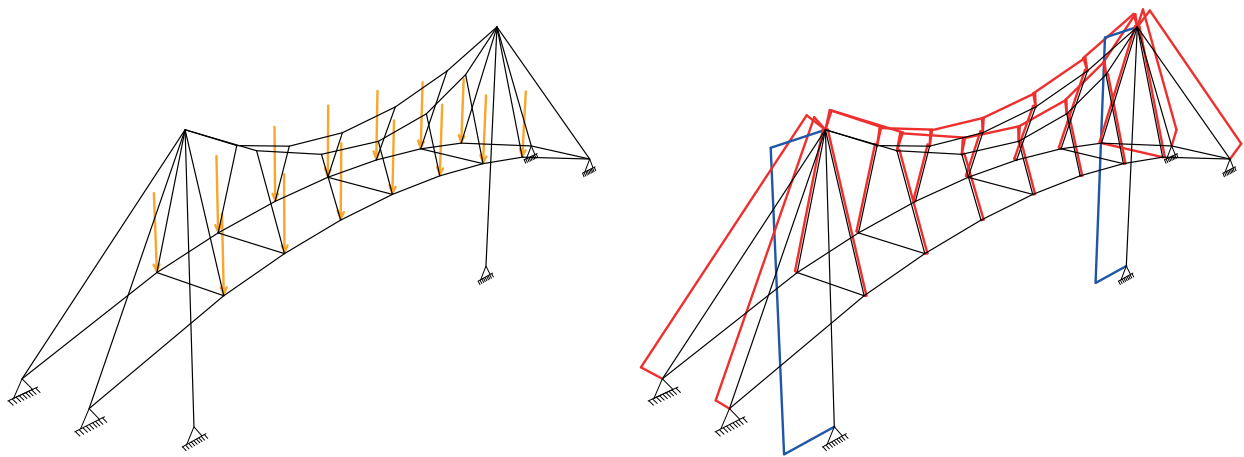




**CHALMERS**  
UNIVERSITY OF TECHNOLOGY

---



# A Computational Tool for the Design of Prestressed Cable Supported Structures

Master Thesis in Structural Engineering and Building Technology

Ahmad Abdul Sater  
Oskar Thor



MASTER'S THESIS 2020

# A Computational Tool for the Design of Prestressed Cable Supported Structures

AHMAD ABDUL SATER  
OSKAR THOR



**CHALMERS**  
UNIVERSITY OF TECHNOLOGY

Department of Architecture and Civil Engineering  
*Division of Architectural Theory and Methods*  
Architecture and Engineering Research Group  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Gothenburg, Sweden 2020

A Computational Tool for the Design of Prestressed Cable Supported Structures  
AHMAD ABDUL SATER  
OSKAR THOR

© AHMAD ABDUL SATER, OSKAR THOR, 2020.

Supervisor: Alexander Sehlström, Architecture and Engineering Research Group  
Examiner: Mats Ander, Department of Architecture and Civil Engineering

Master's Thesis 2020  
Department of Architecture and Civil Engineering  
Division of Architectural Theory and Methods  
Architecture and Engineering Research Group  
Chalmers University of Technology  
SE-412 96 Gothenburg  
Telephone +46 31 772 1000

Cover: Audi A 50 company restaurant, modelled and analyzed in the developed tool.

Typeset in L<sup>A</sup>T<sub>E</sub>X  
Gothenburg, Sweden 2020



## Abstract

Cable supported structures are light-weight structures gaining their load-bearing capacity through the use of one or several cables in combination with other structural members such as struts, beams, and plates. Cables transfer loads in tension only and adapt their shape depending on the loading condition. The relationship between applied loads and deformations is therefore highly non-linear and consequently the geometrical shape of cables requires bespoke design approaches.

Traditionally, architects and engineers have designed cable structures using physical models or with graphical methods. This has changed as advances in numerical analysis and computational engineering have been made over the recent century. Many software programs available today can be used for analysis. However, a key part of structural design requires the ability to "sketch" and "iterate", which is absent in many available packages.

A computational design tool is developed with the aim to support the design in early design stages. After evaluating three existing numerical methods for large displacement structures, a solver with Green strain bar elements is implemented. The tool extends the open-source BHoM code library. Output data from the tool is benchmarked by comparing results with analytical and numerical models.

By exposing the functionality of the tool to user-interfaces such as Grasshopper and Excel, users can explore alternative design proposals by changing a set of parameters. These parameters include the input geometry, initial prestress, section sizes and topology.

Case-studies are used to highlight how this tool is used in Grasshopper. A set of design-objectives are defined and met by controlling a set of parameters, subsequently creating several designs proposals. For the studied cases, which display small strains and large displacements, the tool provides accurate results. Should a different analysis model be required, the tool is suited for modifications and extensions of its capabilities.

Keywords: cable-supported structures, form-finding, prestress, geometrical stiffness, structural design, parametric design.



# Acknowledgements

Firstly, we would like to thank our supervisor Lic. Eng. Alexander Sehlström for his advice and guidance. It has been a pleasure to get an insight into his research. We are also thankful for the never-ending enthusiasm and commitment of our examiner, Senior lecturer, Dr. Mats Ander. Our online meetings have been a great joy during this spring which otherwise would have been defined by the ongoing COVID pandemic.

A great thanks to Isak Näslund and Dr. Al Fisher at BuroHappold, you have been most helpful in teaching us about computational programming.

We are further grateful for the great input and all the valuable conversations received from our opponents, Hanna Brynhildsen and Geir Söderin.

We also want to thank Jonas Gabrielsson at Sweco Structures, who has provided us with seating and necessary tools to complete this thesis.

Finally, we want to show our gratitude for the support and encouragement we have received during our studies at Chalmers University of Technology. A special thanks to Prof. Karl-Gunnar Olsson and Prof. Morten Lund for their inspiration and commitment to us as students.



Ahmad Abdul Sater and Oskar Thor, Gothenburg, June 2020



# Contents

<b>List of Figures</b>	<b>xi</b>
<b>List of Tables</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Purpose . . . . .	3
1.3 Method . . . . .	4
1.4 Limitations . . . . .	5
1.5 Structure of the Thesis . . . . .	5
<b>2 Cable Supported Structures</b>	<b>7</b>
2.1 A Historic Review: From Hemp Ropes to Wire Cables . . . . .	7
2.2 Modern Cable Types . . . . .	11
2.3 Characteristics of a Cable . . . . .	11
2.4 Form-Finding . . . . .	13
2.5 Prestress . . . . .	14
<b>3 Numerical Methods</b>	<b>17</b>
3.1 Dynamic Relaxation . . . . .	17
3.2 Force Density Method . . . . .	21
3.3 Finite Element Method . . . . .	25
<b>4 Evaluation of Numerical Methods</b>	<b>29</b>
4.1 Example 1: Form-Finding of Cable . . . . .	30
4.2 Example 2: Analysis of Truss . . . . .	31
4.3 Example 3: Truss with Diagonal Ties . . . . .	32
4.4 Conclusion . . . . .	34
<b>5 Benchmark</b>	<b>35</b>
5.1 Snap-Through . . . . .	35
5.2 Hardening . . . . .	36
5.3 Snap-back . . . . .	39
5.4 Conclusion . . . . .	40
<b>6 Implementation of FEM_Toolkit</b>	<b>41</b>
6.1 Object Oriented Programming in C# . . . . .	41

6.2	Building and Habitats object Model . . . . .	41
6.3	FEM_Toolkit . . . . .	43
6.4	Exposing FEM_Toolkit to User-Interfaces . . . . .	45
<b>7</b>	<b>Case-Study: Analysis of Audi A 50 Roof Structure</b>	<b>47</b>
7.1	Background . . . . .	47
7.2	Aim and Workflow . . . . .	48
7.3	Part 1: 2D-case . . . . .	50
7.4	Part 2: 3D-case . . . . .	54
7.5	Conclusion . . . . .	60
<b>8</b>	<b>Conclusions</b>	<b>61</b>
	<b>Bibliography</b>	<b>63</b>
<b>A</b>	<b>Code Access</b>	<b>I</b>

# List of Figures

1.1	Summary of method. . . . .	4
2.1	A Chinese and a Tibetan type bridge. . . . .	8
2.2	Early iron chain constructions. . . . .	8
2.3	Menai Suspension Bridge. . . . .	9
2.4	Membrane roof and lattice membrane by Vladimir Shukhov. . . . .	10
2.5	Olympiastadion in Munich. . . . .	10
2.6	Four modern cable types. parallel wires (left), parallel strands, locked coil strand, wire rope. . . . .	11
2.7	Initial configuration of cable study. . . . .	12
2.8	Experimental setup. . . . .	12
2.9	Comparison between results from small and large displacement theory. . . . .	13
2.10	Traditional form-finding methods. . . . .	14
2.11	Three approaches to prestressing. . . . .	15
2.12	The effect of prestress on joint equilibrium. . . . .	16
3.1	Dynamic relaxation scheme. . . . .	19
3.2	Force density method scheme. . . . .	22
3.3	Node equilibrium principle . . . . .	23
3.4	Non-linear finite element method scheme. . . . .	27
4.1	Simple cable inspired by (Luo & Lu, 2006). . . . .	30
4.2	Truss inspired by (Olsson & Dahlblom, 2016). . . . .	31
4.3	Truss with diagonal ties inspired by (Guest, 2006). . . . .	32
4.4	Effect of large loads on displacements. . . . .	33
5.1	Input for snap-through benchmark (Crisfield, 1993). . . . .	35
5.2	Load-displacement curves for benchmark 1. . . . .	36
5.3	Input for hardening benchmark (Crisfield, 1993). . . . .	37
5.4	Load-displacement curve for benchmark 2, $P-\Delta u$ . . . . .	37
5.5	Load $P$ -displacement curve for benchmark 2, $P-\Delta v$ . . . . .	38
5.6	Load-displacement plot for benchmark 2, $P-\Delta u-\Delta v$ . . . . .	38
5.7	Input for snap-back benchmark (Crisfield, 1993). . . . .	39
5.8	Load-displacement curve for benchmark 3, $P\Delta u_2$ . . . . .	39
6.1	Software data exchange strategies. . . . .	42
6.2	The 4 categories of the BHoM-core. . . . .	43
6.3	Exposure of FEM_Toolkit in Grasshopper and Excel. . . . .	45

7.1	Audi A50 restaurant extension. . . . .	47
7.2	Workflow inside Grasshopper. . . . .	48
7.3	Workflow scheme. . . . .	49
7.4	Form-finding of cable. . . . .	50
7.5	Geometry after form-finding. . . . .	50
7.6	Creation of steel material and circular cross-section. . . . .	51
7.7	Displacement without prestress as displayed by FEM_Toolkit. . . . .	51
7.8	Application of prestress. . . . .	52
7.9	Definition of prestress. . . . .	52
7.10	Displacements with prestress displayed by FEM_Toolkit. . . . .	53
7.11	Normal forces with prestress displayed by FEM_Toolkit. . . . .	53
7.12	Normal forces with prestress displayed in Grasshopper. . . . .	53
7.13	3D-structural model. . . . .	54
7.14	Definition of objective and variables in Opossum. . . . .	55
7.15	Displacements (left) and normal forces (right) at different iteration steps . . . . .	56
7.16	Best set of variables found by Opossum after 300 iterations. . . . .	57
7.17	The two studied load-cases. . . . .	57
7.18	Stress distribution for the two load-cases. . . . .	58
7.19	Three sets of prestress variables found with Opossum [N]. . . . .	58
7.20	Deformations for LC1 (orange) and LC2 (cyan) with variables from set 1. . . . .	59
7.21	Deformations for LC1 (orange) and LC2 (cyan) with variables from set 2. . . . .	59
7.22	Deformations for LC1 (orange) and LC2 (cyan) with variables from set 3. . . . .	59



# List of Tables

2.1	Comparison of steel qualities (Pipinato, 2015). . . . .	11
4.1	Evaluation examples. . . . .	29
4.2	Comparison of displacement and normal force for example 1. . . . .	30
4.3	Comparison of displacements and normal forces for example 2. . . . .	31
4.4	Comparison of displacements and normal forces for example 3, $P = 0.9 \text{ MN}$ , $\epsilon_{max} = 2\%$ . . . . .	33
4.5	Comparison of displacements and normal forces for example 3, $P = 9 \text{ MN}$ , $\epsilon_{max} = 10\%$ . . . . .	34
6.1	FEM_Toolkit classes . . . . .	44
7.1	Results at different iteration step from Opossum . . . . .	55
7.2	Comparison of set 1-3, based on sum of the arc's nodal displacements. . . . .	60



# 1

## Introduction

Cable supported structures are light-weight structures gaining their load-bearing capacity through the use of one or several cables in combination with other structural members such as struts, beams, and plates. Cables transfer loads in tension only and adapt their shape depending on the loading condition. Therefore, the relationship between applied loads and deformations is highly non-linear and consequently the geometrical shape of cables requires bespoke design approaches.

This chapter will discuss the advantages of cable supported structures and the importance of conceptual design. Moreover, the benefits of computational approaches, which allow for streamlined workflows, are outlined. This will support the formulation of the purpose and render in guiding questions for thesis.

### 1.1 Background

One way to understand the structural behaviour of cables is to observe how a simple cable reacts to external loads. Imagine a chain suspended between two equally high points and a heavy weight is hung at the middle of the chain. With some intuition, one can conclude that if the weight would be removed the shape of the chain changes. In other words, one can state that there is a relationship between force and form.

Structures that adapts geometries depending on the applied forces are sometimes referred to as ‘form-active’ systems (Engel, 1997). Traditionally, architects and engineers designed cable structures using physical models or with graphical methods. Over the last century, advances has been made in numerical analysis methods, a development accelerated by the introduction of computers. Today, it is common practice to use numerical methods such as the force density method, dynamic relaxation and geometrical non-linear finite element analysis for the design and analysis of cable supported structure to form-find and analyze structures with large displacements.

Cable supported structures exist in many shapes and could refer to different structural systems. In the built environment they can be found in suspension bridges and suspended roof structures. They are distinguished from other structures by the use of one or several cables acting as primary bearing. Cables can be also be combined with other members and thus forming hybrid systems (Engel, 1997). They are suitable where large spans are required because of their light-weight character

and their efficient way of carrying external loads.

Lightness in cable structures is partially achievable due to the high strengths of modern cables but also due to the use of prestress. Cables can be prestressed to increase the geometrical stiffness of a system as well as to control displacements, something the engineer can use to meet predefined design and/or displacement criteria.

The light-weight nature of cable supported structures makes them sustainable based on three perspectives: an *ecological perspective*; a *social perspective*; and, a *cultural perspective* (J. Schlaich, 2004). The low use of material reduces the *ecological* impact. Scarcity of construction material is an increasing concern and it is desirable to strive for more material efficiency in structures. It is, however, important to remember that large prestress in cable structures could contribute to heavy foundations and may counter the material savings in a slender construction. From the *social perspective*, the challenging nature of a design and construction of cable supported structures, encourages the development of skilled labor. This leads to more advanced jobs and an increased need of educated engineers. Lastly, spectacular designs and the sensation of lightness which are possible to achieve with cable structures contribute to the *cultural perspective*.

### Conceptual Design

Conceptual design is primarily used in early design stages and is often characterized by the development of several conceptual proposals. During this design stage, objective design criteria are not always well defined and a creative sketching process is deployed to narrow down several alternatives until a final design proposal emerges. Unfortunately, it is common in practice and education that the conceptual phase is left to architects alone, missing on the opportunity for structural engineers to provide valuable input (Llunji, 2020). This risks excluding engineers from the many important decisions that are made in the conceptual phase which could lead to irreversible bad choices and inevitable expensive solutions (Hsu, 2000).

In design of cable structures, the collaboration between the architect and the engineer needs to be well functioning, as form and forces are highly intertwined. The role of the engineer must not be excluded to sizing and detailing only, but should also include the conception of a design proposal. In addition to their theoretical knowledge, the engineer must utilize creative talent to help achieve a vision (M. Schlaich, 2006).

### Computational Design

Today there is a great variety of computational engineering tools suitable for solving complex problems with high resolution. A major disadvantage of many engineering software, is the requirement of precise input data as well as geometry which makes them less suitable for conceptual design (Wang, 2001). They have a tendency to provide much more results than needed in conceptual design and might disturb the creative process.

Many contemporary software are usually tailored either for architects or engineers and complicates sharing of necessary data. This hinders the collaboration between architects and engineers and may result in diverging conceptual ideas.

By working and sharing similar design platforms the issues above could be avoided. It allows for a streamlined design process from early design to a finished proposal, and earlier proposals can be resummoned. It is also possible to create complex parametric models allowing for fast updates and modifications on a certain proposal. Moreover, with visual programming it is easy to receive instant visual feedback from structural analysis. Common parametric platforms are Grasshopper and Dynamo which are extension to the CAD-environments Rhino and Revit respectively. Visual scripts and computational programs, written for specific projects, can be reused and the acquired knowledge can be shared for the benefit of future work. Data and functions can be stored, shared and further developed by others. It worth mentioning that successful and sustainable knowledge sharing requires a high-level of modular code architecture.

## 1.2 Purpose

The main purpose of this research is to develop a computational design tool for cable supported structures that can be used in conceptual design phases. The tool should be based on a convenient numerical analysis method which includes geometrical non-linear effects. To facilitate for cross-disciplinary collaboration between architects and engineers, the tool must be exposed to a CAD-environment such as Rhino/Grasshopper.

An important part of the thesis is to illustrate how the tool can be used for exploration of concepts. This means that "project" case specific workflows are developed to achieve a set of design-goals. All parameters that affect the design and performance of a structure will be implemented in the tool. By exposing as many parameters as possible to the user, the aim is to facilitate for the creation of many proposals.

Based on the outlined purpose, the objectives are to:

- create a design tool exposed to CAD-environments which can be used in conceptual design;
- study and evaluate numerical methods to select the most promising one;
- program the tool in a modular fashion to allow for future extensions and modifications;
- show how different parametric workflows can be used to achieve different design goals with the developed tool.

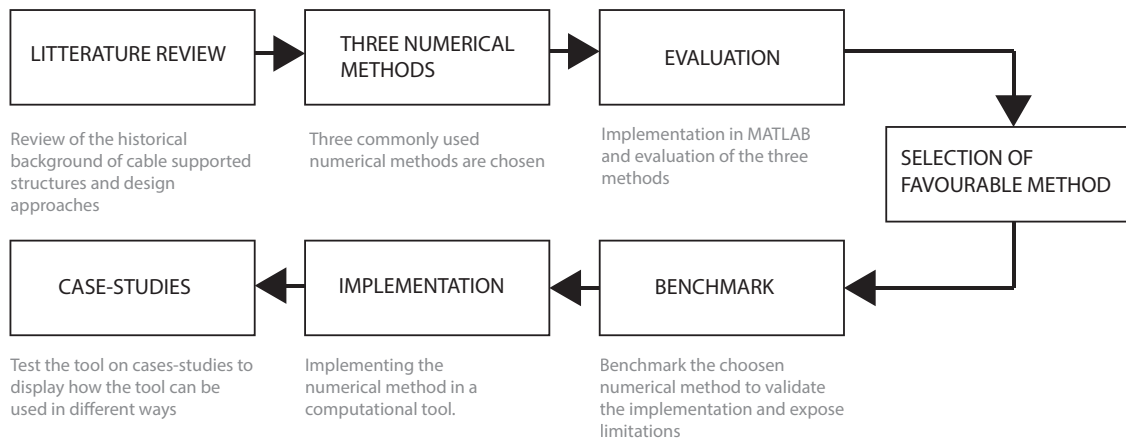
### 1.3 Method

The working method is illustrated in figure 1.1 and takes off with an overview of the history of cables, historic cable structures and traditional form-finding methods. A simple cable is studied to understand the characteristic behaviour of a cable and the necessary theory needed for large displacements.

Three known numerical methods used to design cable structures are studied and tested on three examples in MATLAB. This investigation will contribute to the understanding of the methods and test the suitability of each method for each example. The most promising method will be selected and implemented in a computational tool. To ensure that the tool returns reliable results, the chosen method will be compared with benchmark problems in the literature.

The design tool will be programmed in Microsoft Visual Studio. Available open source libraries will be utilized to perform numerical calculations and to expose the tool to user-interfaces used by many engineers such as Grasshopper and Excel. Object oriented programming in Visual Studio forms a good framework for future development and enables for sharing and modification by many users.

Case-studies will be performed to show how the tool can be used. A suitable workflow for each case-study will be presented and implemented, allowing a set of design-goals to be met. These workflows could include form-finding or to use a number of parameters for form-setting.



**Figure 1.1:** Summary of method.

## 1.4 Limitations

The main focus lies within the conceptual design of cable supported structures. The requirement to find possible equilibrium states could therefore be achieved quicker without considering other non-linearities such as material and boundary condition. The consideration of material and boundary-condition non-linearities can be accounted for in future studies.

