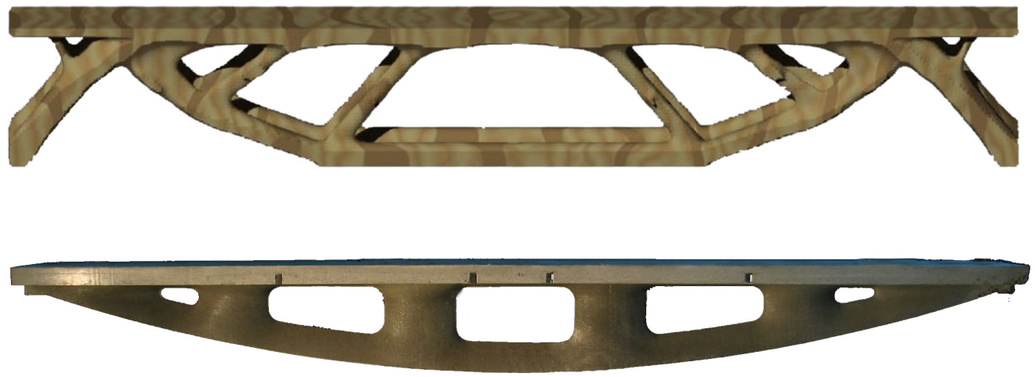
Evolutionary Optimization of Fabric Formed Structural Elements

Evolutionary Optimization of Fabric Formed Structural Elements

Bridging the gap between computational optimization and manufacturability

Master's thesis of Civil Engineering



Diederik Veenendaal
c1041320
June 2008

structural design lab

Faculty of Civil Engineering and Geosciences

TU Delft
Delft University of Technology

Evolutionary Optimization of Fabric Formed Structural Elements

Bridging the gap between computational optimization and manufacturability

Diederik Veenendaal

c1041320

June 2008

Disclaimer

Published in the Netherlands

First edition published 2008

Preface

This is the final report of the Master's thesis project of D. Veenendaal, Master student at the faculty of Civil Engineering and Geosciences of the Delft University of Technology (TU Delft). The subject of this thesis is "Evolutionary Optimization of Fabric Formed Structural Elements: Bridging the gap between computational optimization and manufacturability".
This report is intended for people in the field of architecture, structural engineering and building engineering and some related fields of mathematics, computer sciences, etc. The thesis combines the specific areas of fabric formwork technology, fabric form finding i.e. dynamic relaxation, computational optimization i.e. bi-directional evolutionary optimization and differential evolution as well as structural analysis, specifically finite element analysis. The reader would benefit from some knowledge in these areas, although each of these concepts is discussed and explained in some detail.

The graduation committee consisted of the following members:

Prof. dipl. ing. J.N.J.A. (Jan) Vamberský

e-mail: J.N.J.A.Vambersky@tudelft.nl
tel: +31 (0)15 27 85488
address: Delft University of Technology, Civil Engineering and Geosciences
Department of Design & Construction, section of Structural and Building Engineering
Stevinweg 1, room 1.53 St. II
2628CN Delft, Netherlands

Ir. J.L. (Jeroen) Coenders

e-mail: J.L.Coenders@tudelft.nl
tel: +31 (0)15 27 85711
address: Delft University of Technology, Civil Engineering and Geosciences
Department of Design & Construction, section of Structural and Building Engineering,
Stevinweg 1, room 1.60 St. II
2628CN Delft, Netherlands

Dr.ir. P.C.J. (Pierre) Hoogenboom

e-mail: P.C.J.Hoogenboom@tudelft.nl
tel: +31 (0)15 27 88081
address: Delft University of Technology, Civil Engineering and Geosciences
Department of Design & Construction, section of Structural Mechanics
Stevinweg 1, room HG 6.48
2628CN Delft, Netherlands

Dr.ir. C. (Cor) van der Veen

e-mail: C.vanderVeen@tudelft.nl
tel: +31 (0)15 27 84577
address: Delft University of Technology, Civil Engineering and Geosciences
Department of Design & Construction, section of Structural and Building Engineering
Stevinweg 1, room 2.05 St. II
2628CN Delft, Netherlands

A word of introduction

by professor Mark West

UNIVERSITY
OF MANITOBA

In our building culture, where calculation must always precede construction, the elegance and simplicity of the engineer's calculation method is a prerequisite to elegant and economical construction. If something is difficult for an engineer to predict, it will probably not get built, regardless of how effective, economical, or beautiful the construction method may be. In the case of cast concrete construction, there are two predictions required -- the behavior and performance of the reinforced concrete itself, as well as the behavior and performance of the mold structure forming the concrete. Doing this for conventional uniform-section reinforced concrete beams cast in rigid panelized molds is of course entirely conventional now, though it is worth remembering that this was, at one time, quite a difficult puzzle. While we are heir to the solutions found for conventional concrete beams, variable-section fabric-formed concrete beams (the central subject of this thesis) re-presents both puzzles at once to any engineer entering this field.

The fabric itself, of course, has no problem at all calculating its final three-dimensional form. A loaded membrane is an infallible physical calculator of the shape of its own resistance. The difficulty of predicting the shape given by a flexible mold is ours alone, and in this case it is coupled to the behavior and performance of the reinforced concrete beam taking the membrane's shape. This doubled puzzle, however, is certainly worth pursuing. The structural and material elegance of a tension mold, and the potential structural elegance of a variable section beam, present the prospect of a revolutionary simplification in construction and a commensurate reduction in the materials and energy consumed in concrete construction. Aesthetically, and architecturally, the prospect of flexibly formed structures and buildings presents nothing less than an entirely new language of form. Although not the concern of this thesis, one should not discount the central importance of human pleasure which is also at stake. The beauty inherent in 'naturally' formed structures following the dictates of natural law remains an important prize to be gained at the end of our technical labors.

But first the technical problems must be solved, and these solutions, it should be noted, are quite beyond the abilities of the inventors of fabric-formed concrete structures. The methods in play were all invented and developed by architects and builders playing with materials and physical processes. Quantitative analysis is the realm of engineering, and it is here that the potentials of efficiently curved concrete structures will be realized. This thesis is an early work in constructing a new field of engineering and construction knowledge. It's a precursor of things to come and will be referenced by a growing number of researchers and practitioners being drawn to the promise of this new technology.

At present, full-scale fabric-formed commercial constructions have been limited to walls, columns, and foundations. All of these elements require only simple and self-evident engineering calculations. Unlike columns and walls however, a curved beam cannot rely on standard calculation methods for its design. Because economical molds for reinforced concrete beams have been historically limited to uniform-section rectangular prisms, standard engineering design and calculation methods have been developed specifically for uniform section members. Established design procedures, codes, and software, are all based on the assumption of a uniform section. First physical principles tell us that higher efficiencies can be obtained by following "natural" curved force paths, and we know these curved forms can be easily constructed using a fabric membrane, but some deeper thinking is required to discover how to design curved reinforced concrete beams. This thesis is an early effort in this necessary work.

The difference in methodology between the invention of this technology and the mastery of its engineering is significant. Two distinct research cultures are at work from "either end" of this problem. Their union in a single enterprise repre-

sents a very potent collage (or collaboration). The work done at the Centre for Architectural Structures and Technology (C.A.S.T.) in developing mold-making methods for fabric-formed concrete structures has relied almost exclusively on physical models and empirical experiments. Small physical models using light nylon materials and plaster are used to model full-scale constructions using strong industrial fabrics and concrete. There are several important advantages to working in this way, foremost among them being the fidelity with which these little models can predict the physical behavior of their full-scale equivalents. Mold methods developed in very small physical models can be directly scaled up to much larger constructions with great success. Recently, for example, a 30 cm high column mold model was scaled up to a 3 m tall full-scale mold with no intermediate steps. Indeed, we have found that anything that can be built in a model can be constructed at full-scale. This use of physical models allows a non-punitive environment in which thinking and discovery can take place very quickly and inexpensively. This allows for great cognitive freedom. The method is a bit like shooting a shotgun against a wall and then drawing bulls-eyes around the holes – every event, intended or not, is an opportunity to study what will happen in an equivalent full-scale construction.

We can say that physical models are qualitatively rich, allowing instant feedback on the effects of a physical system as loads are applied and force or stiffness is changed. But physical models are quantitatively poor; it is very difficult to get any kind of measurable data from a physical model (the analog data collection methods of the shell engineer Heinz Isler come to mind). Mathematical models, on the other hand, are quantitatively rich – indeed they are composed of quantities, and quantity is now the primary substance required for this work to advance. The path for future work is clear: the mathematical predictions of this thesis, which studies technologies produced by inventive play in the physical world will, in turn, be subjected to physical tests. The knowledge thus gained from the physical world will once again be metabolized by mathematical analysis in future generations of engineering research. This thesis is a first generation engineering work in a new field. It will be studied and followed by others in the tradition of engineering as a large collective enterprise. The prize, at the end, is nothing less than a new way of building and the empowerment of a new architectural language.

Professor Mark West

Faculty of Architecture
Centre for Architectural Structures and Technology (C.A.S.T.)
University of Manitoba, Winnipeg, Canada

Acknowledgements

The author would like to thank and acknowledge the following persons who have offered their assistance, ideas and support during the course of this thesis project.

First, I would like to thank my graduation committee who have patiently supported this project and offered their guidance whenever it was needed. I would like to commend them on their enthusiasm during our meetings which contributed to keeping myself motivated. Professor J. Vambersky always offered constructive criticism in the most positive ways and made sure that the thesis stayed result-oriented. In addition, he requested that I present my work at a meeting of the *federation internationale du beton*, one month prior to the final presention of this project. This display of confidence in my work in turn boosted mine, which helped during the final weeks of the project.
To similar effect ir. J.L. Coenders showed some of my work at various companies in Britain during a study tour which I personally did not take part in. Furthermore, he is the main proponent of exploring novel strategies in building engineering at this faculty, in the form of the Structural Design Lab. In doing so, he has enabled me to work on an inspiring and innovative subject. He has continuously offered his advice on the progress of the project while giving me the time and support I felt was needed to complete the project to my satisfaction.
Throughout this project I could always rely on ir. P. Hoogenboom for being available to answer any questions on mechanics, helping me to better understand the algorithms I worked with and the results that I gained from them. He has remained appreciative of the effort involved in computer programming and understanding of its many pitfalls.
Halfway through this project, the fourth member of the committee, ir. C. van der Veen, was added to provide his perspectives on the concrete engineering within this subject. He has on several occasions guided me in the modeling and analysis of concrete. In addition, he has been very outspoken on his enthusiasm for fabric formwork and the work that I have done.

Second, I would like to thank Professor Mark West from the faculty of Architecture at the University of Manitoba in Winnipeg, Canada. He has provided me with all the information on fabric formwork that I could have hoped for and often returned my e-mails within days. Because he has not been involved with this project in any official capacity, he has never ceased to amaze me in his willingness to answer my questions or if he himself could not, to refer me to the right people. It is my wish that this thesis may contribute in his ongoing efforts to develop and expand the technologies of fabric formwork.

Third, I would like to thank all my fellow graduate students at room 0.72 for inspiring me with their own work, keeping tabs on my work ethic – using the 'goed-stabiel-faalhaas' system – and my progress, – using our beloved staircase – helping me with any problems I could not solve myself and providing me with the occasional laughs and friendly banter. Without them my final months at this university would not have been as much fun.

Fourth, I would like to thank everyone at Structural Design Lab for being such an innovative group of people, either carrying out or supporting ingenuous work in the field of structural engineering. They have introduced me to many new concepts and methods I was previously unaware of.

Fifth, I would like to thank professor L.A.G. Wagemans for inspiring me to choose such a novel and interesting subject. Unbeknownst to him, one of his slides during one of his courses in my second year never left my thoughts and led me to join the Structural Design Lab. The slide itself was the 'bone bridge' by dr. C.J.K. Williams of the University of Bath, United Kingdom.

Finally, I would like to thank all my friends and family for supporting me during the course of this project.

Summary

The field of structural engineering has seen a steady rise in computational optimization, where optimization algorithms attempt to find more optimal solutions to structural design problems. Although some of these optimization tools have found their way to the engineering practice, they generally extend no further than the earliest design stages, where they function as tools that provide only indicative geometries for the final design. Herein lies the possibility of incorporating manufacturability in the optimization process, whereas currently it is only considered afterwards. This thesis attempts to bridge the gap between computation optimization and manufacturability by developing a computer tool which produces manufacturable results.

In the initial stages of this project suitable optimization algorithms and manufacturing techniques were considered.

Bi-directional Evolutionary Structural Optimization, or BESO, was first explored, as it is an algorithm geared towards structural engineering with several known examples of its use. The algorithm was succesfully programmed and implemented in the finite element program ANSYS, where it was used to optimize several beams under different support conditions. One of the driving parameters, called the Removal Rate of Volume, or RRV, was changed by the author to produce a faster, more robust algorithm. The results were interesting but limited to linear material properties. Working with BESO also revealed that it is fairly straightforward and not as flexible to easily allow inclusion of manufacturing constraints. The decision was made to select a general optimization algorithm, Differential Evolution. Not abandoning BESO completely, an attempt was made to include material non-linearity by creating a flexible process of estimating an amount of longitudinal reinforcement and applying it in ANSYS. The way in which BESO evaluates each result, i.e. the calculation of the Performance Index, was adapted to reflect the non-linear analysis as well. The results of this altered BESO algorithm produced significantly different geometries than previous linear models.

The manufacturing technique of choice is a fairly new development, called fabric formwork technology. Fabric formwork is – simply speaking – the casting of concrete in prestressed polymer fabric. This method allows very different geometries than those typically associated with concrete element cast in conventional molds. Through an interaction of the supports of the fabric, its non-linear material properties, the applied prestressing and the pressures of the fresh concrete, an equilibrium arises that determines the shape of the final product. The inherent difficulties in predicting the geometry of the fabric formed elements, as well as the potential of fabric formwork to produce new and interesting shapes posed intriguing questions. For these reasons it was decided to incorporate this method in this project. The name of the program that was developed during this thesis – named after this manufacturing method – is FABRICFORMER.

To calculate the shape of the fabric within the optimization process, a form finding algorithm had to be coded, for which the Dynamic Relaxation algorithm was chosen. This algorithm, applied to many non-linear problems in different fields, is also used for the calculation of tension structures such as tents. Differences to this application with that of fabric formwork is the interaction between the fabric and the solid parts of the mold. To cope with this, the Dynamic Relaxation algorithm – which was coded in Java – was adapted to include collision detection as well as pressure loads from the concrete. Eventually, the algorithm was succesfully applied to model all three existing mold types for fabric formed beams.

At this point, structural analysis was added to the program, by having FABRICFORMER communicate with ANSYS. Java code was written to initiate batch runs of ANSYS within FABRICFORMER and succesfully transfer data between the two. The translation of the planar fabric mesh, created in FABRICFORMER and used for Dynamic Relaxation, to a volume mesh for ANSYS was particularly difficult as two different types of translations were developed and implemented. Additionally, the aforementioned flexible process of calculating and applying reinforcement was incorporated in FABRICFORMER as well. Unfortunately, no use could be made of this non-linear analysis since it became clear early on that the computational demand would be too high within the constraints of available time and computing power.

To finish the program the Differential Evolution algorithm was used to encompass an iterative process between the fabric formed, dynamically relaxed beam geometries and the subsequent structural analysis in ANSYS. The optimization algorithm was used to vary a set of twelve or fifty parameters – depending on the mold type – that determine the geometry in Dynamic Relaxation. These parameters describe values of the applied prestressing and coordinates related to the supports of the fabric. Some of these coordinates are indirectly related to the prestressing or the mold geometry as they define control points that in turn describe Beziér curves. These smooth, continuous curves are then used to define prestressing or geometric values along the length of the beam.

The development of FABRICFORMER, which spanned several months, led to numerous results which were continuously used for debugging purposes, to refine and/or to upgrade the program. The first succesful results offered optimized geometries for the so-called keel mold beam type, which were proven to be structurally more efficient than rectangular beams of equal volume. The linear analysis underlying the optimization did however lead to narrow geometries, an effect which was later avoided by applying a much higher load to influence the Performance Index. FABRICFORMER was then upgraded to optimize beams of the pinch mold type, which is much more complex to model. The modelling was succesful, though optimization met with less success. The results were optimized, and superior to both rectangular prismatic and curved beams, though – based on observations of BESO results – were clearly still not as optimal as expected. Recommendations have been made to improve upon the optimization of pinch mold beams but time did not allow these improvements to be made personally.

General conclusions and recommendations were made for the improvement of the program and its results.
Overall, the combination of a general optimization algorithm with a complex manufacturing method was succesfully achieved within a single computational tool.

Table of contents

CHAPTER 1 | Introduction

The field of computational optimization and the practice of building engineering seem ideally suited to one another. The civil engineer, though he is no stranger to architecture, is taught that there is no construction more beautiful than one that efficiently carries its loads. A structure has to handle certain given loads using a relatively small amount of material. If this goal is achieved, then the engineer considers the resulting design as an elegant solution. Mission accomplished.

Why then does computational optimization find such limited use in the building industry? One common explanation, is the fact that the building industry is often regarded as a conservative one, especially when compared to other industries. Why fix something that isn't broken? Construction and engineering firms rely on methods that have stood the test of time.

But this is not completely true, as the steady introduction of computer-aided design (CAD) shows. Building plans are drafted using a vast array of computer tools. Many firms employ finite-element analysis to refine their preliminary designs. They do not view computer programs as cumbersome or unpractical. They have proven their potential and worth.

So then what is the reason? If anything, it is simply this: the gap between structural optimization and building engineering hasn't been bridged. Certainly, some examples on connecting the two must exist, but it seems, the body of work on computational, structural optimization has not yet reached some critical mass for the building practice to really take notice. This thesis will explore this theme and attempt to combine optimization and manufacturability for a specific case and within a specific computer model.

This chapter will discuss the case of a structural beam to which this thesis is applied, expand on typical optimization targets in building engineering, define a problem definition and objectives for the entire thesis project and outline the content of the remaining chapters in this report.

1.1 Validating optimization of beams

As an introduction to the problem definition of this thesis, a standard beam is discussed, which was chosen as a relatively simple 'problem' compared to, for instance, a column.

Currently beams are designed and calculated using Euler-Bernoulli beam theory. This theory, the origin of which dates back to the late 18th century, provides an easy and efficient tool for civil engineers during the design process. However, this theory does require several interesting, and with respect to optimization methods, arguably unnecessary assumptions:

- The beam is relatively long and slender.
- The beam cross section is constant along its length.
- The beam is loaded in its plane of symmetry, avoiding-torsional effects.
- Deformations are relatively small.
- The material is isotropic.
- Plane sections of the beam remain plane.

Despite these assumptions the theory has been proven as an effective design tool. Furthermore, other and additional calculations can account for torsion, plastic behavior and non-prismatic beams.
Still, the engineer might become entrenched in certain ideas due to these assumptions. The second assumption is especially restrictive, as material cannot be distributed along the length. Any structural engineer knows that material is used differently at various sections along the beam, depending on the local shear and bending stresses. He usually knows where the weakest links in a beam are, which attests to the fact that material is being used inefficiently elsewhere in the beam. Thus, it is imaginable that using computational optimization methods to consider this complex problem as a whole can offer more efficient solutions. Such an optimized element might result in a significantly decreased expenditure of resources when produced on a large scale, whilst also presenting a more cost-efficient alternative to current elements.

1.2 Problem definition

The discussion in the previous paragraph acknowledges that the potential for more optimal structural elements exists. The problem is defined as such:

Structural elements are currently designed using assumptions which potentially decrease efficiency.

Furthermore the following problem – barring developments in structural materials – is defined:

Element design is rarely innovated upon. Engineers often rely on proven concepts and innovate with respect to an entire structure rather than its parts, the single elements.

And finally, based on findings in Appendices A and B:

New production techniques and construction materials offer possibilities which have often not been explored to their full extent.

Computational optimization may assist in overcoming these problems and could simultaneously incorporate considerations and constraints other than mechanical ones, rather than doing so consecutively.

1.3 Optimization targets

The problem definition was derived by taking the point of view of a structural engineer. As those who work in the building industry know, the building process can often be anything from interesting and inspirational, to long and arduous. This can be attributed to the large investments that come with producing - often unique – buildings, which in turn has led to the many parties involved. Each party, or stakeholder, has varying interests, which can also be translated to specific interests when it comes to structural elements. For this reason the different interests are evaluated in order to define to what ends a structural element may be optimized; the optimization targets.

Typical parties involved in the building process include:
1) The developer/owner
2) The project manager
3) The architect
4) The structural engineer/designer
5) The installations engineer
6) The end-user
7) The government
8) General society

At the moment, it is reasonable to say that in general the structural engineer designs a structural element through iteration by taking other, external demands into account as boundary conditions, while meeting certain governing structural demands. Optimization, in that case, has become somewhat limited.

By evaluating the different needs and wishes of those involved, the 'external demands' mentioned, we can identify what properties exactly we could optimize to increase efficiency.

(1) The developer or owner of the project generally has financially-motivated demands on structural elements. The time and cost required to produce and install a building element have to be minimized. The amount of material has to be used efficiently, but more importantly a beam should be designed in such a way that it can be manufactured easily in high volumes at relatively low costs. Transportation and installation have to be efficient and effective, which translates to limitations in weight and dimensions.
In the case of high-rise projects, the height should be minimal as well. A high-rise building is often bound by a maximum total height and the developer will seek to maximize the number of floors within this height to create more lettable floor space.

(2) The project manager strives to stay within the agreed budget and building time. At no or limited cost to quality, he will seek cheaper or more easily installed structural elements. In countries where labor is cheap, material use is governing, while vice versa the manufacturing, transportation and installation become more important.

(3) The architect will generally attempt to convey a certain architectural concept. This is a highly subjective demand, but it can be translated to many geometric boundary conditions, such as a given curvature, shape, symmetry, etc. Also architects tend to prefer more slender solutions by limiting the total outside surface area (or outer circumference) of a structural element.

(4) For the structural engineer the slenderness is secondary to other concerns, as hollow core sections illustrate. He will judge a beam by the stiffness per unit volume of material and will seek to evenly distribute stresses along the entire cross-section and length. Usually the bending stiffness EI is governing, but in some cases the axial stiffness EA, the shear stiffness GA and/or the torsional stiffness EI_t are important as well.

(5) The installation engineer would prefer to place all installation ducts in open space but is often required to put them in the same plane as the structural members, creating a puzzle for him and the structural engineer to solve. When a beam is tailor-made, the installations require space with specific dimensions, which the structural engineer will try to incorporate in key locations where bending moments or shear forces are minimal. For mass-produced standard elements, even more and/or larger openings are required which are suitable for a general range and number of installation ducts.

(6) and (7) Both the end-user and the government insist on reliable, sustainable and safe structures. The government offers building codes, such as the Eurocode in Europe, to guarantee a certain degree of reliability and safety. These codes present boundaries for engineers to adhere to during design.

(8) Society as a whole has a limited financial interest in individual projects. Over the last few decades an insistence on sustainable development has emerged. The reasons for this interest in sustainability are growing concerns for issues such as the threat of global warming, pollution and resource depletion. Though the building industry represents a relatively small contribution in the global pollution and greenhouse gas emissions, it is nonetheless a contributor. Buildings which are poorly designed and insulated expend a lot of energy on a daily basis, while a large amount of energy is consumed in the production of building materials. The mass-produced structural element should be designed in such a way that waste of material is limited and the material used is done so efficiently.

In summary a structural element could, or should be optimized for,

- material efficiencies, such as
  - o minimal material use, or the total volume,
  - o minimal stresses, or maximum ratio stiffness/ volume,
  - o slenderness, or the ratio surface area/volume,
  - o minimal construction height, or height/cross- sec tional area and/or
- manufacturability,

while taking into account boundaries determined by,

- manufacturing, transportation and installation, such as weight and outer dimensions,
- legislation, or building codes and/or
- internal spaces for installation ducts.

Manufacturability is a particularly difficult goal to optimize for as it comprises of many geometric conditions – thickness, symmetry, maximum dimensions, etc. – and relates to manufacturing time and cost.

In conclusion, element design is determined by many different demands, set by different actors. This complex of demands is usually handled by defining boundary conditions (e.g. program of demands) and generating only a few alternative solutions based on one or a few optimization objectives. The chosen result is arguably a less than efficient outcome, and therefore not a true optimum.

1.4 Definition of the objectives

From the problem definition in the previous section a preliminary objective can be extrapolated for this graduate project and thesis:

To optimize a structural member with respect to material efficiency and manufacturability

This thesis is a follow-up on more theoretical work that has been done, and it introduces manufacturability as a key element in the results.
The underlying question of the thesis is simple:

Can existing structural element design be improved (using optimization methods) in a practical manner?

This question and the preliminary objective imply that the resulting element should be viewed as a product as it has been designed/optimized with manufacturability in mind. The word product implicitly involves economic aspects, as a product should not merely be manufactured, but also marketed and sold. This influenced the decision for the manufacturing method for this thesis and to explain this the following notion by Vambersky et al. (2001) is quoted:

'Modulation is an important economic factor in designing and constructing buildings [..] in prefabrication, this is even more pronounced in terms of standardization and production. [..] The use of modular planning is not supposed to be a limitation on the freedom of the designer' and 'should be used throughout the precast building in every design, as long as it does not conflict with other constructional or architectural requirements.'

In other words, a building product should be modular i.e. suitable for mass production, in turn excluding some of the manufacturing methods that were found. Mass production however, contrasts the potential of optimization, illustrated by McDonough & Braungart (2006):

"To achieve their universal design solutions, manufacturers design for a worst-case scenario, they design a product for the worst possible circumstance, so it will always operate with the same efficacy. This guarantees the largest possible market for a product. It also reveals human industry's peculiar relationship to the natural world since designing for the worst case at all times reflects the assumption that nature is the enemy."

These separate philosophies are difficult to unify, something this thesis will not achieve on its own, so for this thesis it is acknowledged that there are in fact two secondary objectives that deal differently with this dillema.
On one hand this thesis will develop a piece of software, a tool to optimize for any specific situation so that the output is a tailor-made design.
On the other hand this thesis will result in a beam design, potentially modular, that through optimization is superior to current designs, though still adhering to the same, established worst-case scenario.

The primary objective is specified in the following two secondary objectives:

(1) To develop a computational tool and/or system to optimize fabric formed concrete structural elements with respect to material efficiency and manufacturability.

(2) To computationally optimize the shape of a fabric formed concrete structural element with respect to material efficiency and manufacturability.

So that (1) refers to the development of a process and software tool and (2) to the actual results.
Based on literature studies, the decision was made to focus on fabric formwork due to its relative geometric freedom. Other new promising methods had certain geometric limitations and/or lacked sustainable potential.

The primary objective is now:

To bridge the gap between computational optimization and manufacturability, specifically fabric formwork

1.5 Thesis outline

This thesis will explore the combination of optimization algorithms with a specific manufacturing technique, fabric formwork.
Appendices A and B are part of a preliminary study on literature and contain all information of that study that is no longer directly relevant to the thesis. They explore optimization and manufacturing methods that currently exist, from which the methods used in this thesis were chosen.
The most prevalent method for optimization, Evolutionary Structural Optimization (ESO), is introduced in Chapter 2. It details the history and development of ESO, gives some existing examples, explains a relatively new version of the algorithm and ends with some results that were found in the course of this thesis (Chapter 8 also contains ESO related results).
The fabric formwork technology is explained in Chapter 3, which includes a brief history, photographs of various formworks and summaries of all research found relevant to structural engineering.
Instead of ESO, a general optimization algorithm called Differential Evolution (DE) was ultimately used for this thesis. This algorithm is explained in Chapter 4.
To model the fabric of the formwork, a form finding algorithm called Dynamic Relaxation (DR) was implemented. Chapter 5 explains how the algorithm works for tension structures (e.g. tents), how it was adapted to fabric formwork and what the results look like.
To model the concrete and perform structural analysis, the existing software `ANSYS` was used. Chapter 6 explains how concrete is modelled in this particular program and how a flexible non-linear analysis was designed for the optimization process.
The entire software tool that was made during this thesis, dubbed `FABRICFORMER`, combines the content of Chapters 4 to 6. The structure and design of the tool is set out in Chapter 7.
The results from the tool as well as some advanced ESO results are given and discussed in Chapter 8.
Finally, conclusions on the whole thesis and its results are given in Chapter 9. Additionally, some recommendations and suggestions are made for future development and improvement of `FABRICFORMER` or similar tools.

CHAPTER 2 | Evolutionary Structural Optimization (ESO)

Arguably the most popular method for topology optimization in civil engineering is Evolutionary Structural Optimization (ESO) (Xie & Steven, 1993). It was originally proposed in the early 1990's and has given rise to a small, but international community expanding its possibilities and potential. It is probably the first optimization method in civil engineering to be commercialized and applied to real projects. This chapter will introduce ESO and its historical development, explain some of its underlying principles and show how it was used in the context of this thesis.

2.1 The history and basics of ESO

Evolutionary Structural Optimization, or ESO, was developed specifically for engineering purposes using finite element analysis as a framework. It differs as such from other optimization algorithms which were often based on a general mathematical approach and now have a wide variety of applications. ESO on the other hand is purely limited to shape optimization and has very straightforward principles guiding it. It is for this reason that the term 'evolutionary' is an inappropriate, perhaps incorrect adjective to denote this algorithm. According to Wikipedia, evolution is never goal-oriented and can be described as the irreversible accumulation of historically acquired information. Development, in contrast, is made up of predictable directional changes, which is more appropriate when describing ESO.

In basic terms, ESO develops a topology, or shape by removing inefficiently used material based on finite element analysis. It continues to do so until some optimum is found between the stiffness and volume of the shape. The results show very organic, sometimes skeletal structures to be optimal, which was probably the reason for attributing the same qualities as natural evolution to this algorithm. The organic shapes are of course much more complex than conventional rectangular shapes. This has led to limited applicability of ESO outside academic circles. In the following paragraphs some examples that do exist are shown.

Since the early 90's ESO has been improved upon and adapted because of its advantages. The algorithm is fast and can easily be understood due to its goal-oriented nature. The development of ESO has led to several incarnations, which are all briefly discussed:

ESO

Evolutionary Structural Optimization (ESO) works by starting from a dense finite element ground mesh. Material is removed based on stress criteria. In other words, material that is hardly stressed is simply removed, increasing stresses elsewhere until maximum stress is achieved. The result is a 'fully-stressed design'.
One of the more famous results of ESO is an animation of the algorithm, which, using certain constraints and optimality criteria, leads to the shape of an apple (Figure 2.1).

AESO

Additive ESO (AESO) works by starting with a very small, minimal amount of material and adding it near areas of high stresses until maximum stress criteria are met (Van Gemert, 1996).

BESO

The combination of both ESO and AESO, called Bi-directional ESO (BESO) was proposed by Querin et al. (1997) and is under development at RMIT, Australia. The algorithm is able to both add and remove material, decreasing

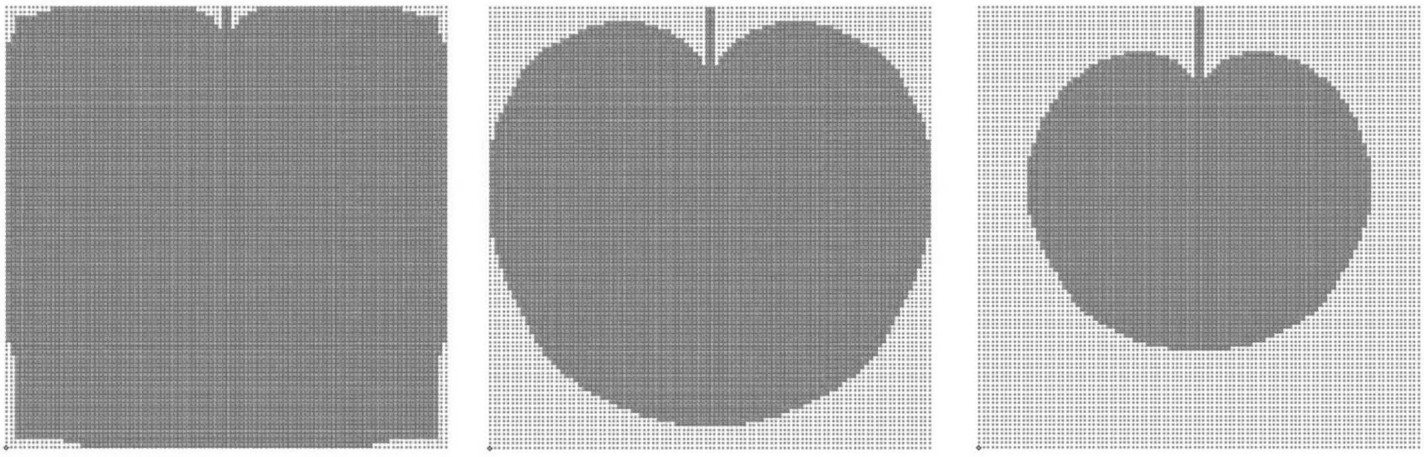


Figure 2.1
ESO made famous by optimizing towards an apple shape.
ISG website (2007)

the number of calculations and reducing the risk of hitting a local optimum. The rate at which material is added or removed is determined by inclusion and evolutionary ratios *(IR/ER)* and a rejection ratio *(RR)*. Recently these ratios have been replaced by a single dynamic parameter, the Removal Rate of Volume *(RRV)*, that can produce different optima with equal volume which are compared based on for example stiffness. (Huang et al., 2006)

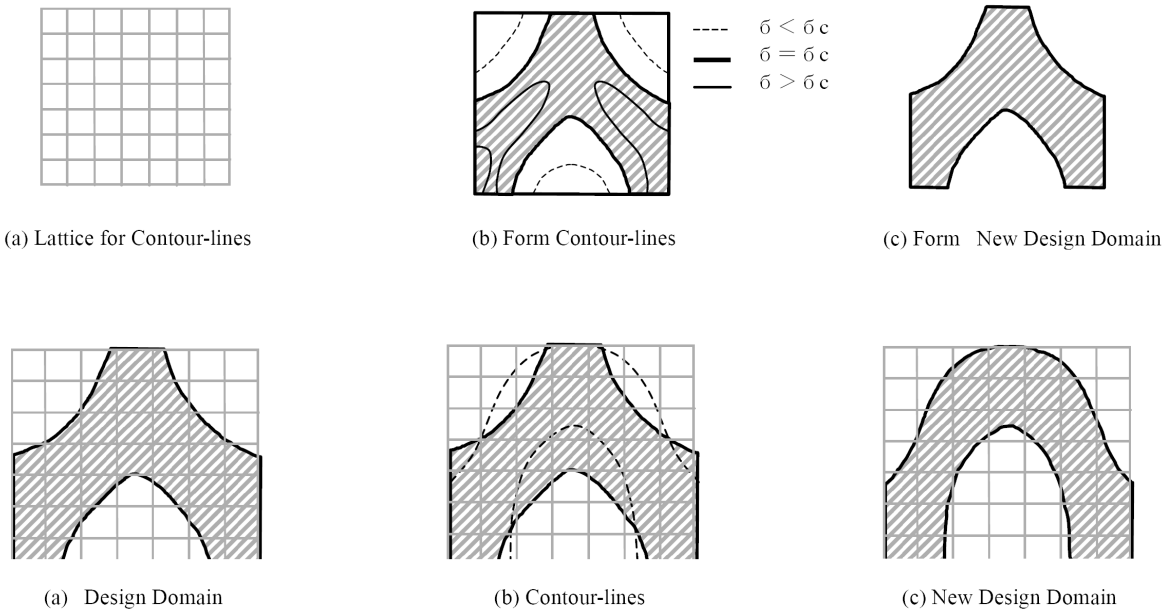
XESO

The latest development is Extended ESO (XESO). It is independently developed at Nagoya University, Japan, since no formal collaboration with RMIT exists. XESO uses stress contour lines for two dimensional problems and stress contour surfaces for three dimensional problems. These contours describe lines or surfaces of equal stress and determine boundaries for whole areas in which material is rejected or added (Figures 2.2 and 2.3). The description of the exact principles behind this algorithm is limitedly available and contrary to the other versions of ESO, development – in the FORTRAN programming language – takes place behind closed doors and no published or open source code exists. The development of this algorithm was triggered by the lack of bi-directionality in ESO as well as another problem, which was described as follows:

"..the rejection procedure [of ESO] is performed throughout the whole evolutionary process of computation based upon that definite initial value [for the rejection ratio] and no attention is paid on the situation of the structure during evolution."

This problem, which is also present in BESO has also been solved by this author, but differently and within BESO, as described at the end of paragraph 2.4.

Figure 2.2
Above the process of forming a new design domain based on contour lines. Below the addition and deletion of material by using contour lines.
Ohmori et al. (2004)



The (in-)efficiency of the material throughout the shape is determined at each iteration by calculating the stresses or complementary energy in a finite element analysis. This offers information for each element and is the basis for its evaluation and subsequent addition/removal.
Early versions of ESO use stresses to evaluate the shape. For isotropic materials such as steel the stress can be calculated in three dimensions, by using the following formula by Von Mises:

$$\sigma^{Mises} = \frac{1}{\sqrt{2}}\sqrt{\left(\sigma_x - \sigma_y\right)^2 + \left(\sigma_y - \sigma_z\right)^2 + \left(\sigma_z - \sigma_x\right)^2 + 6\left(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2\right)}$$

where the stresses σ and τ represent the components of the normal stresses and shear stresses respectively.

Once the maximum design domain, optimality criteria and all boundary constraints have been defined, the program starts from an initial physical domain. It is run through a finite element analysis after which the results are evaluated and certain elements are removed and others are added. The optimization process continues until a convergence condition is met, or the program reaches an oscillary state in which the same material is continuously added and removed again. Several features speed up the convergence of the process; sensitivity analysis of all elements determine where material is most effectively added or removed. The *IR* and *RR* or later *RRV* parameters determine how much material should be added or removed at each stage. (Young et al., 1999)

In order to present the final optimal topology with a manufacturable boundary, an intuitive smoothing technique is applied. The coordinates of every node on the boundary are averaged by the coordinates of neighboring nodes.

There are different optimality criteria possible, which are also valid for other structural optimization methods:
Huang et al. (2006) propose minimizing complementary energy as well as using sensitivity analysis as a criterion, leading to a least weight structure.
Young et al. (1999) use Von Mises stresses and maximum stress criteria as a criterion, leading to a fully-stressed design.
Though these criteria differ, for all practical purposes one can expect a fairly similar outcome.

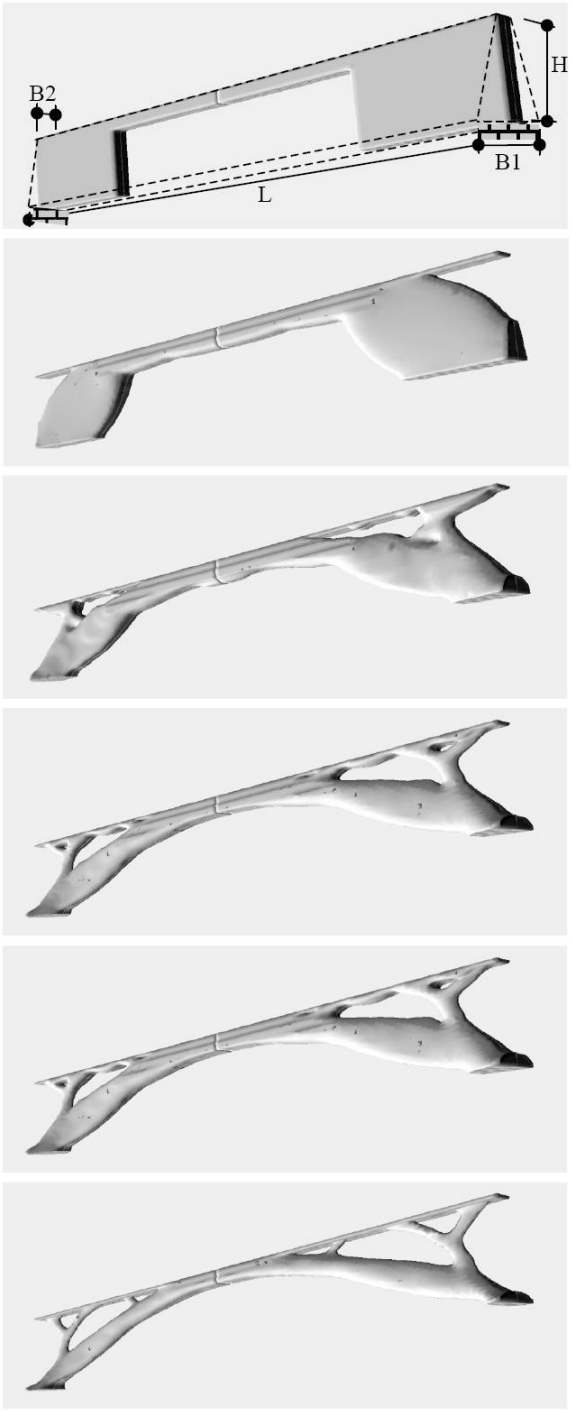


Figure 2.3
XESO in action, from initial design space to final solution for a bridge structure.
Ohmori et al. (2004)

2.2 Examples of manufacturing considerations in ESO

As mentioned, ESO is a direct search algorithm and not a true evolutionary algorithm that arbitrarily searches solution within a search space. The direct nature of ESO is probably the reason why considerations and constraints other than mechanical ones (stresses, strains, etc.) have rarely been included in the optimization process. Commonly such aspects are taken into account afterwards i.e. post-processing, such as smoothing of the mesh. In fact, the only example found was a previous Master's thesis at the Delft University of Technology (Pearse-Danker, 2006).

This particular thesis focused on creating a least-weight topology for large steel nodes. Such nodes are cast to avoid welds in critical high stress areas and reduce the chance of fatigue failure. They can be found in large steel structures, such as offshore platforms.

Using evolutionary structural optimization (ESO) and software program GSA a program was written which also incorporated some manufacturing considerations of steel casting. A void filter was successfully implemented which compared the materials savings by introducing a void to the cost of manufacturing the void itself. Another filter ensured a minimum thickness throughout the node, but met with less practical results. The problem was attributed to the finite element mesh, and it was recommended to model the problem using contour lines instead of a mesh. A mesh would still be generated, but only as a function of the contour lines and for intermediate finite element analysis of each new iteration. Interestingly, this approach was implemented in XESO which was developed around the same time. Pearse-Danker also seems to propose using feature libraries as briefly explained in Appendix A.4.4.

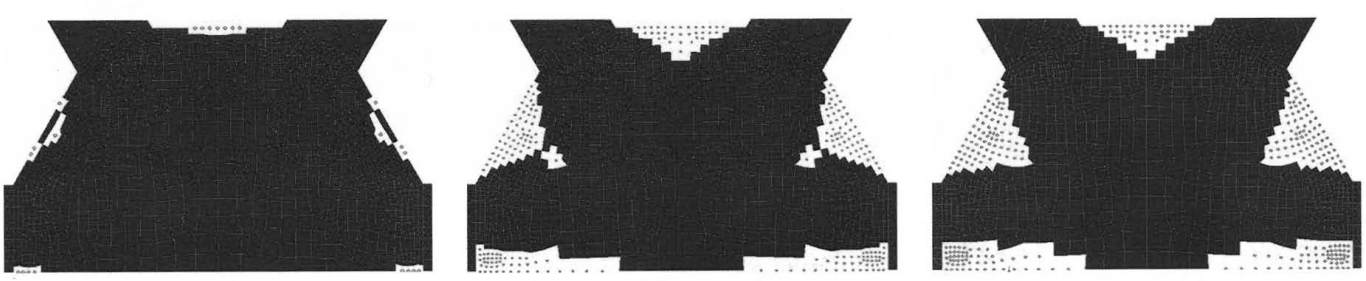


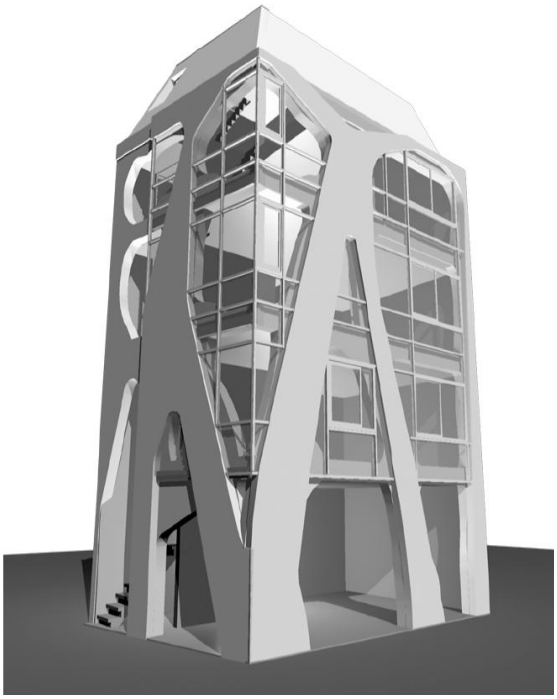
Figure 2.4
Cast joints optimized in ESO are post-processed using a void filter.
Pearce-Danker (2006)

2.3 Examples of shape optimization in building projects.

Even more exceptional than manufacturing considerations in structural optimization, are actual building projects which involve optimization being part of the original design process. In these instances, ESO did not include manufacturing in an explicit manner, but was used as a tool during initial stages of design.

Akutagawa River Side in Takatsuki, Japan

The Akutagawa River Side project is a large scale redevelopment in Takatsuki, Japan. It is aimed to rejuvenate the shopping arcade, which runs from the north front of the nearby Takatsuki Japanese Railway station, and of course this particular urban area as a whole. One of the featured buildings is a first example of computational morphogenesis, or ESO, put into practice. This office building, four stories high, was completed in late April of 2004. Two of its side walls, those facing west and south, were optimized using ESO and built in reinforced concrete. Typical dead and live loads, as well as dynamic earthquake loads were taken into account. The results of the evolutionary design process were verified afterwards in an elasto-plastic numerical analysis based on deflections and cracking patterns.



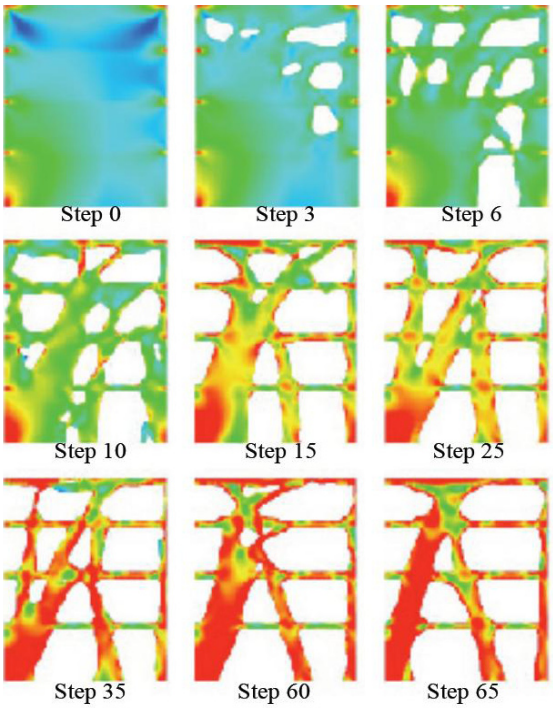
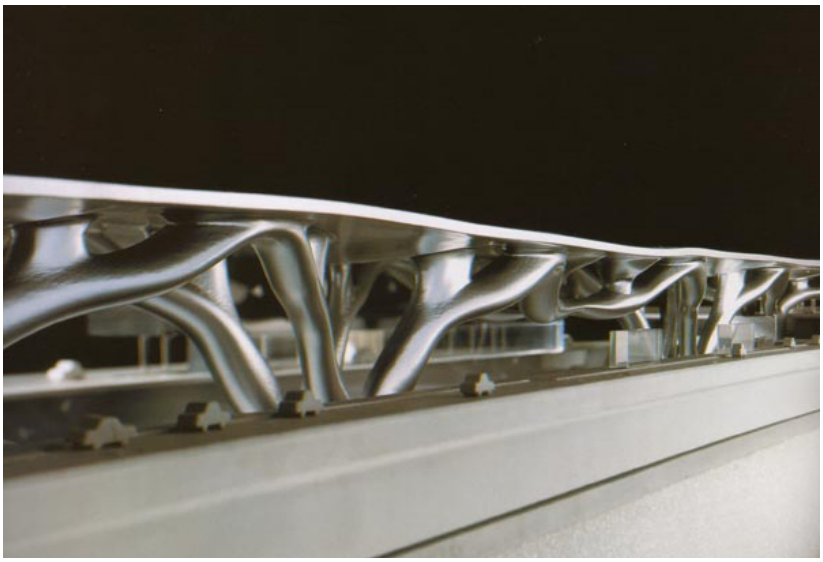


Figure 2.5
3D model of the Akutagawa building **(Top left)**
Ohmori et al. (2004)

Figure 2.6
The Akutagawa River Side office building. Two of the building faces have been optimized with ESO. The evolutionary process of the south wall is shown, where it is visible that the shape and position of the floors were constrained.
(Top right)
Ohmori et al. (2004)

Figure 2.7
Photograph of the Akutagawa building after construction **(Bottom)**
Ohmori et al. (2004)

New station in Florence, Italy



Arata Isozaki & Associates is an architectural firm based in Tokyo and Barcelona and has applied ESO to several of its designs. One of the first attempt was a design entry in a competition for a new station in Florence, Italy (Cui et al., 2003). The new station project is 400m in length, 42m wide and 20m high. A roof encompassing this entire complex provides a huge lower space for facilities while the roof itself is a landing strip for light aircraft.
The design entry by Arata Isozaki features a flat roof, supported at several points by organically shaped columns, designed through the use of ESO.
The entry won second prize and lost out to a different design by Norman Foster.

Figure 2.8
3D models of the Florence station design entry, designed by Arata Isozaki using ESO

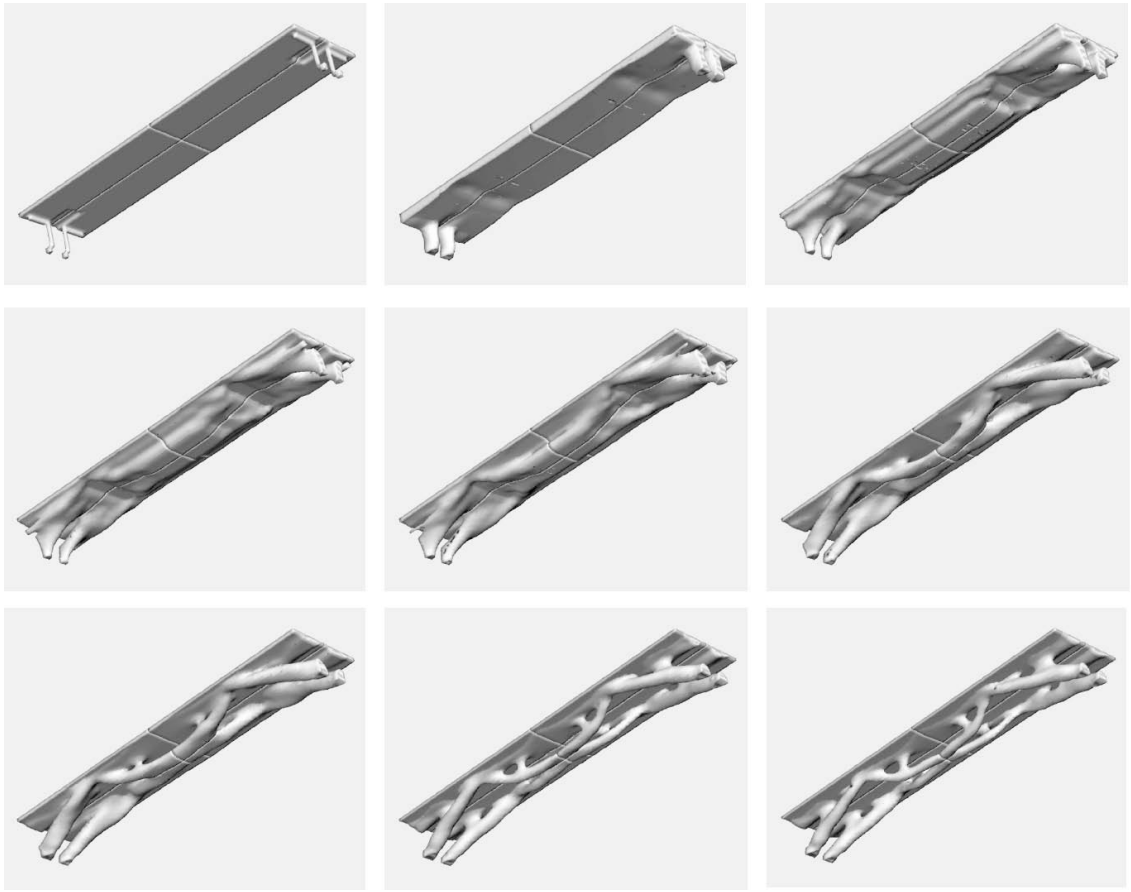


Figure 2.9
The evolutionary design process of the design entry by Arata Isozaki & Associates for the station project in Florence, Italy.
Cui et al. (2003)

Illa de Blanes in Blanes, Spain

Very similar to the Florence station and also by Isozaki, is the Illa de Blanes, a large seaside structure with mixed commercial, public and recreational functions. Besides organically shaped columns, the design also features a roof that was created using ESO.
The project is currently on hold following government and policy changes in Catalonia.

Note: Arata Isozaki also designed one face of the Qatar Convention Centre using ESO as well as the small Gifu Kitakata Apartments in Japan. Unfortunately, limited information is available on these projects.

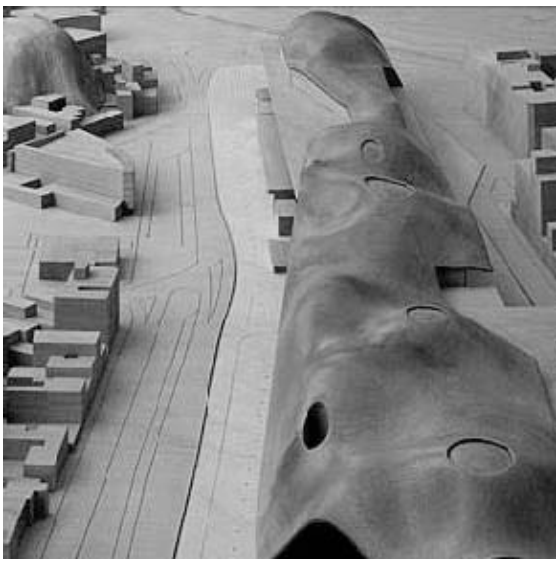


Figure 2.10
Physical model of the Illa de Blanes, designed using ESO

2.4 Concepts and mathematics in BESO

Early on it was decided to implement BESO, due to its superiority to ESO and the limited sources available on XESO. The program was written in ANSYSscript within the finite element program ANSYS. This paragraph will give a mathematical, step-by-step overview of BESO.
The most comprehensive source on the BESO algorithm found was by Huang et al. (2006). However, several aspects were not clearly explained which led to delays during coding of the algorithm. The overview given here will use this paper as a basis, but expand at points where information was found lacking. Furthermore, some improvements are made by this author which parallel apparent characteristics of XESO. These are explained at the end of this paragraph.

In structural engineering the stiffness of a structure and its elements is an important design aspect. The stiffness determines the various deflections that occur within the building and it is these deflections that are often constrained via building codes to suit safety and esthetic requirements. Stiffness determines the optimization process in BESO, so this overview will begin by defining stiffness mathematically using the reader on the theory of elasticity by Blaauwendraad (2004).

In structural engineering stiffness is often expressed in the stress-strain or force-displacement diagrams.

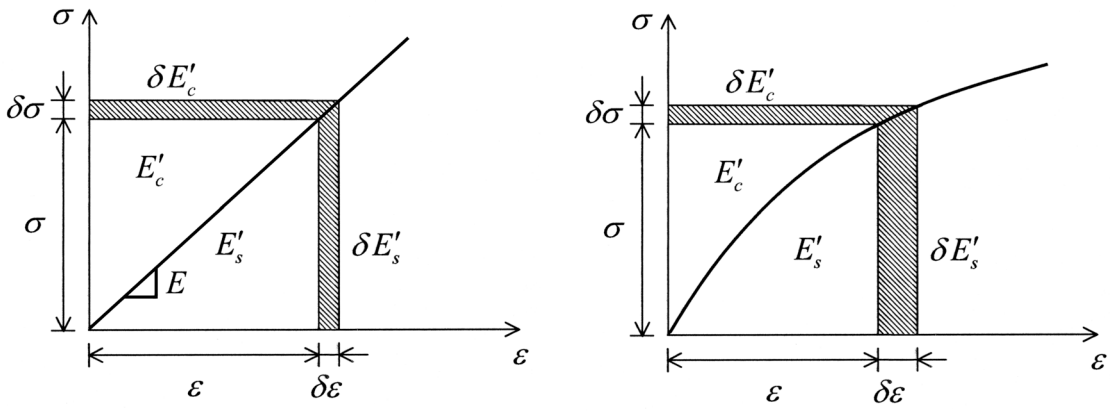


Figure 2.11
Typical stress strain curves, linear and non-linear respectively. The enclosed areas define the various energies.
Blaauwendraad (2004)

The slope of the stress-strain curve is often assumed to be linear. The resulting gradient is better known as the Young's modulus E, or the modulus of elasticity E. A higher value for E, resulting in a steeper stress-strain curve equals greater stiffness. In linear elastic material, the Young's modulus can be expressed as:

$$E = \frac{\sigma}{\varepsilon} \tag{2.1}$$

Suppose that at a certain moment a load σ is present, then a small increment $d\sigma$ causes the strain to increase by $d\varepsilon$. The existing load σ performs an amount of work equal to $\sigma d\varepsilon$. Therefore the total amount of work equals:

$$E_s^{'} = \int_0^{\varepsilon} \sigma d\varepsilon \qquad \text{with} \quad \sigma = \sigma(\varepsilon) \tag{2.2}$$

This is referred to as deformation energy and is interpreted as the amount of work performed by the load σ, or the amount of potential energy which is accumulated in the material. Deformation energy is expressed in strains, and is therefore also called strain energy.

The remaining area, the area above the stress-strain curve, is called complementary energy or mean compliance. It is expressed in stresses. In general, it is not easy to give a physical interpretation of the concept of complementary energy. The complementary energy is defined as:

$$E_c^{'} = \int_0^{\sigma} \varepsilon d\sigma \quad \text{with} \quad \varepsilon = \varepsilon(\sigma) \tag{2.3}$$

It is noted that the total area can be expressed as:

$$\sigma\varepsilon = E_c^{'} + E_s^{'} \tag{2.4}$$

In linear-elastic material the deformation energy is equal to the complementary energy.

$$E_c^{'} = E_s^{'} \tag{2.5}$$

If the E-modulus increases, the material and thus the structure become increasingly stiff. At the same time the complementary energy decreases and the strain energy increases. It is obvious that maximizing the overall stiffness is the same as minimizing the complementary energy, or maximizing strain energy i.e. Statement (i).

When the overall strains and displacements of the structure decrease, the structure also has higher stiffness. In this case both the deformation and complementary energy decrease. It is noted that Statement (i) again holds true for complementary energy but the opposite is now valid for strain energy. This observation is important due to the following, potentially confusing statement made by Xie (1997) and repeated throughout subsequent papers.

"It is obvious that maximizing the overall stiffness is the same as minimizing the strain energy."

In the case of ESO, where material properties are not a variable, this statement however is correct and refers to the so-called pinciple of minimum potential energy.
Furthermore, in literature on ESO mean compliance i.e. complementary energy and strain energy are confused with one another. While they are not the same; for linear-elastic materials, they are equal in value and the mix-up has no significant consequences.

To explain how BESO works we continue under the assumption of minimizing strain energy, which is also a practical approach since ANSYS is able to output the strain energy of each finite element under the command name SENE, defined as (2.6).

$$E_e^{po} = \frac{1}{2}\sum_{i=1}^{n} \{\sigma\}^T \{\varepsilon^{el}\} vol_i + E_e^{pl} + E_s \tag{2.6}$$

where n = number of integration points
σ = stress vector
ε^{el} = elastic strain vector
vol_i = volume of integration point i
E^{pl} = plastic strain energy
E_s = stress stiffening energy

Going back to theory, the formula for strain energy is given by:

$$\min. \quad E_s^{'} = \int_0^{\varepsilon} \sigma d\varepsilon \tag{2.7}$$

or differently expressed,

$$\text{min.} \quad E_s^{'} = \int_V \vec{\sigma}^T \vec{\varepsilon} dV = \int_V \vec{\varepsilon}^T \vec{\sigma} dV \tag{2.8}$$

The material properties can be expressed using the following stiffness formulation;

$$\vec{\sigma} = S \cdot \vec{\varepsilon} \tag{2.9}$$

where S is the material stiffness matrix (also known as K, but to avoid confusion later on with another matrix, S is used instead). Using (2.9) in (2.8) gives:

$$\text{min.} \quad E_s^{'} = \int_V \vec{\varepsilon}^T S \vec{\varepsilon} dV \tag{2.10}$$

The strains can be expressed in displacements using the kinematic matrix B.

$$\vec{\varepsilon} = B \cdot \vec{u} \tag{2.11}$$

Combining (2.10) and (2.11) results in

$$\text{min.} \quad E_s^{'} = \int_V \left[B\vec{u}\right]^T SB\vec{u} dV = \int_V \vec{u}^T B^T SB\vec{u} dV \tag{2.12}$$

The product of the matrices B^TSB is well known to be square and symmetrical is commonly indicated as the stiffness matrix K:

$$\text{min.} \quad E_s^{'} = \int_V \vec{u}^T K\vec{u} dV \tag{2.13}$$

In a linear-elastic material, this becomes:

$$\text{min.} \quad E_s^{'} = \frac{1}{2} \vec{u}^T K\vec{u} \tag{2.14}$$

As explained, BESO works by adding and removing individual elements. It does this by evaluating how the strain energy changes when the ith element is removed or added. This can be expressed as:

$$\Delta E_{s,i}^{'} = \frac{1}{2} \vec{u}_i^T K\vec{u}_i \tag{2.15}$$

BESO allocates a sensitivity number α_i to each existing element, based on this change in strain energy, by dividing it by the weight of the element itself (Figure 2.12):

$$\alpha_i = \frac{\Delta E_{s,i}^{'}}{W_i} = \frac{\frac{1}{2} \vec{u}_i^T C^T \vec{u}_i}{W_i} \tag{2.16}$$

or more generally, and in ANSYS language

$$\alpha_i = \frac{\Delta E'_{s,i}}{W_i} = \frac{\int_{V_i} \vec{u}_i^{\,T} K \vec{u}_i dV}{\rho \cdot V_i} = \frac{SENE}{DENS \cdot VOLU} \tag{2.17}$$

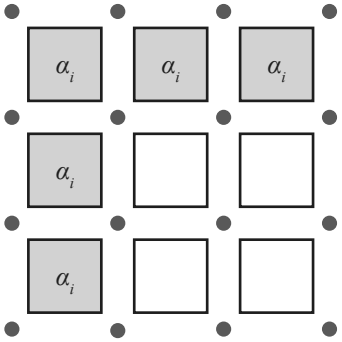


Figure 2.12
Example of 9 square elements and their 16 nodes. Grey elements are 'turned on' or live, white elements are off and do not contribute to the structure.

In BESO some elements are outside the current structure but within the domain i.e. 'DEAD'. It is necessary to determine how these elements might positively contribute to a new structure if they were 'turned on'. The algorithm needs to assign a sensitivity value to elements surrounding the structure. It accomplishes this in a couple of calculation steps, starting by calculating the sensitivity value α_k for each node by averaging those of the surrounding elements.

$$\alpha_k = \frac{\sum_{i=1}^{m} V_i \alpha_i}{\sum_{i=1}^{m} V_i} = \frac{\sum_{i=1}^{m} SENE \cdot VOLU}{\sum_{i=1}^{m} VOLU} \tag{2.18}$$

where m can be no more than 8 for cubic elements (four in the two-dimensional example of Figure 2.13).

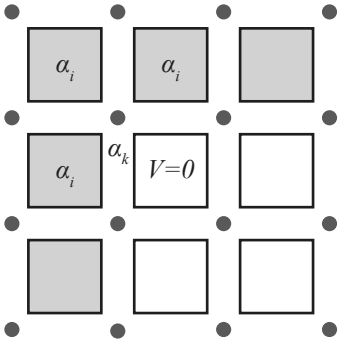


Figure 2.13
Each nodal sensitivity number is the average of the surrounding elements sensitivity values.

To obtain the sensitivity for the candidate elements, it is necessary to extrapolate the sensitivity numbers for the nodes surrounding the structure.

As shown in Figure 2.14, it is assumed that the sensitivity numbers vary linearly along the coordinates, so that the unknown node sensitivity number is calculated by:

$$\alpha_k = 2\alpha_{k1} - \alpha_{k2} \tag{2.19}$$

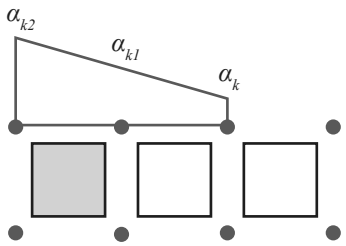


Figure 2.14
Sensitivity of outlying nodes is linearly interpolated from nodes within the current structure

This calculation is done for each of the six orthogonal directions.

$$\alpha_k = \frac{\sum_{n=1}^{N} 2\alpha_{k1} - \alpha_{k2}}{N} \text{ with } \qquad N \leq 6 \tag{2.20}$$

Finally, the smoothed sensitivity number of the candidate element e is calculated by averaging all nodal ones of this element as

$$\alpha_e^* = \frac{\sum_{n=1}^{N} \alpha_n}{N} \tag{2.21}$$

where N is the total number of nodes of the candidate element.

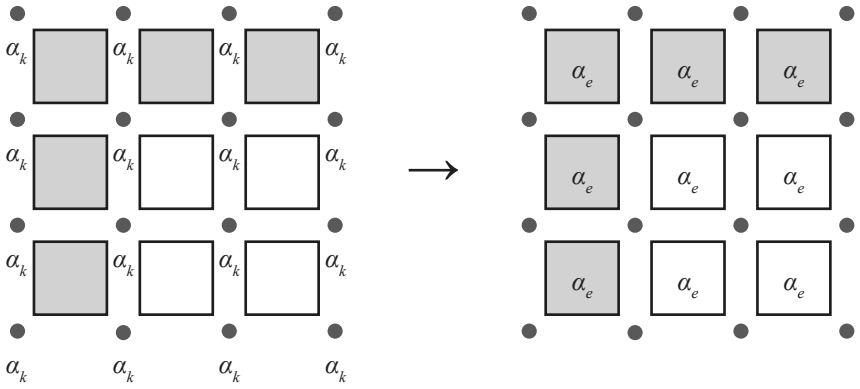
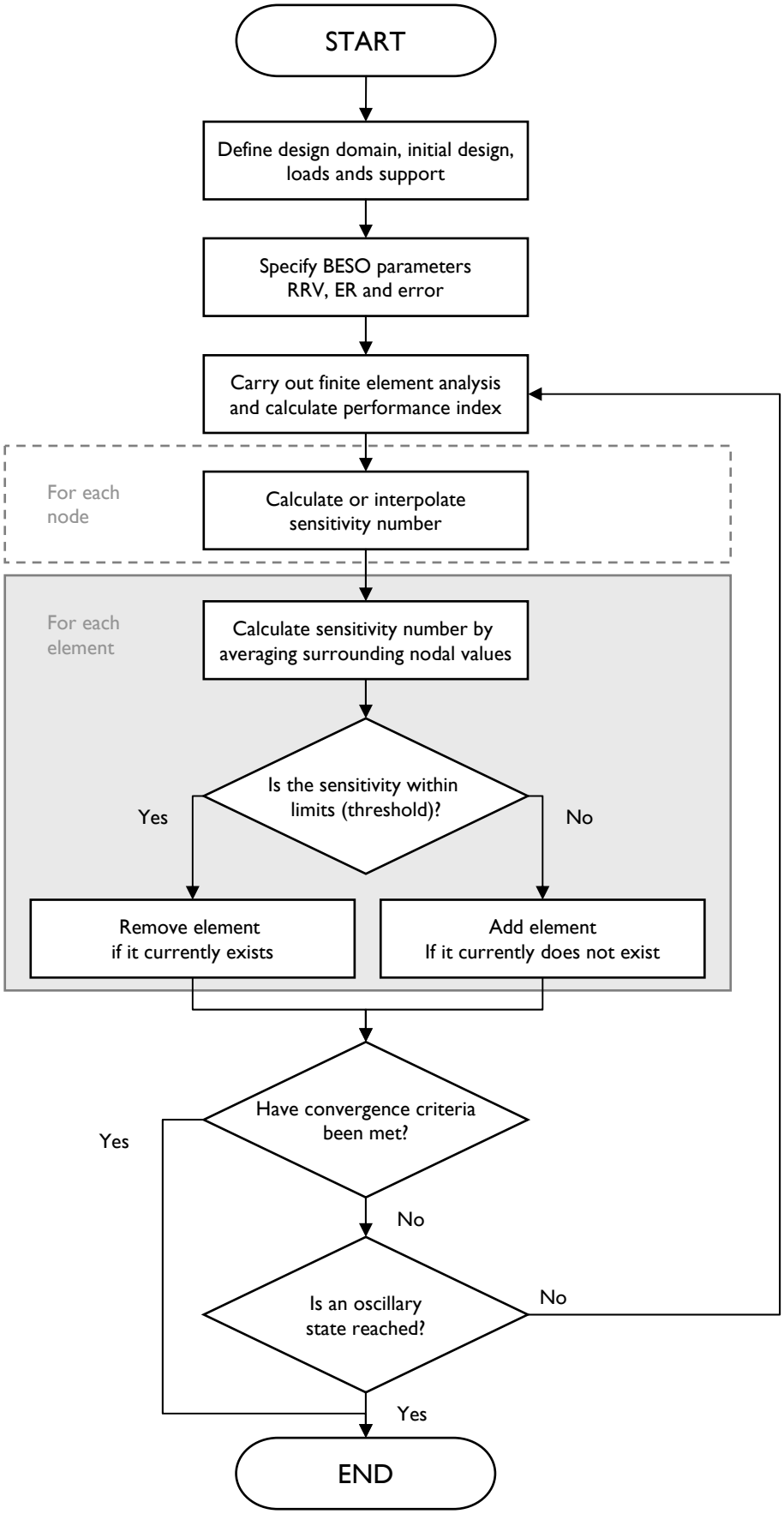


Figure 2.15
All nodal sensitivities are known, and give average element sensitivity numbers prior to volume addition/removal.

Now, all elements, both existing and surrounding, can be evaluated based on the calculated sensitivity numbers (Figure 2.15). Elements in the current structure can therefore be removed if they satisfy

$$\alpha_e^* \leq \alpha_{th} \tag{2.22}$$

The threshold value α_{th} is determined by the Removal Rate of Volume, or RRV_i, traditionally a fixed percentage of the total domain volume. In other words it forces the algorithm to find the solution where the volume is $x\ (= RRV_i \times 100\%)$ percent of the chosen domain. Since this initial domain is in a way quite arbitrary, the idea is to start with a low value for


START
Define design domain, initial design, loads ands support
Specify BESO parameters RRV, ER and error
Carry out finite element analysis and calculate performance index
For each node
Calculate or interpolate sensitivity number
For each element
Calculate sensitivity number by averaging surrounding nodal values
Is the sensitivity within limits (threshold)?
Yes
No
Remove element if it currently exists
Add element If it currently does not exist
Have convergence criteria been met?
Yes
No
Is an oscillary state reached?
No
Yes
END

RRV_i and after convergence add a certain increment called the evolutionary rate ER. This is continued until some chosen upper limit of RRV is met. For the purposes of this thesis the decision was made to change the definition of the RRV. The value of the RRV should not be dictated by the initial domain, but rather by the active volume at that point in time. This should avoid the necessity of a wide range of RRV values and produce more optimal structures since the algorithm is free to change the overall volume. This new relative RRV will be referred to as RRV_{rel}.
Convergence for the RRV_{rel}, or each RRV is met when increments in the performance index PI are sufficiently small. The performance index itself is the ratio of complementary energy to volume. It is calculated for the whole structure as the sum of the performance index of each 'LIVE' element. Also, because ANSYS calculates strain energy, the equation for the performance index becomes

$$PI = \frac{1}{E_s' \cdot W} = \frac{1}{SENE \cdot DENS \cdot VOLU} \tag{2.23}$$

The density is constant when using only one material, so it does not affect the performance. Furthermore, any symmetry planes need to be taken into account if any are used, by multiplying the volume and strain energy.
The whole process is summarized in the flowchart on the previous page. Oscillation refers to an alternate stopping criterion for cases in which the same set of elements are alternately added and removed, avoiding convergence.

2.5 Implementing the BESO algorithm

At first the ESO algorithm was coded in ANSYS as a starting point for BESO, because the ESO algorithm is easier to implement. A working ESO algorithm took one week to program, a working BESO algorithm a little over a month, not because of its higher complexity but mainly due to the relatively unstructured way in which ANSYSscript commands are designated and the limited information that existed on BESO, which led to many trials and errors.

The ESO algorithm is simple to explain. The least stressed elements are continuously removed. The amount removed equals the rejection ratio at that time, RR+ER.
Similar to other papers on ESO, an attempt was first made to approximate a theoretically optimal Michell structure (Figure 2.16) for the case of a block subjected to three point bending.

The following parameters and values (Table 2.1) were used, which were typically used in early papers on ESO.

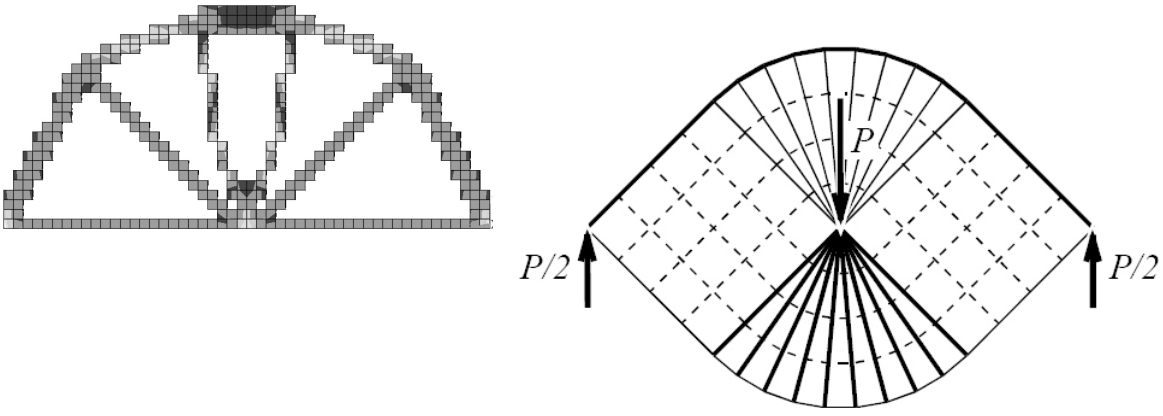


Figure 2.16
The best result from ESO compared to a theoretical Michell truss for the same load case..

TABLE 2.1 Properties for ESO version 1.3 in ANSYS		
Version 1.3		
modulus of elasticity	E	100.000 N/mm²
pointload	F	1000 N
dimensions	$w \times h$	10.000 x 5000 mm
element size		200
ESO parameters		
initial rejection ratio	RR_{init}	1 %
evolution ratio	ER	0.5 %
final rejection ratio	RR_{end}	25 %

As the images show the result is very similar to the Michell truss. The oddly shaped squares (Steps 23,41 and 43 in Figure 2.17) can be disregarded, as they are a result of how ANSYS attempts to display displacements in the chords. The algorithm continues until too much material has been removed in step 42 and the structure no longer functions. Note that both supports are free to move in the horizontal direction, but that this movement is restricted due to the vertical symmetry plane in ANSYS at midspan which is automatically fixed.
This calculation took 1 minute and 9 seconds

The same algorithm was applied to a few other load cases as well as three-dimensional problems, all of which gave expected results. Using the information from Section 2.4, the algorithm was expanded to BESO. As mentioned, the higher complexity and additional difficulties in coding in ANSYSscript made the development time longer.
The first working version of BESO in ANSYSscript was called version 2.04. Previous versions did not work to satisfaction as they were either work in progress or still in debugging stages.

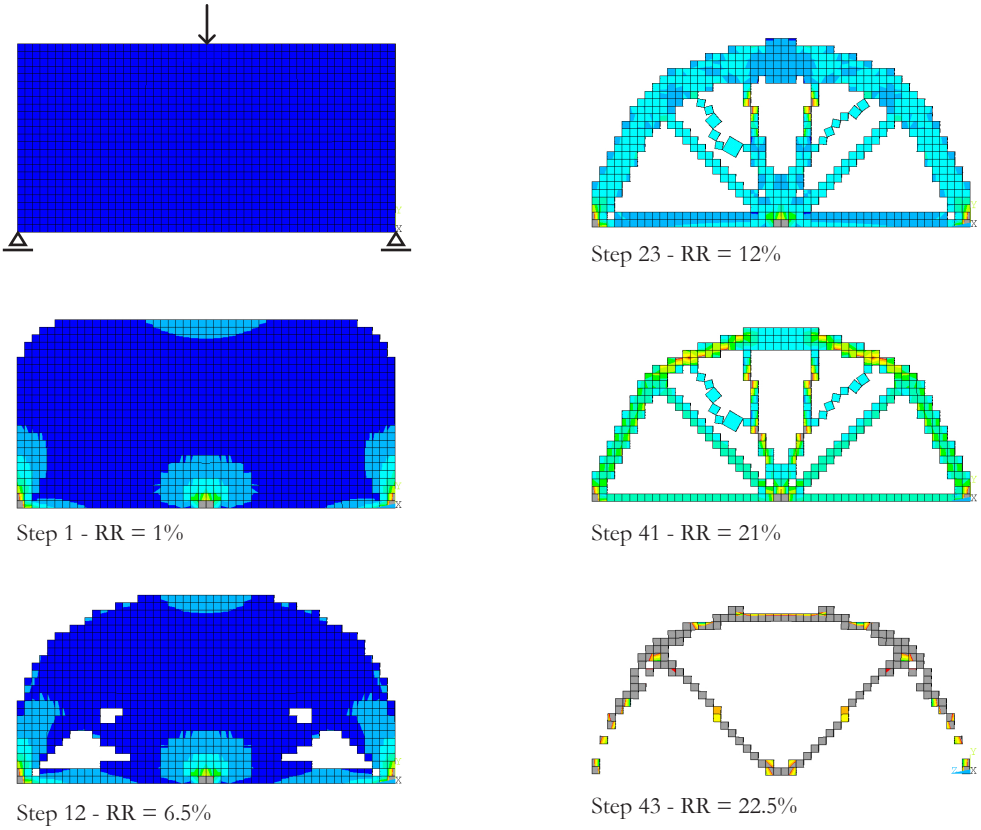
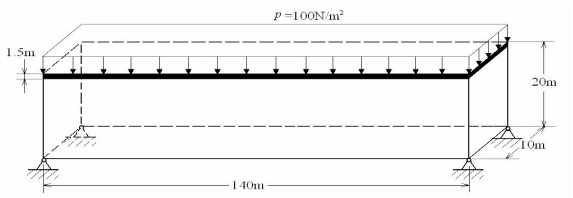
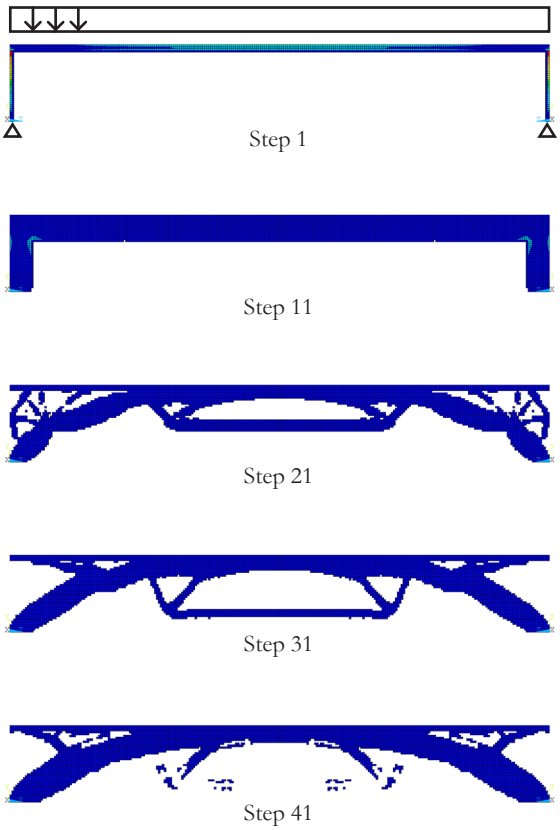


Figure 2.17
The results of ESO for a simply supported volume subjected to a pointload at midspan at several steps throughout the process.

To test version 2.04 the dimensions, loads and support conditions of an example by Huang (2006) was used. This was one of very few BESO examples found in which input and output were comprehensively described.
One problem was the fact that TU Delft licenses for ANSYS did not allow the required amount of elements that were necessary for the fac simile. Because of this, the width of the domain was changed from 10m to 1m, corresponding to a thickness of one element, mirrored along the length. Also, because Huang used $RRV = 0.80$ for a three-dimensional case, it was clear that a lower RRV had to be chosen for this one-element-thick slice to prevent too much material (comparatively) from being removed. The three-dimensional example has a varying volume distribution along the depth so that a longitudinal cross-section halfway through the depth would have more than 20% volume. An estimated $RRV = 0.58$ was used as a result. To reduce computational effort, two planes of symmetry were used to mirror calculations. Because of this the sup-





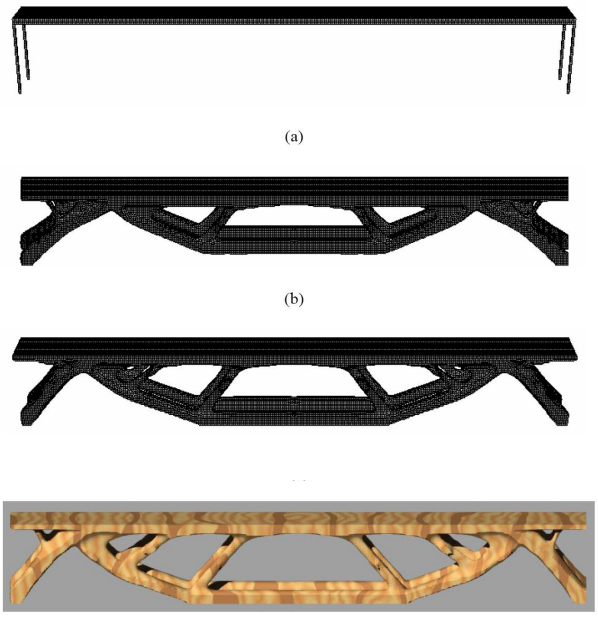


Figure 2.18
The left side of the page shows the results from BESO 2.04 compared to the reference results by Huang et al. (2006) on the right side. The best result by Huang et al. was structure (e) and the arch at the bottom right was forced in a second run where they removed initial domain material at midspan.

ports have to be the same as well. Note that the top three sheets of elements, on top of which a distributed load is applied, remain fixed throughout the process. The following input was used which gave the corresponding output values.

