding, making it is possible to trade between initial and fully optimized form. It was found more difficult then expected, mainly because the enhanced geometry often ends up as a non-smooth, wrinkled and less applicable grid. Consider the structure in Figure 5.0.2. The geometry on the LHS has a greater total length compared to the desired flat option to the right. It won't be possible to obtain that flatter option with beam based form finding because the length of the elements does not easily change. Essentially, the elements can't be shortened when preforming form finding in tension (compare with the classical hanging chain model approach). However this is not a big issue for 2D structures but it does becomes a problem for 3D structures where elements orientated in the extra dimension causes wrinkling to occur.

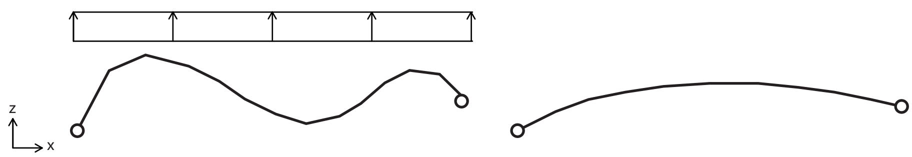


Figure 5.0.2: *The initial structure and the applied form finding load on the left hand side and the desired form on to the right side.*

In an attempt to get round this issue a bespoke cable-beam element (beam that can't take compression) was implemented. It was found helping but not entirely solving the problem. For a structure form found with bars and/or cables, an approach to smoothen out a wrinkled geometry is to apply some prestress in the elements. This is seemingly not a good solution for the beam elements (or cable-beams) since the prestress will impose undesirable moments.

6 Recommendations for further work

Since real time feedback is essential for SMART Form the improvement of speed would be recommended as one of the first priorities for further work. This could be approached in a couple of different ways. Either by looking at more involved time stepping algorithms, or by combining dynamic relaxation with a matrix solver. One could also look at the aspect of multithreaded analysis. Another option would be to implement the 3 DOF beam elements that are presented in the Phd dissertation *Stressed spline structure* by Sigrid Adriaenssens. These elements are much less computationally demanding and could potentially be used for beam based form finding, whereas the more sophisticated 6 DOF elements could be used for sizing or more accurate analysis. Another priority for further work would be to perform more thoroughly benchmarking accounting for combinations of functionality.

References

[1] Sigrid Maria Louis Adriaenssens. "Stressed spline structures". PhD thesis. University of Bath, 2000.

[2] Arup. *Sydney Opera House @ONLINE*. Oct. 2012. URL: http://www.arup.com/Projects/Sydney_Opera_House.aspx.

[3] Michael R. Barnes. Form Finding and Analysis of Tension Structures by Dynamic Relaxation. *International Journal of Space Structures* **14** (1998), 89 –104.

[4] bently. *Generative Design @ONLINE*. Jan. 0. URL: http://www.bentley.com/en-US/Products/GenerativeComponents/.

[5] Horst Berger. Structural form in Architecture. *Structure Magazine* **38** (2007).

[6] Derek Walker Bill Addis. *Happold: The confidence to build*. Taylor and Francis, 1998. ISBN: 978-0419240709.

[7] Marti Broquetas. *From CAD to BIM @ONLINE*. Oct. 2012. URL: http://www.cad-addict.com/2010/02/from-cad-to-bim-part-i-history-of-cad.html.

[8] J.-F. Caron C. Douthe O. Baverel. Form Finding of a grid shell in composite materials. *IASS* **47** (2006).

[9] J.-F. Caron C. Douthe O. Baverel. Form-Finding of a gridshell in composite materials. *Journal of the international association for shell and spatial structures (IASS)* **47** (2006).

[10] Waclaw Zalewski Edward Allen. *Form and forces*. John Wiley and Sons, Inc. Hoboken, New Jersey, 2010. ISBN: 978-0-470-17465-4.

[11] Al Fisher. *Discussions at Buro Happold*. Sept. 2012.

[12] Ariane Ida Fund. "Form-Finding Structures". MA thesis. Massachusetts Institute of Technology, 2008.