Time-dependant effects such as creep, corrosion, fatigue and prestress-relaxation are important factors but are outside the scope of this thesis. This means that detailed analysis is performed in later stages. Likewise for extensive member checks which are of importance for the final design, but nevertheless a time consuming procedure and would not directly contribute to the conceptual design phase.

There are many numerical methods that have been developed for analysing geometric non-linear structures. In this thesis only three methods will be considered for the following reasons: They are all well established numerical methods; and they have been used in practice; many later methods are extensions of these methods.

Only 1D-continuum elements will be studied and implemented. This includes discretized line members such as tension only, compression only, and axial bars. Implementation of beam elements will not be made at this stage. It is however possible to extend the implemented tool to support analysis and form-finding of more complex structures.

Cable structures in roofs and bridges are exposed to dynamic loads and it is often important to check the dynamic behaviour. However this will not be considered as this lies outside the scope of the main focus of the thesis which is to implement a computational tool for conceptual design.

## 1.5 Structure of the Thesis

The report is divided into 8 chapters and structured to mirror the evolution of the project.

Chapter 2 provides a historical background to cable supported structures and explains the non-linear behaviour of cables.

Chapter 3 presents the theory of three numerical methods that can be used to design cable supported structures.

In chapter 4, the three numerical methods are evaluated in a comparative study consisting of three simple structures. Consequently, the most promising numerical method is selected and further studied.

In chapter 5, the selected numerical method is benchmarked against problems in literature in order to check the reliability of its results. Moreover, the limitations of the chosen numerical method are presented and discussed.

In chapter 6, the implementation of a computational tool as a plug-in to common engineering user-interfaces is described.

Chapter 7 presents case-studies which highlight how the tool can be used to design and optimize cable supported structures.

Chapter 8 marks the end of the report and discusses important outcomes and potential further developments.



# 2

## Cable Supported Structures

This chapter address some key aspects of cable structure design such as form-finding and prestressing. It will start with a historical background before presenting many of the challenges of designing cables supported structures caused by the inherited non-linear behaviour of cables.

### 2.1 A Historic Review: From Hemp Ropes to Wire Cables

Before wire cables were invented and used in cable structures, humans used traditional materials such as cane, bamboo, hemp rope or iron chains (Peters, 2012). According to Peters, the earliest known suspension bridges originated from the eastern Himalayas with records dating back to 285 BCE of a catenary shaped bridge in Sichuan.

Early bridges in Himalaya were either of a Tibetan- or a Chinese type. The Chinese bridges, e.g. Anlan bridge in figure 2.1a, is characterized by the simple prestress system created by anchoring the end winches to a gatehouse. This allowed the cables to be prestressed and thereby limit the sag. Tibetan bridges did not have the same refined anchoring system. Figure 2.1b, indicates that cables in Tibetan bridges were constructed without any prestress which lead to larger sag.

The deterioration of natural material such as hemp, cane and bamboo created a desire of more durable alternatives which eventually led to the development of wrought iron chains. An early example of an iron chain can found in the French town of Moustier Sainte-Marie, seen in figure 2.2b. The chain is suspended 200 m between two rocks and supports a hanging gold-painted star. The chain dates back to 13th century and shows that practical metallurgical knowledge existed in Europe at that time. (Peters, 2012). It is however not until 1741, the Wynch Bridge in England was built as the first documented chain bridge in Europe. One of oldest remaining chain is Luding Bridge in Tibet, which was built in 1703 and can be seen in figure 2.2a.

## 2. Cable Supported Structures

---



(a) Anlan Bridge.



(b) Unknown Tibetan bridge (Ward, 1921).

**Figure 2.1:** A Chinese and a Tibetan type bridge.

Source: *Prince Roy (2007)*; image distributed under CC BY 2.0 license;  
[https://commons.wikimedia.org/wiki/File:Anlan\\_Suspension\\_Bridge-1.jpg](https://commons.wikimedia.org/wiki/File:Anlan_Suspension_Bridge-1.jpg).



(a) Luding Bridge.



(b) Suspended iron chain in Moustier Saint-Marie.

**Figure 2.2:** Early iron chain constructions.

Source: *gill\_penney (2007)*; image distributed under CC BY 2.0 license; cropped and saturated; <https://www.flickr.com/photos/gillpenney/2215347512/>.  
*Ced76000 (2015)*; image distributed under CC BY-SA 4.0 license; cropped and saturated; [https://commons.wikimedia.org/wiki/File:Moustiers\\_Sainte\\_Marie\\_-\\_Etoile.JPG](https://commons.wikimedia.org/wiki/File:Moustiers_Sainte_Marie_-_Etoile.JPG).

Tomas Telford is one of the great contributors to the development of modern suspension bridges. Telford developed a chain of flat eyebar system with punched holes in the beginning of the 19th century (Peters, 2012). Tests of the eyebar system showed large stretches at high tensile stresses and Telford decided that stresses in his chains should never exceed a third of its ultimate strength (Timoshenko, 1983). This is similar to the limitations of modern cable design codes. The flat eyebar system was

used in 1826 for the construction of Menai Suspension Bridge, illustrated in figure 2.3.



**Figure 2.3:** Menai Suspension Bridge.

Source: *Mick Knapton (2004)*; image distributed under CC BY-SA 3.0 license;  
[https://commons.wikimedia.org/wiki/File:Menai\\_Suspension\\_Bridge.jpg](https://commons.wikimedia.org/wiki/File:Menai_Suspension_Bridge.jpg).

Telford's flat eyebar system was the standard for suspension systems until the first half of the 19th century when a major breakthrough came by the invention of wire ropes. Wire ropes were cold formed or in other words pulled which gave a higher tensile strength due to the stress-hardening effects. One of the early manufacturers of wire ropes was John Augustus Roebling. Roebling established his wire company in 1841 and successfully used wire cables in his own bridge projects, including Brooklyn Bridge finished in 1883 (Addis, 1990).

In 1895, Vladimir Shukhov designed the first cable-net and membrane structure for Nizhny Novgorod exhibition. The structure, which is shown in figure 2.4, was the first time cable-nets were used in buildings. Shukov's work came to inspire generations of architects and engineers alike.

Frei Otto is one of early architects who contributed to the development of lightweight structures set out by Shukhov. Otto designed one of the most influential cable structures in more modern times, the Olympiastadion in Munich visible in figure 2.5. The tent-like stadium was built in 1972 as part of the preparation for the Olympic games the same year. The stadium was designed together with the engineer Jürg Schlaich whose engineering practice is still a leading firm in the design of lightweight- and cable structures.



## 2. Cable Supported Structures

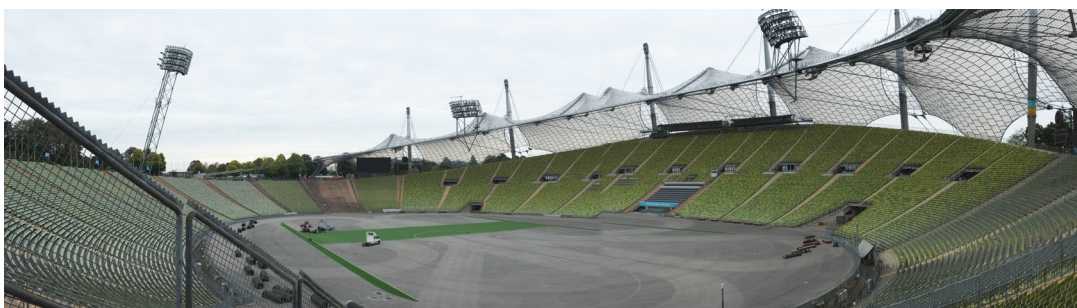
---

Today, cable supported structures are often used whenever large spans are desired which involves everything from suspension bridges to roofs structures. A growing sector is arena stadiums and many contemporary stadium roofs are constructed with cable systems with high prestress.



**Figure 2.4:** Membrane roof and lattice membrane by Vladimir Shukhov.

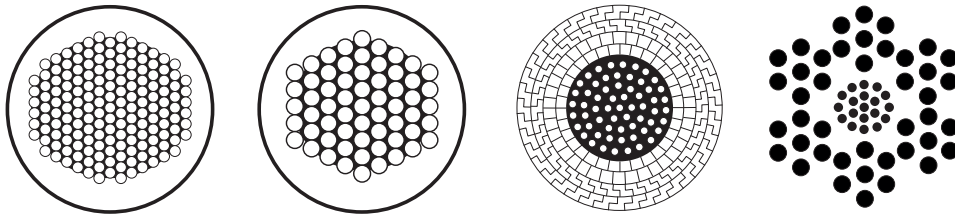
Source: *Andrei Osipovich Karelin (1895); image is in public domain;*  
*[https://commons.wikimedia.org/wiki/File:](https://commons.wikimedia.org/wiki/File:Membrane_Roof_and_Lattice_Shell_of_Shukhov_Rotunda_1895.jpg)*  
*[Membrane\\_Roof\\_and\\_Lattice\\_Shell\\_of\\_Shukhov\\_Rotunda\\_1895.jpg](https://commons.wikimedia.org/wiki/File:Membrane_Roof_and_Lattice_Shell_of_Shukhov_Rotunda_1895.jpg).*



**Figure 2.5:** Olympiastadion in Munich.

## 2.2 Modern Cable Types

Modern cables are still a combination of cold-worked steel wires. The wires are either helically twisted around a straight wire into strands or used as parallel wires. The wrapping of wires and/or strands increase the flexibility of the cable but leads to a loss of tensile strength. Hence, parallel wires are used when maximum strength is desired. Strands are twisted with every layer twisted in the opposite direction than the previous to equal the torsional stiffness. In figure 2.6 some typical cross-section of modern cables are presented. Modern cables are often protected through envelopes or through a tight outer layer constructed of z-shaped wire profiles as in locked coil strands.



**Figure 2.6:** Four modern cable types. parallel wires (left), parallel strands, locked coil strand, wire rope.

The major improvement of cables is due to the accelerated development of steel qualities which allows modern cables to carry very high tensile stresses. In table 2.1 some material properties of different steel qualities are compared.

**Table 2.1:** Comparison of steel qualities (Pipinato, 2015).

	Unit	Cable	Mild	High Str.
<b>Yield stress</b>	MPa	1180	240	690
<b>Tensile strength</b>	MPa	1570	370	790
<b>Breaking strain</b>	%	4	24	
<b>Modulus of elasticity</b>	GPa	205	210	210

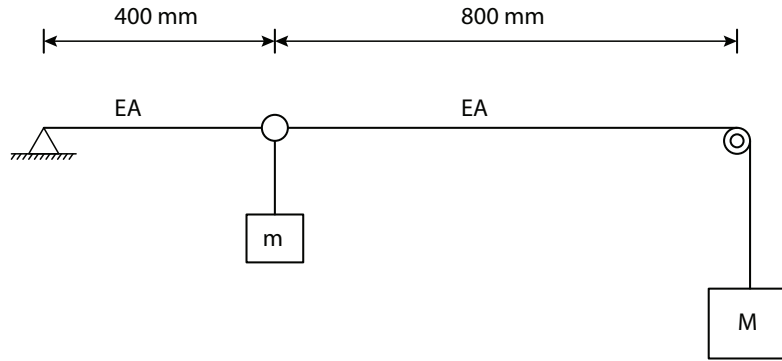
Relaxation is a major concern of modern cables, why allowed stresses still are limited to stresses well below the maximum capacity (CEN, 2006).

## 2.3 Characteristics of a Cable

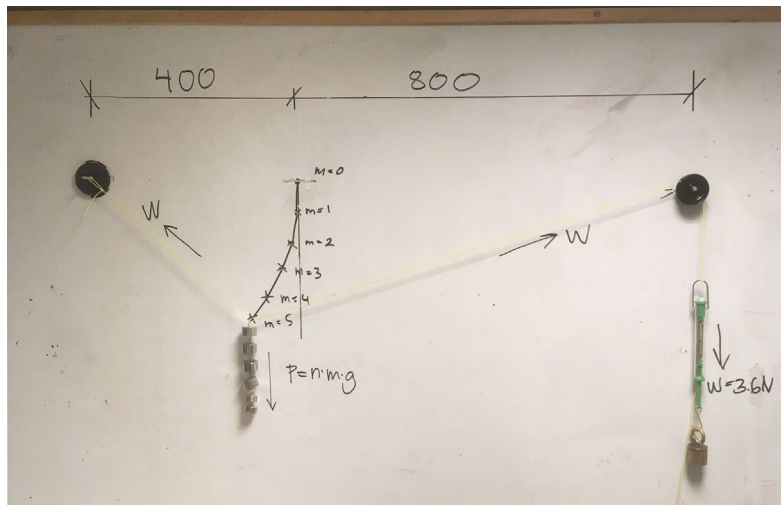
Cables are essentially limited to axial tensile stresses and the shear-, bending-compression- and torsional stiffness are usually negligible. Since the only mode of load transfer is in tension this leads to large displacements and thus geometric non-linearity has to be considered in analysis.

## 2. Cable Supported Structures

A simple example is studied to illustrate the necessity of large displacement theory in the analysis. Figure 2.7 illustrates a cable loaded with a mass  $m$ . The cable is pinned on the left side and supported with a roll-bearing on the right side. A counter-weight of mass  $M$  is connected to the right end of the cable. The setup for the experiment is shown in figure 2.8



**Figure 2.7:** Initial configuration of cable study.

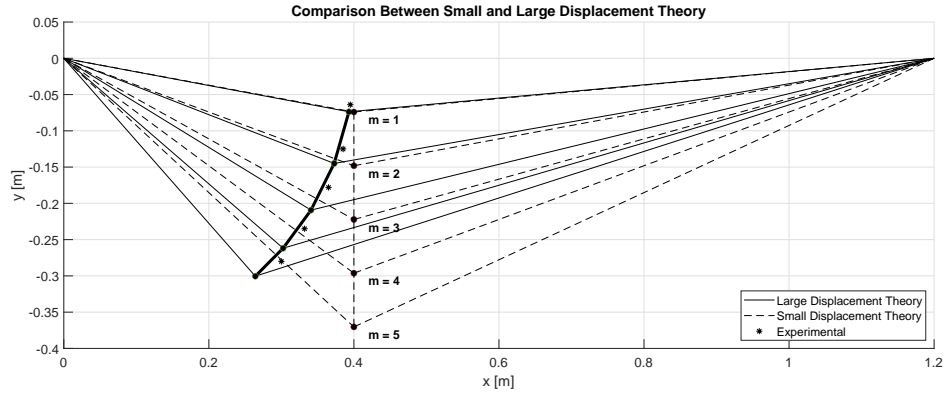


**Figure 2.8:** Experimental setup.

By intuition it is understood that an increase of the counterweight  $M$  reduces the deflection of  $m$ . Correspondingly, a decrease of  $M$  would lead to the opposite. This may be interpreted as if the axial force affects the stiffness of the system and demonstrates why the geometric stiffness should be included in the equations of a geometric non-linear calculation models.

Figure 2.9 shows a comparison of two analyses assuming small and large displacements. Small displacement theory assumes small rotation angles of the initial geometry. Due to the initial straight element and the assumption of small rotation angles the equilibrium equations will neglect asymmetric displacements and consequently following a solution path vertically. In the same figure, it is shown that this

assumption becomes poor as the displacement increase. This showcases the need of an analysis which supports large displacements/rotations. This can be achieved by an iterative calculation were the stiffness of the system is based on an updated geometry.



**Figure 2.9:** Comparison between results from small and large displacement theory.

In the analysis presented, the left hand side of the cable was assumed inextensible and as a consequence the displacement follows the path of an arc with constant radius. The experimental results are recurrently shifted to the right of the analysis result. This indicates that some strains were present in the experiment in figure 2.8.

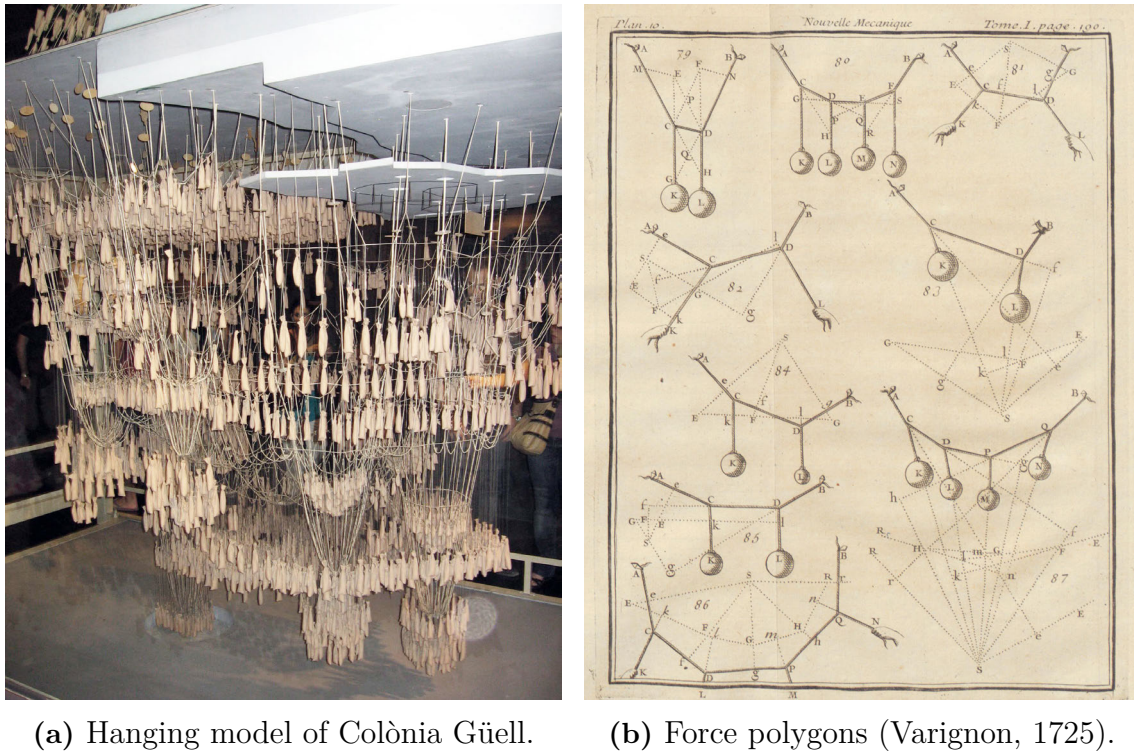
## 2.4 Form-Finding

The aim of form-finding cable supported structures is to derive a shape in equilibrium based on a set of constraints and applied loads. The form-found geometry is often used as an initial shape in further analysis to ensure convergence.

The earliest types of form-finding methods involved the use of physical models. A pioneer in this field was Antoni Gaudi, who used extensive chain-models in his design work. Gaudi used wires to find funicular forms which were inverted into compression arches as shown in figure 2.10a. The method of a using physical model was also adopted by Frei Otto and Heinz Isler.

Another approach to form-finding is Graphic statics. Graphic statics is a graphical method to determine member forces or to find the equilibrium form from a set of applied forces. The geometrical reciprocal relationship between the geometry of the structure and the internal and external forces are used in graphic statics. This relationship can be visually represented by a force and form diagram. If the geometry of a structure is changed it is reflected in the force diagram and vice versa. Because of this duality, graphic statics can used for form-finding as well as for analysis of a system.





(a) Hanging model of Colònia Güell.

(b) Force polygons (Varignon, 1725).

**Figure 2.10:** Traditional form-finding methods.

Source: *Canaan (2009)*; image distributed under CC BY-SA 4.0 license;  
[https://commons.wikimedia.org/wiki/File:Maqueta\\_funicular.jpg](https://commons.wikimedia.org/wiki/File:Maqueta_funicular.jpg).

The first book about graphic statics, *Die graphische statik*, was written by Culmann in 1866. Although his book was a major contribution to the subject, the theory of force polygons can be traced further back to Simon Stevin in 16th century (Stevin, 1586) and Varignon's introduction of the funicular polygons (Varignon, 1725), shown in figure 2.10b. Engineering heroes who used graphic statics as a design tool include Robert Maillart and Gustave Eiffel (Block, Van Mele, Rippmann, & Paulson, 2017).

Numerical form-finding methods were introduced in the 20th century. Among them are Dynamic relaxation (Day, 1965) and the Force density method (Schek, 1974). Both were material independent methods, similarly to graphic statics.

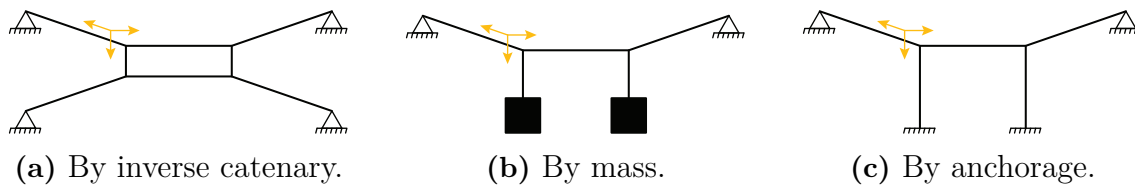
The reverse of form-finding is form-setting which involves modification of loads, prestress and support conditions to achieve a desired form.