TABLE 2.2 Properties for BESO version 2.04 in ANSYS and Huang (2006)

		Version 2.04	Huang (2006)
modulus of elasticity	E	210 GPa	idem
distributed load	q	100 N/m^2	idem
dimensions	$w \times h \times l$	1 x 20 x 140 m	10 x 20 x 140 m
element size		0.5 m	idem
BESO parameters			
removal rate of volume	RRV	0.58	0.80
results			
volume ratio	V	42 %	20 %
performance index	PI	4,9 x 10^{-2} $N^{-1}m^{-4}$	5.9 x 10^{-4} $N^{-1}m^{-4}$

Though not an exact match due to the volume difference, version 2.04 gave expected performance results Figure 2.19) compared to the example by Huang et al. (2006). (However, on visual inspection, it is clear that the shape diverged from Huang et al. (2006) beyond step 31, where at one point the tension chord was removed in favor of an arch like shape. The graph below shows the development of the PI over time and demonstrates how the arch shape of step 71 had a higher PI than the tension chord shape at step 37. Huang et al. (2006) forced the algorithm to form an arch shape as well, but it apparently had a lower PI of 3.51 $N^{-1}m^{-4}$. The total calculation took 4 hours and 11 minutes.

At the moment new licenses obtained by the TU Delft for ANSYS no longer have element restrictions. A quick run of BESO with the correct amount of elements revealed that the performance index had the same order of magnitude as the example in Huang et al. (2006), verifying the correct implementation of BESO.

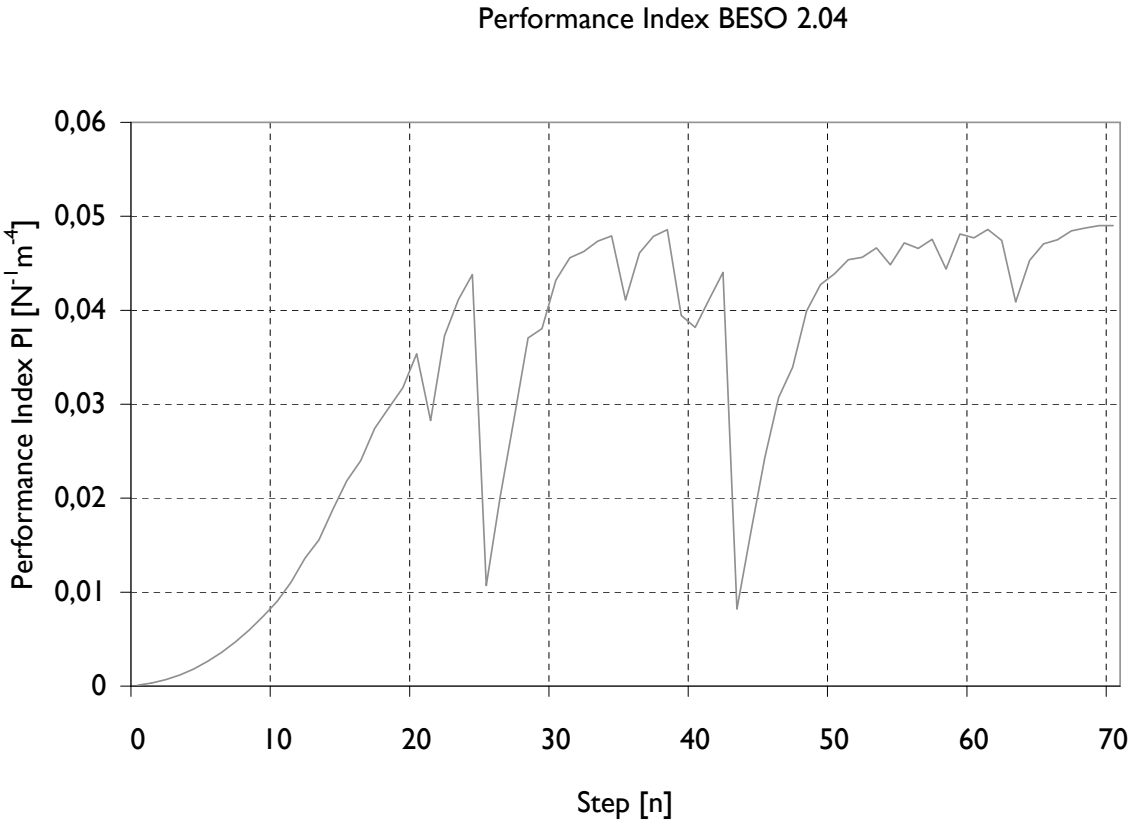


Figure 2.19
The performance index over time for BESO 2.04

Following the previous example, the algorithm was applied to several other, slightly different load and support conditions. Before doing so, the aforementioned RRV_{rel} (see the end of Section 2.4) was introduced to the programming. Because this parameter now refers to the removal rate of volume at one point in time for one iteration, its absolute value is far lower than what is conventional for the RRV. Typically the $RRV_{rel} = 1 \sim 10\ \%$ depending on the element size, while the $RRV = 10 \sim 90\ \%$ depending on the initial domain.

The other values are the same as in the previous example. The structure on this page has rolling supports instead of fixed ones. This in turn leads the structure to retain a bottom tension chord to avoid displacements of the supports. Many of the intermediate shapes during optimization are similar to the optimal result by Huang et al. (2006) with the exception of the included tension chord.
The best solution features quite angular connections, reminiscent of a truss. Several chords and members can clearly be distinguished.

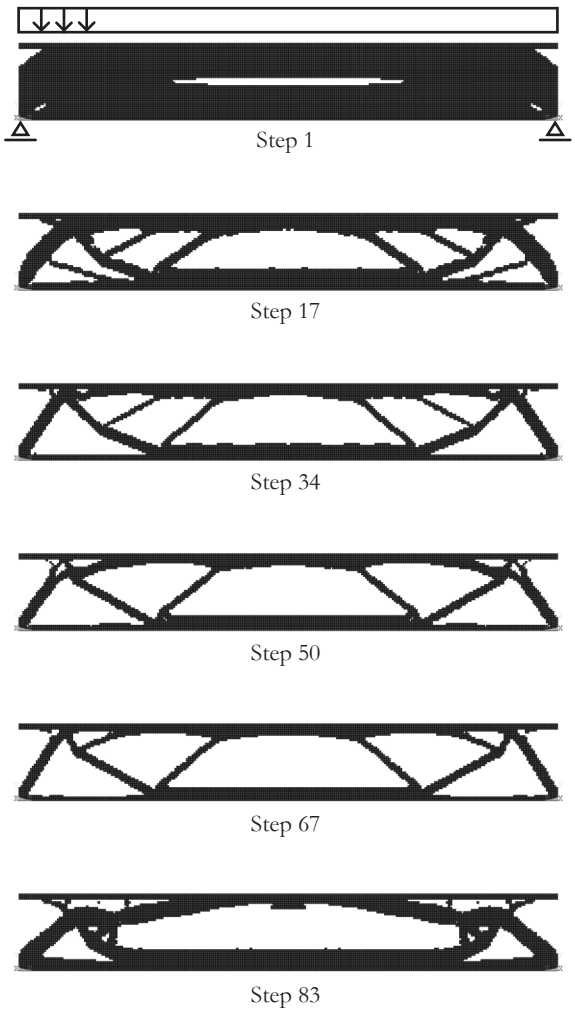


Figure 2.20
The results during optimization using BESO 2.05 for a simply supported, 'standing' volume subjected to a distributed load.

TABLE 2.3 Properties for best solution

Version 2.05		
removal rate of vol.	RRV_{rel}	0.05
best result		
step		68
computational time		7 hours 21 min.
volume ratio	V	36 %
performance index	PI	$2.9 \times 10^{-2}\ N^{-1}m^{-4}$

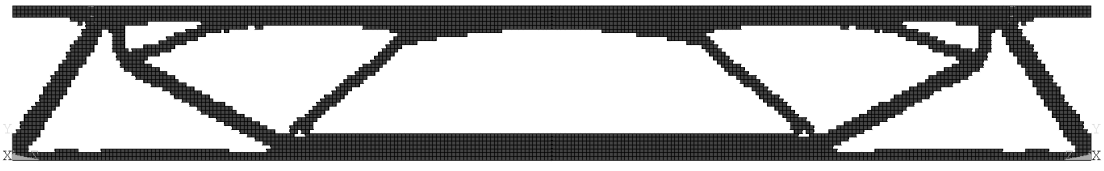
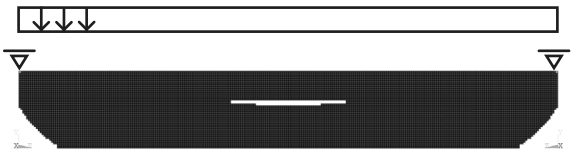


Figure 2.21
The best performing result for this run of BESO 2.05

Another situation was tested in which the supports were at the top. In this case the structure evolves diagonal thin members radiating from the center to transfer the loads to a thick bottom chord.

The resulting PI is the highest of all three cases, which all had the same domain and loads. It seems that this solution offers the highest stiffness (defined by strain energy) per unit of volume.



Step 1

Step 15

Step 29

Step 43

Step 57

Step 71

Figure 2.22
The results during optimization using BESO 2.05 for a simply supported, 'hanging' volume subjected to a distributed load.

TABLE 2.4 Properties for best solution

Version 2.05		
removal rate of vol.	RRV_{rel}	0.05
best result		
step		35
computational time		4 hours 48 min.
volume ratio	V	42 %
performance index	PI	$5.2 \times 10^{-2}\ N^{-1}m^{-4}$

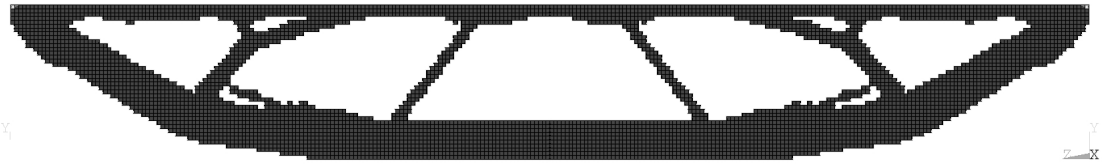


Figure 2.23
The best performing result for this run of BESO 2.05

2.6 Conclusions on the BESO algorithm

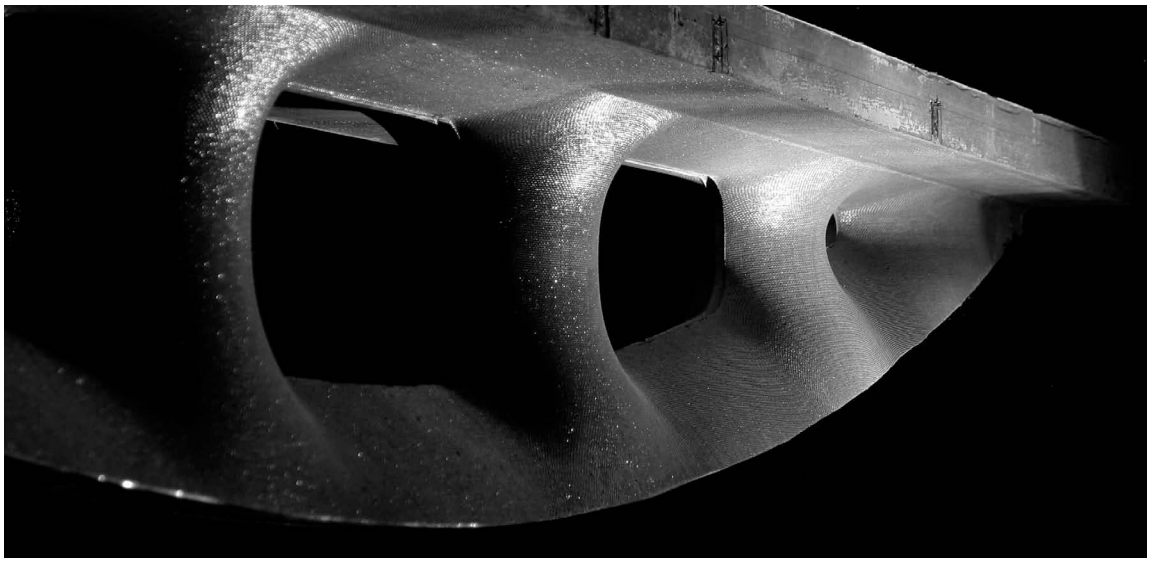
The implementation of BESO in ANSYSscript was succesful. Results are comparable to those of current papers on BESO, while visually all optimal results show some inherent logic in how forces are transferred to the supports.

The addition of a new RRV_{rel} parameter resulted in a faster, more robust algorithm.

The existing results that BESO offers are in no way practical. The material properties correspond to some fictitious steel-like material. Furthermore, the applied load of $100\ N/m^2$ is relatively small and only serves to drive the optimization process. Since this load is unrealistically small, deflections are equally small and ultimate stresses of the material - even though strictly speaking, none are defined - are never reached. Because of this, removal of volume outweighs the maximizing of stiffness, thus results might prove to be unrealistically slender, depending on what type of real material is considered.
These issues are taken into account in a non-linear concrete model that was applied during the course of this thesis project. To this end, the way in which strain energy was calculated for the performance index PI was altered as well. Because this non-linear model was applied both to BESO and to `FABRICFORMER`, the description is not given here, but in Chapters 6 and 7. The results of this model are discussed in Chapter 8.

Even though the results from this chapter can not directly be used in practice, they form a solid reference for the optimization of fabric formed beams, as they indicate what type of shape and topology is preferable. In other words, they offer a qualitative, not quantitative, frame of reference.

On a final note, it seems that the goal-oriented nature of BESO, where pure mechanical properties drive the optimization, would make incorporating manufacturability a difficult task. However, as soon as cost-effective manufacturing methods exist that have total three-dimensional freedom this would become less of a problem.
For the purposes of this thesis, the decision is made to rely on a wholely different, general purpose algorithm as discussed in Chapter 4 in order to incorporate constraints posed by the manufacturing method described in Chapter 3.



CHAPTER 3 Fabric Formwork technology

Fabric formwork technology refers to the use of strong fabrics as a mold for the casting of concrete. It is not a particularly new concept, and there are certainly enough examples of foundations and submarine concrete structures made with fabric formwork. However, recent years have seen more novel applications in building engineering, adding up to wide range of element types. It is fair to say that the true potential of fabric formwork is yet to be fully explored and there is a clear rise in academic interest and research on the topic. The first conference on fabric formwork was held in May of 2008, underlining that developments in this area are both new and exciting. This chapter will detail the current state of the art of fabric formwork and provide information from a civil engineering point of view. Particular interest will go to fabric formed beams, the focus of this thesis.

The flexible nature of fabric allows very different shapes of molds than those resulting from conventional mold materials like steel or timber. The range of possible shapes includes various double curved shapes, sometimes reminiscent of organic shapes in nature. Although these kinds of geometries were already feasible, standard manufacturing methods have not enabled them to be mass-manufactured in any economically competitive fashion and often involve high use of energy. Fabric formwork has the potential to offer affordable, simple casting methods for complex shapes.
Other than technical differences with traditional means of casting concrete, one must also assess the impact of fabric formwork on the resulting elements. From an architectural standpoint especially, the introduction of fabric formwork requires a new way of regarding the design and esthetics of concrete structures. From a civil engineering point of view, a much higher material efficiency can be attained, as will be shown in this chapter. This has different consequences for both serviceability and ultimate limit states. It is also important to keep in mind that this efficiency in material use is offset by a loss of redundancy. This reduction could in turn lead to a lower degree of safety.
Other advantages include those associated with the weight and volume reductions of the mold, such as cost reductions on materials, transportation and storage.

3.1 A brief historic overview of fabric formed concrete

The existence of structural concrete can be traced back to Roman times. Traditionally, fresh concrete, or concrete slurry, has been poured into rigid molds, or formworks, generally made of timber or steel.
The idea of using fabrics instead is much more recent and may be attributed to the developments in the textile industry during the Industrial Revolution. Newly invented looms and other textile machinery provided mass quantities of fabrics with a high and also consistent quality. Indeed, patents dating as far back as the 19th century concerning the use of fabric as a formwork can be found.
One of the first examples of fabric formed structural concrete were low cost school buildings in Mexico by Spanish architect Felix Candela in 1951. Here, fabrics were draped over timber profiles to cast shell structures (Pedreschi et al., not dated).
The first applications on a large scale took place in the mid 1960's, mainly for erosion control structures and pond linings. Another increasingly popular use is found in the building industry for the casting of foundations and footings (West, not dated).
Another early example of using fabric formwork for structural elements is the 1970's Spanish architect Spanish Miguel Fisac. He employed thin plastic sheets as formwork for textured wall panels in some of his designs (West, not dated).

In the early 1980's, but especially from the late '80s onward a large number of patents can be found dealing with various set-ups of fabric formwork. Most of these deal with footings, wall panels and columns.

The main source of current development in fabric formwork can traced to professor Mark West at the University of Manitoba, who has spearheaded the founding of the Centre for Architectural Structures and Technologies (C.A.S.T.). This centre specifically focuses on the academic research on fabric formwork and its impact on architecture and building engineering. At this point fabric formwork has received some interest in the global academic community. In fact, as mentioned, a first conference on the topic was held in May 2008 at C.A.S.T., which hosted most, if not all researchers of fabric formwork.
Dr. Mark West identifies two other notable driving forces behind the development of fabric formwork in the 1980's and 1990's;

- Japanese architect Kenzo Unno has designed several buildings that have in-situ cast fabric formed concrete walls.
- Rick Fearn, a Canadian businessman, developed techniques for the casting of foundation footings in fabric formwork. His company, Fastfoot Industries in Surrey, Canada has been exploiting these methods for several years with success.

In early 2008 the International Society of Fabric Forming (ISOFF) was founded to provide a central point for development of fabric formwork technology. It has the following goals:

- Improving communication between all participants: researchers, architects, manufacturers, distributors and concrete contractors;
- Communicating to the world the commanding environmental benefits of fabric forming over rigid formwork;
- Developing new and innovative fabric forming solutions.

At this moment (June 2008) the following universities have ongoing research on the topic, though in some cases only a single MSc or PhD dissertation.

- Vrije Universiteit, Brussel, Belgium
- University of Manitoba, Winnipeg, Canada
- University of Toronto, Canada
- Universidad Católica de Valparaíso, Chile
- Danish Royal Academy of Fine Arts, Denmark
- University of East London, UK
- Bartlett School of Architecture, University College, London
- University of Bath, UK
- Delft University of Technology, the Netherlands
- University of Edinburgh, Scotland
- Massachusetts Institute of Technology, Cambridge, US

Industry research into applications of fabric forming technology is flourishing around the world, now including:

- RP Schmitz Consulting Engineers, Brookfield, WI, US
- Fab-Form Industries Ltd. , Canada
- Fastfoot Industries. Surrey, Canada
- Enviroform (UK) Ltd., Wakefield
- Umi Architectural Atelier, Tokyo, Japan
- Arro Design, VT, US
- Ellendale Concrete Products, LLC, US
- Sure Safe Industries, San Diego, US

The current applications of fabric formwork found at C.A.S.T., in various patents and several manufacturers found on the internet are:

Hydraulic engineering
- wall structures
- erosion control structures
- pond liners
- subsea mattresses for protection

Building engineering
- foundation footings
- trusses and beams
- columns
- floor slabs
- walls and wall panels
- scour protection for buildings
- shells and domes

A quick online search for fabric formwork yielded an additional number of companies using it for the casting of subsea mattresses; e.g. Seamark Systems UK, Pro-Dive Marine Systems, ULO Systems LLC.

3.2 Fabric formwork methods for beams

The focus of this thesis lies specifically on fabric formed beams, but there are only a few examples available. Several types of beams have been made at C.A.S.T. The first fabric formed, reinforced concrete beam was cast by Con-Force structures, a prefab concrete manufacturer, in Winnipeg, Canada. The University of Edinburgh, Scotland, has also cast fabric formed beams for research in cooperation with C.A.S.T.

The current methods of casting fabric formed beams can all be traced back to the research efforts at C.A.S.T. This paragraph gives a summary overview of these different methods. There are now three distinct methods (Figures 3.1-3.3):

- Flat sheet and spline method
- Keel mold method
- Pinch mold method

The methods, as will be explained in more detail, rely on using flat sheets of fabric as opposed to custom tailoring, sowing, cutting etc. This is a conscious effort by those at C.A.S.T. to develop simple, practical methods of creating fabric molds. This strategy, though noble in its intent to create a widely available technology limits the range of shapes that could be imagined, also because care has to be taken to avoid significant folding and wrinkling of the fabric. But, it is clear that avoidance of folding and wrinkling is not as crucial as in tension structures, because when present to a limited degree, it only causes esthetic problems and slight material inefficiency.

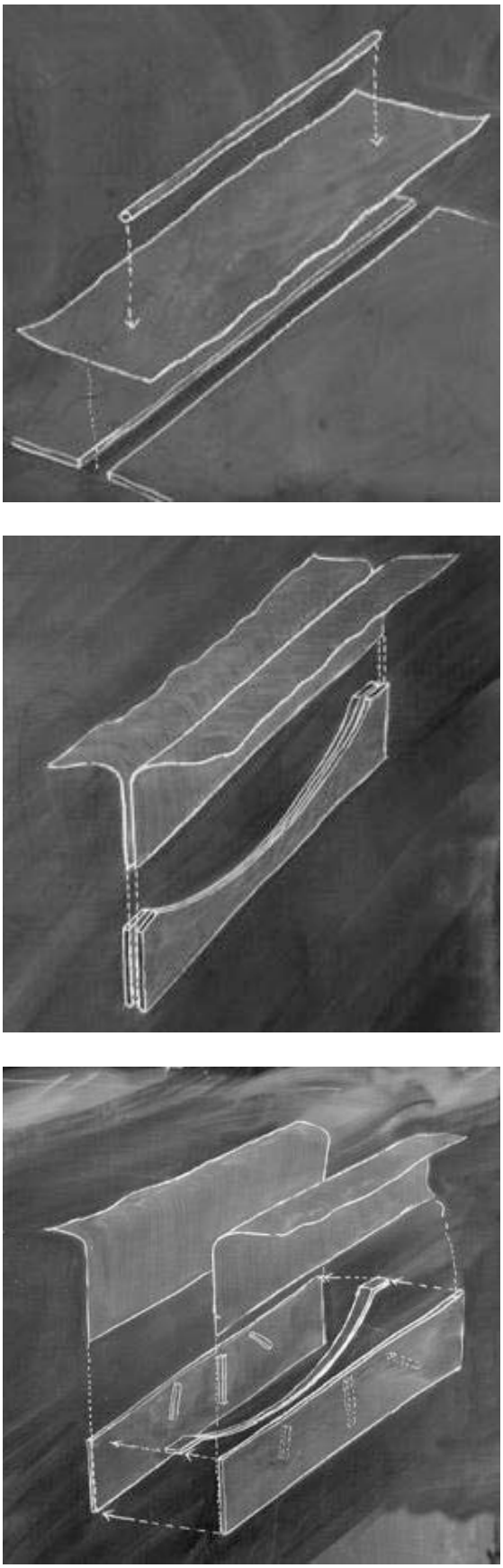


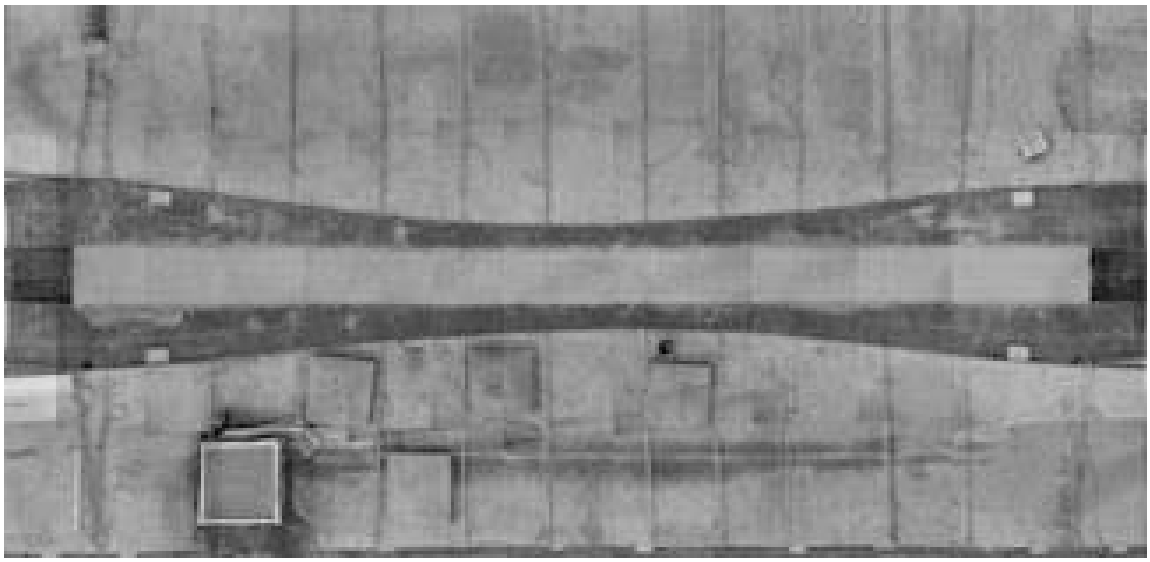
Figure 3.1-3.3
The three existing methods of casting beams in fabric formwork, developed at C.A.S.T. in Canada; the flat sheet and spline mold, the keel mold and the pinch mold.

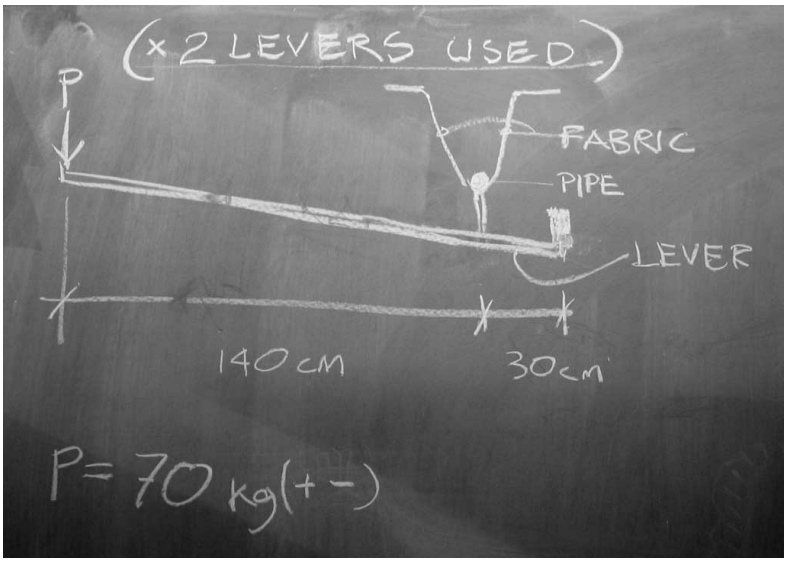
3.2.1 Flat sheet and spline method

The rig of this first method consists of one flat sheet of fabric, one spline and a space to pull the fabric down into an open trough. The spline is used to pull the fabric downwards so that the bottom of the beam follows a rough approximation of the beam's bending moment curve. Pulling the spline downwards also serves to vertically pre-tension the fabric sheet, thus reducing concrete volume in the tension zone of the beam. In one example, two levers were used to each give a downward prestressing force of 650 kg i.e. about 6.5 kN. At the other end, the edges of the fabric sheet were supported by strands of rebar along the length. According to C.A.S.T., when the web is shaped according to the shape of the bending moment diagram, weight savings of 30-40% were possible compared to equivalent rectangular beam.

In past experiments there were problems with plastic behaviour of the spline at midspan, due to the high curvature and stresses at that point. If future experiments take place, splines with larger cross-sections and/or higher plastic yield stresses will be used, possibly made of fiber reinforced polymers.

Major development of this project has seized, due to the other methods offering a greater degree of geometric freedom and larger sections due to the use of two flat sheets instead of a single one.

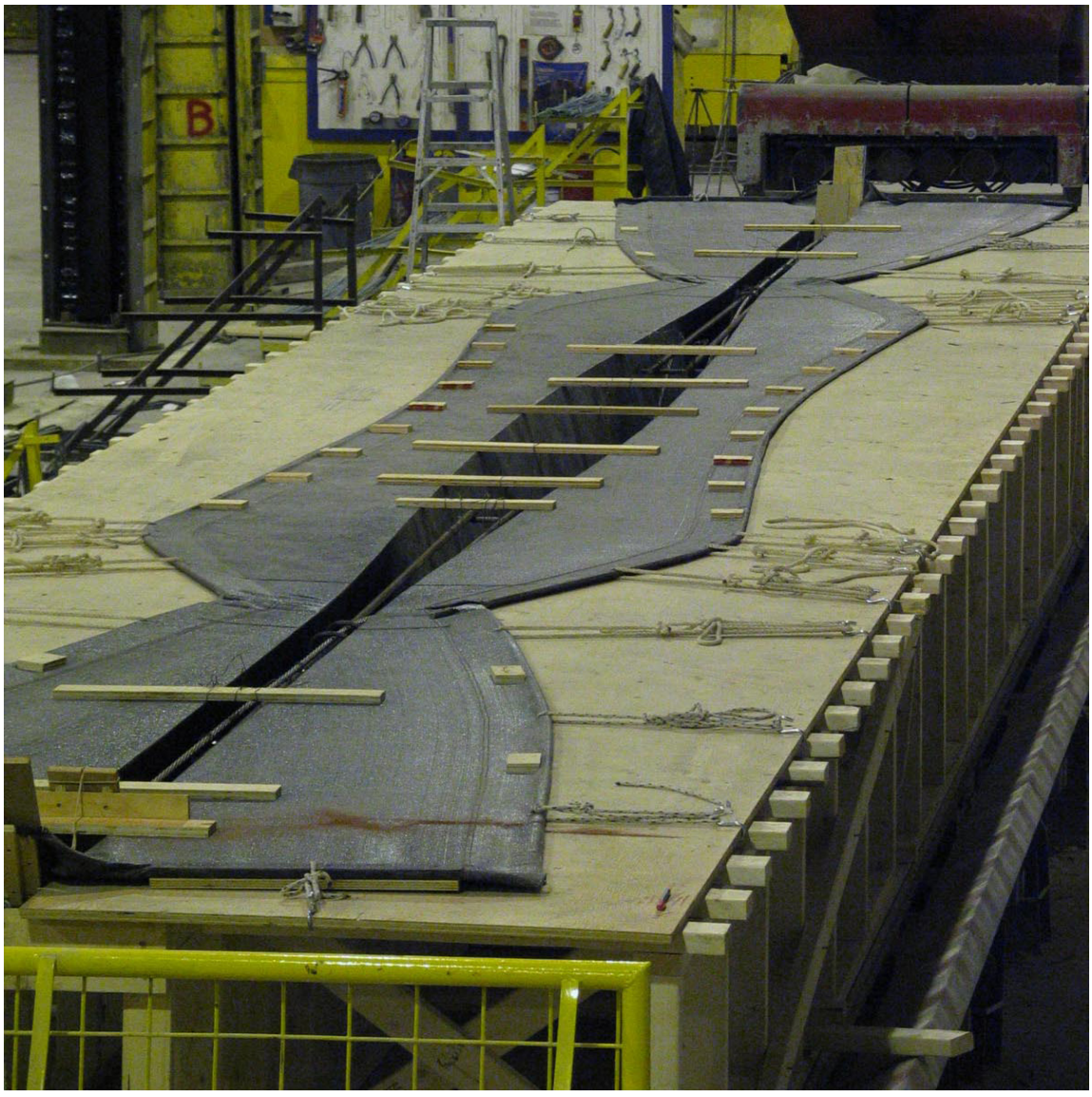


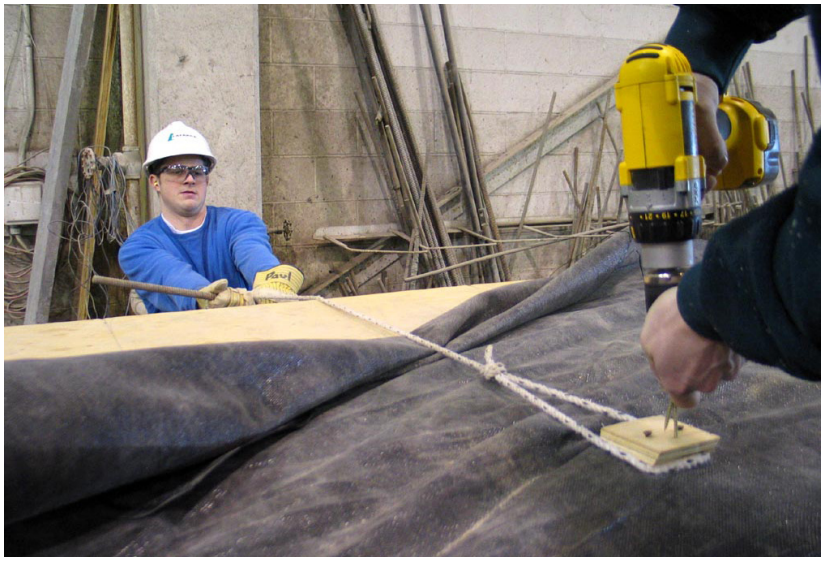

(x 2 LEVERS USED)
P
FABRIC
PIPE
LEVER
140 cm
30 cm
P= 70 kg(+-)

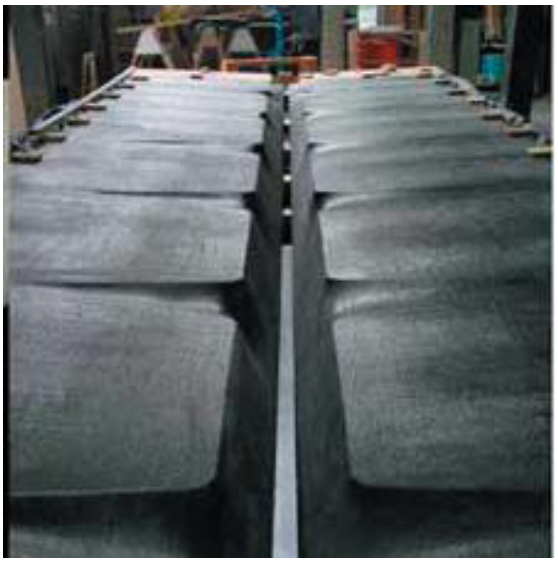
3.2.2 Keel mold method

This method uses a pair of flat sheets of fabric, which are sandwiched between two identical halves of a keel, made of (3/4") plywood. This keel mold has been cut to match the bending moment curve of the beam, providing a fixed geometry for the longitudinal shape of the beam. In some cases a spacer strip has been used to widen the bottom edge of the beam.

At the upper side of the mold, the sheets are prestressed along either a wooden deck, or a set of steel pipes. The prestressing is done at several points along the length of the fabric, using ropes to pull them down or by fastening them with wooden blocks.

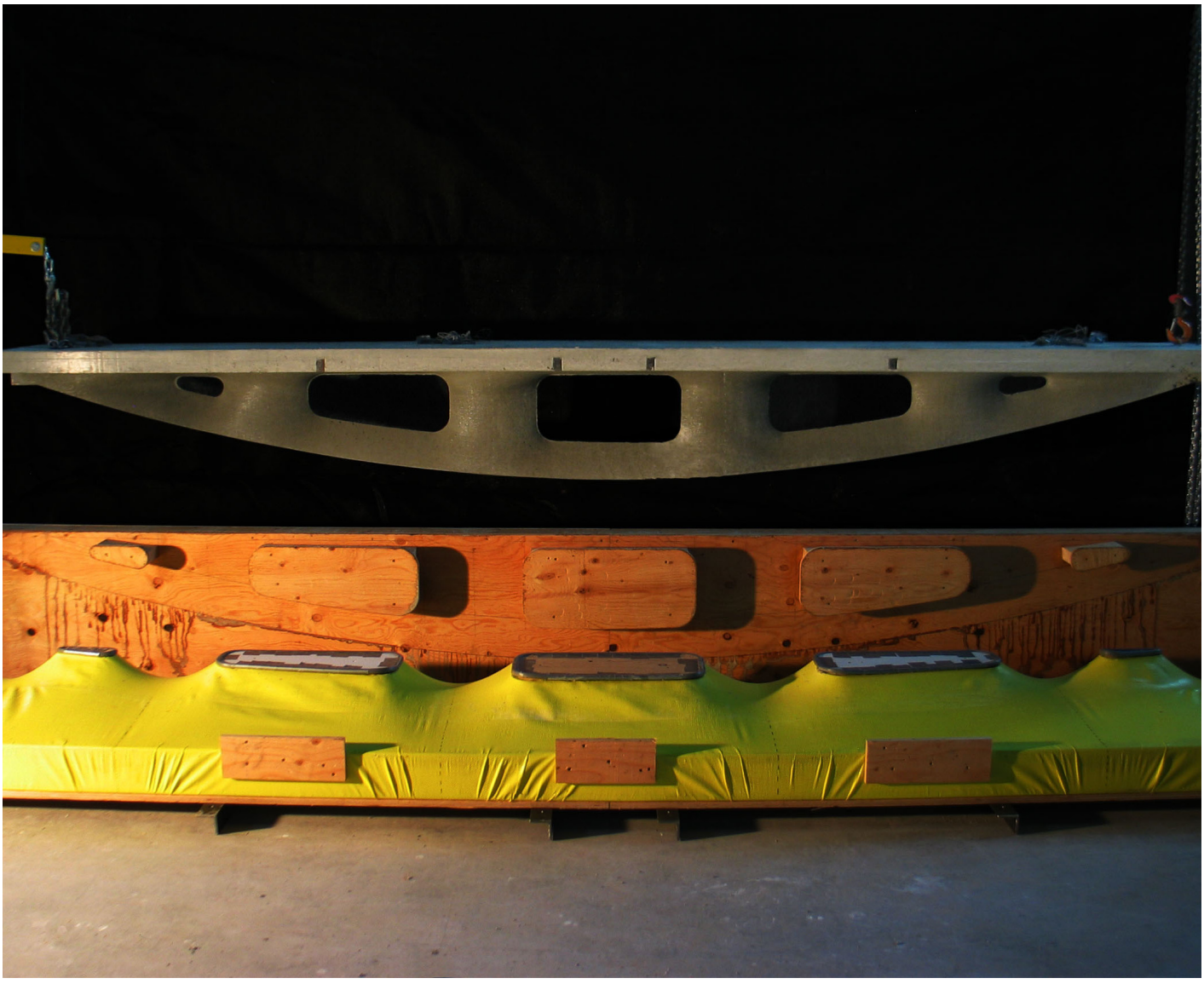

B

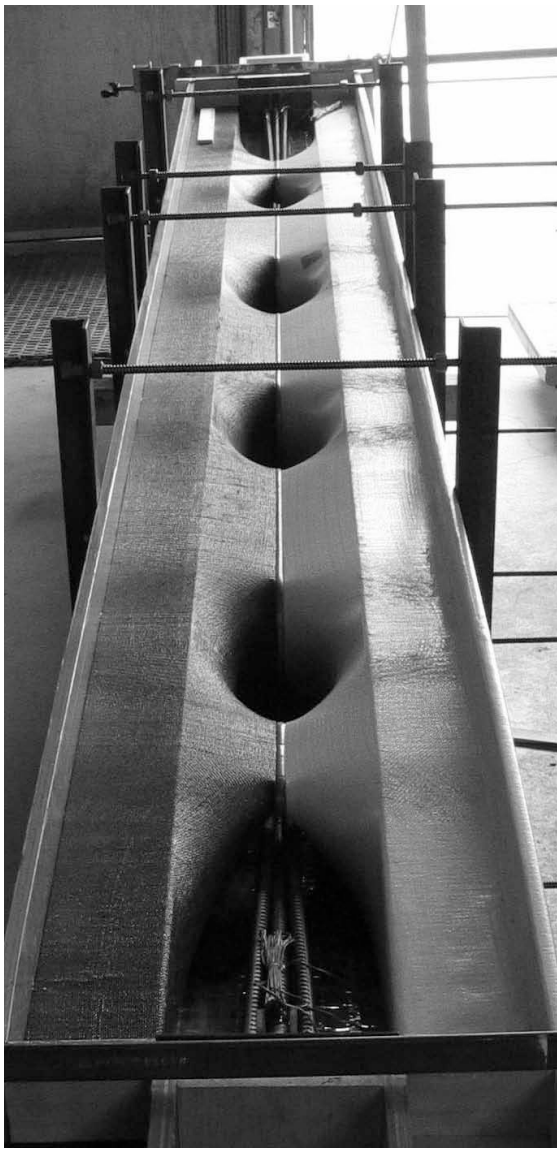


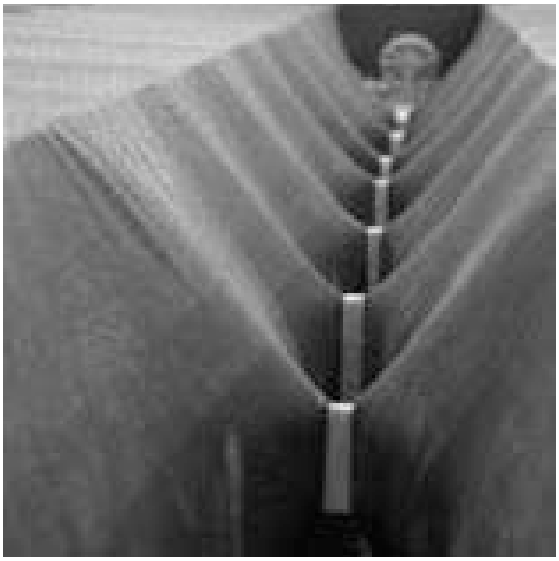


3.2.3 Pinch mold method

The third and last method, called the pinch mold, offers the most exciting shapes of all these methods. Again, a pair of rectangular sheets are sandwiched, in this case by two rigid panels. At certain points, these panels have protruding elements that ultimately 'pinch' the mold resulting in holes throughout the length of the final beam. These 'pinch points' make it possibly to produce concrete trusses in relatively easy ways and potentially allow much more efficient geometries than the other two methods. However, this advantage is at the cost of higher use of timber, and therefore a more expensive and labor intensive mold. C.A.S.T. has made a wide range of plaster models to show the possibilities of pinch molding.







3.3 Fabric types used in fabric formwork

The fabrics used for fabric formwork are cottons, and more often polyolefins. Polyolefin is a category of polymers which include polypropylene and polyethylene. Beams cast in Winnipeg and Edinburgh used either woven polypropylene geotextiles, specifically Propex 2006 and Lotrak 300GT or 315ST, formerly known as Amoco #2006, or coated high density polyethylene (HDPE), specifically Nova Shield.
The geotextiles were chosen for their availability and low cost. The permeability of a geotextile leads to an improved surface finish of the concrete. This effect will be discussed in more detail in Section 3.4.
The HDPE fabric on the other hand has superior mechanical properties and also more easily facilitates demolding than the geotextiles.

From a structural engineering point of view, using fabrics requires a different engineering approach than conventional building materials. The material is orthotropic and highly non-linear as it will stiffen under increasing tension until some ultimate strength is reached. Thus, the Young's modulus is not a constant, and also differs for each direction, as the warp and weft directions of woven fabrics exhibit different mechanical properties. Designing tension structures however, is often done using a linear approximation of the stiffness, which in turn among other factors results in relatively high safety factors in structural calculations. Indeed, manufacturers offer single values for the mechanical properties in the warp and weft directions, and most engineering software for tension structures allow a single value input for these two stiffnesses.

It is noted that there is a significant difference in nomenclature and use of mechanical properties between the textiles industry and civil engineering. In the former the properties are described in wide width tensiles *[N/m]*, grab tensile strengths *[N]* and wide width elongation *[%]* whereas the latter uses strengths *[N/m]*, strains *[‰]* and stiffnesses *[N/m]*. Based on the way in which these values are derived and defined, it is concluded that they are not readily interchangeable.

TABLE 3.1 Properties of fabric formwork materials

	Lotrak 300GT	Propex2006	Nova Shield
Physical properties	Warp / Weft	Warp / Weft	Warp / Weft
Grab tensile strength	1.40 kN	1.4 kN	1.624 / 1.491 kN
Grab or apparent elongation	15 %	15%	
Wide width tensile	30.7/30.7 kN/m	40 kN/m	45.6 / 42.08
Wide width elongation	15/8 %		
Mullen burst	4650 kPa		4692 kPa
Puncture strength	0.667 kN	0.8 kN	
Trapezoidal Tear	0.533 kN	0.5 kN	0.356 kN
UV Resistance	70 % at 500 hr		
Apparent Opening Size	0.425 mm	50 US Sieve	
Permittivity	0.05 sec^{-1}	0.05 s^{-1}	
Flow Rate	160 L/min/m^2	200 l/min/m^2	
Mass per unit area		70	224 g/m^2
			Tested according to ASTM codes

Propex 2006 is a polypropylene woven tape/fibrillated fabric. According to its manufacturer, this engineered geotextile is stabilized to resist degradation due to ultraviolet exposure. It is resistant to commonly encountered soil chemicals, mildew and insects, and is non-biodegradable. Propex costs around 50 eurocent per m^2.

The mechanical properties from Table 3.1 – supplied by their respective manufacturers – are obtained from stress-strain curves developed in accordance with the standard test methods of ASTM, the American Society for Testing and Materials. All test methods are also available in Europe from ISO, the International Standards Organization. In general, these tests prescribe how stress-strain curves are derived for the warp (or machine) and weft (or cross machine, or fill) direction. An approximately linear region is then determined from which three moduli can be constructed, the initial, tangent and secant (or offset) tensile moduli.

Schmitz (2004) used Amoco #2006 and determined such moduli, using the appendices of ASTM D4595 and stress-strain data from the manufacturer (Table 3.2). According to him, there is little interaction between the two perpendicular directions in a woven fabric and a value of zero for Poisson's ratio was chosen for the model.

TABLE 3.2 Properties of Amoco #2006

Physical properties		
Modulus of elasticity	E_{warp}	644 N/mm²
Modulus of elasticity	E_{fill}	329 N/mm²
Shear modulus	G	164 N/mm²
Poisson's ratio	v	0.0
fabric thickness	t	0.76 mm
	Values according to ASTM D4595	

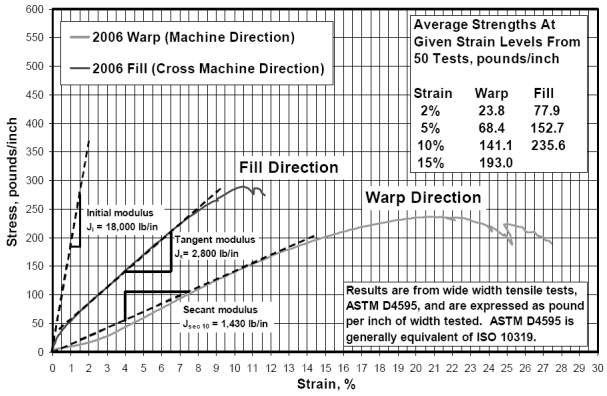


Figure 3.4
Stress-strain diagram for Amoco #2006, using ASTM codes for the test and subsequent derivation of linearized elasticity moduli.
Schmitz (2004)

3.4 Concrete hardening in fabric formwork

Up to this point conventional types of concrete have been cast in fabric molds. In general the fabric can be stripped off after three days of concrete hardening, while during these first days special care has to be taken to protect the flexible formwork.

An interesting apect of woven textile formworks is the permeable nature of textile, which allows air and excess water to pass freely from the fresh concrete as it hardens. This results in a lower water to cement ratio, causing higher compressive strengths as well as fine grained surface finishes, avoiding the need for additional expensive surface treatments (e.g. sand blasting, acid etching, veneers).
Al Awwadi Ghaib & Górski (2000) carried out destructive tests on various concrete mixes in combination with four synthetic fabric types. They also measured the quantities of fluids penetrating through the fabrics. Their conclusions were:

- About 30% of the total discharged liquids drained out in the first 5 minutes of the test. Most of the discharges occurred in 15-30 min.
- The amount of lost cement was governed by the initial w/c ratio in three of four tested fabrics.
- Most of the w/c ratios were smaller than the initial ratios for most of the mixes in three of the four tested fabrics, ranging from -2% to 20%.
- The concrete in the fabric formworks set rapidly, even though the initial w/c ratio was high.

Independent of the type of concrete or fabric type, the same trend was seen in the increased compressive strength. Their analysis showed:

- more rapid setting in fabric than in steel molds: 65% of final strength in 3 days compared to 7 days.
- compressive strength dependant on fabric pore size, highest with moderate pore size (0.35×10^{-3}m) due to reduction of w/c ratio before setting.
- that it is not right to generalize the fact that concrete cast in fabric form is characterized by a higher compressive strength than the concrete cast in steel molds. This is due to fact that at higher pore sizes cement and very fine sand particles also permeate and are lost.
- at lower pore sizes the compressive strength decreased due to clogging of the pores.

After statistical analysis the following equation for a 28-day compressive strength $f'c$ was assumed:

$$f_c' = 15.34 - 33.65(w/c)^3 + 140.92q^3 + 257.2(mos) - 625.4(mos)^2 + 367.72(mos)^3 \quad (3.1)$$

where w/c = water/cement ratio
q = cement quantity in $(kg/m^3)\cdot 10^{-3}$
mos = fabric microsopic opening size in $m\cdot 10^{-3}$

From test data they concluded a relative increase in compressive strength for pore sizes between $0{,}15\cdot 10^{-3}$ to $0{,}57\cdot 10^{-3}$ m, with a maximum at $0{,}35\cdot 10^{-3}$ m. However, these results are inconsistent with the devised relation as the maximum f'_c values of Figure 3.5 are never at $0{,}35\cdot 10^{-3}$ m and the values f'_c for $0{,}15\cdot 10^{-3}$ and $0{,}57\cdot 10^{-3}$ m are never the same, which should be the reference strength for conventionally casted concrete. Also, when plotting Eqn. (3.1) ourselves and comparing them to the plots (Figure 3.5) by Al Awwadi Ghaib & Górski (2000), we must conclude that the relation describes the lower bound of their plots, which implies that a more significant part of his test results deviate from the prescribed values of (3.1). It has not been ascertained where this inconsistency originates from.

Personnel at our own TU Delft concrete laboratory noted that the mixtures by Al Awwadi Ghaib & Górski were uncommon, featuring relatively low amounts of sand, high amounts of gravel and generally either very low or very high densities (2200 or 2500-2600 kg/m³). F. Delijani is currently researching more relevant mixes at C.A.S.T. Also, the increased compressive strength is merely applicable to the top 5 to 10 mm of hardened concrete, which follows the third conclusion, that 'it is not right to generalize the fact that concrete cast in fabric form is characterized by a higher compressive strength than the concrete cast in steel molds', though for different reasons. It would not be unfair to say that the concrete hardening in fabric has limited practical ramifications.

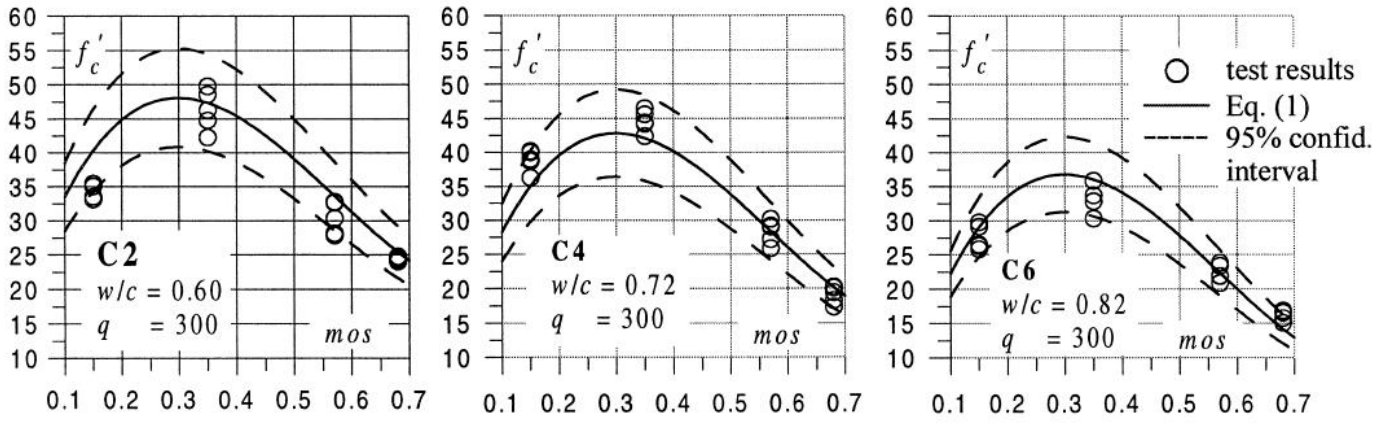


Figure 3.5
28-day compressive strength of fabric formed concrete as a function of w/c ratio and fabric microscopic opening size *Al Awwadi Ghaib & Górski (2000)*

3.5 Existing computational analysis of fabric formed elements

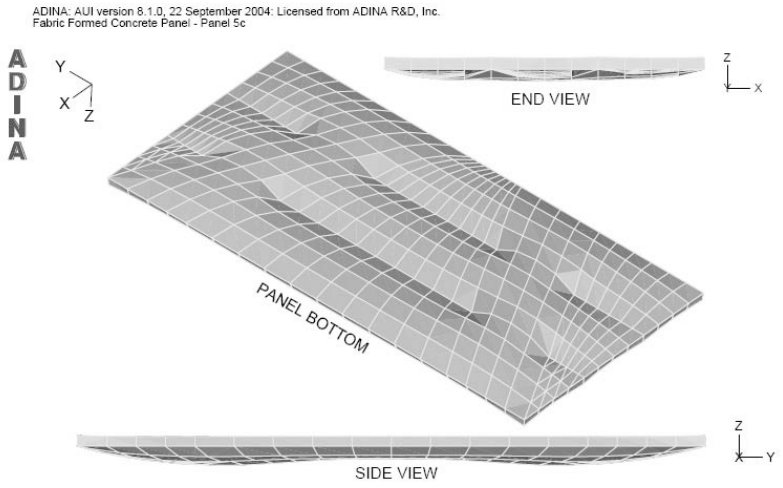
At the moment, there is exceptionally little structural analysis being done on fabric formed elements. Some of the research at C.A.S.T. and the University of Edinburg focus on structural aspects, but these mainly involve destructive testing and subsequent qualitative analyses (see Section 3.6).
One exception is Schmitz (2004) at the Milwaukee School of Engineering. Schmitz used FEA software program ADINA to analyze fabric formed wall panels using a four step iterative procedure:

1. Determine the paths the lateral loads take to the points where the wall panel is to be anchored.
2. Using the load paths, defined in Step 1, model the fabric and plastic concrete material as 2-D and 3-D Solid elements, respectively. These elements define the panel's lines of support.
3. "Form-find" the final shape of the panel by incrementally increasing the 3-D concrete elements until equilibrium in the supporting fabric formwork has been reached.
4. Analyze and design the panel for strength requirements to resist the lateral live load and self-weight dead load being imposed upon it.

The analysis did not include creep, used linearized stiffness in three directions (see Section 3.3), included various design loads and a non-linear reinforced concrete model.
Its conclusions were that a practical method of modelling fabric formwork had been developed and that one resulting panel required less material than panels of uniform thickness. The main proposal for follow-ups was automating the iterative process, since the form-finding still took place by manual node adjustments.
Final recommendations included investigating and analyzing reinforcement patterns, creep effects, other fabric types, fiber reinforced concrete in addition to performing full scale testing.

However, one recommendation was found missing.
Schmitz tested three specific panels, of which only one (see Figure 3.6) was found to be preferable over a panel of uniform thickness, considering the specific loading paths. Since it is unlikely that a uniform panel should always prove to be more optimal from a structural point of view, it seems that the form-finding process can be improved.
It was found that the procedure by Schmitz always used a prestressing of 2%, and this author would also recommend including prestress as a variable in the iterative procedure when form-finding.

No real use was made of this research, since the basically two-dimensional panel poses different problems than beams. Also the software program ADINA is relatively unknown and unavailable at this university.



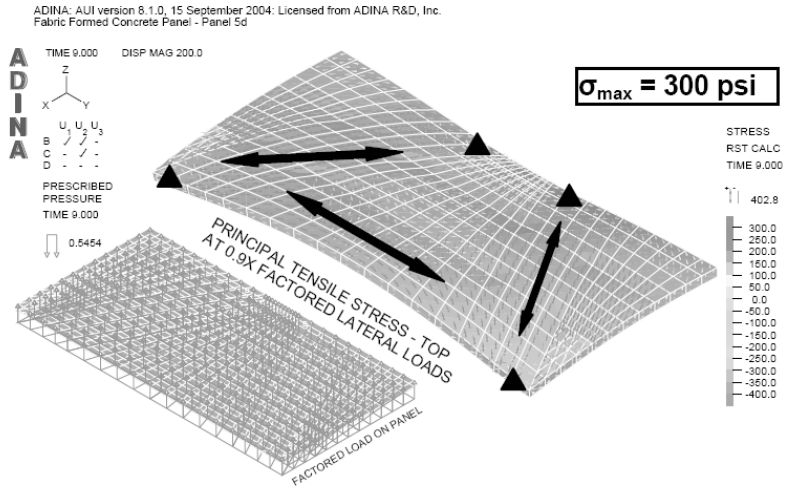


Figure 3.6
Fabric formed panel analyzed in ADINA, performing better than a panel of uniform thickness.
Schmitz (2004)

3.6 Current destructive testing on fabric formed beams

Limited testing has been done on fabric formed beams and the only known examples were carried out by Pedreschi & Lee at the University of Edinburgh and M. West & F. Hashemian at C.A.S.T.

Pedreschi et al. (not dated) tested a spline molded beam. The overall length of the beam was 3.16 metres. The length of the web was 3.00 leaving an 80 mm projection of the flange at each end. The casting rig was constructed using two beams supporting an 18mm thick plywood sheet. The beam was reinforced using two 10 mm mild steel bars, curved to follow the primary curve of the web. At the ends of the beam the reinforcement becomes horizontal and is anchored into the flange. The flange was reinforced with a 50 by 50 mm mesh using 3 mm diameter bars, positioned at the mid depth of the flange.

After testing, they concluded that "initial tests indicated that structural failure occurred at the anchorage of the beam at the bearings. In subsequent prototypes the thickness of the flange was varied to increase its thickness at the supports and increase the strength of the anchorage". Unfortunately, the cited source did not include details on how these subsequent prototypes performed. On the performance of the first beam, they concluded that "the web can be shaped to follow the shape of the bending moment diagram resulting in a 30-40% reduction in the weight of the concrete compared with an equivalent rectangular beam." It did not specify how this equivalency was defined e.g. either some serviceability or ultimate limit state criterion, the latter being most probable.

Figure 3.7
Model of the fabric formed beam cast and tested by Pedreschi et al.

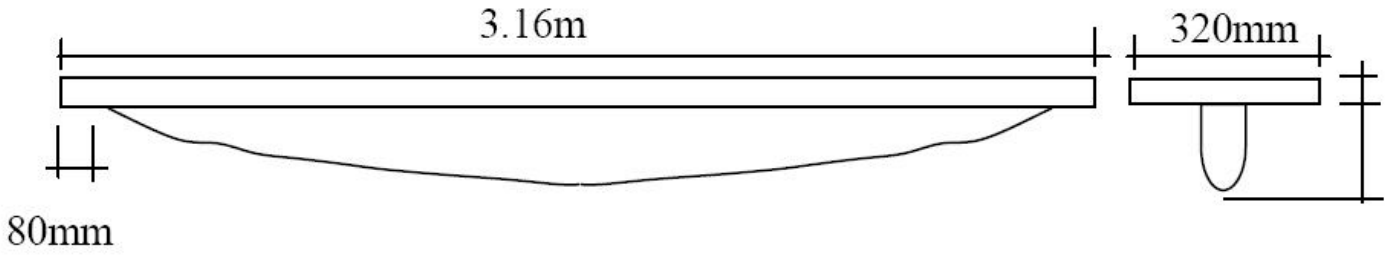
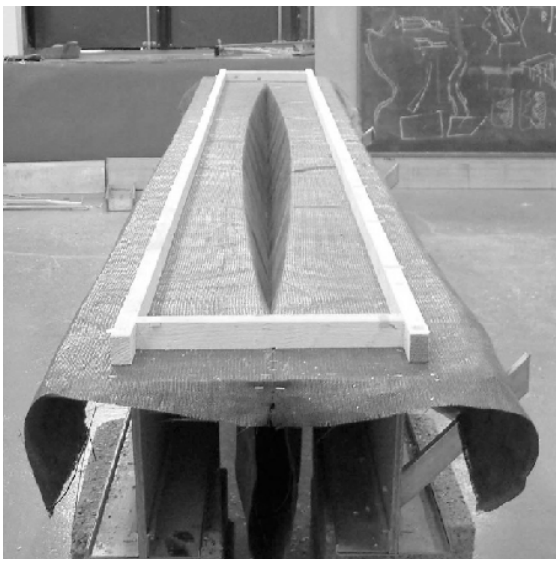


Figure 3.8
Photographs of the fabric formed beam cast and tested by Pedreschi et al.



West et al. tested several under reinforced beams which "were made to test these new formwork methods, they were not 'designed' in any engineering sense, aside from the establishment of their fundamental geometry and reinforcing patterns." As a consequence limited geometrical and mechanical properties are available on these beam tests. Due to their low amount of reinforcement, these beams proved highly flexible and testing equipment was unable to cause failure. There was a notable lack of shear cracking along the beam, which is a result of both the efficient bending-moment shaped form as well as the low stiffness of the beam. Current research at C.A.S.T. by F. Hashemian involves stiffer, 4 meter span beams, but no results are available as of yet.

The following images are from a 5 meter span beam, under reinforced with only one 15 mm steel rebar. The concrete strength is unknown.

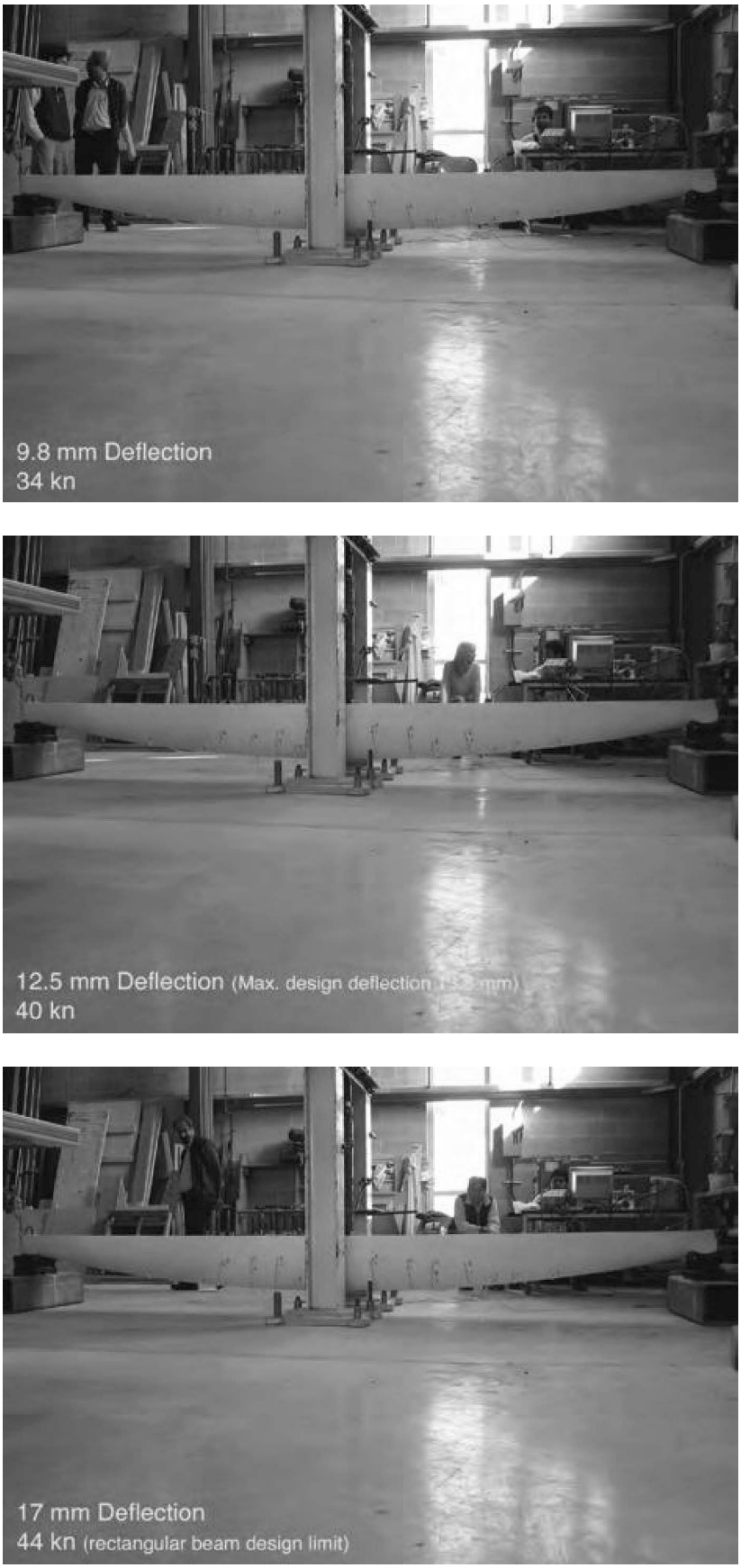


Figure 3.9
Photographs of destructive testing at C.A.S.T, Canada. A fabric formed beam is subjected to a three point bending test.

It would be interesting to compare the performance of this beam to that of a rectangular beam. For this purpose the equivalent moment of inertia I_{eq} is calculated using a rule of thumb for a pointload, Equation (3.2). It is known that the deflection at 40 kN is 12.5 mm, which is $1/400^{th}$ of the span. The modulus of elasticity for cracked concrete E_{cr} is estimated at 30% of full stiffness.

$$\delta = \frac{Fl^3}{48E_{cr}I} = \frac{1}{400}l \tag{3.2}$$

$$I_{eq} = \frac{25}{3}\frac{Fl^2}{E_{cr}} = \frac{25}{3}\frac{40.000 \cdot 5000^2}{30\% \cdot 30.000} = 926 \cdot 10^6 \text{ mm}^4 = \frac{1}{12}b \cdot h^3 \tag{3.3}$$

From this equivalent moment of inertia, a range of possible rectangular cross-sections could be determined using Eqn. (3.3), but first the volume of the fabric formed beam is calculated as a basis for comparison.

As mentioned, no geometric data is available on these beams other than the span. However, using photographs as a reference, with the span in mind, an approximate volume can be estimated. It is estimated that the height at the ends is 100 mm and 520 mm at midspan. A photograph taken from the side allows us to estimate the width to 185 mm. It is assumed that the cross-section A_c consists of a rectangle, and half a circle, Eqn. (3.4), and that the longitudinal shape of the beam h is a parabolic function, Eqn. (3.5) (see Figure 3.10).

$$A_c = b \cdot h_1 + \frac{1}{4}\pi \cdot b \cdot h_2 = \frac{185}{4}\pi \cdot h_2 + 18500 \tag{3.4}$$

$$h_2(x) = C_1x^2 + C_2x + C_3 \tag{3.5}$$

Using boundary conditions, we may solve the unknown coëfficients C_1, C_2 and C_3.

$$h_2(0) = 0 \rightarrow C_3 = 0$$

$$h_2(l/2) = h_{\max} \rightarrow C_1\frac{1}{4}l^2 + C_2\frac{1}{2}l = h_{\max}$$

$$h_2(l) = 0 \rightarrow C_1l^2 + C_2l = 0$$

$$C_1 = -\frac{4h_{\max}}{l^2}$$

$$C_2 = \frac{4h_{\max}}{l}$$

So that,

$$h_2(x) = -\frac{4h_{\max}}{l^2}x^2 + \frac{4h_{\max}}{l}x \tag{3.6}$$

where $h_{\max} = 520\text{-}100 = 420$ mm

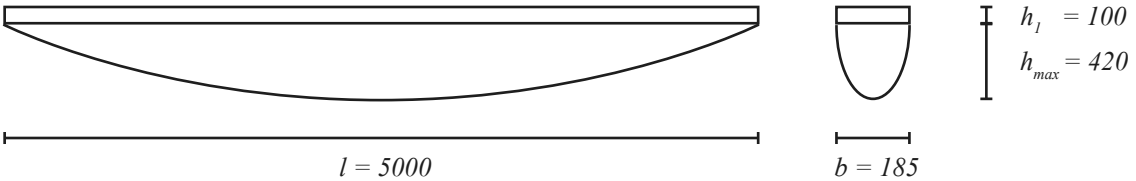


Figure 3.10
Approximated geometry of the fabric formed beam in Figure 3.9, combining photographs and simple geometric math formulas.

Combining Equations (3.4) and (3.6) gives the volume V_c as a function of longitudinal direction x,

$$V_c(x) = \frac{185}{4}\pi\left(-\frac{4h_{\max}}{l^2}x^2 + \frac{4h_{\max}}{l}x\right) + 18500 \tag{3.7}$$

which, integrated from $x=0$ to $x=l$, gives the total estimated volume:

$$\begin{aligned}\int_{x=0}^{l} V_c(x)\,dx &= \left[-\frac{185\pi h_{\max}}{3l^2}x^3 + \frac{185\pi h_{\max}}{2l}x^2 + 18500x\right]_0^l \\ &= -\frac{185\pi h_{\max}}{3l^2}l^3 + \frac{185\pi h_{\max}}{2l}l^2 + 18500l \\ &= \left(\frac{1}{6}\pi h_{\max} + 100\right)\cdot 185\cdot l \\ &= \left(\frac{1}{6}\pi\cdot 420 + 100\right)\cdot 185\cdot 5000 \approx 926\cdot 10^6 \text{ mm}^3\end{aligned} \tag{3.8}$$

It is now possible to compare the volume of various rectangular beams with the equivalent moment of inertia I_{eq} from Eqn. (3.3) with the estimated volume V_c of the fabric formwork from Eqn. (3.8) as shown in Figure 3.11.
These beams differ in ratio between their width b and height h and it is noted that typical design rules suggest that con-

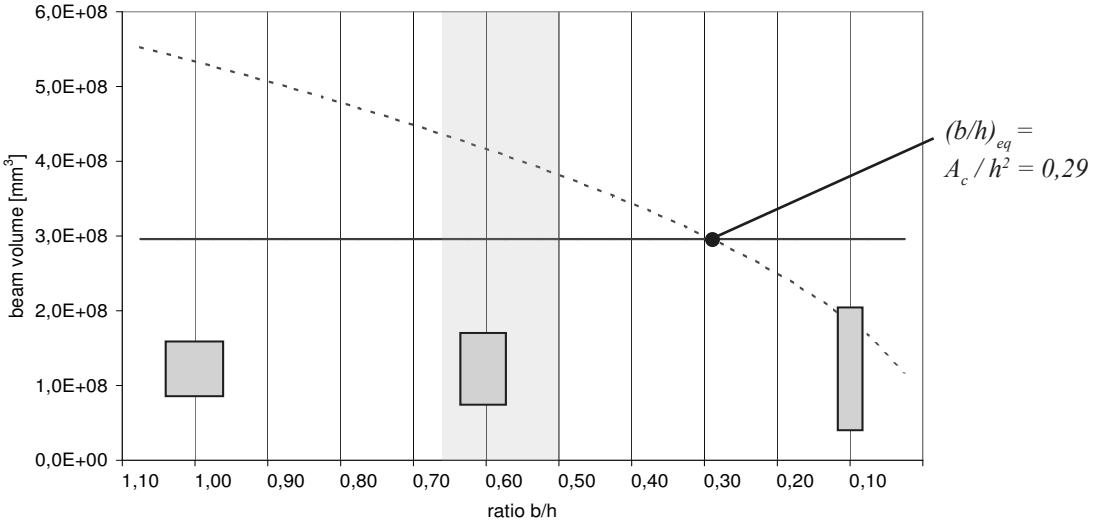


Figure 3.11
Comparison of volume of the fabric formed beam with that of the rectangular beam for various width/height ratios, the stiffness being equal.
The conventional ratio range is shown as a grey area.

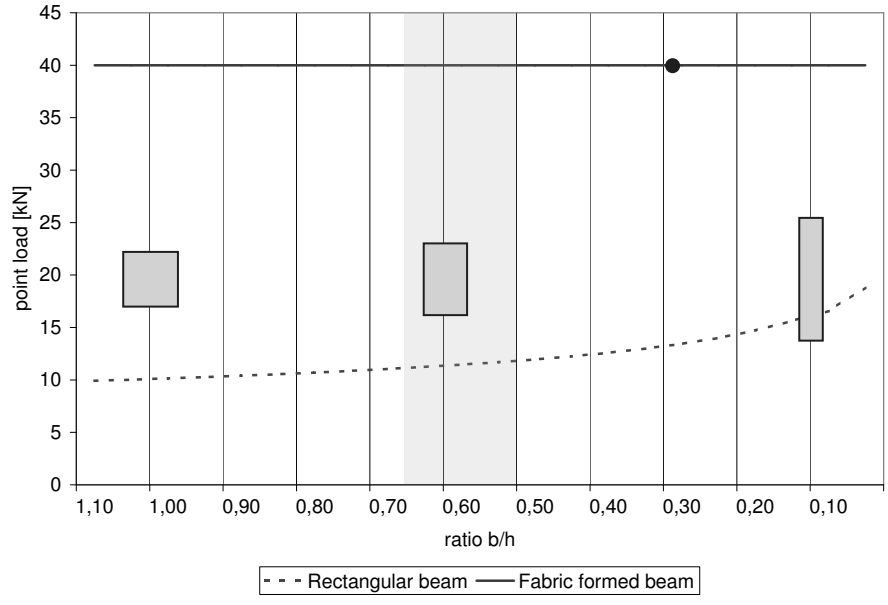
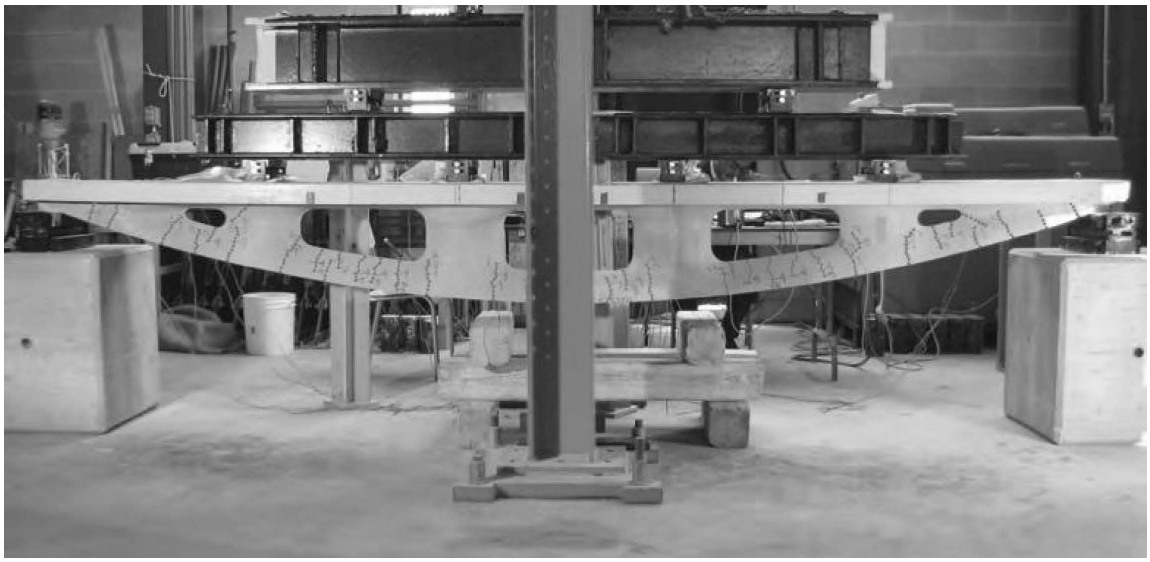
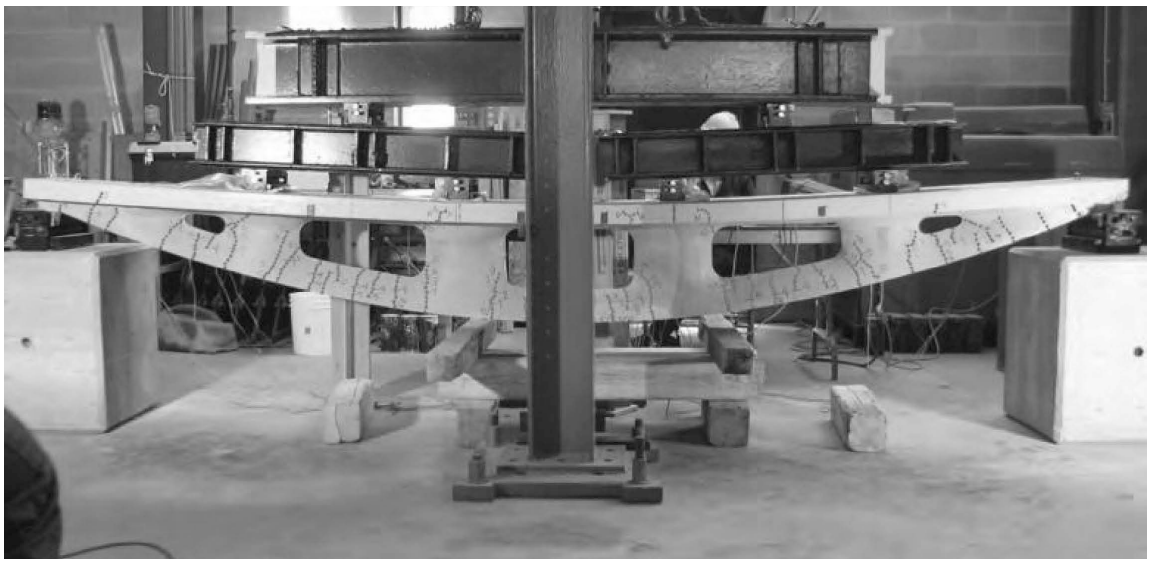


Figure 3.12
Comparison of the load at failure of the fabric formed beam with that of rectangular beams for various width/height ratios, the volume being equal.
The conventional ratio range is shown as a grey area.

crete beams have a ratio b/h of about 1/2 to 2/3.
The point in Figure 3.11 denotes the equivalent b/h ratio and shows that based on these calculations, stiffness is nearly equal between the fabric formed and the rectangular equivalent beam.

A second comparison is made between the same pointload F = 40 kN of the fabric formed to the calculated ultimate pointloads for the same range of rectangular beams, using the same amount of reinforcement (the same cross-sectional area, not total volume) shown in Figure 3.12. These calculations follow the same equations as those used in Chapter 6.4.
The comparison between the fabric formed beam and the rectangular beams quickly reveals that the regular, rectangular beams are less stiff using the same amount of volume and fail far earlier than the fabric formed beam does for the traditional range of width height ratios b/h.
In other words, with respect to material efficiency, this particular fabric formed beam is superior in both the serviceability limit state (deflection), and the ultimate limit state (failure).
However for an equivalent ratio as calculated in Figure 3.11, the deflection is equal and it is noted that M. West from C.A.S.T. frequently states that the tested beams are actually less stiff and 'fail' in deflection/serviceability.
Whatever the case, these findings raise the question whether optimization of these geometries might produce beams that always perform better.
Also, it is recommended that a comparison based on reinforcement ratios should be made as well, to assess their influence on relative stiffness and failure.
A second example of one of the tests yielded actual test data, courtesy of F. Hashemian from C.A.S.T. The test





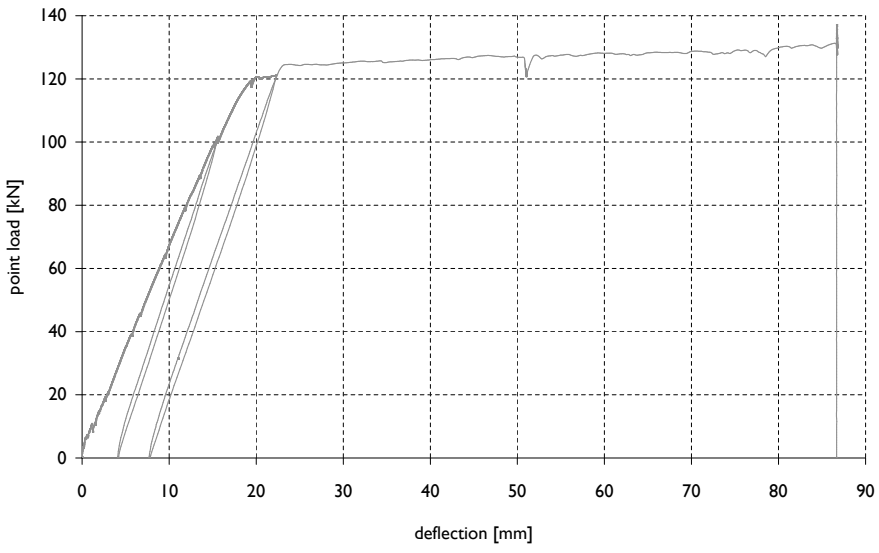
involved a four point bending test on a pinch molded beam. The characteristic concrete compressive strength f'_c = 59 N/mm², derived from testing three cylinders (cylinder diameter = 100 mm), was used for this particular beam. It was reinforced with three 16 mm rebars tied together to form a single curved bar (f_s = 400 N/mm², A_s = 600 mm²).

Using the rule of thumb for a simply supported beam subjected to four point loads (Eqn. 10), we derive the equivalent moment of inertia I_{eq} for this fabric formed beam. It is assumed that the four point loads are equally spaced, that the uncracked modulus of elasticity E is 37.250 N/mm² and that the deflection δ at midspan is 30 mm when each pointload F is 30 kN.

$$\delta = \frac{c \cdot \sigma \cdot l^2}{h \cdot E} = \frac{c \cdot M \cdot l^2}{h \cdot E \cdot W} = \frac{c \cdot 0,6 \cdot F \cdot l^3}{h \cdot E \cdot W} = c \cdot 0,3 \cdot \frac{Fl^3}{EI} \tag{3.10}$$

where $c = 0,21$
$M = 0,6 \cdot F \cdot l$

In this case the dimensions of the beam were more difficult to estimate, but was achieved by tracing the image in AutoCAD, estimating the width from another photograph as well as estimating a reduction factor due to varying cross-sections of the beam. All these estimations do not produce numbers that are meaningful in the quantitave sense, but they did again point out, that an equally stiff rectangular beam requires more concrete and likely failing at a much earlier stage.