[13] British Standards Institution. *Eurocode 3: Design of steel structures*. 2005.

[14] Ola Dahlblom Karl-Gunnar Olsson. *Strukturmekanik, modellering och analys av fackverk*. Studentlitteratur AB, Lund, 2010. ISBN: 978-91-44-06894-7.

[15] Andrew Kroll. *AD Classics: Munich Olympic Stadium / Frei Otto and Gunther Behnisch @ONLINE*. Nov. 2011. URL: http://www.archdaily.com/109136/ad-classics-munich-olympic-stadium-frei-otto-gunther-behnisch/.

[16] Ian Liddell. The engineering of surface stressed structures (?).

[17] McNeel. *developer:rhinocommon - McNeel Wiki*. 2011. URL: http://wiki.mcneel.com/developer/rhinocommon.

[18] *Methods of Analysis for Multi-Storey Frames @ONLINE*. Nov. 2012. URL: http://www.fgg.uni-lj.si/kmk/esdep/master/wg14/l0900.htm.

[19] Karl-Gunnar Olsson. *Private conversation at Chalmers University*. Sept. 2012.

[20] Barb Schmitz. *What a Concept: The Importance of the Early Phase of Design @ONLINE*. Aug. 2011. URL: http://creo.ptc.com/2011/08/05/what-a-concept-the-importance-of-the-early-phase-of-design/.

[21] *Smart Solutions network @ONLINE*. Dec. 2012. URL: http://www.smart-solutions-network.com/group/smartform.

[22] *The New Structuralism @ONLINE*. Nov. 2012. URL: http://thepost-it-effect.blogspot.se/2010/09/new-structuralism-designengineering-and.html.

[23] *The New Structuralism @ONLINE*. Oct. 2012. URL: http://materialpraxis.wordpress.com/2010/09/30/soap-film-frei-otto/s6-1/.

[24] Joseph Schwartz Toni Kotnik. The architecture of Heinz Isler. *Journal of the international association for shell and spatial structures* **52** (2011), 185 –190.

[25] Michela Turrin, Peter von Buelow, and Rudi Stouffs. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Advanced Engineering Informatics* (2011), 656–675.

[26] Matt Tyrnauer. *Architecture in the Age of Gehry @ONLINE*. Oct. 2012. URL: http://www.vanityfair.com/culture/features/2010/08/architecture-survey-201008?currentPage=all.

[27] *University of Exeter Forum @ONLINE*. Nov. 2012. URL: http://www.wilkinsoneyre.com/projects/university-of-exeter-forum-project.aspx?category=education.

[28] *University of Exeter Forum @ONLINE*. Nov. 2012. URL: http://www.burohappold.com/projects/project/university-of-exeter-forum-7/.

[29] Unkown. *Zaha Hadid Biography @ONLINE*. Oct. 2012. URL: http://www.notablebiographies.com/supp/Supplement-Fl-Ka/Hadid-Zaha.html.

[30] Eric Weisstein. *Euler forward method @ONLINE*. Nov. 2012. URL: http://mathworld.wolfram.com/EulerForwardMethod.html.

[31] Chris J.K. Williams. The analytical and numerical definition of the geometry of the british museum great court roof. *Mathematics and Design* (2001), 434 –440.

[32] Y.M. Xie et al. Form finding for complex structures using evolutionary structural optimization method. *Design Studies* **26**.1 (2005), 55 –72. ISSN: 0142-694X. DOI: 10.1016/j.destud.2004.04.001. URL: http://www.sciencedirect.com/science/article/pii/S0142694X0400050X.

A Appendix A

A.1 Derivation of stability condition for rotations

Here an attempt to derive a relationship between time step, stiffness and moment of inertia that ensures a stable solution is presented. This derivation is based on a similar derivation for the relationship between time step, stiffness and mass presented in [3].

Consider the case with a group of nodes lying on a line, free to rotate around the z-axis. Let S be the rotational stiffness for a node, and let this stiffness be identical for all the nodes in the group. T is the out-of-balance torque applied on the node. Consider nodes i and j in the group rotating relative a fixed coordinate system with the velocities ω_i and ω_j. The moment of inertia is identical for each node and set to J.