## 2.5 Prestress

Most cable structures are more or less impractical without the introduction of prestressing due to unacceptable performance during service. Cable systems can be stabilized by increased self weight, by anchoring or by prestressing inverse cables which makes them less sensitive to varying loading conditions, shown in figure 2.11



below.



**Figure 2.11:** Three approaches to prestressing.

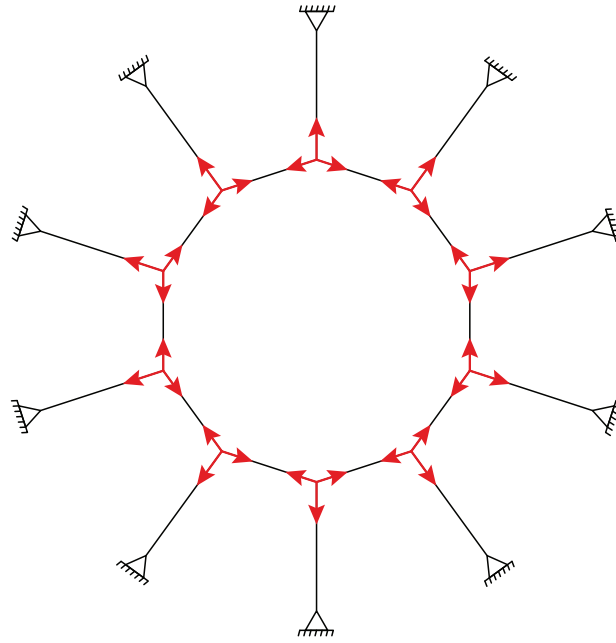
Prestress is used to improve the stability where additional stresses contribute to an increased geometrical stiffness. Moreover, prestress is used to limit deformation by providing form stability as well as ensuring a favorable stress pattern in a structure. Some members may also be prestressed to avoid unfavourable tension in joints. In the case of tensegrity structures, prestress is used to eliminate internal mechanisms (Sehlström, 2019).

By controlling the stress pattern, it is possible to reduce compression in some members and hence reduce the risk of buckling. This possibility allows for light and slender structures, which are more material efficient.

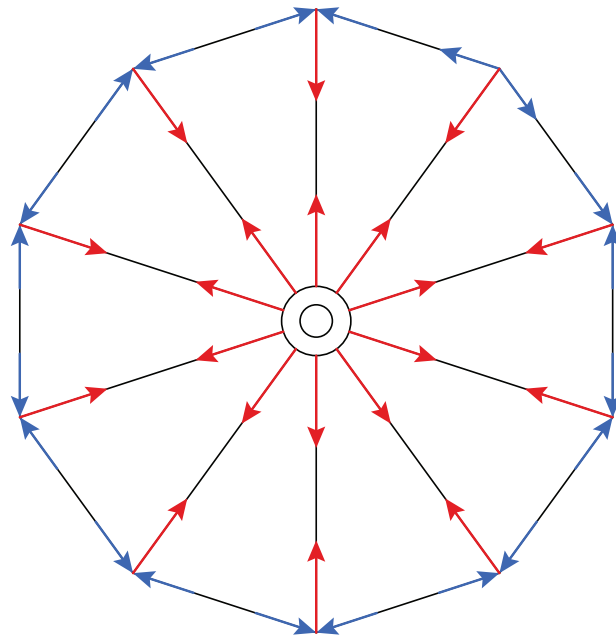
As a conclusion, prestress may be used to:

1. Ensure stability: To provide useful geometrical stiffness and eliminate mechanisms.
2. Achieve desired form: To fulfil design goals.
3. Efficient joinery: To reduce tension in connections.
4. Material efficiency: To reduce unfavourable compression and avoid buckling.

Prestressed systems are often categorized into two different types depending on how prestressing is affecting the reaction forces. Externally-equilibrated systems increases the reaction forces with increased prestressing and auto-equilibrated system maintain the prestress in the structure. Figure 2.12 shows two simple examples to illustrate this.



(a) Externally-equilibrated system.



(b) Auto-equilibrated system.

**Figure 2.12:** The effect of prestress on joint equilibrium.

# 3

## Numerical Methods

Numerical methods are used in design and analysis in order to provide accurate approximations to complex analytical problems. The methods have typically a convergence criteria and employs discretization of objects to a finite number of elements.

In the following chapter, three commonly used numerical methods for the design of cable supported structures are discussed: the dynamic relaxation method; the force density method; and, the non-linear finite element method. They are among the earliest methods available. However, all three are based on different approaches and more recent methods are often based on any of them.

### 3.1 Dynamic Relaxation

The dynamic relaxation method (DRM) solves the static deformations of a structure or mechanism subjected to loads by treating the problem as a dynamic one. In essence the DRM is used to find equilibrium states in structural systems which are discretized into a number of element and where fictitious masses are lumped into nodes. A system can be damped and would oscillate over time until equilibrium is found. The method is used in form-finding where the stiffness can be disregarded or may be configured with actual stiffness of elements to work as a base for analysis.

The method was first presented by Alister Day in 1965 who applied the method for the design of a number of projects including Allianz stadium (Carfrae & Michael, 2019a, 2019b). Over the years the method has been extended by several contributors and can be used for the form-finding of tension space structures (Barnes, 1977) and prestressed networks supported by compression arches (Wakefield, 1980). The development of a spline element (Adriaenssens & Barnes, 2001) and a bending and torsion resistance element (Barnes, Adriaenssens, & Krupka, 2013) have enabled analysis of complex structural systems.

To use dynamic relaxation an overview of the calculation scheme is shown in figure 3.1. The procedure starts by defining an initial geometry and initial velocities set to zero. After applying nodal loads, the axial forces can be calculated. The residual force is then computed and used to calculate nodal velocities and updated nodal coordinates. If the norm of the residual force is smaller than a predefined tolerance value, the solution have converged. If not, new axial forces, velocities and coordinates are recalculated until convergence is reached.

The procedure is elaborated with the calculations below. The initial geometry can be established by defining the topology and coordinate matrices (Veenendaal & Block, 2012). The topology can be described by the use of a connectivity matrix (**C**-matrix) defining the start and end node of each element. The matrix has  $m$ -rows and  $n$ -columns, where  $m$  is equal to the number of elements and  $n$  number of nodes in the system. The **C**-matrix is constructed following the rules:

$$\mathbf{C}_{(i,j)} = \begin{cases} +1 & \text{if first node} \\ -1 & \text{if last node} \\ 0 & \text{else.} \end{cases}$$

The **C**-matrix as well as the coordinate arrays  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  are divided into two parts separating free nodes and fixed nodes as  $\mathbf{C} = [\mathbf{C}_N \ \mathbf{C}_F]$ , and  $\mathbf{x} = [\mathbf{x}_N \ \mathbf{x}_F]$ ,  $\mathbf{y} = [\mathbf{y}_N \ \mathbf{y}_F]$ ,  $\mathbf{z} = [\mathbf{z}_N \ \mathbf{z}_F]$ . The notations  $N$  and  $F$  refers to free and fixed nodes respectively.

The length of each branch is derived from the coordinates differences,  $\mathbf{u}$ ,  $\mathbf{v}$ ,  $\mathbf{w}$ . Uppercase letters refers to the diagonal matrix of corresponding lower case vector and is furthermore calculated as:

$$\mathbf{u} = \mathbf{C}\mathbf{x}, \quad \mathbf{U} = \text{diag}(\mathbf{u}) \quad (3.1)$$

$$\mathbf{v} = \mathbf{C}\mathbf{y}, \quad \mathbf{V} = \text{diag}(\mathbf{v}) \quad (3.2)$$

$$\mathbf{w} = \mathbf{C}\mathbf{z}, \quad \mathbf{W} = \text{diag}(\mathbf{w}) \quad (3.3)$$

The initial nodal velocities,  $\mathbf{v}_i$  with components  $\mathbf{v}_x$ ,  $\mathbf{v}_y$  and  $\mathbf{v}_z$ , may all be set to zero and nodal loads are applied on free nodes through  $\mathbf{p}_i$ , where  $i = x, y, z$ . Initial stresses can be introduced by prescribing a load vector  $\mathbf{f}_0$  or by reducing the initial length of a chosen branch by an equivalent length.

Axial forces are calculated and updated for each loop with the new lengths  $\mathbf{l}$  and engineering strain,  $\epsilon = (l - l_i)/l_i$ . From the equilibrium conditions follows:

$$\mathbf{P}_x = \mathbf{C}^T \mathbf{Q} (\mathbf{C}\mathbf{x} + \mathbf{C}_F \mathbf{x}_F), \quad (3.4)$$

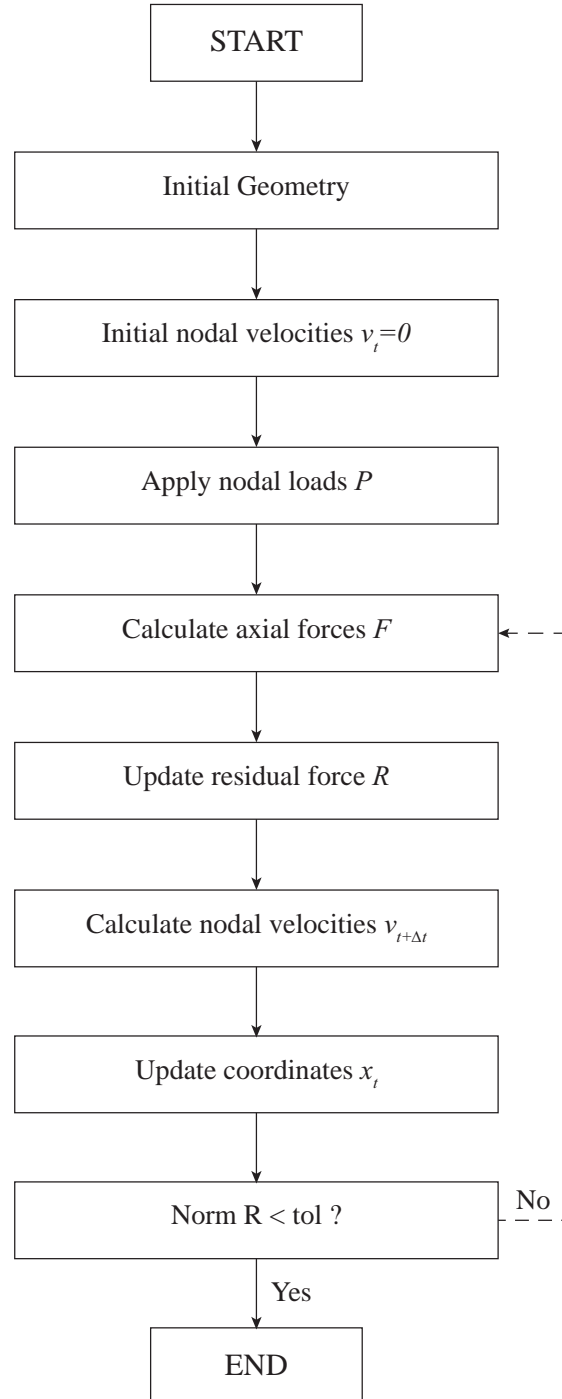
$$\mathbf{P}_y = \mathbf{C}^T \mathbf{Q} (\mathbf{C}\mathbf{y} + \mathbf{C}_F \mathbf{y}_F), \quad (3.5)$$

$$\mathbf{P}_z = \mathbf{C}^T \mathbf{Q} (\mathbf{C}\mathbf{z} + \mathbf{C}_F \mathbf{z}_F), \quad (3.6)$$

with  $\mathbf{D}_N = \mathbf{C}_N^T \mathbf{Q} \mathbf{C}_N$  and  $\mathbf{D}_F = \mathbf{C}_F^T \mathbf{Q} \mathbf{C}_F$ , the residual forces calculated as:

$$\mathbf{r}_x = \mathbf{p}_x - \mathbf{D}_N \mathbf{x}_N - \mathbf{D}_F \mathbf{x}_F, \quad (3.7)$$

$$\mathbf{r}_y = \mathbf{p}_y - \mathbf{D}_N \mathbf{y}_N - \mathbf{D}_F \mathbf{y}_F, \quad (3.8)$$



**Figure 3.1:** Dynamic relaxation scheme.

$$\mathbf{r}_z = \mathbf{p}_z - \mathbf{D}_N \mathbf{z}_N - \mathbf{D}_F \mathbf{z}_F, \quad (3.9)$$

where the total residual is the euclidean norm of  $\mathbf{r}_x$ ,  $\mathbf{r}_y$ ,  $\mathbf{r}_z$ .

By defining the nodal masses as a function of the greatest direct stiffness in a node shown in equation (3.10), the velocities can be calculated following Newton's second law. The integration is similar to a Leapfrog integration which requires the time step to be constant and is usually chosen to  $\Delta t = 1$  (Adriaenssens, Block, Veenendaal, & Williams, 2014). In matrix formulation this can be expressed as:

$$\mathbf{m} = 0.5\Delta t |\mathbf{C}_N| (\mathbf{f}_o/l + \mathbf{EA}/l_i), \quad \mathbf{M} = \text{diag}(\mathbf{m}), \quad (3.10)$$

where  $\mathbf{EA}$  is a vector of length  $m$  and includes the axial stiffness for each element.

The calculated velocities are used for updating the coordinate vectors:

$$\mathbf{v}_x^{t+\Delta t/2} = A\mathbf{v}_x^{t-\Delta t/2} + B\Delta t\mathbf{M}^{-1}\mathbf{r}_x, \quad (3.11)$$

$$\mathbf{v}_y^{t+\Delta t/2} = A\mathbf{v}_y^{t-\Delta t/2} + B\Delta t\mathbf{M}^{-1}\mathbf{r}_y, \quad (3.12)$$

$$\mathbf{v}_z^{t+\Delta t/2} = A\mathbf{v}_z^{t-\Delta t/2} + B\Delta t\mathbf{M}^{-1}\mathbf{r}_z, \quad (3.13)$$

$$\mathbf{x}_{t+\Delta t/2} = \mathbf{x}_{t-\Delta t/2} + \Delta t\mathbf{v}_x^{t+\Delta t/2}, \quad (3.14)$$

$$\mathbf{y}_{t+\Delta t/2} = \mathbf{y}_{t-\Delta t/2} + \Delta t\mathbf{v}_y^{t+\Delta t/2}, \quad (3.15)$$

$$\mathbf{z}_{t+\Delta t/2} = \mathbf{z}_{t-\Delta t/2} + \Delta t\mathbf{v}_z^{t+\Delta t/2}. \quad (3.16)$$

Factor  $A$  and  $B$  in the velocity equations 3.11 - 3.12 refers to the damping coefficients. The speed at which the procedure reaches equilibrium is highly dependent on the chosen timestep and damping factors. The damping factors can be applied as constant or adaptive values. It is also possible to combine the two methods.

## 3.2 Force Density Method

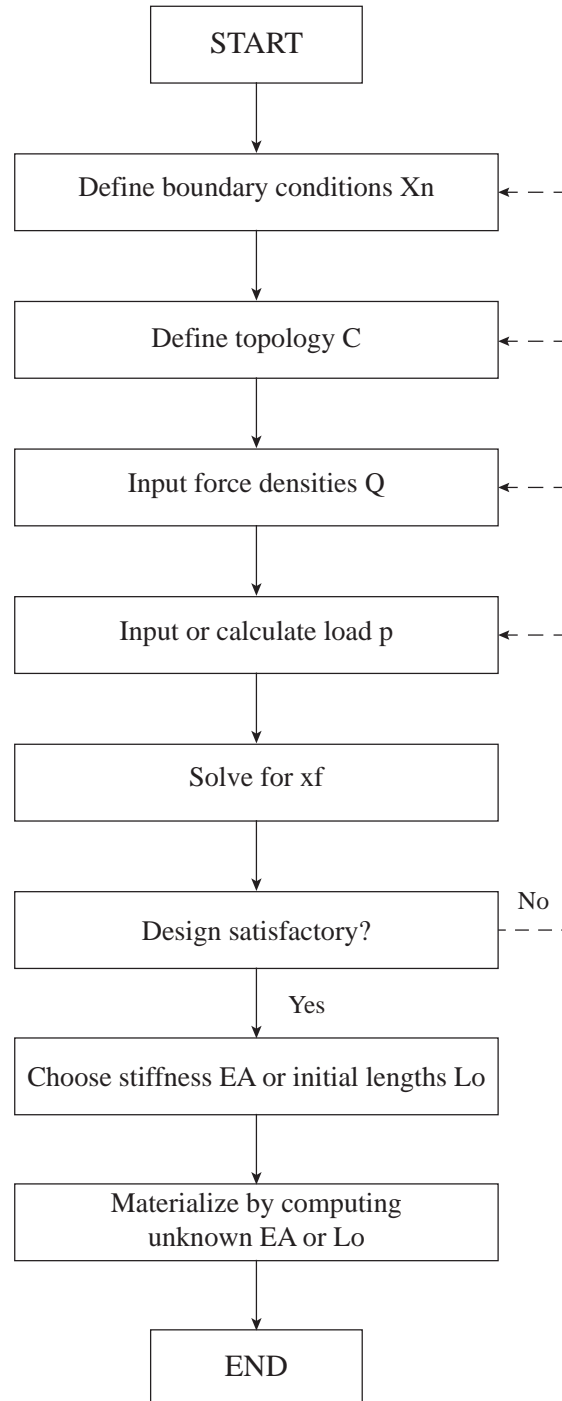
The force density method (FDM) was introduced in 1974 by Schek. The method was developed to allow for a linear and analytical form-finding procedure that is very computationally efficient. Schek formulated the FDM around the ratio between the force and the deformed length of each element. The ratio,  $q = F/L$ , is often referred to as the force density or the force-length ratio. By defining the force densities in a system, the FDM finds the equilibrium state where all forces in the nodes sums up to zero.

The initial selection of force densities seldom results in the final design. To overcome this issue, several extensions of the method have been presented to allow for additional conditions to be considered. Additional constraints have often non-linear implications, which requires the extensions to FDM to be non-linear as well.

The extended versions of the method resolves different non-linear phenomena. Schek derived an extension of FDM for a cable-net of constrained element lengths (Schek, 1974). Haber and Abel (1982) presented the "General Stiffness Method", a method to derive the unknown prestress distribution from a chosen reference shape. Linkwitz 1999 devised a method for double curved cable-nets. Other contributions includes the "Updated Reference strategy" (Bletzinger & Ramm, 2001), "Natural Force Density Method" (Pauletti & Pimenta, 2008) and "Multi-step Force Density method" (Javier Sanchez, 2006), which are extension of the method for surface elements.

In *Shell structures for architecture: form finding and optimization* (Adriaenssens et al., 2014), the use of FDM in practice is demonstrated through a presentation of the design process of the Mannheim Multihalle (1974) as well as Solemar Therme (1987).

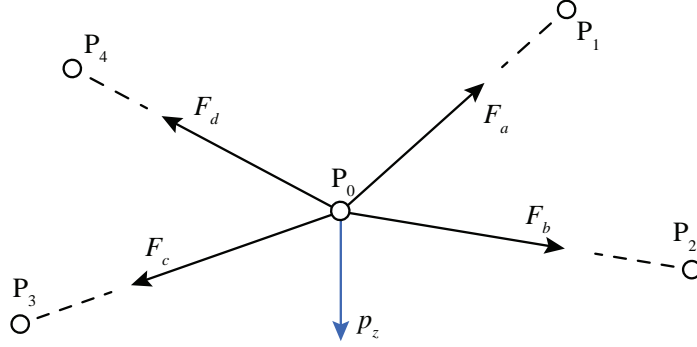
To use the force density method, an overview of the workflow is illustrated in figure 3.2. The scheme starts by defining an initial geometry and boundary conditions. From that, a connectivity matrix (**C**-matrix) can be constructed. Force densities are then applied to each element and loads are placed on nodes. The equilibrium position of the free nodes in a system are subsequently calculated. If the emerged shape is not satisfactory, the user can change the input geometry, boundary conditions, force densities or input loads. When a satisfactory shape is reached, the user can either define element stiffness or the initial lengths and from that calculate the unknown.