3.7 Conclusions on fabric formed beams

It is clear that the fabric molds for structural beams offer a new type of geometry, a geometry that might have been feasible before, but would not have been considered for various practical and economic reasons. The simplicity of the method allows double curved, complex shapes – previously associated with high energy, computer controlled manufacturing methods – at relatively low cost.
The range of possibilities that fabric formwork technology offers for structural elements might result in new design and engineering ethics. Newly shaped beams might prove to use their material in a more efficient way and may let the freeform nature of concrete live up to its potential on a scale, where it is now generally associated with rectangular form. The application of fabric formwork to structural beams is still in its infancy, as evidenced by the fact that all beams featured in this chapter are prototypes and that structural testing has only recently started.

Fabric formwork technology offers both advantages and disadvantages, and those found in the sources cited throughout this chapter are summarized in the table below.

TABLE 3.3 Advantages and disadvantages of fabric formwork technology

Advantages	Disadvantages
The potential geometries are: • Complex e.g. double curved • Structurally efficient • Esthetically unusual and pleasing • Economically efficient (material savings of concrete and rebar) The mold is relatively: • Simple in nature • Inexpensive • Lightweight • Reusable Material savings offer designs where flexibility is used to its advantage Potential for reduction and simplification of reinforcement requirements	Geometric predictability is difficult due to the non-linear geometric behavior of fabrics The complexity requires more structural analysis Material savings might have detrimental effect on: • Decrease in redundancies i.e. safety
Minor advantages	**Minor disadvantages**
The mold consists of widely available materials Improved surface quality of the casted concrete	Geometric accuracy and consistency are relatively difficult to maintain: • Relaxation will occur due to prestress forces in the membrane • Creep can occur due to an increase in temperature as a result of the hydration process of the concrete hardening process Material savings might have detrimental effect on: • Decrease in redundancies i.e. safety

CHAPTER 4 | Differential Evolution

Previous chapters have introduced the concepts of structural optimization as well as fabric formwork technology as a cost-effective manufacturing method for complex shapes. The general objective of this thesis is to bridge the gap between structural optimization and manufacturing. The shapes given by optimization algorithms such as BESO cannot be made in a cost-effective manner at this moment. To do so, one can look towards fabric formwork, but the possibilities this method offers come within specific geometric constraints due to the fabric involved. For this reason, the use of BESO is abandoned in favor of a more general purpose optimization algorithm. Differential Evolution (DE) is such an algorithm and allows, contrary to BESO, the flexibility of setting up a program that can include the form-finding process for fabrics in addition to multiple objective criteria including more than just stresses or energy.

4.1 Introducing Differential Evolution

The DE algorithm is fairly recent compared to other more well known alternatives. In fact, as of yet, no more than two books and a handful of papers have been published on the subject. Application of DE is not yet widespread, so therefore, it is highly likely that this thesis presents its first use in the field of civil engineering.

Storn & Price (1995) first proposed this new method for minimizing non-linear continuous space functions. By applying it to several benchmark problems, it was shown that DE did not necessarily but often did converge faster than two other methods, Annealed Nelder & Mead (ANM) and Adaptive Simulated Annealing (ASA), and was the only method of the three to converge to an optimum in each case.
Lampinen and Zelinka (1999) compared DE to several other algorithms in solving mixed variable (i.e. both discrete and continuous variables) non-linear problems. They concluded that DE presented a superior and relatively simple alternative to algorithms such as Simulated Annealing (SA) and Genetic Algorithms (GA) among others (see Appendix A).
However, it should be noted that some bias may be expected in these two papers, since together, Storn, Price and Lampinen literally wrote the book on the subject (Price et al., 2005). On a side note, this book also comes with a CD-ROM with open source code in various languages.
Recently, Feoktistov (2007) also published what is the second book on DE.
For a more comprehensive overview of the history of DE, the reader is referred to the first book, but may also find more information on Wikipedia.
One may conclude from all these sources that 1997 was the breakout year for DE, but none of them indicate why it never gained more momentum after that, especially after purportedly being superior to the widely used and older GA. One explanation may be given upon closer inspection and comparison between DE and GA. This will show that most features of both algorithms are interchangeable and that the only real, though still important, difference is the mutation scheme (as explained in Section 4.3.4 in detail).

The term 'differential' in DE refers to its mutation scheme; its basic premise is to regard solutions as vectors and use their vector difference to create new trial solutions. Through continuous generation and evaluation of these trial solutions an optimal solution is evolved.
The 'evolution' references the origin of DE which lie in Genetic Annealing (GAn), a derivative of GA and SA. GA uses an analogy with natural evolution to search for optimal solutions. Because this analogy with nature provides such a clear and vivid explanation of GA and GA in turn led to DE, the principles behind the former are explained before going into the mathematics behind the latter.

4.2 Principles behind evolutionary algorithms

As mentioned, evolutionary algorithms such as GA use concepts from biological evolution to create a mathematical method of optimization. The idea is that the guiding mechanisms in nature produce more optimal solutions and translated to mathematics can do the same for numerical problems.

The Oxford dictionary defines evolution as *the process by which different kinds of living organisms are believed to have developed, especially by natural selection*, while Wikipedia describes it as *the process of change in the inherited traits of a population of organisms from one generation to the next*.

Literature on evolutionary algorithms such as GA and DE frequently uses the same terminology as in evolutionary biology. The concept of evolution is summarized (Figure 4.2) in these terms before parallels with the algorithms are drawn in Section 4.3.

A species of living organisms, or **individuals**, evolve over time as each **generation** uses **reproduction** to create new generations. The total number of individuals at one point in time is called the **population**, and because in evolutionary algorithms there is only one generation at a time, the population size is equal to the generation size.

Each individual with its unique properties is determined by its DNA. **Chromosomes** are organized structures of DNA, consisting of regions called **genes** which correspond to units of inheritance i.e. inheritable traits. Possible variations of these genes are called alleles. The make up of these alleles generate a specific individual genotype. When translating this genotype, or genetic composition, to observable qualities of an individual the term phenotype is used. Those individuals with superior qualities, or phenotype, have a higher likelihood of contributing to the genes of new generations through **natural selection**.

To create new individuals, or **children**, for a new generation, the properties of selected, current individuals, or **parents** are combined through **recombination**, or **crossover**. Some genes are changed randomly through **mutation**, which guarantees a certain amount of variation in new generations.

These principles can be translated to mathematical concepts. Consider each population, each generation of individuals as a collection of vectors. Each vector can be equated to a chromosome as they determine the properties of an individual. The dimensions, or components of the vector correspond to the genes of this chromosome. Using crossover, several components of two parent vectors combine to form new vectors, while mutation will alter specific components.
Each new vector or new set of vectors is subject to some fitness, or cost function which evaluates and assigns a fitness, or value to this vector. Using some selection, or evaluation scheme the original vectors and the new ones are compared and selected to form the new generation.

Eventually, GA proves to find optimal or near-optimal solution for many problems. There are also many ways to determine how crossover, mutation and selection take place, each way affecting the behavior of the algorithm to better suit certain types of problems. It is interesting to note that although GA has proven to work in many instances, there are different explanations and hypotheses on why they perform so well.

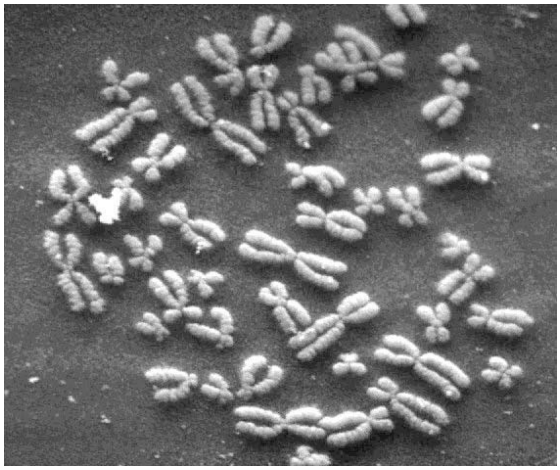


Figure 4.1
Photographed human chromosomes in a hypotonic solution

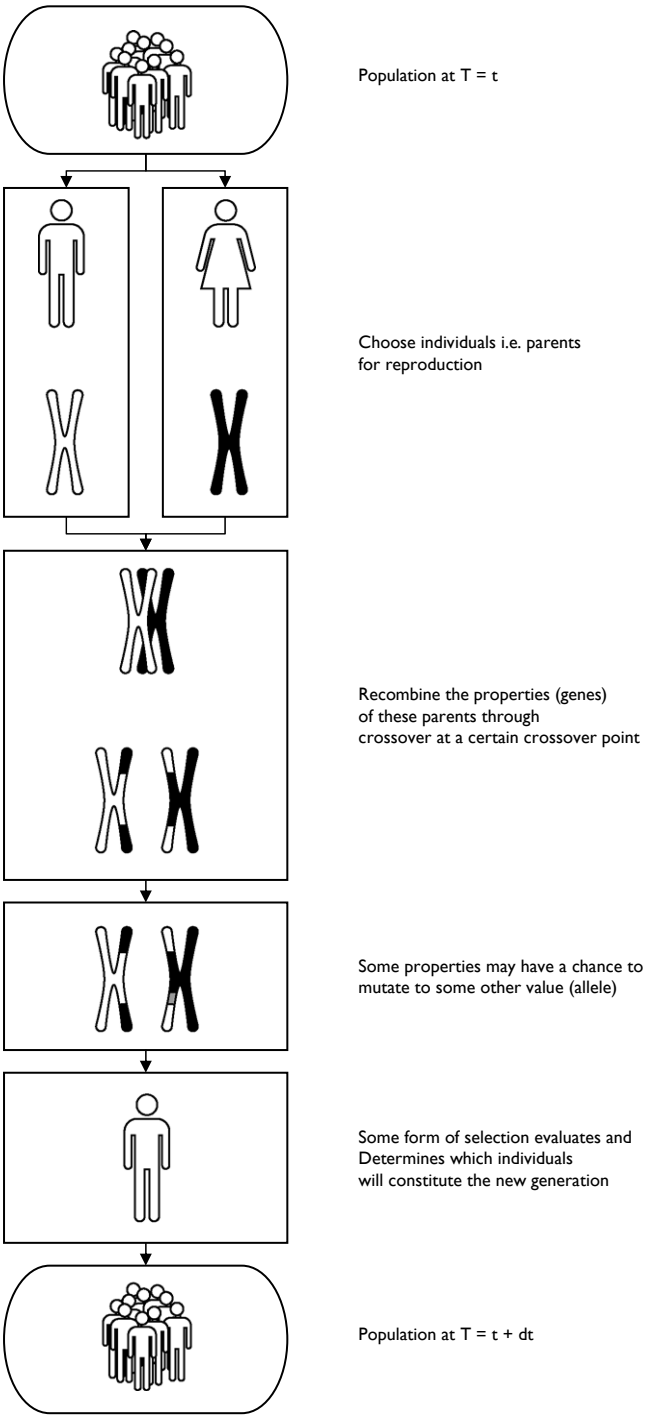


Figure 4.2
Flow chart generally showing one evolutionary iteration including cross-over and mutation

4.3 Concepts and mathematics in Differential Evolution

This paragraph will further explain the various aspects of DE. As mentioned, most of the features are applicable to GA except for the mutation scheme. Note that Price et al. (2005), in their seminal book on DE, confusingly use the term selection for both the method in which DE generates trial vectors - as part of the reproduction along with mutation and crossover - as well as how the new trial vectors are evaluated and compared with their corresponding parent, or target vector. The flowchart on the next page (Figure 4.4) offers a quick overview of how DE operates.

4.3.1 Population structure

The total number of vectors is equal to the population size. Each vector may also be called a population member. Population P has Np population members. Each population member x_i has a number of properties that have to be optimized. The total number of these properties, or dimensions, is D (Eqn. 4.1, Figure 4.3). Optimization of these dimensions takes place during an unknown number of generations g_{max}.

$$P_{x,g} = \left(\mathbf{x}_{i,g}\right), \quad i = 0,1,...,Np-1, g = 0,1,...g_{\max}$$
$$\mathbf{x}_{i,g} = \left(x_{j,i,g}\right), \quad j = 0,1,...,D-1 \tag{4.1}$$

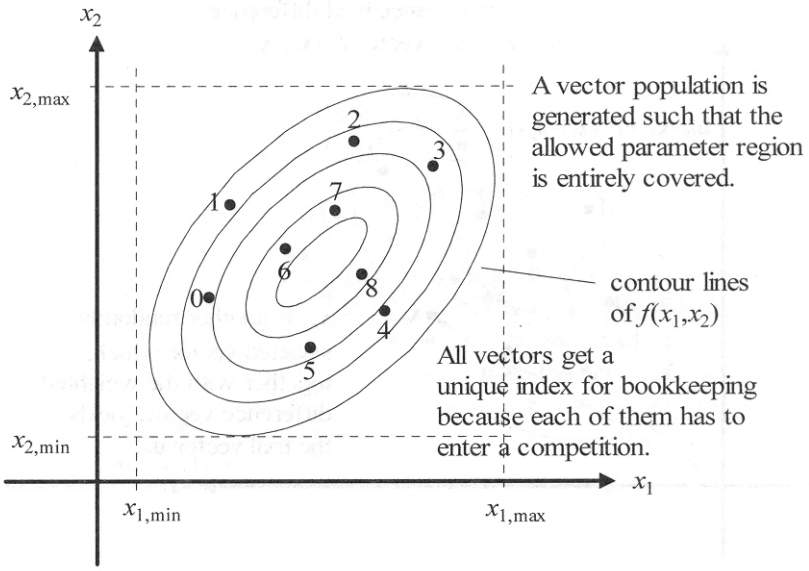


Figure 4.3
Initialized population in 2D space
Price et al. (2005)

The larger the number of dimensions D is, the more difficult this optimization problem will prove to be. A larger population size Np on the other hand, will facilitate a higher probability of fast convergence towards an optimal solution.

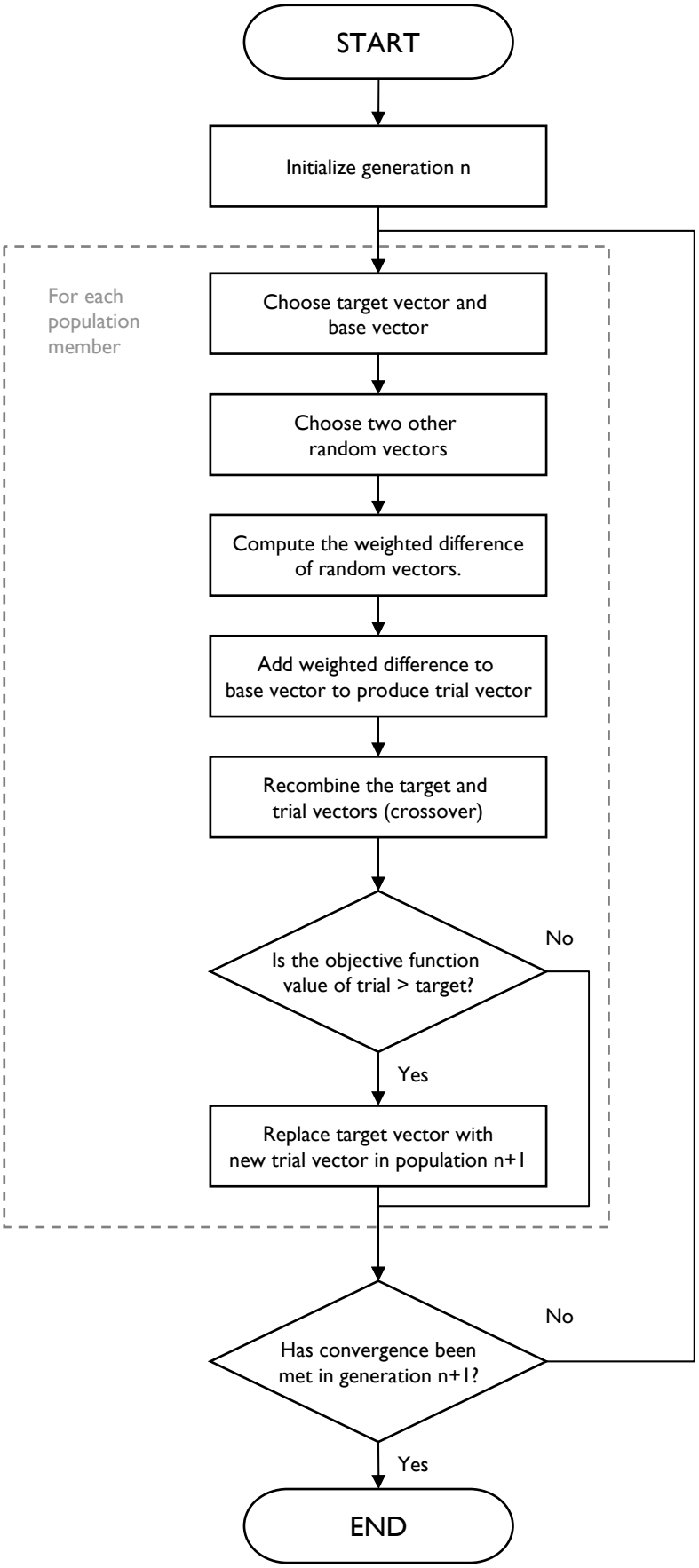


Figure 4.4
Flow chart describing the process of DE

4.3.2 Initialization

The first population can be initialized by any uniform or non-uniform distribution, depending on what is known in advance about the (location of) optima. Price et al. (2005) suggest a random or Halton point distribution as a uniform distribution. They show that, generally speaking, neither is superior to the other. For non-uniform distributions only a Gaussian distributed population is briefly discussed. For several standard benchmark problems, the Gaussian distribution proves equally or less effective than a uniform distribution, so the nature of the problem will determine whether it is appropriate. Since no clear advice is given on which distribution is appropriate to the problem of this thesis, the uniform random distribution was chosen. In this case each vector component j is chosen randomly between its feasible lower and upper bounds, U and L (Eqn. 4.2).

$$x_{j,i,0} = \text{rand}_j(0,1) \cdot \left(b_{j,\text{U}} - b_{j,\text{L}}\right) + b_{j,\text{L}} \tag{4.2}$$

4.3.3 Selection

For each population member, or target vector, three other vectors are selected to calculate a new recombined trial vector against which the target vector is compared. In evolutionary terms, one might say that a child (trial) of three parents is compared to one of the parents (target). It is clear that the analogy with biological evolution has become tenuous in DE since no such situation is known to occur in nature.
The three other vectors are called the base vector x_{r0} and the random vectors x_{r1} and x_{r2}. Usually the base vector is chosen randomly, using a *rand(0,1)* function, but Price et al. (2005) offer two additional methods, called permutation selection and random offset selection (Figure 4.5). These methods ensure that each vector in the population will serve as a base vector exactly once. Comparison shows all three selection methods to perform similarly, though the latter helps convergence when the population size Np is extremely small. For the purposes of this thesis the standard random selection method was implemented.

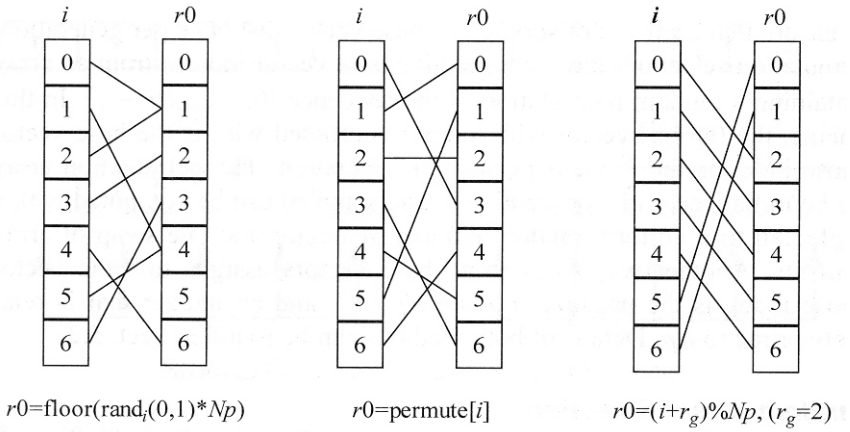


Figure 4.5
Three ways of pairing a base and target vector; random method, permutation selection and random offset selection
Price et al. (2005)

Sources on DE also offer biased base vector selection where instead of random selection, the best performing vector so far is always chosen. Another method generates a vector that lies between the target vector and the best-so-far vector. At this point typical nomenclature in DE is introduced that respectively indicate each of these types of selection.

`DE/RAND`
`DE/BEST`
`DE/TARGET-TO-BEST`, or `DE/RAND-TO-BEST`

For this thesis `DE/RAND` was chosen i..e. the remaining two random vectors are randomly chosen from the population (Figure 4.6). Price et al. (2005) discuss in detail how the performance of DE is affected by so called degenerate vector combinations i.e. when the random vectors coincide with each other, the target and/or the base vector. Without expand-

ing on this subject, it is sufficient to say that for this thesis mutually exclusive indices were implemented i.e. all four vectors are different population members. In this case, DE is said to achieve both good convergence speed and probability with a small population.

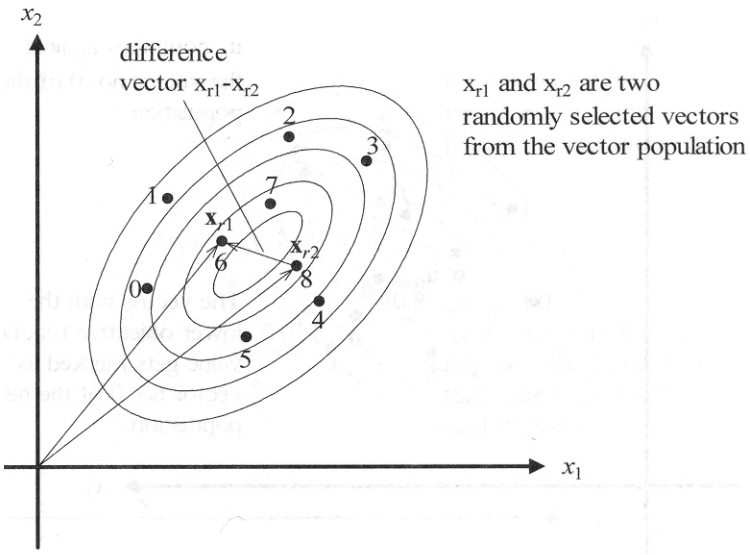


Figure 4.6
Difference vector in 2D space is generated by two random population vectors
Price et al. (2005)

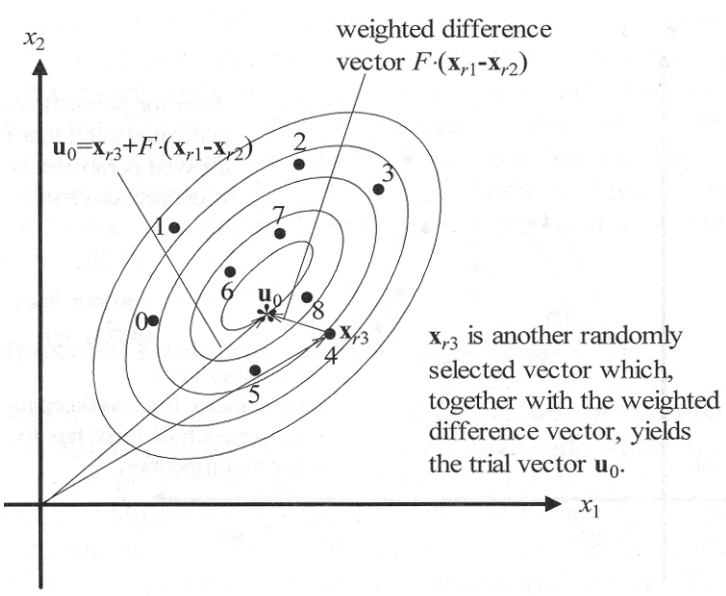
Figure 4.7
Similar to Figure 4.8, only within the context of the entire population
(Bottom left)
Price et al. (2005)

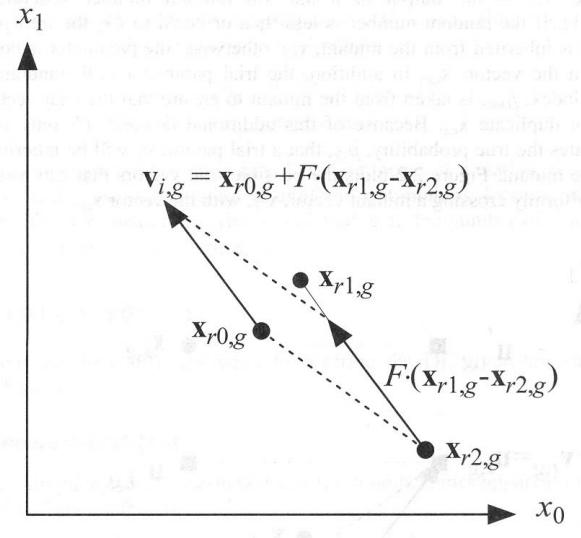
Figure 4.8
Differential mutation; the weighted differential is added to the base vector to produce the mutant vector
(Bottom right)
Price et al. (2005)

4.3.4 Mutation

The vector differences of the random vectors are added to the base vector. The effect of this differential strategy is analogous to mutation in GA, where each vector component has a certain probability of being randomly altered. The purpose of mutation is to avoid convergence towards local optima and create a certain variance within the population.
DE uses a mutation scale factor F where $F = (0,1)$. The value of 1 is an empirically derived upper limit. Lower limits exist, but depend on the population size and the value of other parameters. The scale factor F gives the weighted difference between the random vectors, which added to the base vector gives the mutant vector v (Eqn. 4.3, Figures 4.7-4.8).

$$\mathbf{v}_{i,g} = \mathbf{x}_{r0,g} + F \cdot \left(\mathbf{x}_{r1,g} \cdot \mathbf{x}_{r2,g} \right) \tag{4.3}$$





Initially, this thesis used a constant factor F, where several values were used (see Section 4.4 and Chapter 8). It is also possible to generate a new value for F for each component, called jittering, or for each vector, called dithering. Both these methods can use various distribution types. Because jittering scales each component seperately it is able to rotate the vectors as well. At one point, due to unsatisfactory results, the choice was made to implement jittering. Using a probability distribution function based on the power law, jitter was introduced into the algorithm. In this case, the scale factor F is continually generated by the following function:

$$F_j = pow\left(rand_j(0,1), q\right) = rand_j(0,1)^q, \quad q = \frac{1}{F} - 1 \tag{4.4}$$

Although jittering was implemented, the computational demands of `FABRICFORMER` did not allow multiple evaluations of the program with both types of scaling. It is therefore not known which strategy is best suited for the type of problem `FABRICFORMER` presents.

4.3.5 Recombination

Discrete recombination, better known as crossover, is commonly used in evolutionary algorithms. Typically a crossover point is chosen, a vector index C. Two vectors are recombined to form a new one. Vector components 0 to $C\text{-}1$ are taken from one vector and supplemented with components C to $D\text{-}1$ from the other. This method is called one-point crossover. Some newer algorithms use N-point crossover, using multiple points at which components are then taken from the other vector. Others may use uniform crossover where for each component it is seperately determined from which vector it is taken. This uniform crossover can use any type of probability distribution function to do so.
DE typically uses so called uniform binomial crossover as was used for this thesis. In this case at least one component is crossed with index j_{rand}. For every other component the crossover factor Cr determines the probability that crossover occurs (Eqn. 4.5, Figure 4.9)

$$\mathbf{u}_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} & \text{if } \left(rand_j(0,1) \le Cr \text{ or } j = j_{rand}\right) \\ x_{j,i,g} & \text{otherwise} \end{cases} \tag{4.5}$$

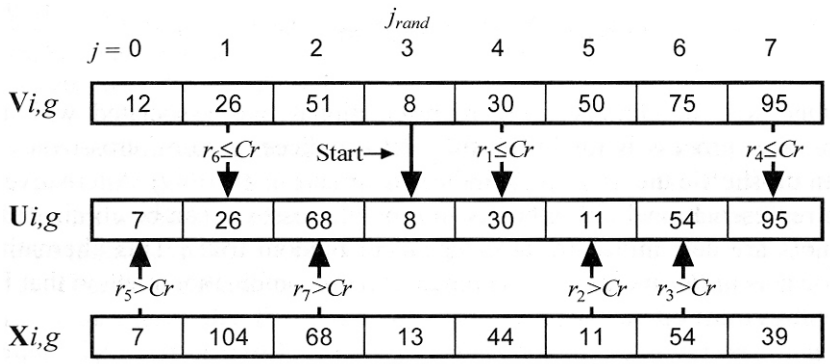


Figure 4.9
Uniform binomial crossover for vector with dimension $D = 8$
Price et al. (2005)

It should be clear that a value of $Cr \sim 0$ produces minimum disruption because few mutant components $\mathbf{v}_{i,g}$ are crossed, whereas $Cr \sim 1$ favors a high degree of components from $\mathbf{v}_{i,g}$ for the new trial vector $\mathbf{u}_{i,g}$. Most GAs recommend using $Cr = 1/D$ because change is minimal (on average, one component will cross) and the algorithm is relatively more stable (Price et al., 2005). Specific to DE, Storn & Price (1997) show, based on various test problems, that either $0 \le Cr \le 0.2$ or $0.9 \le Cr \le 1$ give the best performance.
During this thesis, values for Cr in both ranges were used. Both ranges performed, but it seems that lower range values proved to be more succesful and reliable. Near the end, most runs were done with $Cr = 2/D$. The last runs for the pinch mold geometries were using $Cr = 0.9$ due to the dependancy of the components.

4.3.6 Evaluation

There are various ways of selecting individuals to form a new generation and to reproduce. Price et al. (2005) discuss these various selection types. DE itself uses one-to-one survivor selection. In this case, each parent (target vector x) with index i is compared to the child (trial vector u) of the same index (Eqn. 4.6). The worst performing vector is discarded. The advantage is that, contrary to some other methods, the best solution so far is always kept and no solution worse than the worst-so-far is ever chosen. On the downside, it is possible that a trial vector that is better than most of the current population will be rejected if its target is even better. It became clear that this in fact happens very often in DE.

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g} & \text{if } f\left(\mathbf{u}_{i,g}\right) \leq f\left(\mathbf{x}_{i,g}\right) \\ \mathbf{x}_{i,g} & \text{otherwise} \end{cases} \tag{4.6}$$

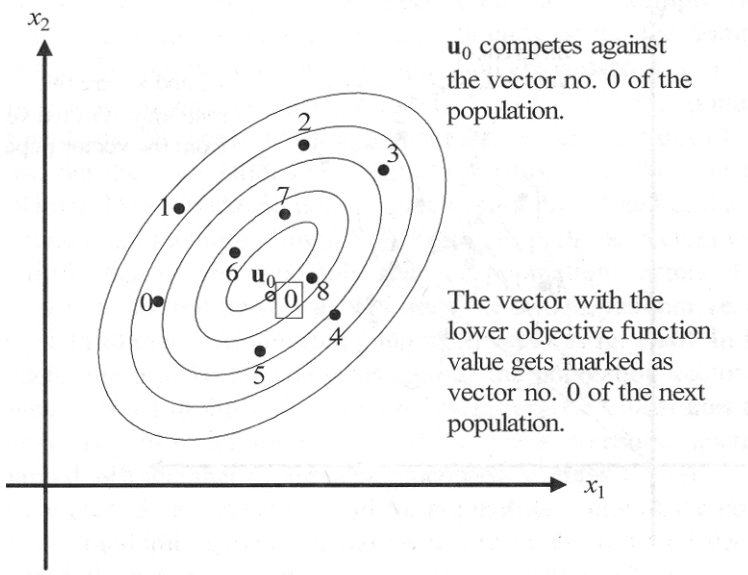


Figure 4.10
Selection of the new vector 0 through evaluation of the objective fuction i.e. cost or fitness function
Price et al. (2005)

4.3.7 Termination

There are various ways of specifying how the DE algorithm should stop, most obvious when improvement of the performance i.e. the fitness or cost becomes minimal. However, because `FABRICFORMER` is a computationally demanding program, no termination criteria were defined, and progress and termination of the algorithm was done manually based on continuous output.

4.4 Implementing Differential Evolution

The previous paragraph already indicated some choices that were made in implementing DE. In general choosing between various options of DE was difficult since most advice by various authors was based on empirical evidence. These observations often lacked any conclusive explanations on why DE behaved as it did. Several mathematical problems that were used for benchmarking in the various sources also did not provide any frame of reference, since this author was largely unfamiliar with them. The open source code CD-ROM provided by Price et al. (2005) with their first book did not include Java, but still might have been useable. However, it was decided to use only the theory and pseudo-code within the book to gain a better understanding of the algorithm since no previous personal experience with evolutionary algorithms existed.

This thesis used `DE/RAND/1/BIN`, or classic DE, where `RAND` refers to the selection of base vector, `1` refers to the number of differentials used to create the mutant vectors and `BIN` to the type of crossover. Price et al. (2005) recommend classic DE for situations where parameter dependence is low i.e. the value of one parameter is not directly influenced or determined by the value of another. A crossover factor Cr of 0.2 is standard, but for limited parameter dependance 0.9-0.95 can be used. In the former case scale factor F should be 0.3 to 0.5 and in the latter $F \geq 0.8$. The population size $Np=50$ was adequate for four different $D=30$ problems. In some cases a low end $Np = 5 \cdot D \cdot Cr$ is suggested.

If function evaluations are very time consuming `DE/BEST/1/BIN` with a small amount of jitter is recommended.

The following table shows a few succesful runs of `FABRICFORMER` and their respective DE parameters.

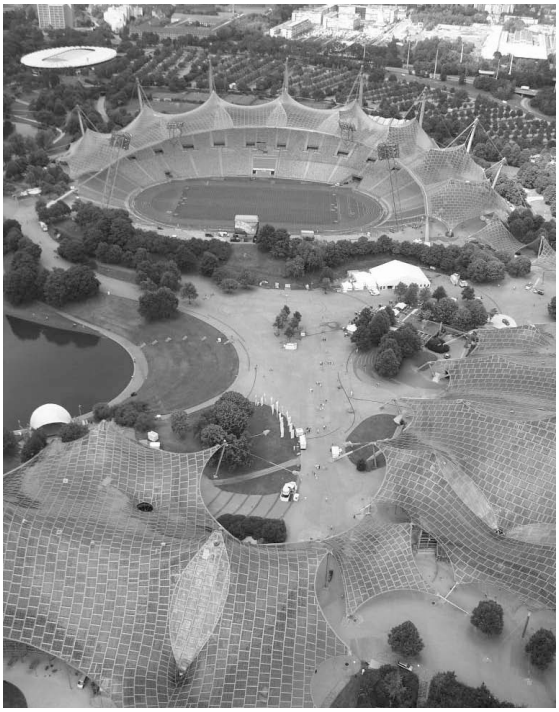
TABLE 4.1 DE parameters for FabricFormer run

Run no. 9		
crossover	Cr	0.2
scale factor	F	0.5
population size	N_p	10
dimension	D	12
Run no. 26		
crossover	Cr	0.9
scale factor	F	0.2
population size	N_p	25
dimension	D	12
Run no. 27		
crossover	Cr	0.9
scale factor	F	0.7
population size	N_p	50
dimension	D	12
Run no. 28		
crossover	Cr	2/D = 0.04
scale factor	F	0.3
population size	N_p	50
dimension	D	12
Run no. 50+		
crossover	Cr	2/D = 0.04
scale factor	F	0.3 (jitter)
population size	N_p	50
dimension	D	40

For the results of these runs, the reader is referred to Chapter 8. The experience during this thesis was that high values of crossover led to early convergence, but possibly local optima. There had been several bugs in early versions of FABRICFORMER that interacted with DE in such a way that is unclear whether the sub-optimal results were due to a high value of Cr or these specific bugs. In any case, after this period the crossover factor was changed to and kept within the low range values.
The scaling factor F was usually chosen to reflect the suggestions by Price et al. (2005) except for Run no. 26, as an experiment due to unsatisfactory results in previous runs.
The population size had been kept small due to the computational requirement of FABRICFORMER, but this proved to work adversely as the variance of some parameters was too small. After Run no. 27 the population size was kept at $Np = 50$.
The final runs, which included pinch molding, had a large number of components D. At this point, no large changes were made to the DE parameters except for introducing some jitter. If proven unsuccesful, future runs should use DE/BEST/1/BIN instead of DE/RAND/1/BIN and could perhaps include larger populations.

4.5 Conclusions

The DE optimization algorithm was shown to be easy to grasp and quick to implement. Finding appropriate parameters remains difficult since their individual effect is clear but their overall influence on convergence and optimization is less obvious.



CHAPTER 5 Dynamic Relaxation and membrane form finding

The Olympic Park in Münich, shown on the left, is one of many well-known structures designed by German architect Frei Otto. As the bottom photograph shows, he made use of scale models to derive the geometry of his designs. Physical modelling, such as this, is a relatively simple and elegant method of designing complex structures, such as double curved tension structures or shells. Although architects such as Frei Otto and Antonin Gaudi have proven the worth of physical models, recent decades have seen the rise of various computational procedures to calculate the geometry of such complex shapes, but more importantly the prestressing forces. This thesis project has used a method called Dynamic Relaxation to define the geometry of fabric molds. In this chapter this computational method will be explained in detail, and the example of the Münich Olympic Park will only serve as a reminder that algorithms such as these do not monopolize the design and analysis of complex shapes.

5.1 Computational form-finding of membranes

The shape of tension structures is not know in advance because it depends on the interaction of boundary conditions, the prestressing and the (non-linear) material properties. Most engineering firms use computer software, be it either widely available software packages or their own in-house programs. Each of these programs typically relies on one of the following algorithms to solve the problem of tension structures, although other methods exist;

1) Force Density or Surface Stress Density Method
2) Dynamic Relaxation
3) Transient Stiffness
4) Updated reference strategy (Wüchner & Bletzinger, 2005)
5) Conjugate Gradient

Though the latter two methods have been applied to tension structures in academic circles, no commercial software was found that employed them. Lewis (2003) devotes one chapter to the comparison of Dynamic Relaxation with Transient Stiffness and concludes that while the latter is faster at solving single iterations and simple structures, Dynamic Relaxation is overall faster at solving complex shapes.
When looking at Force Density and Dynamic Relaxation, one can see a clear geographic division in both software companies and universities that use them. Examples of software programs using Force Density are the German program FORMFINDER, German EASY, Italian FORTEN 2000 and Singaporese WINFABRIC. Examples that use Dynamic Relaxation are British INTENS, British OASYS GSA and New Zealand SURFACE. A fairly comprehensive overview of these and other programs can be found at http://www.bruno.postle.net/links/tents/software/.
For the purposes of this thesis, the choice was made to use Dynamic Relaxation based on the following arguments. First, relatively more sources were found explaining the principles of Dynamic Relaxation. Second, Lewis suggests that Dynamic Relaxation was very efficient at solving triangular meshes. This last point is now irrelevant for the following reason. Originally an unstructured meshing of the complex fabric molds was envisioned to cope with the constantly varying geometries FABRICFORMER would have to generate. Unstructured meshing algorithms such as Advancing Front Technique or Delauney Triangulation only work with triangular meshes. In the end, a structured triangular mesh was deemed adequate to mesh the fabric. In fact, an unstructured mesh was found to be unfavorable when translating warp and weft behavior of fabrics to a triangular mesh. Doing so requires that all elements have some reference to the actual warp or weft direction to ensure their physical behaviour is accurate. This would require a secondary structured background mesh, or would result in inaccurate geometries. In conclusion, only the first argument – more sources on the subject – remains valid.

5.2 Introducing Dynamic Relaxation

The method of Dynamic Relaxation (DR) is a pseudo-dynamic process in time which is used to solve static problems. In other words, one makes an analogy between the static solution of a given problem and the equilibrium state of damped structural motion. The algorithm was originally devised by A.S. Day and soon found applications for various structural problems, which included highly non-linear aspects. It was A.S. Day himself who found the earliest use for DR in tension structures as he applied it to analysis of hanging roofs (Barnes, 1998). A discretized shape, i.e. a finite element mesh, is put into motion by translating some external load P to an acceleration of the loaded nodes. This results in motion of the nodes, an oscillation (Figure 5.1), which will eventually reach static equilibrium due to (viscous) damping of the movements (Figure 5.2). The solution that has been reached is one of minimum potential energy. An equilibrium has been reached between the external loads and the internal elastic strains. In the case of tension structures, this solution is analogous to a feasible, stable minimal surface. It is noted that the term 'stable minimal surfaces' refers to surfaces of mean zero curvature, as it is also, confusingly, used sometimes to denote soap-film surfaces which also have minimum surface area.

The DR method was improved upon by Cundall for the analysis of rock mechanics for reasons of convergence (Barnes, 1998). He introduced 'kinetic damping', a fictitious manner in which the simulated motion would converge to a solution, replacing the then current notion of viscous damping. In this procedure, no actual damping occurs, but the kinetic energy of the whole system is continuously checked for peaks. Whenever such a peak in kinetic energy is detected, all motion is arrested (velocities are set to zero) and the algorithm is restarted at that geometry (Figure 5.3). In general, the subsequent peaks decrease in value as finally equilibrium is achieved. In this way, no damping factors have to be specified, no traditional oscillation takes place and convergence is more rapid.

Using the example of a swinging pendulum (Figure 5.4), one could imagine a pendulum slowly coming to a stop under the dampening influence of air resistance. With kinetic damping, the problem would be resolved more quickly as it swings along the lowest point where it has highest velocity, thus maximum kinetic energy. It remains an iterative process nonetheless, due to the usual case problems being more complex and multi-modal in comparison to the simple pendulum.

The method of kinetic damping was used for this thesis, and the reader is referred to Lewis (2003) or Barnes (1998) for more information on viscous damping.

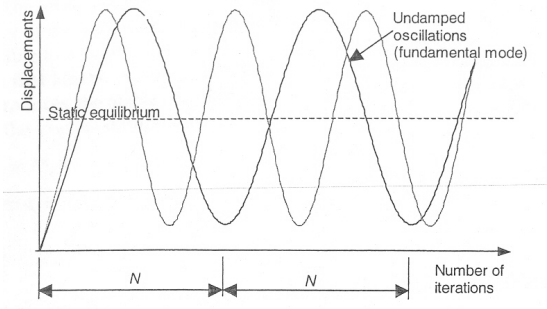


Figure 5.1
Undamped structural motion in time **(Top left)**
Barnes (1998)

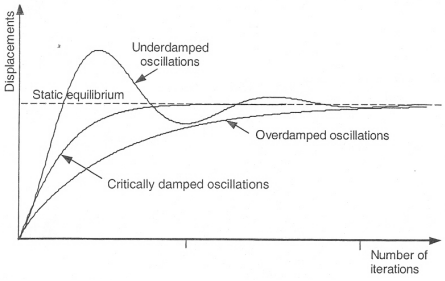


Figure 5.2
Three types of damped structural motion over time **(Top right)**
Barnes (1998)

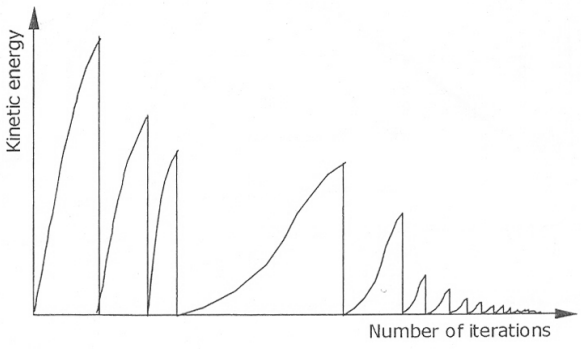


Figure 5.3
Kinetic energy during dynamic relaxation using kinetic damping. The kinetic energy is set to zero when a maximum occurs.
Barnes (1998)

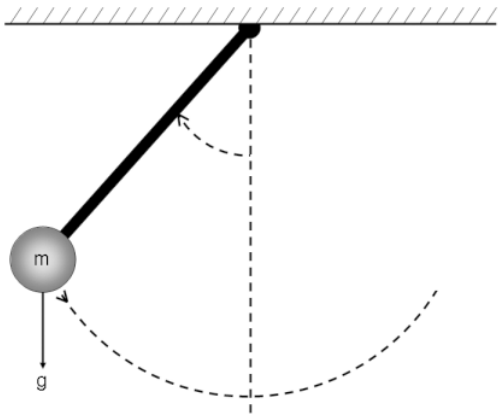


Figure 5.4
A swinging pendulum

5.3 The mathematics behind Dynamic Relaxation

As DR deals with structural motion, the starting point of the algorithm is Newton's Second Law of Motion for constant mass (i.e. the acceleration of an object is proportional to the force applied, and inversely proportional to the mass of the object):

$$R_{ij}^{t} = M_i \cdot A_{ij}^{t} \tag{5.1}$$

which can be expressed in central difference form

$$R_{ij}^{t} = M_i \cdot \left(V_{ij}^{(t+\Delta t/2)} - V_{ij}^{(t-\Delta t/2)} \right) \Big/ \Delta t \tag{5.2}$$

giving the recurrence relation for nodal velocities

$$V_{ij}^{(t+\Delta t/2)} = V_{ij}^{(t-\Delta t/2)} + \frac{\Delta t}{M_i} \cdot R_{ij}^{t} \tag{5.3}$$

in turn providing the motion of any node i in Cartesian direction j *(x,y,z)* at time t.

$$\delta_{ij}^{(t+\Delta t)} = \delta_{ij}^{t} + \Delta t \cdot V_{ij}^{(t+\Delta t/2)} \tag{5.4}$$

where

R = residual force
M = lumped nodal mass
A = acceleration
V = velocity
δ = displacement
Δt = time increment

The lumped nodal masses M are a function of the elastic and geometric stiffness of the connecting elements. This will be discussed in more detail further on. The time increment Δt is no more than a stepsize for the algorithm. It provides a means to guarantee numerical stability and determines the speed at which the algorithm attempts to find a solution. The remaining unknown in the above set of equations is the residual force vector R . This vector for node i for each direction j is the sum of the externally applied loads F and the internal member loads R due to tension stiffening.

$$R_{ij}^{(t+\Delta t)} = F_{ij} + \sum_{i,m=0}^{m=n} \Delta R_{ij,m}^{(t+\Delta t)} \tag{5.5}$$

where the summation of residual forces consists of:

$$\Delta R_{ij,m}^{(t+\Delta t)} = \frac{T_m^{(t+\Delta t)}}{L_m^{(t+\Delta t)}} \left(\delta_k^{(t+\Delta t)} - \delta_i^{(t+\Delta t)} \right) \tag{5.6}$$

for each element m. The tension coëfficient T is determined by the in-plane stiffness, the strain and any initial prestressing.

$$T_m^{(t+\Delta t)} = \frac{EA_m}{L_m^0} \left(L_m^{(t+\Delta t)} - L_m^0 \right) + T_m^0 \tag{5.7}$$

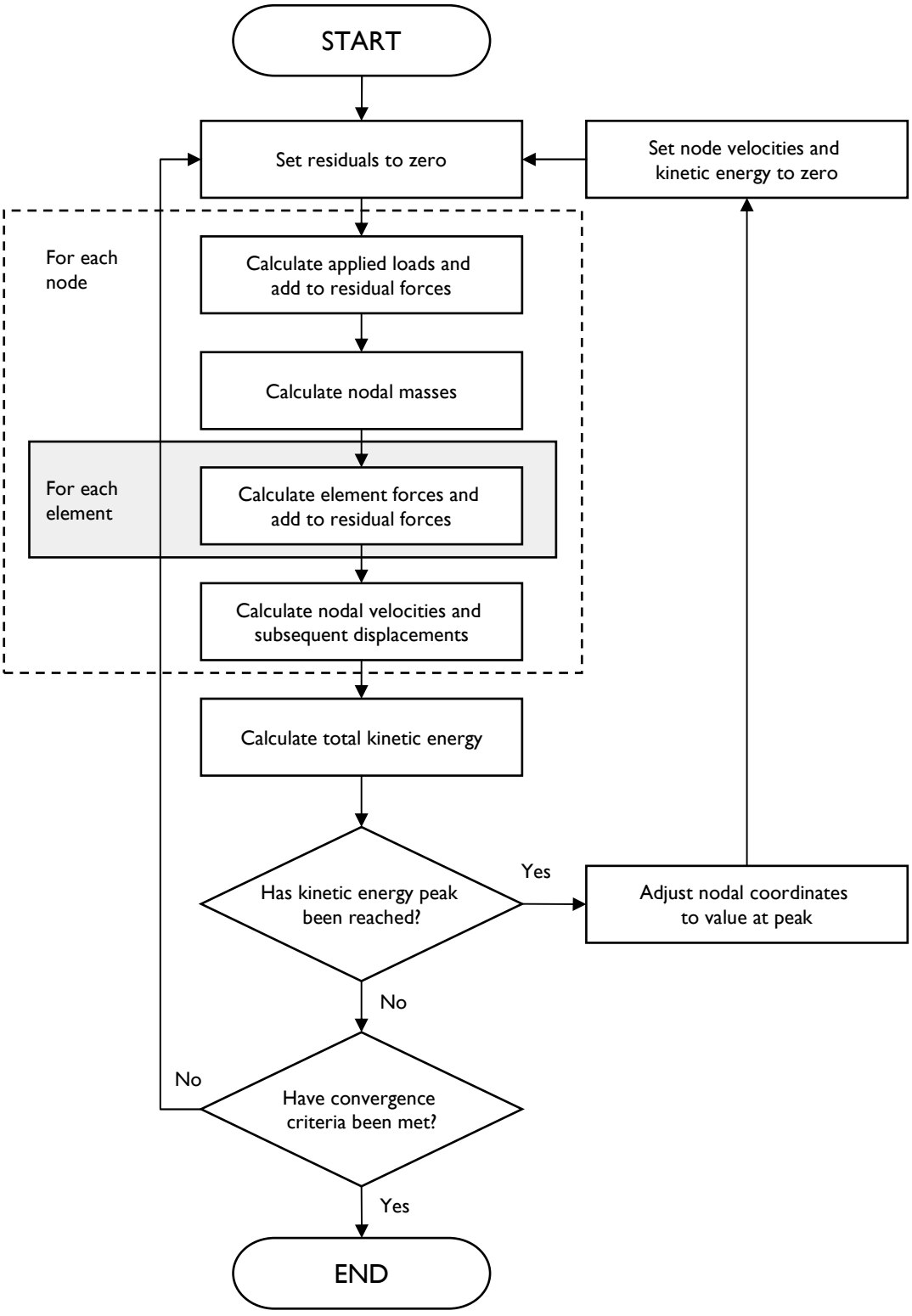


Figure 5.5
Flowchart describing the BESO algorithm

where

F = externally applied load
T = tension coëfficient at element m
T^0 = initial tension coëfficient (e.g. pretension)
EA = element stiffness
L = current element length
L^0 = initial element length

It is noted that the actual material properties of the elements are not required for form-finding, since the shapes are a function of the relative stresses, so even the member forces may be scaled from the desired prestress level. For structural analysis however, the final geometry can be used, but the values have to be scaled to be physically valid.

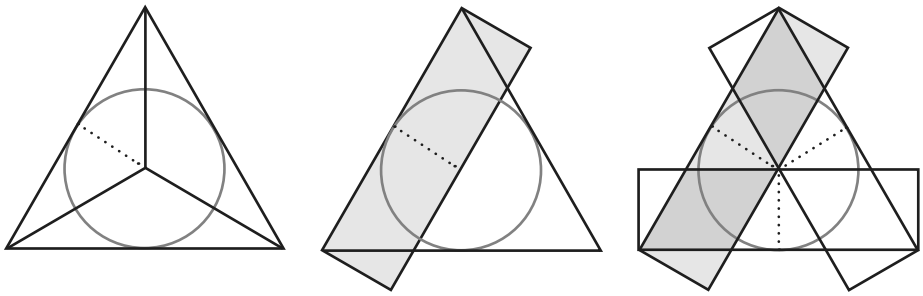


Figure 5.6
The element stiffness, of each side of the triangle, is calculated by taking the inradius of that side times the length of the side.

Calculating the element stiffness EA in a cable-net is very straightforward, but in this case some analogy had to be made between the fabric and its elements. No information was found in the DR sources, so it was decided to calculate the stiffness EA as the product of the linearized modulus of elasticity from Schmitz (2004) (Chapter 3.3), the thickness of the fabric and the inradius of the neighbouring triangle elements. Whether the element borders one or two triangles determines if this calculation has one or two terms. In Figure 5.6 it is visible that when this EA is multiplied by the length of the element the triangle area is actually used twice. This is actually an approximation of the interaction of the multiple fabric directions. It is very important to note that this approximation has not yet been verified through experiments.
The inradius r, the radius of the largest circle inscribed by a triangle, is the quotient of the triangle's area Δ (Eqn. 5.8) and its semiperimeter s (Eqn. 5.9). The area Δ is calculated using Heron's formula:

$$\Delta = \sqrt{s(s-a)(s-b)(s-c)} \tag{5.8}$$

$$s = \frac{a+b+c}{2} \tag{5.9}$$

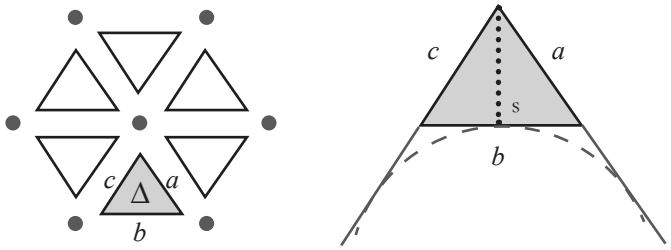


Figure 5.7
The area of the triangle is calculated using Heron's fomula, which uses the length of the sides and the semiperimeter of the triangle.

Which can be rewritten as:

$$\Delta = \frac{1}{4}\sqrt{\left(a+\left(b+c\right)\left(c-\left(a-b\right)\right)\left(c+\left(a-b\right)\right)\left(a+\left(b-c\right)\right)\right)} \tag{5.10}$$

where a,b and c are the sides of the triangle.

The element stiffness EA for element m can then be calculated as:

$$EA_m = \sum E \cdot r \cdot t \tag{5.11}$$

where

E = *490 N/mm* (Schmitz, 2004)
t = *0.76 mm*

and the inradius r is:

$$r = \frac{\Delta}{s} \tag{5.12}$$

At the end of each iteration the kinetic energy of the current structure is calculated. None of the sources actually specify how the kinetic energy should be calculated, so this was assumed to be according to conventional mechanics.

$$E_k^{t+\Delta t} = \sum \frac{1}{2} M_i \cdot \left(V_{ij}^{(t+\Delta t/2)} \right) \tag{5.13}$$

All literature that was read on the subject naturally concludes by stating that DR reaches a solution when convergence is met. However, none of the sources specified how convergence was measured, so through a process of trial and error, a convergence strategy was devised that was threefold. The DR method used for this thesis converges when one of three events occur:

1) The change in kinetic energy between iterations is sufficiently small and/or
2) The change in displacement between iterations is sufficiently small, or
3) A maximum number of iterations was reached i.e. the solution was not convergent

The reason for including both (1) and (2) was that the following two situations were encountered:

a) After an early peak in kinetic energy, displacements would initially be small, yet the motion was merely accelerating once again and no actual solution had yet been reached. A different time increment might have pre-empted this situation, but might not have been as foolproof.
b) The variations in kinetic energy were large due to some oscillating mode in the structure. Displacements however, remained small. These oscillations occurred despite measures in calculating the position kinetic peak more precisely (Section 5.3.1).

Eventually, one source was found, but in hindsight did not affect the convergence strategy that had already been chosen. The source mentioned was a user manual for `GSRELAX`, a plug-in module for the analysis program `GSA`. It mentioned using the residuals i.e. the acceleration as a means of measuring convergence by checking if the moment had been reached at which all node residuals did not exceed some limit. It did not specify how this limit was determined. Of course, if this limit refers to some minimum change in the value of the residuals, it does not differ fundamentally from using the change in kinetic energy. The latter is a quadratic extrapolation of the former (Eqn. 5.13), and in both cases a limit to the change can be chosen accordingly.
The procedure in `GSA` also ended the algorithm if some number of iterations (3) or processing time was exceeded.

The flow chart in Figure 5.5 illustrates the entire iterative procedure of Dynamic Relaxation. On a side note, Barnes (1998), though an otherwise excellent source of information, incorrectly states that node velocities are reset if the current kinetic energy is higher than the previous level. Of course these energy levels at reset should be vice versa.

5.3.1 Kinetic energy peak

As the algorithm progresses in discrete time increments, the kinetic energy at each step is of course discretely determined as well. As mentioned, whenever a kinetic energy peak has been reached i.e. the current kinetic energy level is lower than the previous one, the algorithm is reset. However, when a drop has been detected, the actual maximum level has been overshot. Both Barnes (1998) and Lewis (2003) propose a procedure for estimating this maximum point to further assist convergence.

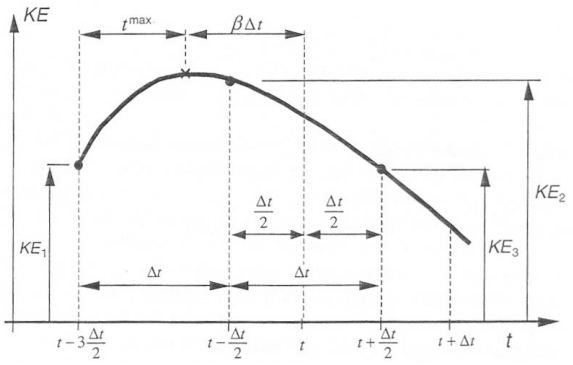


Figure 5.8
Quadratic interpolation of the time at the maximum kinetic energy peak during dynamic relaxation
Barnes (1998)

As seen in Figure 5.8. the level of kinetic energy that runs through three calculated points is assumed to be a quadatric polynomial $P(x) = ax^2 + bx + c$. For $x=0$, $x=\Delta t$ and $x=2\Delta t$, this gives three equations with unknowns a and b. Solving these gives t^{max}, and a correction for the displacement.

$$\begin{aligned} x = 0, &\quad P(0) = c = KE_1 \\ x = \Delta t, &\quad P(\Delta t) = a\Delta t^2 + b\Delta t + c = KE_2 \\ x = 2\Delta t, &\quad P(2\Delta t) = 4a\Delta t^2 + 2b\Delta t + c = KE_3 \end{aligned}$$

Solving the above set of equations gives the remaining coëfficients:

$$\begin{aligned} a &= \frac{1}{2\Delta t^2}\left[KE_3 - 2KE_2 + KE_1\right] \\ b &= \frac{1}{2\Delta t}\left[4KE_2 - 3KE_1 - KE_3\right] \end{aligned}$$

So that the time at the kinetic energy peak is given by:

$$t^{\max} = -b/2a = -\Delta t\,\frac{1}{2}\,\frac{4KE_2 - 3KE_1 - KE_3}{KE_3 - 2KE_2 + KE_1} \tag{5.14}$$

It is noted that Lewis (2003) forgets the factor ½, though the final equation is given correctly. At this point the offset β (called q in some other papers) from time t is calculated:

$$\begin{aligned} \beta\Delta t &= \frac{\Delta t}{2} + \Delta t - t^{\max} = \frac{3}{2}\Delta t + \frac{1}{2}\Delta t\,\frac{4KE_2 - 3KE_1 - KE_3}{KE_3 - 2KE_2 + KE_1} \\ &= \Delta t\,\frac{KE_3 - KE_2}{KE_3 - 2KE_2 + KE_1} \end{aligned} \tag{5.15}$$

So that the actual displacement is corrected using this final formula:

$$\begin{aligned}\delta_{ij}^{t^{\max}} &= \delta_{ij}^{(t+\Delta t)} - V_{ij}^{(t+\Delta t/2)}\Delta t - \beta\Delta t \cdot V_{ij}^{(t-\Delta t/2)} \\ &= \delta_{ij}^{(t+\Delta t)} - (1+\beta)V_{ij}^{(t-\Delta t/2)}\Delta t - \frac{\Delta t^2}{M_i}\cdot R_{ij}^t\end{aligned} \tag{5.16}$$

where

KE_i = kinetic energy E_k at $i=[1,2,3]: t=\left[t-\frac{3}{2}\Delta t, t-\frac{1}{2}\Delta t, t+\frac{1}{2}\Delta t\right]$

In summary, whenever a kinetic peak has been reached, all nodes are readjusted to the coordinates that follow from (5.16) to produce more accurate intermediate results and overall both stabilize the process and speed up convergence. Indeed, removing this option proved its necessity during the course of this thesis project, as some fabric models would oscillate in turn delaying convergence to a considerable extent, and in some cases even diverge.

5.3.2 Determining nodal masses

The nodal mass M will determine the inertia of each node as it is subjected to forces. Because the process of motion is entirely fictitious, these values need not be realistic. Their value will influence convergence or cause divergence so some care must be taken when examining them. There were three differing definitions found for the nodal mass in a cable-net structure. Using (5.7) the last two are expanded.

$$M_i = \frac{\Delta t^2}{2}\cdot\sum\left[\frac{EA}{L^0} + g\frac{T^0}{L}\right]_m \tag{5.17}$$

$$M_i = \frac{\Delta t^2}{2}\cdot\sum\left[\frac{EA}{L^0} + \frac{T}{L}\right]_m = \frac{\Delta t^2}{2}\cdot\sum\left[\frac{EA}{L^0} + \frac{EA\varepsilon}{L} + \frac{T^0}{L}\right]_m \tag{5.18}$$

$$M_i = \lambda\frac{\Delta t^2}{2}\cdot\sum\left[\frac{EA}{L^0} + \frac{T}{L^0}\right]_m = \lambda\frac{\Delta t^2}{2}\cdot\sum\left[\frac{EA}{L^0} + \frac{EA\varepsilon}{L^0} + \frac{T^0}{L^0}\right]_m \tag{5.19}$$

where

g = factor that accounts for cables sliding across eachother
λ = unspecified, constant convergence parameter

Barnes (1998), Lewis (2003) and Han & Lee (2003) respectively all offer slightly different definitions of the nodal mass, or stiffness, though all call it the sum of two terms; the elastic stiffness and the geometric stiffness. The former refers to the elastic stiffness of the elements and is the same in each case, however, the latter differs. The geometric stiffness is the increase in structural stiffness due to lateral loads and is the quotient of either the initial or current tension coëfficient and either the initial or current length.
In the first option, the nodal mass M decreases as tensile strain increases. In the second, it depends on the values of EA and T^0 whether the stiffness increases or decreases. And in the last, the stiffness increases as tensile strain increases.

It would be logical for the geometric stiffness to increase as the actual tensile strain increases, so the first definition (5.17) is ruled out as it does not include strain. It would also seem logical that the strain is dependent on the current length L as in (5.18) but that the tension coëfficient T^0 is related to the initial length L^0 as in (5.18). It seems that none of the mass definitions would be entirely correct, and this author proposes to redefine the nodal mass as:

$$M_i = \frac{\Delta t^2}{2} \cdot \sum \left[\frac{EA}{L^0} + \frac{EA\varepsilon}{L} + \frac{T^0}{L^0} \right]_m \tag{5.20}$$

However, due to the fact that no prestressing force was applied in advance, this definition was ultimately untested, because as $T^0=0$, so in effect, the value of the nodal masses became identical to definition (5.18) by Lewis.
Moreover, the actual definition of these ficititious masses is rendered moot, due to them only influencing the degree of convergence in each iteration, not the final geometry.

5.3.3 Numerical stability

The time increment Δt determines the stepsize and convergence of the algorithm. The value may be chosen arbitrarily up to a certain limit, but computation time is reduced when the value approaches this limit. The limit is defined as a function of the ratio between mass matrix M and stiffness matrix K.

$$\Delta t \leq \sqrt{2 \frac{M}{K}} \tag{5.21}$$

However, this is of particular importance for viscous damping, and in fact it is meant to prevent divergent oscillations and has to do with the eigenvalue of the structure. For kinetic damping, the masses will be fictitious and instead of adhering to (5.21), the time increment Δt can be chosen arbitrarily from which the nodal mass can be calculated using (5.22) instead of (5.20).

$$M_i = \frac{\Delta t}{2} K_i \tag{5.22}$$

Comparing (5.22) to (5.20) one notices that the time increment Δt is no longer quadratic and that for $\Delta t = 1$, both nodal masses would be the same.

5.4 Adapting Dynamic Relaxation to fabric formwork

Up to this point, DR (or any other form finding algorithm) has not been applied to fabric formwork, since this is a fairly new manufacturing technique. Several issues arise when adapting Dynamic Relaxation to Fabric Formwork.

1) Fluid pressures load the fabric as the fresh concrete is cast into the mold, which are unlike conventional load types for tension structures.
2) Contact between the mold and the fabric results in sharp corners not normally present in tension structures

Both issues will be discussed in the following seperate subparagraphs.

5.4.1 Fluid pressures of fresh concrete

The basic physics behind liquid concrete pressures is the formula for hydrostatic pressure. There are two differences compared to traditional civil engineering applications of hydrostatic pressure i.e. hydraulic structures. First, the liquid density ρ is that of concrete i.e. 2400 kg/m^3. Second is the direction of the pressure, which is perpendicular, or normal to the fabric mold surface and has to be calculated at each point (Of course this holds for hydraulic structures as well, but they generally feature vertical contact surfaces with water so are uniform in direction).

Intermezzo: Before explaining how liquid concrete pressures were integrated in FABRICFORMER, it is noted that this problem is very similar to the phenomenon of water ponding in membranes. It is traditionally a situation that the structural engineer of tensile structures tries to avoid, but nonetheless mathematical research has been done to model ponding. Katsikadelis and Nerantzaki (2002) use a 'boundary only method' based on the 'analog equation method' to produce an accurate modelling of ponding. They concluded quite unremarkably that transverse displacements are more than one order larger than in-plane displacements. Tuan (1996) used a 'fourth-order Runge-Kutta numerical integration' with similar results, and also wrote an iterative finite element simulation in ANSYS for comparison. He made the same conclusions, but also investigated partial ponding, concluding among other things that an inflexion point of zero curvature will form inside the ponding. Overall, sources on ponding are scarce.

FABRICFORMER calculates concrete pressure forces at each node. First the normal vector and surface area of each triangle is calculated, where the triangle is defined by vectors u,v,w with lengths a,b,c.
The normal vector is given by the cross product of two random vectors describing two of three sides of the triangle.

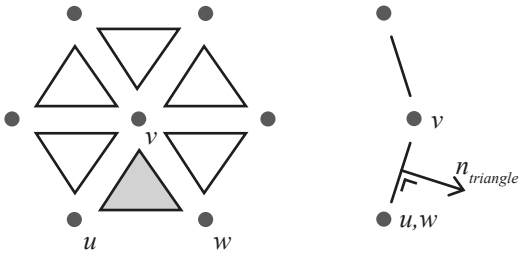


Figure 5.9
For each triangular mesh element the normal vector can be calculated from three nodes

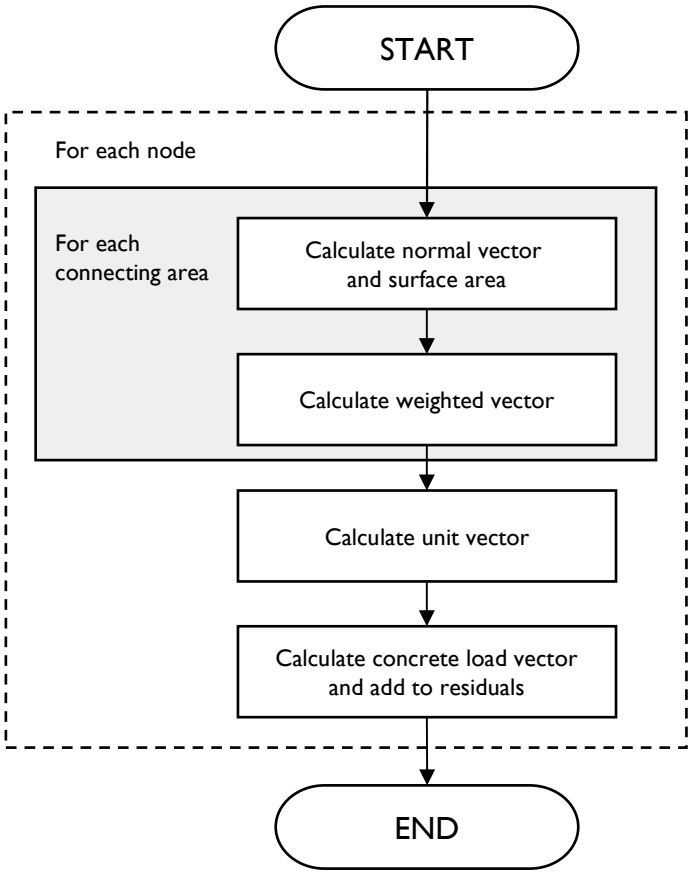


Figure 5.10
Flowchart describing how the concrete loads are calculated for each node

$$n_{triangle} = u \times v = \begin{bmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \end{bmatrix} \tag{5.23}$$

The surface A is calculated using Heron's formula (5.8) and the formula for the semiperimeter s (5.9).
The normal vector of each node is then calculated as a weighted sum of all surrounding triangle normals.

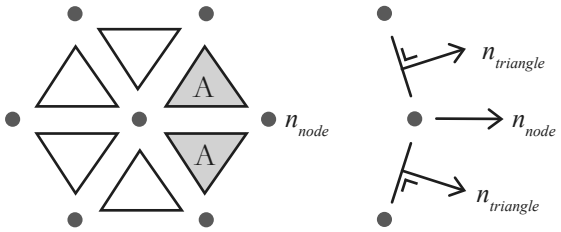


Figure 5.11
For each node the concrete load vector is calculated as a weighted sum of all triangle load vectors

$$n_{node} = \sum \frac{n_{triangle}}{A_{triangle}} \tag{5.24}$$

The unit vector is calculated and used to calculate the nodal concrete load q_c.

$$\hat{n}_{node} = \frac{n_{node}}{\|n_{node}\|} \tag{5.25}$$

$$q_c = \rho g h \cdot \hat{n}_{node} \tag{5.26}$$

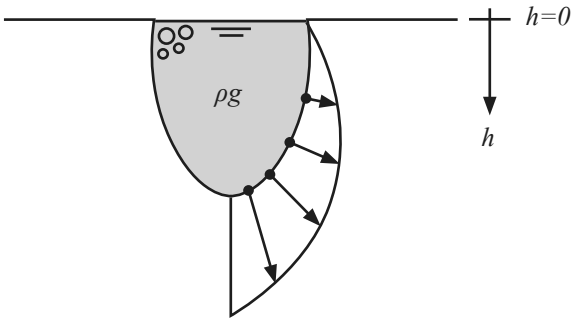


Figure 5.12
The final result of the load vector calculations is an accurate representation of the fluid pressures from the fresh concrete.

This procedure is summarized in the flowchart (Figure 5.10). Figure 5.12 shows the results in cross-section.

5.4.2 Modelling contact between mold and fabric

When fabric formwork is used, a flat sheet of fabric is filled with concrete. The fluid pressures of the concrete are balanced out by the prestressing forces on the outer edges of the flat sheet. The contours of the concrete element are determined by the edge mold, which doubles as a kind of pulley in the sense that it changes the direction of the applied forces. Either a horizontal plane of plywood, or steel pipes, have been used as an edge mold.
The difficulty in adapting DR to fabric formwork lies in the changed direction of the forces along the edge mold. The program has to ensure the following:

1) Elements that span across the edge retain their correct length and use a temporary third point, the edge, as a reference to calculate this (see Figure 5.13a and 5.14).
2) Nodes passing over the edge stay within the plane of the fabric – which now folds – and their residual force vector and velocity vector change direction accordingly (see Figure 5.13b and 5.15).

For the pinch mold the fabric also interacts at locations where solid parts of the mold press against the fabric effectively 'pinching' the fresh concrete. These locations will be referred to as pinch points. The calculations for contact with the pinch points are nearly identical to those for the edge mold, so for the remainder of this paragraph only the interaction with the edge mold will be described.

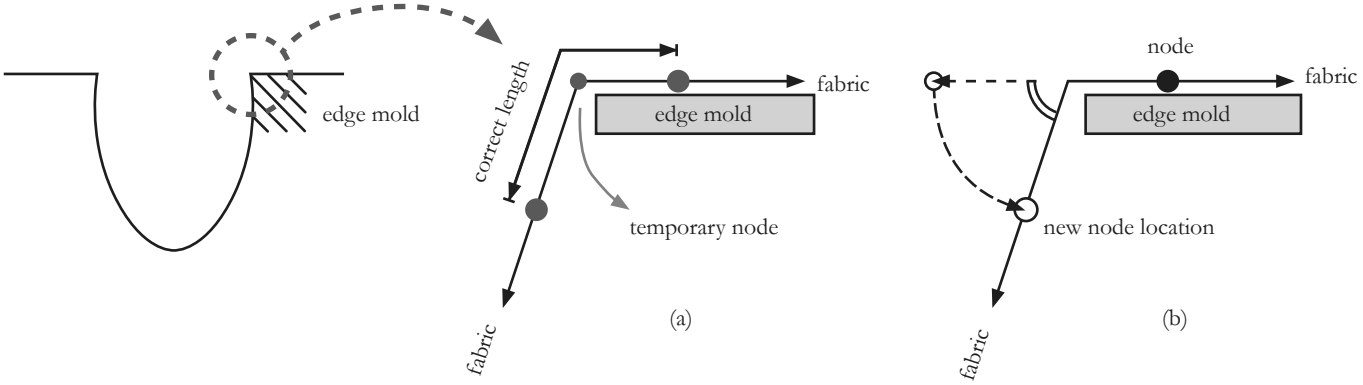


Figure 5.13
At the edge, two unique situations can occur, where elements span across the corner, and nodes could pass along them.

The elements that span across the edge of the mold are identified by checking if their two connecting nodes are on opposites sides. In that case, a temporary edge point is linearly interpolated. This approximation prevents the need for an iterative procedure along the (Beziér) curved edge (see Section 5.5.1) to determine what the shortest distance between these points actually is. On the other hand it requires some post-processing corrections to prevent any divergent behavior. Using the temporary edge point, the actual length between the nodes and real direction of the nodal forces is calculated. This length now determines the element tension forces, which are added to the residuals using the correct direction i.e. their direction unit vectors.

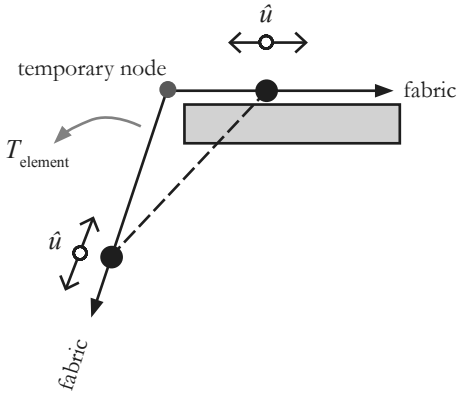


Figure 5.14
Element spanning across the edge of the mold use a temporary node to calculate their actual tension coëfficient and length.

The flowchart in Figure 5.16 summarizes these calculation for which no math is given, since the calculations are fairly straightforward.

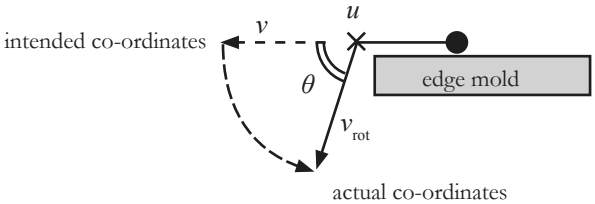


Figure 5.15
Nodes displacing beyond the edge of the mold are rotated to their actual coordinates, including their nodal forces and velocities

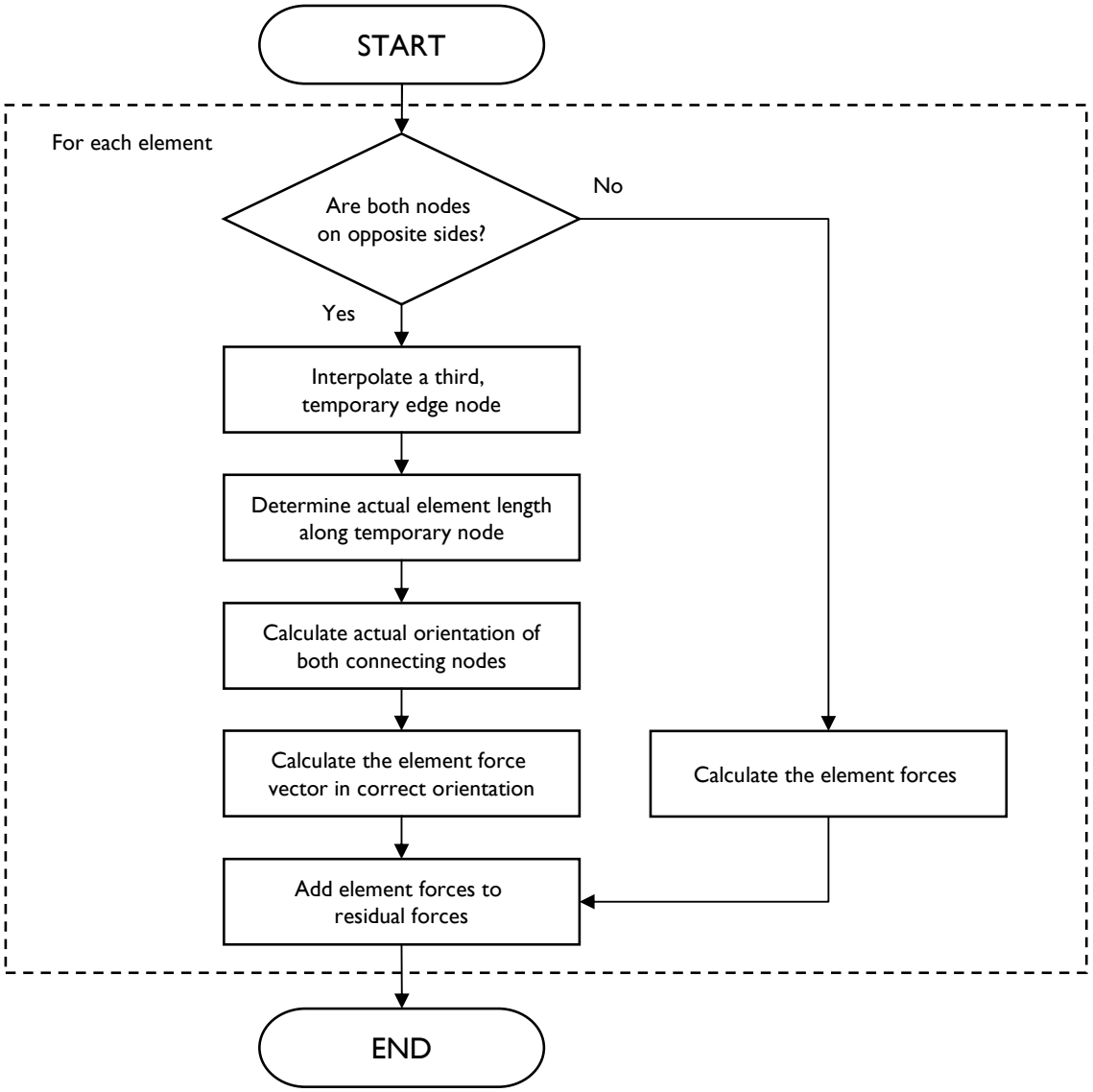


Figure 5.16
Flowchart describing how properties for elements spanning the corner of the edge mold are recalculated.

Some of the nodes will be pulled over the edge, which is possible in either direction i.e. situation (2). In the earlier calculations for situation (1) for all elements, and thus for all nodes it was determined which are located at the edge of the mold. Based on their distance to the edge and displacement all nodes could be identified that were pulled up or down over the edge.
For each of these nodes it is essential to calculate how far they 'want' to pass over the edge, then determine how they will rotate over the edge, what their actual new coordinates will be as well as the new direction of their nodal forces and velocities (see Figure 5.15 and the flowchart in Figure 5.17).

The distance to the edge is used to calculate a temporary edge point, through which displacement occurs. Then the 'surplace' displacement vector v can be calculated. This is the remaining displacement vector that will take place on the other side of the edge mold, but needs to be rotated to the new direction. The orientation of the edge can be determined using the derivative of the Beziér curve (5.27) that determines the shape of the edge mold. This is the rotational axis u (perpendicular to Figure 5.15) along which the surplus displacement vector v will rotate.

$$B'(t) = \sum_{i=0}^{n-1} \binom{n-1}{i} \left\{ n\left(P_{i+1} - P_i\right) \right\} (1-t)^{n-1-i}\, t^i \tag{5.27}$$

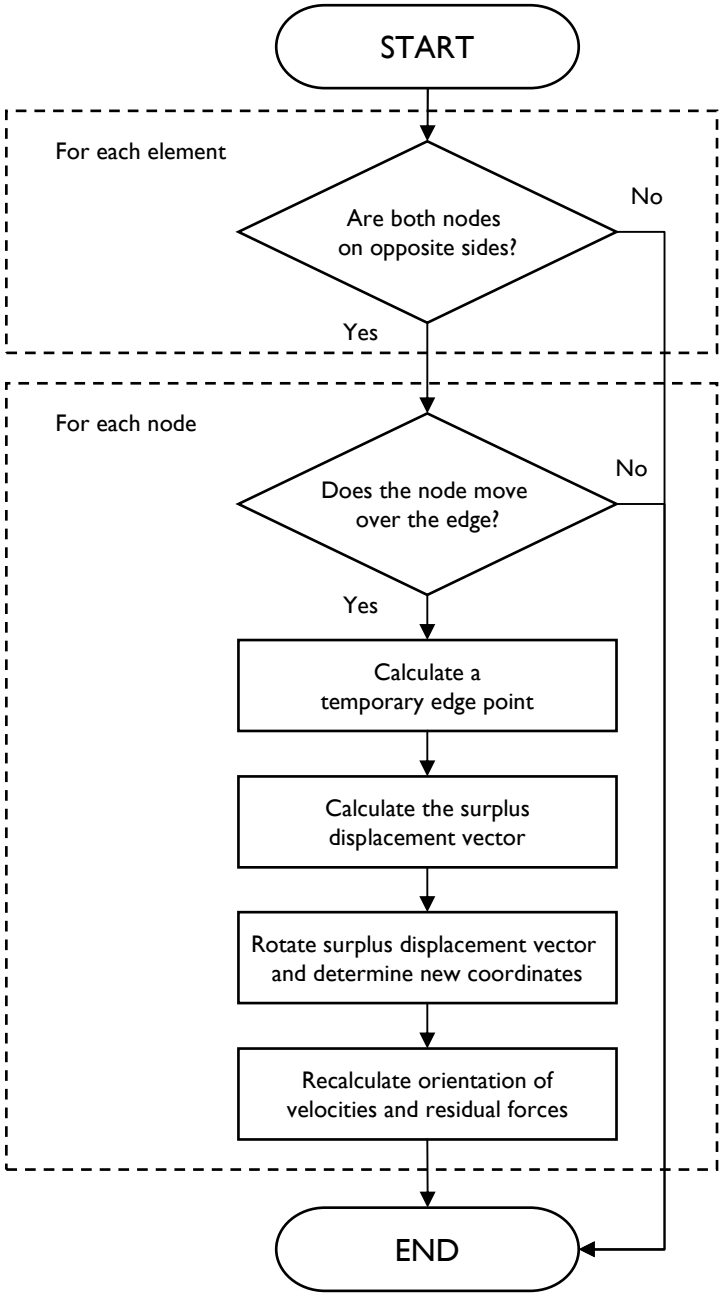


Figure 5.17
Flowchart describing how properties for nodes moving across the corner of the edge mold are recalculated.