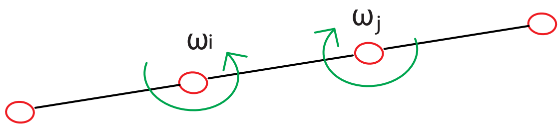


Figure A.1.1: *Rotational velocity of to adjacent nodes i and j.*

Starting from eq. (2.2.18), neglecting the damping C

$$\omega_i^{(t+\Delta t)} = \omega_i^t + \frac{\Delta t}{J} H_i^t. \tag{A.1.1}$$

For the next time interval

$$\omega_i^{(t+2\Delta t)} = \omega_i^{(t+\Delta t)} + \frac{\Delta t}{J} H_i^{(t+\Delta t)}. \tag{A.1.2}$$

Where $H_i^{(t+\Delta t)}$ is calculate from the rotations from the previous iteration times the rotational stiffness, see eq. (2.2.15)

$$H_i^{(t+\Delta t)} = H_i^t - S\Delta t(4\omega_i + 2\omega_j)^t. \tag{A.1.3}$$

Where

$$S = \frac{EI}{L}. \tag{A.1.4}$$

Inserting eq. (A.1.3) into eq. (A.1.2)

$$\omega_i^{(t+2\Delta t)} = \omega_i^{(t+\Delta t)} + \frac{\Delta t}{J}[H_i^t - S\Delta t(4\omega_i + 2\omega_j)^t], \tag{A.1.5}$$

is obtained. H_i^t can be extracted from eq. (A.1.1) giving

$$H_i^t = \frac{J}{\Delta t}(\omega_i^{(t+\Delta t)} - \omega_i^t). \tag{A.1.6}$$

Inserting eq. (A.1.6) into eq. (A.1.5) in order to eliminate H_i^t

$$\omega_i^{(t+2\Delta t)} = \omega_i^{(t+\Delta t)} + \omega_i^{(t+\Delta t)} - \omega_i^t + \frac{\Delta t}{J}[S\Delta t(4\omega_i + 2\omega_j)^t]. \tag{A.1.7}$$

Which is then rewritten to

$$\omega_i^{(t+2\Delta t)} - 2\omega_i^{(t+\Delta t)} + \omega_i^t + = \frac{2\Delta t^2 S}{J}[(2\omega_i + \omega_j)^t]. \tag{A.1.8}$$

In direct analogy the same yields for adjacent node j

$$\omega_j^{(t+2\Delta t)} - 2\omega_j^{(t+\Delta t)} + \omega_j^t + = \frac{2\Delta t^2 S}{J}[(2\omega_j + \omega_i)^t]. \tag{A.1.9}$$

Let

$$\omega_{ij} = \omega_i - \omega_j, \tag{A.1.10}$$

and by subtracting eq. (A.1.9) from eq. (A.1.8) we get

$$\omega_{ij}^{(t+2\Delta t)} - 2\omega_{ij}^{(t+\Delta t)} + \omega_{ij}^t = \frac{2\Delta t^2 S}{J}\omega_{ij}^t. \tag{A.1.11}$$

Where ω_{ij} is the rotational velocity of node i relative node j. The limiting case for stability is when the velocity relative an adjacent node is equal to the opposite of the velocity for the same nodes in the previous iteration. Meaning that

$$\omega_{ij}^{(t+2\Delta t)} = -\omega_{ij}^{(t+\Delta t)} = \omega_{ij}^t. \tag{A.1.12}$$

Making use of this conclusion eq. (A.1.11) can be simplified into

$$4\omega_{ij}^t = \frac{2\Delta t^2 S}{J}\omega_{ij}^t, \tag{A.1.13}$$

which can be further simplified to

$$\Delta t = \sqrt{\frac{2J}{S}}. \tag{A.1.14}$$

Here

S is the rotational stiffness,

J is the moment of inertia,

Δt is the time step.