**Figure 3.2:** Force density method scheme.



To illustrate the theory of the force density method, a node  $P_0$  in the cable-net shown in figure 3.3 is studied. Four cables connect to node  $P_0$  and the node is loaded with an external force  $p_z$ . The axial forces  $F_i$ , with  $i = a, b, c, d$  are given from Hooke's law as:



**Figure 3.3:** Node equilibrium principle

$$F_i = \frac{EA}{l_0} e_i, \quad (3.17)$$

where  $EA$  is the axial stiffness,  $l_0$  is the initial length of the element and the elongation of an element is  $e_i = l_i - l_{0,i}$ . The element length  $l_i$ , where  $k = 1, 2, 3, 4$  can be calculated as:

$$l_i = \sqrt{(x_k - x_0)^2 + (y_k - y_0)^2 + (z_k - z_0)^2}. \quad (3.18)$$

In a Cartesian coordinate-system the element force  $F_i$  can be deconstructed to retrieve the x-component:

$$F_{i,x} = F_i \cos \alpha_i. \quad (3.19)$$

By adding all element forces in the  $x$ -direction together with the applied force  $p_x$ , the equilibrium of the is obtained from:

$$F_a \cos \alpha_a + \dots + F_d \cos \alpha_d + p_x = 0. \quad (3.20)$$

The direction cosines in equation (3.19) can be rewritten:

$$\frac{x_1 - x_0}{l_a} F_a + \dots + \frac{x_4 - x_0}{l_d} F_d + p_x = 0. \quad (3.21)$$

With the of equation (3.17), equation (3.21) can be expressed:

$$\frac{x_1 - x_0}{l_{0,a}} \frac{EA}{l_0} (l_i - l_{0,a}) + \dots + \frac{x_4 - x_0}{l_{0,d}} \frac{EA}{l_0} (l_i - l_{0,d}) + p_x = 0. \quad (3.22)$$

To handle the non-linearity in equation (3.22), the force density quantity  $q_i$  will be defined and inserted in the following way:

$$q_i = \frac{F_i}{l_i} \implies (x_1 - x_0)q_a + \dots (x_4 - x_0)q_d + p_x = 0. \quad (3.23)$$

The goal is to extract the position of the free node which in this case is  $x_0$  and this can be done since the equation now is linear. The same principle applies to  $y_0$  and  $z_0$ , which can computed:

$$\begin{aligned} x_0 &= \frac{p_x + x_1 q_a + x_2 q_b + x_3 q_c + x_4 q_d}{q_a + q_b + q_c + q_d}, \\ y_0 &= \frac{p_y + y_1 q_a + y_2 q_b + y_3 q_c + y_4 q_d}{q_a + q_b + q_c + q_d}, \\ z_0 &= \frac{p_z + z_1 q_a + z_2 q_b + z_3 q_c + z_4 q_d}{q_a + q_b + q_c + q_d}. \end{aligned} \quad (3.24)$$

So far, the equilibrium of one node has been derived. However, when dealing with a complex structure with several nodes and elements the number of equations can get overwhelming and tedious. Therefore, matrix formulations will be introduced similarly to the approach in dynamic relaxation outlined in section 3.1.

The topology of an extensive structure can be represented in a connectivity matrix  $\mathbf{C} = [\mathbf{C}_N \quad \mathbf{C}_F]$  as described in section 3.1. In a similar fashion, the node-vector  $\mathbf{x}$  can be divided into free nodes and fixed nodes,  $\mathbf{x} = [\mathbf{x}_N \quad \mathbf{x}_F]$ .

With  $\mathbf{D}_N = \mathbf{C}_N^T \mathbf{Q} \mathbf{C}_N$  and  $\mathbf{D}_F = \mathbf{C}_F^T \mathbf{Q} \mathbf{C}_F$ , the equilibrium position of nodes can be calculated as

$$\begin{aligned} \mathbf{x}_N &= \mathbf{D}_N^{-1} (\mathbf{p}_x - \mathbf{D}_F \mathbf{x}_F) \\ \mathbf{y}_N &= \mathbf{D}_N^{-1} (\mathbf{p}_y - \mathbf{D}_F \mathbf{y}_F) \\ \mathbf{z}_N &= \mathbf{D}_N^{-1} (\mathbf{p}_z - \mathbf{D}_F \mathbf{z}_F) \end{aligned} \quad (3.25)$$

### 3.3 Finite Element Method

The finite element method have had a profound effect on the engineering practice. By discretizing a system into a number of finite elements, approximations over a domain for partial differential equations can be solved numerically. In the 1960 and early 1970, finite element analysis developed rapidly and was expanded to solve non-linear problems using different iterative procedures such as the Newton-Raphson's method (Oden, 1990).

The general procedure for the finite element method involves derivation of element equations, discretizing into a finite number of elements, calculation of element stiffness, defining external forces and finally assembly into a global system. Boundary conditions are thereafter introduced and nodal unknowns can be computed. Lastly, element quantities such as forces and displacements can be computed (Bhatti, 2006). The procedure for solving a finite element system consisting of Green-Lagrange bar elements in 3D is summarized in figure 3.4.

Non-linear behaviour due to large rotations which can be present in cable structures must be accounted for. The finite element can be defined so that it includes contributions from large rotations. A way to achieve this is to use the Green strain applicable for large displacements. Commonly the resulting tangent stiffness matrix is composed by the linear stiffness matrix and a geometric non-linear stiffness matrix contribution (Crisfield, 1993).

$$\mathbf{K}_T^e = \mathbf{K}_L^e + \mathbf{K}_{NL}^e, \quad (3.26)$$

where the linear stiffness matrix is:

$$\mathbf{K}_L^e = EA \frac{l^2}{l_0^3} \begin{bmatrix} c_x^2 & c_x c_y & c_x c_z & -c_x^2 & -c_x c_y & -c_x c_z \\ c_x c_y & c_y^2 & c_y c_z & -c_x c_y & -c_y^2 & -c_y c_z \\ c_x c_z & c_y c_z & c_z^2 & -c_x c_z & -c_y c_z & -c_z^2 \\ -c_x^2 & -c_x c_y & -c_x c_z & c_x^2 & c_x c_y & c_x c_z \\ -c_x c_y & -c_y^2 & -c_y c_z & c_x c_y & c_y^2 & c_y c_z \\ -c_x c_z & -c_y c_z & -c_z^2 & c_x c_z & c_y c_z & c_z^2 \end{bmatrix}. \quad (3.27)$$

The direction cosines  $c_x, c_y, c_z$  are defined as:

$$c_x = \frac{x_2 - x_1}{l}, \quad c_y = \frac{y_2 - y_1}{l}, \quad c_z = \frac{z_2 - z_1}{l}. \quad (3.28)$$

The non-linear part of the matrix is:

$$\mathbf{K}_{NL}^e = EA \frac{l^2 - l_0^2}{2l_0^3} \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}. \quad (3.29)$$

The internal force vector is defined as:

$$\mathbf{f}_{\text{int}}^e = EA \frac{l^2 - l_0^2}{2l_0^2} \begin{bmatrix} -c_x \\ -c_y \\ -c_z \\ c_x \\ c_y \\ c_z \end{bmatrix}. \quad (3.30)$$

Green-Lagrange strain is not linear which means that an iterative procedure must be used to solve the equation system. Since the Green-Lagrange element is a function of displacements  $\mathbf{a}$ , the equations system can be expressed as follows:

$$\mathbf{K}(\mathbf{a})\mathbf{a} = \mathbf{f}_{\text{ext}}. \quad (3.31)$$

By using the concept of Taylor series, the next iteration can be approximated as:

$$\mathbf{K}(\mathbf{a}^{i+1}) \approx \mathbf{K}(\mathbf{a}^i)\mathbf{a}^i + \mathbf{K}_T^i \Delta \mathbf{a}^i = \mathbf{f}_{\text{ext}} \implies \mathbf{K}_T^i \Delta \mathbf{a}^i = \mathbf{f}_{\text{ext}} - \mathbf{K}(\mathbf{a}^i)\mathbf{a}^i, \quad (3.32)$$

where the residual force vector  $\mathbf{R}^i$  is:

$$\mathbf{R}^i = \mathbf{f}_{\text{ext}} - \mathbf{K}(\mathbf{a}^i)\mathbf{a}^i. \quad (3.33)$$

The equation system is now linearized and the Newton-Raphson iterative procedure can be used to solve the non-linear problem. The displacement  $\Delta \mathbf{a}^i$  increment is therefore:

$$\begin{aligned} \mathbf{K}_T(\mathbf{a}^i) &= \mathbf{R}^i(\mathbf{a}^i), \\ \mathbf{K}_T(\mathbf{a}^i)\Delta \mathbf{a}^i &= -\mathbf{R}^i(\mathbf{a}^i), \\ \Delta \mathbf{a}^i &= -\mathbf{K}_T^{-1} \mathbf{R}^i. \end{aligned} \quad (3.34)$$

The displacement vector can now be updated:

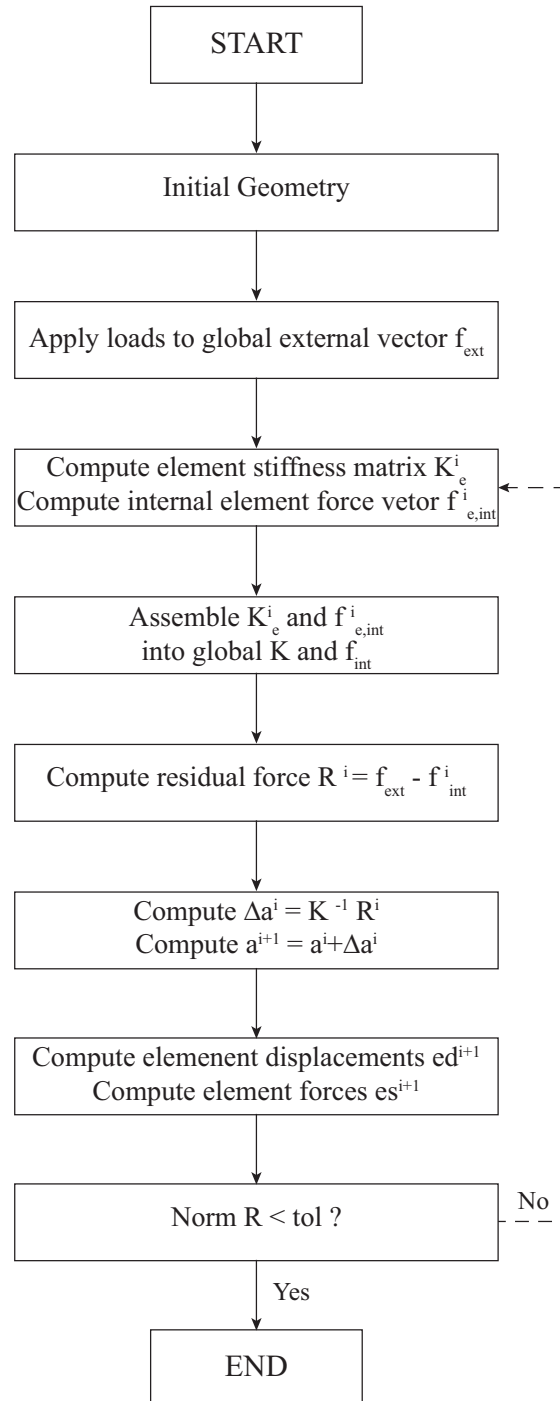
$$\mathbf{a}^{i+1} = \mathbf{a}^i + \Delta \mathbf{a}^i, \quad (3.35)$$

and consequently the residual force  $\mathbf{R}^i$  can be calculated as:

$$\mathbf{R}^{i+1} = \mathbf{f}_{\text{ext}} - \mathbf{K}(\mathbf{a}^{i+1})\mathbf{a}^{i+1}. \quad (3.36)$$

If the residual  $\mathbf{R}^i$  is small enough the solution has converged. However, should the residual not satisfy the convergence criteria a new iteration must be executed. The convergence criteria which ends the iterative process is defined as:

$$\text{conv} = \frac{\|\mathbf{R}^i\|^2}{1 + \|\mathbf{f}_{\text{ext}}\|^2} < \text{tol}. \quad (3.37)$$



**Figure 3.4:** Non-linear finite element method scheme.



# 4

## Evaluation of Numerical Methods

To decide which of the three numerical methods discussed in chapter 3 to implement, each method was evaluated and assessed. The purpose of evaluating the methods is to assess the strength and weaknesses of each method. An important aspect is to avoid a solver which is hard to modify or extend but at the same time provide necessary control of the tool (Wakefield, 1999). Consequently a "black box" solver should be avoided.

Moreover, several programming aspects are reflected upon including: (1) execution speed; (2) ease of implementing different member types; (3) constraints; and (4) how prestress can be assigned.

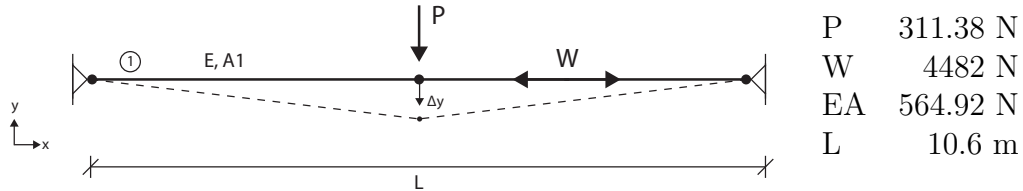
Three examples were studied and solved using MATLAB, and they are summarized in table 4.1 together with a short description of each example. The three examples have all different purposes and the results from each method will be compared. In addition, the results from each method will be referenced to the commercial software RFEM by Dlubal which is stiffness matrix FEM solver and is a common tool in structural engineering practices.

**Table 4.1:** Evaluation examples.

Nr	Geometry	Description
1	Cable	Form-finding and analysis of a cable with initial prestress
2	Truss	Analysis of a statically undetermined truss system with small strains
3	Braced Frame	Analysis of a frame with cable elements with large strains.

## 4.1 Example 1: Form-Finding of Cable

In example 1, a horizontal taut cable is subjected to a vertical point load  $P$  and prestress  $W$ , illustrated in figure 4.1. The purpose of this example is to test the capability of form-finding and analysis for a simple cable structure.



**Figure 4.1:** Simple cable inspired by (Luo & Lu, 2006).

From the results in table 4.2, one can conclude that results obtained both from NL-FEM and DRM, converge into the same equilibrium state with a small increase in the cable force  $N$  compared to the initial prestress force  $W$ .

The force density is defined as  $q = W/L$  for the FDM ( $q$  is an input value). This force density only includes the initial prestress force. This explains why the normal force is smaller in the FDM. For NL-FEM and DRM, the contribution from the strain force is included and found through outlined iterative procedures in chapter 3.

The initial state of the cable may cause convergence problem with the NL-FEM method. The existence of a prescribed displacement or prestress is crucial to "activate" the geometrical stiffness and avoiding a singular stiffness matrix.

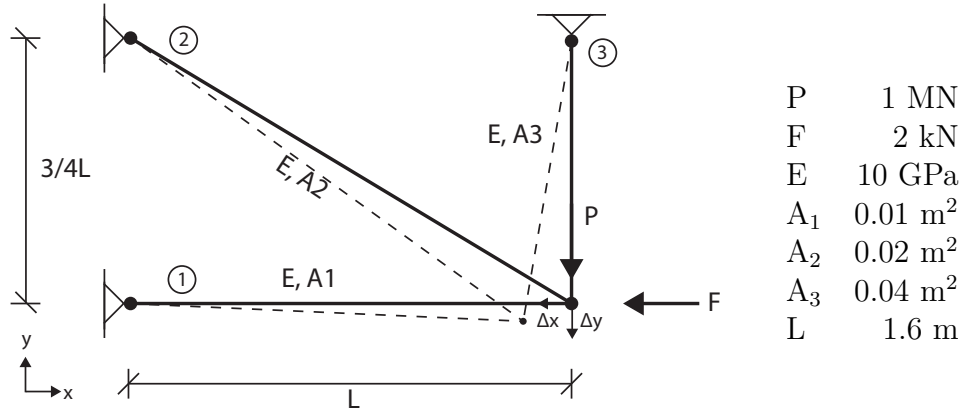
**Table 4.2:** Comparison of displacement and normal force for example 1.

Method	$\Delta y$	$N$
<b>DRM</b>	1.000	1.000
<b>NL-FEM</b>	1.000	1.000
<b>FDM</b>	0.999	0.999
<b>RFEM</b>	<b>1.000</b>	<b>1.000</b>
ref	177.9 mm	4448.5 N



## 4.2 Example 2: Analysis of Truss

Example 2 is a simple truss consisting of three bar elements as illustrated in figure 4.2. Its stiffness values and geometry is based on Example 9.1 in *Structural mechanics: Modelling and analysis of frames and trusses* but extended with an additional vertical bar to establish a statically indetermined structure (Olsson & Dahlblom, 2016). The structure is analyzed for a vertical load  $P$  and a horizontal load  $F$ , and the aim is to calculate the member forces and free-node displacement. The force density method is excluded from this example since it is best suited for form-finding.



**Figure 4.2:** Truss inspired by (Olsson & Dahlblom, 2016).

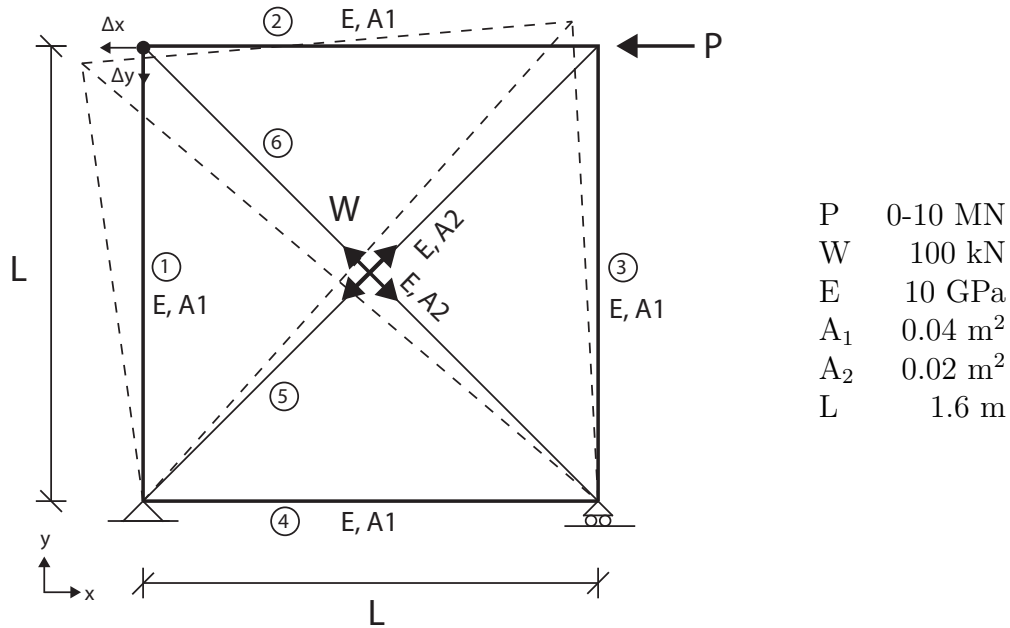
The results are presented in table 4.3 with the result weighted against RFEM. The maximum deviation between results obtained from the DRM and RFEM is 0.8% which is considered negligible. It is hard to point out where the relatively small deviations stem from, but a justifiable argument could be the difference in the strain models assumed by each analysis method. However, since the difference is only at 0.8 %, it is safe to assume that both NL-FEM and DRM show that they can be used for analysis of bar elements in a simple truss structure for small strains.

**Table 4.3:** Comparison of displacements and normal forces for example 2.

Method	$\Delta x$	$\Delta y$	N1	N2	N3
DRM	1.008	1.000	0.999	1.001	1.001
NL-FEM	1.002	1.000	1.003	1.006	1.002
RFEM	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>
ref	-8.304 mm	-1.113 mm	-520.3 kN	-597.6 kN	380.3 kN

### 4.3 Example 3: Truss with Diagonal Ties

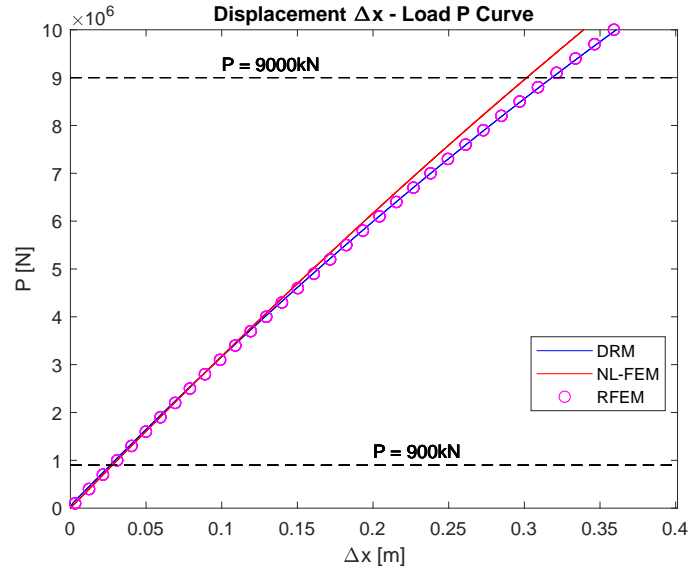
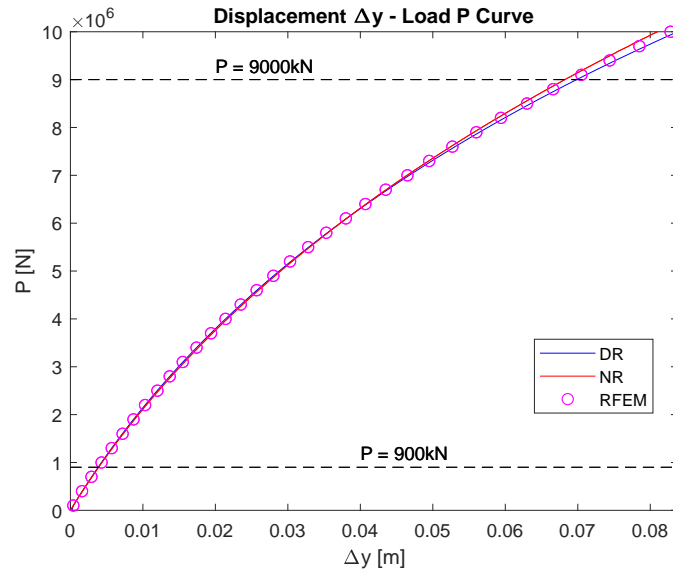
A frame structure illustrated in figure 4.3, with prestressed cross-bracing is studied in Example 3. The purpose is to investigate if any method has any limitations with regard to large strains. The load  $P$  will be incrementally increased from 0 to an exaggerated load of 10 MN.