Using the two arbitrary vectors that lie within the planes on each side of the edge, we can determine the angle of rotation θ by projecting these vectors onto a plane perpendicular to the rotational axis u i.e. the orientation of the edge. Now, with both the angle and axis of rotation, the surplus displacement vector v is rotated to match the new, correct direction v_{rot}. This rotation is done using Rodrigues' formula of rotation.

$$v_{rot} = v \cdot \cos\theta + u \times v \cdot \sin\theta + u^T v u \cdot (1 - \cos\theta) \tag{5.28}$$

A unitvector of the rotated displacement vector v_{rot} is used to adjust the direction of the residual forces R_{ij} and velocities V_{ij} as well.

5.5 Implementing Dynamic Relaxation

The first stage to implementing DR was to code the basic algorithm in Java. Initially, the programming proceeded very promisingly, and early results showed that implementation was easy to accomplish (Figure 5.18). However, applying the specific algorithm (Appendix C) for membranes proved exceedingly more difficult and after three weeks did not produce satisfactory results. The choice was made to fall back on DR for cable-net structures, which also means that `FABRICFORMER` approximates membrane behaviour analogously. In hindsight, the problems with using membrane-specific DR could be attributed to numerical instability of its trigonometric equations and other calculations, because apparently Java uses 'floating-point data' that can ultimately cause small numerical errors, cumulatively leading to divergent behavior. As this possible explanation was considered after the fact, for future work it is recommended to use the following paper as a starting point i.e. 'Miscalculating Area and Angles of a Needle-like Triangle', W. Kahan (2004), and reassess how membrane-specific DR should be coded. Overall, implementing a new and functional DR algorithm for all types of fabric formwork molds became the most time-consuming part of this thesis project taking up several months of programming. A second approximation is acknowledged in the use of a constant modulus of elasticity E. Doing so is very common in civil engineering practice, illustrated by the existing software often allowing only single values as an input for E. Non-linear relations in the form of stress-strain curves can be derived from tests such as bi-axial testing. Such results have been implemented in FEM software, but in general non-linear material analysis of fabrics is not standard practice. The model in this thesis also used a linearized modulus of elasticity, but adding a non-linear relation would not be particularly difficult as long as such information is available. Of course doing so adds another iterative procedure as the elasticity has to be determined at each step, altogether increasing the computational load of the program.

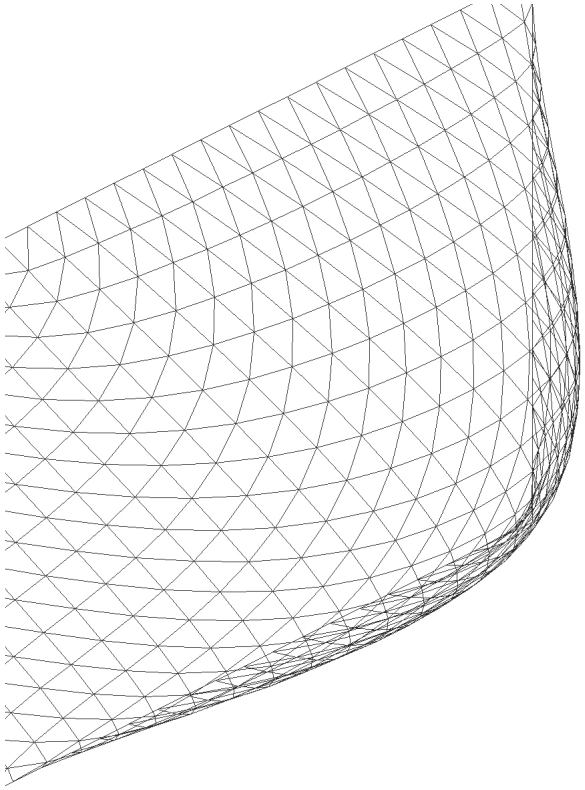


Figure 5.18
One of the earliest outputs of the DR algorithm, displayed in a 2D isometric Java Swing GUI. This cable-net is supported at the sides and subjected to an evenly distributed load from one side.

The most basic model of fabric formwork in Dynamic Relaxation is shown in Figures 5.19 and 5.20. Figure 5.19 shows the mesh prior to DR. A triangular mesh is generated and fixed i.e. supported at the bottom edge. At a certain height, where the horizontal plane mold is located, the fabric mesh folds at a 90 degree angle. At this opposite edge the prestressing will be applied causing the fabric to be pulled over the edge of the horizontal mold. The concrete pressures are applied simulaneously and will pull the fabric in the opposite direction of the prestressing forces. The DR algorithm finds equilibrium between these two types of forces. Figure 5.20 shows such an outcome for a situation where all the mold parts

and prestressing forces are identical along the length. Unsurprisingly the resulting beam is prismatic i.e. the cross-section is the same along the longitudinal axis.

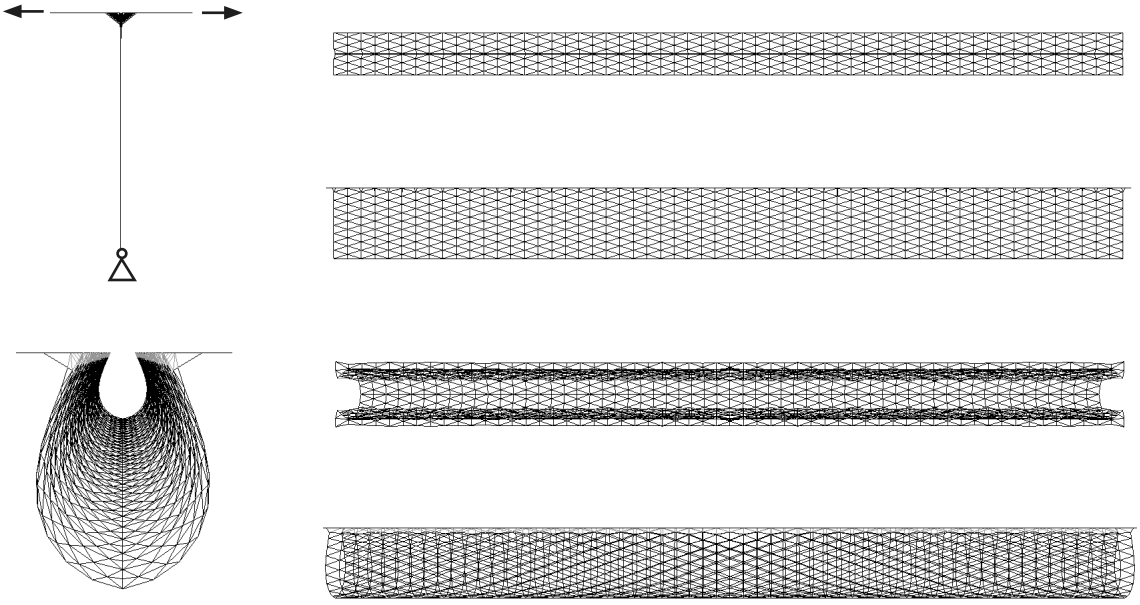


Figure 5.19
Fabric mesh in the DR algorithm, output in the Java3D GUI, prior to loading.

Figure 5.20
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures.

Variables to optimize are derived directly from the possibilities of keel and pinch molding, which are the two most actively pursued mold types of fabric formwork for beams at the moment (Chapters 3.2.2-3.2.3). Though initial versions of `FABRICFORMER` were modelled after the less complex spline mold (Chapter 3.2.1), results of these models are not shown, as they were merely a stepping stone to the other mold types.

5.5.1 Modelling the keel mold

The keel mold consists of basically three elements that, in interaction, determine the ultimate shape of the mold. These are:

- Prestressing along the length of the fabric
- Edge shape of the horizontal plane mold
- Keel shape of the vertical plane mold

The prestressing is typically applied at several points along the length of the beam, at one side of the fabric. In many instances, though not necessarily, the fabric is fastened at the edge with timber blocks. The prestressing forces then transfer to these timber blocks (Chapter 3.2.2).
Although the prestressing is generally applied discretely, for the modelling in fabric formwork it is done continuously along the length. A five point i.e. four degree Beziér curve B describes this prestressing.

$$B(t) = \sum_{i=0}^{n} \binom{n}{i} P_i (1-t)^{n-i} t^i \tag{5.29}$$

$$\text{where } \binom{n}{i} = \frac{n!}{(n-k)!k!}$$

n = number of degrees = 4

For a four degree curve the formula for a Beziér curve (5.29) becomes (5.30):

$$B(t) = P_0(1-t)^4 + 4P_1(1-t)^3 t + 6P_2(1-t)^2 t^2 + 4P_3(1-t)^1 t^3 + P_4 t^4 \ , \ t \in [0,1] \qquad (5.30)$$

$$\text{where } P_0, P_1, P_2, P_3, P_4 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \begin{pmatrix} x_2 \\ y_2 \end{pmatrix}, \begin{pmatrix} x_3 \\ y_3 \end{pmatrix}, \begin{pmatrix} x_4 \\ y_4 \end{pmatrix}$$

Point P_2 is fixed halfway along the length of the beam and serves as the mirrorpoint for the curve. One of its coordinates is therefore known. Because of the mirroring, points P_0 and P_4 as well as points P_1 and P_3 are each others mirror image. This reduces the number of variable coordinates by four. Since point P_0 is at the start of the beam, one if its coordinates is also known. If x is the longitudinal direction of the beam, only y_0, x_1, y_1 and y_2 remain; a total of four variables describing the curve.

For manufacturing an appropriate translation is necessary between the Beziér curve and the actual prestressing to determine how this distributed force needs to be applied. When using the aforementioned timber blocks, such calculation is not necessary, since the shape of the fabric inside and outside the mold is known, and could be directly fastened.

Figure 5.21 shows the effect of only varying the prestressing along the length. At endspan the prestressing is set considerably lower than at midspan which is why the fabric is closer to the edge of the horizontal plane mold. From the sides it is visible that the lower prestressing gives way to higher volumes of concrete. In effect, the fabric billows out at these sections.

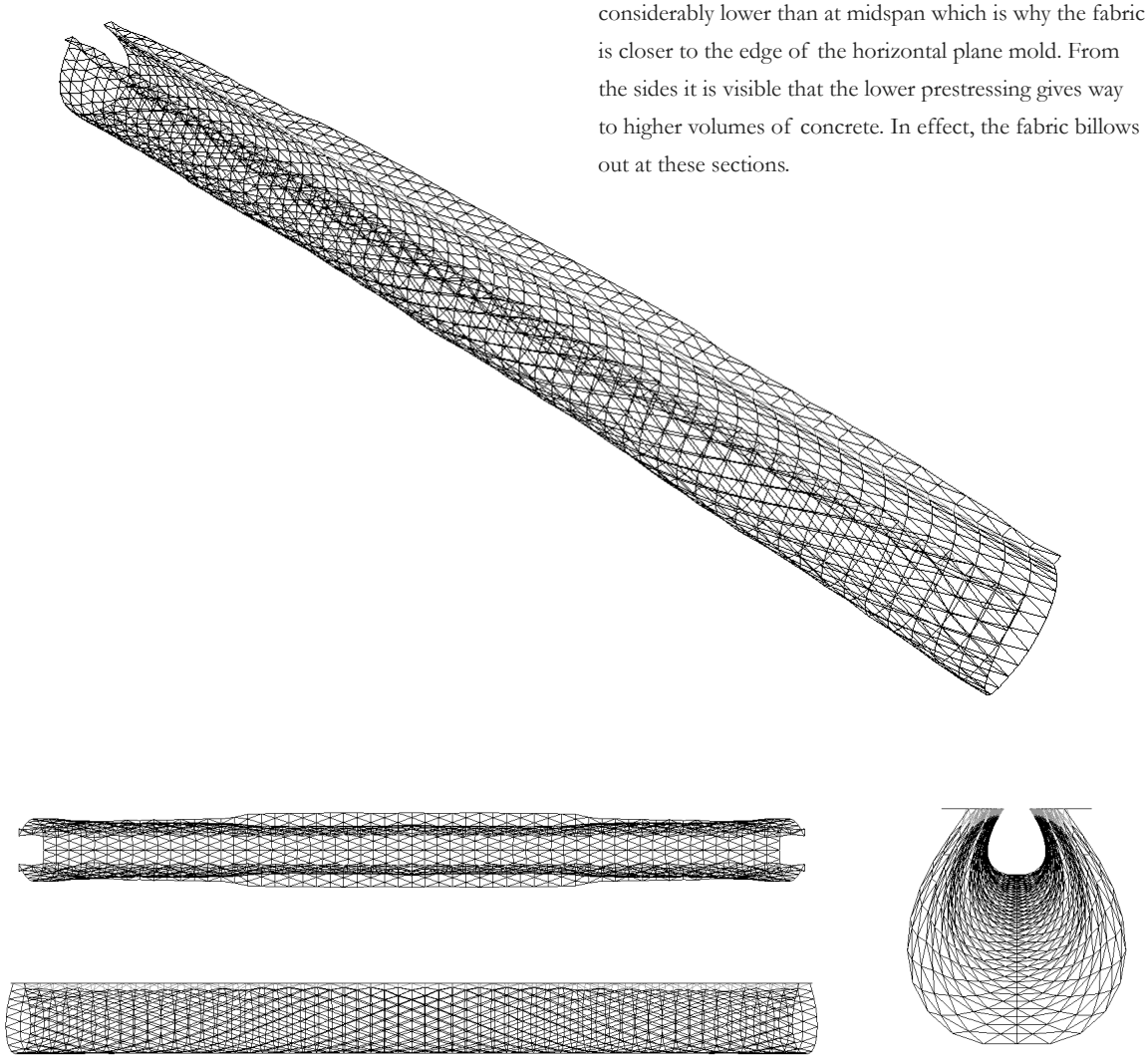
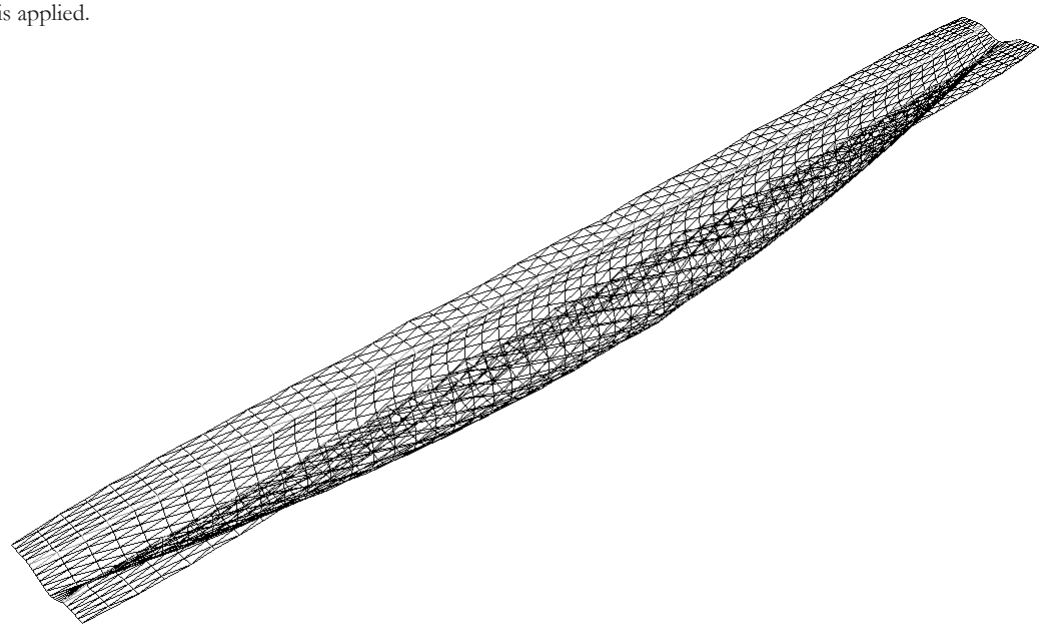


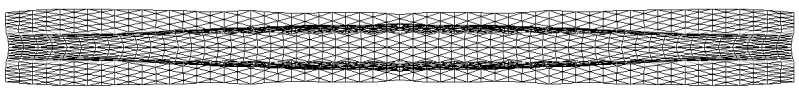
Figure 5.21
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures. The prestressing, defined by a Beziér curve, varies along the length

The edge shape of the horizontal plane has, up to this point, been made from steel pipes or timber sheets. In the first case the edge shape would be a straight line, whereas in the second case the timber sheets could be cut, or sawn, to give any type of planar shape. Again, the decision was made to model this variable using a continuous Beziér curve (Eqn. 5.30), resulting in four more variables.

One possible result of varying the keel shape is shown in Figure 5.22. It is clear that this shape has a large determining influence upon the final beam shape. When viewed from the side along the longitudinal axis, it is also visible that the interaction between prestressing and concrete pressures will determine whether the cross-section is U- or V-shaped.

The keel shape of the vertical plane is made of timber, cut or sawn to produce the required bottom shape of the beam. Along this keel, two flat sheets of fabric are fastened along one side, opposite the side where prestressing is applied.





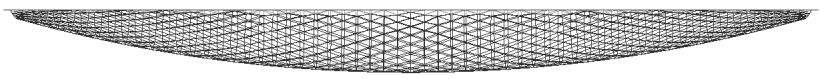


Figure 5.22
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures. The depth of the keel mold, defined by a Beziér curve , varies along the length

Similar to the keel, the edge shape is modelled using a Beziér curve with four undetermined variables.

Changing only the edge shape, as in Figure 5.23 where it causes the top 'flange' to become narrow at midspan, has a relatively minor effect on the overall beam shape. The edge shape directly influences the amount of material at the top half of the beam and also slightly determines the cross-section. It is worthwhile to optimize though its effects, for this type of structure, are limited. It is also important to keep in mind that the edge shape determines the horizontal surface upon which the external loads are applied. It should not be allowed to form a too narrow top to avoid any peak stresses from occurring, though this requirement could also be taken into account during finite element analysis and subsequent optimization by penalizing high and peak stresses.

Putting all these variables together results in a total of twelve variables that have to be optimized during Differential Evolution.

It is also noted that other possibilities exist to model the curved keel and edge shape as well as the prestressing forces other than using Beziér curves. Some considered alternatives were:

- Polynomials of degree n
- Trigonometric functions
- Discrete shapes

Beziér curves were chosen as a way to describe relatively arbitrary curves in only a few variables. One drawback of these curves is the fact that for curves within the design domain the coordinates of the control points P_i could possibly lie outside the domain. Allowing a larger area for the control points to exist in turn allows curves outside the design domain. Such curves would not represent physically feasible beams and would disrupt the optimization process, since no feedback is drawn from such beams.
This phenomenon was resolved by restricting the control point to the design domain, so that all resulting beams are feasible. The disadvantage is that this restricts the range of curves that can be generated.

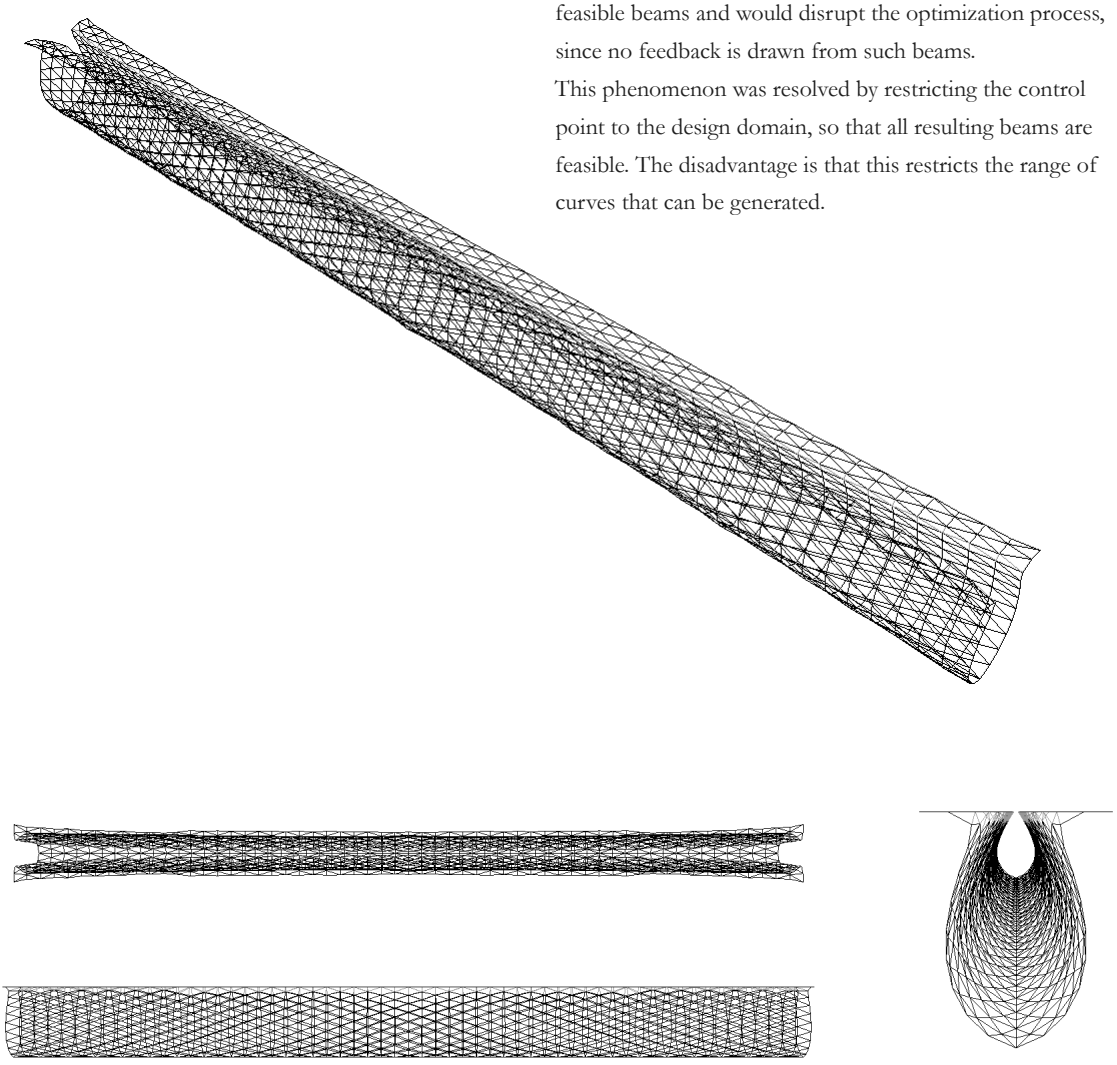


Figure 5.23
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures. The width of the edge mold, defined by a Beziér curve , varies along the length

5.5.2 Modelling the pinch mold

There is significant similarity between the keel and the pinch mold. Both of these types support the fabric at the bottom edge and prestress it at the outer top edges (in contrast to the spline mold). When comparing the keel and pinch mold obvious differences are the pinch points but also the wide spacer strip, allowing for a large variable width for the bottom flange of the beam. It can be concluded that the pinch mold adds upon the possibilities of the keel mold and so is not inherently different. For this reason the model of the pinch mold in Dynamic Relaxation uses the model for the keel mold as a starting point for which additional variables are introduced.

One additional variable is the spacing width which determines the base width of the beam i.e. the bottom 'flange' width. This spacing width corresponds to a single-curved timber sheet. The two flat sheets of fabric are fastened to the opposite sides of this timber sheet.
These two sides of the timber could also be cut or sawn in a certain pattern such as a continous curve allowing for even greater geometric freedom. Using Beziér curves and supposing that both sides are symmetrical, this would require no more than four additional variables. However, for sake of simplicity, these variables were not programmed into the model.

Shown in Figure 5.24 is the consequence of altering the spacing width. This particular example is much too bulky to be viewed as a mechanically optimal beam. Increasing the prestressing would result in a more efficient, nearly rectangular beam. However, for practical purposes, this is an unlikely candidate for fabric forming, since the prestressing forces would be too high. This in turn would dramatically increase the strength requirements for the fabric, which might be impossible or too costly. This reasoning shows how the spacing width and the pinch points work together, since the latter will decrease concrete volume near and at the neutral axis of the beam, creating efficient truss-type beams.

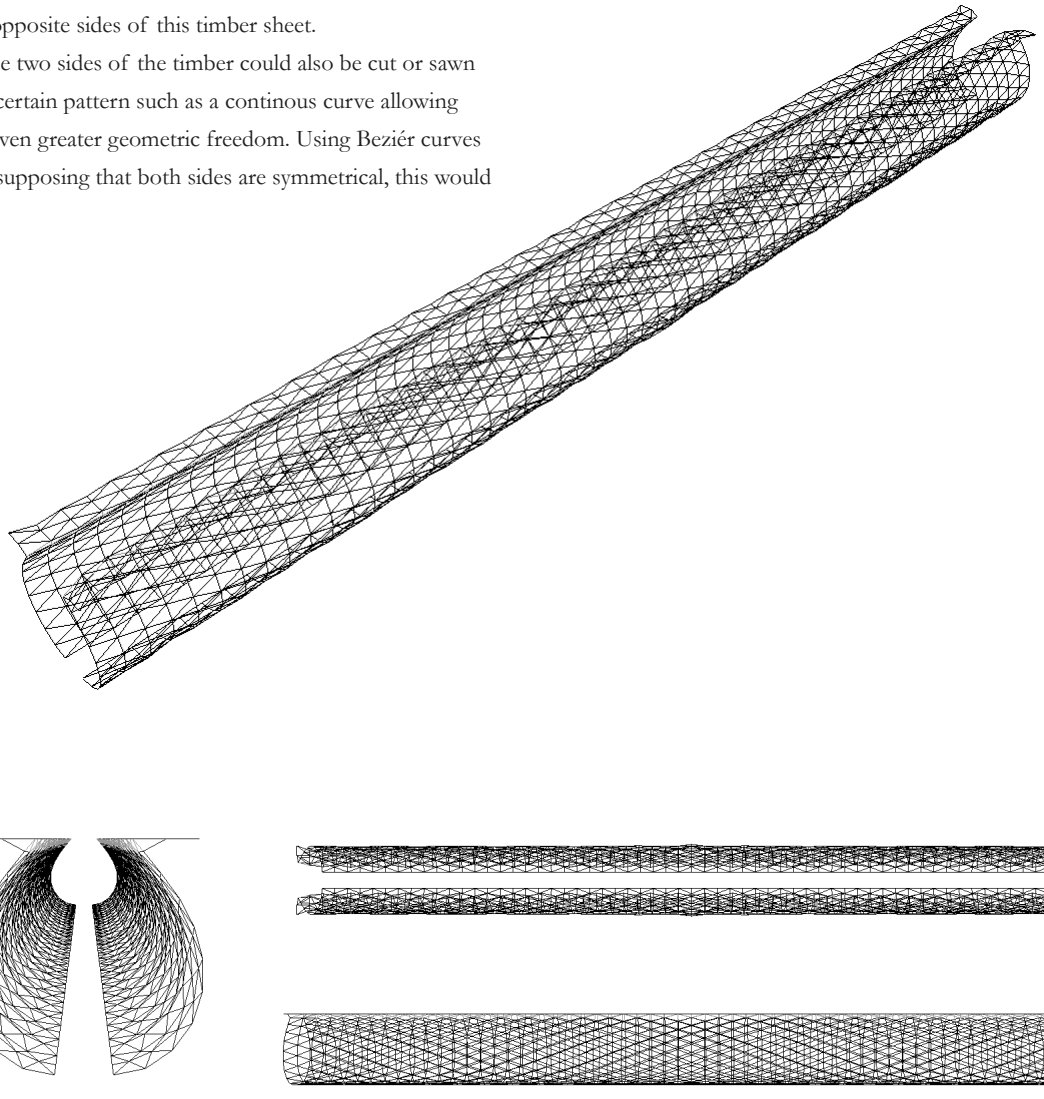


Figure 5.24
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures. The spacer width has been added and given a certain size.

Examples of truss-like concrete beams that have been cast can be found in Chapter 3. To model these points, the following was assumed:

1) There are at most three pinch points - when mirrored, six pinch points - in the beam
2) The pinch points are simple polygons, either convex or concave
3) The pinch points are quadrangles.
4) The pinch points can be pressed to a variable depth.

The first assumption is derived from the results of Bi-Directional Evolutionary Structural Optimization (BESO) in Chapter 8, which show that optimal reinforced concrete beams feature roughly five holes in longitudinal cross-section. Allowing a higher number of pinch points would only lead to much higher computational loads, while producing marginally more efficient beams, if any.

The second and third assumptions take into account a certain simplicity of the pinch point shapes. Although one could imagine far more complex shapes, simple quadrangles would be both easy to manufacture and more than sufficient to model the roughly quadrangular and triangular holes from the BESO results.

The fourth and final assumption expands on what has been developed in pinch molds up to this point. By allowing the depth of the pinch point to be variable, instead of holes, the mold could also produce webs. This immediately introduces fluid concrete pressures at these points, though this addition bears its practical consequences.

The choice was made to model the quadrangles using their Cartesian coordinates in a two-dimensional plane. In this case, each pinch point adds nine variables i.e. four sets of two coordinates and a pinch depth. For three pinch points this gives 27 new variables.
Together with the spacer width, there are now forty variables in total for optimization, a considerable larger number that for the keel mold (originally twelve).

The pinch points have been modelled so that they may overlap, allowing for slightly more complex geometries.

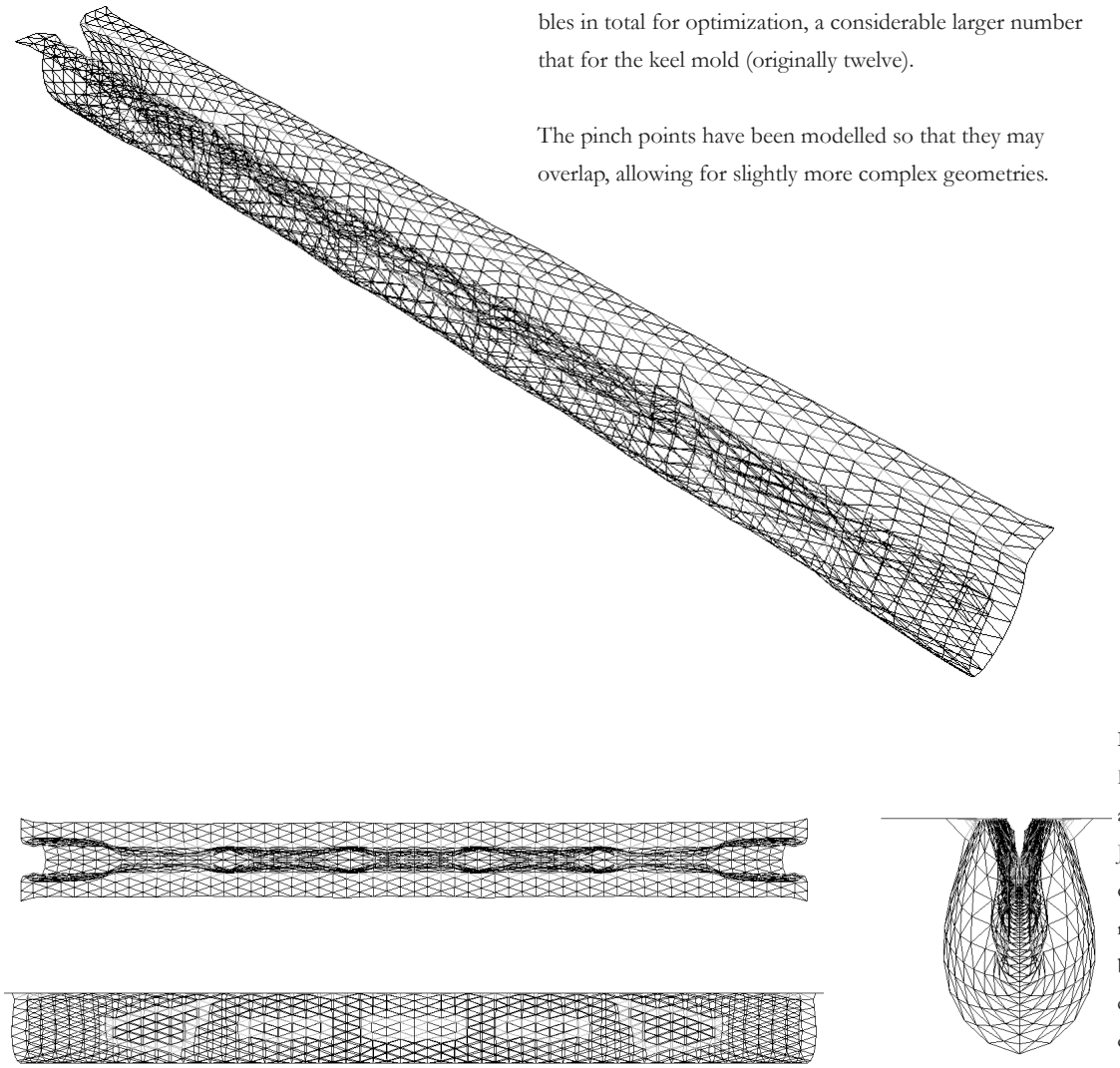


Figure 5.25
Fabric mesh in the DR algorithm, output in the Java3D GUI, subjected to concrete pressures. Three mirrored pinch point have been added to the model, described by three convex quadrangles, 'pushed' into the fabric to a certain depth.

However, as a future recommendation, the pinch points should not overlap but rather should be interdependant. In this case changing the coordinates of one pinch point would subsequently affect the shape and location of the other pinch points. This strategy would ultimately result in the generation of a relatively larger number of sensible beam shapes. The additional programming this would require is not inconsiderable and for this thesis it was decided to forego implementation of this idea. It is expected that this decision comes at a higher computational load for the program, but of course this cannot be verified at this time.

Figure 5.25 shows an example with three pinch points. The middle one borders the mirror plane, so that the beam has five pinch points in total. The pinch depth has been varied so that the middle pinch point will form a hole in the final beam and the outer pinch points will result in local webs. The outermost pinch points also demonstrate the flexibility of the quadrangles. During optimization the corner points can move anywhere within the beam. If points are relatively close to each other, resulting pinch points will become triangular or even form lines along which the fabric is pressed inwards.

5.6 Conclusions and recommendations

Dynamic Relaxation has succesfully been adapted to model every type of existing fabric formwork for beams. In its current state it will prove more than sufficient for the optimization process of fabric formed structural beams. However, for future purposes, several improvements and additions are possible, most of which have been listed throughout this chapter. These improvements and additions will lead to more accurate modelling of the formwork and will enable users to derive practical data from the models such as accurate fabric strains, prestressing forces and so on.

The algorithm could benefit from better collision detection between the fabric and the solid mold parts. In some instances, visual inspection has shown that some nodes oscillated, exhibiting behavior due to numerical errors. These errors could be traced to numerical errors due to Java's floating point calculus and/or due to linear interpolation in some of the calculations. Much effort was expended to solve and remove these oscillations, but this was only succesful to a degree. For newer versions these inconsistencies should be resolved by assessing the consequences of floating point calculus and/or improving the various interpolation routines. Another option would be to replace everything with an existing collision detection algorithm from literature, since the current algorithm was coded from scratch without reference to any sources.
Other improvements are adding membrane modelling as described in Appendix C and interdependancy of pinch points as discussed in the previous paragraph and shown in Section 8.6.

Several new additions to the program may also prove worthwhile. It would be possible to use pattern generation such as in current software for tension structures. These algorithms would provide optimal shapes for cutting patterns. It is also necessary to investigate the tear resistance of the seams under fresh concrete loading. (It is important to realize that current fabric formwork philosophy includes using only single flat sheets of fabric for simplicity.)
Another difference with tension structures is that there is no need to avoid folding of the fabric. However, folding might result in significant deviations from the intended design and/or esthetic requirements. For this reason, subsequent research might focus on the influence of folding on fabric formwork products as well as incorporating the phenomenon in the DR modelling. A good starting point is the paper of analysis of partly wrinkled membranes with DR by Haseganu and Steigmann (1994). One final addition is the introduction the non-linear, bi-axial properties that fabrics actually have, replacing the current linear, uni-axial modulus of elasticity. This implies bi-axial testing of the prospective fabrics prior to adding this to the algorithm.

Outside improvement of the algorithm itself, one can benefit from superior computational power, since this thesis used a standard university workstation. Better hardware is an obvious choice, but the program may also be adapted for parallel computation. Topping and Khan (1994) wrote a comprehensive paper on parallel computation schemes for DR.

The recommendations for the DR algorithm applied to fabric formwork are summarized in table 5.1.

TABLE 5.1 Recommendations

Improvements
Improve collision detection • More accurate interpolation routines • Investigate existing forms of collision detection Add interdependancy of pinch points Replace cable-net modelling with membrane modelling Investigate possible numerical errors due to floating point calculus General refactoring of the algorithm to improve speed (Superior hardware and/or parallel computation)
Additions
Add pattern generation and assess tear resistance of seams Assess influence of folding, improve modelling of folding Assess and possibly implement different types of generating certain variables (other than Beziér) Introduce non-linear material behaviour (based on results of bi-axial testing)

CHAPTER 6 | Finite element model for ANSYS

The software program ANSYS 10.0, later 11.0 was used for the finite element model of the resulting fabric formed beams. During each optimization iteration the FABRICFORMER program generates many shapes through Dynamic Relaxation. Each shape is then translated and transferred to code in ANSYSscript which ANSYS uses to start evaluation based on finite element analysis. In this analysis each beam is subjected to distributed loads and self weight, and its behaviour - defined by strain energy throughout the beam - is used to grade it with a performance index similar to the BESO algorithm. This chapter will detail how the concrete and reinforcement is modelled in ANSYS, as well as how the performance index is derived. Considerable effort has been made to devise a method for non-linear analysis, but this was ultimately only used in the BESO algorithm, and not in FABRICFORMER due to a lack of available computational power.

6.1 Existing ANSYS concrete modelling

The computer program ANSYS has several options for modelling reinforced concrete. All options involve the so-called SOLID65 element, an 8-node brick element capable of simulating cracking and crushing (Figure 6.1).

There appears to be limited material available on modelling concrete in ANSYS. Most sources pertaining to concrete modelling in ANSYS are of Chinese origin and not translated to English. Several other sources in English use ANSYS for the modelling of railway sleepers. It also became clear that at this university, other packages such as DIANA and CATENA are preferred over ANSYS for concrete analysis, and the latter is generally associated with mechanical engineering or analysis of steel.
However, there is an excellent paper by Barbosa and Ribeiro (1998) who have compared various methods of modelling reinforced concrete in ANSYS with actual test data. The version they used is relatively old - 5.3 instead of 11.0 - but the information provided remains relevant and applicable.

Barbosa and Ribeiro modelled a beam considering only the longitudinal reinforcement. They compared both discrete and smeared representations of the reinforcement (Figure 6.2) as well as linear elastic and three non-linear models for the concrete material. Discrete representations intuitively relate to actual reinforced concrete. This type of model involves connecting various nodes using two-dimensional LINK elements, effectively modelling the actual discrete reinforcement bars. Smeared reinforcement works by assigning a volume of SOLID65 elements with a local

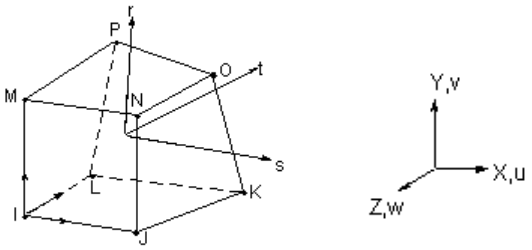


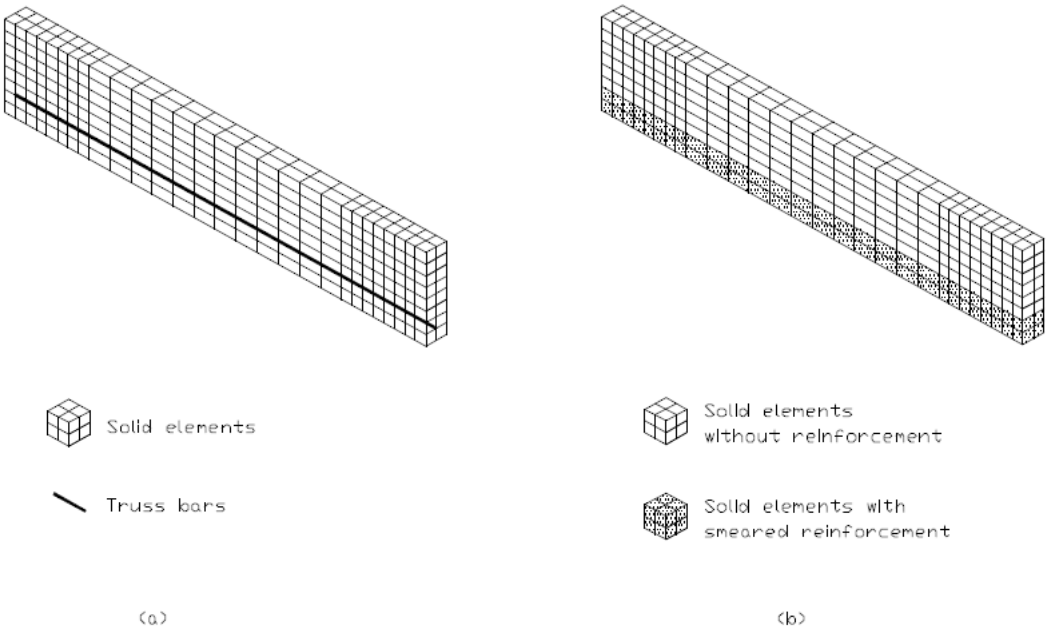
Figure 6.1
SOLID65 Element
ANSYS 11.0 User Manual (2008)

percentage of reinforcement as well as the orientation of the reinforcement (Figure 6.6). It is possible to assign up to three sets of percentages and respective orientations to model different types of reinforcement in one volume.

The three non-linear models Barbosa and Ribeiro used were:

- a Drucker-Prager yield criterion
- a multi-linear uniaxial stress-strain relation, simulating a parabolic curve and
- the same relation only with the additional option of crushing that ANSYS allows

Figure 6.2
Proposed mesh for a case beam with either discrete or smeared reinforcement modelling.
Barbosa and Ribeiro (1998)



The Drucker-Prager yield surface is a smooth, more conservative version of the Mohr-Coulomb yield surface, used to describe how a brittle material such as concrete fails due to interaction of local normal and shear stresses (Figure 6.3). The latter two models define a stress-strain curve as a set of points, that approximate the actual stress-strain curve of concrete (Figure 6.4). The option of crushing adds an additional non-linear aspect to the calculations by accounting for reduced strength in concrete that fails due to compression.

For this project the smeared models were of particular interest since they can provide a relatively quick and flexible way of modelling the concrete. Barbosa and Ribeiro concluded that *the best results have come from the elasoplastic models, perfectly plastic and with work hardening, which have been able to reach ultimate loads very close to expected values, especially the model that followed a multilinear stress-strain relation in concrete compression.*

They also concluded that in their model, there was no significant difference between the discrete and smeared models, and that combining crushing and plasticity gave early convergence problems. ANSYS also presented difficulties in finding in the actual ultimate loads because analyses stopped due to lack of convergence. This effect was also observed during this thesis. They argued that the highest analysed loads could be considered the ultimate loads.

So, in summary

- Non-linear stress-strain relations were necessary
- Discrete and smeared reinforcement are both adequate
- Combining crushing and plasticity gives convergence issues
- Analysis stops due to lack of convergence, which can be interpreted as the point of ultimate loading

Figure 6.3
Shear stress-strain curve for the Drucker-Prager criterion
ANSYS 11.0 User Manual (2008)

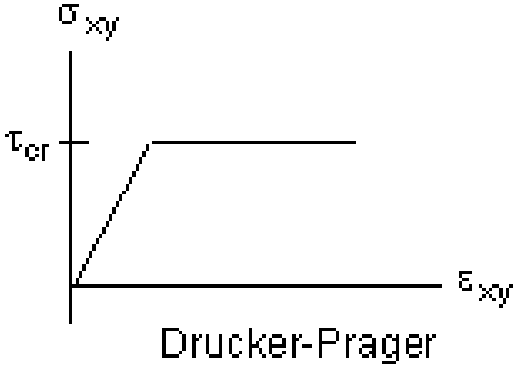
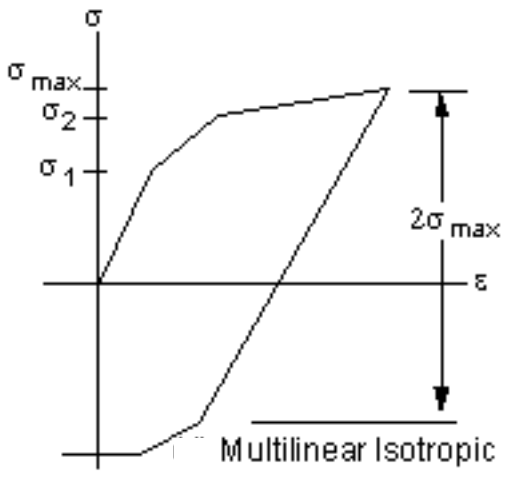


Figure 6.4
Typical stress-strain curve for multilinear isotropic material modelling
ANSYS 11.0 User Manual (2008)



```
ET,1,SOLID65
KEYOPT,1,7,1

MP,PRXY,1,.25
MP,EX,1,2.8484848484848484E10
MP,DENS,1,2400

MP,EX,2,2.0E11
MP,PRXY,2,0.25
MP,DENS,2,7800

! STRESS-STRAIN for CONCRETE
TB,MISO,1,1,2,
TBTEMP,0
TBPT,,9.9E-4,2.82E7
TBPT,,0.0035,2.82E7

! FAILURE ENVELOPE FOR CONCRETE
TB,CONCR,1,1,,1
TBTEMP,0
TBDATA,,1,1,2980000.0,-1,,
TBDATA,,,,1,,,

! STRESS-STRAIN for STEEL
TB,MISO,2,1,2,
TBTEMP,0
TBPT,,0.0025,5.0E8
TBPT,,0.0325,5.0E8
```

SOLID65 concrete elements:

Poisson ratio	ν	= 0,25
Young modulus	E	= 28.485 N/mm^2
Density	ρ	= 2400 kg/m^3
plastic strain	$\varepsilon_{c,pl}$	= 0,99 ‰
ultimate strain	$\varepsilon_{c,u}$	= 3,50 ‰
ultimate stress	$\sigma_{c,u}$	= 28,2 N/mm^2
cracking stress	σ_{cr}	= 2,98 N/mm^2

Reinforcement steel:

Poisson ratio	ν	= 0,25
Young modulus	E	= 200.000 N/mm^2
Density	ρ	= 7800 kg/m^3
plastic strain	$\varepsilon_{s,pl}$	= 2,50 ‰
ultimate strain	$\varepsilon_{s,u}$	= 32,50 ‰
ultimate stress	$\sigma_{s,u}$	= 500 N/mm^2

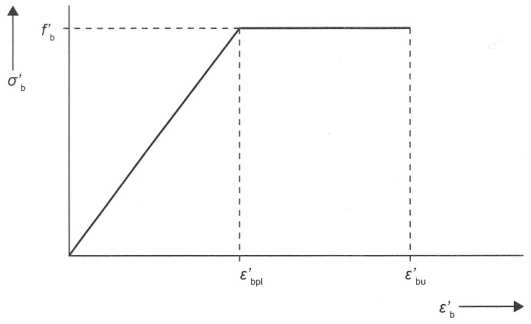


Figure 6.5
Typical compressive stress-strain curve for concrete in Dutch engineering
Soons et al. (2001)

6.2 Applied ANSYS material model

Using the conclusions of the previous paragraph provided a basis for a model, but a bilinear stress-strain curve is used instead for both the concrete and steel modelling as is customary in Dutch engineering codes. Figure 6.5 shows a typical bi-linear stress-strain curve, where f'_b is the ultimate compressive stress of concrete, ε'_{bpl} is the plastic strain and ε'_{bu} is the ultimate strain. (The b stands for beton i.e. Dutch for concrete c)

Initial models used concrete B25, which is a generally low strength mix. This concrete was chosen to verify ANSYS using existing hand calculations for a B25 concrete beam as a basis for comparison (see Section 6.4). For the actual fabric formed beams the choice was made to model B65 concrete which is, at the moment, a typical mix for prefabricated structural elements.

The block of ANSYS code on the left shows the modelling of the concrete and steel. Material 1 refers to concrete, material 2 to the reinforcement steel. Note that the units in ANSYS are typically entered in N and m. The box in the corner below translates this code into material properties. Some parameters in the code correspond to these statements used in the analysis.

- There is complete shear transfer around both open and closed cracks.
- Concrete crushing is disabled.
- Tensile stress relaxation after cracking is included.

The bi-linear modelling that ANSYS offers, referred to as BISO, does not allow a horizontal branch (Figure 6.5) in the stress-strain curve (this problem did not occur in version 10.0). To circumvent this, initial concrete models used a factor 1,01 for the ultimate stress at ultimate strain. Later, the multi-linear model, MISO, was used to model the bi-linear curve as it became apparent that MISO does allow horizontal branches in the curve.

6.3 Applying smeared reinforcement to fabric formed beams

The smeared reinforcement model provided by ANSYS is defined by three parameters, the reinforcement ratio and the reinforcement orientation, defined by angles θ_i and φ_i. (Figure 6.6)

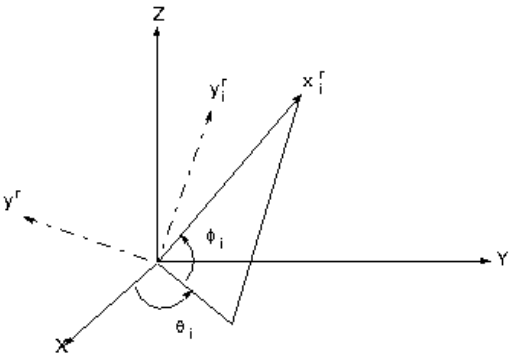


Figure 6.6
Two angles determine the orientation of reinforcement in the global system of coordinates
ANSYS 11.0 User Manual (2008)

The problem posed by the fabric formed beams is how to model the reinforcement in a way that it can cope with the wide range of possible shapes. Three assumptions were made to this end;

First, the beam is reinforced with prestressing tendons. They are not used as prestressing steel, but rather as reinforcement steel. The tendons are flexible enough to follow the shape of the beam and maintain an equal amount of concrete cover along the length. The high ultimate stresses that prestressing steel allows drastically change the amount of steel that is required and resulting reinforcement ratios will no longer conform to customary values.

The second assumption is to calculate the reinforcement ratio at the middle of the beam using an approximation method proposed in Walraven (2003). (see Section 6.3.1)

The third assumption is to model this tendon in discrete sections. In each section the reinforcement ratio and orientation are calculated. (see Section 6.3.2)
The use of smeared reinforcement instead of discrete reinforcement is particularly appropriate in this respect. Modelling a curved reinforcement bar, or tendon, using the discrete model would be troublesome as two-dimensional elements would then be used to model a smooth, continuous curve. This would require either a very specifically tailored mesh, designed in advance, or a relatively fine mesh. Generally speaking, in this case, where flexibility is important, there is no longer a benefit to using the discrete representation.

In summary, the concrete model assumed the following:

- reinforcement consists of prestressing steel
- the reinforcement ratio is calculated at midspan using an approximation method
- smeared reinforcement is modelled in discrete sections

6.3.1 Approximation method for non-rectangular cross sections

For non-rectangular cross sections, one can assume an alternative stress-strain diagram using a fully plastic stress distribution to avoid complicated calculations.
The method uses two assumptions, i.e. reductions, to correct large deviations from more accurately calculated results.

These are:

- the ultimate stress is reduced to $0,95\,f'_c$
- the height of the compression zone x is reduced to $0,80x$

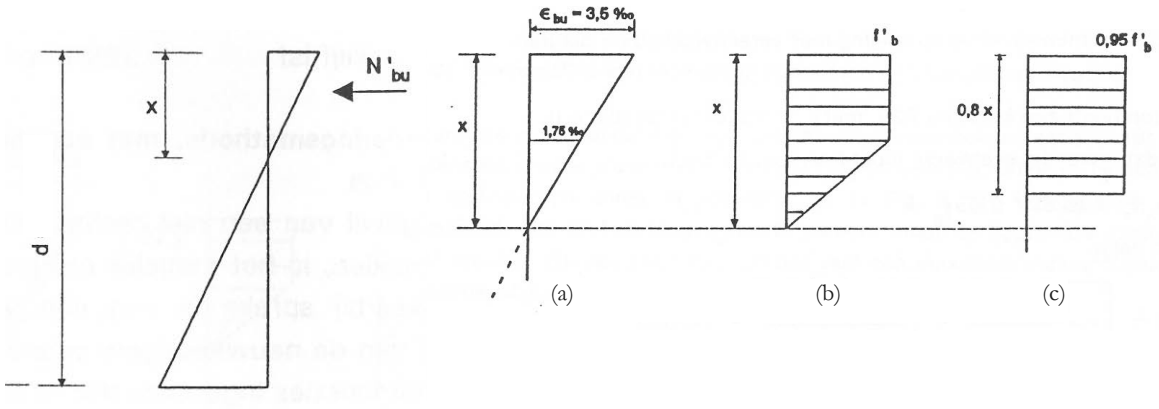


Figure 6.7
Strain diagram for a cross-section with a visual description of the approximation method in (c)
ANSYS 11.0 User Manual (2008)

Using these assumptions the compression force, equal to the area of the stress diagram (Figure 6.7c), is calculated:

$$N_{cu}^{'} = 0,8 \cdot 0,95 f_c^{'} \cdot bx = 0,76 bx f_c^{'} \approx 0,75 bx f_c^{'} \tag{6.1}$$

The calculated factor $0,76$ is approximately the conventional value of $0,75$ (Figure 6.7b,c). In addition, the location where the force acts is at $0,4x$ from the top compared to the conventional value of $0,39x$ for a rectangular cross-section.

For equilibrium the concrete in compression has to be equal to the steel in tension.

$$A_s f_s = \left(0,8x\right)^2 \cdot 0,95 f_c^{'} \tag{6.2}$$

The amount of steel is:

$$A_s = x^2 \cdot 0,64 \cdot \frac{0,95 f_c^{'}}{f_s} \tag{6.3}$$

The ratio between the outer strains is proportional to the distances x and d-x to the neutral axis (Figure 6.7).

$$\varepsilon_s = \frac{d-x}{x} \cdot \varepsilon_{cu}^{'} \rightarrow x = d \cdot \frac{\varepsilon_{cu}^{'}}{\varepsilon_s + \varepsilon_{cu}^{'}} = 0,90h \cdot \frac{\varepsilon_{cu}^{'}}{\varepsilon_s + \varepsilon_{cu}^{'}} \tag{6.4}$$

where the height d is approximately equal to $0,9h$, so that the concrete cover is $0,1h$.
Combining (6.3) and (6.4) gives an upper limit for the steel surface area

$$A_s = 0,64 \cdot 0,81 h^2 \cdot \left(\frac{\varepsilon_{cu}^{'}}{\varepsilon_s + \varepsilon_{cu}^{'}} \right)^2 \cdot \frac{0,95 f_c^{'}}{f_s} \tag{6.5}$$

Equation (6.5) can now be solved in ANSYS. The questions remains whether this approximation method is adequate for use with fabric formed beams as well as for flexible optimization algorithms. It is certain that the optimization process will apply this method to various impractical solutions during optimization. For this reason it is decided to only assess the validity of the reinforcement properties for optimal solutions that result from the FABRICFORMER program. This is a practical approach, since these optimal results would be the only ones that actually have to be manufactured, and thus reinforced.
With respect to the non-prismatic nature of the fabric formed beams, it can be argued that although the approximation method only discusses non-rectangular prismatic beams, the method is still applicable. The main reason is that for assessment of longitudinal reinforcement, the single cross-section at midspan remains governing as this is still the location of maximum bending and minimum shear forces. There is no foreseeable reason why the amount of reinforcement calculated at midspan should be inadequate elsewhere along the beam, as long as the beam itself has a reasonable geometry.

6.3.2 Smeared longitudinal reinforcement in discrete sections

Typical examples using smeared reinforcement models are prismatic beams. These models would only have to assign the three reinforcement parameters (volume ratio and two angles) to a single, longitudinal volume (as in Figure 6.2). In a non-prismatic, curved beam this is no longer possible as the reinforcement ratio and the local orientation change along the length. To cope with this, the beam volume is divided into sections, each of which has its locally determined parameters for the reinforcement.

Figure 6.8a shows an example of a curved beam (by no means an optimal geometry) with a proposed location for the longitudinal reinforcement. The reinforcement tendon is place at a certain distance from the bottom i.e. the concrete cover c. This cover is typically between 20 and 50 m, and is often approximated as one tenth of the beam height i.e. $c = 0,10h$ during initial design calculations.

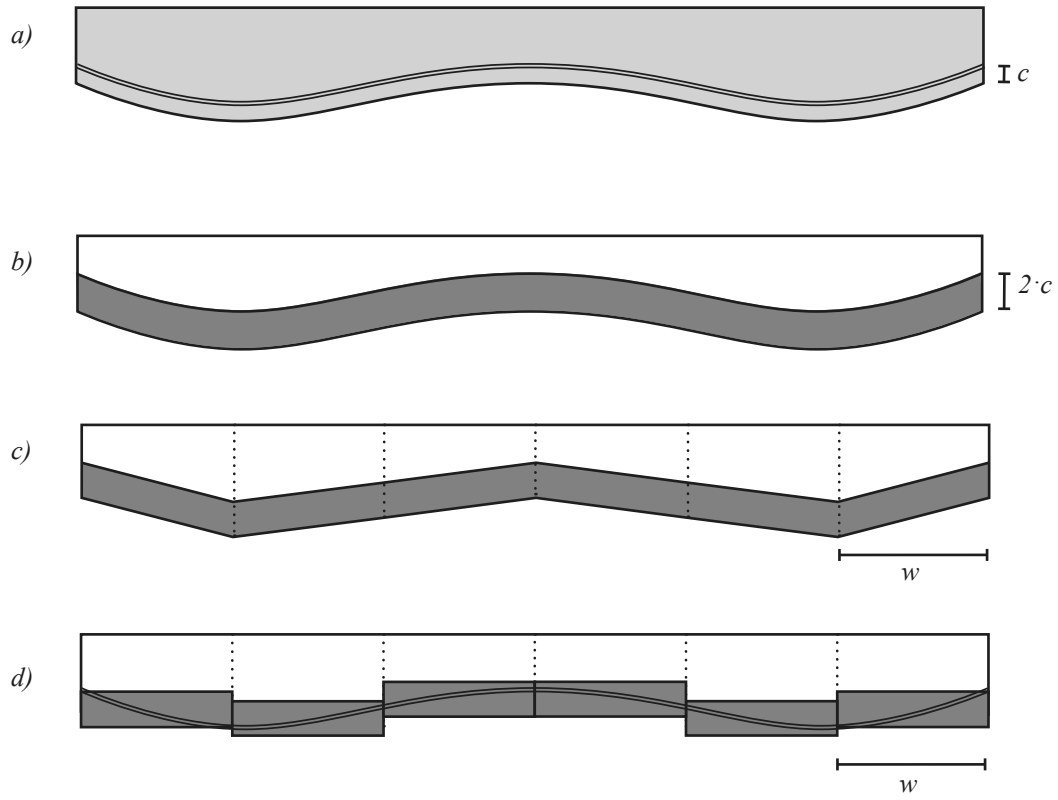


Figure 6.8
Smeared reinforcement, applied in discrete sections, to model the actual curved reinforcement

For the smeared reinforcement, the reinforced volume is assumed to be twice the concrete cover in height i.e. $2c = 0{,}20h$, which assumes that the tendon diameter is included. In this manner, the location where the resultant tensile force acts should be close to the actual center of gravity of the steel. With smeared reinforcement, the model now looks like Figure 6.8b.
The beam is now divided in a discrete number of sections. For six sections, the division resembles Figure 6.8c, where w is the section length determined by user input. In ANSYS the selected areas, that are either reinforced or not, have to be rectangular along global axes, so the model is changed to Figure 6.8d.
For each division the volume ratio of the reinforcement is calculated using the cross-sectional area that ANSYS provided and the steel area according to Equation (6.5). The orientation angles are calculated as the slope of the division according to Figure 6.8c. All parameters are then assigned in ANSYS and non-linear analysis of the reinforced beam is now possible.
When comparing the actual path of the reinforcement steel in Figure 6.8d with the final model it is obvious that the approximation deviates significantly at several points. However the example given is much more coarse than the actual calculation carried out by FABRICFORMER. The number of divisions is much higher so that the smeared model resembles Figure 6.8b much more closely than that of 6.8d.

6.4 Verification of the ANSYS model

Due to the use of a bi-linear stress-strain curve, the ANSYS model was compared to hand calculations. For this comparison hand calculations on a rectangular reinforced concrete beam were carried out and the same dimensions were used with the aforementioned concrete model. These hand calculations will be discussed first before going into the ANSYS model and subsequent data comparison.

The following calculations are taken from the reader for the course CT2051b/3051b at the faculty of Civil Engineering at this university. It calculates the critical moments up to failure of a reinforced concrete beam subjected to two point loads. The results are graphed in a moment-curvature diagram. The equations and values are exactly the same as in the reader, while some – for this case – irrelevant calculations are omitted. The same input was used to verify the calculations in ANSYS. This was done because there was no experience at this faculty with the use of ANSYS for reinforced concrete and a certain confidence in, and familiarity with ANSYS was required for its applied use with fabric formed beams.

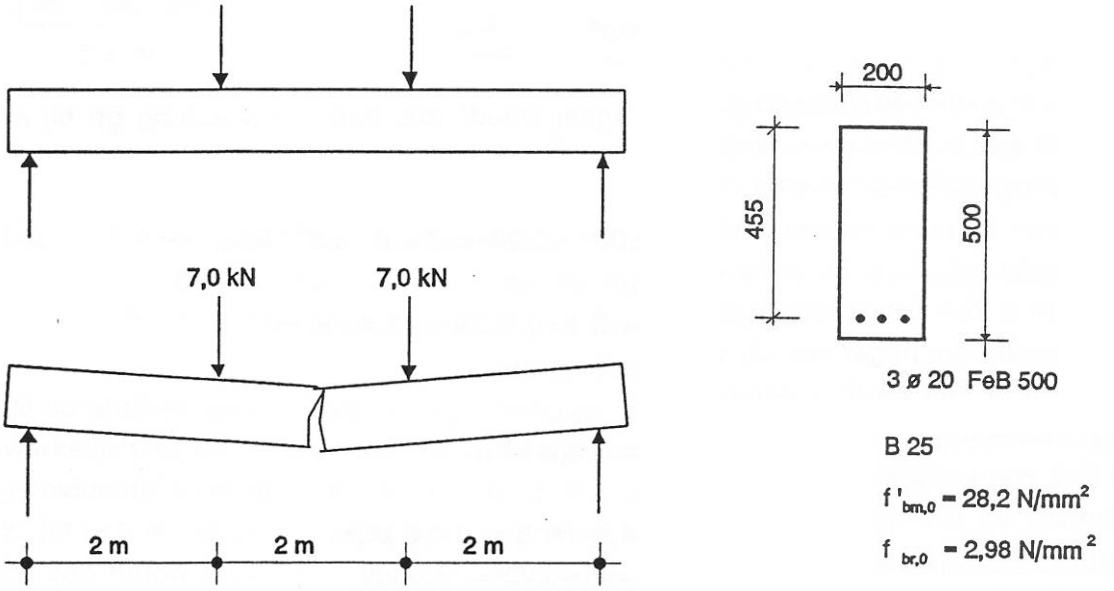


Figure 6.9
Reinforced beam on two support with two point loads, including the dimensions of the cross section and the reinforcement. *Walraven (2004)*

Figure 6.9 illustrates the loading and support conditions of the beam as well as the dimension of the cross-section and the longitudinal reinforcement.
The first critical moment occurs at cracking of the concrete tensile zone. The moment at cracking M_r is

$$M_r = W \cdot f_{cr,0} = 1/6 \cdot 200 \cdot 500^2 \cdot 2,98 = 24,8 \cdot 10^6 \text{ Nmm}$$

where the bending tensile strength is $2,98$ N/mm^2 for B25 concrete. The strain is

$$\varepsilon_{\text{cr}} = f_{cr,0} / E_c = 2,98/28500 = 0,1050$$

The linearly derived curvature κ_r is

$$\kappa_{\text{r}} = \frac{\varepsilon_1 + |\varepsilon_2|}{h} = \frac{2 \cdot 0,105 \cdot 10^{-3}}{500} = 0,4 \cdot 10^{-6} \text{ mm}$$

At this point the reinforcement steel comes into play, its contribution up to this point is considered negligible. Figure 6.10 shows the plastic and ultimate strains of the concrete and steel. Now that the concrete has cracked, the reinforcement steel will take over. The new situation directly after cracking is calculated.

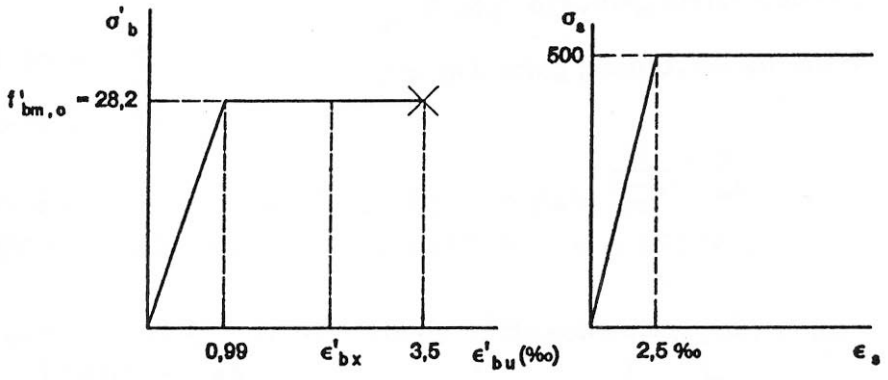


Figure 6.10
Bi-linear stress-strain curves for concrete and steel as used in the calculation example. The strains are given for when the material exhibits plastic behaviour. *Walraven (2004)*

With respect to deformation, it follows that

$$\frac{\varepsilon_c'}{\varepsilon_s} = \frac{x}{d - x} \tag{6.6}$$

Horizontal equilibrium has to be satisfied,

$$\begin{aligned} N_c^{'} &= N_s \\ \tfrac{1}{2} \cdot b \cdot x \cdot E_c \varepsilon_c^{'} &= A_s \cdot E_s \varepsilon_s \end{aligned} \tag{6.7}$$

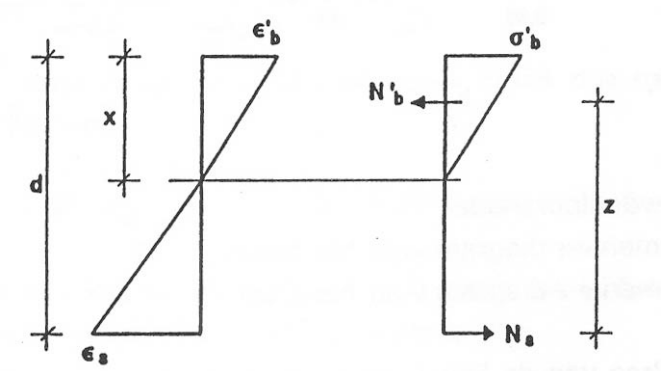


Figure 6.11
Strain and stress diagram for the beam cross-section prior to reaching the plasticity limit.
Walraven (2004)

and the moment complies with

$$\begin{aligned} N_s \cdot z &= M_r \\ A_s \cdot E_s \varepsilon_s \left(d - x/3\right) &= M_r \end{aligned} \tag{6.8}$$

Combining Equations (6.7) and (6.8) gives the following

$$\frac{x}{d} = -n\omega + \sqrt{\left(n\omega\right)^2 + 2n\omega} \tag{6.9}$$

$$\text{where } \omega = \frac{A_s}{bd} \text{ and } n = \frac{E_s}{E_c^{'}}$$

In this case, $d = 455$ mm, $\omega = 942/200 \cdot 455 = 1{,}04‰$ and $n = 200.000/28.500 = 7$, so that with (6.9) it is found that $x = 143$ mm. From (6.8) it is known that steel strain $\varepsilon_s = 0{,}38‰$ and from (6.6) that the concrete strain is $0{,}177‰$ so that the curvature is

$$\kappa_r = \frac{\left(0{,}38 + 0{,}177\right) \cdot 10^{-3}}{455} = 1{,}24 \cdot 10^{-6} \text{ mm}^{-1}$$

The next critical moment occurs when the concrete reaches its plastic strain limit, $\varepsilon_{cpl} = 0{,}99‰$. From (6.6) we know that the corresponding steel strain $\varepsilon_s = 2{,}15‰$. This is indeed lower than the plastic strain limit for steel, so that the concrete displays plastic behaviour first.
In this situation, the steel tensile force is

$$N_s = A_s \cdot E_s \varepsilon_s = 942 \cdot 2{,}15 \cdot 10^{-3} \cdot 200000 = 405 \text{ kN}$$

and the corresponding moment is

$$M_{cpl} = N_s \left(d - 1/3\, x\right) = 405 \cdot 10^3 \cdot \left(455 - 143/3\right) = 165 \cdot 10^6 \text{ Nmm}$$

and the curvature is

$$\kappa_{cpl} = \frac{\left|\varepsilon_{cpl}^{'}\right| + \varepsilon_s}{d} = \frac{\left(0{,}99 + 2{,}15\right) \cdot 10^{-3}}{455} = 69 \cdot 10^6 \text{ mm}^{-}$$

Next, the steel will reach its plastic strain limit. The steel tensile force is

$$N_{spl} = A_s \cdot f_{s,rep} = 942 \cdot 500 = 471 \cdot 10^3 \text{ N}$$

The concrete compressive force is calculated according to Figure 6.11.

$$\begin{aligned} N_c' &= N_{c1}' + N_{c2}' \\ &= ½ \cdot \frac{0,99}{\varepsilon_c'} \cdot x \cdot b \cdot f_c' + \left(1 - \frac{0,99}{\varepsilon_c'}\right) \cdot x \cdot b \cdot f_c' \end{aligned} \tag{6.10}$$

Using (6.6) in (6.10) to substitute the concrete strain ε_c' it is found that

$$\begin{aligned} N_c' &= ½ \cdot \frac{0,99}{\varepsilon_{spl}} \cdot (d - x) \cdot b \cdot f_c' + \left(x - \frac{0,99 \cdot (d - x)}{\varepsilon_{spl}}\right) \cdot b \cdot f_c' \\ &= 1117(455 - x) + \{x - 0,396(455 - x)\} \cdot 5640 \\ &= 6756x - 508 \cdot 10^3 \end{aligned}$$

And using horizontal equilibrium x is found.

$$\begin{aligned} N_{spl} &= N_c' \\ 471 \cdot 10^3 &= 6756x - 508 \cdot 10^3 \\ x &= 145 \text{ mm} \end{aligned}$$

The moment when the steel becomes plastic is

$$M_{spl} = 346,3 \cdot 0,392 + 124,7 \cdot 0,444 = 191 \text{ kNm}$$

while the curvature is

$$\kappa_{spl} = \frac{\left|\varepsilon_c'\right| + \varepsilon_s}{d} = \frac{(1,168 + 2,5) \cdot 10^{-3}}{455} = 8,06 \cdot 10^{-6}$$

The final critical moment is at total fracture when the top fiber of the concrete reaches ultimate strain and fails due to crushing. Again, equilibrium must hold, as the concrete reaches ultimate strain ε_{cu}' = *3,5‰* .

$$\begin{aligned} N_c' &= N_{spl} \\ N_{c1}' &= \frac{1}{2} \cdot \frac{0,99}{3,5} \cdot x \cdot b \cdot f_c' = 798x \\ N_{c2}' &= \frac{2,51}{3,5} \cdot x \cdot b \cdot f_c' = 4045x \\ N_c' &= 4843x = N_{spl} = 471 \text{ kN} \end{aligned}$$

so that,

$$x = \frac{471 \cdot 10^3}{4843} = 97 \text{ mm}$$

and the steel strain is

$$\varepsilon_s = \frac{d-x}{x} \cdot 3,5 = 12,90$$

As shown in Figure 6.12, it is possible to determine z_1 using Equation (6.6) and using x as the distance from the neutral axis to the point where the concrete strain $\varepsilon_c' = 0,99‰$. As their must be equilibrium, z_2 can be calculated as well. Here, the distances z_1 and z_2 are *392* and *444* mmm respectively.

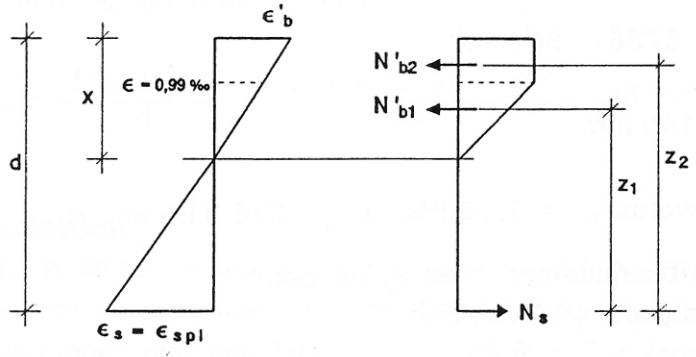


Figure 6.12
Strain and stress diagram for the beam cross-section where some concrete is fully plastic in behaviour.
Walraven (2004)

$$M_u = N_{c1}^{'} \cdot z_1 + N_{c2}^{'} + z_2 = 0,376 \cdot 77,5 + 0,420 \cdot 392,5 = 194 \text{ kNm}$$

The curvature at failure is

$$\kappa_u = \frac{\varepsilon_{cu}^{'} + \varepsilon_s}{d} = \frac{(3,5+12,9) \cdot 10^{-3}}{455} = 36 \cdot 10^{-6}$$

The results from hand calculations can be compared to results from non-linear analysis in ANSYS. The relation between moments and curvature is shown in the diagram in Figure 6.13. The diagram demonstrates that the ANSYS non-linear model compares well to manual calculations in an overall sense. Two clear differences are at first cracking (Figure 6.13a) and at ultimate failure (6.13c), where the ANSYS model seems to have some kind of additional reserve.

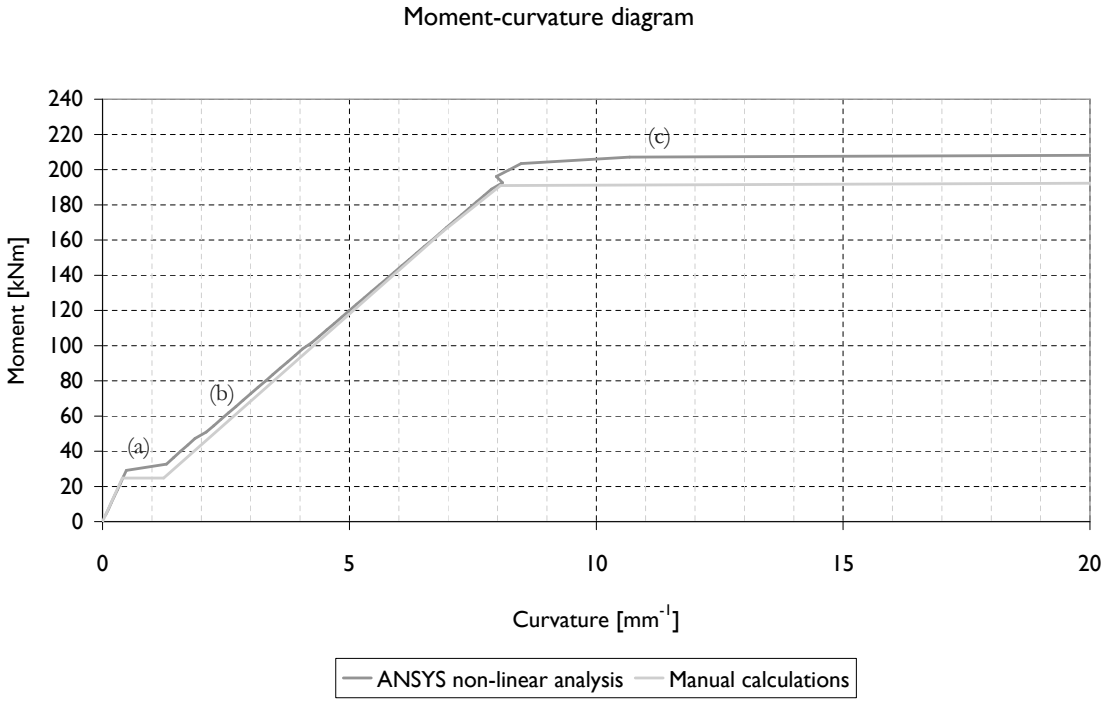


Figure 6.13
Moment-curvature diagrams for hand calculated and ANSYS computed reinforced concrete beam.

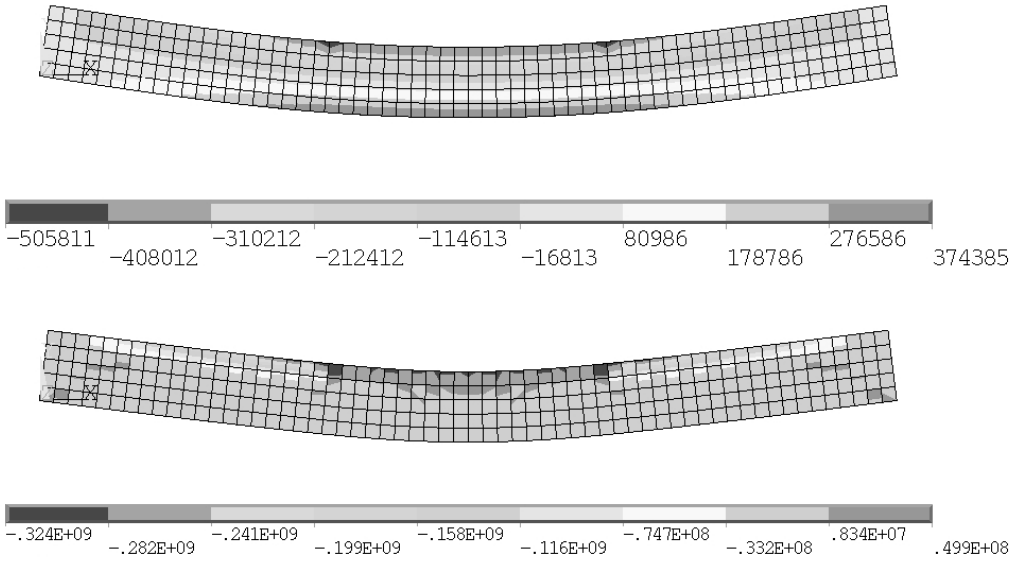


Figure 6.14
Two ANSYS plots of the stress in X-direction in N/m^2 at approximately 36 and 211 kNm.

The differences between both models can be accounted for. The hand calculations do not take into account the contributions of the reinforcement steel prior to cracking, whereas ANSYS non-linear analysis does (Figure 6.13a). The difference in behavior after first cracking (Figure 6.13b) is attributed to the effect of tension stiffening i.e. the concrete contributes some post-cracking stiffness and is not completely factored out. The higher ultimate capacity that the ANSYS model exhibits (Figure 6.13c) is also due to some degree of stiffness that the cracked concrete still possesses. The difference between both lines is about 16 kNm.

Another difference, with less influence on the results is the distance d to the centre of mass of the reinforcement steel. Because the meshing of the beam is confined to 50 mm elements, and smeared reinforcement is applied to whole elements, the centre of mass for the steel is at 450 mm from the top. In other words, d = 450 mm whereas the hand calculated example uses 455 mm. This means that the hand calculated example has a slightly higher arm and is able to resist larger moments.

All in all, the ANSYS concrete model is comparable to hand calculated models, which demonstrated its ability to accurately model concrete material behaviour in non-linear analysis. This gave sufficient confidence in utilizing ANSYS for this type of analysis.

6.5 Finite element mesh

For any calculation in finite elements, some type of element mesh is required. There are generally two kinds of meshes; structured and unstructured meshes. The ANSYS User Manual prefers to use the terms free or mapped meshing versus unstructured or structured meshes. It defines a mapped mesh as one that is 'restricted in terms of the element shape it contains and the pattern of the mesh' while a free mesh has 'no restrictions in terms of element shapes, and has no specific pattern applied to it. Figure 6.15 shows the difference between these two types of meshes.

The cubic mesh used in BESO in Chapter 2 is repetitive i.e. a structured mesh. Initially, such a mesh was thought to be less preferable for use in FABRICFORMER. The

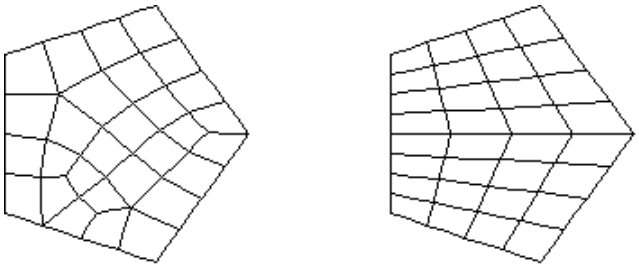


Figure 6.15
An example unstructured and structured mesh, or free and mapped mesh. *ANSYS 11.0 User Manual (2008)*

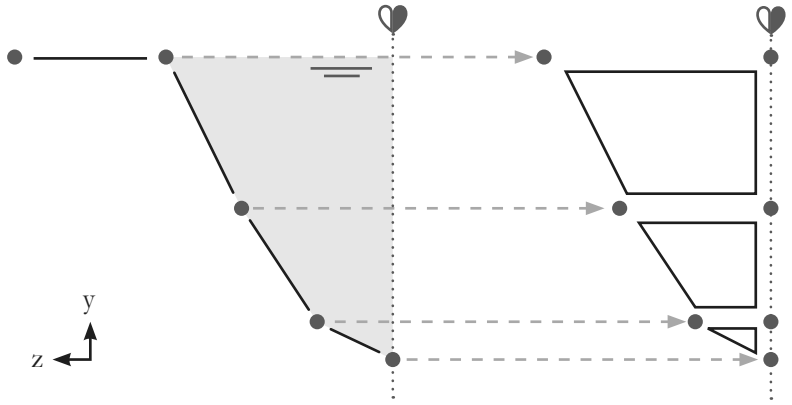


Figure 6.16
Translating the DR mesh to extruded volumes extending to the plane of symmetry. These are subsequently meshed using an ANSYS free mesher.

optimization process of FABRICFORMER continuously presents new and different geometries, each of which have to be analyzed in finite elements. To accurately mesh these complex geometries with many double-curved faces, some kind of unstructured mesh was needed. ANSYS provides two kinds of automated unstructured meshing algorithms; one based on Delaunay Triangulation, one on the Advancing Front Technique.