**Figure 4.3:** Truss with diagonal ties inspired by (Guest, 2006).

The displacement of top left node is presented in figure 4.4. From the plots it can be seen that all methods yields similar displacements for low values of  $P$ . However, as  $P$  increases the results of NL-FEM diverges from the results of DRM and RFEM. This is explained by the different strain measures used in the methods. In NL-FEM the Green strain is used, while in DRM and RFEM engineering strain is implemented. The maximum deviation of the displacements of NL-FEM compared to RFEM is only at 0.5% for  $P = 0.9$  MN. However, for  $P = 9$  MN the maximum deviation is at 5.8%.

To provide a better understanding of why the deviation occurs, two different load levels are studied. The first load level corresponds to a maximum strain of 2% presented in table 4.4, and the second a maximum strain of 10% illustrated table 4.5.

(a) Load-displacement curve,  $P$ - $\Delta x$ .(b) Load-displacement curve,  $P$ - $\Delta y$ .**Figure 4.4:** Effect of large loads on displacements.**Table 4.4:** Comparison of displacements and normal forces for example 3,  
 $P = 0.9 \text{ MN}$ ,  $\epsilon_{max} = 2\%$ .

Method	$\Delta x$	$\Delta y$	N5	N6
DRM	1.000	1.001	1.000	1.000
NL-FEM	0.995	1.003	1.000	1.000
RFEM	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>
ref	-26.29 mm	-3.820 mm	0 kN	1283 kN

**Table 4.5:** Comparison of displacements and normal forces for example 3,  
 $P = 9 \text{ MN}$ ,  $\epsilon_{max} = 10\%$ .

Method	$\Delta x$	$\Delta y$	N5	N6
DRM	1.004	1.010	1.000	1.000
NL-FEM	0.942	0.974	1.000	0.996
RFEM	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>	<b>1.000</b>
ref	-317.0 mm	-69.13 mm	0 kN	13990 kN

## 4.4 Conclusion

To begin with, there seems to be no consensus on which method is preferable and the three methods have all been used to design cable supported structures. The methods have been extended rigorously over the years and it is hard to differentiate the possible capabilities. The choice of a preferable method is partially subjective and based on the previous knowledge of the designer.

The simplicity of the force density method is based on the basic definition of force densities for a structure. By changing the force density values, alternative designs can be can form-found quickly. However, the method could not be used in example 2 and 3 as it would require non-linear extensions to use in analysis. A non-linear extension would contradict the argument of the method's simplicity. Moreover, the concept of force densities could be somewhat abstract for some users, and requires experience to get adequate results. Therefore, it has been concluded that the force density method is not suitable to implement in our tool.

DR proved capable of solving all the three investigated examples. The method has several good inherent qualities when dealing with geometrical non-linear problems. One of the great strengths of the DRM is the method's clear relation to real physical motion which is represented in the solution procedure. The exception is the damping and nodal masses which are lacking a direct physical representation.

Non-linear finite element method proved capable of solving Example 1 and Example 2. In example 3, where the effects of an exaggerated load were studied, NL-FEM showed different results compared to DRM. This difference may be explained by the non-linear geometric stiffness and is a consequence of the Green strain definition. However, the scenario with high strains, as discussed in example 3, is highly unlikely and would result in yielding or even failure.

There are many advantages with finite element analysis. The method is known to many engineers and frequently used in engineering practices and the knowledge of the method is wide spread. The generality of the method allows for a variety of possible theories to be implemented which means that a FEM package can be extended and modified to suit specific purposes. This is why the NL-FEM is selected as most promising method to proceed with.

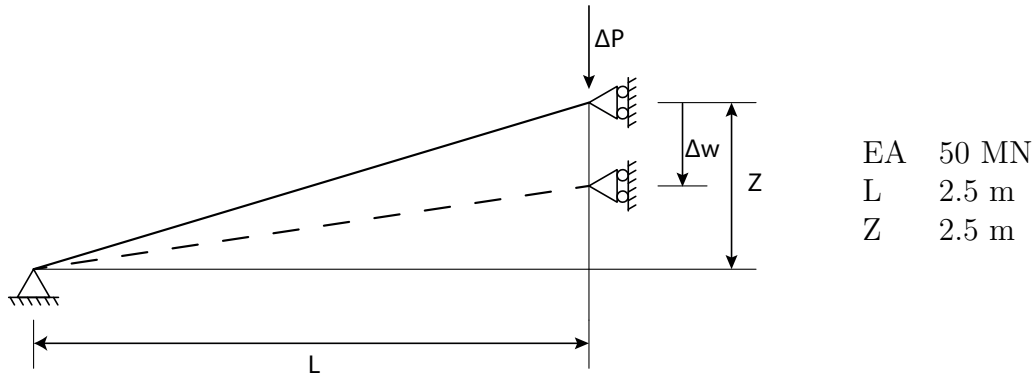
# 5

## Benchmark

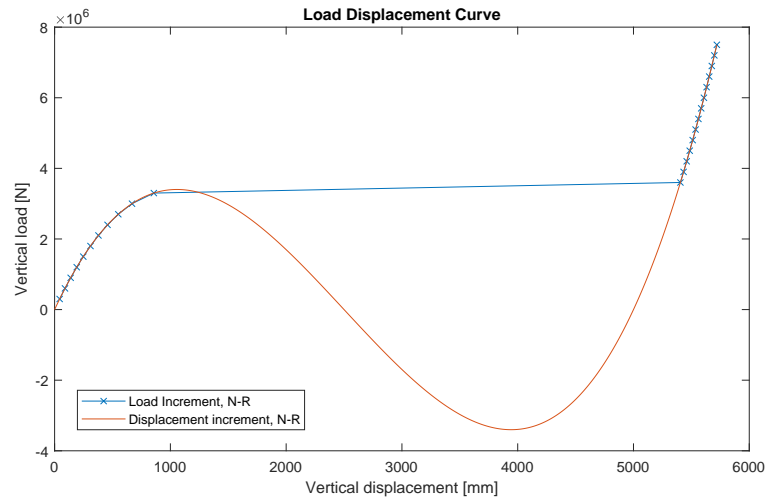
In order to check the limitations of the selected FEM method using Newton Raphson iterations and Green strain definition, it will be benchmarked against three examples extracted from *Non-linear finite element analysis of solids and structures* (Crisfield, 1993).

### 5.1 Snap-Through

A simple bar with geometry and properties according to figure 5.1, is studied. Two different analyses tracing the load-displacement relation are performed and compared. The first analysis is with constant load-increments  $\Delta P = 0.5$  MN, and the displacement  $\Delta w$  is calculated using equation (3.35). The second analysis is done by displacement-increments where the applied force  $P$  is calculated for a constant displacement increment  $\Delta w = 5.72$  mm.



**Figure 5.1:** Input for snap-through benchmark (Crisfield, 1993).



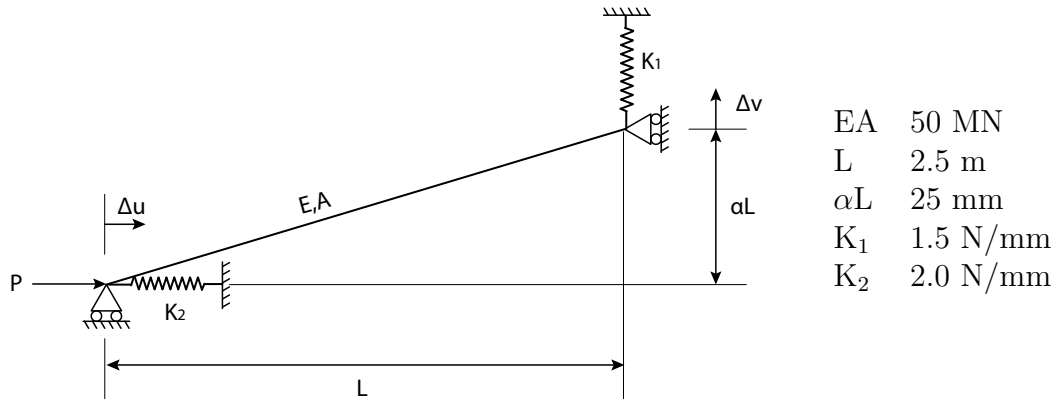
**Figure 5.2:** Load-displacement curves for benchmark 1.

Figure 5.2 shows the load displacement relationship of the studied bar. The results are identical until at approximately  $\Delta w = 1000$  mm. At the limit point, the load-increments analysis fail to capture the ‘snap-through behaviour of the bar’. Snap through is an instability which causes the displacements to leap from one steady configuration to the next. In this study, the load increments fail due to the tangent at the local maxima is almost flat and the next Newton-Raphson equilibrium point can only be found at the far end of the graph. In contrast, the displacement increments successfully catches this snap-through behaviour.

The results of displacement-controlled analysis are identical to the results in (Crisfield, 1993).

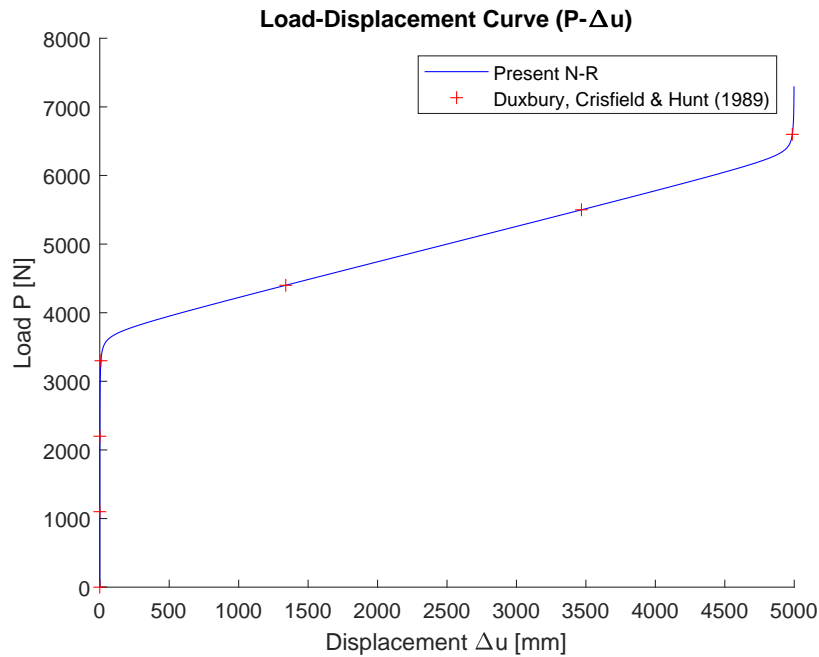
## 5.2 Hardening

In the second benchmark the structure in figure 5.1 is extended with an additional degree of freedom and with two springs as illustrated in figure 5.3 to study a continuous hardening problem. The analysis is carried out with load-increments and converges with sufficiently small load steps,  $\Delta P=1100$ N.

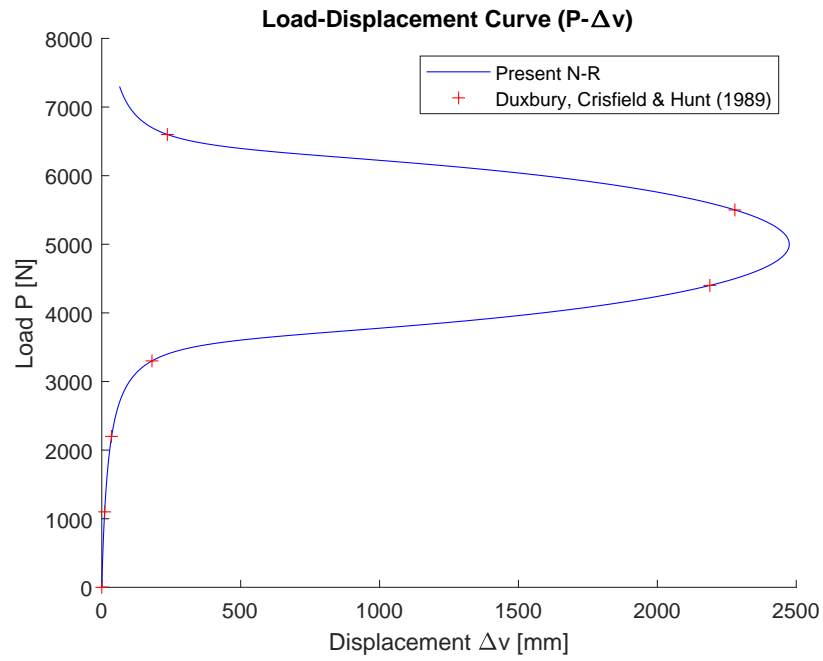


**Figure 5.3:** Input for hardening benchmark (Crisfield, 1993).

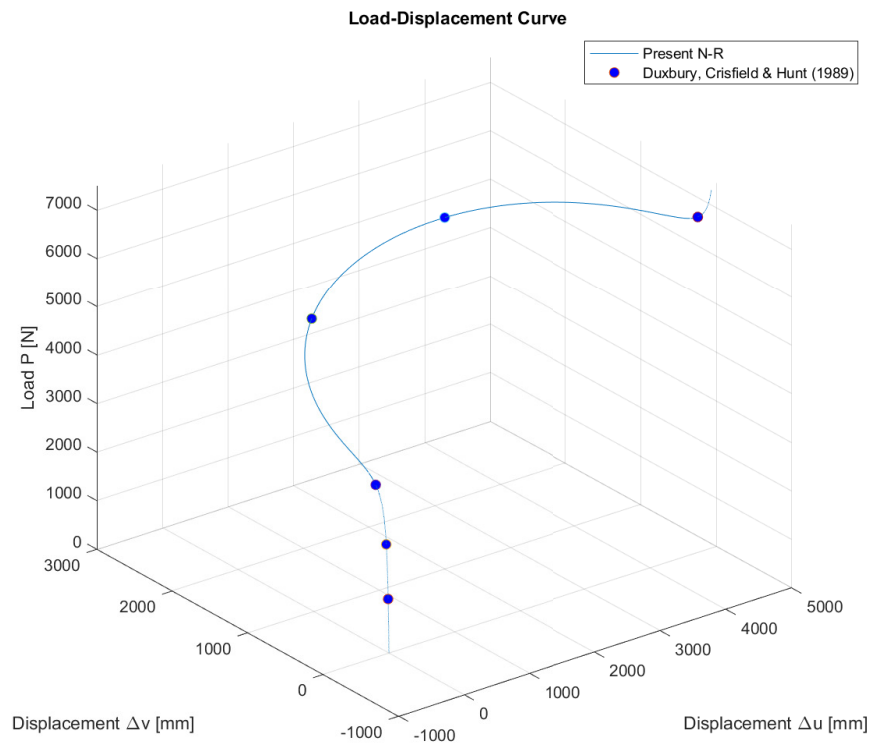
The load-displacement curves for the horizontal displacement  $u$  and vertical displacement  $v$  are plotted in figure 5.4 respectively figure 5.5. Upon comparing the obtained results with the literature data from “Benchtests for geometric nonlinearity” (Duxbury, Crisfield, & Hunt, 1989) it is observed that the results are identical. This confirms that the implemented method is capable of solving continuous hardening problems.



**Figure 5.4:** Load-displacement curve for benchmark 2,  $P$ - $\Delta u$ .



**Figure 5.5:** Load P-displacement curve for benchmark 2,  $P-\Delta v$ .



**Figure 5.6:** Load-displacement plot for benchmark 2,  $P-\Delta u-\Delta v$ .





### 5.4 Conclusion

In the first benchmark in section 5.1, load-increments with Newton-Raphson failed to capture the snap-through behaviour. For this benchmark, which has only one degree of freedom, the equilibrium could be found by changing to displacement-increments. Benchmark 2 in section 5.2, proves the capability to solve stress hardening problems. Snap-through did not exist in this problem and could be solved with NR iterations and load-increments.

The third benchmark in section 5.3 displays the weakness of NR iterations. Neither load-increments nor displacement-increments could catch the snap back. To overcome this limitation its possible to implement a more advance iteration procedure such as the arc-length method. A snap back is caused by a decrease in both load and displacement simultaneously which is an unexpected scenario for a real structure. Loads or displacement are often applied incrementally and the solution path during snap-back is an unrealistic event in most practical application.

As a consequence of the conclusion, The FEM design tool is implemented with Green strain definition and Newton-Raphson iteration technique. It will follow the equilibrium path up to limit points where snap-back or snap-through can appear. It will support load-increments as this is of most practical use for analysing structures where the displacement pattern are not easy to predict.

# 6

## Implementation of FEM\_Toolkit

An implementation of a non-linear FEM solver has been made. The implementation is written in the object-oriented programming language C#. The tool extends the 'Buildings and Habitats object Model' (BHoM) and uses the linear algebra library MathNet.Numerics. In this chapter, the outlines of the implementation is given followed by a description of how it is exposed to the user.

### 6.1 Object Oriented Programming in C#

Object-oriented programming is a programming paradigm that allows for the creation of objects which can interact with one another. The objects can contain data and methods which may be exposed to allow for modification and usage of the object. There are many object-oriented languages, but the most popular ones are class-based. Such languages defines classes which the objects are instances of, e.g. a 'Car' class can be defined demanding each instances of the class to contain information about the properties of each specific instance of Car.

The implementation of FEM\_Toolkit is made using the C# object-oriented programming language for the following reasons:

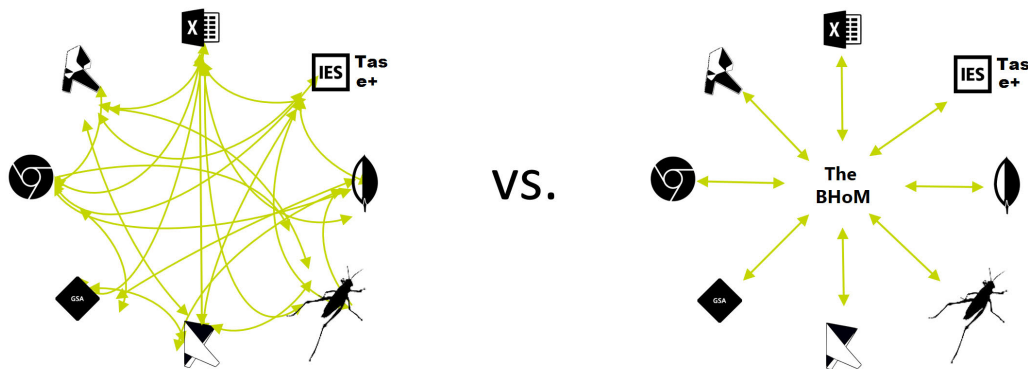
- C# is part of the .NET framework which runs on Windows operating systems.
- C# is a relatively simple, safe and high performance language.
- The availability of open source repositories (e.g. the BHoM is written in C#).

C#-code can not be run natively and has to be compiled first. The compilation of the code results in an assembly containing DLL and EXE-files that are used to run the program. Assemblies can then be exposed in different settings within the .NET framework where its objects and methods can be used (Liberty, 2005). For further details on programming in C# it is recommended to read *Programming C#: building .NET applications with C* (Liberty, 2005) and *The C# programming language* (Hejlsberg, Torgersen, Wiltamuth, & Golde, 2008). The code for the FEM\_Toolkit is written and compiled in Microsoft Visual Studio.

### 6.2 Building and Habitats object Model

The Buildings and Habitats object Model (BHoM), is an open-source collective computational project effort, with the main aim to facilitate for seamless co-creation of

code and collaboration across the architecture, engineering and construction industry. By standardizing data and creating a common core, it limits the number of links required to exchange data between software. Figure 6.1 illustrates the traditional way of data exchange between software versus the BHoM way of exchanging data.



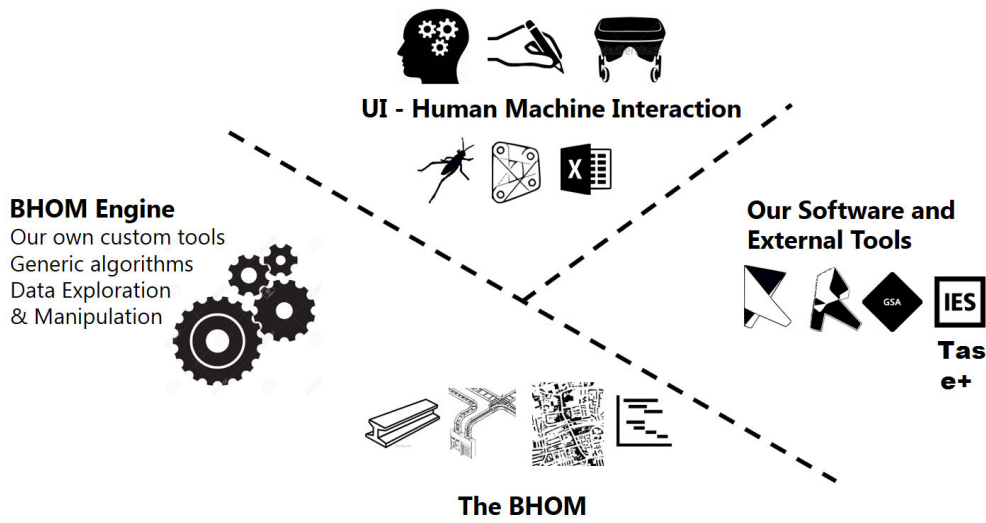
**Figure 6.1:** Software data exchange strategies.

Source: Copyright BHoM. Image reproduced with permission.

<https://github.com/BHoM/documentation/wiki/What-Is-the-BHoM-For>.

The core of the BHoM is structured around the 4 main categories of code illustrated in figure 6.2. They categories can be described as:

- The **oM**, consisting of data entities with a set properties that can manipulated, such as lines, beams and loads.
- The **Engine**, which is used to manipulate data in the **oM**.
- The **Adapter**, which allows for communication with external engineering softwares such as Revit, Robot or RFEM.
- The **UI**, exposes the **oM** and the **Engine** to user-interfaces such as Excel, Dynamo and Grasshopper.



**Figure 6.2:** The 4 categories of the BHoM-core.

Source: Copyright BHoM. Image reproduced with permission.

<https://github.com/BHoM/documentation/wiki/Structure-of-the-BHoM>.

The BHoM is designed to be extendable, which means that a toolkit can be created in an independent repository allowing for specific tasks to be carried out. This includes linking with external software or discipline specific functionality. For this thesis, a toolkit is created for non-linear finite element calculations.

The code can be used either by compiling repositories, which are found on GitHub, or by downloading an installer available on The Buildings and Habitats object Model website (BHoM, 2020). It is worth noting that most of the code is written in C# but there are some code in other languages also. Additionally there is an extensive wiki on GitHub for further reading (“Structure of the BHoM”, 2020).

### 6.3 FEM\_Toolkit

Our contribution is the *FEM\_Toolkit* which is developed to solve geometric non-linear problems used for structural systems. It extends the BHoM-core, with dependencies on the **oM**, **Engine** and **UI**. Subsequently, it is written in the same coding style as any of the available toolkits part of the BHoM code library.

As noted in section 3.3, FEM is formulated using matrix and vector operations and requires linear algebra operations to solve linear equation systems. Math.NET Numerics is an open-source library which provides numerical methods and algorithms which is used in the *FEM\_Toolkit*. It is distributed under the MIT/X11-License (“Math.NET Numerics”, 2020).

The *FEM\_Toolkit* consists of two categories of objects: *FEM\_oM* objects, where

properties and results are stored; *FEM\_Engine* objects, which takes care of computations and queries. The classes in the toolkit are displayed in table 6.1.

**Table 6.1:** FEM\_Toolkit classes

<b>FEM_oM</b>	
FEMResults	Stores results from the analysis.
Settings	Defines analysis settings.
<b>FEM_Engine</b>	
Assem	Assembles vectors and matrices from local degrees of freedof into global.
Edof_BC_fext	Creates topology-matrix (edof), boundary condition (bc) and external load-vector (fext).
GreenBarElementStress	Calculates element forces for a bar.
GreenBarForceVector	Creates internal element force vector.
GreenBarStiffness	Creates element bar stiffness matrix.
Partioning	Partitions global system into free dofs from constrained dofs in equation system.
Results	Creates BHoM results, such as BarDisplacement, BarForce, BarStrain, etc.
Run	Triggers the non-linear FEM procedure.
SolveEq	Solves the equation $\Delta \mathbf{a} = \mathbf{K}^{-1} \mathbf{R}$ and checks if solution has converged.

*Run* triggers the solver and requires the following BHoM-object input:

- Bar
- BarPreStressLoad
- Constraint6dof
- PointLoad

*FEMResults* is the output from *Run* and its values can be displayed with *BHoM.oM.Structure.Query* methods:

- DeformedShape
- Visualize
- PlotBarForce

## 6.4 Exposing FEM\_Toolkit to User-Interfaces

Since the *FEM\_Toolkit* is an extension to the BHoM core, its classes are automatically exposed to various user-interfaces supported by BHoM, such as Grasshopper, Excel and Dynamo. Figure 6.3 illustrates how this looks in Excel and Grasshopper. A benefit of exposing the code to existing software is not having to create a bespoke user-interface and at the same time allowing the user to use built in capabilities of existing interfaces.

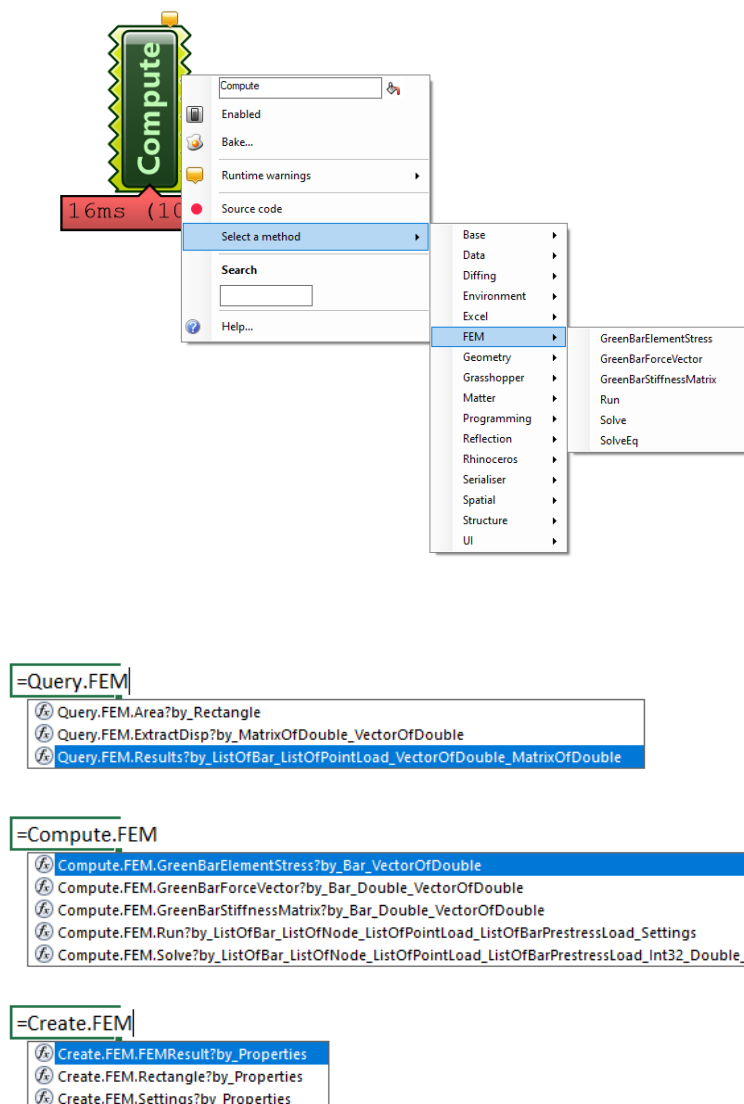


Figure 6.3: Exposure of FEM\_Toolkit in Grasshopper and Excel.





# 7

## Case-Study: Analysis of Audi A 50 Roof Structure

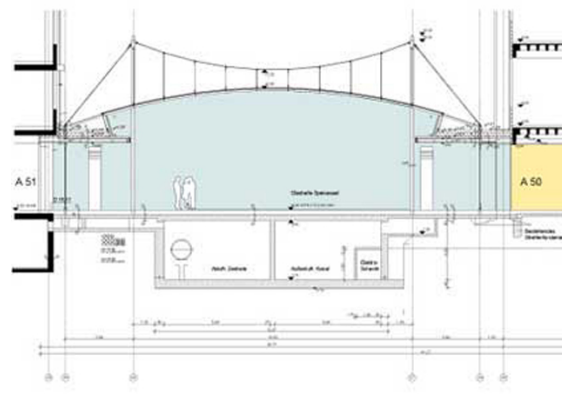
The FEM\_Toolkit is used on a two part case-study to show how the tool can be used in practical application. The studies are approached with different workflows to highlight the flexibility of the tool. This means that each case-study will have different design goals which require different workflows.

### 7.1 Background

In 1998, German architects A-N-S together with structural engineers Schlaich Bergemann Partner designed a lightweight glass roof between two existing buildings for Audi in Ingolstadt, Germany. The roof consists of a radial structure suspended by three-dimensional cable systems which are prestressed. The light structural system spans 24-meter, and consists of slender cross-section profiles, and enhances the sensation of lightness inside the room.



(a) Exterior view.



(b) Section drawing.

**Figure 7.1:** Audi A50 restaurant extension.

Source: *By permission of A-N-S Architects; <http://www.ans-architekten.de/>.*

## 7.2 Aim and Workflow

The aim of studying the roof is to show how the FEM\_Toolkit can be used for a real structure. The study does not strive to find an improved design or to verify the existing structure. Instead the study investigates how some questions, that might have risen during design process could be approached.

The FEM\_Toolkit is exposed to Rhino/Grasshopper which allows for parametric modelling to be used. This means that the structural model, including the initial geometry, section properties, loads and prestress, can be defined with a set of parameters and modified in the same pipeline. Moreover, post-processing and visualization of results are available to use and extract.

The general workflow, as illustrated in figure 7.3, consists of several steps. The process begins with an initial geometry which is form-found if not in equilibrium. The form-finding is followed by a non-linear analysis, where section- and material-properties, prestress, and external loads are defined. The next step is control if the design is feasible and if eventual modifications or optimisations are required. The extent of what is feasible is defined by the designer and can differ from case to case. The same model could also be exported to an external FEM-software for further studies.

Two studies are made, a 2D-case and 3D-case. Both cases follow the outlined workflow but differs in how modifications to the structure are made. The 2D-case is modified by letting the user change the input parameters, while the 3D-case input parameters are changed through an optimisation algorithm. Figure 7.2 shows how the workflow discussed above is implemented in Grasshopper using the same colour-code as in figure 7.3.

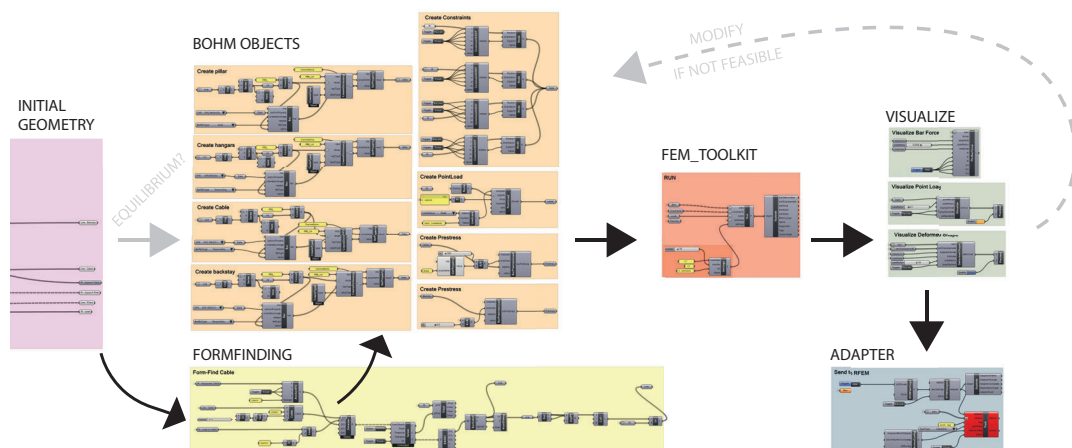


Figure 7.2: Workflow inside Grasshopper.

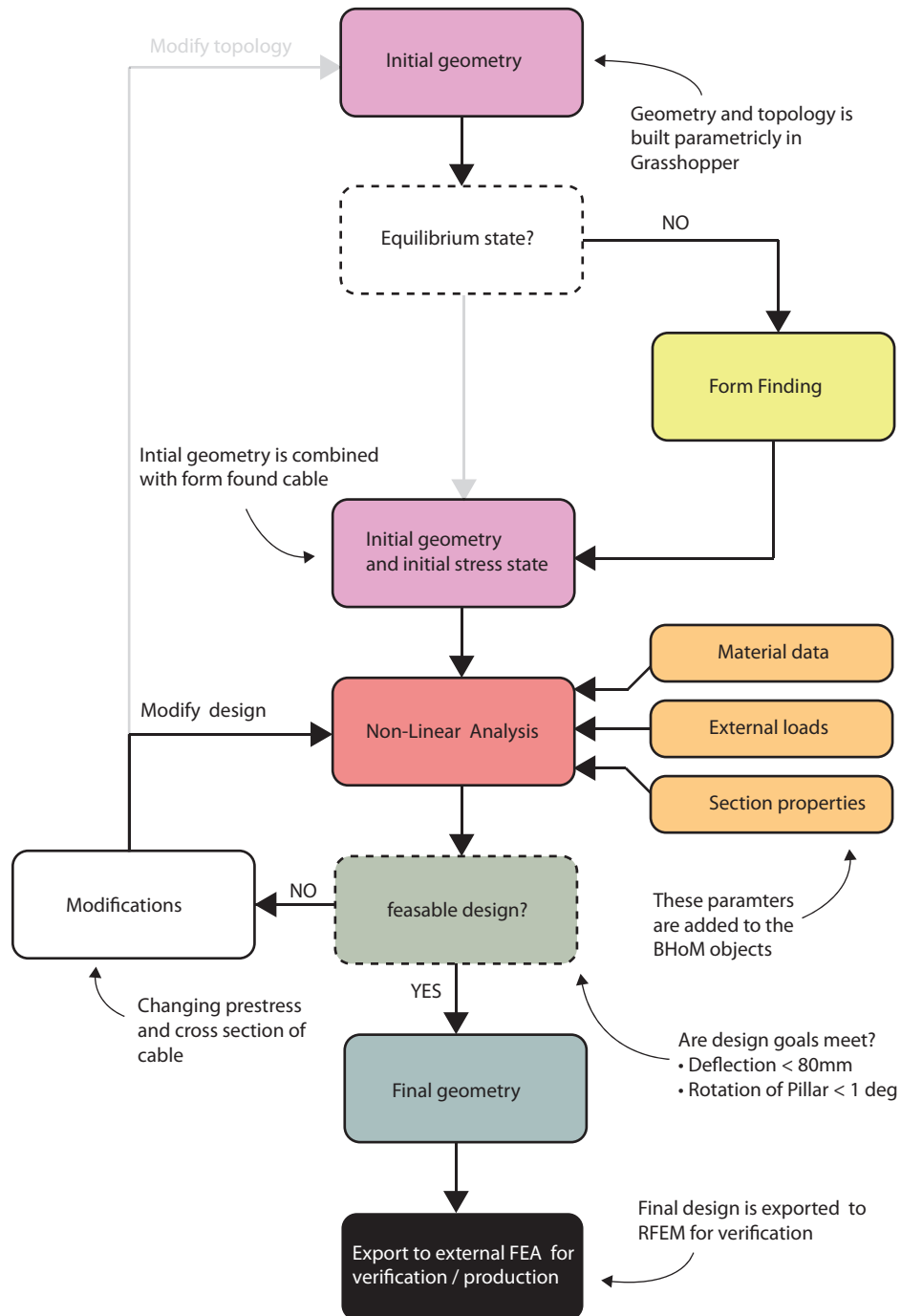
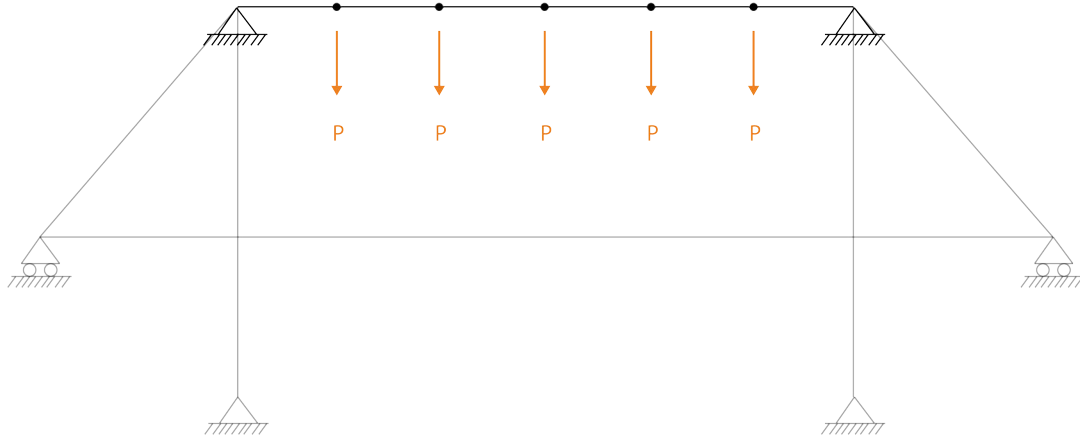


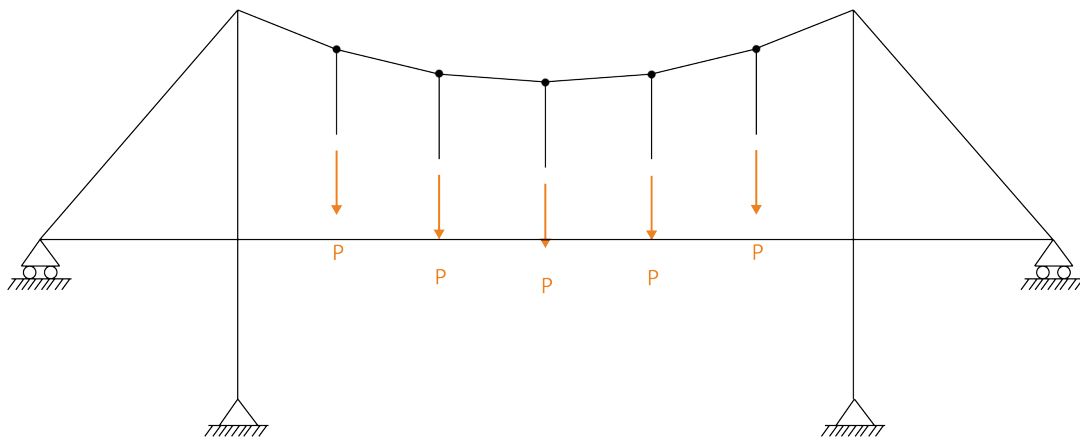
Figure 7.3: Workflow scheme.