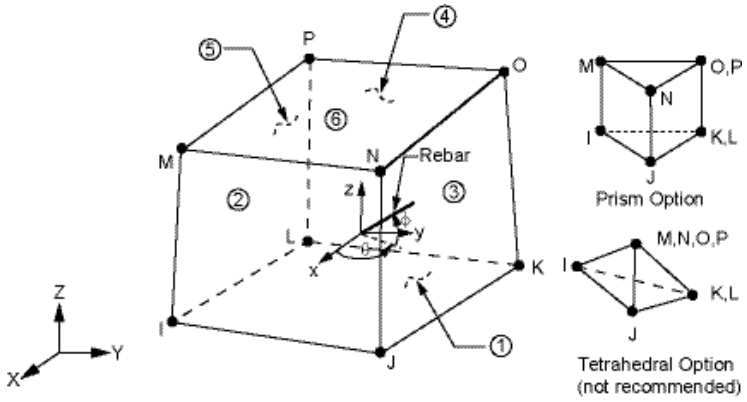


Figure 6.17
SOLID65 Element, also in prism and tetrahedral form. *ANSYS 11.0 User Manual (2008).*

Transferring the planar mesh from DR to the volume mesh in ANSYS proved more troublesome than initially expected. At first, the logical strategy was to directly copy the coordinates of all nodes that were in contact with the concrete and add an additional node at the plane of symmetry to extrude the volume (Figure 6.16). (Note that the use of 'extrusion' is not entirely accurate as the extruded plane is not parallel to its opposite face which is in the plane of symmetry) Then, the volumes were added together and subsequently meshed in tetrahedral elements using the free meshing algorithm based on Delaunay Triangulation. Both free meshing algorithms that ANSYS offers use some form of triangulation, so it is a requirement that all elements are triangular plane or tetrahedral volume elements. Figure 6.17 shows the typical SOLID65 element and its tetrahedral form.

It soon turned out that all resulting meshes feature a significant percentage of poorly shaped elements. Although ANSYS provides several types of mesh improvements, poorly shaped elements could never be avoided. The alternative meshing algorithm based on the Advancing Front Technique was slower and also failed to mesh the volume in many instances, ultimately proving even more unreliable. The poorly shaped elements distort the analysis results and therefore the total optimization process. At this stage the amount of such elements sometimes added up to a quarter of the entire mesh. Choosing a finer mesh decreased this share but also led to unacceptably large numbers of elements.

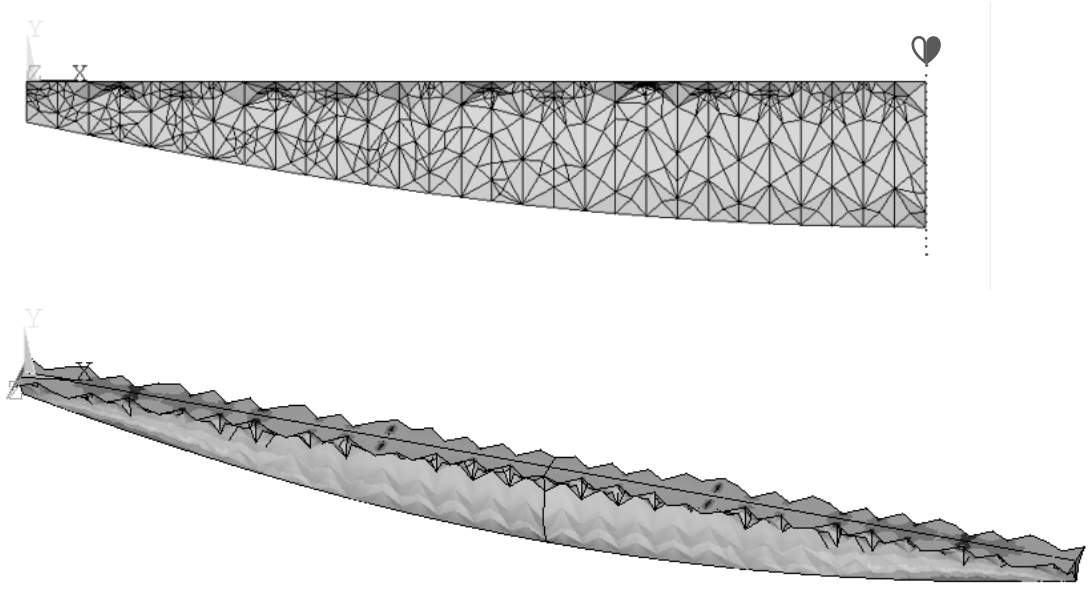


Figure 6.18
Example of a free meshed fabric formed beam. The x-coordinates of the nodes have been averaged resulting in parallel slices in the yz-plane, prior to meshing.

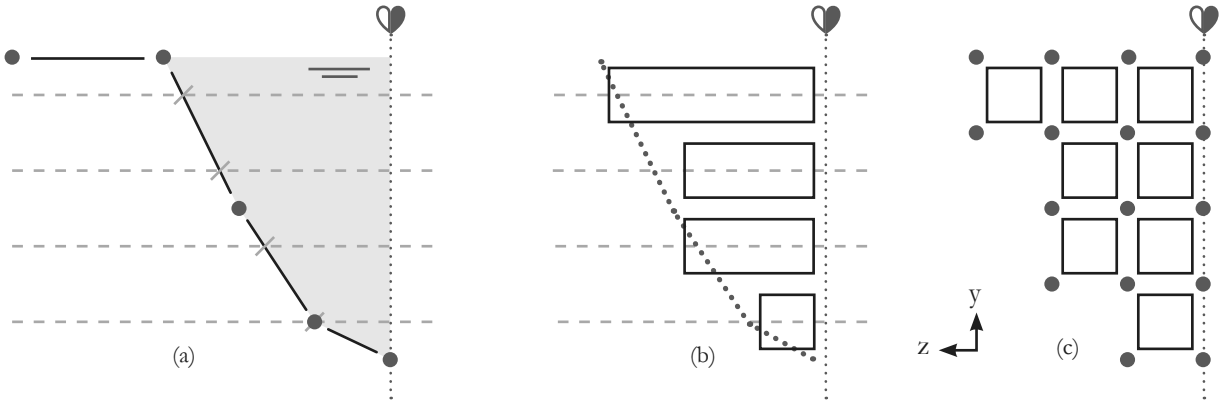


Figure 6.19
Basic steps in translating the DR mesh to a structured block mesh, similar to the one typically used in BESO.

The first solution was to introduce a degree of structuring to the DR mesh. The nodes of the fabric mesh that are parallel to the cross-section of the beam, i.e. in yz-plane, are averaged. The mesh now effectively features parallel slices, reducing the difficulty with which ANSYS has to mesh the entire volume. This indeed results in relatively smaller numbers of elements and also reduces the amount of poorly shaped elements to several percent. Figure 6.18 shows what such a mesh looks like. Notice that several nodes are aligned along vertical lines; these are the averaged x-coordinates of the sets of nodes perpendicular to the longitudinal axis.

This method proved to be more than adequate for the optimization of keel mold beams and ensured a certain acceptable balance between computational time and accuracy. However, applying this method to pinch mold beams is likely less acceptable, since the amount of poorly shaped elements is related to curved features and sharp angles. These types of features only increase in number in pinch molds compared to the keel mold. Due to the fact that the amount of poorly shaped elements as well as the total number of elements would again prove a problem, an alternate strategy was devised.

Recalling the structured, cubic mesh in BESO, and its accompanying lack of mesh-related problems in ANSYS, the decision was made to revert to this type of mesh.

The translation of the DR mesh to ANSYS was rewritten to accommodate this. In the new situation, FABRICFORMER applies a grid to the xy-plane. The size of the grid squares corresponds to the preferred element size. At each grid point, along a line in z-direction, the program checks whether any triangular elements of the fabric mesh intersects with this line (Figure 6.19a). The distance between this intersection and the plane of symmetry spans the concrete volume and so determines the new volume block that is input in ANSYS (6.19b). The length of this block is rounded to accommodate the element size. The volumes are added together and meshed (6.19c).

Should the line intersect with multiple triangular elements, then the line is crossing both sections of air and of concrete. FABRICFORMER accounts for this and changes the volume block accordingly (Figure 6.21).

The resulting mesh consists of volumes that can be mes-

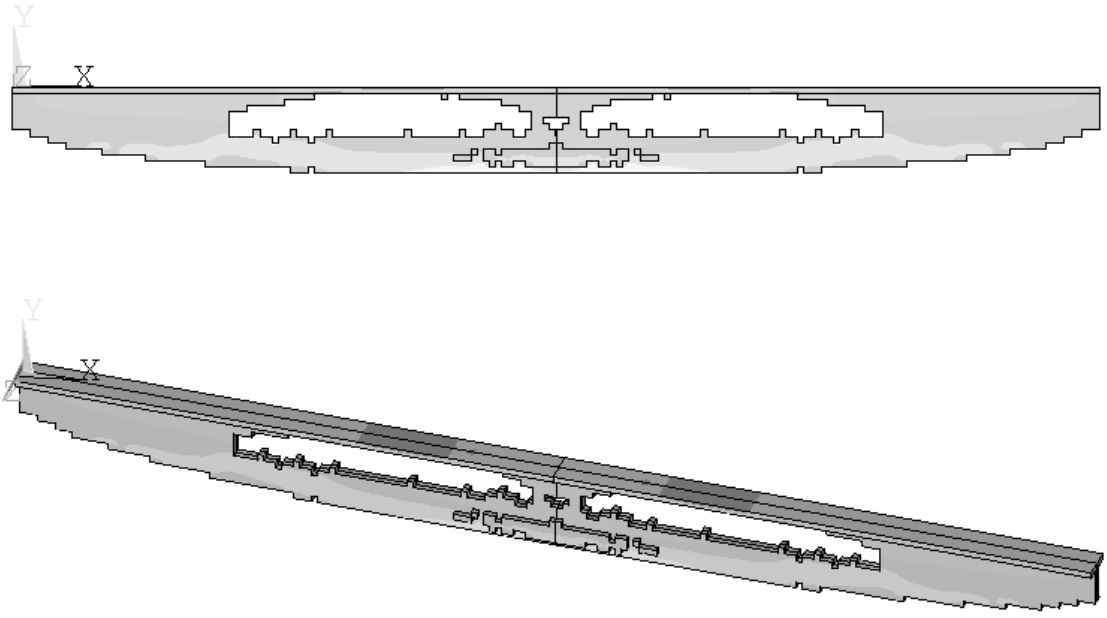


Figure 6.20
Example of the block meshed fabric formed beam. The DR mesh is replaced by cubic elements to improve element quality and speed up the calculations.

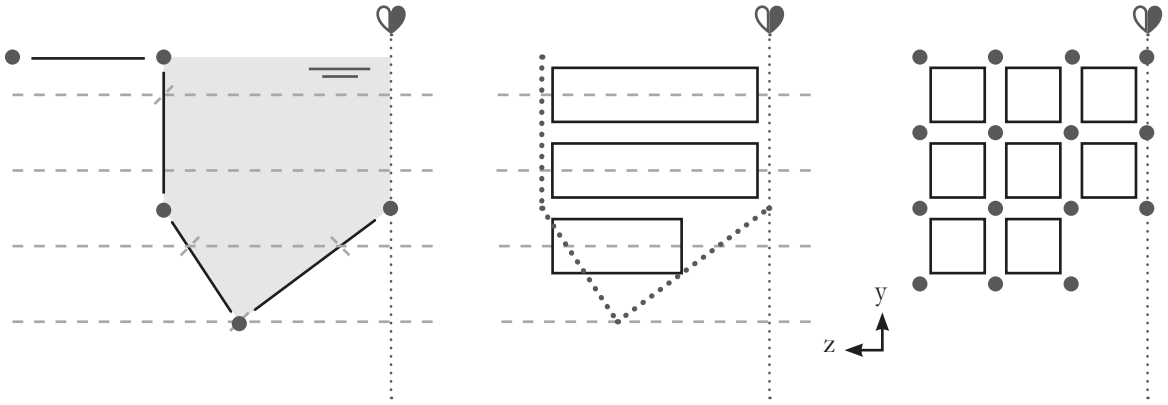


Figure 6.21
The DR mesh curves inward at one point. FABRICFORMER detect multiple intersections at one grid point and takes this into account

hed with an exact number of cubic elements and analyzed in the same fashion as BESO in Chapter 2 (Figure 6.20). One final change was made by generating an entire volume domain first, meshing it with cubes and then specifying whether each element was DEAD or LIVE, the same as in BESO. It turned out that this approach was computationally faster.

6.6 Performance Index for non-linear analysis

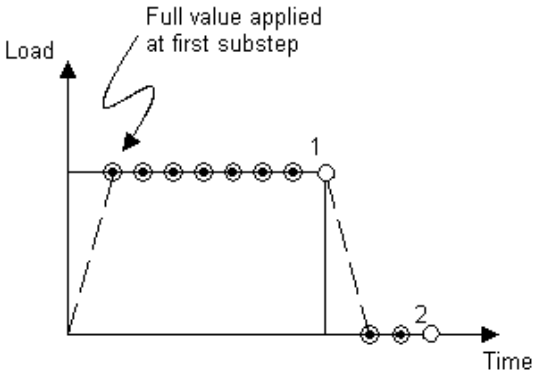
For linear analysis the performance index as used for the BESO algorithm is perfectly applicable. The formula for this performance index is recalled:

$$PI = \frac{1}{E_s^{'} \cdot W} = \frac{1}{SENE \cdot DENS \cdot VOLU} \tag{2.23}$$

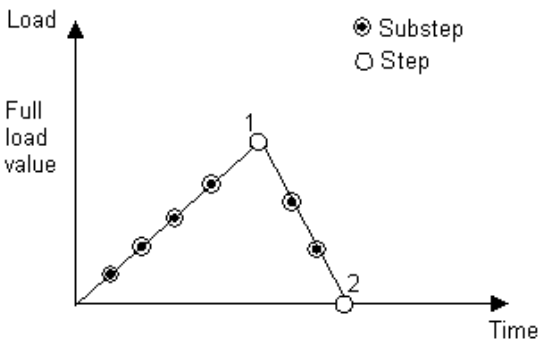
The performance index is dependant on the value of the applied loads. The higher the loads, the higher the performance index. Because the analysis is linear during optimization, the absolute value of these loads becomes unimportant, as they do not change how the various solutions perform relative to one another.
However, in non-linear analysis, the value of the load is much more important. For example, under a small load, one beam might be superior to the other in performance, but as it might turn out, it cracks earlier and behaves plasticly at an earlier stage. From zero to ultimate loading, the other beam is the better one.
To deal with this possibility, it was decided to change the way in which the performance index is calculated. The load is applied in increments (which is what ANSYS does in non-linear analysis) and the performance index is then calculated at each stage. Once the beam reached ultimate failure and breaks, there is a complete overview of how it behaved over time. By integrating (2.23) over time t, the new performance index for non-linear analysis is obtained. Note that $t=0$ corresponds to zero load and $t=T$ with ultimate load.

$$PI = \frac{1}{E_s^{'} \cdot \rho \cdot V} = \frac{1}{\int_t \int_V \vec{u}^T K \vec{u} dV dt \cdot \rho \cdot V} \tag{6.11}$$

Contrary to the BESO examples in Chapter 2, the density is relevant, because the reinforcement steel will locally alter the weight of the concrete. However, the density was still omitted from the actual FABRICFORMER program because the amount of steel used as well as its location remains relatively fixed and is not explicitly optimized (see Section 6.3.2).
An additional advantage is the fact that the optimal result does not need to be checked for ultimate loading and whether ultimate stresses have been surpassed. The beam has been optimized by using the actual material stress-strain curves and by analyzing it up to ultimate load. Therefore, the resulting optimal beam is the beam that has performed the stiffest under the total range of realistic loads.
One drawback remains the way in which ANSYS applies the load, which is by also incrementally applying self-weight. This means that the loads are not exactly realistic and that care has to be taken when choosing external loads.
For example: An external point load has a value which is much higher than the ultimate load. ANSYS finds the moment of failure at only one tenth of the specified point load. This means that the self-weight at this point is also only one tenth.



(a) Stepped loads



(b) Ramped loads

Figure 6.22
ANSYS can apply the full load at once, or incrementally, called stepped and ramped loading respectively. Using two load steps one could first apply the full gravtiy load before apllying external loads in order to get more relevant results.
ANSYS 11.0 User Manual (2008)

Despite this the beam may be rated with a high performance index, although the behaviour during loading no longer has any bearing to its real-world behaviour.

This problem was avoided by carefully choosing the loads to be slightly higher than the expected ultimate loads. Nonetheless, it is worthwhile to rewrite the code so that the self-weight it immediately applied first before the external loads are introduced and performance is calculated. In that case the value of the external loads can be specified more arbitrarily.
At a later stage, the ANSYSscript in the text box directly below was developed to accomplish this (see also Figure 6.22).

```
/SOLU
ESEL,ALL
NSEL,ALL

ANTYPE,0
ACEL,,9.81
TIME,1
AUTOTS,0
KBC,0
SSTIF,ON
SOLCONTROL,ON
NSUBST,1,1000,1
OUTRES,ALL,ALL
CUTCONTROL,NOITERPREDICT,1
CNVTOL,F, ,0.02,2,1,
SOLVE

ANTYPE,0
ACEL,,9.81
TIME,2
AUTOTS,0
KBC,1
SSTIF,ON
SOLCONTROL,ON
NSUBST,20,1000,20
OUTRES,ALL,ALL
CUTCONTROL,NOITERPREDICT,1
CNVTOL,F, ,0.02,2,1,
ASEL,S,LOC,Y,0
SFA,ALL,,PRES,15000
SBCTRAN
ESEL,ALL
NSEL,ALL
SOLVE
FINISH
```

In the text box below the ANSYSscript routine to calculate the new performance index according to (6.11) is given. Again, the density is not included in this code. Also note that the last loading step (MAXSUBST) should not be included, since at this point ANSYS failed to converge and no data was stored.

```
/POST1
SET,,1
ETABLE,VOLU,VOLU
SSUM
*GET,VOLU,SSUM,,ITEM,VOLU
*GET,MAXSUBST,ACTIVE,,SET,NSET,LAST,1
*DIM,PI_ARRAY,ARRAY,MAXSUBST
*DIM,TIME_ARRAY,ARRAY,MAXSUBST

*DO,STEP,1,MAXSUBST-1
   SET,,STEP
   ETABLE,SENE,SENE
   SSUM
   *GET,SENE,SSUM,,ITEM,SENE
   SSUM
   *SET,PI_ARRAY(STEP),1/(SENE*VOLU)
   *GET,TIME,ACTIVE,,SET,TIME
   *SET,TIME_ARRAY(STEP),TIME
*ENDDO

*VOPER,INT_PI,PI_ARRAY,INT1,TIME_ARRAY,0,1

*SET,PI,0
*DO,STEP,1,MAXSUBST-1
   *SET,PI,PI+INT_PI(STEP)
*ENDDO

FINISH
```

6.7 Conclusions and recommendations

This chapter has demonstrated that reasonably accurate modelling of concrete in ANSYS is possible in such a way that it can be flexibly applied throughout the optimization process. Several approximations and inaccuracies have been discussed, but in the end it is argued that these are relatively unimportant. The reason for this is that all inaccuracies are carried through in the optimization process in all generated solutions and comparison negates some of the effects that these inaccuracies might have. However, the final optimal solutions should be subjected to additional scrutiny and perhaps more detailed analysis. Such analysis might also provide insight in ways to improve the concrete modelling in FABRICFORMER.

More specifically, the smeared reinforcement approach shows excellent comparability with hand calculations and all deviations can be explained.

The approximation method by Walraven (2004) is easy to implement and results from Chapter 8 show that the resulting reinforcements are within a realistic range. However, other than these results, it has not been thoroughly shown that it is completely applicable to non-prismatic beams, and it is recommended to either verify this or else replace the method entirely. One obvious, but cumbersome alternative is to have ANSYS iterate the reinforcement properties by calculating the requirements prior to ultimate failure, applying them and re-analyzing the beam.

The meshing of the complex shaped fabric formed beams proved to be exceedingly problematic in ANSYS. ANSYS features two free meshing algorithms that cannot always cope with these geometries in such a way that analysis is reliable. Significant effort was undertaken to translate the mesh from DR to a mesh that ANSYS could handle, which was succesful to a degree, but difficult for pinch molded shapes.
Ultimately, the meshing method reverted to the use of cubic elements. It is recommended to search for a reliable free meshing algorithm or program, because a free mesh is theoretically the more efficient and accurate model of a complex shape. If such a method does not exist, then the cubic elements will definitely suffice.

A third improvement is to have ANSYS apply self-weight completely before the other loads to get a more realistic performance index based on actual behaviour (see Section 6.6).

Some additions to the concrete analysis are possible.
At the moment FABRICFORMER only optimizes for evenly distributed loads and self-weight. Obviously this can result in geometries that perform poorly for certain load cases such as point loads. It might be prudent to include multiple load cases and alter the model in ANSYS to derive overall performance based on these parallel analyses. These could include specific effects such as torsion or dynamic loading.

The concrete in FABRICFORMER remains limited to conventional reinforcement. It is worthwhile to expand upon FABRICFORMER by including modelling of fiber reinforced concrete and/or prestressing. It is noted that this author consciously avoided fiber reinforcement, since at the moment little to nothing is known about its sustainability characteristics. There are papers by Padmarajaiah & Ramaswamy (2002) and Thomas & Ramaswamy (2006) on the modelling of fiber reinforced concrete in ANSYS.

The reinforcement strategy in this project included the use of steel tendons. Chapter 3.2 shows that currently helical strands of steel are used, using circular pieces to hold them at a certain cover distance from the fabric.
Alternative strategies could include suspending strands or entire prefabricated reinforcement cages. There is also the possibility to protrude the fabric to position reinforcement steel inside.
Other than strands, tendons or cages, one could also investigate the use of discretely distributed straight bars of steel that approximately follow the curvature of the geometry.
One final option is to use high strength tensile fabrics (such as carbon fiber fabrics) that are not removed but function as reinforcement.

TABLE 6.1 Recommendations
Improvements
Further verification of the approximation method
Flexible, reliable unstructured meshing method (not ANSYS) to replace current block mesh
Have ANSYS first apply self-weight prior to other loads
Additions
Multiple load cases
Modelling of e.g. fiber reinforcement, or prestressing
Investigate various reinforcement strategies (fiber reinforcement, prestressing, strands/tendons, bars/cages, high strength carbon fabrics)
(Superior hardware and/or parallel computation)

CHAPTER 7 | FABRICFORMER software architecture

The various algorithms and methods that have been seperately discussed in the preceding Chapters 3 to 6 form integral parts of the overall program that was designed to reach the objective of optimizing a manufacturable fabric formed beam. The program is called FABRICFORMER. The general outline and structure of the program will be discussed in the following paragraphs. Design and programming of FABRICFORMER took considerably more time than had been anticipated in advance. The resulting complexity of the program is illustrated by its size, amounting to about 6 kiloLines of Java and ANSYSscript code.

7.1 Design process and software

The FABRICFORMER program has been coded in the Java language, mainly because the author was most familiar with this language. Additionally Java3D was installed which offers additional support for calculations and visualisations in 3D. The entirety of the program was designed and structured in ECLIPSE for Java, a freely available integrated development environment, or IDE.

Because some of the calculations were outsourced to ANSYS (see Chapter 6), the program includes some ANSYSscript code providing instructions for an ANSYS batch run (i.e. where ANSYS runs in the background on a single set of calculations). Figure 7.1 shows FABRICFORMER in its most basic terms; an optimization iteration (DE) with form finding (DR) and structural analysis (ANSYS).

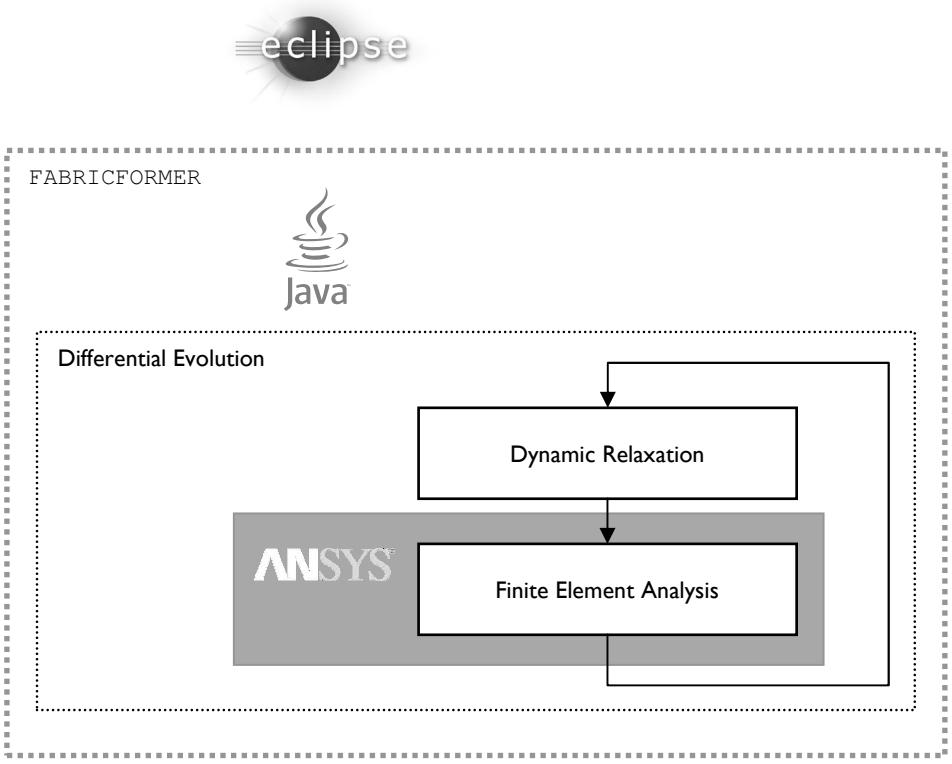


Figure 7.1 Simple diagram showing how FABRICFORMER functions

The overall sequence of programming for FABRICFORMER went as follows:

- Code and test Dynamic Relaxation
- Code and test Differential Evolution
- Code and test Java3D GUI
- Expand Dynamic Relaxation for spline mold
- Code and test outsourcing to ANSYS batch run
- Change Dynamic Relaxation for keel mold
- Expand Dynamic Relaxation for pinch mold
- Tie Dynamic Relaxation, Differential Evolution and ANSYS batch run together and test

This is roughly how the design process of FABRICFORMER took place between November 2007 and April 2008. This included getting familiar with Java, Java3D, ECLIPSE, the algorithms and using batch runs for ANSYS. The process also included the programming of several specific mathematical operators, such as Heron's formula, Rodigues' formula (see Chapter 5) as well as other vector operations that are not included in Java's standard Math class.

Prior to November 2007, ANSYSscript files were written for the various runs of ESO and BESO (Chapter 2). Links to software to code in ANSYS were found at the ANSYS Customer Portal (http://www.ANSYS.com), but at that point most of the programming had already been done in Windows' NOTEPAD.

On a side note, at one point a form-finding method was developed that was ultimately abandoned (see Appendix D). The development of this method took place in MUPAD PRO, a math program which is very similar to MAPLE, and at that point FABRICFORMER would have outsourced the form-finding to MUPAD PRO.

7.2 Input for FABRICFORMER

Most of the values that FABRICFORMER requires are input in the Java Values.class. These are:

- dimensions of the design domain
- ranges (minimum/maximum) of the optimization variables
- values for the creation of the fabric mesh
- values for Differential Evolution
- values for Dynamic Relaxation
- values for ANSYS including concrete properties

Table 7.1 lists typical input for FABRICFORMER.

The design domain was chosen after calculating a typically sized cross-section of a rectangular beam based on the arbitrarily chosen span of 9 m.
The number of rows and columns (i.e. number of nodes in x and y-direction) for the fabric mesh used in DR was determined over time by trial and error. The specific values in Table 7.1 provide a compromise between precision and computational time and correspond to the mesh size that was chosen for the ANSYS mesh as well i.e. both mesh element size have the same order of magnitude and comparable size, in another word; resolution.

The parameters for Differential Evolution are from the last runs of the program, where the population Np counted 50 beams, the number of optimization variables i.e. Dimension was 40 (see Chapter) for a pinch mold. The crossover and scaling factor were determined as detailed in Chapter 4.4.

The Dynamic Relaxation parameters are largely taken from the work of Schmitz (2004) which linearized the modulus of elasticity E of a typical formwork fabric to 490 N/mm. The Wide Width Tensile WWT is an empirically derived maximum fabric stress of the same fabric and is 40 N/mm (see Chapter 3.3). As of yet, FABRICFORMER provides no checks on stresses (but is able to visualize them in a Java3D window by colorcoding elements that exceed this WWT), and uses the WWT primarily to determine the maximum allowable prestressing. This is set to 150% of the WWT, reasoning that failure of the fabric should lead to a thicker and/or stronger fabric, not discarding of the generated shape. This provides the current program with a certain freedom to explore shapes.
The time increment Δt was set to 0.7, and it is recalled that Barnes (1998) recommends values no higher than 1.0.

The values for ANSYS are the properties of B25 concrete. As mentioned in Chapter 6.2 these values should have been changed to properties for B65, typical of the prefab industry. However non-linear analysis was not activated during the use of FABRICFORMER due to a lack of computational power at this time. These properties are therefore not actually used, except for the modulus of elasticity. Due to the nature of the linear analysis that took place instead, it was no longer relevant to change these values at this time.

The line load of 10 kN/m was applied in linear analysis after it became clear that non-linear analysis would take too much time to calculate during optimization. In the latter case, the value of the load would have been unimportant as long as it would guarantee failure (about 120 kN/m or

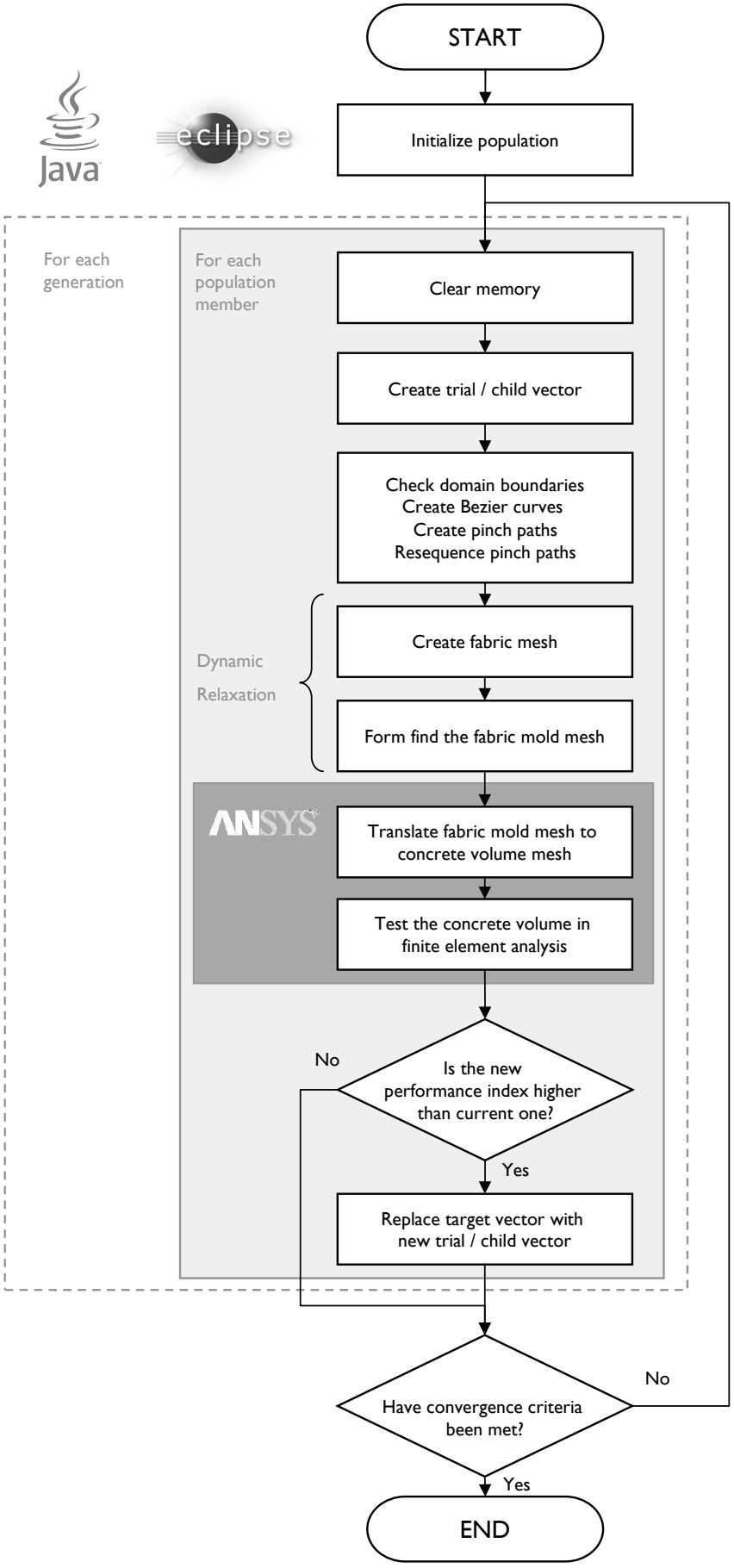


Figure 7.2
Flowchart describing how FABRICFORMER functions.

higher, depending on the current volume) as discussed in Chapter 6.6. The load of 10 kN/m was seen as an appropriate load, significant enough to play a role in optimization and much higher than the load of 100 N/m² that typical examples in BESO literature have used.
Other values that have to be specified in other classes are

- the names of various text files and directories for output

TABLE 7.1 User input

Domain dimension		
width	= 0.300 m	
height	= 0.800 m	
length	= 9.000 m	
Fabric mesh		
yRows	= 60	
xColumns	= 24	
Differential Evolution		
Np	= 50	Np
Dimension	= 40	D
crossP	= 0.2	
scaleF	= 0.4	
Dynamic Relaxation		
EAConst	= 490 N/mm	EA
WWT	= 40 N/mm	
TimeIncrement	= 0.7	Δt
tIntervals	= 24	
densConcrete	= 2400·10⁻⁹ kg/mm³	ρ_c
gravityConst	= 9.81 m/s²	g
ANSYS variables		
elementSize	= 0.025	
fs	= 500e6 N/m²	f_s
eps_spl	= 0.00250	$\varepsilon_{s,pl}$
eps_su	= 0.03250	$\varepsilon_{s,u}$
Es	= 200000e6 N/m²	E_s
fc	= 28.2e6 N/m²	$f_{c,c}$
ft	= 2.98e6 N/m²	$f_{c,t}$
eps_cpl	= 0.00099	$\varepsilon_{c,pl}$
eps_cu	= 0.00350	$\varepsilon_{c,u}$
LineLoad	= 10.000 N/m	q

7.3 Output for FABRICFORMER

As soon as it became apparent that the optimization process would be very demanding on time, there was as an emphasis on the output of FABRICFORMER. The risk of the software failing during optimization requires that output be continuous, and that the latest data could be used to resume the optimization process without any loss of information. To achieve this, FABRICFORMER continuously outputs information from within ANSYS and directly after evaluation of the performance of each solution. Although ANSYS is capable of outputting virtually all data in one single database .db file, continuously doing so would take up too much harddisk space in the long run. As a compromise, FABRICFORMER saves the values of the twelve or forty parameters (keel mold, or pinch mold respectively) that determine the shape of the solution that is sent to ANSYS.

The following output files are generated from within ANSYS in the folder
..\POPULATIONMEMBER[VECTOR]\

- FABRICFORMER[NUMBER].PNG
- OUTPUT[GENERATION].TXT
- PI.TXT
- V.TXT

where

[VECTOR]
is a number that corresponds to the particular beam, so it ranges from 0 to 49 (i.e. a population of fifty solution vectors)
[NUMBER]
is a number that ANSYS automatically and cumulatively appends to each file name, starting at number 000.
[GENERATION]
is the number of the generation, starting at 1.

The graphic .png files are side and front views of each solution (e.g. Figure 8.8 in Chapter 8). The colors correspond to the range of stresses in the longitudinal x-direction, from compressive stresses at the top to tensile stresses at the bottom.
The OUTPUT[GENERATION].TXT textfile is a standard ANSYS summary of its output and detailes the calculations at made as well as any warnings or errors that occurred. It is used to track any bugs in FABRICFORMER that occurred as a result of the ANSYS mesh generation or finite element analysis.
The last two PI.TXT and V.TXT files are temporary fi-

les, used to transfer the performance index PI and volume V back to Java for evaluation of the solution.

FABRICFORMER itself also outputs various textfiles in the ..\POPULATIONMEMBER[VECTOR]\ folder of each population member:

- INPUT[GENERATION].TXT
- VALUES_INPUT_MEMBER[VECTOR].TXT
- VALUES_INPUT_MEMBER[VECTOR]_BASIC
- VALUES_OUTPUT_RUN.TXT

The INPUT[GENERATION].TXT file contains the ANSYSscript that is generated in Java and sent to ANSYS for its batch run. It can be seperately reused to generate and analyze one particular solution. It is therefore used after optimization to reopen the best solution and perform any additional calculations or other post-processing operations. The VALUES_INPUT_MEMBER[VECTOR].TXT and VALUES_INPUT_MEMBER[VECTOR]_BASIC.TXT text files both store the twelve or forty parameters - that change during optimization - that were used in each particular solution.

The VALUES_OUTPUT_RUN.TXT file stores the performance index and beam volume for each solution.

Three main text files are generated in the main directory ..\ of the FABRICFORMER run:

- VALUES_INPUT_AT_GENERATION.TXT
- VALUES_OUTPUT_RUN.TXT
- VALUES_STATIC.TXT

The VALUES_INPUT_AT_GENERATION.TXT file stores the twelve or forty parameters – that change during optimization – that were used in each particular solution and each population member. If FABRICFORMER or the hardware should fail, the latest values in these text files can be used to resume the optimization.
The VALUES_OUTPUT_RUN.TXT file stores the performance index and beam volume for each solution and each population member. This file is used to generate the performance index graphs and evaluate whether the optimization is converging. The values_static.txt file is stored once at the start of FABRICFORMER and is a summary of all the variables that were used as described in the previous paragraph.

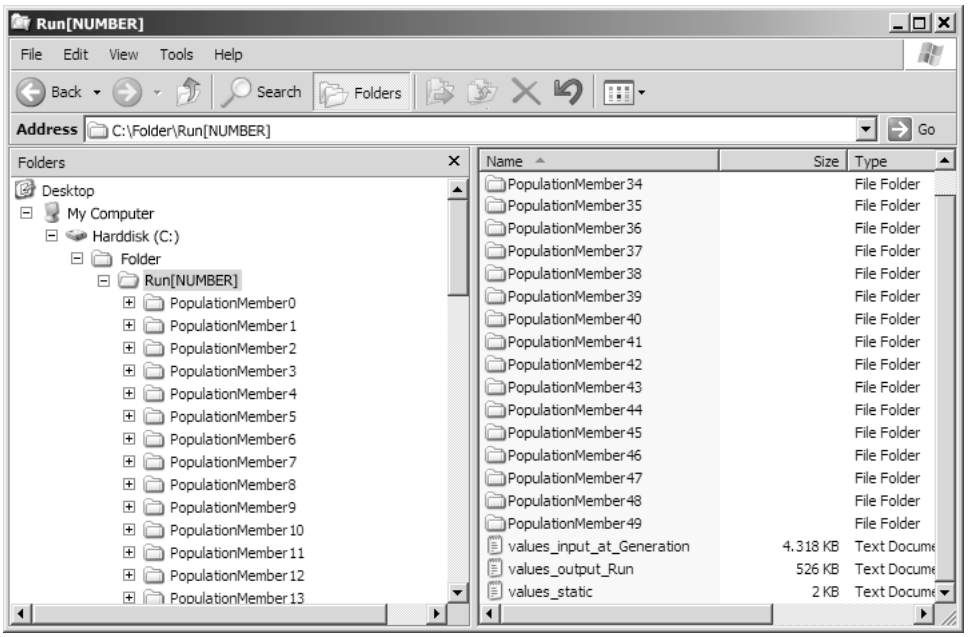


Figure 7.3
Screenshot of Windows Explorer, showing some of the folder structure and files that FABRICFORMER outputs.

7.4 Classes and class diagrams

The Java language is object-oriented (OO), which means that it structures the software in different components called classes. These classes are related to each other and several defined types of relations determine in what way they interact. In typical object-oriented programming-the programmer designs his software using diagrams such as those associated with the Unified Modelling Language (UML) prior to writing code. However, in conjunction with the graduation committee of this thesis, the decision was made to dispense with the UML modelling of the program and start programming straight away in order to save time and because there is no emphasis on standard software en-

gineering practices for theses at this faculty. As a result, the current structure of FABRICFORMER, as shown in Figure 7.2, does not conform to conventional class structures in UML. The program VISUAL PARADIGM was used to reverse the Java source code to class diagrams which indeed revealed that there are no typical class relations, suggesting that all the classes in FABRICFORMER are seperate. Of course this is not possible, and the way in which the classes interact simply do not follow UML convention. To understand how FABRICFORMER is structured, the reader is again referred to Figure 7.2.
In this paragraph the seperate classes are shown as single UML class diagrams. For future development of FABRICFORMER, it would be possible to restructure it according to a standard UML model using a program such as VISUAL PARADIGM.

The program consists of ten classes. They are:

- Main.class
- DifferentialEvolution.class
- CreateMesh.class
- DynamicRelaxation.class
- PinchMold.class
- Calc.class
- Values.class
- runANSYS.class
- StreamGobbler.class
- render3D.class

The Main class is the central core of the program and initializes or runs the other classes. The code of the Main class corresponds sequentially to the program structure as seen in Figure 7.2.

Main.class
~(various constructors referring to Values.class) ~WorkingFolder : String ~Run : String
+main()

The DiffentialEvolution class contains all the information from Chapter 4 in code form.

DifferentialEvolution.class
+Init() +Boundaries() +generateTrialVector() +vectorIndices() +PowerLawF()

The CreateMesh class generates a triangular fabric mesh for Dynamic Relaxation. It stores this informates in various arrays containing Cartesian coordinates of the nodes and connectivity data for the elements.

CreateMesh.class
~(various constructors referring to Values.class) ~maxStrings : int ~ElementTributaryWidth : double[]
+Nodes() +MinimumzHeight() +Elements() +nodeElements() +Areas() +calcStiffness() +NodeAreas() +ElementAreas() +TriangleData() +ElementStiffness()

The Dynamic Relaxation and PinchMold classes are the largest class, containing all the information in Chapter 5 in code form. The PinchMold class was created after dynamic relaxation of the keel mold was succesful. This was done to clearly divide new lines of codes that deal exclusively with modelling the pinch mold.

DynamicRelaxation.class
~(various constructors referring to Values.class) +elementTributaryWidth[] : double
-Relax(): double[][]

PinchMold.class
+intersectLineWithPath2D() +spanLineOverPath2D() +intersectLineWithArea() +domainPath() +pinchPath() +checkElementCrossesPath() +improveEdgePoint() +resequencePinchPoint() +orderPinchPoints()

The Calc class contains various additional mathematical formulas used for form finding and collision detecion as well as various vector operations.

Calc.class
+Heron() +Angle() +Distance() +Length() +LengthVector() +CrossProduct() +DotProduct() +VectorHeron() +AngleAtFirstCoords() +AngleVectors() +UnitVector() +RodriguesVector() +OrthogonalProjectedVector() +Inradius() +Bezier() +DerivariveBezier() +InterpolateBezier() +Interpolate() +subVectors() +multVector() +multVectors() +addVectors()

The Values class contains all the user input values that are used throughout the other classes. This was done to centralize user input.

Values.class
+width : int +height : int +length : int +yRow : int +xColumns : int +maxElements : int +maxNodes : int +maxAreas : int +xWidth : double +yHeight : double +zDepth : double +elementSize : double +tolerance : double +tIntervals : int +EAConst : double +densConcrete : double +gravityConst : double +TimeIncrement : double +KE_max : double +WWT : double +PrestressMin : double +PrestressMax : double +EdgeMoldMin : double +EdgeMoldMax : double +KeelMoldMin : double +KeelMoldMax : double +e : double +fs : double +eps_spl : double +eps_su : double +Es : double +fc : double +ft : double +eps_cpl : double +eps_cu : double +Ec : double +LineLoad : double +ScaleFactor : double +Np : int +Dimension : int +crossP : double +scaleF : double

The runANSYS class contains all the ANSYSscript code as well as the means to create new directories and to read or write text files.

runANSYS.class
+fileWriter() +inputWriter() +outputWriter() +staticWriter() +generationWriter() +runWriter() +mkDirectory() +fileReader(): double +getScaled(): double +Round(): doubl

The StreamGobbler class is necessary for the in- and output of text files to communicate with ANSYS and store data of the optimization process as discussed in the previous paragraph.

StreamGobbler.class
~is : InputStream ~type : String
~StreamGobbler() +run()

The Render3D class only functioned during the development stages of FABRICFORMER. It opened a GUI with a Java3D screen to visualize the Dynamic Relaxation process. Besides testing, it was also used to capture video of the form finding for presentation purposes. More information on the GUI is given in Section 7.5.

render3D.class
~(various constructors referring to Values.class) ~simpleU : SimpleUniverse ~scene : BranchGroup ~scene2 : BranchGroup ~vertices : double[] ~colors : double[] ~normals double[] ~maxLines : int ~lines : double[] ~StrainData : boolean[][] -fabricNormals : Shape3D -fabricLines : Shape3D #objFabric : TransformGroup #objConcrete : TransformGroup ~b1 : Button ~b2 : Button ~b3 : Button ~b4 : Button ~b5 : Button ~b6 : Button ~b7 : Button ~LoadsVisible : boolean
+appearance() +appearanceNormals() +createGeometry() +createNormals() +createLines() +createSceneGraph() +createSceneGraph2() +actionPerformed() +render3D() +destroy() -updateColors() -updateVertices() -updateNormals() -drawOriginAndBoundaries()

Alternatively to the standard UML diagrams, one possible way to present the program is Figure 7.4, where each arrow is a class getting a variable or using a method from the class it points to. The diagram shows how the Main class is the core of the program, and that several classes use input values from the Values class and mathematical formulas from the Calc class.

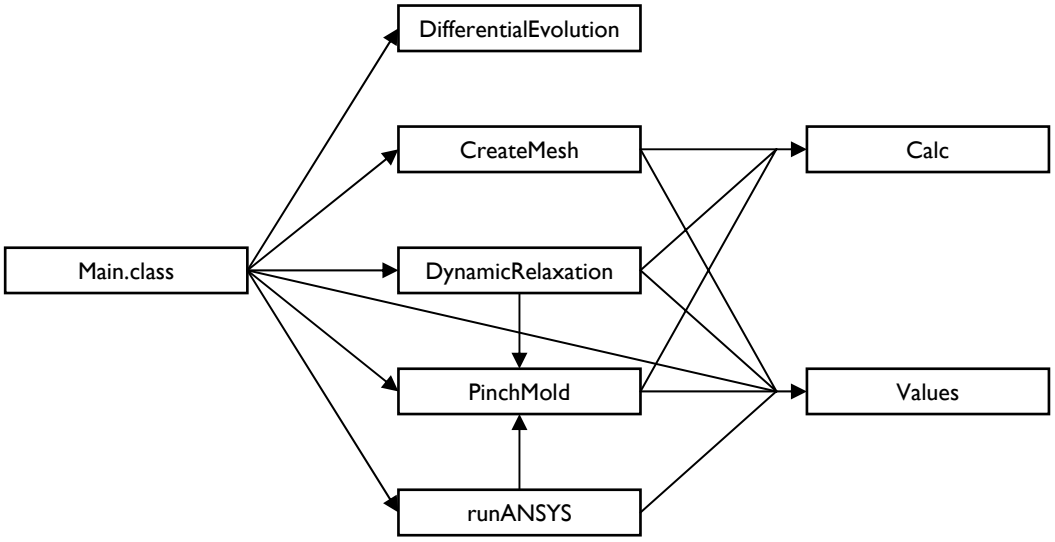


Figure 7.4
Basic diagram of FABRICFORMER showing how the classes relate to one another. Note that the code does not correspond to conventional UML relations, but that the classes are definitely connected.

7.5 Graphical User Interface (GUI)

As of yet, FABRICFORMER has no GUI and has been executed from within ECLIPSE. However, during development a GUI was programmed to visualize the Dynamic Relaxation process. The earliest version was merely an isometric representation, using two-dimensional lines in Java Swing. Soon, the GUI was upgraded to use a three-dimensional representation. Screenshots of these visualizations are shown throughout Chapter 5.5. To program the three-dimensional output it was necessary to install Java3D which expands the Java code to include various ready-made 3D-modelling and rendering tools. The resulting visualization proved to be invaluable during the debugging of FABRICFORMER as it quickly indicated what general type of error the program had encountered.

The GUI is primarily a window showing the 3D-model of the fabric mesh, both before, during and after Dynamic Relaxation. Using the mouse, it is always possible to zoom in and out, as well as translate and rotate the model. The buttons at the top in Figure 7.5 read 'Hide loads', 'Top', 'Side', 'Section' and 'Isometric'.

The 'Hide loads/Show loads' button can hide or show the prestressing and concrete loads on the nodes of the fabric mesh. It was implemented in an early stage to verify the realism of the concrete pressures, and Figure 7.6 shows an earlier version showing the loads as well as two additional buttons. These two buttons were used to first apply the prestressing load and then add the concrete to evaluate the intermediate geometry of the fabric mold. This two stage loading was later deemed unnecessary and dropped.

The latter four buttons have designations that are self-explanatory and give corresponding fixed views of the model. These views were used in the screenshots in Chapter 5.

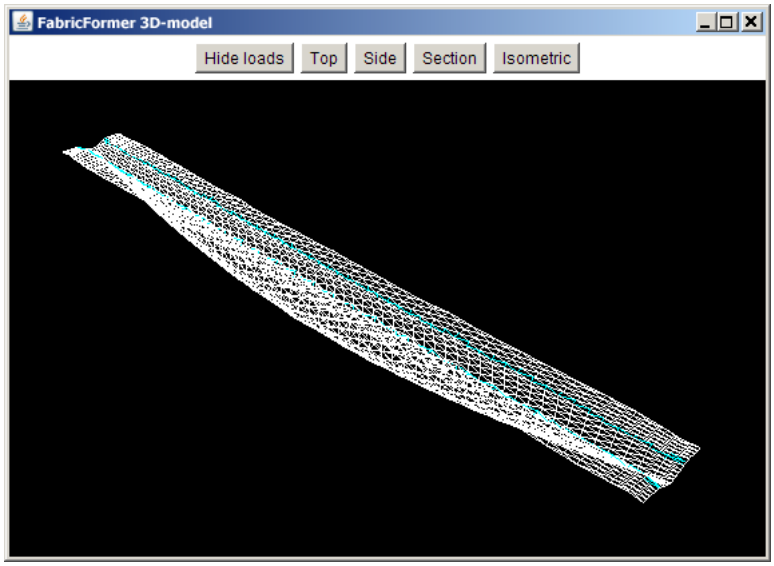


Figure 7.5
Screenshot of the FABRICFORMER GUI

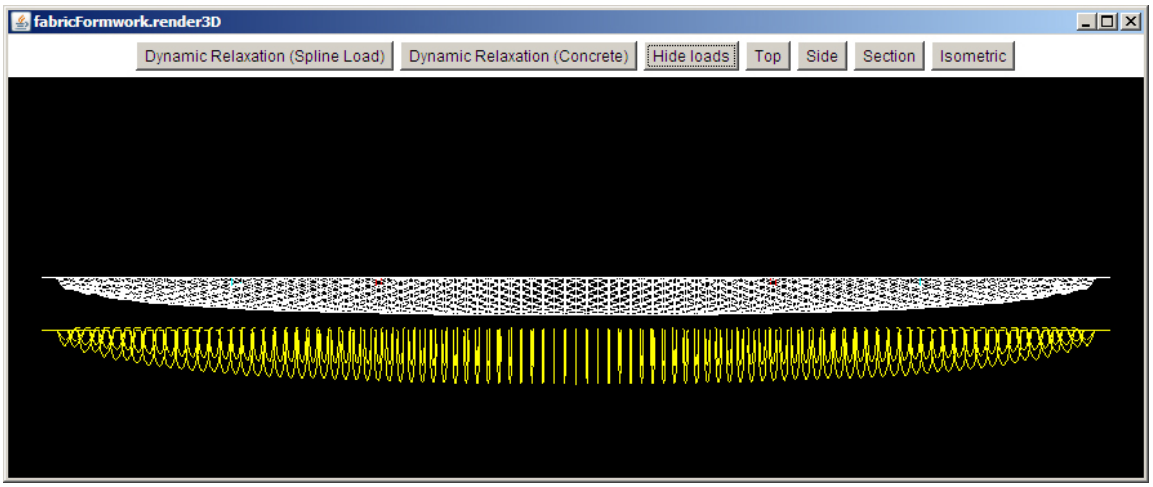


Figure 7.6
Early screenshot of the FABRICFORMER GUI including visualization of the concrete pressures.

7.6 Computational time and hardware

Several times throughout this report it has been mentioned that limited computational power was available for this thesis. The demand of FABRICFORMER as an optimization program is significant and each run took several days before conclusions could be made on the performance of the program and the quality of the results. This caused debugging and development of the program to take up several months. Two computers were used to run the program, because the first one ceased to function properly. The computer specifications are:

- Intel Pentium IV
  3.41 GHz
  2.00 Gb RAM
  Windows XP
- Intel Pentium IV
  2.99 GHz
  0.99 Gb RAM
  Windows XP

Inquiries were made at several faculties at the TU Delft for high performance computers running ANSYS, but none were found. The possibility came up to use all computers at this faculty between the hours of 22.00 and 07.00. No use was made of this offer because no software could be installed, the offer came at late stage and this author had no previous experience with parallel computation. Future development at this university could try and make use of this possibility. As mentioned in the conclusions of Chapter 5, a paper by Topping & Khan (1994) described a parallel computation scheme for Dynamic Relaxation and furthermore it is evident that, for instance, a generation of 50 solutions in Differential Evolution may easily be evaluated on 50 seperate computers.
Alternatively, some other finite element program than ANSYS could be uses for which remote high performance servers do exist e.g. DIANA.

In accordance with good programming practice, the various components of the program were timed in millise-

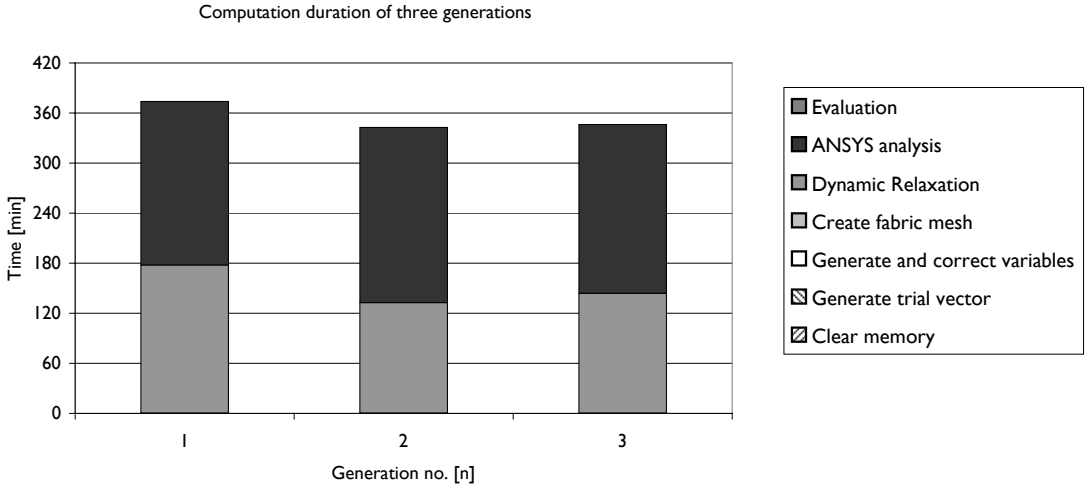


Figure 7.7
Diagram showing the computational demand in minutes for three generations in FABRICFORMER each calculating fifty beams.

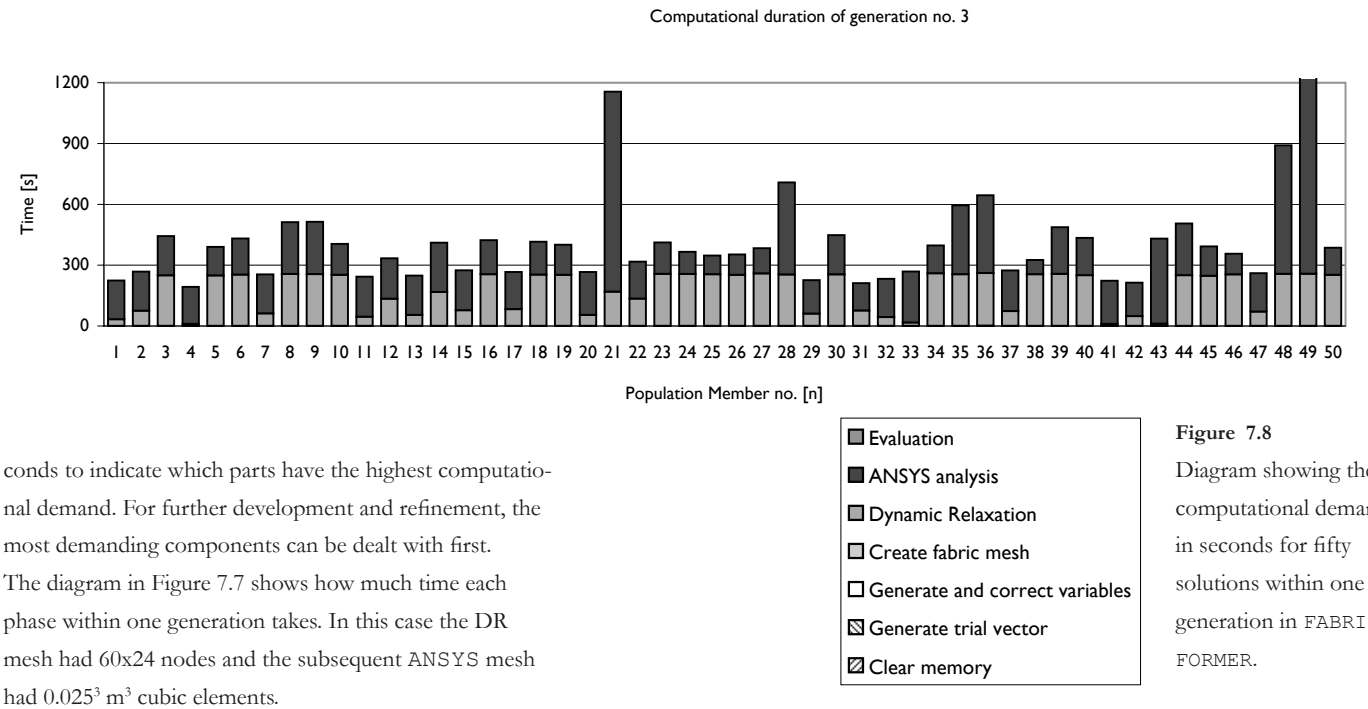


Figure 7.8
Diagram showing the computational demand in seconds for fifty solutions within one generation in FABRICFORMER.

conds to indicate which parts have the highest computational demand. For further development and refinement, the most demanding components can be dealt with first.
The diagram in Figure 7.7 shows how much time each phase within one generation takes. In this case the DR mesh had 60x24 nodes and the subsequent ANSYS mesh had 0.025^3 m^3 cubic elements.
The various phases correspond roughly to the flowchart in Figure 7.2. Based on the evaluation of three generation of pinch mold optimization (50 beams), it can be said that one generation takes about 6 hours to complete for an ANSYS element size of 25 mm and a DR element size that is comparable. The most demanding phases are the Dynamic Relaxation and the ANSYS structural analysis, the former taking about 2 to 3 hours and the latter about 3 hours for fifty beams.
The next diagram in Figure 7.8 gives the duration of each individual beam for the third generation. From this diagram it can be concluded that the DR phase has an upper limit of about 260 seconds and a few beams take significantly more time to calculate in ANSYS.
The upper limit in DR is indeed defined in the code as a maximum number of iterations (i.e. 2000 cycles) which was empirically derived as a stage when the final form has always been found, but oscillations of some nodes might cause the algorithm do never converge. The conclusions in the DR chapter (see Chapter 5.6) allude to this phenomenon as it was recommended to improve the collision detection within the form finding model to prevent such non-convergent behaviour.
To ascertain the cause of the spikes in ANSYS calculation time, the operations in ANSYS itself is divided in different stages. The output text files already provide this information. Figure 7.9 breaks down the analysis for two beams in the third generation, beam numbers 42 an 49. The latter beam is one of the anomalous spikes and the diagram illustrates that solving the finite element problem is the cause of the problem. From reviewing the specifics of the out-

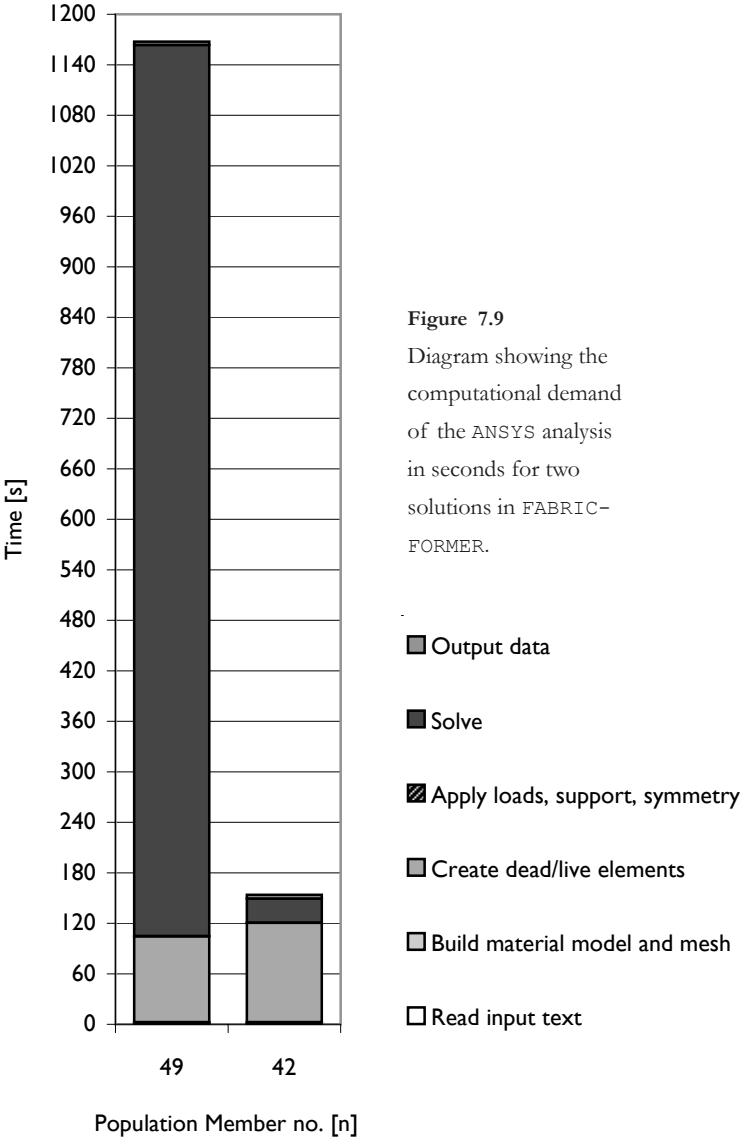


Figure 7.9
Diagram showing the computational demand of the ANSYS analysis in seconds for two solutions in FABRICFORMER.

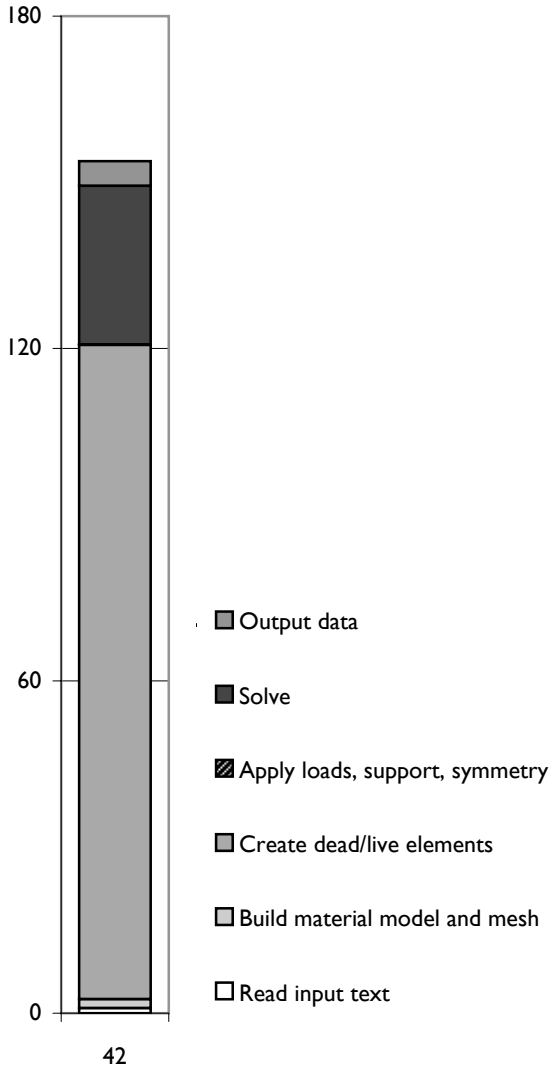


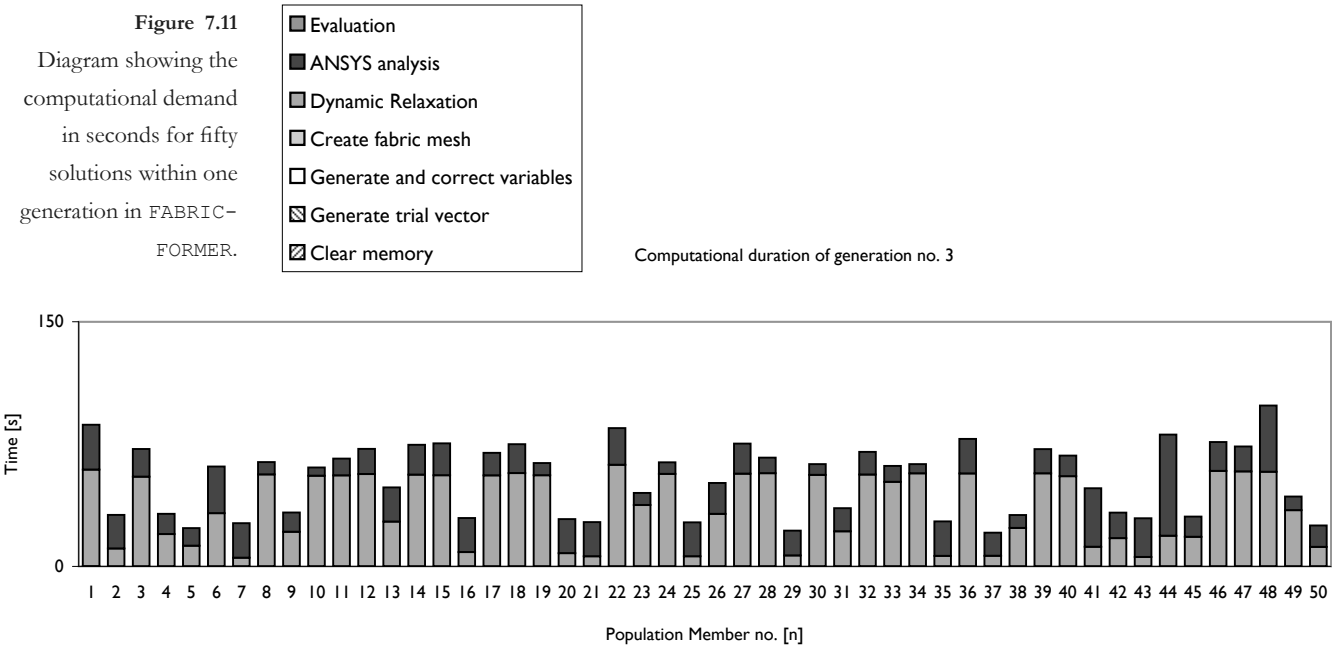
Figure 7.10
Diagram showing the computational demand of the ANSYS analysis in seconds for a solution in FABRICFORMER.

put file it became clear that ANSYS engaged in non-linear analysis due to instability of the beam, for beams that had extremely little volume i.e. were highly flexible. Non-linear analysis could not be prevented in advance, because several options connected to non-linear analysis have to be switched on when using live and dead elements.
Figure 7.10 shows a close-up of the typical computational demand in ANSYS, again for solution 42 from Figures 7.8-7.9. When analys runs smoothly, the most demanding phase is the creation of 'DEAD' or 'LIVE' elements from the initial design domain. Alternative strategies without element birth/death were also attempted, but generally took even longer. Nonetheless, the author intuits that the total creation of the mesh could be optimized considerably.

Figure 7.11 shows another generation with a significantly lower computational demand. In this case the DR mesh was 30 x 12 nodes and the ANSYS mesh had 0.050^3 m^3 elements. Based on these sample test, it can be concluded that half the mesh size results in only 13% of the compotional load, a significant drop.

These two mesh grades have both been used, because the finer mesh gives more accurate results and yields better convergence, while the coarser one is clearly much faster. Grades outside this ranges are obviously too slow to compute or too coarse too be of any significance.

Figure 7.11
Diagram showing the computational demand in seconds for fifty solutions within one generation in FABRICFORMER.



7.7 Conclusions and recommendations

Combining the various algorithms and concepts, discussed in Chapters 3 to 6, was succesful. Therefore, assuming that the results in Chapter 8 are satisfactory, it is possible to say that combining computational optimization and manufacturability, or specifically Differential Evolution and Fabric Formwork is feasible. The ramifications of this conclusion are further discussed in the main conclusions on this thesis in Chapter 9.

The FABRICFORMER program is an operational piece of software but is not yet user-friendly, because this was not an objective within the context of this thesis. To achieve general usability, the following steps should be taken:

The program should be a stand-alone product, no longer dependent on ECLIPSE. This is relatively easily accomplished. Ideally, the program should also be independent of ANSYS, therefore including its own finite element analysis component or being flexible enough to make use of a number of finite element programs. Obviously this would take considerably more effort.

A user-friendly graphical user interface (GUI) should be programmed to include all user input and program output. It should also show the progress of the optimization process, e.g. a live performance diagram and current solution geometries i.e. the shape of the best beam/product so far. To facilitate the GUI, the output should be streamlined for visualization instead of being stored in various graphic .png files and .txt text files.

At this point no (practical) convergence criteria for the Differential Evolution have been specified. These should be included in the software, either coded or by user input. Currently the program is stopped manually (or when 10.000 generations have been reached, but that is unlikely).

Generally, the software could be refactored to conform to UML standards and conventions. This would make the program code more clear and legible to other programmers. Software exists to reverse engineer the current program and refactor the code to conform to UML diagrams.

Finally, the program could be altered to include parallel computation efficiently dividing the work load among more than one computer.

It is also recommended to use higher performance hardware than what was available for this thesis.

TABLE 7.2 Recommendations

Improvements
Export FabricFormer as a stand-alone product
Create internal Java-coded finite element analysis, or include flexibility to use various finite element programs
Streamline the various image and text output to a single source e.g. a GUI.
Specify input of overall convergence criteria to automatically stop the program
Conform software structure to UML conventions
Additions
Add a graphical user interface (GUI) that includes input, progress reporting and output
Parallel computation
(Superior hardware)

CHAPTER 8 Results and interpretations

The improvements to the BESO algorithm and the creation of the FABRICFORMER program have both led to results that are summarized in this chapter. The data and data interpretations of both the BESO algorithm and FABRICFORMER will be discussed, prior to drawing any definitive conclusions on the overall thesis. A comparison is made between the results and conventional, rectangular beams and parabolically shaped beams to indicate whether the algorithms have indeed resulted in optimized geometries

8.1 Results from BESO for non-linear reinforced concrete beams

The BESO algorithm, as described in Chapter 2 was used to optimize reinforced concrete beams. The algorithm featured an adapted RRV_{rel} (see Section 2.5) and a continuous process of recalculating longitudinal reinforcement area and orientation (see Section 6.3).

Contrary to conventional BESO, the non-linear analysis takes place by incrementally loading the beam until failure has been reached. Then the performance index is calculated by retrospectively calculating the strain energy as loading increased i.e. integrating strain energy over time (see Section 6.6)

An advantage of this method is that the performance index becomes independent of the load and more faithfully represents the stiffness of the beam as a function of applied load.

One drawback of using ANSYS is that besides the external loads, self weight is applied incrementally as well. This drawback causes the calculation of the strain energy over time to be inaccurate, since self weight is obviously always be present from the beginning. This may cause converged solutions to not be actual optimal solutions, though it is unknown if this occurs.
In any case, for now it is recommended to check that failure of the converged solution occurred at a high percentage of the prescribed load (which has been chosen to be sufficiently high to assure failure) so that this effect becomes increasingly negligible.
At a later stage – too late to implement – this problem was solved, as described in Section 6.6.

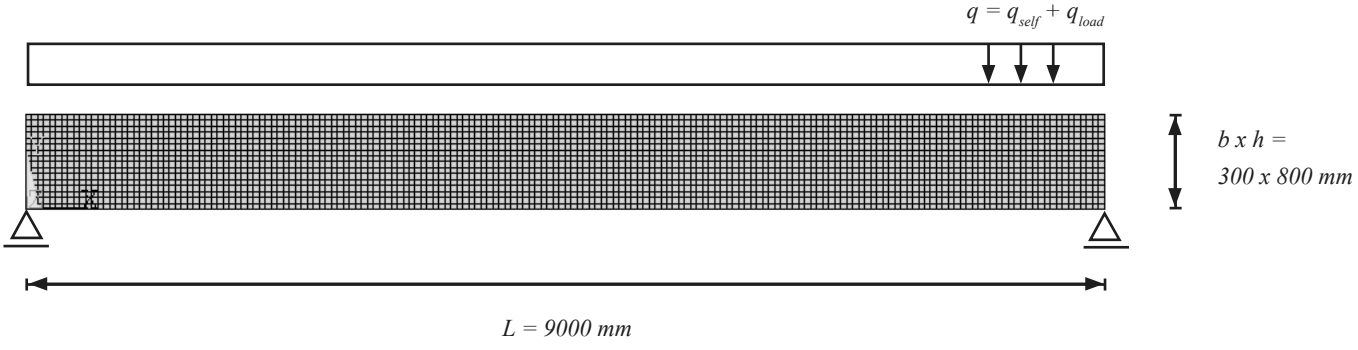


Figure 8.1
General load case, support conditions and initial/design domain for all runs of BESO and FABRICFORMER in this chapter. The symmetry plane at midspan is fixed in x-direction.

This first succesfully completed run used the properties of B25, which were also used for the hand calculated concrete beam in Chapter 6.4. Table 8.1 shows the various input values for this run.

TABLE 8.1 Properties for BESO run

Concrete B25		
Modulus of elasticity	E_c	28.485 N/mm²
plastic strain	$\varepsilon_{c,pl}$	0.99 ‰
ultimate strain	$\varepsilon_{c,u}$	3.50 ‰
compressive strength	$f_{c,c}$	28,2 N/mm²
flexural tensile strength	$f_{c,t}$	2,98 N/mm²
density	ρ	2400 kg/m³
Steel FeB 500		
modulus of elasticity	E_s	200.000 N/mm²
plastic strain	$\varepsilon_{s,pl}$	2.50 ‰
ultimate strain	$\varepsilon_{s,u}$	32.5 ‰
strength	f_s	500 N/mm²
density	ρ	7800 kg/m³
Beam dimensions and BESO parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
element size		0.050 m
load (must approach or surpass failure)		60.000 N/m² = 18 kN/m
height of smeared reinforcement zone		0.100 m
removal rate of volume	RRV	7.5 %
convergence value, or 'error'		1 %

This run took 3 days and 2 hours to converge on an Intel Pentium 4 3,00 GHz with 1 Gb RAM. This clearly shows that the non-linear analysis significantly slows down the BESO process. In the graph below, it is shown how the performance index developed during this time. On the next page, we will focus on the peaks at steps 13 and 40.

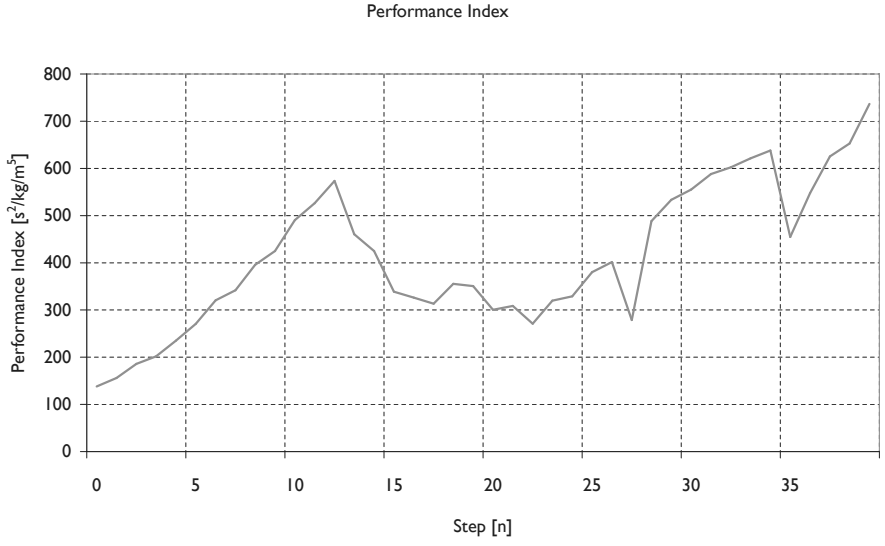


Figure 8.2
Graph showing the performance index during optimization

At step 13, where the first peak occurs but convergence has not been reached yet, the performance index is 574, and the volume is a little over 25% of the total domain. It is visible that BESO creates a compression and tension member in the middle of the span a a result of the maximum flexural moment. At some point, where both large shear forces and moments are present, the first diagonal member is found. As we approach the supports, the height of the beam diminishes.

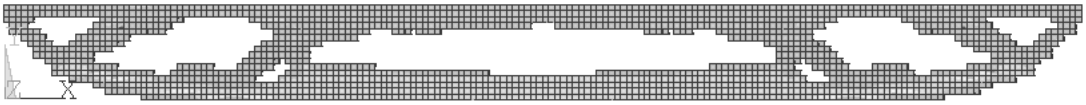


Figure 8.3
Beam optimized in BESO at step 13.

In contrast to linear BESO, there is no moment diagram shaped bottom flange, rather the middle section is completely straight. It is clear from this that non-linear BESO takes advantage of placing the longitudinal reinforcement as low as possible for as long as moments are significant.

As the algorithm continues it slowly expands the opening in the middle of the beam and cross-section, and finally converges at a similar, but different shape at step 40. By now, the difference in performance index with the previous iteration is less than 1%.
At this point the performance index is 736 and the volume is a little over 23% of the total domain. When inspecting the estimated reinforcement, it is found that the percentage of reinforcement at midspan is 1,26%, which is well over the prescribed percentage for B25 in rectangular beams, which is 1,38%.

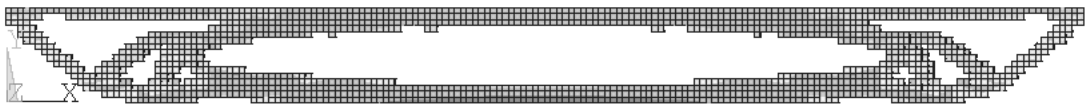


Figure 8.4
Beam optimized and converged in BESO at step 40.

In the graph below, the development of the volume of the beam is shown as a percentage of the total domain.

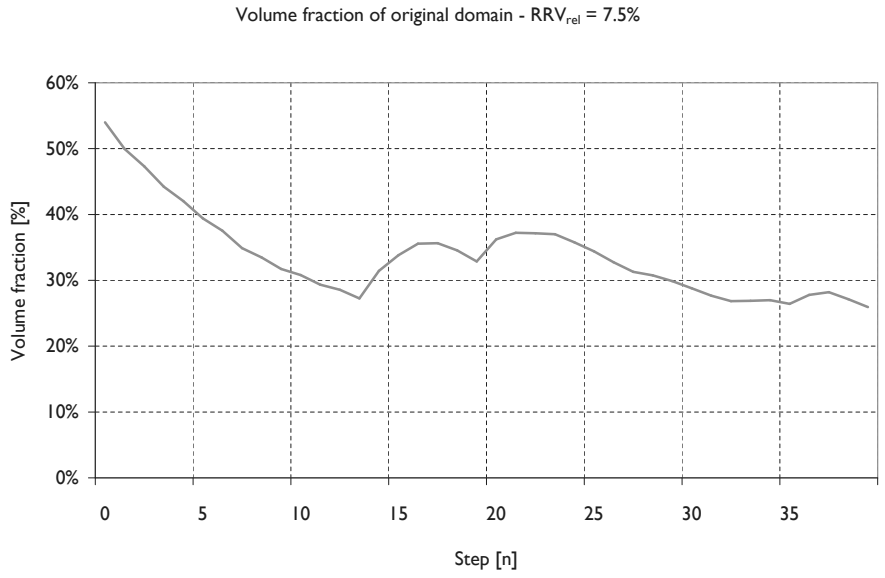


Figure 8.5
Volume fraction of the current volume compared to the initial design domain during optimization of BESO.