### 7.3 Part 1: 2D-case

The first case-study is simplified into a 2D-structure where the arc-shaped roof is simplified into a horizontal bar. The initial task is to form-find a shape of a catenary in equilibrium, illustrated in figure 7.4. The assumed self-weight 11 kN of the roof is used and applied to the nodes. The form-finding results are illustrated in figure 7.5

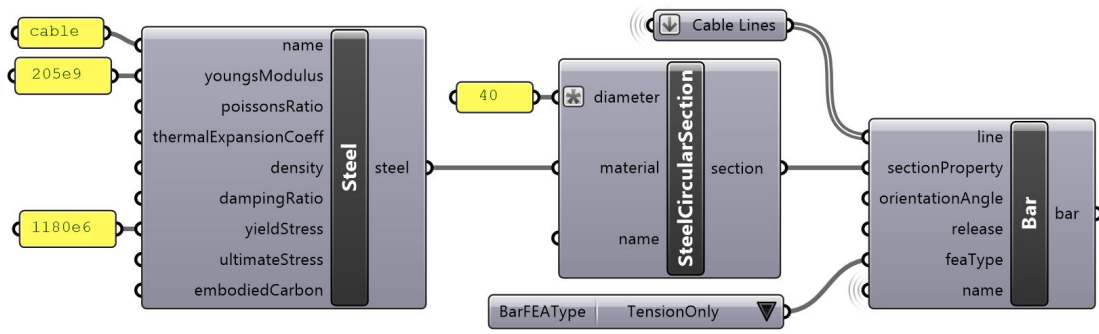


**Figure 7.4:** Form-finding of cable.

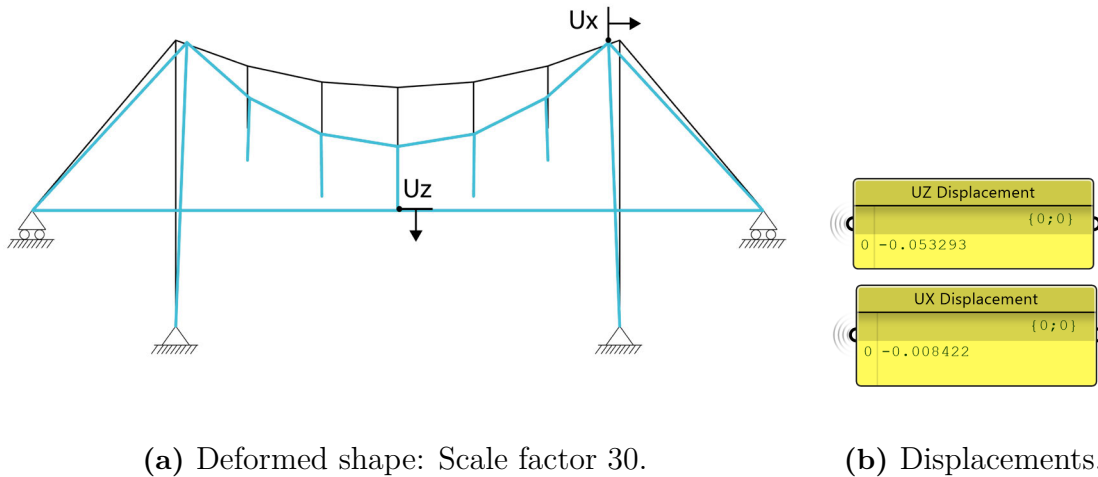


**Figure 7.5:** Geometry after form-finding.

The main cable is now in equilibrium and can be modelled with the rest of structure. The goal is to study how the entire structure reacts to the same applied loads. To do that, section and material properties must be defined. Figure 7.6 illustrates the definition of cables as tension only members with steel circular sections. The columns and the horizontal bar are defined as CHS 168.3x7.1 respectively CHS 244.5x16. The system is then solved without any defined prestress and the results are presented in figure 7.7.



**Figure 7.6:** Creation of steel material and circular cross-section.



(a) Deformed shape: Scale factor 30.

(b) Displacements.

**Figure 7.7:** Displacement without prestress as displayed by FEM\_Toolkit.

The nodal displacements in the main cable are large and the column is not standing vertically any longer. To minimize the large displacements prestress can be applied to the main cable and the backstay as illustrated in figure 7.8. The process of changing the prestress values using sliders is smooth and allows direct visual response. There are different possible combinations of prestress that limit the displacements. The visible prestress values in figure 7.9 results in a reduction of the maximum displacement to 8.2mm and is shown in figure 7.10.

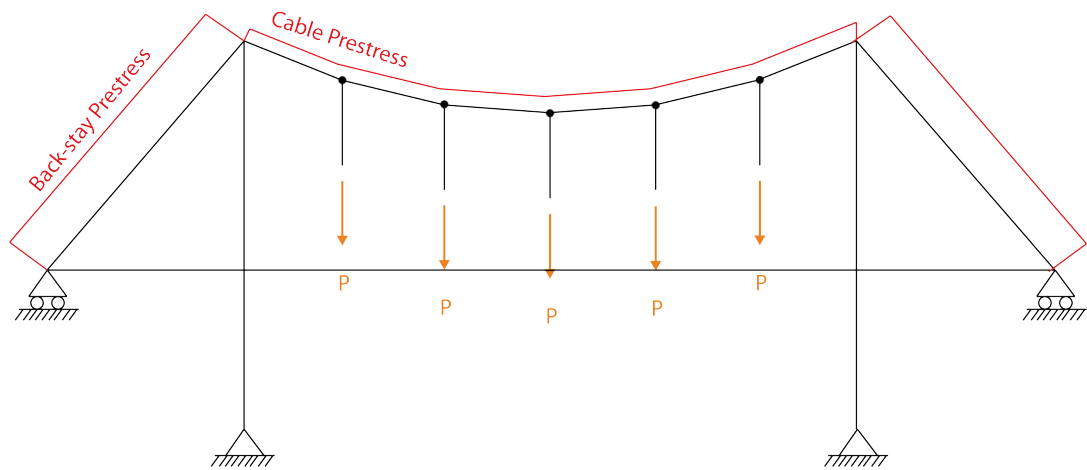


Figure 7.8: Application of prestress.

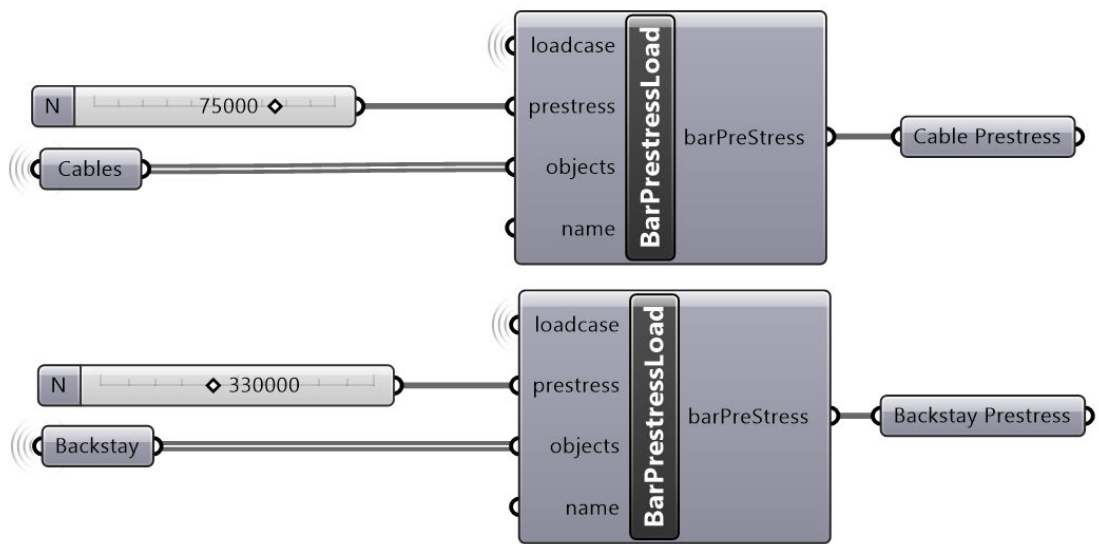
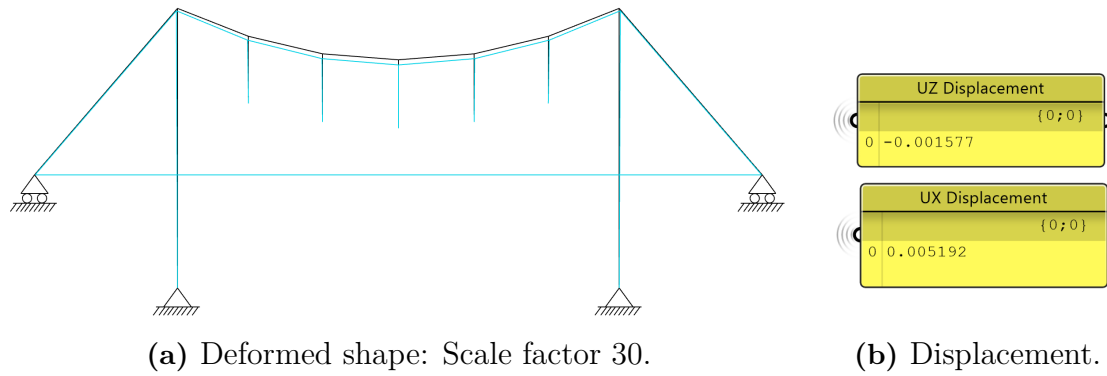
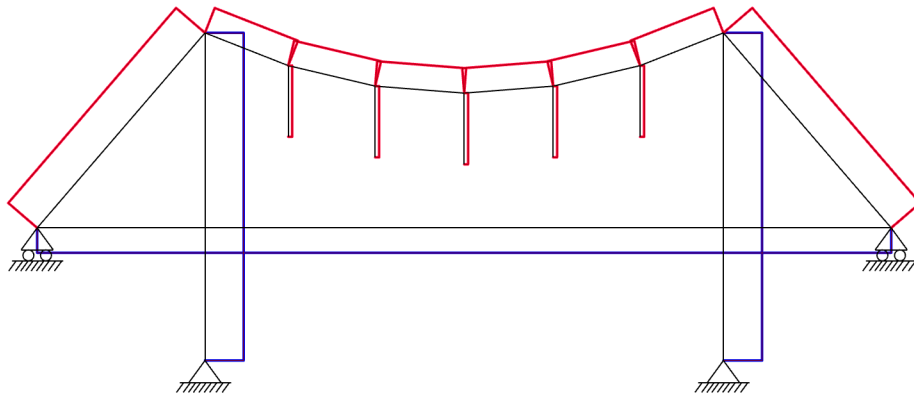


Figure 7.9: Definition of prestress.

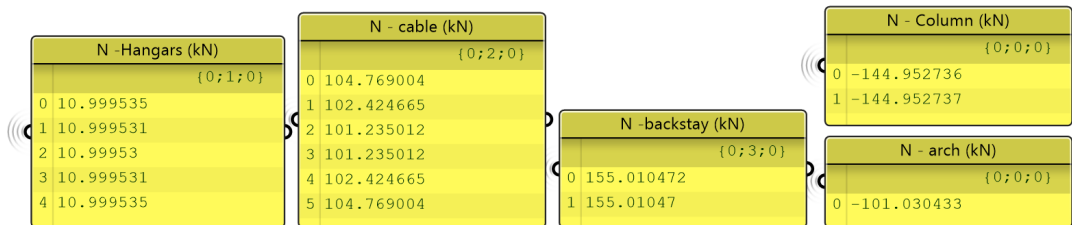


**Figure 7.10:** Displacements with prestress displayed by FEM\_Toolkit.

The sectional forces are displayed in figure 7.11 and 7.12. The model was exported to RFEM and it was found that the maximum deviation in sectional force between RFEM and NL-FEM is less than 1%.



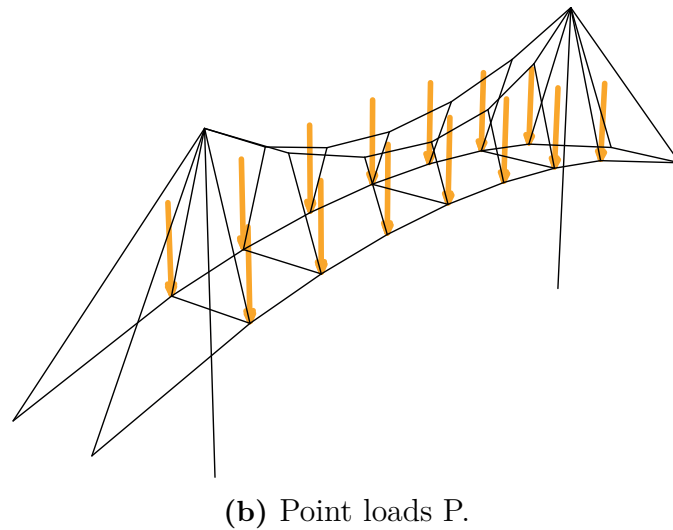
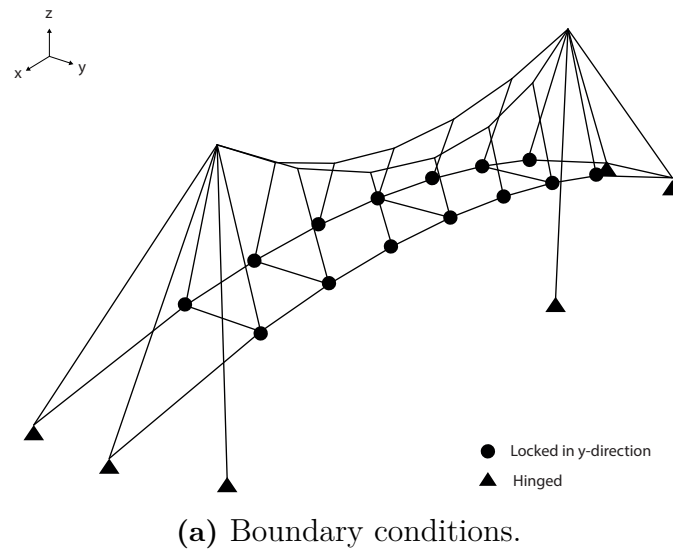
**Figure 7.11:** Normal forces with prestress displayed by FEM\_Toolkit.



**Figure 7.12:** Normal forces with prestress displayed in Grasshopper.

## 7.4 Part 2: 3D-case

The roof is extended to a more complex 3D-model as illustrated in figure 7.13. The horizontal bar is replaced with arcs connected to main cables by vertical hangers. Furthermore, the arc is restrained from moving in the  $y$ -direction. The structure is analysed with nodal loads,  $P = 22$  kN. The goal is to minimize the rotation angle of the columns, and nodal displacements, and the normal forces by prestressing the main cable, the backstays, and the hangers.



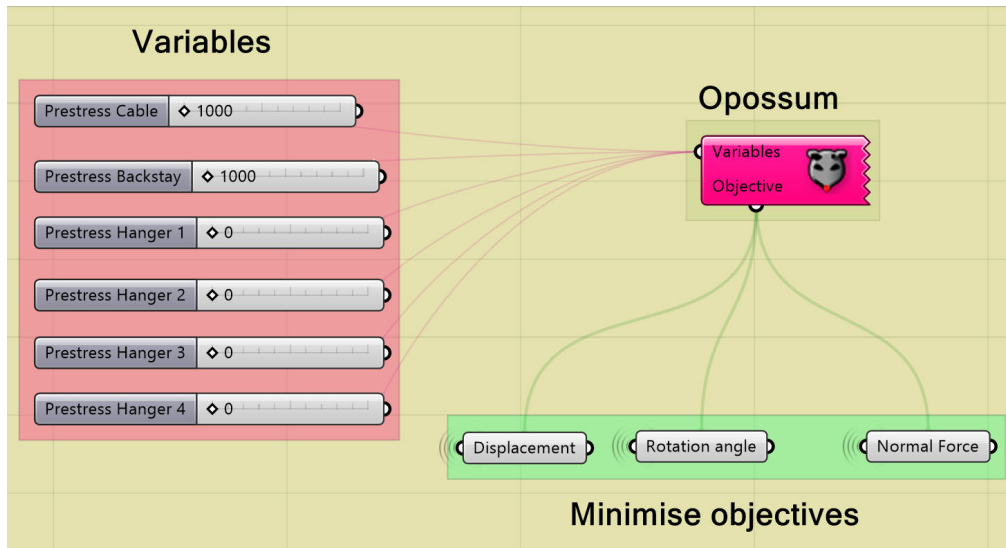
**Figure 7.13:** 3D-structural model.

### Optimisation

Part of a designer's role is to narrow several alternatives proposals into one feasible choice. While there exist many feasible options with respect to declared design goals, it is sometimes desirable to find an optimised solution.



In this case-study, the prestress in cable members are optimised with respect to three design goals. The multi-objective model-based optimisation tool Opossum is used (Wortmann, 2018). Since both Opossum and FEM\_Toolkit are available in Grasshopper, the tools are easily connected. The model-based algorithm implemented in Opossum has proven to be fast and stable and figure 7.14 shows how variables and objectives are defined in Opossum.



**Figure 7.14:** Definition of objective and variables in Opossum.

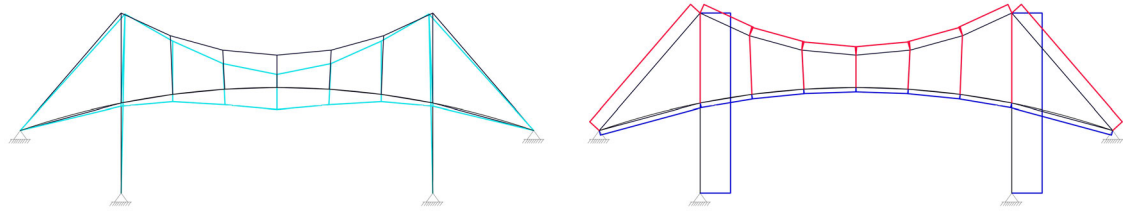
Opossum ran over 300 iterations and the results from each iteration was saved. The prestress of each cable was defined individually and are used as optimisation variables. The objectives of this optimisation are to minimize the following:

- **Objective 1** - Sum of displacement in the arc.
- **Objective 2** - Rotation angle of column.
- **Objective 3** - Sum of normal force in the main cables.

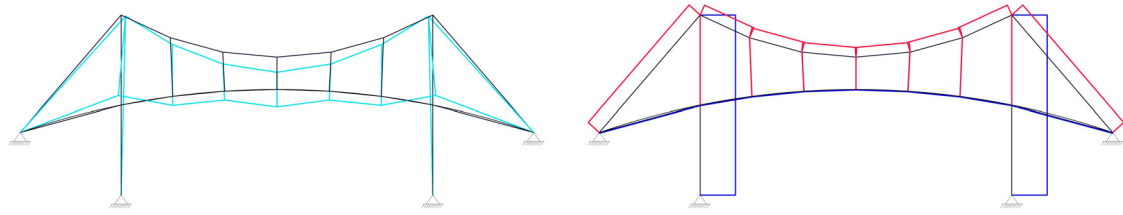
The results of iteration step 1, 10, 100 and 300 are displayed in table 7.1 and figure 7.15. It is noteworthy to mention that the optimisation converged quickly, and minor improvements were made in the later iterations.

**Table 7.1:** Results at different iteration step from Opossum

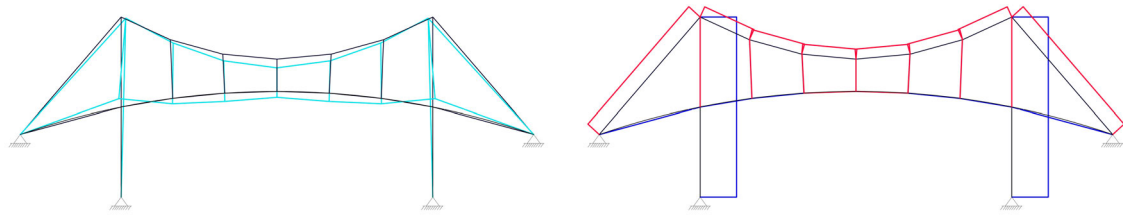
Iteration	Objective 1	Objective 2	Objective 3
1	160.679	-79.356	54.125
10	1.301	1.349	0.614
100	1.352	1.394	0.424
300	0.005	1.383	0.068



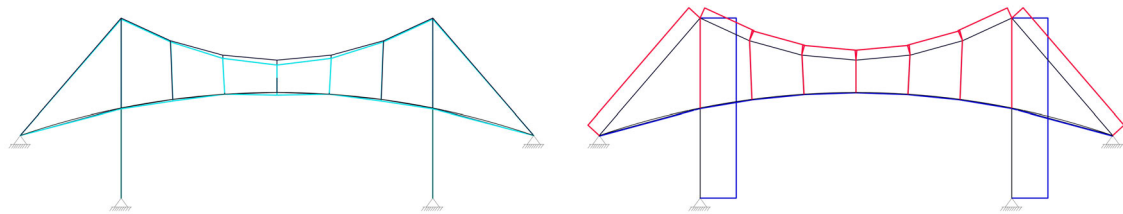
(a) Iteration 1, scale factor 0.1.



(b) Iteration 10, scale factor 10.



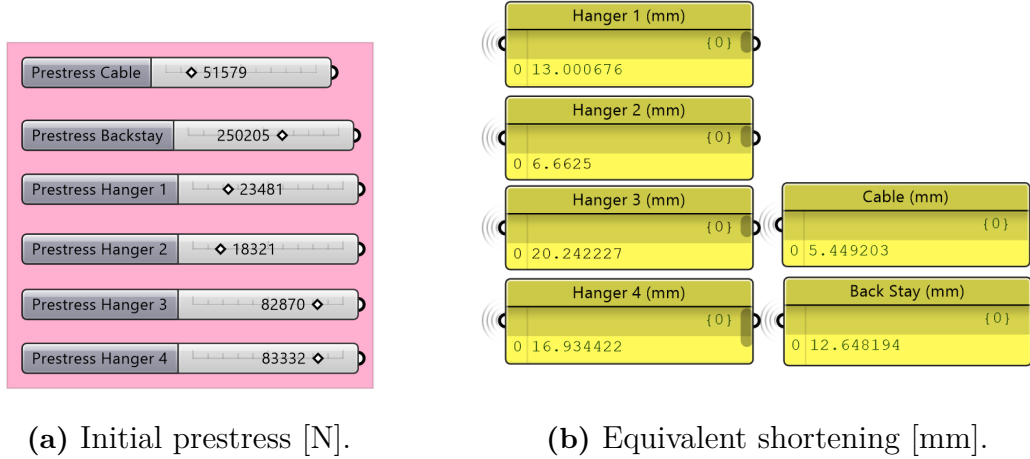
(c) Iteration 100, scale factor 10.



(d) Iteration 300, scale factor 10.

**Figure 7.15:** Displacements (left) and normal forces (right) at different iteration steps

The initial prestress in the tension elements can be translated to member shortenings and is illustrated in figure 7.16 below.



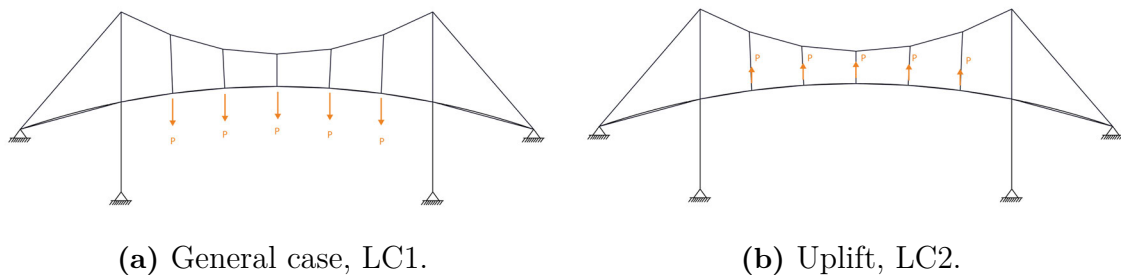
(a) Initial prestress [N]. (b) Equivalent shortening [mm].

**Figure 7.16:** Best set of variables found by Opossum after 300 iterations.

The final model was exported to RFEM for verification. It was found that normal forces differ less than 0.6%. The normal force in the column is 419.48 kN in RFEM compared to 422.0 kN in the FEM\_Toolkit. Deviations in the displacement values were smaller than 0.1%.

### Uplift

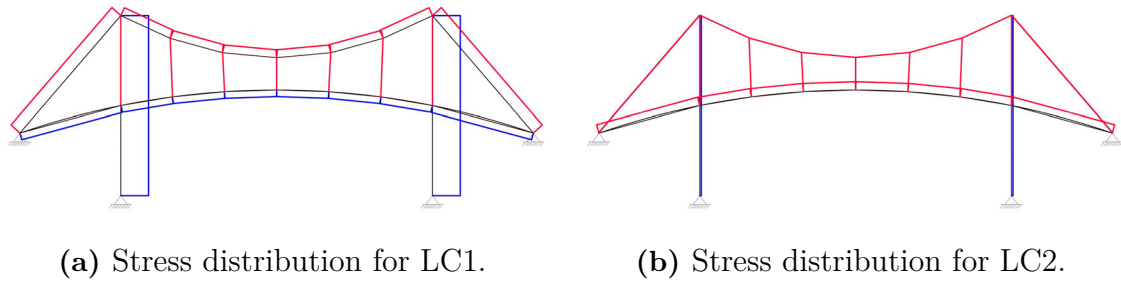
The last study on the Audio A50 roof structure is to optimise the prestress values based on two load-cases showed in figure 7.17. The first load-case (LC1) is identical to the previous load-case, with force  $P = 22$  kN. The second load-case (LC2) is an uplift scenario with point loads of 8 kN.



(a) General case, LC1. (b) Uplift, LC2.

**Figure 7.17:** The two studied load-cases.

The force patterns are different for the two load-cases, seen in figure 7.18. The arc is compressed in LC2 and tensioned in LC2 and the optimisation will be based on these diverging cases.

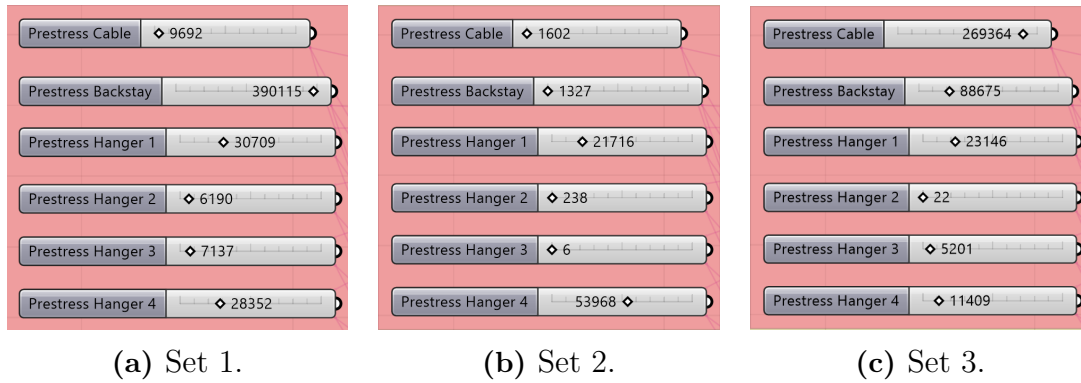


**Figure 7.18:** Stress distribution for the two load-cases.

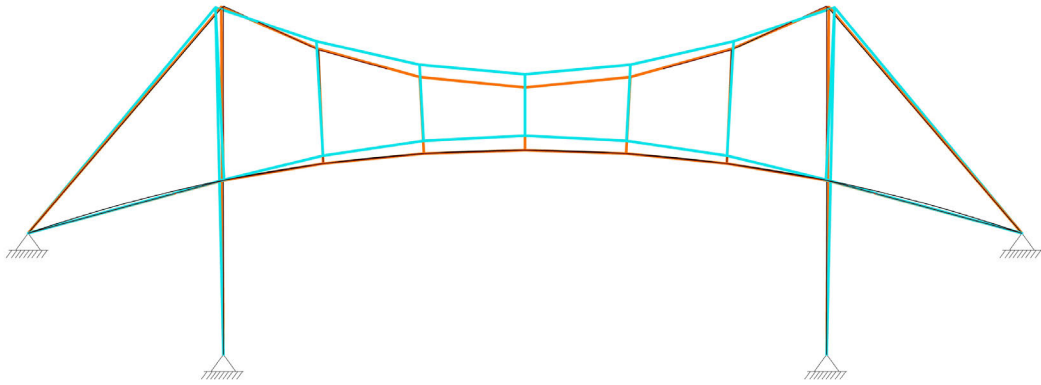
Opossum is used again to find a set of prestress configurations for both load-cases. The objective is to minimize the arc's vertical displacements. In addition, an optimisation is done with the objective to minimize the vertical displacements for both cases simultaneously. In total, three different optimisations are made and consequently three different prestress configurations are found:

- Set 1 - Optimised with objectives from LC1
- Set 2 - Optimised with objectives from LC2
- Set 3 - Optimised with objectives from both LC1 and LC2

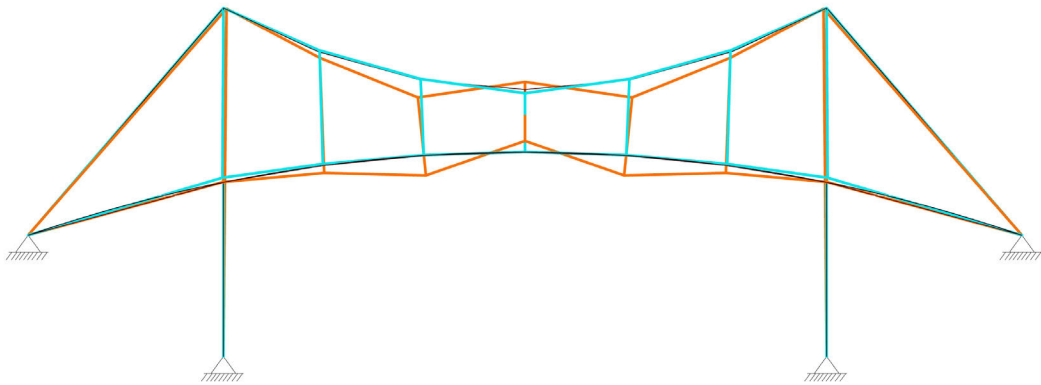
The three prestress configurations are displayed in figure 7.19 below.



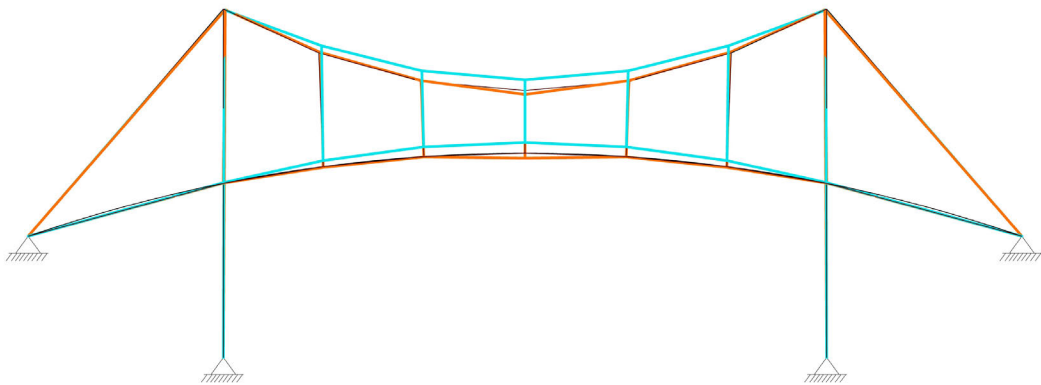
**Figure 7.19:** Three sets of prestress variables found with Opossum [N].



**Figure 7.20:** Deformations for LC1 (orange) and LC2 (cyan) with variables from set 1.



**Figure 7.21:** Deformations for LC1 (orange) and LC2 (cyan) with variables from set 2.



**Figure 7.22:** Deformations for LC1 (orange) and LC2 (cyan) with variables from set 3.

Table 7.2 shows the sum of the vertical displacements for the three different prestress sets for the two load-cases. Prestress values from set 1 and set 2 give minimum  $\sum U_z$  for their respective load-cases but maximum  $\sum U_z$  for the opposite load-case. Prestress variables in set 3 shows a trade-off between LC1 and LC2, resulting in prestress values that might fulfil the design goals for both load-cases.

**Table 7.2:** Comparison of set 1-3, based on sum of the arc's nodal displacements.

Load-case	Setup	$\sum U_z$
LC1	Set 1	0.00103
	Set 2	0.63913
	Set 3	0.07612
LC2	Set 1	0.24976
	Set 2	0.0597
	Set 3	0.19056

## 7.5 Conclusion

A two-part case-study was preformed to demonstrate how the implemented FEM\_Toolkit could be used to analyse complex structures. The implementation to Grasshopper allowed the toolkit to be combined with other tools and made it possible to implement workflows with either an heuristic approach or a more advanced optimisation. Moreover, the extension to BHoM gave the ability to easily share data with other software which could be used to verify the results.

# 8

## Conclusions

In this project, we set out to develop a computational tool called FEM\_Toolkit to be used in conceptual design phase for cable supported structures. To allow for several design proposals to be investigated, the tool has been exposed to CAD-environments such as Rhinoceros. Moreover, the tool had to be flexible enough to support varying workflows and programmed in a way so that it can be extended with desirable functionalities. The FEM\_Toolkit implements a Newton-Raphson finite element solver for Green strain bar elements and is thus capable of solving large-displacement problems with or without prestress.

In cable supported structures, form and forces are inseparable and requires a close collaboration between architects and engineers. By exposing the tool to user-interfaces known to both architects and engineers, it is our belief that collaboration can be become easier. By facilitating the cooperation, our aim to is to allow for the engineer to take an active role in the conceptual phase.

The implemented solver was chosen after evaluating three numerical methods for large-displacement problems: the dynamic relaxation method (DRM); the force density method (FDM); and, the non-linear finite element method (NL-FEM) with Newton-Raphson and Green strain bar elements. While all three proved capable of solving form-finding problems, only DRM and NL-FEM could be used in analysis. However, DR was ruled out in favor for NL-FEM, which is better known among practicing engineers.

The FEM\_Toolkit is software agnostic using open-source libraries such as BHoM and MathNet.Numerics. The FEM\_Toolkit could be exposed to numerous user-interfaces used both by architects and engineers including Grasshopper and Dynamo. Our tool is open-sourced in the same spirit as the used libraries, and is available to download from GitHub (see Appendix A) and can be modified by others.

With the tool, it is possible to meet design objectives by using different workflows, including the use of prestress to reduce displacements and the use of multi-objective optimisation algorithm.

Due to the diverse nature of projects, a promising way forward is to allow for extension to be made based on project needs. The tool can be clarified and extended for wider applications. Some of the suggestions below can be of interest for future research and some are practical additions to the toolkit:

- **Incorporate more element types.** The tool can be extended to support beam and shell elements to make it more useful for more complex structures.
- **Load-cases and Load-combinations.** The tool can be extended to support line loads, temperature loads, and nodal displacements. It is also important to consider multiple load-combinations in structural design (e.g. ULS and SLS). The case-studies revealed that dealing with multiple load-combinations in the tool can be improved.
- **Component descriptions.** Functionality description of each method could be added to the code. Moreover, a clear documentation explaining how to run the code can also be provided.
- **Solver feedback.** To increase the transparency, the toolkit can be extended with a feedback-output. This could ease the process of detecting error without the need of using software such as Visual studio.
- **Arc-length iterations.** Implementation of a different numerical iteration algorithm to avoid (snap-back/snap-through) convergence errors.
- **Dynamic behaviour.** Light-weight structures are prone to vibrations and dynamic studies are usually required. The toolkit could be extended to support static eigenfrequency analysis.
- **Prestress sequence.** The sequence of prestressing more complex cable supported structures can be complicated and requires phased analysis.



# Bibliography

- Addis, W. (1990). *Structural engineering: The nature of theory and design*. Ellis Horwood Ltd.
- Adriaenssens, S., & Barnes, M. R. (2001). Tensegrity spline beam and grid shell structures. *Engineering Structures*, 23(1), 29–36. doi:[https://doi.org/10.1016/S0141-0296\(00\)00019-5](https://doi.org/10.1016/S0141-0296(00)00019-5)
- Adriaenssens, S., Block, P., Veenendaal, D., & Williams, C. (2014). *Shell structures for architecture: Form finding and optimization*. Routledge.
- Barnes, M. R. (1977). *Form finding and analysis of tension space structures by dynamic relaxation*, City University London. Retrieved from <https://openaccess.city.ac.uk/id/eprint/11887/>
- Barnes, M. R., Adriaenssens, S., & Krupka, M. (2013). A novel torsion/bending element for dynamic relaxation modeling. *Computers & Structures*, 119, 60–67. doi:<https://doi.org/10.1016/j.compstruc.2012.12.027>
- Bhatti, M. (2006). *Advanced topics in finite element analysis of structures: With mathematica and matlab computations*. Wiley.
- BHoM. (2020, April 6). The buildings and habitats object model. Retrieved from <https://bhom.xyz/>
- Bletzinger, & Ramm. (2001). Structural optimization and form finding of light weight structures. *Computers and Structures*, 79.
- Block, P., Van Mele, T., Rippmann, M., & Paulson, N. (2017). *Beyond bending: Reimagining compression shells*. Edition Detail.
- Carfrae, T., & Michael, D. (2019a). Alistair day and the origins of dynamic relaxation. part 1: Bringing an idea to life. *The Structural Engineer: journal of the Institution of Structural Engineer*, 97(6), 20–26.
- Carfrae, T., & Michael, D. (2019b). Alistair day and the origins of dynamic relaxation. part 2: A practice viewed through projects. *The Structural Engineer: journal of the Institution of Structural Engineer*.
- CEN. (2006). Design of steel structures, part 1-11: Design of structures with tension components. European Committee for Standardization.
- Crisfield, M. A. (1993). *Non-linear finite element analysis of solids and structures*. Wiley New York.
- Culmann, C. (1875). *Die graphische statik*. Meyer & Zeller.
- Day, A. (1965). An introduction to dynamic relaxation. *The Engineer*.
- Duxbury, P., Crisfield, M., & Hunt, G. W. (1989). Benchtests for geometric nonlinearity. *Computers & structures*, 33(1), 21–29.
- Engel, H. (1997). *Structure systems: Tragsysteme*. Gerd Hatje.

- Guest, S. (2006). The stiffness of prestressed frameworks: A unifying approach. *International Journal of Solids and Structures*, 43(3-4), 842–854.
- Haber, & Abel. (1982). Initial equilibrium solution methods for cable reinforced membranes part 1 - formulations. *Computer methods in applied mechanics and engineering*, 30.
- Hejlsberg, A., Torgersen, M., Wiltamuth, S., & Golde, P. (2008). *The c# programming language*. Pearson Education.
- Hsu, W. (2000). Conceptual design: Issues and challenges. *Computer-Aided Design*.
- Javier Sanchez, P. M., Miguel Angel Serna. (2006). A multi-step force-density method and surface-fitting approach for the preliminary shape design of tensile structures. *Engineering structures*, 29.
- Liberty, J. (2005). *Programming c#: Building .net applications with c. "* O'Reilly Media, Inc."
- Linkwitz, K. (1999). About formfinding of double-curved structures. *Engineering Structures*, 21.
- Llunji, M. (2020). *Towards a new engineering*. MSPROJECT. Retrieved from <https://www.xarg.org/ref/a/9940665016/>
- Luo, Y., & Lu, J. (2006). Geometrically non-linear force method for assemblies with infinitesimal mechanisms. *Computers & structures*, 84(31-32), 2194–2199.
- Math.NET Numerics. (2020, May 14). Retrieved from <https://numerics.mathdotnet.com/>
- Oden, J. T. (1990). Historical comments on finite elements. *A history of scientific computing*, 152–166.
- Olsson, K.-G., & Dahlblom, O. (2016). *Structural mechanics: Modelling and analysis of frames and trusses*. John Wiley and Sons Ltd.
- Pauletti, R. M., & Pimenta, P. M. (2008). The natural force density method for the shape finding of taut structures. *Computer Methods Applied Mechanics Engineering*.
- Peters, T. F. (2012). *Transitions in engineering: Guillaume henri dufour and the early 19th century cable suspension bridges*. Birkhäuser.
- Pipinato, A. (2015). *Innovative bridge design handbook: Construction, rehabilitation and maintenance*. Butterworth-Heinemann.
- Schek, H.-J. (1974). The force density method for form finding and computation of general networks. *Computer methods in applied mechanics and engineering*.
- Schlaich, J. (2004). Conceptual design of light structures. *Journal of the International Association for Shell and Spatial Structures*, 45(3), 157–168.
- Schlaich, M. (2006). Challenges in education - conceptual and structural design. *IABSE Symposium Report*, 92, 20–26. doi:10.2749/222137806796168840
- Sehlström, A. (2019). *Prestress in nature and technics* (Thesis for Degree of Licentiate of Architecture, Architecture and Engineering Research Group, Chalmers University of Technology).
- Stevin, S. (1586). De beghinselen der weeghconst. *The Principal Works of Simon Stevin*, 1.
- Structure of the BHoM. (2020, May 15). Retrieved from <https://github.com/BHoM/documentation/wiki/Structure-of-the-BHoM>

- Timoshenko, S. (1983). *History of strength of materials: With a brief account of the history of theory of elasticity and theory of structures*. Courier Corporation.
- Varignon, P. (1725). *Nouvelle mécanique ou statique, dont le projet fut donné en m.dc.lxxvii*. ETH-Bibliothek Zurich, [www.e-rara.ch](http://www.e-rara.ch).
- Veenendaal, D., & Block, P. (2012). An overview and comparison of structural form finding methods for general networks. *International Journal of Solids and Structures*, 49(26), 3741–3753. doi:<https://doi.org/10.1016/j.ijsolstr.2012.08.008>
- Wakefield, D. S. (1980). *Dynamic relaxation analysis of pretensioned networks supported by compression arches*. Unpublished Doctoral thesis, City University London.
- Wakefield, D. S. (1999). Engineering analysis of tension structures: Theory and practice. *Engineering structures*, 21(8), 680–690.
- Wang, L. (2001). Collaborative conceptual design - state of the art and future trends. *Computer-Aided design*.
- Ward, F. K. (1921). *In farthest burma: The record of an arduous journey of exploration and research through the unknown frontier territory of burma and tibet*. Seeley, Service & co., limited.
- Wortmann, T. (2018). Genetic evolution vs. function approximation: Benchmarking algorithms for architectural design optimization. *Journal of Computational Design and Engineering*, 6. doi:[10.1016/j.jcde.2018.09.001](https://doi.org/10.1016/j.jcde.2018.09.001)



# A

## Code Access

Below are some links to access the code for the FEM\_Toolkit, BHoM and MathNet.Numerics:

- Link to FEM\_Toolkit: [github.com/SaterAhmad/FEM\\_Toolkit](https://github.com/SaterAhmad/FEM_Toolkit)
- Link to BHoM: [github.com/BHoM](https://github.com/BHoM)
- Link to MathNet.Numerics: [github.com/mathnet/mathnet-numerics](https://github.com/mathnet/mathnet-numerics)