For newer BESO runs, the concrete properties for B65 should be used, which is a more conventional concrete type in the prefab industry. Furthermore, it is recommended that the self weight of the beam should not be applied incrementally, but fully present during non-linear analysis, by implementing code suggested in Section 6.6.

8.2 Results from FABRICFORMER for keel mold fabric formed beams

The first succesful run of FABRICFORMER used the properties for B25 as did the first run of BESO. Non-linear analysis for reinforcement was not included due to expected savings on computational time. The program used the input values as specified in Table 8.2.

TABLE 8.2 Properties for FabricFormer run

Concrete B25		
Modulus of elasticity	E_c	28.485 N/mm^2
density	ρ	2400 kg/m^3
Beam dimensions and FF parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
ANSYS element smartsize		grade 10 (finest)
fabric stiffness	EA	490 N/mm
time increment (DR)	Δt	0.100 m
load		5 kN/m
population size	N_p	10
crossover	Cr	0.2
scale factor	F	0.5

During the run, the program had to be paused for various reasons, while simultaneously some improvements pertaining to memory management were implemented. It is therefore difficult to indicate how long this run actually took. Subsequent runs will provide more reliable estimates for the computational time.

The graph in Figure 8.6 shows how the program converged to a solution. No stopping criteria were defined, so the program was manually stopped at 236 generations.
After 200 generations it became clear that the solution no longer improved significantly, though the resulting shape was not as optimal as expected. For the remaining generations the population size was doubled, using 10 new population members with all new values. This caused the perturbations in the graphs, and did not produce very different convergence behavior.

The beams shown in Figure 8.7 are both best solutions, after 41 and 177 generations respectively. The performance graph shows that the performance index has improved significantly, while visual inspection suggests that improvement is limited, as the height at midspan should be the highest (later found to be due to a bug, which was corrected). Also, the cross-sections are very slim indeed. This observation would suggest that volume has a relatively high influence on the performance index so for new runs the load should be increased so the influence of strain energy will increase relatively. Another improvement would be to increase the width of the area at the top through which the distributed load is applied, effectively forcing wider beams to evolve.

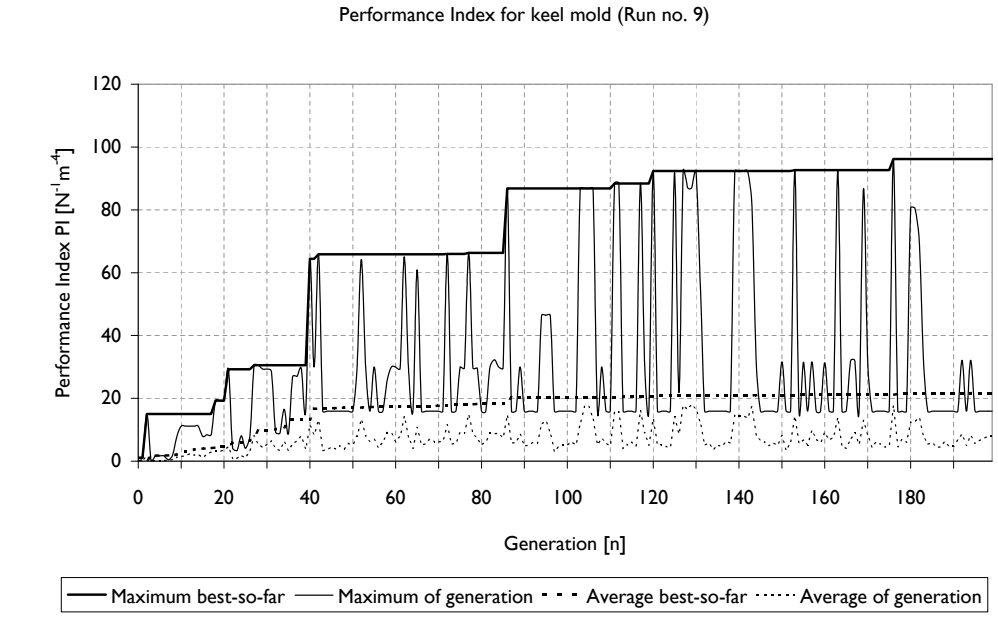


Figure 8.6
Performance index over time for Run no. 9.

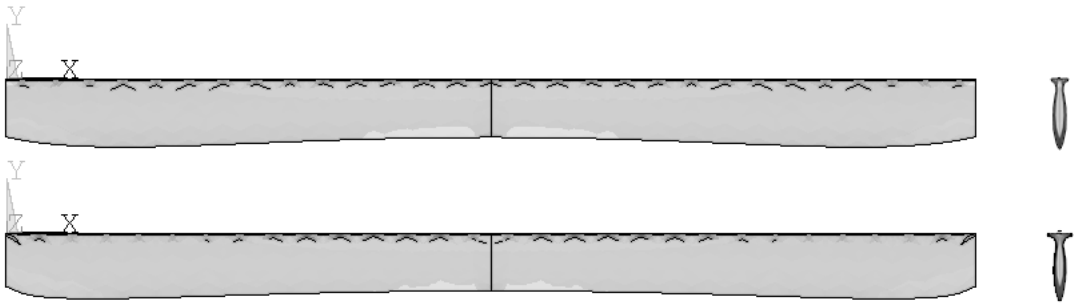


Figure 8.7
Front and side views of the optimized beam at generations 41 and 177.

Furthermore, the average new performance index – which is the performance of the 10 new beams that is compared to the current best 10 solutions – remains extremely low. After 30 generations, only one population member supplies the best solutions. Because of this, it would seem that the crossover and scale factor should be changed. For subsequent runs a scale factor of 0.1 (i.e. closer proximity of new solution vectors) was chosen for stability.

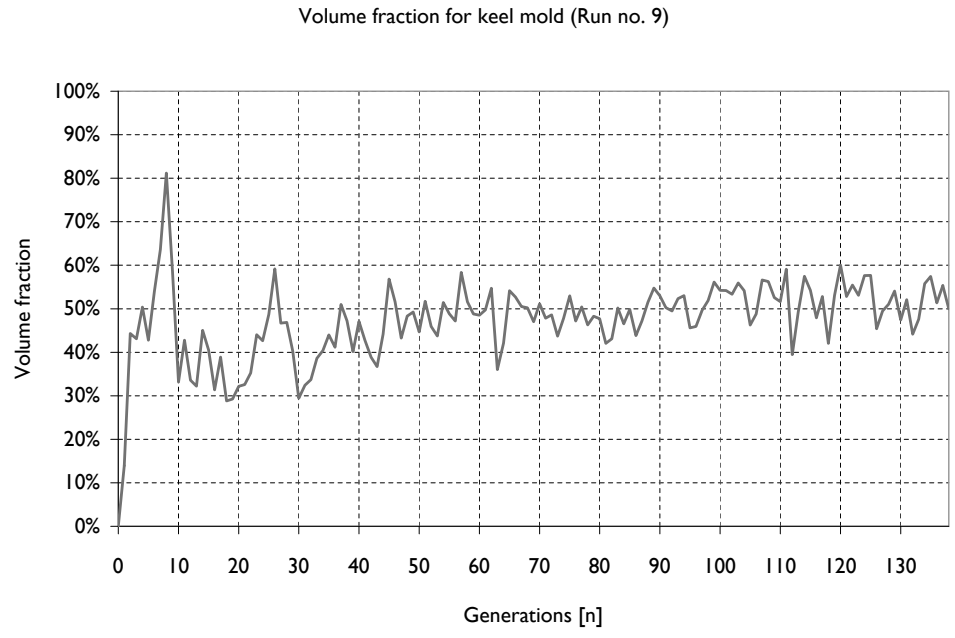


Figure 8.8
Volume fraction over time for Run no. 9.

The second succesful run of FABRICFORMER used the properties for B65. Non-linear analysis for reinforcement was not included due to expected savings on computational time. The program used the input values as specified in Table 8.3.

TABLE 8.3 Properties for FabricFormer run 33

Concrete B65		
Modulus of elasticity	E_c	39.394 N/mm²
density	ρ	2400 kg/m³
Beam dimensions and FF parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
ANSYS element smartsize		grade 10 (finest)
DR nodes (rows x columns)		30 x 12
fabric stiffness	EA	490 N/mm
time increment (DR)	Δt	0.100 m
load		1.6 kN/m
population size	N_p	50
crossover	Cr	2/D
scale factor	F	0.3

This run took about 7 days and 16 hours.

The graph in Figure 8.9 shows how the program converged to a solution. No stopping criteria were defined, so the program was manually stopped at 165 generations. The convergence graph shows a much cleaner convergence than that of Run no. 9 in Figure 8.6. This was due to a bug that was removed, which also caused the previous geometry to be visibly, clearly sub-optimal.

The beam shown in Figure 8.10 is the best solution, after 141 generations. The top image shows the free mesh that ANSYS automatically generated. The bottom two views show the stress distribution in the x-direction, clearly showing how the stresses have been nicely distributed due to the optimized geometry. The discrepancy at midspan is due to some inaccuracy in the symmetry plane during Dynamic Relaxation, one that was not solved yet at that time. However, this discrepancy occurred in every generated solution, likely having no relevant effect on the optimization process.

The geometry is certainly something one might expect, offering a curved shape that follows the bending moment and retaining a certain height to cope with shear forces.
The cross section is still very narrow as the load was not yet increased to influence the Performance Index differently.

From this point on, the focus was shifted towards implementation of the pinch mold method, as this result was deemed satisfactory within the context of this thesis and its objectives.

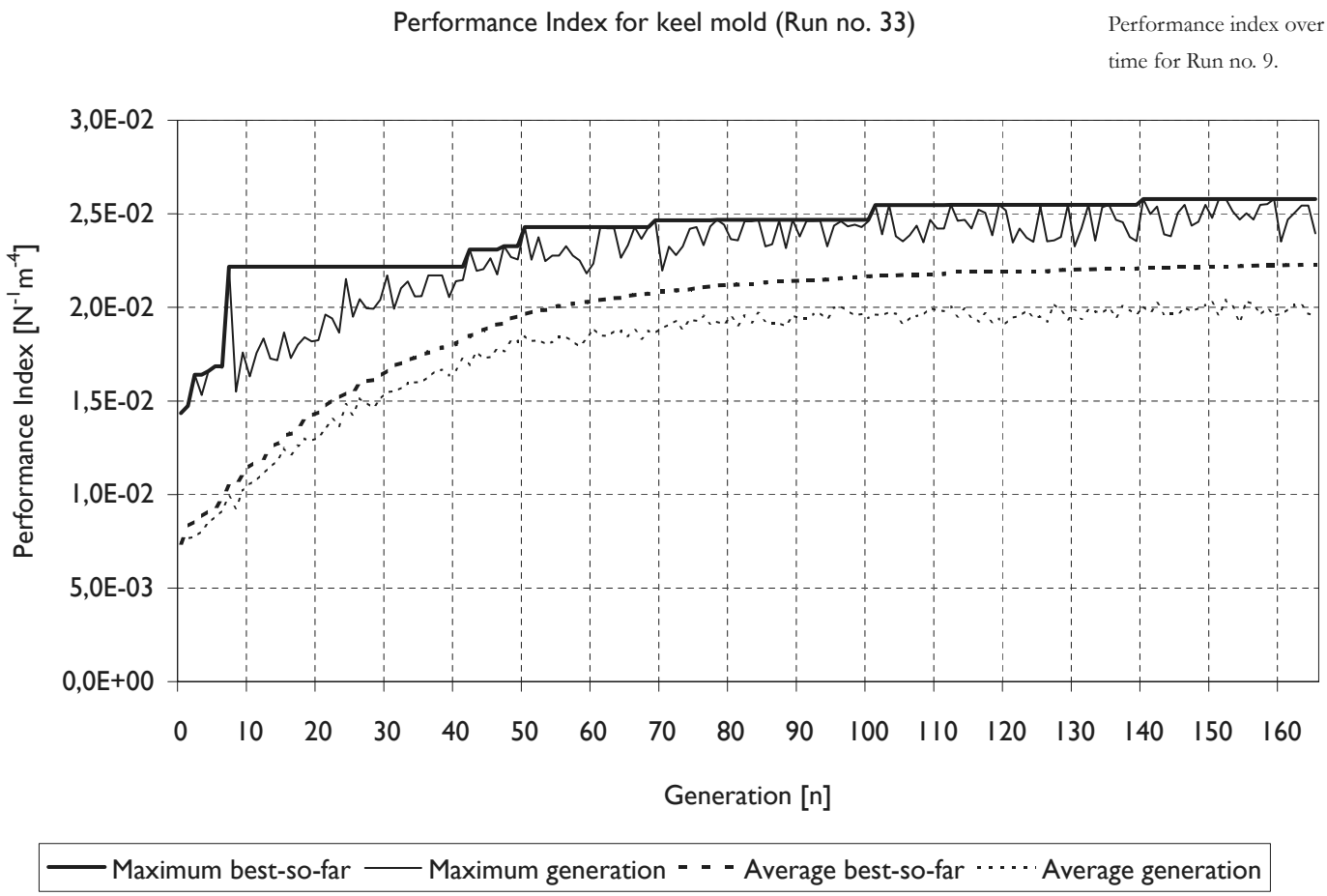


Figure 8.9
Performance index over time for Run no. 9.

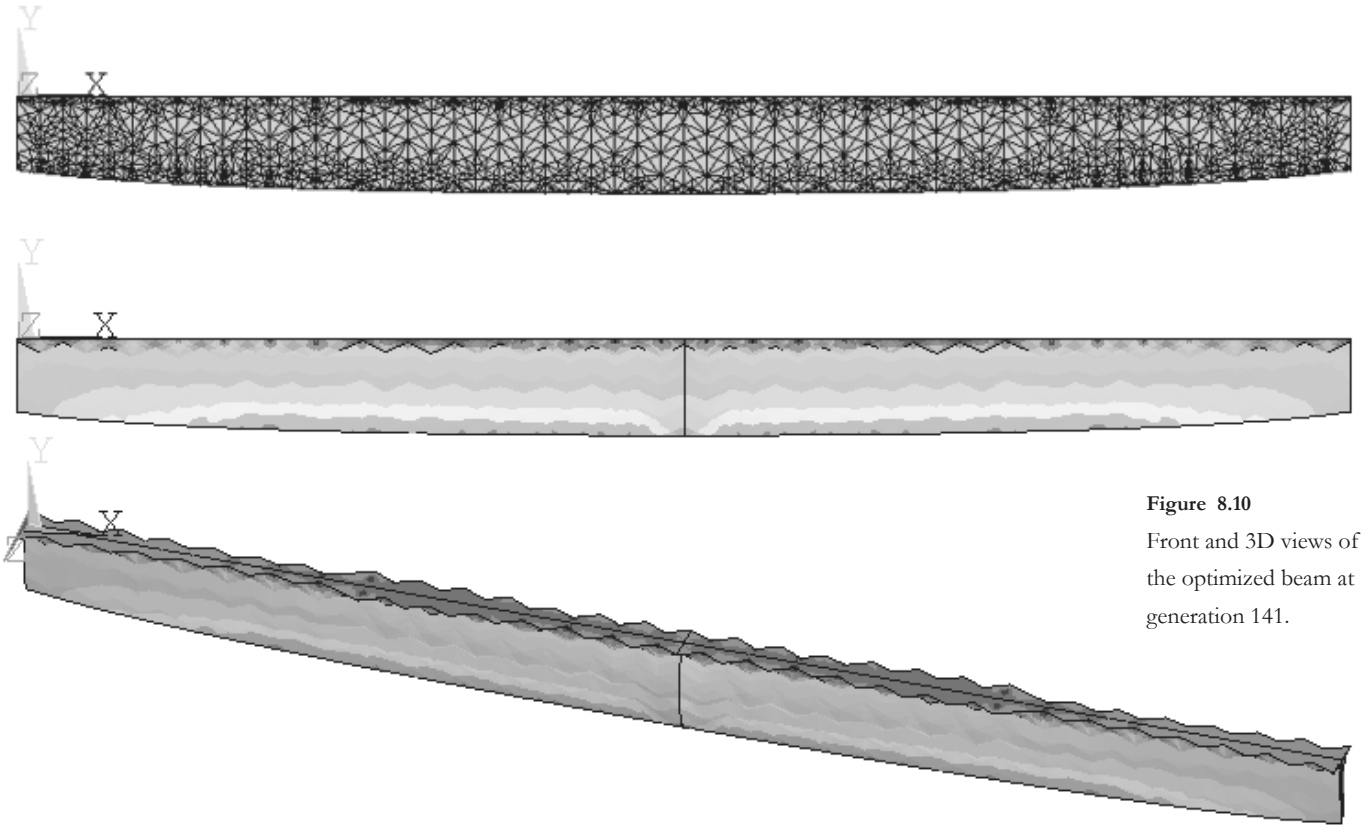


Figure 8.10
Front and 3D views of the optimized beam at generation 141.

8.3 Results from FabricFormer for pinch mold fabric formed beams

The third succesful run of FABRICFORMER used the properties for B25. Non-linear analysis for reinforcement was not included due to expected savings on computational time. The program used the input values as specified in Table 8.4.
This is the first succesful run for the pinch mold method.

TABLE 8.4 Properties for FabricFormer run 67

Concrete B25		
Modulus of elasticity	E_c	28.485 N/mm²
density	ρ	2400 kg/m³
Beam dimensions and FF parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
ANSYS element size		0.050 m
DR nodes (rows x columns)		30 x 12
fabric stiffness	EA	490 N/mm
time increment (DR)	Δt	0.100 m
load		1.5 kN/m
population size	N_p	50
crossover	Cr	2/D = 0.04
scale factor	F	0.4 (jitter)

This run took about 6 days and 20 hours.

The graph in Figure 8.11 shows how the program converged to a solution. No stopping criteria were defined, so the program was manually stopped at 278 generations. The convergence graph shows a clean convergence.

The beam shown in Figure 8.12 is the second best solution, after 147 generations. Unfortunately, the best result was not stored, because the harddrive where the results where saved was full, which was only discovered afterwards.
The meshing method was changed to use cubic elements to improve both mesh quality and speed (see Chapter 6.5).

The geometry seems reasonably optimal, following the bending moments, retaining a tension chord, and removing material in the center of the cross-section. However, compared to results from both linear and non-linear BESO, the holes are too large, and the optimization process did not lead to multiple pinch points. The program is able to create at most six holes, but the process led to lumping. The conclusion was made that the pinch points, and how they are generated and optimized, need to be interdependent, which is only the case to a limited degree (as explained in Section 8.6). This addition would require significant programming for which there was no time left during the thesis project.
A higher crossover value for the parameters that determine the pinch point is also recommended.
As in previous runs, the beams is extremely narrow. To counteract this, either the load could be changed, or the method of calculating the performance index could be altered.

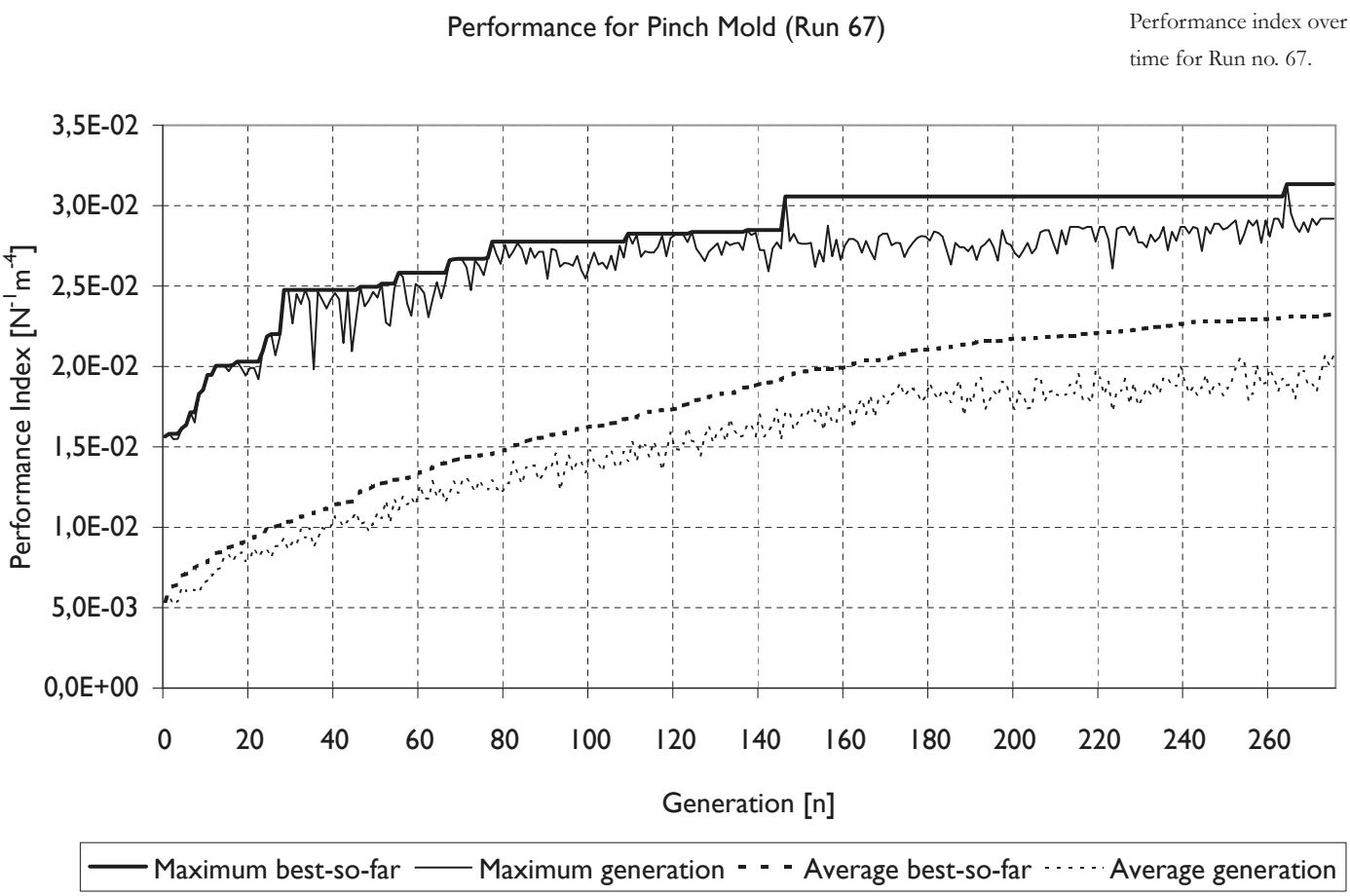


Figure 8.11
Performance index over time for Run no. 67.

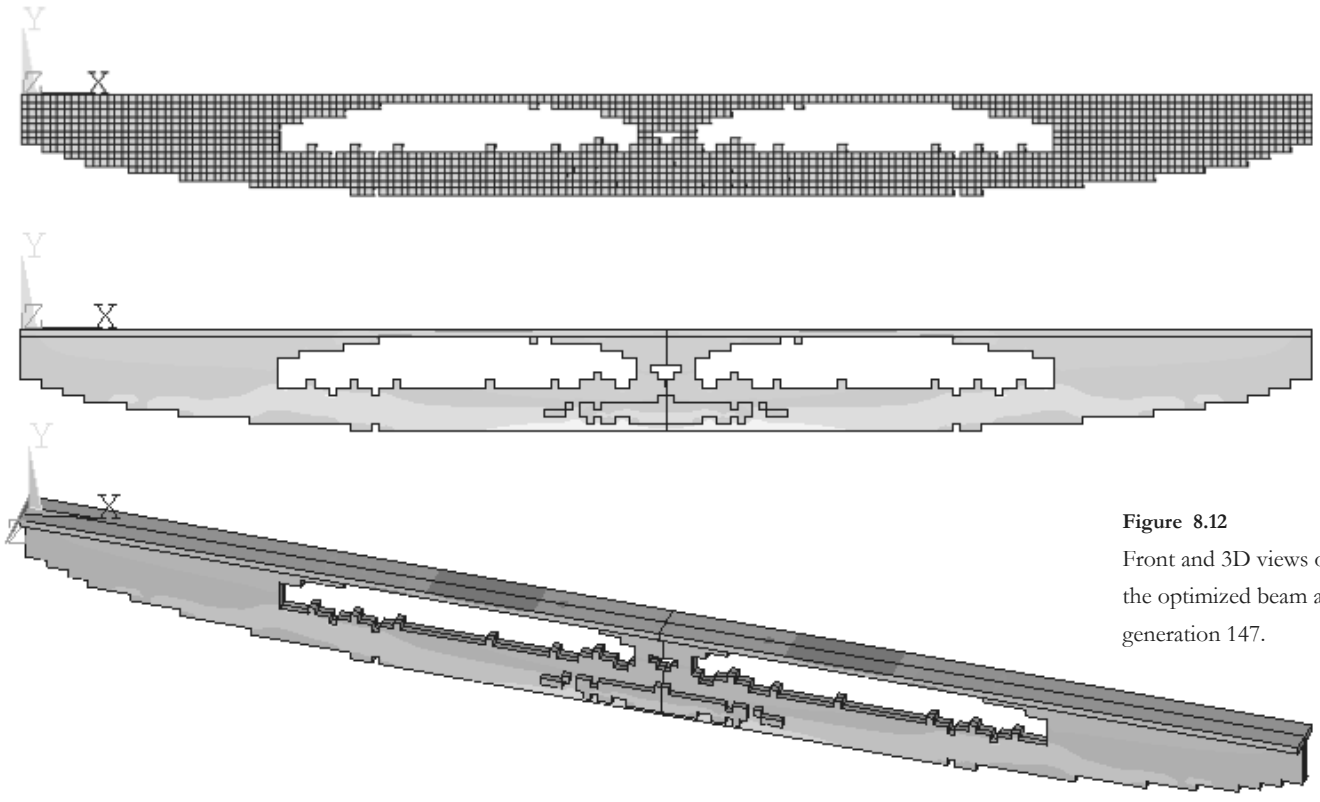


Figure 8.12
Front and 3D views of the optimized beam at generation 147.

The fourth succesful run of FABRICFORMER used the properties for B25. Non-linear analysis for reinforcement was not included due to expected savings on computational time. The program used the input values as specified in Table 8.5. This is the second succesful run for the pinch mold method.

TABLE 8.5 Properties for FabricFormer run 72

Concrete B25		
Modulus of elasticity	E_c	28.485 N/mm^2
density	ρ	2400 kg/m^3
Beam dimensions and FF parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
ANSYS element size		0.025 m
DR nodes (rows x columns)		60 x 24
fabric stiffness	EA	490 N/mm
time increment (DR)	Δt	0.100 m
load		1.5 kN/m
population size	N_p	50
crossover	Cr	0.2
scale factor	F	0.4 (power law)

This run took about 23 days and 20 hours!

For this run the computer was left running while using a finer mesh for both DR and ANSYS. It was thought that perhaps a higher accuracy of the calculations might result in better convergence and avoid the lumping of the pinch points as in the previous Run no. 67.

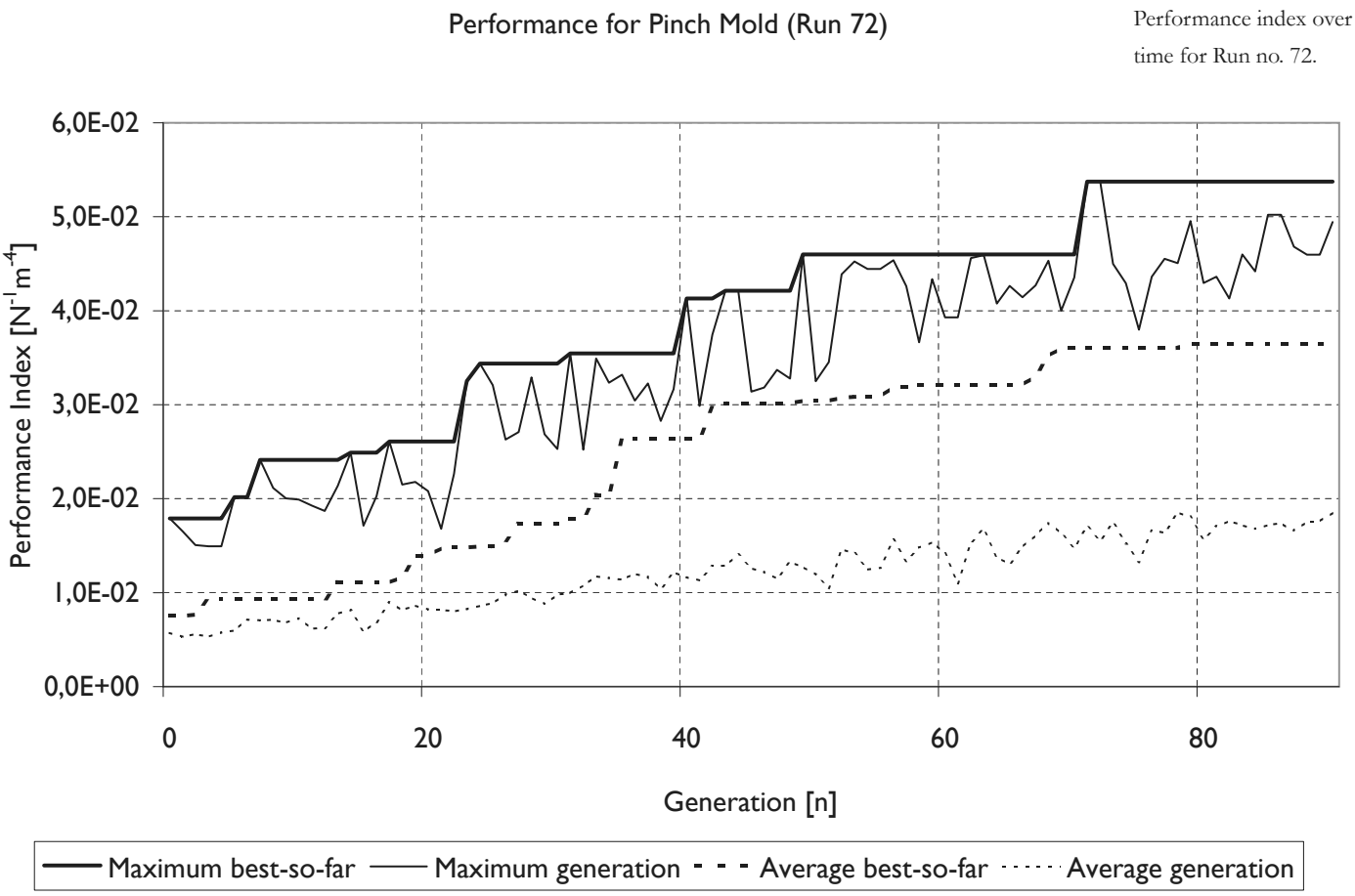
The graph in Figure 8.13 shows how the program converged to a solution. No stopping criteria were defined, so the program was manually stopped at 69 generations. The convergence graph shows that convergence occurred, but would probably have continued for some time if one compares the shape to that of previous runs.

The beam shown in Figure 8.14 is the best solution, after 72 generations. Unfortunately, the similarity of this beam with that of the previous Run no. 67 clearly shows that using a finer mesh does not improve convergence with respect to the pinch points. The recommendation of introducing interdependancy of the seperate pinch points still stands. A higher crossover value for the parameters that determine the pinch points is also still recommended.

Once more, the conclusion is drawn that the optimized beams have become too narrow, due to the – currently – linear nature of the optimization. Such beams would be vulnerable to flexural buckling and placing reinforcement might prove difficult as there is limited space in the cross-section. The optimization process needs to be adapted to prevent these narrow geometries:

- Increase the load to bias any gains in stiffness over reductions in volume (i.e. the Performance Index is defined as stiffness over volume), or
- introduce maximum stresses and strains to cut off the linear analysis at some point. These stresses and strains need to be properly defined for the result to be meaningful, or
- implement the non-linear analysis described in Chapter 6.

Figure 8.13
Performance index over time for Run no. 72.



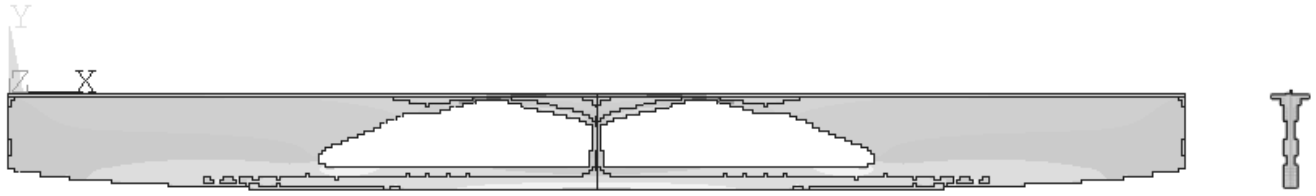


Figure 8.14
Front and side views of the optimized beam at generation 72. As before, there is only one mirrored pinch point.

The following run ran very shortly to assess the effects of acting upon the following three recommendations made earlier:

- Seeding the coordinates of the pinch points based on the BESO algorithm results
- Increasing the load to favor stiffness over volume in the Performance Index.
- Using a high crossover factor which is suggested for interdependant optimization variable i.e. the pinch points.

The first recommendation is clarified in Figure 8.15 where the results of BESO in Chapter 2.5 for a beam with supports at the top are shown. Several of the intermediate results of that BESO optimization have been overlayed to show how the holes in the cross-section vary. Based on that image a general range has been specified for the possible coordinates of the first, second and third pinch points. These in turn have been used in the initialization phase of the Differential Evolution to generate to the pinch point coordinates of the first generation of fifty beams.

The program used the input values as specified in Table 8.6. Note that based on the second recommendation, the load has been increased threefold.

TABLE 8.6 Properties for FabricFormer run 82

Concrete B25		
Modulus of elasticity	E_c	28.485 N/mm^2
density	ρ	2400 kg/m^3
Beam dimensions and FF parameters		
length		9.000 m
domain width		0.300 m
domain height		0.800 m
ANSYS element size		0.025 m
DR nodes (rows x columns)		60 x 24
fabric stiffness	EA	490 N/mm
time increment (DR)	Δt	0.025 m
load		4.5 kN/m
population size	N_p	50
crossover	Cr	0.9
scale factor	F	0.4 (power law)

In addition to seeding the pinch point variables, the variables determining the keel shape, edge shape and prestressing have all been taken from the previous run, to improve the initial results of this run. The beam shown in Figure 8.16 is the best solution, after just 5 generations.
What is most obvious is that the cross-section makes full use of the design domain width and would be a far more practical beam to manufacture than previous results.

It would still be recommended to add non-linear analysis or define maximum strains or stresses in the linear analysis. This would avoid the necessity of changing the applied load until the result would be satisfactory. Instead, if these linear ultimate strains or stresses are defined properly, a load would only have to be very high but not precise.

As before, changing the interaction of the pinch points explicitly within the optimization process would also be preferable, as seeding the pinch points coordinates based on BESO results has not resulted in multiple sets of pinch points.

The effect of the higher crossover factor Cr is difficult to evaluate as these results are only fifth generation (last results prior to drafting this report).

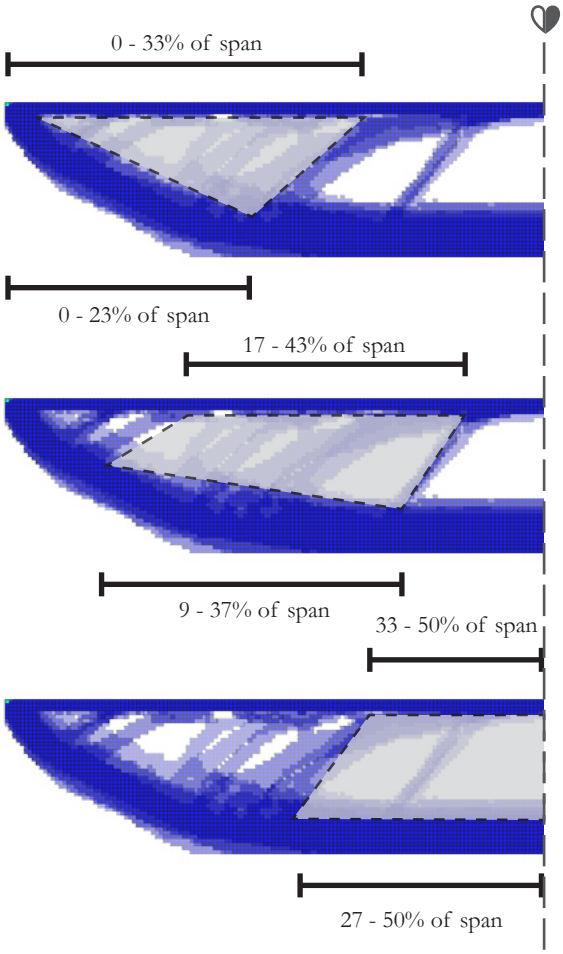


Figure 8.15
Transparant overlays of intermediate BESO results from CHapter 2.5 for one particular support case. The domains of three possible holes in the geometry are defined as ranges for the top two and bottom three coordinates.

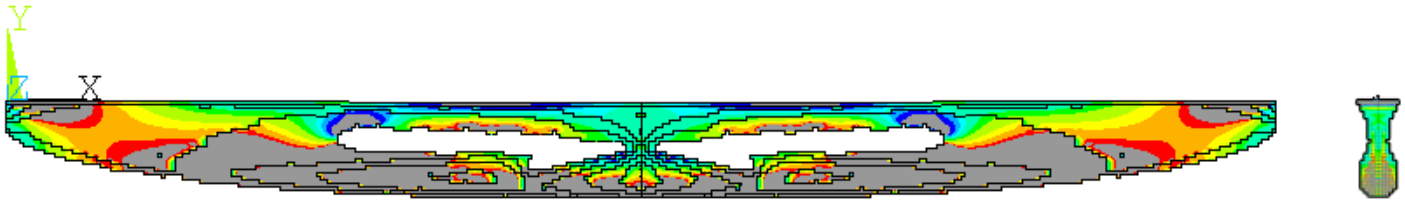


Figure 8.16
Front and side views of the optimized beam at generation 5. As before, there is only one mirrored pinch point.

8.4 Comparison of results based on linear analysis

To verify the results from FABRICFORMER a comparison can be made with rectangular beams. The expected improved performance of the fabric formed beams can be measured by calculating the deflection at midspan for a specific load. This calculation can be done by re-entering our data in ANSYS and applying said load. Doing so gives the following deflections:

TABLE 8.7 Results from linear analysis

Run 33		
Concrete volume	V	0.49 m³
Midspan cross-section	A	0.102 m²
Maximum height	h	0.648 m
Average width/height ratio at midspan	$b_{avg} / h = A / h^2$	0.24
Averaged cross-section	$A_{avg} = V / 9$ m	0.054 m²
External load + averaged self weight	$q = 1500 + \rho g A_{avg}$	1500 N/m + 1283 N/m
Deflection at midspan	δ	5,0 mm

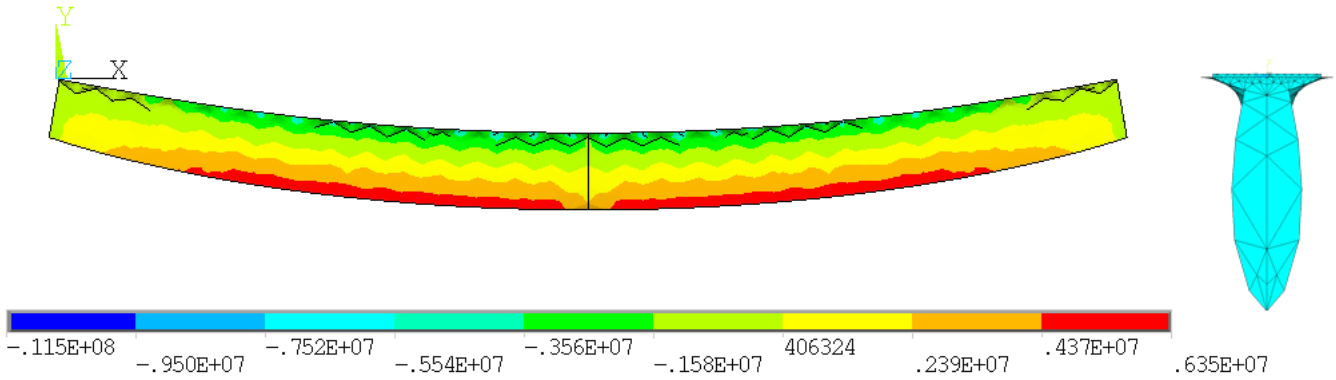


Figure 8.17
Front view and cross-section of the optimized beam at generation from Run 33, subjected to loading. Resulting stresses in x-direction are in [N/m²]

TABLE 8.8 Results from linear analysis

Run 67		
Concrete volume	V	0.52 m³
Midspan cross-section	A	0.100 m²
Maximum height	h	0.7 m
Average width/height ratio at midspan	$b_{avg} / h = A / h^2$	0.20
Averaged cross-section	$A_{avg} = V / 9$ m	0.058 m²
External load + averaged self weight	$q = 1500 + \rho g A_{avg}$	1500 N/m + 1367 N/m
Deflection at midspan	δ	1,1 mm

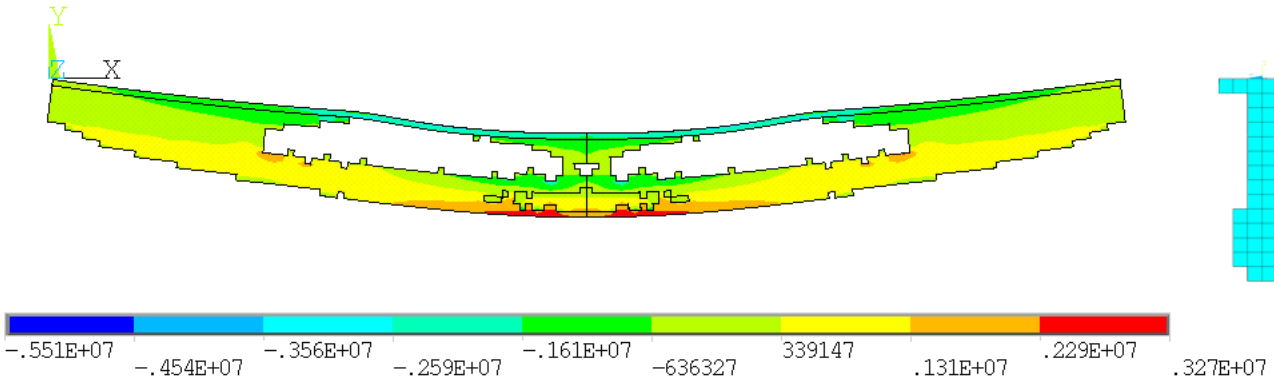


TABLE 8.9 Results from linear analysis		
Run 72		
Concrete volume	V	0.33 m³
Midspan cross-section	A	0.069 m²
Maximum height	h	0.725 m
Average width/height ratio at midspan	$b_{avg} / h \; w = A / h^2$	0.13
Averaged cross-section	$A_{avg} = V / 9$ m	0.036 m²
External load + averaged self weight	$q = 1500 + \rho g A_{avg}$	1500 N/m + 851 N/m
Deflection at midspan	δ	0,9 mm

Figure 8.18
Front view and cross-section of the optimized beam from Run 67, subjected to loading. Resulting stresses in x-direction are in [N/m²]

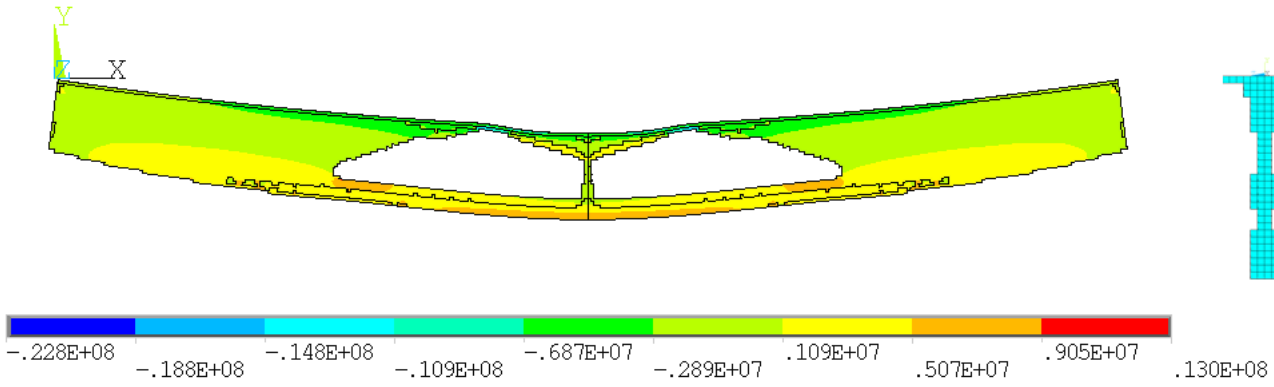


TABLE 8.10 Results from linear analysis		
Run 82		
Concrete volume	V	0.49 m³
Midspan cross-section	A	0.129 m²
Maximum height	h	0.675 m
Average width/height ratio at midspan	$b_{avg} / h = A / h^2$	0.28
Averaged cross-section	$A_{avg} = V / 9$ m	0.054 m²
External load + averaged self weight	$q = 1500 + \rho g A_{avg}$	1500 N/m + 1280 N/m
Deflection at midspan	δ	1.7 mm

Figure 8.19
Front view and cross-section of the optimized beam from Run 72, subjected to loading. Resulting stresses in x-direction are in [N/m²]

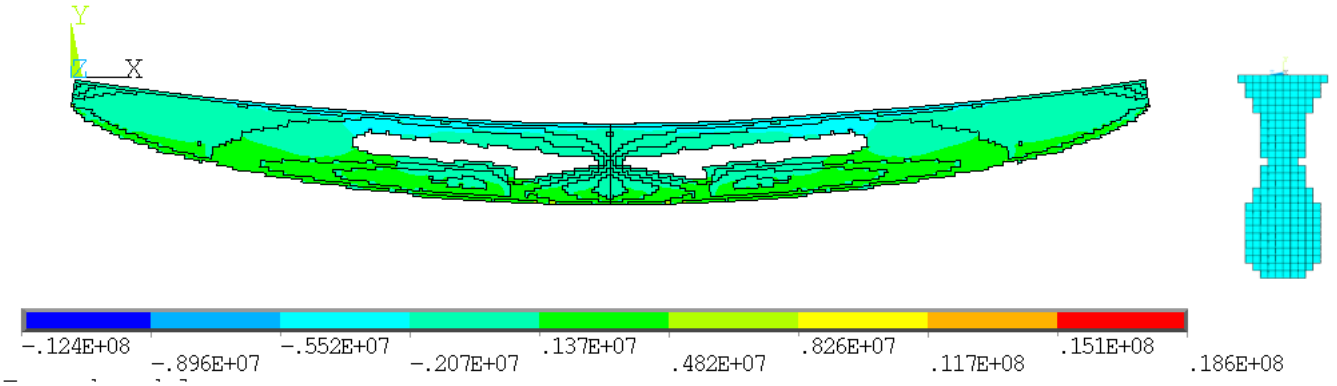


Figure 8.20
Front view and cross-section of the optimized beam from Run 82, subjected to loading. Resulting stresses in x-direction are in [N/m²]

Before making the comparison, it is noted from all three results that the height is not 0,80 m, the maximum height of the design domain. It would be expected that the optimization would definitely maximize the height to gain the highest stiffness. In the first result, the keel mold beam of Run 33, it is likely that optimization would eventually have led to such a maximization and the reader is reminded that only one of twelve optimization parameters determines the height at midspan.

In the other three results, the algorithm actually did find a form that was 0,80 m in height, but unfortunately the approximations made in translating the fabric mesh to the ANSYS block mesh (see Chapter 6.5) resulted in the lower volumes to be approximated to zero blocks, instead of one. A possible improvement would be to round up the geometry of the fabric mesh, resulting in – on average – a larger block mesh volume.

The same load q is used in standard equations for rectangular beams, as well as the modulus of elasticity E for concrete and the chosen span of 9 m. The averaged value A is used, while its width over height ratio, b/h is varied to see how different ratios compare to the fabric formed results. The deflection for a simply supported beam, subjected to an evenly distributed load, is calculated using:

$$\delta = \frac{5}{384}\frac{ql^4}{EI} \tag{8.1}$$

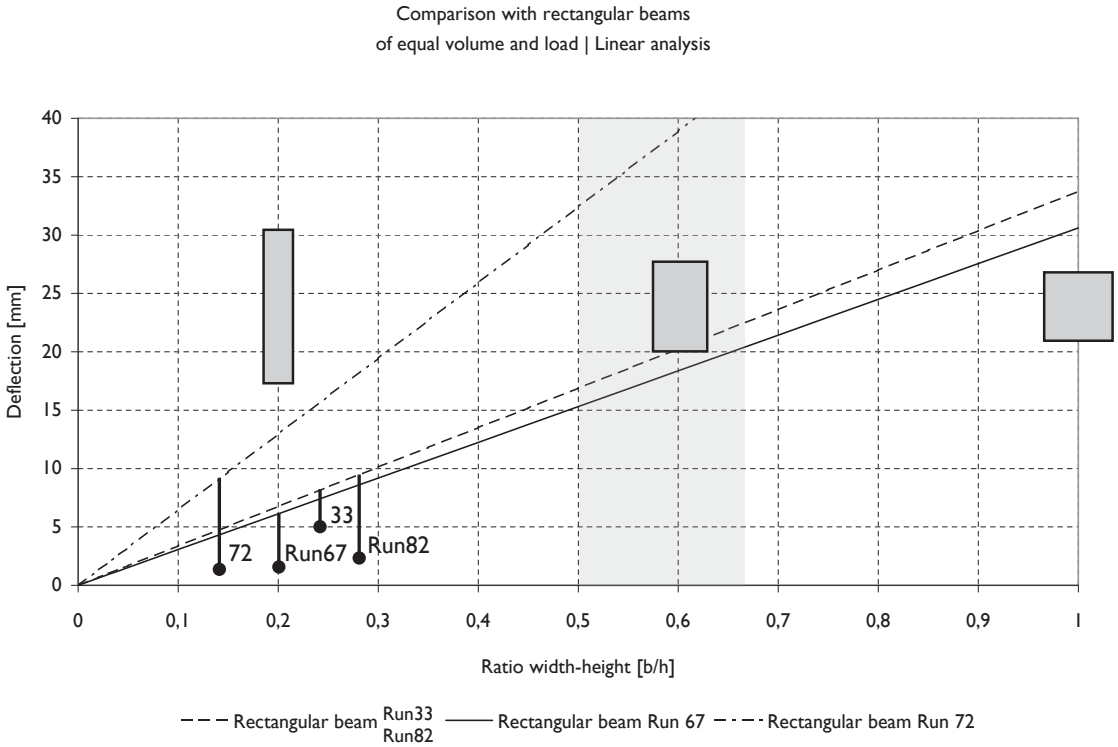


Figure 8.21
Graph showing the deflection of the optimized beams compared to rectangular beams of equivalent volume with varying width over height ratios b/h.

where

$$EI = 28.485 \cdot \frac{1}{12} bh^3$$
$$q = 1500 + \rho_c gbh$$
$$l = 9$$

and b is the width, h is the height, concrete density ρ_c = *2400 kg/m³* and gravity constant g = *9,81 N/kg*.

Equation 8.1 is plotted for all three equivalent volumes and varying b/h ratios in the graph in Figure 8.21. The deflections and averaged midspan b_{avg}/h ratios of the three fabric formed result are plotted as points with a vertical line emphasizing the difference in deflection. From this graph it may be concluded that for linear analysis all fabric formed, optimized results have a more efficient volume distribution compared to rectangular beams with respect to resulting deflections. Also, the pinch mold beams give larger gains compared to the keel mold beam. The reductions in deflection at midspan are 38%, 82%, 89% and 91% for Runs 33, 67, 72 and 82 respectively for equal volume and b/h ratio. Comparing the solutions with rectangular beams of equal volume and height h, offers a different picture, as the reductions in deflection become -15%, 75%, 80% and 78% respectively. This suggests that the keel molded beam would benefit from additional optimization, most likely in the cross-section (Figure 8.17), where a lot of material is present at the center of the height. Here, where the neutral axis is, the material will contribute poorly to the stiffness in terms of the cross-sectional moment of inertia.

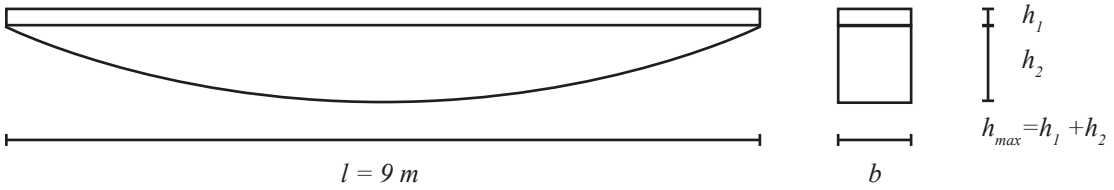
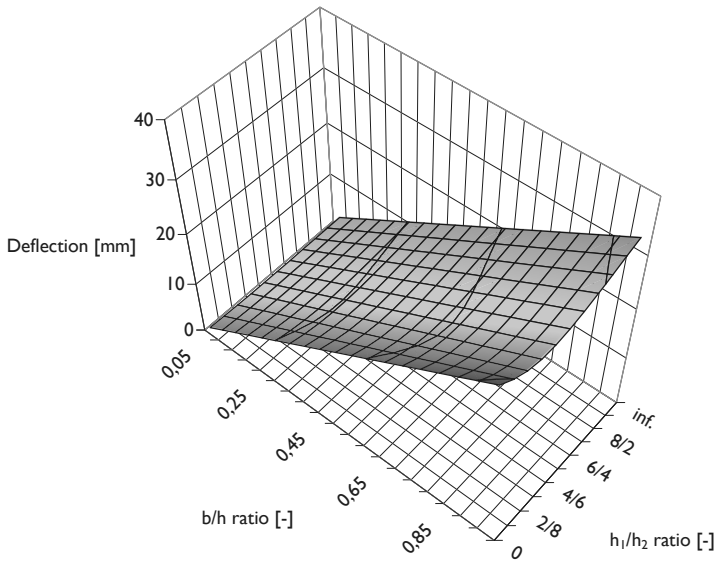


Figure 8.22
Combined rectangular and parabolic cross-section. Appendix E shows general equations to calculate the deflection depending on the h_1/h_2 ratio.

One final comparison is made between the result of Run 82 and a beam that is parabolically shaped with a rectangular cross-section (see Figure 8.22). This is a beam that more optimally uses its volume to achieve stiffness and may very well be cast using conventional methods. Some optimum exists between the ratio h_1/h_2, as that would allow the beam to both deal with shear forces and bending moments effectively. Some calculations in MAPLE were done to calculate the deflection due to distributed loading for different values of h_1/h_2 (see Appendix E).
The results are plotted in Figure 8.23 for a rectangular beam with the same load and span as Run 82 and constant volume. The plot reveales an optimum where deflection is 12% less than that of a parabolic or rectangular beam of **equal b/h_{max} ratio**, from which was concluded that the optimal ratio h_1/h_2 is about 4/6. Another interesting conclusion is that the difference in deflection of a parabolic beam is a fraction lower than that of a rectangular beam (less than 1%) of **equal b/h_{max} ratio**.
The beam corresponding to the optimal ratio h_1/h_2 of 4/6 is compared to the result of Run 82. For equal b_{avg}/h_{max} ratio, the deflection is 89% less. This demonstrates that gains from optimizing the longitudinal shape are small when compared to those of optimizing the cross-section. Thus the conclusion may be drawn that from a structural point of view, the pinch mold is far more interesting to develop than the keel mold.

Figure 8.23
Deflection plotted as a function of the b/h ratio and h_1/h_2 ratio, showing an optimum near h_1/h_2 = *4/6*



Note: for constant volume, when instead of the b/h_{max} ratio the height h is fixed, the rectangular beam ($h_2 = 0$) has the lowest deflection, vice versa when width b is fixed.

8.6 Conclusions and recommendations

All results that have been optimized based on linear analysis deflect less at midspan compared to rectangular beams of equivalent volume. Beams made with the pinch mold are significantly more stiff than those cast in the keel mold.

The optimization runs take a lot of time. It is advised to improve both the software and hardware, as has been discussed in more detail in the recommendation of the previous chapters.

Evolutionary parameters, i.e. the dimension D, crossover and scaling factors Cr and F are difficult to determine. With respect to the crossover factor, it would be recommended to tailor it to each optimization variable; choosing a low factor of about 0.2 for independent parameters, and 0.9 for interdependant parameters. The dimension of 50 is most adequate for keel mold optimization i.e. 12 parameters, but should probably be higher for the pinch mold optimization of 50 parameters.

The pinch mold optimization results are not fully optimal as the expectation was that multiple pinch points would be optimal yet all results show lumped single pinch points. The problem most likely lies in the lack of interdependancy of pinch points. For example, given two pinch points (Figure 8.24a), should one pinch point start to overlap another, the program does not take action to counter this (Figure 8.24b). Ideally it would change the geometry of the second pinch point to follow the changes of the first (Figure 8.24c). Since, this effect was observed at a late stage in the project, there was no time to program such behaviour.

Finally, the analysis should include one of the following: (1) a well tuned load to provide beams of sufficient width, (2) maximum stresses or strains at which linear analysis is cut off, or (3) non-linear analysis with real limits for ultimate stresses and strains.

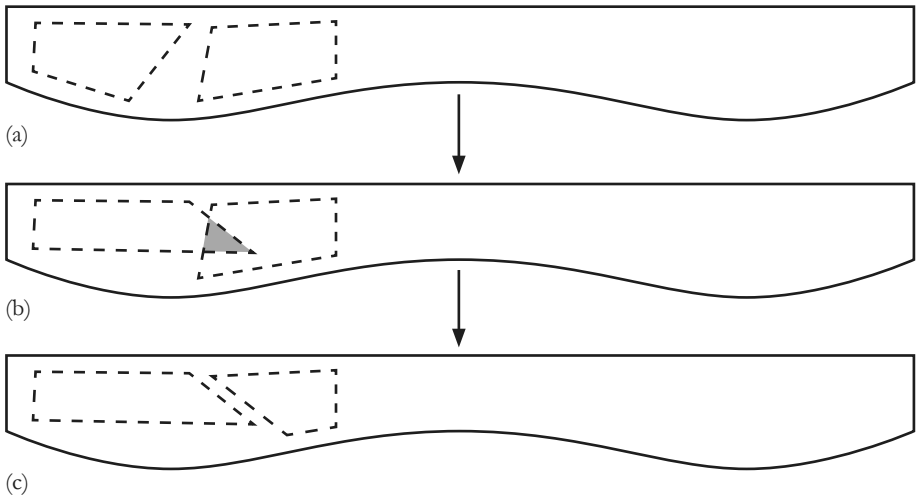


Figure 8.24
The current program allows overlap of pinch points, and no action is taken (b). Based on the results it was concluded that the program should counteract this as in (c).

CHAPTER 9 | Conclusions and recommendations

The various elements and algorithms that this thesis project is built on have never been combined. This means that various observations were made as a result of their interaction. To organize these observations and their subsequent conclusions, each of the previous chapters ended with a set of specific conclusions and recommendations. This chapter attempts to draw more general conclusions by assessing the original objectives, discussing improvement of both the FABRICFORMER program and its results. At the end a summary is given of each set of chapter conclusions and recommendations before providing an overview of the main thesis conclusions and recommendations.

9.1 Achievement of the objectives

At this point the question remains whether the objective of this thesis has been achieved, and if so, to what degree. The objectives, stated in Chapter 1 are once again reiterated:

To bridge the gap between computational optimization and manufacturability.

And specifically applied to the following cases:

(1) *To develop a computational tool to optimize fabric formed concrete structural elements with respect to material efficiency and manufacturability.*

(2) *To computationally optimize the shape of a fabric formed concrete structural element with respect to material efficiency and manufacturability.*

Without any doubt it can be said that the primary objective has been reached. Had computational optimization previously been associated with geometries that could not be manufactured easily and economically, now it has been demonstrated that optimization can take manufacturing into account. Most, if not all examples of optimization algorithms extend no further into the engineering practice than being used as early stage design tools. The gap between computational optimization and manufacturability has been bridged, but to what extent? To answer this, it is best to evaluate the two secondary objectives.

(1) The development of a computational tool was achieved, as FABRICFORMER is a fully functional software program.
In contrast FABRICFORMER, the software that was developed during the course of this thesis, has successfully combined:

- constraints of manufacturing (fabric form finding i.e. Dynamic Relaxation)
- computational optimization (Differential Evolution)
- structural analysis (ANSYS finite element analysis)

Of course, more work on FABRICFORMER is certainly needed, if – for example – it is to have practical appeal in the market place. The major points of improvement of FABRICFORMER are:

- more realistic fabric modeling (Chapter 5.6)
  - o i.e. bi-directional model with warp and weft direction (Appendix C)
  - o i.e. non-linear material model derived from biaxial testing
- non-linear analysis of reinforced concrete (modeled, not yet implemented, Chapter 6)
- provisions for detailing of the supports

(2) The results of FABRICFORMER are promising, but remain inside the realm of the computer. Unfortunate-

ly, because the optimization process did not include non-linear evaluation of the solution, the results are not practical yet, requiring non-linear analysis for verification as well as reinforcement calculations.
On the other hand, relatively little effort is needed to successfully translate the results to actual fabric formwork, especially compared to anything up to this point. Most of the geometric data on the fabric, the mold and the resulting beam is known. The prestressing forces required to reproduce the beam in reality can be derived from the program.

In addition, there is confidence that increased computational power could easily produce better, more optimal results. Not only will the program be allowed to run faster, but the user may react earlier to intermediate results, affecting changes more rapidly to steer the algorithm in the right direction.

All in all, the secondary objectives have been achieved, while the first objective of developing a tool has been met to a higher degree than the second of producing truly manufacturable results.

9.2 Potential for further optimization of the results

Fabric formwork presents a novel method of casting concrete and although the amount of research on the topic is steadily increasing, information on practical applications of fabric formed structural elements remains scarce. This lack of information led to the decision to solely optimize for mechanical properties in FABRICFORMER as has been customary in Evolutionary Structural Optimization (ESO). As mentioned in Chapters 2 and 4, the objective function is stiffness over volume, where stiffness is quantified by calculating the strain energy.
It is unclear how other considerations should come into play and how these should figure into a quantified objective function in a general way. Of course, for a specific case, where certain demands are put on, for example, the esthetics of the geometry these can be incorporated in either the constraints, or the objective function. When esthetic qualities can be quantitatively derived from the geometry for the objective function, it is a simple matter to add them as a consideration in FABRICFORMER. In this respect, FABRICFORMER is a flexible program whereas current algorithms such as ESO are not.
Another aspect often considered is cost. Naturally the cost is implicitly taken into account by the attempt to minimize concrete volume. However, the actual cost of fabric forming is also dependant on the mold which consists of solid timber or steel parts and fabric. The amount of these materials is dependant on the type of mold technique (spline, keel or pinch), the weight of the concrete and the size and location of the prestressing forces. Once fabric forming progresses towards full-scale production and economies of scale are attained, insight can be gained in expected investments i.e. material and labor costs. At that point, cost can be included in the optimization as well.

9.3 Implications of FABRICFORMER

The fact that FABRICFORMER has proven to viably combine optimization and manufacturability may have the following implications for future developments in structural engineering:

- Manufacturability might more readily be included explicitly in optimization processes, while relying less on post-processing to include such considerations.
- Computational optimization might gain broader appeal within the structural engineering community, viewed less as a limited, early stage design tool and more as software able to encompass a larger part of the design and engineering process.
- Future software may be developed that uses computational optimization as a central and integral component, rather than the current situation where some mathematics and engineering analysis programs offer it as a modular component, heavily dependant on programming on the part of the user for its use.

9.4 General strategy for optimization with manufacturability

The combination of optimization and manufacturability remains specific to the manufacturing method that has been chosen. For example, FABRICFORMER, though it might be expanded to include different types of elements, will not function for anything else than fabric formwork technology (or possibly tension structures). To provide recommendations for any future work within this subject, but not specifically related to fabric formwork, the following list of steps is given outlining the design process of this thesis in general terms:

1) Choose structural material(s) and manufacturing technique
2) Inventory any constraints and design domain posed by this technique
3) Identify aspects within the technique that are variable and can be optimized i.e. the design variables
4) Identify objectives for optimization and how they relate i.e. the objective function
5) Choose an optimization algorithm including appropriate optimization parameters with respect to the design variables and how they interact i.e. are dependant or independent of each other.
6) Devise a method to generate shapes within the design domain that explicitly take into account the constraints posed by manufacturing
7) Devise various analyses that together calculate the objective function
8) Apply the optimization algorithm and optimize the geometry
9) Apply any remaining post-processing required to produce a manufacturable result

To evaluate the process, one could take the following steps:

a) Perform a sensitivity analysis of the optimization parameters
b) Perform a sensitivity analysis of the various components in the objective function

Note that pure optimization in structural engineering excludes the manufacturing constraints in step (6) and relies heavily on step (9) instead to incorporate them. In some existing cases the degree to which the solution conforms with the manufacturing constraints is used in the objective function, an arguably inefficient strategy.

9.5 Overview of chapter conclusions and recommendations

This paragraph contains summaries of the conclusions and recommendations of each chapter. Table 9.1 is a summary of the conclusions and recommendations of Chapter 2 on the BESO algorithm.

TABLE 9.1 Chapter 2

Conclusions	Recommendations
• Implementation of BESO in ANSYSscript was succesful • Adapting the RRV to the new RRVrel produced a faster and more robust BESO algorithm • Current BESO results are impractical and unrealistic due to fictitious material models and small applied loads • Current BESO results give qualitative information on optimal shapes and topologies	• Implement a non-linear analysis with a realistic material model to improve the practicality of BESO results

Table 9.2 is a summary of the conclusions and recommendations of Chapter 3 on fabric formwork technology.

TABLE 9.2 Chapter 3

Conclusions	Recommendations
The potential geometries are: • Complex e.g. double curved • Structurally wefficient • Esthetically unusual and pleasing • Economically efficient The mold is relatively: • Simple in nature • Inexpensive • Lightweight • Reusable • The mold consists of widely available materials • Improved surface quality of the casted concrete • Potential for reduction and simplification of reinforcement requirements • Geometric predictability is difficult due to the non-linear geometric behavior of fabrics Geometric accuracy and consistency are relatively difficult to maintain due to: • Relaxation • Creep • The complexity requires more structural analysis Material savings might have detrimental effect on: • Decrease in redundancies i.e. safety • Decrease in redundancies i.e. safety	• Comparative analysis of different fabrics • Research on creep and relaxation

Table 9.3 is a summary of the conclusions and recommendation of Chapter 4 on Differential Evolution

TABLE 9.3 Chapter 4

Conclusions	Recommendations
• Differential Evolution was succesfully implemented • Finding appropriate evolutionary parameters is difficult and no conclusive recommendations exist for their use	• Carry out a sensitivity analysis of the evolutionary parameters

Table 9.4 is a summary of the conclusions and recommendations of Chapter 5 on Dynamic Relaxation.

TABLE 9.4 Chapter 5

Conclusions	Recommendations

• Dynamic Relaxation was succesfully implemented and adapted to model all existing mold types for fabric formed beams	• Improve collision detection
	  • More accurate interpolation routines
	  • Investigate existing forms of collision detection
	• Add interdependancy of pinch points
	• Replace cable-net modelling with membrane modelling
	• Investigate possible numerical errors due to floating point calculus
	• General refactoring of the algorithm to improve speed
	• Add pattern generation and assess tear resistance of seams
	• Assess influence of folding, improve modelling of folding
	• Assess and possibly implement different types of generating certain variables (other than Beziér)
	• Introduce non-linear material behaviour (based on results of bi-axial testing)

Table 9.5 is a summary of the conclusions and recommendations of Chapter 6 on concrete modeling in ANSYS.

TABLE 9.5 Chapter 6

Conclusions	Recommendations
• Accurate modelling of reinforced concrete in ANSYS is possible • A flexible non-linear concrete model using smeared reinforcement was developed	• Flexible, reliable unstructured meshing method (not ANSYS) to replace current block mesh approximation • Further verification of the approximation method • Have ANSYS first apply self-weight prior to other loads • Multiple load cases • Modelling of e.g. fiber reinforcement, or prestressing • Investigate various reinforcement strategies (fiber reinforcement, prestressing, strands/tendons, bars/cages, high strength carbon fabrics)

Table 9.6 is a summary of the conclusions and recommendations of Chapter 7 on design and software architecture of the FABRICFORMER program.

TABLE 9.6 Chapter 7

Conclusions	Recommendations
• FABRICFORMER is an operational piece of software • FABRICFORMER succesfully combines Differential Evolution, Dynamic Relaxation and ANSYS finite element analysis	• Export FabricFormer as a stand-alone product • Create internal Java-coded finite element analysis, or include flexibility to use various finite element programs • Streamline the various image and text output to a single source e.g. a GUI. • Specify input of overall convergence criteria to automatically stop the program • Conform software structure to UML conventions • Add a graphical user interface (GUI) that includes input, progress reporting and output • Implement parallel computation

Table 9.7 is a summary of the conclusions and recommendations of Chapter 8 on the results of the BESO algorithm for non-linear analysis and of the FABRICFORMER program. Many recommendations of the previous chapters help to achieve the recommendation in this table.

TABLE 9.7 Chapter 8

Conclusions	Recommendations
• FABRICFORMER proves to succesfully optimize the geometries of fabric formed beams • All results deflect less at midspan that rectangular beams of equal volume based on linear analysis. • Pinch mold beams have relative higher gains in stiffness than keel mold beams compared to rectangular beams of equal volume based on linear analysis.	• Crossover factor Cr should be high for interdependant variables, low for independent variables • The dimension D should preferably be higher than 50 for pinch mold optimization • Change the program to avoid thin beams – which are currently optimal within the context of the performance index and linear analysis – by (1) fine-tuning the loads, (2) introducing some stress or strain limit to cut off linear analysis, or (3) implement non-linear analysis with real stress and strain limits

9.6 Summary of main conclusions and recommendations

The previous paragraphs form a plethora of conclusions and recommendations, some of which are not as significant as others. An attempt is made to derive the most important conclusions that have been drawn from this thesis project. The conclusions are:

- The gap between computational optimization and manufacturability has been bridged to a large extent for evolutionary algorithms and fabric formwork technology.
- The explicit combination of computational optimization and manufacturability has been proven to be feasible and viable.
- The reinforced concreted beam has been geometrically optimized with respect to stiffness and volume for a particular load case using non-linear analysis.
- The fabric formed structural beam has been geometrically optimized with respect to stiffness and volume for a particular load case using linear analysis.

The following recommendations will improve the `FABRICFORMER` program and its results:

- Improve the fabric modeling with respect to non-linear and bi-axial behaviour
- Implement non-linear analysis for reinforced concrete
- Explore or develop a reliable, flexible unstructured meshing methods to replace the current block mesh approximation
- Include remaining aspects to make the results truly manufacturable e.g. supports
- Improve quality of the program from software engineering and user-friendliness point of views
- Investigate and provide for other possible considerations in the optimization

The following suggestions are made to explored during further development of `FABRICFORMER` or as a part of new research projects.

- Combine BESO and general optimization as primary and secondary stages within a single optimization process.
- Combine parametric associative design and general optimization as reciprocal elements within a single optimization process.

References

Books and readers

Aarts, E. & Korst, J. 1989, *Simulated annealing and Boltzmann machines*, Anchor Press, Essex.

Adriaansen, W.L.M. et al. 1996, *Overspannend staal, Deel 1: Basisboek,* 3rd ed., Staalbouwkundig Genootschap, Rotterdam.

Alsmarker et al., 1995, *Timber engineering STEP 1,* Centrum Hout, Almere.

Blaauwendraad, J. 2004, *Theory of elasticity energy principles and variational methods,* TU Delft Press, Delft

Briedé, K.J. & Blok, R. 1995, *Tabellen voor bouw- en waterbouwkunde,* 7th ed., Spruyt, Van Mantgem en De Does, Leiden

Clarke, J.L. 1996, *Structural design of polymer compositers – EUROCOMP design code and handbook,* Spon, London.

Cox, H.L. 1965, *The design of structures of least weight*, Pergamon Press, Bath.

Degarmo, E.P. et al. 2003, *Materials and processes in manufacturing,* 9th ed., John WIley & Sons, New York.

Dorigo, M. & Stützle, T. 2002, *Ant colony optimization*, MIT Press, Cambridge.

Feoktistov, V. 2007, *Differential evolution: in search of solutions*, Springer, New York.

Grünewald, S. 2004. *Performance-based design of self-compacting fibre reinforced concrete,* Delft University Press, Delft.

Günther, H.-P. et al. 2005, *Use and application of high-performance steels for steel structures,* IABSE-AIPC-IVBH, Zürich.

Hemp, W.S. 1973, *Optimum structures,* Oxford University Press.

Hendriks, C.F. et al. 2000, *Durable and sustainable construction materials,* Aeneas, Best.

Honselaar, C.H. et al. 1999, *Vezelversterkte kunststoffen in constructieve toepassingen,* vm119, Vereniging FME-CWM, Zoetermeer

Hopkinson, N. et al. 2006, *Rapid manufacturing. An industrial revolution for the digital age.* John Wiley & Sons Ltd, West Sussex.

Jackson, N. & Dhir, R.K. 1996, *Civil engineering materials,* 5th ed., MacMillan Press Ltd, Hong Kong.

Lewis. W.J. 2003, *Tension structures: form and behaviour,* Telford, London.

McDonough, W. & Braungart, M. 2002, *Cradle to cradle.* Remaking the way we make things, North Point Press, New York.

Nijhof, A,H.J. 2004, *Vezelversterkte kunststoffen : mechanica en ontwerp,* DUP Blue Print, Delft.

Price, K.V. et al. 2005, *Differential evolution: a practical approach to global optimization,* Springer, London

Reinhardt, H.W. 2004, *Betonkunde,* Delftse Universitaire Pers, Delft.

Soons, F.A.M. et al., 2002, *Infomap "Nieuwe Stijl",* TU Delft Press, Delft.

Starr, T.F. 2000, *Pultrusion for Engineers,* Woodhead publishing, Cambridge.

Ros, D. et al. 2003, *Vezelversterkte kunstoffen in civiele draagconstructies : achtergrondrapport bij CUR-Aanbeveling 96,* CUR, Gouda.

Vambersky, J.N.J.A. et al. 2001, *Designing and understanding precast concrete structures in buildings,* TU Delft Press, Delft.

Walraven, J.C. & Galjaard, J.C. 1997, *Voorgespannen beton,* BetonPrisma, 's-Hertogenbosch.

Walraven, J.C. 2003, *Gewapend beton; college CT2051B/3051B,* TU Delft Press, Delft.

Wright, P.K. 2001, *21st century manufacturing,* Prentice-Hall, New Jersey.

Xie, Y.M. 1997, *Evolutionary structural optimization,* Springer, London.

Articles and scientific papers

Al Awwadi Ghaib, M. & Górski, J. 2001, 'Mechanical properties of concrete cast in fabric formworks', *Cement and Concrete Research,* vol. 31, pp. 1459-1465, Pergamon.

Allaire, G. et al., 1995. 'Shape optimization by the homogenization method'. *Numerische Mathemathik,* vol 75, pp. 27-68. Retrieved: December 2006, from SpringerLink

Anon. 2005, 'Extruder, slipformer, or both?', *Concrete Issues, issue 1.* Retrieved: December 2006, from http://www.concreteissues.com/index.php?mid=122.

Bahreininejad, A, 2003. 'A hybrid ant colony optimization approach for finite element mesh decomposition'. *Structural-Multidisciplinary Optimization,* vol. 28, pg. 307-316, 2004.

Barbosa, A.F. & Ribeiro, G.O. 1998, 'Analysis of reinforced concrete structures using ANSYS nonlinear concrete model', *Computational Mechanics*, Barcelona, Spain

Barnes, M.R. 1998, 'Form-finding and analysis of prestressed nets and membranes', *Computers & Structures,* vol. 30, no. 3, pp. 685-695, Civil-Comp Ltd and Pergamon Press, Great Britain.

Bendsøe, M. & N. Kikuchi, 1988. 'Generating optimzal topologies in structural design using a homogenization method'. *Computer Methods in Applied Mechanics and Engineering,* vol. 71, pp. 197-224, North-Holland: 1988.

Bledzki, A. & K. Goracy, 1992. 'The use of recycled fibre composites as reinforcement for thermosets', pp. 352-356.

Buitelaar, P., 2004, 'Heavy Reinforced Ultra High Performance Concrete', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pg. 25-35. Retrieved: from http://cgi.uni-kassel.de/~dbupress/download_frei.pdf.cgi?3-89958-086-9

Cerny, V. 1985, 'Thermodynamical approach to the traveling salesman problem: an ecient simulation algorithm', *J. Optim. Theory Appl.*, vol. 45, pp. 44-51.

Chang, D.-Y. & J.-S. Moh, 2006. 'A golbal optimization method based on simulated annealing and evolutionary strategy'. ICIC 2006, LNCS 4113, pp. 790-801, 2006. Springer-Verlag Berlin Heidelberg 2006.

Chang, D.-Y. & J.-S. Moh, 1999. 'Improved simulated annealing search for structural optimization'. *AIAA Journal,* vol. 38, no. 10, October 2000.

Cui, C. et al., 2003. 'Computational morphogenesis of 3D structures by extended ESO method'. IASS, vol. 44, no. 1, no. 141, pp. 51-61, 2003,.

Dorigo, M. & Stützle, T. 2004, 'The ant colony optimization metaheuristic: algorithms, applications, and advances', *International Series in Operations Research & Management Science,* vol. 57, pp. 250-285. Retrieved: December 2006, from SpringerLink

Dugat, J. et al. 1996, 'Mechanical properties of reactive powder concretes', *Materials and Structures,* vol 29. no. 4, pp. 233-240. Retrieved: from http://www.springerlink.com/content/28t30w237676525t

Fehling, E., 2004, 'Design relevant properties of hardened Ultra High Performance Concrete', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 327-338.

Fenyes, P., 1992. 'Structural optimization with manufacturing considerations', *Structural Opimization,* vol. 5, pg. 116-122. Springer-Verlag, 1992.

Hasançebi, O. & F. Erbatur, 2002. 'On efficient use of simulated annealing in complex structural optimization problems'. *Acta Mechanica,* vol. 157, pp. 27-50, 2002. Springer-Verlag 2002.

Haseganu, E.M. & Steigmann, D.J. 1994, 'Analysis of partly wrinkled membranes with dynamic relaxation', *Computational Mechanics,* vol. 14, pp. 596-614, Springer-Verlag.

Heinz, D. et al. 2004, 'Fire resistance of ultra high performance concrete (UHPC) – Testing of laboratory samples and columns under load', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 703-716.

Herold, G. & Scheydt, C. 2006, 'Development and durability of ultra high performance concretes', *BFT,* issue 10, pp. 4-14.

Holschemacher, K. et al. 2004, 'Ultra high strength concrete under concentrated loading', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 471-480.

Hörnlein, H. et al., 2001. 'Material optimization: bridging the gap between conceptual and preliminary design'. *Aerospace Science Technology,* pg. 541-554, 2001

Huang, X. et al., 2006. 'A new algorithm for bi-directional evolutionary structural optimization'. *JSME*, series A, February 2006.

Huang M. & J. Arora, 1997. 'Optimal design of steel structures using standard sections'. *Structural Optimization,* vol. 14, pg. 24-35. Springer-Verlag 1997.

Jungwirth, J. & Muttoni, A. 2004, 'Structural Behavior of Tension Member in UHPC'. *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 533-546.

Kahan, W. 2004, 'Miscalculating area and angles of a needle-like triangle'

Katsikadelis, J.T. & Nerantzaki, M.S. 2002, 'The ponding problem on elastic membranes: an analog equation solution', Computational Mechanics, vol. 28, pp. 122–128, Springer-Verlag

Kawamura H., et al., 2002. 'Truss topology optimization by a modified genetic algorithm'. *Structural Multidisciplinary Optimization,* vol. 23, pp. 467-473. Springer-Verlag 2002.

Kepler, J., 2002. 'Structural optimization as a design and styling tool – with emphasis on truss structures.' Retrieved: December 2006, from http://www.caam.rice.edu/~caam210/fibnet/kepler.pdf Books and readers

Khoshnevis, B., 2004. 'Automated construction by contour crafting – related robotics and information technologies', *Journal of Automation in Construction,* vol. 13, no. 1, pp. 5-19.

Kirkpatrick, S., 1983. 'Optimization by simulated annealing: quantitative studies'. *Journal of Statistical Physics,* vol. 34, no. 5/6, pp. 975-986, 1984. Plenum Publishing Corporation.

Lampinen, J. & I. Zelinka, 1999. 'Mixed variable non-linear optimization by differential evolution.'

Liu, J. et al., 2000. 'Metamorphic development: a new topology optimization method for continuum structures'. *Structural Multidisciplinary Optimization,* vol. 20, pp. 288-300. Retrieved: December 2006, from SpringerLink

Lundy, M. & Mees, A. 1986, 'Convergence of an annealing algorithm', *Math. Prog.,* vol. 34, pp. 111–124.

Ma, J. & Schneider, M. 2002, 'Properties of Ultra-High-Performance Concrete', *Leipzig Annual Civil Engineering Report, no.7,* pp. 25-32. Retrieved: December, 2006, from http://aspdin.wifa.uni-leipzig.de/institut/lacer/lacer07/l07_05.pdf

Makar, J. & Beaudoin, J. 2003, 'Carbon nanotubes and their application in the construction industry'. *1st International Symposium on Nanotechnology in Construction,* pp. 331-341.

Ni, C. et al. 1988, 'A concurrent engineering approach to integrate forming effects into structural design and analysis of stamped parts'.*Proc. 7th Int. Conf. on Vehicle Structural Mechanics.* SAE Publication P-210, Paper no. 880916. Warrendale: Society of Automotive Engineers

Ohmori, H. et al., 2004. 'Computational morphogenesis and its applications to structural design'. ISSS, pp. 13-20, 2005.

Orgass, M. & Klug, Y. 2004, 'Fibre reinforced ultra-high strength concretes', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 637-648.

Padmarajaiah, S.K. & Ramaswamy, A. 2002, 'A finite element assessment of flexural strength of prestressed concrete beams with fiber reinforcement', *Cement & Concrete Composites,* vol. 24, pp 229-441

Pantalides, C. & S.-R. Tzan, 1996. 'Simulated annealing for the design of structures with time-varying constraints'. Structural Optimization, vol. 13, pp. 36-44. Springer-Verlag 1997.

Querin, O.M. et al. 1997,'Improved computational efficiency using bi-directional evolutionary structural optimisation (BESO).' Abstract for 4-th World Cong. on Computational Mechanics.

Racky, P. 2004, 'Cost-effectiveness and sustainability of UHPC', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 797-806.

Richard, P. & Cheyrezy, M.1995, 'Composition of reactive powder concretes'. *Cement and Concrete Research,* vol. 25, no. 7, pp. 1501-1511.

Roux, N. et al. 1996, 'Experimental study of durability of reactive powder concretes'. *Journal of Materials in Civil Engineering.*

Scheurer, F., 2003. 'The Groningen twister. An experiment in applied generative design. ' Retrieved: January 2007, from swiki.arch.ethz.ch:8000/CAAD-Extern/uploads/1013/GA2003-Scheurer.pdf

Schmidt, M. et al., 2004, 'Ultra High Performance Concrete (UHPC), *Proceedings of the International Symposium on Ultra High Performance Concrete.* Retrieved: from http://cgi.uni-kassel.de/~dbupress/download_frei.pdf.cgi?3-89958-086-9

Serra, M. & P. Venini, 2005. 'On some applications of ant colony optimization metaheuristic to plane truss optimization'. *Structural Multidisciplinary Optimization,* vol. 32, pg. 499-506, 2006.

Shim, P. & S. Manoochehri, 1999. 'A hybrid deterministic/stochastic optimization shape configuration design of structures'. *Structural Optimization,* vol. 17, pp. 113-129, Springer-verlag 1999.

Shim, P. & S. Manoochehri, 1997. 'Generating optimal configurations in structural design using simulated annealing'. *International Journal for Numerical Methods in Engineering,* vol. 40, pp. 1053-1069, 1997.

Socha K., 2004. 'ACO for continuous and mixed-variable optimization'.

Staquet, S. & Espion, B., 2004, 'Early-age autogenous shrinkage of UHPC incorporating very fine fly ash or metakaolin in replacement of silica fume', *Proceedings of the International Symposium on Ultra High Performance Concrete,* pp. 587-600.

Storn R. & K. Price, 1995. 'Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces'. *Technical Report TR-95-012,* ICSI, March 1995.

Teo, J., 2005. 'Differential evolution with self-adaptive populations'. KES 2005, LNAI 3681, pp. 1284-1290, 2005. Retrieved: December 2006, from SpringerLink

Thomas, J. & Ramaswamy, A. 2006, 'Finite element analysis of shear critical prestressed SFRC beams', *Computers and Concrete,* vol. 3, no. 1, pp. 65-77.

Tuan, C.Y. 1998, 'Ponding on circular membranes', *International Journal of Solid Structures,* vol. 35, no. 34, pp. 269-283, Pergamon, Elsevier, Great Britain

Topping, B.H.V. & A.I. Khan, 1994. 'Parallel Computation Schemes for Dynamic Relaxation', *Engineering Computations,* vol. 11, no.6, pp. 513-548.

Wapperom, H. 1999, 'Moderne beeldhouwkunst als werkomgeving', *Cement,* no. 6, pp. 51-54.

Wapperom, H. 2005, 'Volautomatisch 3D-frezen in polystyreen', *Cement,* no. 8, pp. 71-73.

Weiler, B. & Grosse, C. 1996, 'Pullout behaviour of fibers in steel fiber reinforced concrete', pp. 116-127. Retrieved: December 2006, from http://www.mpa.uni-stuttgart.de/publikationen/otto_graf_journal/ogj_1996/beitrag_weiler.pdf.

West, M. 2003, 'A brief description of fabric-formed concrete'. Retrieved: December 2006, from http://www.umanitoba.ca/faculties/architecture/cast/pdf_downloads/1_pr.pdf.

Wüchner, R. & Bletzinger, K.-U. 2005, 'Stress-adapted numerical form finding of pre-stressed surfaces by the updated reference strategy', *International Journal for Numerical Methods in Engineering,* vol. 64, pp. 143-166, Wiley InterScience.

Xie M. & G. Steven, 1992. 'A simple evolutionary procedure for structural optimization'. *Computers & Structures,* vol. 49, no. 5, pp. 885-896, 1995. Elsevier Science Ltd., Great Britain.

Yakovlev, G. et al., 2005. 'Cement based foam concrete reinforced by carbon nanotubes'. ISSN 1392-1320, *Materials Science,* vol. 12, no. 2.

Yildiz, A. et al., 2003. 'Integrated optimzal topology design and shape optimization using neural networks'. *Structural Multidisciplinary Optimization,* vol. 26, pg. 251-260, 2003

Young,V. et al., 1999. '3D and multiple load case bi-directional evolution (BESO)'. Structural Optimization, vol. 18, pp. 183-192. Retrieved: December 2006, from SpringerLink

Zhenyu, G. et al., 2006. 'Self-adaptive chaos differential evolution'. ICNC 2006, Part I, LNCS 4221, pp. 972-975, 2006. Retrieved: December 2006, from SpringerLink

Theses

van Gemert, R.J. 1996, *Additive evolutionary structural optimisation.* University of Sydney, Sydney.

Pearse-Danker, H. 2006, *Implementing manufacturing considerations into evolutionary structural optimization of cast steel nodes,* TU Delft, Delft.

van Roosbroeck, M. 2006. *The construction of prefab concrete shells,* TU Delft, Delft.

Schmitz, R.P. 2004, Fabric-formed concrete panel design, Milwaukee School of Engineering, Milwaukee, Wisconsin

van de Straat, R. 2007, *Optimisation of structural transfer zones in muti-use buildings,* TU Delft, Delft.

Building codes

NEN6720, 1995, *Voorschriften beton, TGB 1990, Constructieve eisen en rekenmethoden (VBC 1995),* 2nd ed., Nederlands Normalisatie-instituut, Delft.

CUR Aanb. 97, *Aanbeveling 97, Hogesterktebeton,* Civieltechnisch Centrum Uitvoering Research en Regelgeving, Gouda.

CUR 2003-6. See Ros, D. et al (2003)

Hand-outs

Walraven. J.C. *From conventional to high performance steel fibre concrete.* TU Delft, Delft.

Walraven, J.C. *High strength concrete.* TU Delft, Delft.

Walraven, J.C. *Self compacting concrete: development and applications.* TU Delft, Delft.

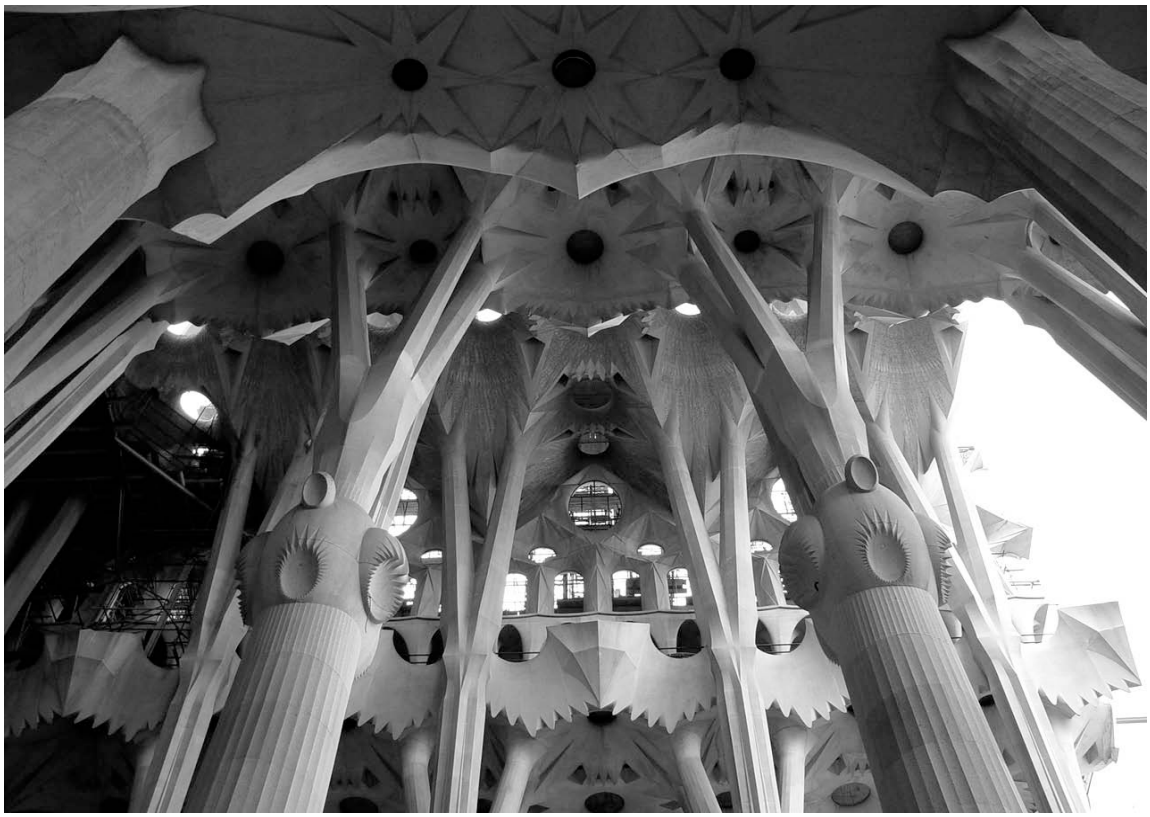


Figure A.1
The Sagrada Familia in Barcelona, Spain was the life's work of Antoni Gaudí. Even after 40 of his years, when he died, it was not completed and construction continues to this day.
Gaudí's famous hanging chain models exemplify how he used nature to guide his structural design and likewise computational optimization is now used to further the design process of the remaining, unfinished parts.

APPENDIX A | Computational optimization applied in structural design

The creation continues incessantly through the media of man. But man does not create...he discovers. Those who look for the laws of Nature as a support for their new works collaborate with the creator. Copiers do not collaborate. Because of this, originality consists in returning to the origin.

— Antoni Gaudí (1852-1926)

This appendix was the third chapter of the preliminary study on existing literarture for this thesis. Some of the original content, specifically that which dealt with Evolutionary Structural Optimization (ESO) and Differential Evolution (DE) has been removed. Instead, these now form the introductory paragraphs for their respective chapters, Chapter 2 and 4 of this report.

Finding a better way to do things. Optimization is as simple as that. And what better way to achieve this than by having the computer do all the hard work.
At the same time optimization is very complex, warranting a lot of scientific research in this field, called *computational optimization*. Because how do you define 'better'? And how do you find better solutions for real problems, problems that are complex and cannot be described through simple analytical math? These difficult problems are referred to as *combinatorial optimization problems* as they are a combination of several parameters that form the objective function. The *objective function* is that which needs to be optimized, and may quantifiably show how different solutions compare. As an objective function becomes dependant on more variables, it becomes increasingly complex, resulting in possible *local optima* besides the best solution, the *global optimum*. Finding these global optima is finding a better way to do things.

The basic premise of computational optimization is the continuous comparison of possibilities through numerical math until a satisfactory solution is obtained.
For instance; a given problem is too complex to evaluate at first glance. An optimization method comes into play and generates solutions A and B. Both solutions are compared, and when A turns out to be superior in some way, solution A is given preference and/or solution B is discarded. This is an ongoing process as the optimization method generates many solutions throughout the realm of possibilities, the *search space*. At some point the optimization method is terminated, either because it ran too long, or the solution no longer improves or because it reached a satisfactory solution, one that is good enough.

How well a certain optimization method performs is usually judged by the number of comparisons, or iterations, it needs to reach its goal and whether or not the method is susceptible to *stagnating*, which refers to ending up in local optima while a global optimum exists.

It is interesting to note that this approach to optimization bears resemblance to systems in nature. Strategies such as '*trial-and-error*' and '*survival of the fittest*' come to mind. It is this observation that has spawned many different types of optimization methods that mimic nature by incorporating analogies to natural phenomena.
In this chapter the history of optimization applied to structural design is discussed (A.1). An introduction is given to several optimization algorithms; some specifically geared towards structural design (A.2) and some general algorithms which have found applications in other fields as well (A.3). To complete the chapter some examples are given of structural optimization with manufacturing considerations (A.4), computational optimization in actual building projects (A.5) and optimization software for structural design (A.6). Some conclusions are drawn with respect to the rest of this thesis based on the content of this chapter (A.7).

A.1 Historic overview of optimization in structural design

The first steps in optimizing structures were taken by Michell (1904), Cox (1965) and Hemp (1973). The layout of planar trusses was optimized analytically and geometrically to produce least-weight structures. The so-called Michell structures have been proven to be theoretically sound and have become a benchmark test in structural optimization. If your method produces a Michell truss under the same conditions, you're on the right way. In Michell structures the truss members follow the force trajectories, creating the lightest structures possible to carry the loads. It is interesting to note that outside the area of the supports all cells have four edges, but as a whole form a stable system.

Although these structures can be translated to more practical and simplified solutions, they have limited use as they apply to very specific, very extreme loading and support conditions. Some suggestions were made for practical applications, but literature on Michell structures is usually academic in nature.

On a side note, Cox already acknowledged the aim of this thesis in his conclusions:

> *Problems of fabrication have to be taken into account, and those and other considerations may outweigh the question of weight-efficiency.*
>
> *— H.L. Cox*

At some later point, structural optimization was categorized in three distinct types:

- Size optimization
  Size optimization refers to modifying the cross-section of individual members or the thickness of finite elements.

- Topology optimization
  This optimization refers to modifying the design space where redundant areas/elements are removed to decrease the weight of the structure and/or adding areas/elements where unacceptable peak stresses occur.

- Shape optimization
  Shape optimization is the most complex and involves modifying the surface of the structure, in many cases to reduce occurring stress peaks.
  However, the definition of shape optimization is not very strict because 'shape' is a very general term and so, in some scientific papers refers to any combination of these three categories.

There are many topology optimization methods, and papers have been written on integrating these methods with shape optimization. In this way, shape optimization delivers precise, practical results while the method also benefits from the speed at which topology optimization removes and/or adds material. Currently, most structural optimization methods can be characterized as either such a combination or topology optimization with some form of post-processing. A combination of size and topology optimization is common for optimizing trusses.

The success of structural computational optimization is largely due to the invention of finite element analysis, which offered a practical manner in which structures could be analyzed by a computer. Combining structural optimization and finite element methods began in the late seventies. The driving forces behind developments in this field were the automotive and aeronautics industries, where material savings play an important part in design and outweigh other considerations.

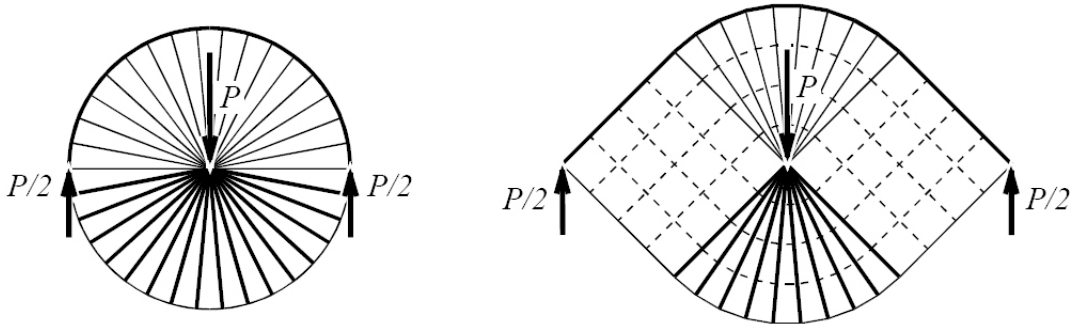


Figure A.2
Approximate renditions of Michell structures. Each structure is uniquely suited to carry the imposed loads to its supports. The structures consist of curves crossing each other at right angles. The curves either follow the tensile or compressive force trajectories.
Source: Kepler (2002)

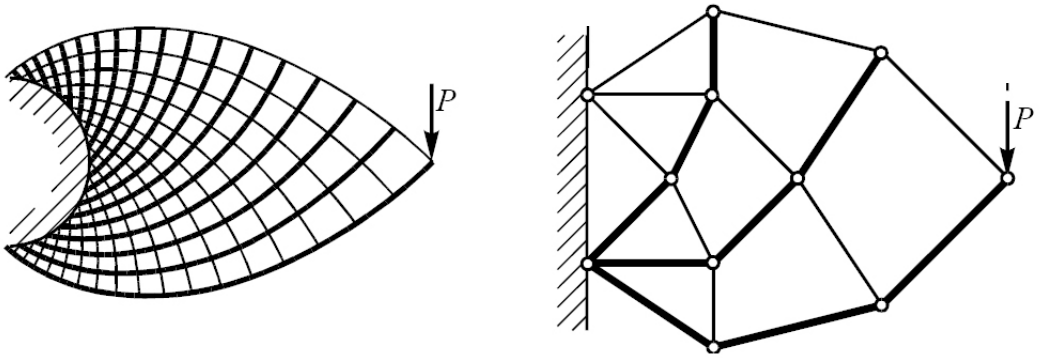


Figure A.3
Another rendition of a Michell structure and a simplified truss based on it show how, to a certain degree, feasible results can be extrapolated.
Note that it is still ill-equipped to carry any load other than the point load P.
Source: Kepler (2002)

A.2 Specific algorithms for structural design

There are several ways in which optimization algorithms can and are being categorized, but as of yet there is no seminal work on the classification of these algorithms. For the purposes of this thesis the algorithms discussed are categorized as either specific, or tailored algorithms, or as general algorithms.

A.2.1 Homogenization method

It was the *homogenization method*, introduced by Bendsøe & Kikuchi (1988), that is now widely recognized as the mayor catalyst for optimization in structural design.
They proposed a design method that no longer produced final designs that were topologically equivalent to the initial choice of design and avoided any need for remeshing of the finite element approximation.
The method uses a fixed geometry with one mesh. The material used is an artificial, somewhat fictitious composite with varying density, due to varying content and orientation of microscopic voids. The final result is a greyscale image showing the material distribution which needs to be interpreted for the final shape. Such post-processing can occur by introducing a cut-off value for certain densities, called a *lumping process*; assigning a density of 0 or 1 below or above a certain value. This produces so-called black-white topology designs. Similarly, *penalizing* forces the density to take on a value of 0 or 1 upon convergence to the optimal density, adding a number of iterations to the optimization process. (Allaire et al., 1997)
Another variation is the *boundary variations method*, where moving mesh schemes define the shape of the structure.

The homogenization method is suitable for arbitrary topologies and multiple load cases, but the size of the ground mesh is unknown and highly influences the results. Furthermore, post-processing is needed to practically interpret the output, causing more computational requirements.

A.3 General algorithms and structural applications

There is a seemingly endless collection of optimization algorithms each designated with a colorful name. Over the years, these methods have been developed and improved upon, often by hybridization i.e. incorporating properties of other algorithms to create a superior combination., usually with an extended acronym to name it.

A.3.1 Genetic Algorithms (GA)

At the moment, Genetic Algorithms (GA) are the most popular general algorithm, finding uses in virtually every business sector. It is an algorithm that draws inspiration from evolution and is widely discussed in an ever increasing body of work.
Basically it works by simulating procreation and genetics. With every iteration, new solutions (children) are generated by combining (crossover) properties (genes) of the existing solutions (parents). Some properties are randomly changed (mutation) to increase the chance of reaching a global optimum. The parents and children are compared and a selection is made for the new population.

A.3.2 Simulated Annealing (SA)

This optimization method was introduced in the early eighties (Kirkpatrick, 1983 and Cerny, 1985). Subsequently, Lundy & Mees (1986) demonstrated the asymptotic convergence behaviour of Simulated Annealing (SA) theoretically.
SA is also known as Monte Carlo annealing, probabilistic hill climbing, statistical cooling and stochastic relaxation. Simulated annealing however, has become the preferred term. This name originates from the analogy with the physical annealing process of solids this method uses.

In condensed matter physics and metallurgy, annealing is known as a thermal process in which the atomic structure of a solid is forced in a highly ordered state. The solid is heated to a temperature that allows many atomic rearrangements and then cooled slowly, minimizing energy. During this process occasionally jumps to higher states of energy occur, before a global optimum is reached.

Through analogy a certain probability is introduced based on the Boltzmann distribution, where it is possible to 'climb out' of local optima by accepting worse solutions. The chance of this happening slowly decreases throughout the process.
So, normally one would only accept better solutions, while the worse ones are discarded. SA allows for some chance of sometimes accepting these worse solutions in favor of finding the overall best solution at some point in time.
The following formula shows the Boltzmann distribution is active if the energy E in the newer state j is higher, thus worse, than the former state i.

$$\mathrm{P}_T\left\{\text{accept } j\right\} = \begin{cases} 1 & \text{if } E(j) \le E(i) \\ \exp\left(\left(E_i - E_j\right)/k_B T\right) & \text{if } E(j) > E(i) \end{cases}$$

where k_B is the Boltzmann parameter.

The formulation of the Boltzmann parameter highly determines the performance of the algorithm and Hasançebi and Erbatur (2002) propose some refinements of this parameter for structural design purposes.

The following parallels are drawn within the analogy (Aarts & Korst, 1989):

- Possible solutions of the problem are equivalent to the states of a physical system.
- The cost of a solution or value of the objective function is equivalent to the energy level of a state.
- The temperature T can be chosen as the control parameter of the optimization process.
- The lowest energy state E corresponds to the optimum solution.

A set of specified parameters govern the convergence of the algorithm and are referred to as the cooling schedule. Such a cooling schedule determines how T decreases and when the optimization algorithm is terminated.

By now, the technique has been successfully applied in many fields of optimization, including optimum structural design. Most papers on simulated annealing in structural applications are from the early nineties and typically involve toplogy or size optimization in two- and three-dimensional trusses or frames. The most prolific authors in this period are G.S. Chen, R.J. Balling and R.K. Kincaid.

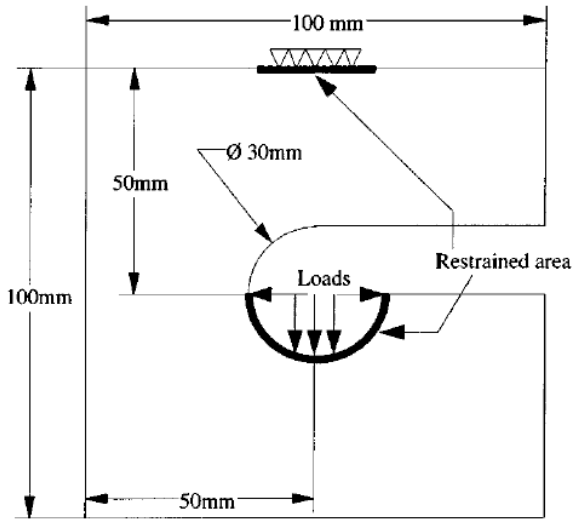
It is very noticeable that since 1995 SA is hardly featured in research on structural optimization. Besides the paper by Hasançebi and Erbatur (2002) mentioned earlier there are only two sets of papers, discussed below. This reduced application of SA could be attributed to the increased popularity of other algorithms such as ESO and GA.

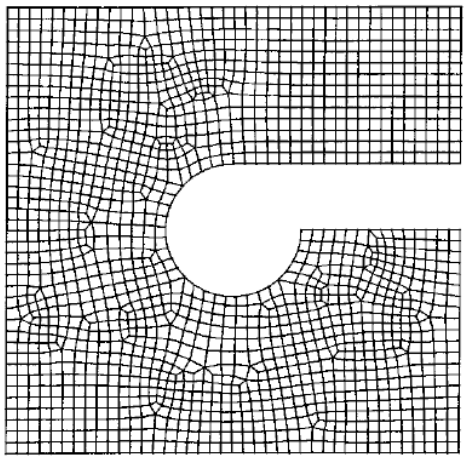
Tzan and Pantalides (1996) presented an algorithm using a sensitivity analysis to identify which design variables need to be modified and an automatic search space reduction to decrease the number of iterations needed. They applied this to size optimization of a dynamically loaded multi-storey frame.

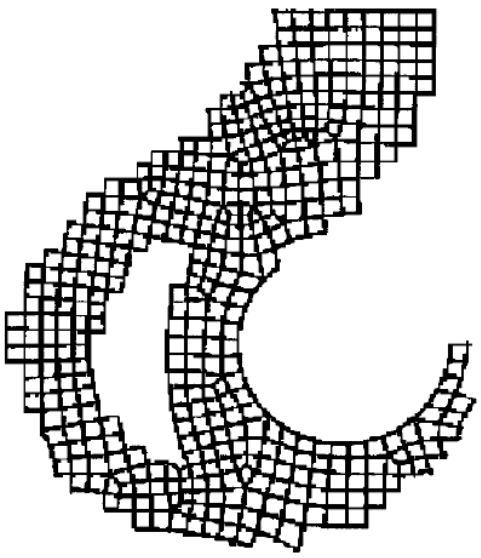
Shim and Manoochehri (1998 & 1999) applied SA to the shape optimization of two non-linear two-dimensional problems by using linear approximation. They compared SA to a deterministic enumeration approach in solving the same two problems. After observing faster convergence to a global optimum of the former and initially faster convergenco to local optima by the latter, they proposed a hybrid method of both techniques. This combination was able to reach a global optimum even faster.

Although SA seems less popular in structural optimization, general work on the algorithm continues. Recently, Chiang and Moh (2000 & 2006) have proposed combining

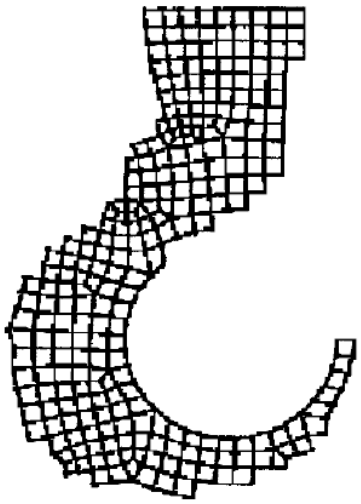
SA with simulated evolution (GA). This entails searching from a population as in the method of simulated evolution, instead of from a single point, a single candidate state. This development seems to be a logical reaction to the decreased popularity of SA, simply by incorporating evolutionary strategies as found in the succesful GA.
This new algorithm is referred to as the Region-Reduction Simulated Annealing (RRSA) method because it locates the optimum by successively eliminating the regions with low probability of containing optimum. In addition to faster convergence than conventional SA, other advantages of RRSA include the capability of searching multiple optima, and the ease of determining the initial temperature, equilibrium criterion and the stopping rule.







(A) $\sigma_{allow} = 1000\ N/mm^2$



(B) $\sigma_{allow} = 2000\ N/mm^2$

Figure A.4
Shape optimization of a hook with the initial design space, the load and support conditions and two optimal solutions for different allowable stresses.
Source: Shim & Manoochehri (1998)

A.3.3 Ant Colony Optimization (ACO)

In 1991 M. Dorigo developed Ant System (AS) to solve combinatorial optimization problems. The first results were promising enough to warrant development of this optimization algorithm and subsequent research led to the improved Ant Colony Optimization (ACO).

As its name suggest ACO draws inspiration from ants, specifically their optimization of traffic routes between the colony and their food sources. Ants individually find their way to a food source while depositing pheromone trails for their fellow ants to follow. In an equal amount of time, more ants will travel shorter routes, depositing more pheromones, leading to a preference for these optimal routes.

The analogy works as follows;
Individual search agents move through adjacent solutions within the search space based on a local, probabilistic decision policy. The agent, or ant, evaluates each solution, or step, using the objective function and deposits a certain amount of pheromones accordingly. Of course, if more than one ant passes a certain solution, more pheromones are deposited, making this 'route' even more attractive.This information is then used by all search agents to augment their local decision policies in each new iteration. Pheromone evaporation avoids the algorithm from converging too fast and stagnating at local optima.
(Dorigo & Stützle, 2002)

ACO is very popular, because it is able to find global optima in a wide range of problems and is particularly suited to handle dynamic problems. This popularity has led to many additions to ACO improving its convergence, mostly involving altering the decision policy of the search agents and the way the ant population works. Because of this the newer versions of ACO often deviate from the original analogy but retain the basic principle of individual agents incrementally building solutions.

Of particular interest is the extension of ACO to continuous and mixed-variable optimization problems, since ACO was initially applied to discrete domains due to the fact that the ants move in incremental, discrete steps. (Socha, 2004)

Strangely enough, ACO has hardly been applied to structural design problems. Two examples were found:

Bahreininejad (2004) used a hybrid ACO algorithm, combining ACO with some local search algorithm, to generate finite element meshes.

Serra and Venini (2006) used ACO for size optimization of two examples of plane trusses with buckling constraints. They concluded that ACO was succesful in finding global optima or near optimal solutions. Interestingly, while referencing the original papers by Dorigo from 1992, they characterize ACO as a novel approach, perhaps illustrating its limited use in structural optimization.

For a comprehensive overview of ACO, the reader is referred to the book by Dorigo & Stützle (2004).

A.4 Examples of optimization with manufacturing considerations

The examples given until now, show that structural optimization largely functions as an academic exercise. It is also, to some extent, used as an early help in structural design. How does the structurally optimal design look like? How do we approximate this shape? Results of structural optimization aid the designer when the first rough solutions are drafted.

However, more integral approaches exist and in this paragraph examples are shown of optimization with manufacturing, either considered during or after the optimization procedure. In the latter case, some form of post-processing alters the initial result of optimization to conform to a more practical, final design solution.

A.4.1 Shape optimization of sheet metal products

As mentioned, the automotive industry is one of the main driving forces in the development of structural optimization. The forming of sheet metal parts specifically offers interesting case studies for structural optimization.

Ni et al. (1988) and Fenyes (1992) both worked on optimization with structural and forming constraints simultaneously taken into account.
For instance, the latter introduced manufacturability

requirements by adding and controlling a sidewall thinning parameter to the problem of thin-wall beam type members. The thinning parameter takes into account thinning and stretching of the steel due to the forming process. Two forming constraints were incorporated as well to account for strain during and elastic springback after stamping of the sheet metal. The members were optimized for minimum weight. The optimization was reduced to simple formulae, which were not valid for other cross-sectional geometries or forming conditions and not accurate enough for three-dimensional modeling. Fenyes concluded that more sophisticated computational tools than those he used would be required to specify the forming geometry and die (mold) conditions of complicated three-dimensional problems such as automobile parts.
Since then more optimization of such parts — made through various sheet metal processes each with different forming constraints — has been done resulting in multiple papers and books on the subject.

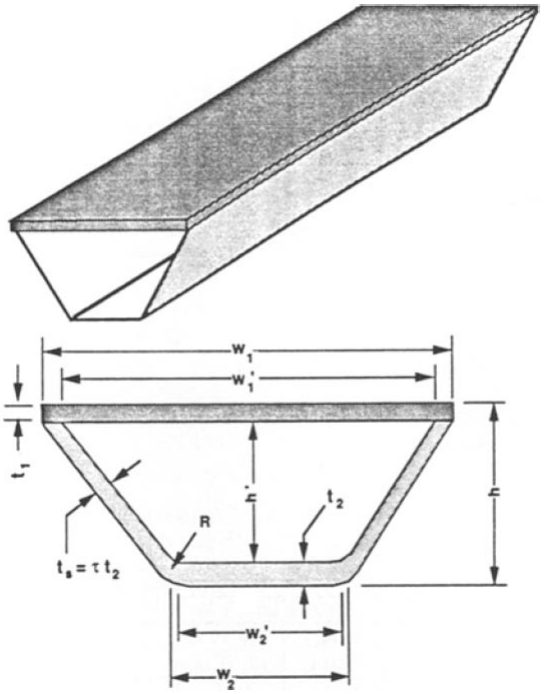
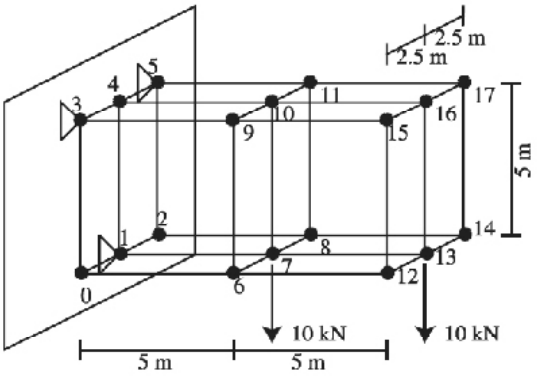


Figure A.5
The dimensions of this beam were optimized based on manufacturing considerations and how their size effects the total strain.
Source: Fenyes (1992)

A.4.2 Size optimization of feasible trusses

One of the first ways in which manufacturing could be taken into account in civil engineering was planar truss optimization. In size optimization of trusses the sectional area of each truss member initially was a continuous variable. By making the possible values discrete and having them correspond to commercially available steel sections, the optimization algorithm would provide optimized, yet feasible trussed structures. In these cases, weight-efficiency is obviously not as good as in purely theoretical results as a compromise is made with respect to manufacturing considerations. There is a considerable amount of research done on this topic and the reader is referred to Huang & Arora (1997). They used three algorithms (SA, GA and a third, called Branch and Bound) without drawing any general conclusions with regard to how they compare, but offer many references to similar work.
Kawamura et al. (2002) use GA to optimize both planar and three-dimensional trusses. This was the most advanced truss optimization found but did not yet incorporate manufacturing by discretizing the section sizes.



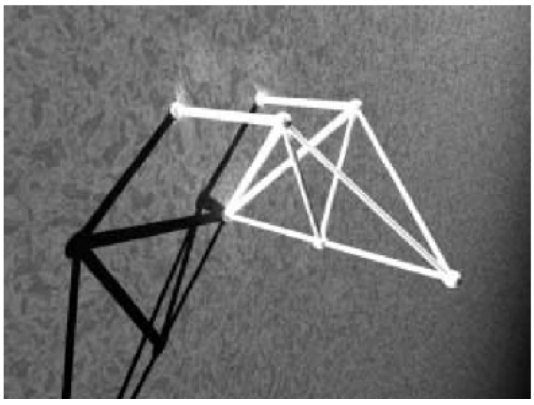


Figure A.6
Size, shape and topology optimization of a three-dimensional truss, by altering member size, location of the nodes and presence of the members respectively. This optimization was done using GA.
Source: Kawamura et al. (2002)

A.4.3 Free material optimization of fibre reinforced polymer structures

The homogenization method (Chapter A.2.1) offers a lot of information on the continuously varying density or microstructure of the material in a structure. A lumping process, or penalization, simplifies this information to a discrete zero-one situation, where material is interpreted as either being present or not. This post-processing penalization was the basis of some criticism by Hörnlein et al. (2001) who instead proposed Free Material Optimization (FMO). They felt that the results of the penalization were unpredictable and therefore potentially sub-optimal, as they prove to be highly dependant on parameters such as the penalty parameters and the mesh size. FMO differs from topology optimization because it not only considers the distribution of the material within a design space, but also the material properties at each point. This means that the material being optimized is no longer isotropic, but in fact will turn out to be orthotropic or anisotropic depending on the load cases. The material is optimized by finding a material elasticity matrix E by minimizing the potential energy so that the optimized body is as stiff as possible. At a certain point the problem is discretized using the finite element method, and a global minimum can be found. These results are also post-processed, but specifically for application with fiber-reinforced polymers made using tape-laying method (B.4.9). This is a fairly free-form manufacturing technique, and by distributing the fibers according to the post-processed results, the material optimization comes into play. The post-processing provides information on the orientation and density of the fibers as well as the necessary ply thicknesses. Software based on FMO has mainly been used in the aerospace industry.
It is noted that in this case manufacturing is not considered simultaneously with structural optimization, but only afterwards. Also, the criticism of the conventional homogenization method seems very harsh, since contrary to homogenization, FMO admittedly takes a step from conventional to advanced materials which heavily limits and also alters the scope of its applications. This makes FMO simply different, certainly not superior, which seems to be implied by its creators.

A.4.4 Combining topology and shape optimization for realistic structures

Over recent years, there have been several approaches to combining the strengths of topology and shape optimization, by integrating them. Topology optimization generally leads to skeletal and non-smooth structural geometries, which makes for poses many problems for integration with subsequent shape optimization. Most solutions use either some smoothing algorithm, or so-called feature libraries to identify arbitrary and complex shapes resulting from topology optimization and transform these to more logical geometries which could be manufactured. Problems arise when encountering complex shapes and altering them using the ultimately finite number of entries in a feature library.
Artificial intelligence in the form of neural networks has been implemented by Yildiz et al. (2003) to combine both types of optimization whilst overcoming the problems of feature libraries. Neural networks can cope with new geometries and are able to produce smooth and simple shapes in a more efficient way.
The drawback of neural networks is the fact that they need time to learn how to deal with problems.

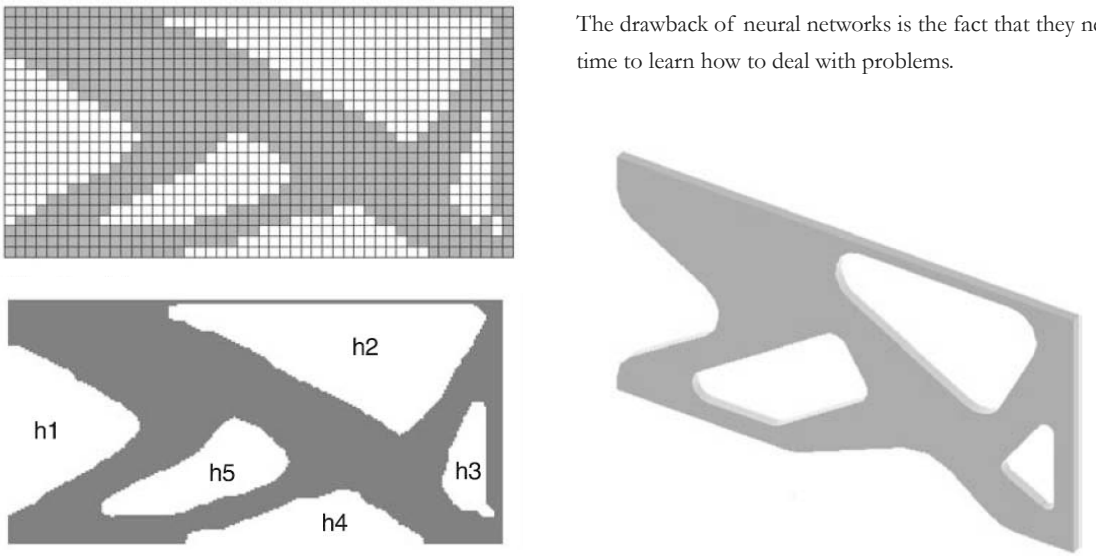


Figure A.7
Shape optimization of an optimized topology using neural networks instead of feature-libraries to intelligently recognize feasible shapes.
Source: Yildiz et al. (2003)

A.5 Examples of shape optimization in building projects.

Even more exceptional than manufacturing considerations in structural optimization, are actual building projects which involve optimization being part of the original design process. Some examples that used ESO, have been moved from this literature study to Chapter 2.

A.5.1 Groningen Twister in Groningen, Netherlands

The project, located in the downtown area of Groningen, connects the city center with the main train station. It is an underground parking area for some 300 bicycles, with a concrete slab roof that functions as a pedestrian area.
The main challenge in the proposed design was supporting this concrete slab by randomly sized and oriented columns. To this end a CAD-tool was developed in Java, by means of a collaboration between Kees Christiaanse Architects & Planners (KCAP), Rotterdam, an engineering team from Ove Arup & Partners, Amsterdam and the chair for Computer Aided Architectural Design (CAAD) at the ETH Zurich. The project was initiated in February 2003 and is currently under construction.
The program uses some form of particle swarm intelligence algorithm to optimize the design problem. It is discussed by Scheurer (2003) in somewhat more detail and a very similar project using GA formed the basis of a Master's thesis at this university by Van de Straat (2007).

Figure A.8
A model of the Groningen Twister. In the foreground the randomly oriented columns are visible which carry the pedestrian area above.
Source: Scheurer (2003)

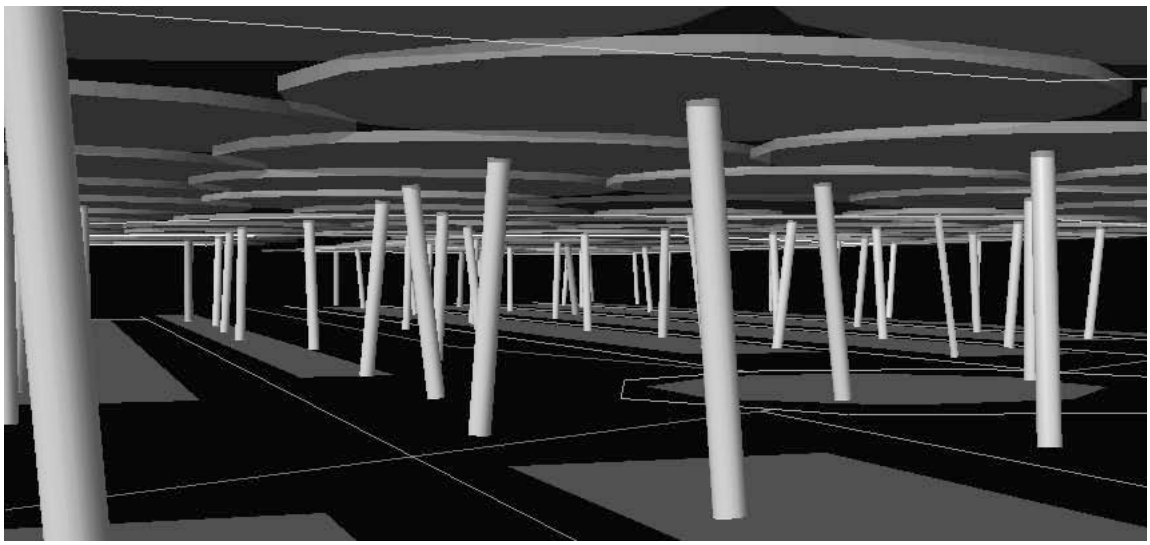


Figure A.9
A program was specifically written in Java for the purposes of realizing the Groningen Twister. The program outputs a 3D-model of a possible solution which has been optimized by some particle swarm optimization algorithm.
Source: Scheurer (2003)

A.6 Structural optimization software

The next logical step for computational optimization is to translate it to user-friendly software and making it readily available as a design tool.

These days there is a vast array of finite element software and such programs have become an important tool for any engineering company. It has been shown that optimization algorithms in structural design rely heavily on finite element meshes as a basis for calculations. In fact, the finite element method (FEM) is probably an important reason behind the progress in shape optimization. Because the FEM and structural optimization are so intertwined, it is no wonder that many of these software packages have taken the small step of adding optimization algorithms to their program.

The table shown lists software programs that offer some type of structural optimization. This overview is not comprehensive by any means. The upper part of the table shows in-house software, while the bottom part lists independently developed optimization programs.

There seem to be many software packages available for almost every finite element program. Because these are commercial packages, it was not possible to test and compare them. Information on how the optimization works is often sparse, but in some cases the description matches that of ESO, though this is uncredited.

All packages stress their usefulness in weight reduction in the early stages of design. Some offer multi-objective optimization or can also optimize for compliance (strain energy), frequencies or any constraint response.

As promotional images of these programs generally show optimization of machine or engine parts, their market seems to consist mainly of mechanical engineering companies.

Table A.1 Optimization software

Optimization program	Developer	Modeller/FEM solver	Algorithm
DesignXplorer	ANSYS	ANSYS	*"traditional and non-traditional algorithms"*
PEO	Dassault	CATIA	Simulated Annealing (SA), Conjugate Gradient
COSMOSWorks	Dassault	SolidWorks	ESO-like description with remeshing
Optistruct	Altair	HyperWorks	*"gradient based optimization"*, XESO-like description
Hypershape	Altair	CATIA	*"gradient based optimization"*, XESO-like description
TopoSLang	Dynardo	ANSYS	combination of *"topology and shape optimization"*
eoCAD	Even	CATIA	ETH Evolutionary Algorithm
TOSCA	FE-Design	ANSYS, ABAQUS, etc.	ESO-like description
PLM Optimization	Noesis	CATIA, etc.	Sequential Quadratic Programming (SQP) Generalized Reduced Gradient (GRG)
HEEDS	Red Cedar		some *"multi-agent approach"* algorithm

A.7 Conclusions

Finally, some conclusions and recommendations are made on computational optimization and for this thesis.

On computational optimization:

- The field of computational optimization is very active and competitive.
- Many optimization algorithms exist, but comparisons or benchmark tests are sparse and often biased.
- A comprehensive, categorized overview does not exist and is difficult to establish due to constant developments and hybridization.
- Computational optimization in the building industry is rare, but the tailored ESO algorithms are gaining popularity as a practical design tool.
- Many finite element programs with some optimization possibility exist, but it is unclear to what extent their optimization capabilities are used outside mechanical engineering. Also, it is difficult to assess and compare these programs due to limited availability of information.

On this thesis:

- ESO is a proven algorithm in structural design, but it remains unclear how it will cope with complex manufacturing constraints. ESO still seems a good algorithm to start with, due to its simple elegance and the amount of information available on the topic.
- DE seems very promising to use if ESO proves incapable of handling a mixed-variable optimization problem, which is likely to be the case in this thesis.

Figure B.1

The Diana, Princess of Wales Memorial Fountain at Hyde Park, London, is an elleptic watercourse with a circumference of 260 metres. It features uniquely textured surface effects and forms machined out of 545 granite stone blocks, totalling to 600 tonnes. The manufacturing process took only 28 weeks to complete by using 3D models and computer numerical controlled (CNC) cutting machines instead of traditional methods.

Source: http://www.texxus.com

APPENDIX B | Construction materials and modern manufacturing

This appendix was the fourth chapter of the preliminary study on existing literarture for this thesis. Some of the original content, specifically that which dealt with fabric formwork technology has been removed. Instead, that information now forms the introductory paragraphs for the chapter of fabric formwork technology, Chapter 3.

A building, or at least its structural frame, is made of one of only a few typical construction materials. The very nature of a building calls for large volumes of an economically attractive material that has a reasonable to high load-carrying capacity. Especially in the industrialized countries, the use of these 'cheap' materials needs to be low labor intensive. The cost of labor in such countries has led to an increased focus on developing prefabrication of structures and structural elements.

There are four main building materials: concrete, steel, wood and, increasingly, fiber-reinforced polymers. In this chapter each material is discussed, as well as their specific manufacturing processes. (B.1 to B.4) An attempt is made to include to newest developments in these areas. For this reason an additional paragraph (B.5) deals with rapid manufacturing, a term associated with newly emerging manufacturing processes that have not yet found widespread application in the building industry.

The aim of this chapter is to limit the scope of this thesis by selecting a specific material and manufacturing method, which is best suited to combine with computational optimization.
The assessment of each material focuses on the mechanical and physical properties, and the relative durability and sustainability in their use.
A suitable manufacturing method is selected based on its ability to approximate the optimized shapes presented in the previous chapter as well as some other criteria.

B.1 Concrete

Concrete is a very common construction material and is used in a wide range of building applications. It is a man-made composite material, consisting of cement, water and aggregate. It exhibits a high compressive strength, but is less resistant to tensile forces. Other virtues of this material include low permeability and free form production at room temperature. The aggregate within the concrete is composed of natural materials such as gravel and sand or crushed rock. The cement and water act as a binding medium for the aggregate and together they form a rock-like material, concrete. There are many types of cement and aggregates which together offer a wide range of concrete materials exhibiting different properties.

This paragraph aims to provide a comprehensive overview of the current state of concrete as a material and product.

B.1.1 Historic overview

The widespread application of concrete coincided with the industrial revolution, after Portland cement was discovered in the 18th century. However, contrary to steel, its modern

Figure B.2
The Pantheon in Rome, remains the most impressive of Roman structures, featuring a dome with an interior circumference of 43,3 metres. The dome was built of Roman concrete of unknown mixture and has survived largely unscathed since the year 125 AD.

competitor, the history of concrete can be taken as far back as the early Assyrians and Egyptians. It was the Romans however, who discovered the exceptional properties gained by the addition of pozzolana, a volcanic ash. The famous Roman military engineer and architect, Vitruvius Pollio, had this to say (Morgan, 1914):

There is also a kind of powder which from natural causes produces astonishing results. It is found in the neighborhood of Baiae and in the country belonging to the towns round about Mt. Vesuvius. This substance, when mixed with lime and rubble, not only lends strength to buildings of other kinds, but even when piers of it are constructed in the sea, they set hard under water. [..]
Hence, when the three substances, all formed on a similar principle by the force of fire, are mixed together, the water suddenly taken in makes them cohere, and the moisture quickly hardens them so that they set into a mass which neither the waves nor the force of the water can dissolve.
— Vitruvius Pollio

Unfortunately, the secret of concrete seemed to be lost throughout much of the middle ages and any development was put on hold until the material reemerged at the time of the industrial revolution. Luckily, since then, development has been steady and has resulted in new types of cement, refinement of the concrete mixtures, the discovery of reinforcement and new additive ingredients, as well as innovations in the production and application of structural concrete. Development continues to this day and remains the focal point of much academic research.

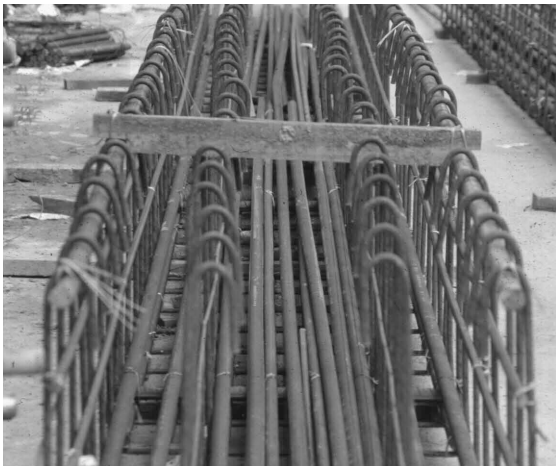
B.1.2 Innovations in concrete as a material

The ongoing research and development of concrete has resulted in several reinventions of concrete as a building material. In fact, most of these developments have occurred over the last few decades. In many cases, the building practice has a hard time keeping up.

The developments in concrete can be subdivided in the following categories, as each is briefly discussed:

- **Reinforced concrete**
  Plain concrete is not well equipped to handle tensile forces. It will exhibit a relatively low bearing capacity and brittle behavior. The first idea to augment concrete came in the form of steel reinforcement, effectively creating a new composite material. The combination of concrete and steel, when properly designed, uses the best of both worlds and is able to cope with high bending moments as the composite mechanical properties effectively deal with both high compressive and tensile forces.

Reinforced concrete requires insight in the different



stresses acting inside an element or structure. The material can be tailored to deal with localized tensile stresses as well as shear forces by carefully distributing steel reinforcement bars.
For cases where greater chemical or electrical resistance or the risk of corrosion is high polymer reinforcement bars are available.

- **Prestressed concrete**
  The second step in combining concrete with steel was prestressing, pioneered by Eugène Freyssinet in the early 20th century. By prestressing steel tendons placed along a length of concrete it is possible to introduce a compressive force, but also a bending moment. The former reduces or eliminates tensile stresses in the concrete while the latter can counteracts the bending moment due to dead and/or live loads. Often concrete is combined with both reinforcement and prestressing steel. There are three major types of prestressed concrete; pre-tensioned concrete and unbonded or bonded post-tensioned concrete. More information can be found in the book by Walraven (1997).
  The obvious advantages of prestressing concrete are increased spans and/or material reductions.

- **Fibre-reinforced concrete (FRC)**
  An even more recent development is the addition of various fibers which improve the mechanical properties of concrete during post-cracking behavior, namely the tensile strength. The principle of fiber reinforcement is the same as conventional reinforcement bars, but in reality there are some differences in yielding behavior. The reinforcement is no longer discretely, but randomly distributed, thus the composite material becomes approximately isotropic in nature.
  Examples of the following applied fiber materials were found:
  - Steel (SFRC)
  - Inox
  - Cellulose
  - Thermoplastic polymers
    - Polypropylene (PP)
    - Nylon
  - Asbestos
  - Glass fiber
  - Carbon fiber

  Steel fibers are currently most common, but carbon is sometimes preferred in high strength concrete, because it has a higher tensile strength than steel.

Figure B.3
There are many different shapes of steel fibers available for concrete reinforcement. The size and shape determine how the reinforcement reacts to various types of cracking mechanisms, and the shape specifically determines the pull-out capacity of the fiber.
Source: Weiler & Grosse (1996)

  The advantages of fiber reinforcement are (Walraven):
  - Simplicity; no complex reinforcement cages needed
  - Homogeneity; the effects of localized high stresses/concentrated loads can be anticipated more easily

  Depending on the type of fiber, some of the following advantages may be expected as well (Orgass & Klug, 2004):
  - Increase of fracture energy, subsequent improvement of ductility
  - Increase of strength
  - Reduction of tendency for cracking
  - Decrease of microscopic crack growth
  - Gain in fire resistance
  - Decrease of early shrinkage
  - Reduction of internal stresses within fresh concrete

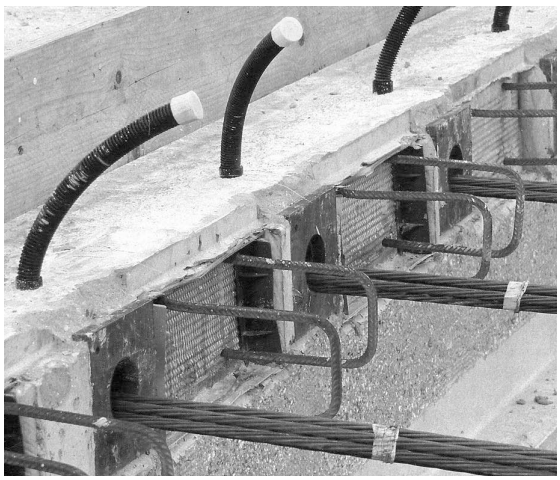
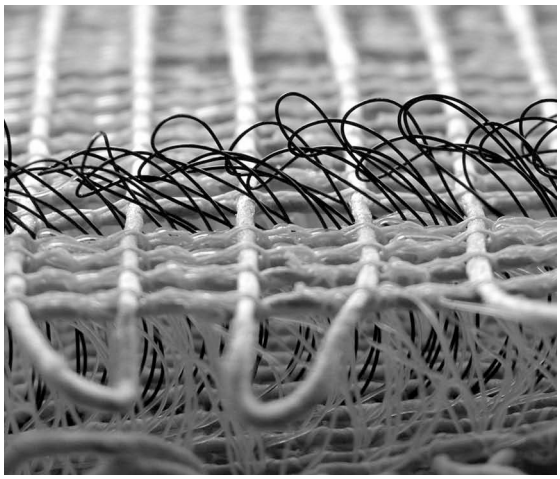


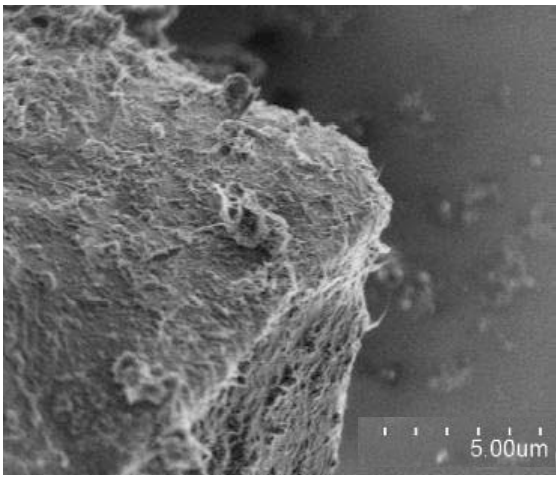
Figure B.4-6
From left to right:

- Traditional steel reinforcement cage on site prior to casting the concrete.
- Several prestressing tendons prior to casting the concrete
- Steel fibers protruding from a concrete test sample.

Figure B.7-8
From left to right:

- Textile reinforcement prior to pouring the concrete.
- Carbon nanotubes distributed on a relatively large cement grain as seen through a scanning electron microscope (SEM)





The strength of the concrete is improved by (Walraven):

- o High fiber content - constrained by loss of work ability
- o Higher ratio of the length to the diameter of the fibers (*L/D* ratio)
- o High pull-out capacity improved by the fiber shape
- o Degree of bonding between fiber and matrix
- o Fiber orientation according to the direction of loading (it is no longer isotropic!)

Some final remarks:
It has been shown that a combination of fiber with conventional reinforcement is best equipped to handle the largest tensile stresses. However certain minimal reinforcement requirements can be dropped and a decrease in the amount of shear reinforcement can be expected as well, eliminating the need for reinforcement steel at certain points. (Jungwirth & Muttoni, 2004)

The combination of short and long fibers has shown resulting flexural tensile strengths up to 45 MPa. For each cracking behavior there are suitable fibers to become active as micro- or macrocracks occur. (Orgass & Klug, 2004)

The concrete flow influences the orientation of the fibres as X-ray photographs have shown. This effect is even more pronounced in samples with longer fibers. This can be used advantageously by ensuring the concrete flow is similar to the force paths in the hardened, loaded state. (Grünewald, 2004)

- **Textile reinforced concrete (TRC)**
  A recent, and logical, development in fiber reinforcement is so-called textile reinforced concrete, or net-reinforced concrete. The introduction of textile production techniques to create two- or threedimensional continuous reinforcement, results in a practical way of manufacturing reinforcement. The strength of the concrete becomes more predictable compared to FRC, and the material cost may be lower. On the other hand, preparations in placing the reinforcement take more time, and as in conventional reinforced concrete a cover is needed and double-curved shapes are hard to achieve. (Van Roosbroeck, 2006)
  Developments in this area are all fairly new, but applications can already be found such as shear reinforcement and repairing or strengthening existing structures.

  This type of reinforcement is many ways similar to certain types of fiber reinforced polymers as discussed in paragraph B.4.3.

- **Carbon nanotubes (CNTs)**
  Even more recent and arguably more exiting, are carbon nanotubes. Nanotechnology has led to the possibility of creating the smallest fibers imaginable, exhibiting the most promising properties. CNTs are manufactured using one of the following three production methods, each with its own colorful name; electric-arc discharge, laser ablation and chemical vapour deposition. The result is the same as each method creates free carbon atoms which, under the right conditions, form tubes of controllable length rather than graphite or amorhphous carbon. Two types of tubes can be distinguished; single walled and multi walled nanotubes (SWNTs or MWNTs).
  Experiments have shown moduli of elasticity exceeding 1 TPa, yield strengths of 63 GPa and yield strains of 6%. At the moment they seem to be, in themselves,

the mechanically strongest and most thermally conductive material known to man. They also show interesting electrical behaviour, such as conductivity, depending on their shape and size. (Makar & Beaudoin, 2003)
An experiment with foam concrete made of CNTs and Portlandcement by Yakovlev et al. (2006) has shown an increase in compressive strength of 70% in addition to a decrease in thermal conductivity by 12-20%. The latter result can be explained by the fact that the thermal conductivity is an anisotropic property, highly dependant on the orientation and distribution of the CNTs. The use of CNTs can extend to glass and polymers, besides cementitious mixtures.

- **High performance concrete (HPC)**
  Concrete strength depends on the water-cement ratio. If this ratio is too low, the material loses workability to an extent that is is no longer a viable building material. The invention and addition of so-called superplasticizers in the 1990's made it possible to lower the w/c ratio from a minimum of 0,4 to 0,3 without a loss in workability. The use of admixtures such as these superplasticizers has brought about much higher strengths than would normally be possible, above 65 N/mm^2. These types of concrete are referred to as high performance concretes (HPC) or high strength concretes (HSC).

  Besides higher strengths, HPC offers numerous advantages:
  - o Faster hardening and strength development
  - o Less deformation due to creep and shrinkage
  - o Increased abrasion resistance
  - o Increased durability
  - o High chemical resistance
  - o Flowability

  The newest recommendations and building codes adopt rules for HPC, showing that HPC is increasingly used in construction.

- **Self compacting concrete (SCC)**
  An unexpected property of the high performance concretes was self-compaction. Experiments started to focus on improving this quality, as it was soon recognized that self-compaction would bring along many advantages.
  By now, self compacting concrete (SCC) has become an economically viable product, specifically in the precast concrete industry, where many companies have made the transition to this new concrete mixture.

  The advantages of SCC are mainly due to the fact that mechanical compaction is no longer necessary, and also because of the flowability and viscosity of the material. The advantages are (Walraven):
  - o Noise reduction; making pouring at night also a possibility
  - o Less labor-intensive
  - o Better labor conditions; less noise, vibrations and dust
  - o Better homogeneity, no longer dependant on the quality of compaction
  - o High concrete quality even with high reinforcement density
  - o Casting in remote areas out of reach of vibrators
  - o High shape complexity
  - o More cohesive; less leaks through joints in formwork

  The right combination of flowability and viscosity will result in homogeneous, evenly distributed concrete.
  In contrast, SCC has few and minor disadvantages.
  - o The bending stiffness is slightly reduced compared to conventional concrete
  - o Formwork pressures are slightly higher, though measures are not necessary

  Though the material has become more expensive, examples in the building industry have already shown SCC and HPC to be cost-effective, thus economically viable. In summary:
  - \- The mixture is more expensive due to additives
  - \+ Energy consumption is reduced; no compaction necessary
  - \+ The mould can be lighter and simpler (no vibration loads/dynamic) and has a longer service life
  - \+ Maintenance costs are reduced due to increased durability
  - \+ The labor conditions have improved leading to better health and less absence of the laborers.
  - \+ Less labor is required, no compaction and less reinforcement

  It is interesting to note that self compacting fiber-reinforced concrete (SCFRC) has been proven to show a higher bearing capacity than FRC with the same fiber content and concrete strength.

- **Ultra high performance concrete (UHPC)**
  Research on high strength concrete has led to com-

pression strengths exceeding 200 N/mm². By hardening the concrete under high temperatures or pressures significantly higher strengths are possible still. Strengths as high as B200 are available as prefabricated products, while under laboratory conditions, concrete strengths of up to 800 N/mm² have been achieved. The term ultra high performance concrete (UHPC) was coined to denote these highest strengths. Unfortunately the difference between HPC and UHPC is not well defined. Two definitions are offered:

1. HPC is concrete with a low w/c ratio with a compressive cube strength up to 200 MPa, while UHPC has compressive strengths above that.
2. HPC achieves high compression strength due to superplasticizers, while UHPC also contains silica fume and quartz sand and powder to replace conventional, more coarse aggregates.

An alternative term is reactive powder concrete (RPC), which is defined as concrete with compression strengths between 200 and 800 N/mm². RPC is developed by adhering to four principles (Richard & Cheyrezy, 1995):

- Enhancement of homogeneity by elimination of coarse aggregates
- Enhancement of compacted density by optimization of the granular mixture, and application of pressure before and during setting
- Enhancement of the microstructure by post-set heat-treating
- Enhancement of ductility by incorporating small-sized steel fibers

The advantages of UHPC are the same as with HPC, or even more so. However, UHPC shows brittle behavior at failure and the tensile strength increases under-proportionally with the compression strength (Holschemacher et al., 2004). These drawbacks are partially compensated by the addition of fiber reinforcement.

B.1.3 Mechanical properties

There are several mechanical properties that altogether provide a good overview of how a certain material will behave under loading conditions. The range of these various properties is taken from several sources and will form the basis for a solid comparison with the other materials that are evaluated in this thesis.

The properties of UHPC are particularly difficult to ascertain, because this material does not yet show up in national building codes. The information on these properties has therefore been taken from several research papers. These sources are seperately shown in table B.1 and summarized in the second table B.2 as one UHPC column.
Heavy Reinforced Ultra High Performance Concrete (HRUHPC) in table B.1 refers to a steel and steel fiber reinforced concrete, developed for specific high strength applications such as the walls of bank vaults.

Table B.1

property	*range*					*unit*
	UHPC	**UHPC**	**UHPC**	**HRUHPC**	**RPC**	
Fiber content		0-2	4-12		2-3	%
Characteristic cube strength f'_{ck}	70-220	120-270	160-400	160-400	200-800	N/mm²
Compressive strength f'_b					194-520	N/mm²
Tensile strength f_b						N/mm²
Average tensile strength f_{bm}		6-15	10-30	100-300		N/mm²
Modulus of elasticity E'_b	34.700-53.400	60.000-100.000	60.000-100.000	60.000-100.000	62.000-74.000	N/mm²
Yield strain ε'_{bpl}						‰
Ultimate strain ε'_{bu}						‰
Poisson's ratio υ	0.18				0,19-0,28	-
Thermal expansion α						K^{-1}
Density ρ		2500-2800	2600-3200	3000-4000		kg/m³
Fracture energy		150-1500	5000-4000	20.000-400.000	1000-40.000	N/m
Source	Ma & Schneider (2002)	Buitelaar (2004)	Buitelaar (2004)	Buitelaar (2004)	Dugat et al. (1996)	

The properties other than the strengths and strains are briefly explained:

The modulus of elasticity E, or Young's modulus is a measure of stiffness of a material. It is its tendency to deform and therefore determined by the ratio of stress divided by strain. This ratio is often assumed to be constant for a certain range of strains, up to a certain strain limit (or yield strain) in accordance with Hooke's law.

Fracture energy is a measurement for the ductility. Ductility is the extent to which material yields under shear stress and exhibits plastic deformation without fractures occurring (up to the ultimate strain).

Poisson's ratio υ is a measurement of the thinning that occurs transversely to the direction of tension. It is the relative contraction strain divided by the relative extension strain.

Concerning the properties of UHPC, the following is noted:

The modulus of elasticity and tensile strength both develop underproportionately to the compressive strength of UHPC. This phenomenon is generally compensated by the addition of fiber reinforcement.

The upper bound fracture energy of UHPC seems to increase remarkably, though this is somewhat deceiving because this increase is caused by the fiber reinforcement. In traditional concrete, cracking is more pronounced after which the steel reinforcement takes over the tensile forces.

No general information was found on the development of yield and ultimate strains in UHPC, but numerous case studies mention values similar to normal concrete or values up to ultimate strains of 25‰ (Schmidt et al., 2004).

Limited information was found on the development of the thermal expansion coefficient of UHPC, but two sources mention higher, experimental values for hardened UHPC from 12^{-5} (Fehling et al., 2004), 15^{-5} up to $22^{-5}\,K^{-1}$ (Staquet & Espion, 2004).

The first source also notes that both shrinkage and creep of UHPC are generally less than in conventional concrete due to the density of the microstructure. (Fehling et al., 2004)

Table B.2

property	*range*			*unit*
	Concrete B15 to B65	**HPC** C53/65 to C90/105	**UHPC**	
Characteristic cube strength	f'_{ck}	15-65	65-105	70-800	N/mm^2 or MPa
Compressive strength	f'_{b}	9-39	39-60	N/A-520	N/mm^2
Tensile strength	f_{b}	0,90-2,15	2,15-2,55		N/mm^2
Average tensile strength	f_{bm}	1,8-4,3	4,3-5,1	6-300	N/mm^2
Modulus of elasticity	E'_{b}	26.000-38.500	38.500-40.100	34.700-100.000	N/mm^2
Yield strain	ε'_{bpl}	1,75	1,75-1,90		‰
Ultimate strain	ε'_{bu}	3,50	3,50-2,50		‰
Poisson's ratio	υ	0,1-0,2		0.18-0.28	-
Thermal expansion	α	9-12			$10^{-6}K^{-1}$
Thermal conductivity	k	1,8-2,5			W/mk^{-1}
Density	ρ	2500	2500	2500-4000	kg/m^3
Fracture energy		130	140-150	150-40.000	J/m^2 or N/m
Source		NEN6720 Jackson & Dhir	CUR Aanb. 97		

B.1.4 Durability

Concrete is generally known as a highly durable material, emphasized by the survival of historic Roman landmarks, such as the Pantheon. However, concrete is still susceptible to the influences of a chemical, physical or mechanical nature. Careful design and detailing, while considering the environment of the structure, will avoid most problems and yield a building that will last many generations.
The following influences exist (Reinhardt, 2004, pg. 219):

- Chemical attack
  Acids, sulfates and some salts (e.g. ammonium chloride NH_4Cl and ammonium nitrate NH_4NO_3) in combination with rain- or groundwater or in gaseous form will react with the cement from the outside. Contaminants in the aggregates in the concrete can react with the water or cement and create expansion.

- Physical attack
  Other than abrasion, concrete is also vulnerable to frost, fire and solar radiation. Temperature or shrinkage stresses inside the concrete may also cause significant damage when neglected.

- Corrosion of the reinforcement
  Chloride penetration or carbonation from the outside will cause the corrosion of the reinforcement steel. Carbonation is the leaching of calcium hydroxide ($Ca(OH)_2$) through a reaction with carbon dioxide CO_2. This acidifies the content of the concrete pories (which chloride penetration does directly) resulting in deterioration of the steel coating.
  Chloride contamination of the fresh concrete mix will directly affect the concrete quality.

The resistance and thus the durability of the concrete is largely dictated by protection against concrete carbonation and chloride attack.
The resistance to these deterioration mechanisms is primarily determined by the pore size, the cumulative porosity and the connectivity of the pores. These in turn largely determined by the w/c ratio. The addition of water-reducing plasticizers and optimizing the granular composition of the mix by adding fine particles positively affect the w/c ratio. These additions are inherent to HPCs and UHPCs.

Roux et al. (1996) and Herold & Scheydt (2006) tested and compared concrete mixes for durability by respectively testing C30, C80 and RPC200 and C30/37, C90/105, RPC and UHPC for porosity, permeability, water absorption, corrosion behavior and in the former case also for migration of chloride ions, electric resistivity and resistance to abrasion.

They drew similar conclusions with regard to the newer RPC and UHPC concretes having shown “extremely high resistance to the penetration of aggressive agents” and “excellent durability characteristics [..] yielding a significant increase in the life expectancy”.
This is not surprising since the w/c ratio is 15-10% in conventional concrete while it is around 0,5-1,0% for RPC

Fire damage to concrete occurs in the form of cracking, or often as spalling, which may happen in an explosive fashion shortly after exposure or less violently over longer periods. At temperatures higher than 500C, the compressice strength may reduce significantly. (Jackson & Dhir, 1996)
Heinz et al. (2004) tested the fire resistance of UHPC and concluded it was more sensitive to fire than conventional concrete. This outcome was attributed to the higher density of the concrete microstructure. Recommendations included the improvement of fire resistance by using steel or polypropylene (PP) fibers and using fire resistance surface coatings.
Concrete manufacturer Ductal however has developed a concrete mix of 150-180 MPa to be fire resistant and has behaved excellently during testing, comparable to normal concrete.

B.1.5 Sustainability

Concrete is:
- manufactured from raw finite materials and
- contains the energy rich component of cement

but,
- the raw material supply will suffice for many centuries to come,
- concrete has a long life and requires little maintenance; provided it is designed and constructed correctly,
- the material can be reprocessed to a wide range of high-grade aggregates after the first life cycle and
- concrete is suitable for reuse as a material, as long as the elements can be dismantled and possibly adjusted.

In fact Hendriks et al. (2000) states that in the Netherlands, 90% must be reused, of which approx. 65% is being reused as an aggregate, primarily as a base course material in road construction. Prefab concrete manufacturer VBI states on its website, that at the request of the client up to 20% of recycled aggregate can be reused in new products.

Racky (2004) compared the energy and raw material consumption for C40/50, C80/95 and C180. Energy consumption was nearly three times higher per unit volume for C180 compared to C40/50, and raw material consumption more than one and a half times more. But when taking the increased strength into account, thus the reduced volume needed, the results show lower energy requirements and much lower material needs (both concrete and formwork). It continues by showing the potential of UHPC as a cost-effective material, concluding that UHPC is the best option for the "minimization of use of non-renewable resources".

B.1.6 Manufacturing processes

Concrete is perfectly suited for construction on site, or casting *in situ*. Prefabrication, or casting at an industrial plant has also become increasingly popular. Before discussing several specific production methods, both principles are briefly introduced.

- Cast in situ
  The oldest and most typical way of constructing concrete structures is by casting in situ, pouring concrete on site. This is possible because concrete hardens at room temperature, which is actually one of its greatest advantages. Casting concrete requires some sort of mold, typically made of timber or steel, supported to carry the weight of the concrete in fresh state and perhaps lined to provide a smooth surface in the final and hardened state.

  Constructing a mold on site avoids major transportation issues, as both mold materials and concrete ingredients can all be supplied separately, the latter in bulk. The mold itself can be a single, tailor-made work of art or a modular system, reusable to economically produce repetitive, structural systems.

- Prefabricated
  The practical opposite of casting in situ is prefabrication. The advantages offered are all a result of the controlled, climatic environment that factory conditions provide. Both the mold and the concrete can be carefully made, resulting in high quality, high strength, and consistent structural elements at reduced cost. If the logistics are carefully planned, the building time is reduced as well because effectively more construction tasks may be performed simultaneously; elements are prefabricated, while on site preparations are already underway for assembly. When it comes to prefab elements, transportation and joints become major design issues to be taken into account with care.
  Prefabrication takes place in different ways and has become increasingly automated using robotics and various ways of discharging the concrete mix.

The following production methods apply to casting in situ and/or prefabrication. With the exception of traditional molds, slip- and jumpforming and pneumatic formwork, all methods explained are strictly applied in prefabrication processes.

- Molds
  The advantage of decreasing the overall building time is easily achieved by casting elements at a separate location. The molds, as mentioned, are often made from timber or steel.

- Slipforming or jumpforming
  This method uses a mechanically moving formwork. It is both applied on a large and small scale.
  In the former case a giant formwork is vertically jacked

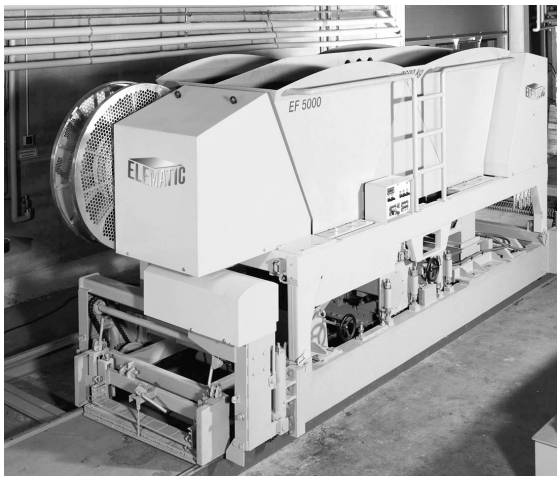


Figure B.10-11
From left to right: A slipforming machine is used by many prefabrication companies for the production of hollow-core slabs, wall panels, T-beams and solid slabs.

An extrusion machine differs slightly and has similar purposes.
Courtesy of Elematic

up while concrete is added to produce, for example, a core for a high-rise building. Slipforming uses a continuously moving mold, while jumpforming uses a discretely moving mold.
In the latter case of small scale slipforming, a machine is used in a factory to repetitively produce prefab elements. Slipforming is especially suited for robust elements of great variety, and similar to high-rise cores used in a vertical sense to create wall elements (Anon., 2005).

- Extrusion
  Most prefabrication companies use either or both slipforming and extrusion for their production process. Extruders are more suited for producing lighter and longer products. For instance, hollowcore slabs with the same thickness made with an extruder has shown to be 10% lighter than those made by slipforming. The compaction of the concrete mix in both slipforming and extrusion is achieved by high frequency vibrations or shear compaction. Extruders can produce slabs with a width of 600–2400 mm (Anon., 2005).

- Spinning, or centrifugal casting
  Centrifugal casting is a method where a axi-symmetrical mold is rotated, while a liquid material inside is spun and hardens. On a small scale it is used for metal or plastic objects, but in the prefab industry spun concrete is an excellent method for producing columns, poles or tubes. Round, but also hexagonal or octagonal shapes can be easily made. Prestressing tendons can be placed in a longitudinal direction, parallel to the rotational axis.

- Mold milling and hot wire cutting
  Recent years have seen double-curved shapes being designed in new 3D-computer CAD-software. Such shapes require double-curved molds in turn, traditionally made by hand, from materials such as wood, but more recently from wax, polyurethane, polystyrene and concrete itself.
  The most cost-efficient thus far seems to be polystyrene and research efforts focus on this material and optimizing the combination of hot-wire cutting and milling to create these double-curved shapes (Wapperom, 1999). By using computer-controlled machines, extruded polystyrene (EPS) blocks are cut and milled, typically in 20mm thick slices. The high precision of these machines allow small and detailed elements. On the other hand, a single mould of dimensions up to 13 x 4.00 x 1.85mm is also possible. By combining several molds, for example in multiple slices, virtually any size can be realized. (http://www.polysiertegels.be)
  The molds can easily be removed afterwards and waste and residue materials can be recycled, though some decrease in quality may be expected.
  Disadvantages of these molds include the buoyancy of EPS in concrete and the rough surface finishing requiring expensive foils or wax layers (Wapperom, 2005).

- Contour crafting (CC)
  This technique refers to a layer-based manufacturing process. A computer-controlled nozzle applies consecuvtive layers of concrete. The nature of this method implies vertically stacking these layers, but examples exist of tapered elements and cones. Domes and vaulted structures are also possible, but have not yet been made. Machines exist to also repetitively place reinforcement dowels. Contour crafting offers rapid fabrication times of a finished product with minimal to no material losses, without the use of any mould. Ambitious plans have been drawn to develop this

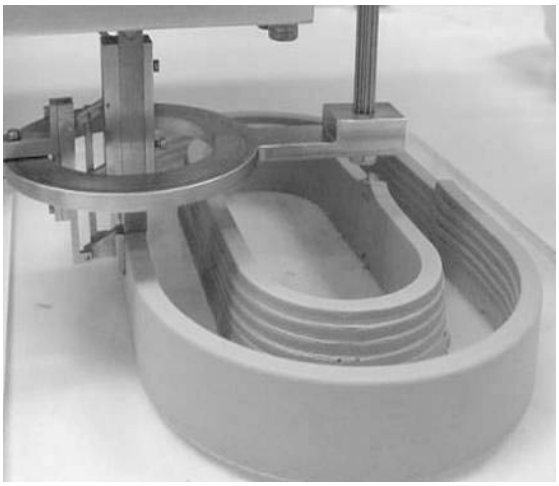


Figure B.12-13
From left to right:
This 5-axis milling machine is claimed to be the largest in the Netherlands. It is computer numerical controlled (CNC) to mill prototypes and models.
Source: http://www.nedcam.com

A CNC machine deposits layers of concrete, crafting contours while producing a vertical hollow section, which is reinforced and filled with concrete afterwards.
Source: http://www.contourcrafting.org

method into producing fully prefabricated, finished homes as well as extraterrestrial structures. (Khoshnevis, 2004)

- Fabric formwork
  See Chapter 3.

- Pneumatic formwork
  Double-curved architecture and blob designs ask for new construction methods. Using air, fabric formwork, or membranes, can be pressurized to easily produce double-curved shapes. The inside of the fabric can be used as a mould by spraying layers of concrete, more commonly referred to as shotcreting. Shotcreting which is easily combined with both conventional and fiber reinforcement and is used both on site and in prefabrication.

- Easy adjustable mould
  The easy adjustable mould is an experimental production method for double-curved planar elements. The mould has been proposed by ir. M. Quack, ir. H. Jansen and dr.ir. K. Vollers for respectively, the production of non-structural, glass fiber reinforced facade elements, the deformation of thermoplastics and the deformation of glass plates. The idea was further refined and tested by ir. M. van Roosbroeck for production of prefabricated concrete shells.
  The principle of the adjustable mould is a bed of pins, individually adjustable in height and covered by a rubber mat or polymer fabric.
  At the moment this technology is purely suited to closed sections for shells.

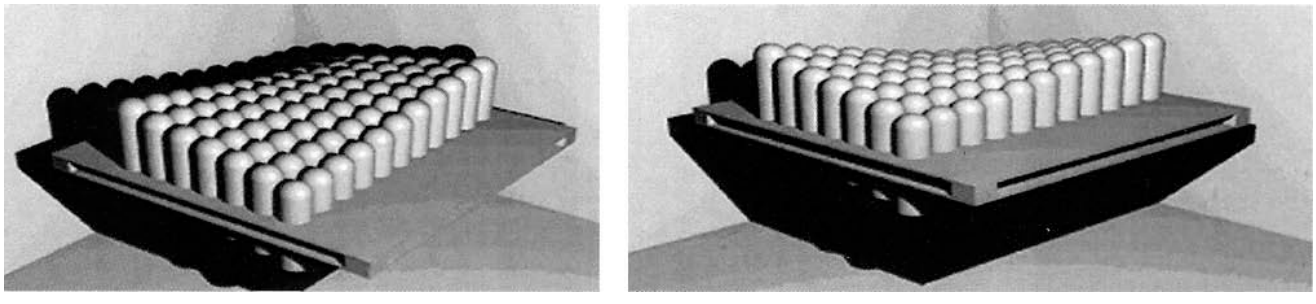


Figure B.14
The easy adjustable, or pin-bed mold is still experimental visualized in these sketches and models.
Source: Van Roosbroeck (2006)

B.2 Steel

The major competitor to concrete is steel. Steel is a metal alloy containing mainly iron and a small percentage of carbon. The development of steel, as with concrete, started in the industrial revolution, although some examples exist of early cultures producing some types of steel. The use of steel as a construction material can be attributed to the invention of the railroad. As steel was mass-produced for laying vast lengths of railroad tracks people began to understand the potential of structural steel, although the Old Testament illustrates that structural use of metal existed in early times as well:

> *Today I have made you a fortified city, an iron pillar and a bronze wall to stand against the whole land—against the kings of Judah, its officials, its priests and the people of the land.*
>
> *— Jeremiah 1:18*

In the 19th century several new manufacturing processes enabled the production of affordable, good quality structural steel.

The strength of steel is influenced through alloying and heat treatment. Alloying implies the addition of elements such as carbon and manganese to increase strength at the cost of weldability and other fabrication properties. Adding chromium, copper and nickel will result in weathering and stainless steels which are more resistant to corrosion. Heat treatment will alter the microstructure and grain size of the steel. (Jackson, Dhir, 1996)

B.2.1 Mechanical properties

Steel is structurally graded based on the minimum yield strength, which is the strength at which plastic deformation starts to occur. From this point on, steel exhibits very ductile behaviour, deforming considerably before the ultimate tensile strength is reached. Based on this classification, three types of structural steel can be distinguished: conventional steel, high performance steel and very high performance steel.

- High performance steel (HPS)
  The designation refers to higher performance in tensile strength, toughness, weldability, cold formability and corrosion resistance. The steels have become possible due to developments in production processes which enable fine microstructures within the material, enhance the alloying composition and decrease impurities in the material..
  Besides higher performance, advantages of HPS are improved durability and reduction in material use. (Günther et al., 2005)

Table B.3

property	*range*		*unit*
	Steel	**HPS and VHSS**	
	S235 to S460	S460 to S960	
Minimum yield strength	R_{ch} or $f_{y;rep}$	215-460	400-960	N/mm^2 or MPa
Tensile strength	R_m or $f_{t;rep}$	340-680	500-1150	N/mm^2
Modulus of elasticity	E'_b	210.000		N/mm^2
Yield strain	ε'_{bpl}	12,1-16,9		‰
Ultimate strain	ε'_{bu}	170-240		‰
Poisson's ratio	υ	0.27-0.30 (steel)		-
Thermal expansion	α	12		$10^{-6}K^{-1}$
Thermal conductivity	k	52		W/mK^{-1}
Density	ρ	7850		kg/m^3
Fracture energy				J/m^2 or N/m
Source		Soons (2002) Adriaansen et al. (1996) Jackson & Dhir	EN10025-6:2004	

- Very high strength steel (VHSS)
  Most steel applications use steel grades up to S690, but steel grades have been standardized up to S960 MPa yield stress as grades up to 1100 MPa can be manufactured.
  Steel above S690 is referred to as very high strength steel (VHSS).

The development of higher strength steel has been in motion since the '70s, but application in the building industry remains limited to tower cranes, high strength cables etc. One major limiting factor is the absence of national building codes including these higher graded steels.
Current research deals with introducing different alloying additions and improvements of the microstructure, though the results are very specific and it is safe to pose that development in steel is fairly limited.

B.2.2 Durability

Steel is generally considered to be durable, and many historical structures still standing attest to that. The main mechanism impacting steel durability is corrosion, an inevitable process requiring counteractive measures, regular inspection and maintenance.

There are basically four ways to counteract corrosion (Jackson & Dhir, 1996):

- Alloying additions
  Copper, nickel or chromium lead to 'weathering steels' with more than 12 percent chromium leading to highly corrosion resistance, or stainless steels. Stainless steel on the other hand, is vulnerable to pit corrosion. Pit corrosion is a local corrosive effect caused by chlorides that is difficult to inspect visually, possibly leading to a weakest link in the structures.
- Paint coatings
- Metallic coatings
- Cathodic protection
  Like metallic coatings, cathodic protection involves sacrificing another metal. A sacrificial, anodic metal is electrically connected to the structural steel, which is regularly inspected for excessive corrosion. This type of protection is common in highly corrosive situations, such as in the marine conditions.

Steel is also vulnerable to fire and needs some kind of protection. Steel structures are often encased, coated or otherwise treated to increase their fire resistance.

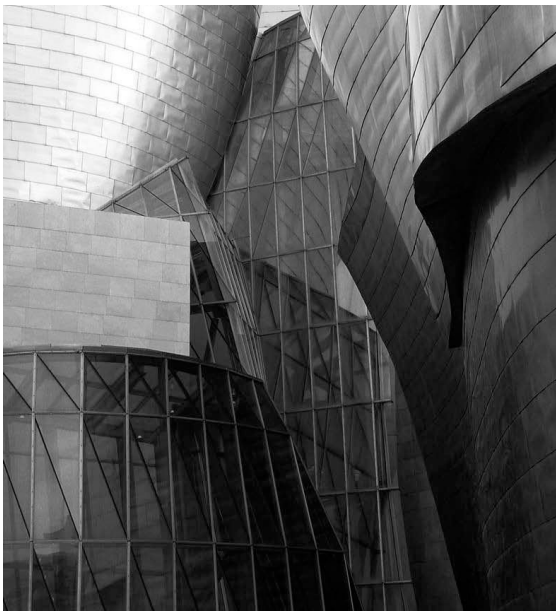


Figure B.15-17
The images show the Campo de Volantin pedestrian bridge in Bilbao, Spain, the roof of the British Museum in London and the entrance to the Guggenheim museum in Bilbao, Spain. The images illustrate how steel is used to create free form structures. It can be bent into shape, but in general, the geometric complexity of a building is derived from the topology of the structure, rather than the shape of the elements. In other words the careful arrangement of nodes and members result in the form free nature of the design.

At a temperature of 400ºC the strength and stiffness of steel will start to decrease. At around 600ºC the yield point is only half that of the normal situation and a mere 10% at 800ºC. (Adriaansen et al., 1996)

B.2.3 Sustainability

Steel is:

- manufactured from raw finite materials and
- produced by various high-energy processes, however,
- the reserve of materials will probably last several hundreds of years,
- steel can be reused fully by smelting and without loss of quality at 40% of the initial energy necessary for primary manufacturing,
- the residual value is high, an additional incentive for reuse and recycling and
- the current modular use of steel (standard sections), make dismantling and subsequent reuse viable.

Hendriks et al. (2000) claims that over 90% of steel in the Dutch building industry is collected and reused. Adriaansen et al. (1996) claims that 20% of new steel is made of recycled scrap metal.

The recycling of steel becomes difficult when the steel has to be seperated from other components or surface coatings. In some cases, such as in the automotive industry, this is not viable and the steel is in fact mixed with these other materials, diluted as such and ultimately downcycled to a lower quality and grade. It can no longer be used for the production of automobiles itself.
In principle, steel is a sustainable material, and the design and/or recycling processes are at fault when downcycling occurs.

B.2.4 Manufacturing processes

Steel is produced as sheet material and extruded sections. The manufacturing involves some type of heat treatment and rolling to form the product. The results however, do not vary significantly in terms of geometry of the final product, so the different methods are not discussed.
The final geometry of a steel structure, composed of plate steel and/or steel sections, is achieved by a combination of welding, flame cutting, rolling, drilling and bolting.

B.3 Timber

Perhaps wood is the first construction material ever used. Wood was certainly available to the earliest civilizations. When trees are cut down and processed for use the wood is generally called timber.

There are several types of timber for structural purposes. The most common is simple sawn timber, categorized as soft- and hardwoods. A problem of wood is the presence of imperfections, such as knots, that form weakest links in the material. To cope with problem there are numerous wood products that are made from layers or particles, bonded with adhesives. This decreases the probability of local weaknesses and improves the homogeneity of the product, making it more reliable than sawn timber.

Examples are (Alsmarker et al., 1995):

- Glued laminated timber (Glulam)
- Laminated Veneer Lumber (LVL)

and wood-based panels such as:

- Plywood
- Fiberboard
- Particleboard
- Oriented Strand Board (OSB)

Glulam and LVL are of particular interest, since they are primarily used as alternatives for structural members.

Table B.4		*property*	*range*				*unit*
			Softwoods	**Hardwoods**	**Glulam**	**LVL**	
			C14 to C50	D30 to D70	GL20 to GL 36	Kerto-S-LVL	
Bending strength		$f_{m,k}$	14-50	30-70	20-36	48-51	N/mm²
Tensile strength parallel to grain		$f_{t,0,k}$	8-30	18-42	15-27	42	N/mm²
Tensile strength		$f_{t,90,k}$	0,4-0,6	0,6	0,35-0,45	0,6	N/mm²
Compressive strength parallel to grain		$f_{c,0,k}$	16-23	23-34	29-31	42	N/mm²
Compressive strength perpendicular to grain		$f_{c,90,k}$	2,0-3,2	8,0-13,5	6,0-6,3	6-9	N/mm²
Shear strength		$f_{v,k}$	1,7-3,8	3,0-6,0	2,8-3,5	1,5-5,1	N/mm²
Modulus of elasticity	parallel to grain	$E_{0,mean}$	7000-16.000	10.000-20.000	10.000-14.500	14.000	N/mm²
	5-percentile value parallel to grain	$E_{0,05}$	4700-10.700	8000-16.800	8000-11.600	12.400	N/mm²
	perpendicular to grain	$E_{90,mean}$	230-530	640-1330			N/mm²
Shear modulus		G_{mean}	440-1000	10000-1250		960	N/mm²
Density	characteristic value	ρ_k	290-460	530-900	360-440	500	kg/m³
	mean value	ρ_{mean}	350-550	640-1080		520	kg/m³
Source			EN 338:2003	EN 338:2003	prEN 1194:1993	Alsmarker et al. (1995)	

B.3.1 Mechanical properties

The mechanical properties of timber are anisotropic i.e. they vary depending on the direction of the grain. Wood is also, in terms of strength, inferior to other common construction materials. An additional problem is the organic nature of sawn timber, which leads to diversity in the properties. Because of this, a probabilistic approach is taken when dealing with timber. The classification of timber is based on the probability that the mechanical properties within one class have a certain value. Therefore the values are designated as either characteristic, mean or 5-percentile values. The classification is done by visually and mechanically grading timber samples according to carefully outlined rules.

Wood will not conduct electricity and its thermal conductivity is low as wood is a natural insulator.
Creep and shrinkage are important factors in timber design and the amount of deformation due to these effects is highly influenced by the moisture content and temperature levels.

B.3.2 Durability

Wood is an organic material and is therefore subject to natural processes of decomposition. By proper design and through drying and treatment of the wood, timber structure can be erected that will last. Century old buildings made with structural timber elements are common in most older cities and remain perfectly functional. In the US, wood is the preferred construction material for residential buildings.

- Biological attack
  Timber is vulnerable to fungi, insects, termites and marine borers. The species of wood and its natural resistance to these biological agents highly influences the durability.
  Other factors are those that determine the level of exposure to these agents, such as humidity, temperature and climate variations. Ventilation, drainage,protecting the end grain of the wood and preventing moisture from accumulating are good methods of limiting biological attack.

The natural durability of timber depends on the wood type. Tropical hardwoods for instance, are very durable, while young softwoods are substantially less durable, and often more suited for temporary needs or short lifetime requirements.
Most countries have a durability classification based on standardized soil burial tests. The time it takes before the test samples — buried wood stakes — are affected by biological attack is recorded and provides a measure for durability. The classification usually does not offer a measurement in years, but rather a relative classification compared to other species. In spite of this, species considered durable often survive in ground contact from 20 years and up.

There are three ways of improving this natural durability:

- Preservative treatment
  For applications in the ground or water, or harsh outside weather conditions, several preservation methods exist to enhance the durability. These involve some sort of treatment or impregnation with preservatives, such as quaternary ammonium compounds, bifluorides, creosote oil and strongly fixing metal salts.
  These preservatives have some disadvantages. The chemicals in preservatives are poisonous to biological agents and in this toxicity lies the problem.
  Creosote contains certain hydrocarbons that are detrimental to the healt, causing skin cancer and damaging the nervous system. The metal salts, consist of heavy metals, all of which are poisonous to a varying degree, can cause irritation and in the case of chromium are carcinogenic. Bifluorides are rarely used nowadays and are acidic in nature. The use of these compounds is in some cases banned or restricted, but still not uncommon.

- Heat and/or pressure treatment
  A second way of making wood durable is modifying the physical structure of the wood. This involves some type of heat and/or pressure treatment of the wood, requiring higher energy consumption but no chemicals. These softwoods can attain a durability comparable to hardwoods. Several brands are commercially available, and the market share of this type of wood has probable increased over the last few years.

- Surface treatment
  A familiar method of improving the durability is applying some kind of surface coating, such as paint, varnish or stain. Though these might be toxic or otherwise environmentally unfriendly, water-based systems are increasingly available and less problematic.

When temperature exceeds 250ºC timber starts to decompose, giving of flammable gases and turning into charcoal. (Jackson & Dhir, 1996) In large sections, timber is a safer material than steel or reinforced concrete, because timber chars at predictable rates and timber is a very good thermal insulator, thus a poor thermal conductor. The char on the exterior is even less conductive (one sixth of pure solid timber) and provides a protective coating for the timber inside that is still able to carry loads. Combustibility is therfor dependent on the surface/volume-ratio and the density of the wood species. Charring rates can be calculated, as shown in Alsmarker et al. (1995) according to Eurocode 5. When fire damages large sections, exposed joints will become the weakest links, especially those carried out in steel or other materials that are more vulnerable to fire.

Figure B.17-18
The images show the entrance to the Weald & Downland museum in West Sussex, England and the Jean-Marie Tjibaou Cultural Center in Nouméa, New Caledonia. The images illustrate how timber is - in similar fashion as steel - used to create free form structures.

B.3.3 Sustainability

The general energy requirements for the production of wood are low. In theory, timber is also a sustainable resource, because the production, or forestation can be renewable, and as a product timber can decompose to biological nutrients for nature to reuse. Sustainable forest management and seals of approval do exist. Still, largely non-sustainable and forests are harvested at alarming rates, certainly in some countries where legislation and supervision are not as strict or well coordinated.
If used timber has remained in a good condition, it is likely

reused mostly due to the fact that it can easily be machined and altered. Besides reuse, wood can be recycled, or rather downcycled, in successive stages; resawn into smaller dimensions, converted to smaller particles for use in chipboard, fiberboard or even as pulp in the paper industry. In the last, final stage, it can be burned to convert to energy. (Jackson & Dhir)

The problem in sustainability of timber lies in the preservatives used in treating the wood. The reprocessing of treated wood is generally fairly limited, and is therefore often incinerated to recover the intrinsic energy.

B.3.4 Manufacturing processes

Wood is an excellent building material due to the posssibilities for machining. The beauty and complexity of many timber structures is derived from the possibility of sawing, drilling, milling and bending the wood as well as the various ways in which the elements can be joined.

The manufacturing processes on the other hand, are straightforward. After the timber is sawn into standard sizes it can be resawn into sheets and plies or shredded to particles and fibers. These elements are reassemblied by adhesives and applying pressure. Certainly the specific ways in which these processes take place differ, but they not to an extent that they are relevant to describe seperately in this thesis. The reader is referred to Alsmarker et al. (1995) for some more information on these various wood products.

B.4 Composites

Though reinforced concrete is technically a composite material as well, this paragraph refers specifically to reinforced plastics, or fibre-reinforced polymers (FRP). FRP are a composite of a liquid resin and fibres, cured to form a solid structural material. There are many resins and polymers available, resulting in a wide range of possible properties.

B.4.1 Historic overview

The principle of combining strength of different materials is an old idea. Ancient Egyptians already knew how to spin crude glass fibers and combine these with natural resins to form decorative articles (Starr, 2000). Other examples exist as well of their intuitive use of composites.

> *That same day Pharaoh gave this order to the slave drivers and foremen in charge of the people: "You are no longer to supply the people with straw for making bricks; let them go and gather their own straw."*
>
> *— Exodus 5:6 and 5:7*

The modern day composite can be traced back to John Hyatt, who developed celluloid in 1868. Composites became increasingly popular after succesful use in radar domes and aircraft in the Second World War. Today the most innovative use of fiber-reinforced polymers is found in aviation, aerospace and naval industries, where lightweight engineering materials are very important.

B.4.2 General composition

As mentioned, composites can be made using a number of resins and fibers. In this section both components are discussed at some length. The manufacturing of composite products is often done in layers referred to as lamina. The total product, consisting of lamina, is called a laminate. Other than a resin and a type of fiber reinforcement, the composite may also contain any of the following materials:

- Additives e.g. UV- and ozone-stabilizers, fire retardants and pigments
- Fillers e.g. calcium carbonate ($CaCO_3$), silica (SiO_2), hollow or solid glass microspheres, thermoset or

thermoplastic microspheres
- Non-reinforcing fibers e.g. thermoplastic polyester fibers (PET), polyamid fibers (PA or nylon), cotton
- Corematerials e.g. honey comb (aramid, aluminium), foam (PVC, PS, PUR), wood, fibermats

The composition can be tailored to its specific use. The various combinations of materials within the composite will greatly affect the final properties of the material. The proper use of these ingredient will determine:

- Stiffness
- Strength
- Fracture behaviour (fatigue/delaminations)
- Electric conductivity/resistance
- Thermal conductivity/resistance
- Durability (chemical and physical resistancies)
- Mechanical resistance to damage

B.4.3 Fibers

Some of the fibers used in composites are:
- Glass
- Aramid
- Carbon
- High density polyethylene (HDPE)
- Natural fibers (e.g. jute, flax, sisal en hemp)

However, for compliance with the EUROCOMP Design code (Clarke, 1996), a precursor for the new Eurocode, only the following glass fiber reinforcements may be used:
- E-glass (electric-grade)
- C-glass (chemical-grade)
- ECR-glass (electric-corrosion-resistant)

The form of the reinforcement will determine the quality and mechanical behavior of the final product. An overview of these forms is given (Clarke, 1996):

- Chopped strand mat (CSM)
  This is a non-woven planar material in which the glass fiber strands are chopped into short lengths and fairly evenly distributed and randomly orientated. The non-aligned nature of these materials with the random crossing of fibers does not allow fiber content to exceed about 25%..

- Continuous filament mat (CFM)
  The properties of CFM are similar to those of CFM, except that the fibers are continuous and swirled at random.

- Woven rovings (WR)
  Woven rovings are bidirectional reinforcements constructed from continuous strands of multifibers and used in automated composite processes such as pultrusion and filament winding.
  Fiber content may be restricted to lower than 40% unless process compaction is high, thus in a balanced roving the fiber content will be 20% in each of the two directions.

- Woven fabrics
  There is a wide array of possible weave patterns and depending on the selected pattern the fiber content may exceed 50%, also depending on the method of composite compaction.

- Non-crimp fabrics or stitched fabrics
  These fabrics exist as uni-, bi- or multidirectional fiber reinforcements. When this fabric is made from unidirectional (UD) fibers laid in parallel to each other, it can attain a fiber content of over 50% in the main direction.

In general the directionality of the fiber reinforcement is divided in the following classes, whcih depend on the chosen manufacturing process:

- Uni-directional (UD)
  The fibers lay mainly in the longitudinal direction.
- 0º/90º
  The fibers are distributed in equal amounts in the 0º and 90º directions.
- Quasi-isotropic (QI)
  The fibers are distributed in equal amounts in 0º, 45º, -45º and 90º directions, approximating an isotropic material.
- Isotropic
  The fibers are short and arbitrarily oriented.

B.4.4 Resins

There are basically two classes of resins; thermosets and thermoplastics.
Thermosets cure from a liquid state and become permanently solid. Thermoplastics become elastic and flexible at or above the glass transition temperature T_g where the stiffness of a the polymer decreases significantly. During production thermoplastics are processed at temperatures above T_g. At even higher temperatures they can be melted and therefore reused.
Thermosets are generally stronger than thermoplastics and have better thermal resistance but usually do not lend themselves to recycling.

Common thermosetting resins are:
- Polyester
- Vinyl ester
- Phenolic resin
- Epoxy (for high end applications)

For structural applications, typical thermoplastic resins are:
- Polypropylene (PP)
- Polystyrene (PS)
- Polyimide (PI)
- Polyetherimide (PEI)
- Polyamide (PA or Nylon)
- Polyamide-imide (PAI)
- Polyphenylene sulfide (PPS)
- Polyetheretherketone (PEEK)

B.4.5 Molding compounds

In some manufacturing processes intermediate compounds are used in which fibers and resin are already combined but not yet fully cured. The use of such compounds is often beneficial for the consistency in product quality. In general they contribute to the surface quality, lower styrene emissions and increased dimensional accuracy.

- Sheet Molding Compound (SMC) and
  Bulk Molding Compounds (BMC)
  Both molding compounds have a dough-like texture. They require relatively high molding pressures and steel moulds, making the production process more expensive. BMC fibers are shorter than those in SMC.

- Blanks
  Blanks are thick slices with discontinuous fibers, pre-impregnated with either thermoplastic or thermosetting resins.
  A common example is glass mat reinforced thermoplast (GMT). GMT is typically composed of polypropylene with a fiber content of 20% to 40%. After being heated in an infrared oven a high shape complexity can be achieved, but because the fibers do not melt the final surface quality might not be optimal.

- Prepregs
  Continuous bundles of fibres or fiber fabrics are impregnated with resin. While in cold storage they remain in a semi-hardened state.

B.4.6 Mechanical properties

As shown, the fibers can be oriented in many different ways. Though this implies that the material is anisotropic, the ways in which the fibers are manufactured and placed often lead to an approximately orthotropic, or even quasi-isotropic material. An orthotropic material has two perpendicular axes with elastic symmetry.
Design calculations are done according to the theory of elasticity, as the composites have shown to behave linearly up to the point of fracture. (CUR 2003-6)
Strength, stiffness and stress-strain properties of composites are highly variable and are a function of (Clarke et al., 1996),

- the fiber content (volume fraction of fibers) in the section of the composite, which itself is determined by
  - o the type of reinforcement and
  - o the manufacturing process,
- the fiber and matrix resin used and
- the directionality of the fibers with respect to the external loads.

The tables illustrate this as well. Table B.5 shows the properties of several uni-directional (UD) composites, while the Table B.7 compares the total range of these UD composites to mat and woven reinforced composites.

Many of the manufacturing methods are layer-based, resulting in lower nominal strain and strength values for the whole *laminate* than for the seperate *lamina*. Also, nominal values exist for the so-called interlaminar shear and tensile strengths, which are used to check the occurrence of failure between these lamina. This failure mode is called *delamination*.

- Strain
  Clarke et al. (1996) propose that the nominal yield strain should be 1,2% given that the fiber content is at least 20% and at least 15% in every direction. In the case of construction under heavy, constant pressure or where first ply failure is allowed, in the serviceability limit state, the nominal yield point must not exceed 0,27%.
  There are formulas for detailed calculation of the specific yield strains, which take into account the production method, the type of fiber reinforcement and several conditions during service. Furthermore reduction factors exist that take in to account any misalignment of the fibers, damage to the fibers, short fibers, voids, etc.

- Modulus of elasticity
  The range of the bending and shear stiffness of a single lamina, E and G, are shown in the graph. and compared with timber and concrete.
  This graph emphasized the impact that the type of fiber reinforcement has on the mechanical properties of the composite material.
  The values are calculated using the theories of Halpin & Tsai, and Manera who have derived semi-empirical equations (CUR 2004-6).

  When looking at the how the modulus of elasticity develops, it can be noticed that mat fiber-reinforced polymers are comparable to timber, while UD fiber-reinforced polymers are comparable to concrete. The upper bound modulus in the third example of table B.5 contains aramid fibers and almost approaches the bending stiffness of steel. However the comparison falters, when looking at the properties in the transversal direction. Here the bending stiffness is a mere third.

Other important properties that vary due to the composition of the material are the density, the coefficient of thermal expansion and the thermal conductivity.
In general the composites exhibiting superior properties are the more expensive. For each new project that incorporates fiber-reinforced polymers, the function and surroundings of the material will determine which fiber type and which resins are best suited to use.

B.4.7 Durability

The durability of various polymers and the reasons for their environmental degradation varies greatly, depending on the type of polymer. The detrimental changes that occurs, either physical or chemical, occur at a molecular level and since polymers differ in molecular structure, results in polymer-specific behavior. This makes it possible to find a

Table B.5

property	*range*						*unit*
	UD		**UD**		**UD**		
	Longitudinal	Transverse	Longitudinal	Transversal	Longitudinal	Transverse	
Compressive strength	f		300	80-150	270-1500	55-250	N/mm² or MPa
Tensile strength	f		300-1450	30-120	800-1700	30-70	N/mm²
Flexural strength	f		300-1450	40-180			N/mm²
Shear strength	f				35-90	35-90	N/mm²
Modulus of elasticity	E	30.800-51.400	8900-19.300	50.000	8.000	40.000-150.000	5500-10.000	N/mm²
Shear modulus	G	2800-6000				2500-6000		N/mm²
Compressive strain	ε	1,6	1,1					‰
Tensile strain	ε	2,4	0,34					‰
Shear strain	ε	0,58						‰
Poisson's ratio	υ	0,26-0,30				0,27-0,32		-
Thermal expansion	α	6,1-13,0	15,2-62,1	5,4-9		-2-7	20-60	$10^{-6}K^{-1}$
Thermal conductivity	k	0,55-0,82	0,31-0,53	0,25				W/mK^{-1}
Density	ρ			1800-2100		1350-2000		kg/m³
Source		CUR 2003-6		Pultron.com		Nijhof		
Type		-		50-85% glass		60% E-glass, SM-carbon, HT-carbon and HM-aramid		

polymer that behaves optimally for a certain situation but at the same time makes it hard to make general statements on the durability of polymers. Generally speaking, though, many, but not all polymers are resistant to substances normally considered to be aggressive, making them extremely suited in, for instance, chemical industrial facilities. (Jackson & Dhir, 1996)

Possible resulting molecular damage is rupture, cross-linking and degradation of the polymer chains.

- Biological attack
  Most synthetic polymers appear very resistant to biological attacks, such as microbial or fungal attacks. Additives can have an adverse effect on the performance of polymers in this respect.
- Fire
  All polymers are combustible, but vary in the specific temperature and ease at which they burn. Synthetic polymers also produce toxic substances when burning. The careful choice of fiber and resin, and the addition of mineral fillers and flame retardant can make a fiber-reinforced polymer more than suitable as a safe construction material. Polymers require attention, because they have obvious shortcomings when exposed to fire.

Glass fiber reinforced plastics are also susceptible to water penetration. The durability depends on maintaining a good adhesion of the fiber and matrix. This results in particular attention to the surface quality.

Table B.6
Main agents and modes of degradation in polymers

oxygen at moderate temp.	thermal oxidation
oxygen at high temp.	combustion
oxygen + UV-radiation	photo-oxidation
water	hydrolysis
heat alone	pyrolysis
ionising radiation	radiolysis
micro-organisms	biological attack
atmospheric oxygen + water + solar radiation	weathering, atmospheric degradation
solvents, organic liquids	softening/dissolution

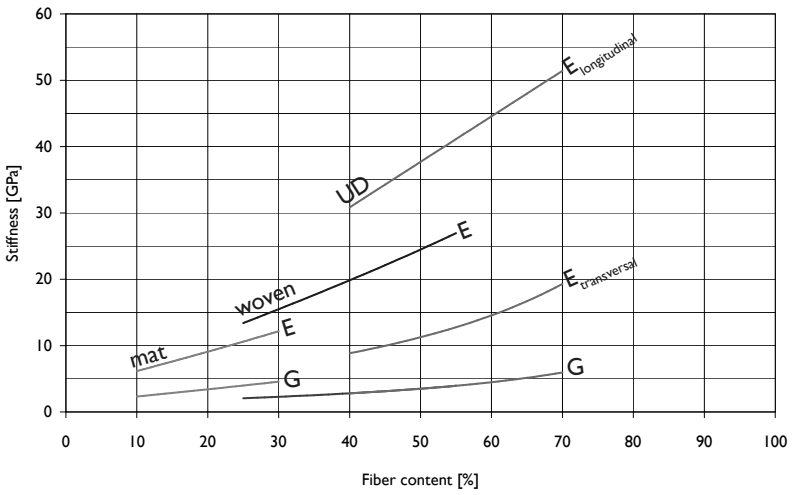


Figure B.19
Stiffnesses of the polymer as a function of the fiber content for different reinforcement types

Table B.7

	property	*range*				*unit*
		Mat	**Woven**	**UD**		
				Longitudinal	Transverse	
Compressive strength	f			270-1500	55-250	N/mm² or MPa
Tensile strength	f			300-1700	30-120	N/mm²
Flexural strength	f			300-1450	40-180	N/mm²
Shear strength	f			35-90	35-90	N/mm²
Modulus of elasticity	E	6200-12.200	13.400-27.000	30.800-150.000	5500-19.300	N/mm²
Shear modulus	G	2300-4600	2100-3900	2500-6000		N/mm²
Compressive strain	ε	1,9	1,5	1,6	1,1	‰
Tensile strain	ε	1,6	1,7	2,4	0,34	‰
Shear strain	ε	1,5	1,4	0,58	0,58	‰
Poisson's ratio	υ	0,33	0,18-0,21	0,26-0,32		-
Thermal expansion	α	16,3-64,1 / 33,2-105,8	11,0-37,2 / 21,3-82,3	-2-13,0	5,4-62,1	$10^{-6}K^{-1}$
Thermal conductivity	k	0,20-0,36	0,25-0,54	0,25-0,82	0,25-0,53	W/mK^{-1}
Density	ρ			1350-2100		kg/m³
Source		CUR 2003-6	CUR 2003-6			

B.4.8 Sustainability

The problem of plastics lies in their different properties and processing techniques. But to discuss the sustainability of fibre-reinforced polymers, the distinction between thermoplastics and thermosets has to be emphasized again.

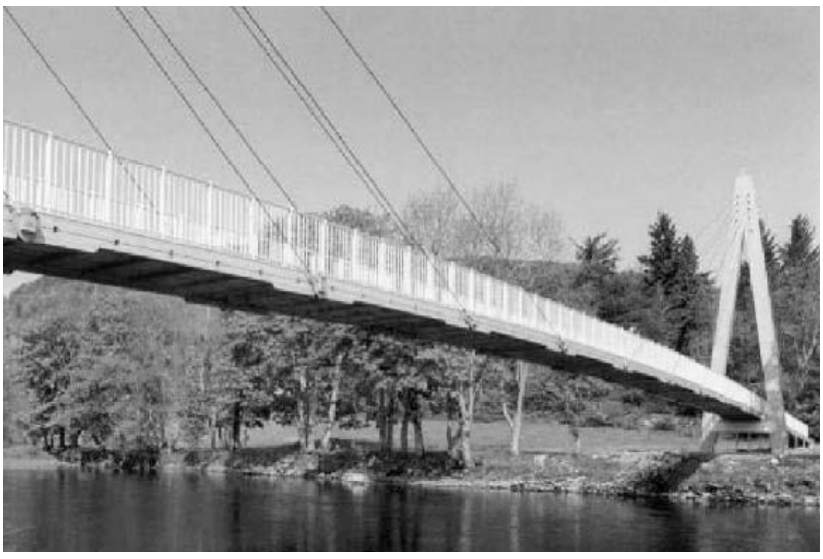
Certainly in the case of thermoplastics, recycling can be achieved to a high degree by heating. Practically, it is hard to distinguish various pure plastics and manufacturers cannot always be traced. This makes the waste processing a highly impractical process in which the original constituent materials are hard to retrieve. So, despite their high initial cost, plastics are either dumped or incinerated. In the latter case, some investment costs are retrieved, due to the high calorific value of polymers.

Hendriks et al. (2000) puts it bluntly: "*In the case of composites, there can be no question of high-grade recycling. [..] The high initial value of these plastics should be considered lost.*"

Before the 80's recycling possibilities were even only associated with remeltable thermoplastics. Thermosets were either deposited in refuse dumps or incinerated, but remained popular due to lower costs and better mechanical properties.
Three recycling alternatives now exist:

- Comminution
  This process, more commonly known as shredding, is fairly straightforward. The plastic material is shredded into smaller waste parts that are downcycled as fillers for other products.
- Particle recycling
  Bledzki & Goracy (1992) discuss particle recycling for waste management of glass fiber reinforced thermoset plastics. Particly recycling deals with disintegration of SMC/BMC products into a new material in a recycled form. The glass fibers are damaged as little as possible. The product is recycled to fibers and powdery fractions in a multistage process. They showed three seperate possibilities:
  - SMC paste was possible with up to 10% of the mass with recycled powdery fractions. One commercial example was given.
  - RTM printed circuits (for computers) were possible
  with up to 50% of the weight fibrous fractions.
  - BMC products could consist of up to 30% of recyclates while mechanical properties were reduced by no more than 20-30%.
- Pyrolysis
  De Marco et al. (1996) proposed pyrolysis, mentioned earlier as a mode for deegradation. Pyrolysis is basically heating without oxygen present and will decompose the organic parts of the plastic material to gases and liquids, and inorganic components, fiberglass and CaCO3. It is possible to recoup these constituent materials relatively unmodified.
  SMC product could be recycled through pryolysis for use in BMC. They concluded that no significant difference existed between BMC with recycled SMC components and virgin BMC. On the other hand, they also

Figure B.20-21 The Fiberline pedestrian bridge in Kolding, Denmark and the Aberfeldy pedestrian bridge in Perth and Kinross, Scotland. The material is used in a similar way to structural steel. The I-sections that these bridges are made from are pultruded



mentioned that only low quantities (6%) of pyrolysed SMC could be recycled in BMC with no adverse effects on the mechanical properties.

Two manufacturers of pultruded profiles, Fiberline and Pultron, were interviewed on recycling and admitted that no granulated remains were used in their pultruded profiles and instead their products were downcycled as filler in other products. The former speaks of recycling as costing too much, and both mention technical impossibilities in recycling pultruded profiles.
It seems that recycling of fiber reinforced polymers is still in development and not yet cost-effective. Indeed in both the academic and commercial world sustainability is seldomly a topic, while emphasis is placed on the mechanical virtues and durability of these products.

B.4.9 Manufacturing processes

Composite menufacturing methods range in nature from low capital-intensive, high-labor content, to the exact opposite, high-capital intensive, low-labor content technique. The following methods are presented in roughly this order.

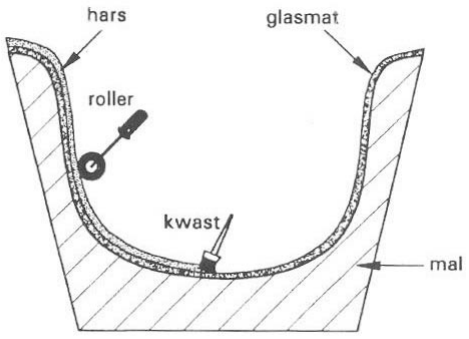
- Contact moulding
  The simplest of composite fabrication techniques is contact moulding. Because of the high labor content this method is typically used for prototypes or one-piece components in a single or limited quantity. Yacht hulls are a good example of its general application.

  A mould, usually made of plywood, is coated with a release agent and a gelcoat. This gelcoat results in a smooth surface but is also used for coloring and/or protection against ultiaviolet radiation or abrasion depending on requirements. Following this initial coating, each layer of fibres is then placed and coated with a resin. The application of resin is done by brushes and rollers or by spray deposition.
  Spray deposition is possible by hand and through robotics. This technique increases the overall quality of the final product and results in slight gains in production time.
  The material can be hardened at room temperature in the open air or under a vacuum, through a method called vacuumbagging, to allow (reinforcement ratio?) fiber contents up to 50%.

  Advantages
  - Low capital-intensive; single mould and simple tools
  - Fibers can be oriented in appropriate direction
  - Large dimensions possible
  - High shape complexity possible

  Disadvantages
  - High labor content
  - High emissions of styrene due to open mould; requires ventilation and protective gear
  - Smooth finishing at one side only
  - Reproducibility limited
  - Void content varies resulting in local weaknesses
  - Quality highly dependant on operator skill

- Resin injection
  Resin injection, resin infusion, resin-transfer moulding (RTM) or reaction injection moulding (RIM) offer many advantages over contact moulding.
  A mould, consisting of a lower and upper jig, is pressure injected. There are a few different methods; either the appropriate reinforcement is placed before the resin is injected such as with RTM or both are injected at the same time in a closed mould cavity such as with RIM.
  In some cases the upper jig is replaced by a foil; this method is called vacuum infusion.



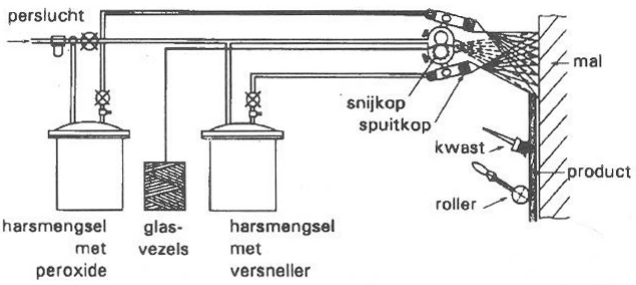


Figure B.22-23
Contact moulding and spray deposition

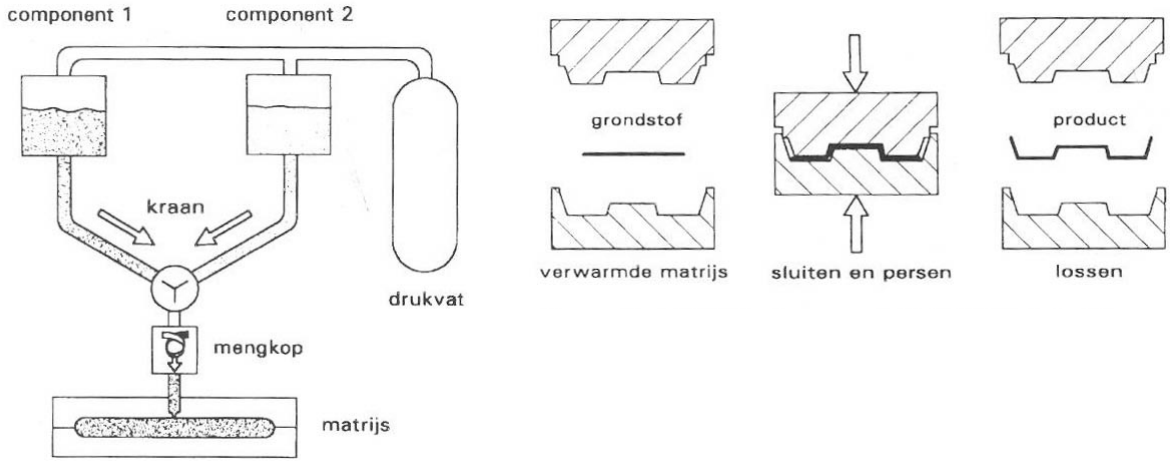


Figure B.24-26
From left to right: resin injection moulding (RIM), SMC/BMC compression moulding and pultrusion

Advantages
- Shorter production time
- Smooth finishing on both sides
- Low styrene emissions
- Over 50% fiber content/reinforcement ratio
- Clean process
- Larger range of reinforcement applicable
- Complex shapes possible in closed mould

Disadvantages
- Higher investments in mould and machinery
- Optional foil is relatively vulnerable
- Surface requires additional finishing
- Occasional release problems

- Compression moulding
  This term covers a variety of fabrication techniques. In general the fibres and resin are stamped or pressured between two jigs. Different techniques vary in temperature and pressure. Using higher temperature and pressures can increase the overall quality of the product and reduce production time, but results in more expensive moulds. A number of these methods employ a prepared, but uncured compound, discussed in more detail in Section B.4.5.

  Advantages
  - Good surface finishing

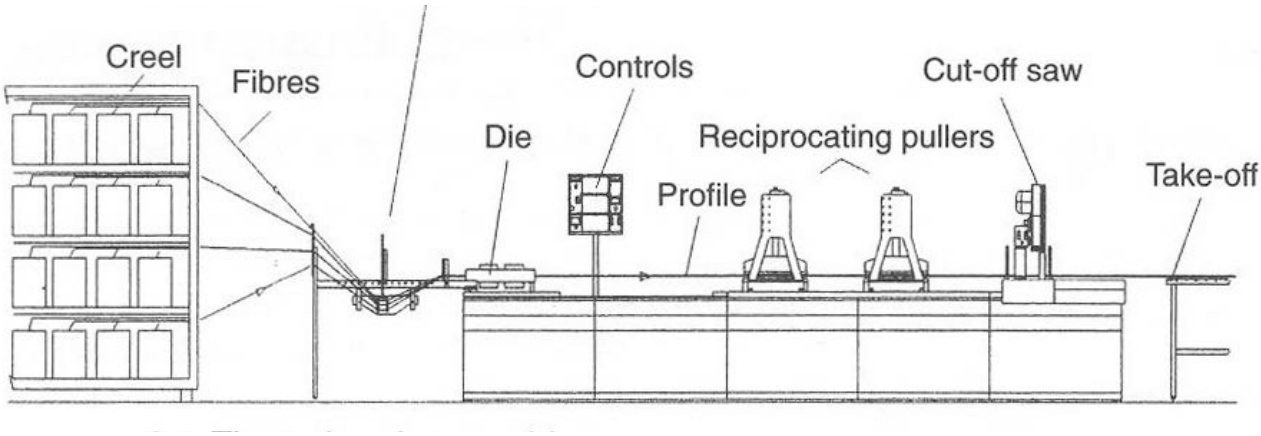
  Disadvantages
  - Mould is generally expensive

- Pultrusion
  Pultrusion is the only truly continuous process in making composite elements.
  Fibers are pulled from rolls through a resin bath, a die with the shape of the required cross-section and hardened in a heated part of the die. For complex shapes the resin can also be injected in a cavity of the die. If the section is hollow, the use of mandrels allows virtually any cross-section to be made. Because of the nature of the pultrusion process, the fibers are mainly oriented in the longitudinal direction, though the use of fiber fabrics or fiber mats is possible.

  Advantages
  - High quality, controlled, continuous process
  - Accurate products
  - Suitable for different fibers and resins, or combinations of them
  - Fiber content/reinforcement ratio up to 70%

  Disadvantages
  - Longitudinal direction has the best mechanical properties/Limited orientation
  - Wear and tear of die results in slight differences in products
  - Larger thicknesses require longer hardening; lower production speed or longer mould

- Filament winding
  In this process continuous fibers are spun around some form of cylindrical mandrel. This mandrel is rotated along its longitudinal axis while the fibers are injected with resin. The angle of spinning is variable so that the strength and stiffness are adjustable, but the tangential direction will have better mechanical properties than longitudinal.
  The general application for this technique is in the production of both small and large diameter pipes. The mandrels are usually made of steel or a composite material. When the product is finished the mandrel is removed. This is possible due to clever use of



inflatable, sectional, soluble or otherwise removable mandrels.

Advantages
- Orientation of fibers is variable
- Large one-piece hollow section possible
- Combinations of fibers resins possible

Disadvantages
- Cross-section is rotationally symmetrical
- Fibers not possible in longitudinal direction
- Open process; styrene emissions

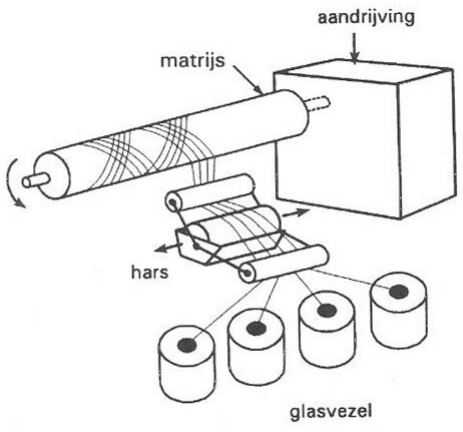
- Autoclave

  An autoclave is a sealed, pressurized device designed to heat solutions to temperatures above their boiling point.

  A one-sided mould with a vacuum foil is placed inside the autoclave. The mould is pressure-heated to the required temperature as the fiber reinforced polymer hardens. Typically a thermoset resin such as epoxy is used in the form of prepregs, or blanks. Through this process very high fiber contents/reinforcement ratios and low void contents can be achieved.

  Advantages
  - Good mechanical and thermal properties
  - Hardening process controlled; good quality in low numbers
  - High fiber content/reinforcement ratio and low void content
  - Fibers can be freely oriented
  - Clean process, low waste
  - Dimensional accuracy
  - Complex shapes possible e.g. ribs, sandwiches, small thicknesses

  Disadvantages
  - High labor content with highly skilled personnel
  - High pressure and temperature require relatively expensive moulds and energy consumption
  - Long cycle; not suitable for large series
  - Prepregs are expensive and require cold storage



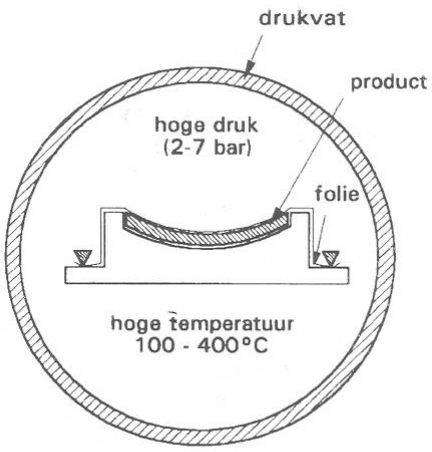


Figure B.27-28
Filament winding and an autoclave.

Table B.8 *range of fiber content (%)*

	Mat	Woven	UD
Spray deposition	10-20	N/A	N/A
Contact Moulding	10-20	25-40	40-50
Compression moulding	20-30	40-50	50-60
Filamentwikkelen	N/A	N/A	50-70
Prepreggen	N/A	40-55	50-70
Pultrusion	20-30	40-55	50-70
Total	10-30	25-55	40-70

B.4.10 Machining techniques

Though there is some dependance on the direction because of the laminas, fibre reinforced polymers can easily be machined after production. Drilling and milling are possible. Sections can be sawn to smaller lengths and sheets can be cut using water jetting. If a tougher fiber is used, such as aramid instead of glass, then machining possibilities are somewhat more limited.

B.5 Rapid manufacturing (RM)

This paragraph concerning some of the most modern manufacturing processes has been compiled using Degarmo et al. (2003), Hopkinson et al. (2006) and Wright (2001).

Since the late eighties, a new breed of manufacturing processes has been developed that combines the power of computers and robotics to bridge, or actually close the gap between design and manufacturing.
Rapid prototyping (RP) refers to modern methods of producing first physical models of a new design, a prototype. The term RP encompasses the manufacturing of prototypes by combination of computer-aided design (CAD) and computer-aided manufacturing (CAM) and by robotically driven production machinery. These machines are computer numerically controlled (CNC) using onboard microprocessors.The available computer controlled production methods are:

- machining
- casting
- solid freeform fabrication (SFF), or unconstrained manufacturing

The advantages of RP are:
- High dimensional accuracy
- Freeform geometries
- Reduced labor
- Reduced production development time

The disadvantages are:
- Capital intensive machinery and hardware
- High energy consumption
- Skilled technicians and users needed
- Inaccuracy in vertical, or z-axis, due to 'stairstepping', inherent to the layer-based RP methods.

The next step after RP is rapid manufacturing (RM). Hopkinson et al. (2006) define RM as *'the use of a computer-aided design (CAD) based automated additive manufacturing process to construct parts that are used directly as finished products or components'*. RM is a new field, still in its infancy, and generally directs its attention to enhancing RP SFF processes to achieve the definition stated above.

Degarmo et al. (2003) identify four categories of SFF processes while Hopkinson et al. (2006) use three categories, which are also mentioned:
- Photopolymer-based (or liquid-based)
- Deposition-based (or solid-based)
- Powder-based (idem)
- Lamination-based (also solid-based)

Before explaining these categories in more detail it is noted that the nature of these methods is additive i.e. material is added to build up the prototype. This sharply contrast conventional and RP machining techniques which generally involve subtracting material.
The reader is referred to Hopkinson et al. (2006) for more references to specific academic and commercialized examples of the SFF processes introduced here.

B.5.1 Photopolymer-based processes

These processes use UV radiation to solidify photocurable liquids, or photopolymer resins.

- Scanned laser polymerization (SLP)
  uses a helium-cadmium (HeCd) laser that selectively cures the resin. This is repeatedly done in layers, by curing successive cross sections on a liquid resin surface. Stereolithography (SLA) is a form of SLP and the first example of a commercially available SFF process when it was launched in 1987. One drawback of SLP is the need for partially supporting (e.g. in the case of cantilevers) the model during fabrication, since the resin is unable to sufficiently carry the solidified material.
- Solid ground curing (SGC)
  is similar to SLP, but uses a photomask and a high-intensity UV lamp. The photomask is an electrostatically charged glass that selectively attracts a powder, or toner, to form the negative shape of the required layer. The UV lamp then exposes the entire resin surface to radiation. This is repeated until the model is finished. The model can be quite large, since large SGC equipment exists.

B.5.2 Deposition-based processes

In these type of processes material is physically deposited to form the model.

- Fused deposition modelling (FDM)
  or extruded deposition (ED)
  uses robotically guided extruders to form laminated three-dimensional objects. Wax, nylon, ABS, polyester and even metal compounds can be used. In the case of thermoplastics, the extruders are supplied by filament from a spool and have a heated nozzle to deposit the polymers.
- Inkjet deposition (ID)
  uses an inkjet mechanism to selectively deposit uniformly spaced, thermoplastic or wax micro-droplets. The droplets adhere to each other to produce the model.
- Contour crafting (CC)
  This is the only RP or RM method found with practical applications in the building industry. See Section B.1.6.

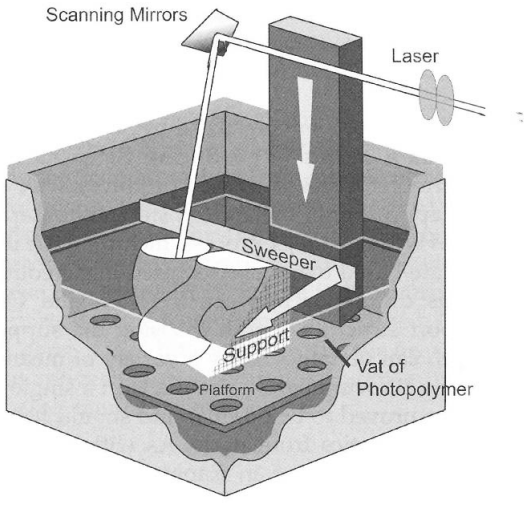


Figure B.29
This schematic of stereolithography (SLA) illustrates how the liquid photopolymer in the bath is selectively cured by a laser, as the solid model is slowly lowered, one layer at a time.
Source: Hopkinson et al. (2006)

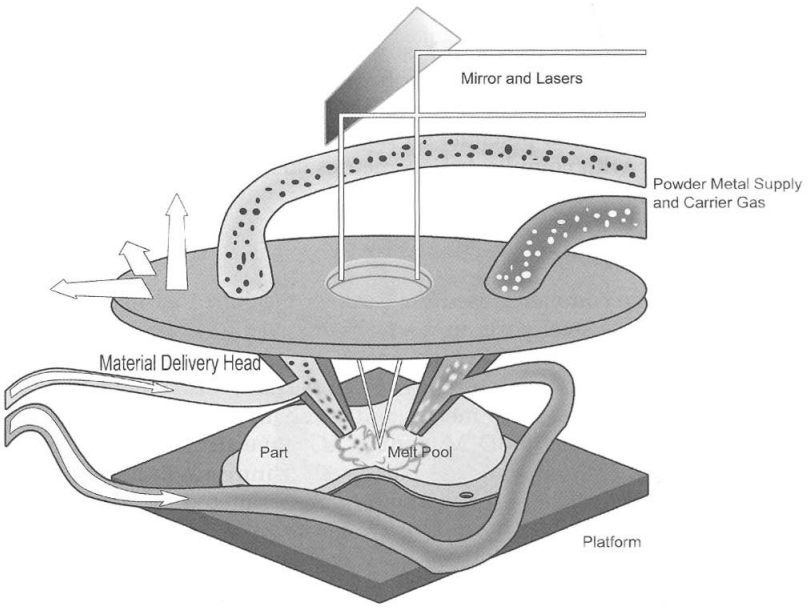


Figure B.30
Fused deposition modelling (FDM) directly deposits a metal powder combined with a carrier gas, which is then thermally bonded, or fused, to the rest of the solid model by lasers.
Source: Hopkinson et al. (2006)

B.5.3 Powder-based processes

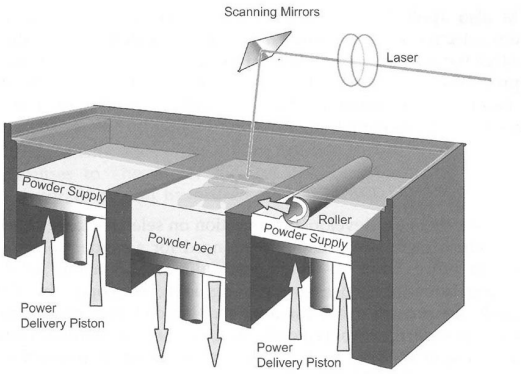


Figure B.31 This schematic of selective laser sintering (SLS) shows the process in three consecutive steps. Layer by layer, the powder compound is cured by laser. *Source: Hopkinson et al. (2006)*

The term 3D-printing definitely refers to powder-based processes as well and is mentioned often, but definitions vary and it seems 3D-printing is both used synonymously with powder-based processes in general as with specific process types similar to or exactly like the following examples.

- Selective laser sintering (SLS) or fusing and sintering (SLFS)
  uses CO_2 lasers to melt, or fuse powder particles in a build chamber. Successive layers of powder are deposited and selectively sintered until the model is complete. Afterwards, the supporting, unsintered powder in the chamber is removed.
  SLFS processes can use a wide range of materials, including plastics, wax, metals and ceramics. SLS processes rely on various polymeric materials, including nylons, polyamides, polycarbonates, elastomers and acrylic styrene.
- Selective inkjet binding (SIB)
  works by depositing a binder material (e.g. colloidal silica) into a layer of powder. The binder joins the powder material and no thermal mechanism is needed (such as lasers). A new layer of powder is applied and the process is repeated. SIB can be used for a 'variety of engineering materials' and ceramic powders.

B.5.4 Lamination-based processes

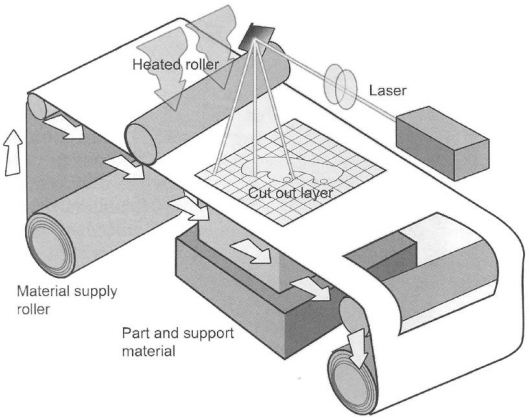


Figure B.32 Laminated object modelling (LOM) bonds material by layer, and instead of using laser to cure the material, it is used to cut the outline of each consecutive cross section.

In lamination-based processes, the model is built up from stacked sheets of material. In some of these processes, no thermal treatment is needed and therefor, unlike the other methods, the material does not go through a phase change. This of course leads to reduced energy requirements.

- Laminated Object Modelling (LOM).
  Material sheets or laminae are sequentially bonded together and patterned. Sheets of paper, or less commonly, polyester, metal or ceramic tape, are glued or heat-rolled to bond with the layer below it. After bonding, a laser or cutter is guided to cut the required cross section. When the model is finished the excess unbonded sheet material is removed. Some trimming, hand finishing and curing are needed afterwards.

B.5.5 Comparison of various RP processes

Wright (2001) maintains that SLA is the most accurate and commercially attractive SFF process, while FDM is more suitable for constructing single prototypes made of metal or structural plastic.
He concludes that not SFF, but machining and casting will remain central to RP, especially for high-strength prototypes and longer batch runs of several prototypes.

B.5.6 Applications of RM in the building industry

All the referenced literature on RP and RM show that most of their applications are found in the automotive, aeronautics and medical industries, where complex, high precision parts are already designed and manufactured with SFF.

However, in Hopkinson et al. (2006) R. Soar devotes a whole, somewhat chaoticly written chapter to applications for the construction industry and emphasizes that the inherent complexity of buildings makes it especially suitable for RM. The various materials and functions, and consequent geometric complexity of buildings and its elements could lead to advantages when comparing RM to conventional building methods.

There are two mechanisms that Soar believes will drive a move towards RM in the building industry.
The first is the emergence of freeform design – contrasting the current practice of modularisation – which requires innovative manufacturing techniques.
The second is the emergence of structural optimization, which leads to typically organic and complex geometries.

> *'Additive manufacturing technologies potentially offer the first approach by which optimized structures can be derived, within a CAD system, and be reproduced faithfully into a physical structure or component.'*
>
> — *R. Soar*

This source also reaffirms that practical applications are, as of yet, limited to CC (see Section B.1.6) as it is the only example given outside various undergoing research efforts.

Mold milling and hot wire cutting (see Section B.1.6) is mentioned, because it also enables the production accurate concrete elements through CAD-software and CNC machines. However, this is not technically RM, because only a mold is produced and not the final product.

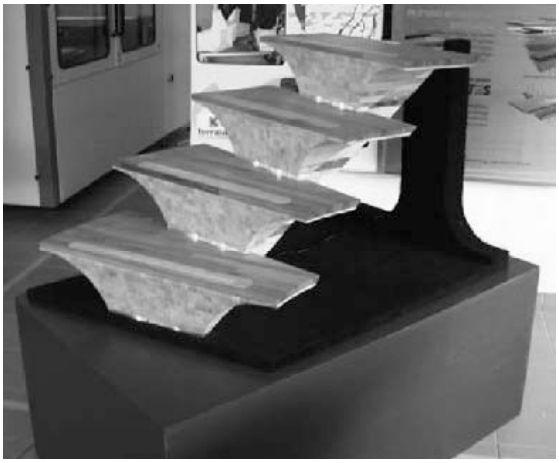


Figure B.33
French company Stratoconception used a lamination-based method to manufacture this staircase made of wood and polymethyl methacrylate (PMMA), also known as acrylic glass.
Source: RM platform (http://www.rm-platform.com)

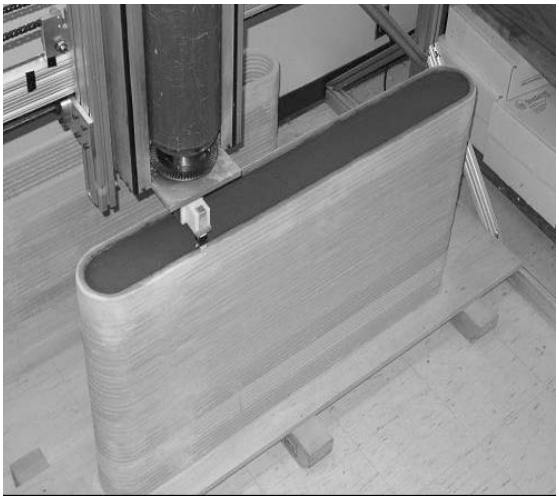


Figure B.34
Contour crafted (CC) hollow wall section which has been filled with concrete afterwards.
Source: http://www.contourcrafting.org

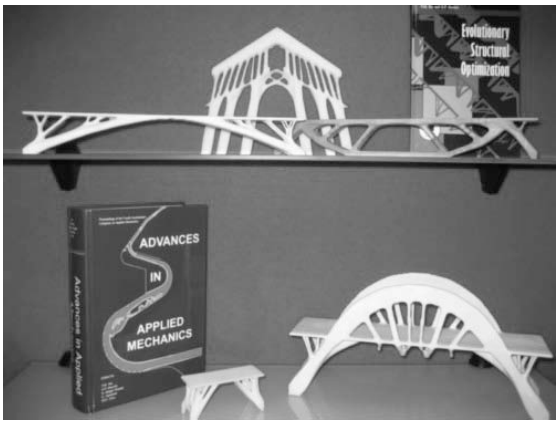


Figure B.35
The developers of ESO (see chapter 3.2.2) used 3D-printers to create these miniature models in wax and plaster.
Source: Innovative Structure Group (http://www.isg.rmit.edu.au)

B.6 Comparison of construction materials

Before making any conclusions, the results of the four paragraphs concerning the four construction materials are summarized and compared.

- Mechanical properties
  The diagrams on the left show the range in compressive and tensile strengths taken from the Tables (B.1 to B.6) presented in the earlier paragraphs. (The steel compressive strengths have been taken equal to the tensile strengths. Literature does not give any values, because in the case of compression, buckling is governing for steel.)
  The compressive strength for concrete is superior, unless uni-directional carbon and aramid fiber reinforced polymers are considered. In that case the transversal compressive strength is significantly lower.
  For tensile strength both steel and composites perform well, though again, carbon and aramid fibers are superior.
  The diagrams on the right side of the left page show the bending stiffness, or E-modulus, in absolute and relative values (divided by the density).
  Concrete and uni-directional composites (again, not for the transversal direction) perform well, while steel has the highest E-modulus. However, relative to the weight, it is visible that, particularly timber is very stiff, while steel is less so. This diagram underlines the validity of structural use of all of these materials.

  Table B.9 compares three other properties.
  - The Poisson's ratio does not vary much.
  - The thermal expansion is quite large perpendicular to grain in wood and transversal to the fiber direction in composites.
  - The thermal conductivity in steel is very high, not surprisingly, as this is the case for most metals.

- Durability
  Structures are designed to last at least several decades to a century. All construction materials that were discussed are used for these structures, so they can all be considered to be durable. It is still noted that timber is arguable the least durable, while concrete is the most durable. (see Table B.10)

- Sustainability
  The term sustainability can be used to describe three types of ecological responsibilities:

Figure B.36
Comparative diagram of material compressive and tensile strengths

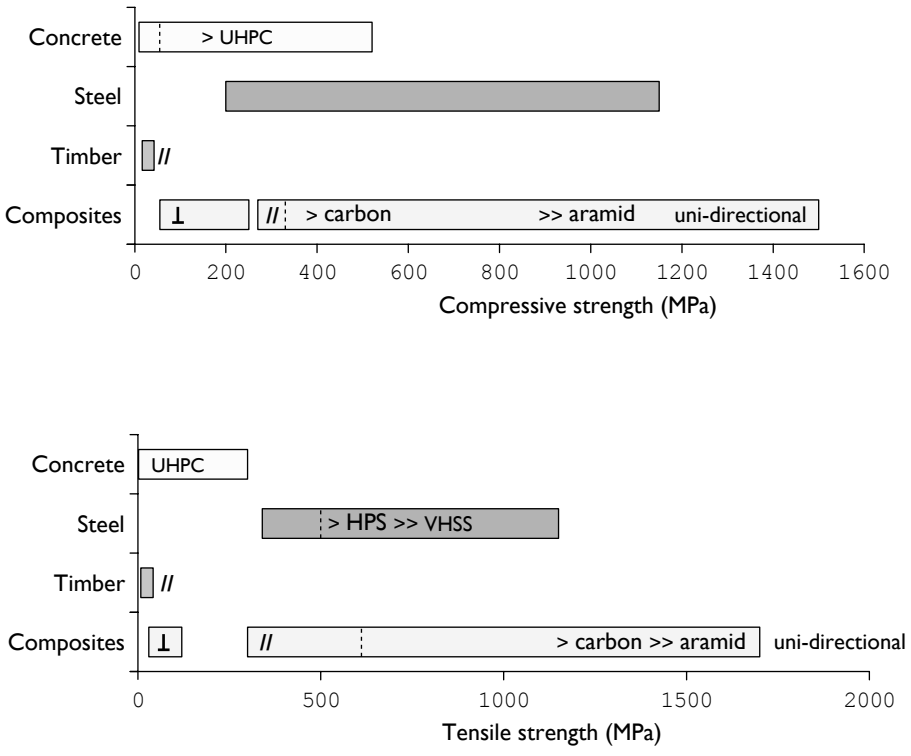


Table B.9		Concrete	Steel	Timber		Composites		
				Parallel to grain	Perpendicular to grain	Longitudinal or parallel to plane	Transversal or perpendicular to plane	
Poisson's ratio	υ	0,1-0,28	0,27-0,30			0,18-0,33		(-)
Thermal expansion	α	9-12	12	3,1-4,5	15,3-45,0	-2-64,1	5,4 -105,8	$10^{-6}K^{-1}$
Thermal conductivity	k	1,8-2,5	52	0,1-1,4		0,20-0,82		W/mK^{-1}

- Effectiveness
  This refers to actual recycling, where the materials can be recycled to a new product of equal quality. Reuse is also effectiveness, because nothing changes and the lifetime of the material is extended.

- Efficiency
  Though some processes are named recycling, they in fact refer to downcycling. The material is reused in a consectutive lifecycle, but there is a certain loss of quality (and value), such that it cannot return to the original state.
  Although the lifetime of the material is extended, it is most definitely finite.

- Energy consumption
  The production processes for the material require a certain amount of energy. This energy can ofcourse be derived from either finite or renewable energy sources.

Table B.10 attempts to qualitatively compare the four materials based on these three aspects. Composites are shown to be the least sustainable.

- Manufacturing processes
  Within the context of this thesis it is important to ascertain which of the manufacturing processes offered by these materials are suitable to combine with computational optimization. The choice of process and material will largely define the constraints within the optimization process.

Assumption: It can be argued that to optimally bridge the gap between computational optimization and manufacturing, as the optimization process will have more constraints, the manufacturing process should have less. This will enable the best, most useful combination of both aspects.

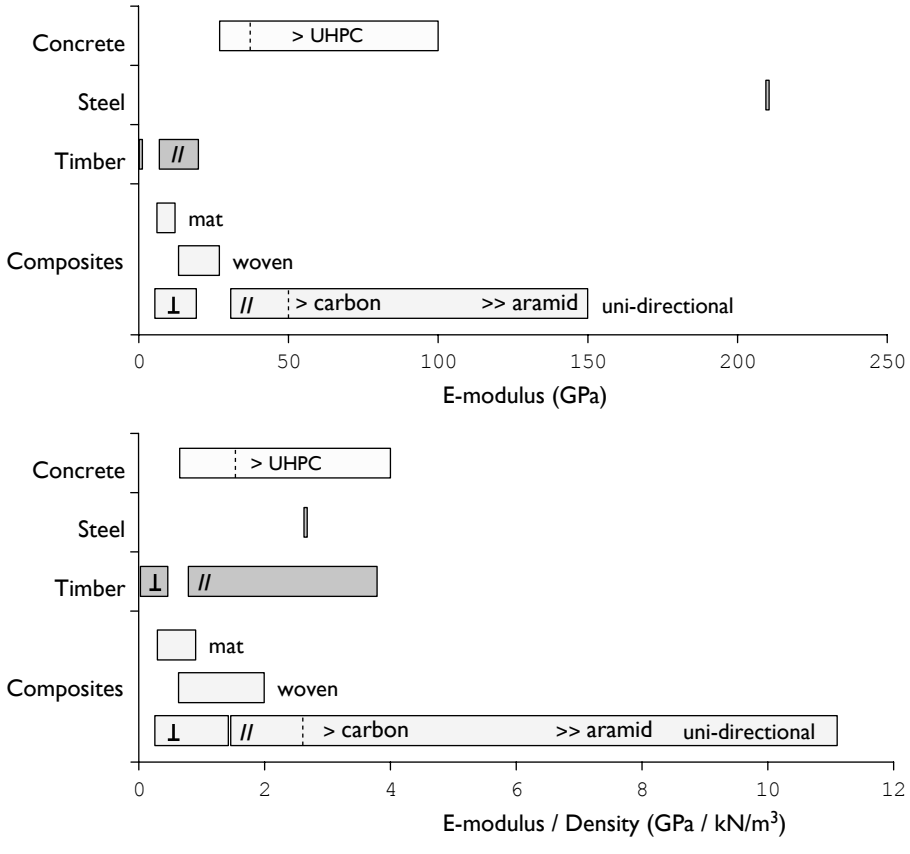


Figure B.37
Comparative diagram of material bending stiffness, both in absolute values and relative to the density.

Following this assumption, the processes that have less geometric constraints, i.e. that allow more or less free form fabrication, are mentioned once more (see also Section B.1.6, B.4.9 and B.5):

Concrete

- Conventional molds
- Mold milling and hot wire cutting
- Contour crafting (CC)
- Fabric formwork

Composites

- Contact moulding
- Resin injection
- Autoclave

and,

- Rapid manufacturing (RM)

Table B.10		Concrete	Steel	Timber	Composites
Durability		*Very good*	*Good*	*Average*	*Good*
	Major causes of degradation	Chemical attack and corrosion of reinforcement	Corrosion	Biological attack	Oxidation and solar radiation
Sustainability		*Average*	*Poor to good*	*Poor to very good*	*Very poor to poor*
	Effectiveness, or recycling	Poor	Poor to very good	Poor to very good	Very poor to poor
		up to 20% in new product	theoretically 100%, pratically 20 - 90% in new product	theoretically 100% bio-degradable, though adiitives prohibit this	perhaps 6 % up to 30% of some constituents but generally downcycled
	Efficiency, or downcycling	Good	Very good	Very good	Good
		Granulated as filler for roads etc.	Molten and diluted to lower grade	Resawn or shredded to particles of fibers	Shredding or particle recycling
	Energy consumption	High	Very high	Average	Very high

B.7 Conclusions

Finally, some conclusions and recommendations are made on manufacturing processes and for this thesis.

On manufacturing processes:

- Most manufacturing processes are limited in such a way that the elements produced have a constant cross section along the longitudinal axis e.g. extrusion, pultrusion.
- Most developments in manufacturing apply to concrete and composites.
- New concrete and several standard composite manufacturing processes offer the most form free fabication.
- Rapid manufacturing (RM) is still in its infancy; expensive and applicable to specific materials, with products limited in size and mechanical properties. This field has hardly been approached from a civil engineering standpoint.

On this thesis:

- Fiber reinforced polymers are not sustainable. With respect to sustainability and current technology, the choice for composites in this thesis is deemed unacceptable.
- Manufacturing processes that have a higher degree of geometric freedom have more potential to combine with computational optimization, These should definitely be considered for this thesis.

APPENDIX C | Dynamic Relaxation for form finding of prestressed membranes

This appendix is largely taken – literally – from Barnes (1999) and describes how, according to that source, Dynamic Relaxation can be adapted to solve membranes modelled as triangular elements. Initially this entire set of calculations was programmed and implemented, but it never functioned properly. In retrospect, the cause of this is thought to lie in the numerical errors inherent to Java's floating-point calculus as mentioned in Section 5.5. For future development of FabricFormer it is still recommended to implement these calculations as they are probably more adept and accurate at modelling fabrics than the currently used cable-net analogy.

C.1 Modelling and definition of weave and stress directions

Membrane surfaces are normally fabricated from panels in which the centrelines and welded seams follow geodesic paths over the surface. The centrelines of the panels define the warp direction of the fabric weave, and in most cases the seams will also be nearly parallel with the warp fibres. Accepting for the present this slight approximation, it is convenient to model the surface with a sufficiently large number of triangular facet elements in which one side of every element is aligned with a "warp control line" (Figure C.1).

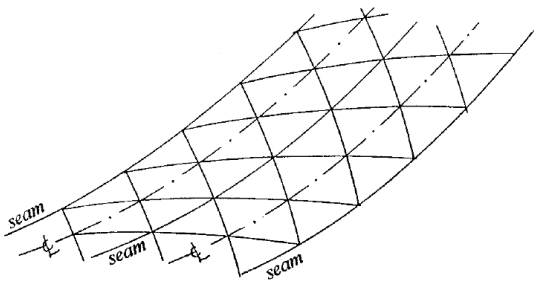


Figure C.1
Triangular facet elements with warp control lines along the seams of the actual fabric.

In form-finding, the prescribed stresses (σ_x warp and σ_y weft), and for subsequent load analyses the stress/strain relations, will be referenced parallel and perpendicular to the warp control lines. These lines (and the element idealization) must therefore be established during the form-finding process. One method of doing this is analogous to a soap film in which cotton threads are floating. If the cotton threads are lightly stretched across the surface they will follow geodesic paths, with the constant soap film tension automatically imposing a balance of stress across the threads. But because it is deformed by the thread tensions the surface will not be a true soap-film or minimum surface. With a numerical model however, this problem can be avoided by the use of warp control strings in which the tensions govern trajectories in the plane of the surface but have no effect normal to the surface. At end nodes, for example on boundary scallops, the effect of the strings must also be entirely discounted. The form of the surface is thus dependent only on the specified stresses in the membrane elements which, although held constant in individual elements, may be graded throughout the entire surface (see Section C.4).

The tension in the warp strings is most conveniently controlled by specifying constant tension coefficients (T/L values) in their component links. The computational sequence of setting the nodal residual forces may be summarised as follows:

(1) For each warp string node, set residuals $\{\mathbf{R}\}$ before any other element type.

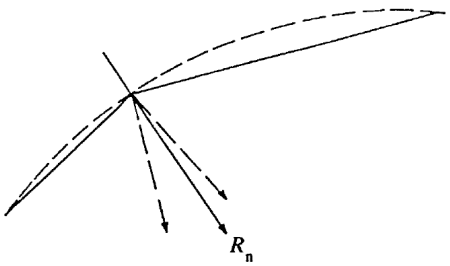


Figure C.2
The surface node normal vector is the mean of the normal vectors of the adjacent triangular membrane elements.

(2) Set surface node normal vectors $\{\mathbf{v}_n\}$ as the mean of the normal vectors of adjacent triangular membrane elements, with a weighting for each component triangle normal which is inversely proportional to its area – see Figure C.2 – (this applies formally when the surface is locally either cylindrical or spherical with triangles surrounding the node having equal base lengths, but otherwise is only an approximation converging when the mesh is sufficiently refined).

(3) For a node normal vector {vn} the normal residual component is $\mathbf{R}_n = \{\mathbf{v}_n\}^T \{\mathbf{R}\}$

(4) For each geodesic (warp string) node, subtract the normal component from the global residuals. Thus, new $\{\mathbf{R}\} = \{\mathbf{R}\} - \mathbf{R}_n\{\mathbf{v}_n\}$

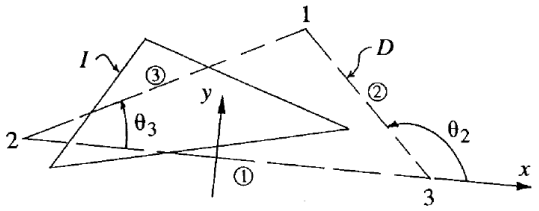
(5) Set residuals of all ridge cable and boundary scallop nodes to zero.

(6) Continue residual summation for the other (real) element types.

C.2 Link Forces in terms of Membrane Stresses

If the triangular element in Figure C.3 is assumed to deform from an initial state **I** to state **D** such that the sides remain straight, the strains induced within the element may be taken as constants. Thus the strain parallel to any side i is Δ_i/l_i where l_i is the length of side i in state **I** and Δ_i is its extension in moving to state **D**.

Figure C.3
Deformation of a triangular element from state I to D.



If side 1 is always parallel with the (moving) x axis the strains relative to x and y axes can be expressed in terms of the side extensions:

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{1}{l_1} & 0 & 0 \\ \frac{a_3c_2 - a_2c_3}{Ql_1} & \frac{c_3}{Ql_2} & \frac{-c_2}{Ql_3} \\ \frac{a_2b_3 - a_3b_2}{Ql_1} & \frac{-b_3}{Ql_2} & \frac{b_2}{Ql_3} \end{bmatrix} = \begin{Bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{Bmatrix} \qquad \text{(C.1)}$$

or: $\{\varepsilon\} = [\boldsymbol{G}]\{\mathrm{D}\}$
where $a_i = \cos^2\theta_i$, $b_i = \sin^2\theta_i$, $c_i = \sin\theta_i \cos\theta_i$
and $Q = (b_2c_3 - b_3c_2)$ with all θ_i in the deformed state.

Equations (C.1) relate the "convected" strains in the three side directions to convenient orthogonal strains, and in taking the axes as moving with the element the rigid body movements are eliminated from the strain terms.
The link attraction forces between nodes (or equivalent side tensions) can be expressed in terms of the internal stress resultants (σ_x, σ_y ,τ_{xy} per unit width) by applying the principle of virtual work. For an infinitessimal virtual deformation $\{\Delta^*\}$ from the deformed stress state the virtual strains are:

$$\{\varepsilon^*\} = [G']\{\Delta^*\}$$

where the terms in $[\boldsymbol{G}']$ are identical with $[\boldsymbol{G}]$ except that the side lengths are also those for the deformed state.
Equating the virtual work of the equivalent link tensions to the virtual work of the stresses:

$$\{\Delta^*\}^T \{\boldsymbol{T}\} = \{\varepsilon^*\}^T \{\sigma\} \boldsymbol{A}$$

where A is the (deformed) area of the element.

Hence the side tensions equivalent to the internal stress resultants are:

$$\begin{Bmatrix} T_1 \\ T_2 \\ T_3 \end{Bmatrix} = A\{G'\} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \tag{C.2}$$

It is computationally convenient to separate out the direct and shear stress components. Thus the link tensions from (C.2) due solely to direct stresses (σ_x and σ_y) are:

$$T_1 = \frac{\sigma_x A}{l_1} + \frac{\sigma_y A}{l_1 Q}\left(a_3 c_2 - a_2 c_3\right)$$
$$T_2 = \frac{\sigma_y A \cdot c_3}{l_2 Q}$$
$$T_3 = -\frac{\sigma_y A \cdot c_2}{l_3 Q}$$

After trigonometric manipulation, but without approximation, the following can be derived:

$$T_1 = \frac{h}{2}\left(\sigma_x - \sigma_y\right) + \frac{\sigma_y l_1}{2\tan\alpha_1}$$
$$T_2 = \frac{\sigma_y A \cdot c_3}{2\tan\alpha_2} \tag{C.3}$$
$$T_3 = -\frac{\sigma_y A \cdot c_2}{2\tan\alpha_3}$$

where the terms in equation (C.3) are as defined in Figure C.4, all for the current deformed state.

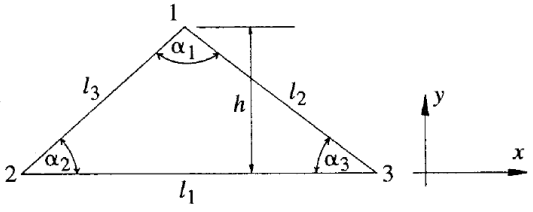


Figure C.4
Parameters of the triangular elements as used in Equation (C.3).

For the particular case of minimum surface systems with uniform stress ($\sigma_x = \sigma_y = \sigma$), equations (C.3) give link tension coefficients:

$$\frac{T_i}{l_i} = \frac{\sigma}{2\tan\alpha_i} \tag{C.4}$$

C.3 Generalization for Non-standard Elements

The foregoing has assumed that side 1 of all elements is parallel with the warp stress / fibres. If this is not so then the nearest element which does comply with this condition can be used to provide the reference x axis for warp stress. In Figure C.5 for example, the elements a adjoining edge scallops are non-standard and element b can be used as their reference.

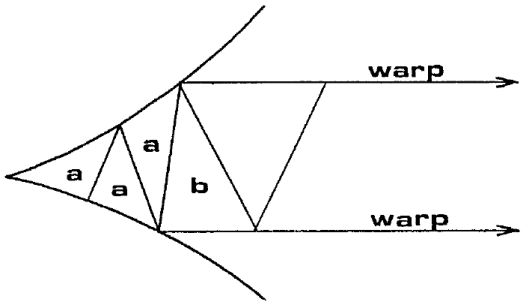


Figure C.5
Non-standard elements a can use element b as a reference to define the warp direction.

Referring to Figure C.6a, the perpendicular distances (h_1, h_2, h_3) from the nodes of a non-standard triangle to side 1 of its reference triangle define the x axis.

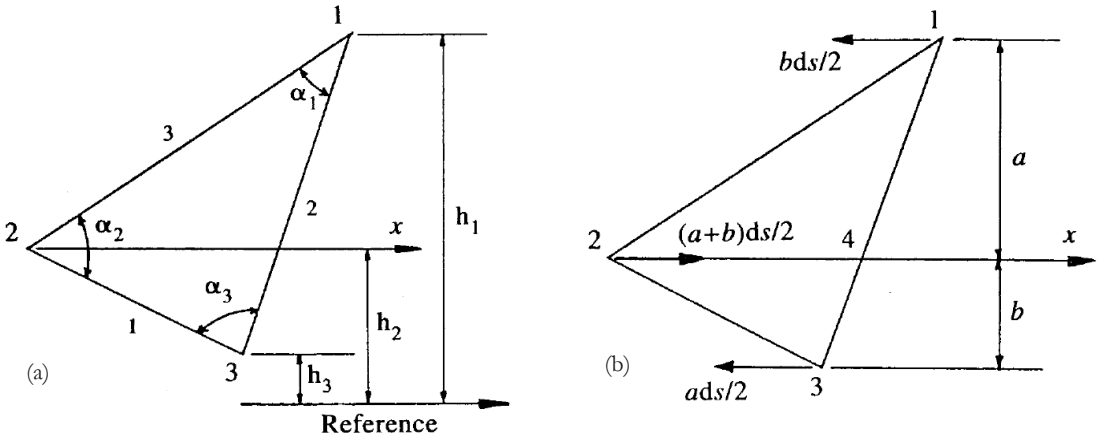


Figure C.6
Parameters as used in Equation (C.5) to calculate a non-standard element using a reference axis.

The tension coefficient along any side i due solely to a stress s y in all directions is given by equation (C.4). To account for the stress variation ds = $(\sigma_x - \sigma_y)$ in direction x only, the triangle can be considered as two component elements with an imaginary link joining node 2 to the intercept point 4 of the x axis with side 2 (Figure C.6b). The tension in this link due to the stress variation ds is thus $(a + b).ds/2$. The whole of this force is applied at node 2, but the reaction at point 4 is apportioned as static equivalents at nodes 1 and 3. When these additional nodal forces are resolved into components along the main element sides, and the uniform stress components are superimposed, the following expression for tension coefficients is obtained:

$$\frac{T_i}{l_i} = \frac{\sigma}{2\tan\alpha_i} + \frac{\left(\sigma_x - \sigma_y\right)}{4A}\left(h_i - h_j\right)\left(h_i - h_k\right) \qquad \text{(C.5)}$$

Equation (C.5) is entirely general, and equations (C.3) can now be seen as merely a special case with the reference axis being the element side 1. It is also apparent that the slight approximation referred to in Section C.1 (that seam lines can be assumed parallel with the panel warp fibres) is unnecessary since the control line / reference axis can be taken as the panel centreline.

C.4 Stability, Form Controls and Patterning

During the process of form-finding everything is continuously changing – the overall form, element sizes and shapes, and reference axes (always belonging to warp control lines). However, it is sufficient to reset tension coefficients for membrane elements from equation (C.5) only at kinetic energy peaks. This, together with the simple calculations involved makes convergence rapid.

Concerning numerical stability during formfinding, consider a patch of elements:

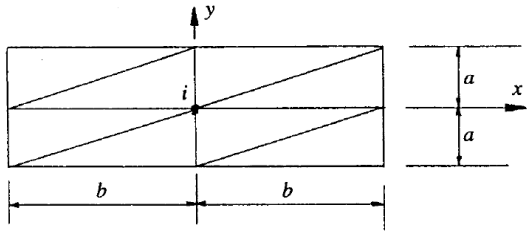


Figure C.7
Example mesh.

The geometric stiffness at node *i* is:

$$\sum \frac{T}{l} = a \cdot \sigma_x \cdot \frac{2}{b} + b \cdot \sigma_y \cdot \frac{2}{a} \leq 2\left(\sigma_x + \sigma_y\right) \cdot \frac{b}{a} \qquad \text{(C.6)}$$

To this must be added the geometric stiffness of the warp control strings which, for every node along the string, is twice the warp string tension coefficient. Additionally, at boundary scallop or ridge cable nodes the elastic stiffness of the cable links must be included in the same way as for cable nets.

The triangle aspect ratios ($b/a > 1$) which partially govern stability will change throughout the process of form-finding. It is therefore necessary to estimate the greatest distortions in membrane element shapes that may occur. This is similar in concept to the use of the factor g in form-finding of geodesic nets which allows for increasing geometric stiffness.

The specified membrane stresses (σ_x warp and σ_y weft) may be varied throughout a surface region provided that, along any warp control string (centre or seam of fabric panel), the stresses match in adjacent elements either side of the control string. This condition places tight constraints on the way in which stresses should be allowed to vary, and frequently σ_x and σ_y will be set at constant values throughout a surface region. One of various exceptions to this is the case of smooth conical surfaces in which it may be essential that warp prestress is increased towards the top ring support. Because the radius of curvature in the hoop (weft) direction decreases towards the top ring (Figure C.8), the warp stress must be increased

otherwise the conic surface will collapse in on itself. Grading σ_x to increase towards the top ring will avoid this and provide a better performance in terms of maximum stresses induced under snow or inward pressure loadings. The simplest way of grading stresses automatically is to use constant tension slip strings along the warp control lines. The additional stress (increasing σ_x) will then vary inversely with the spacing between strings, with the stress σ_y constant throughout the region.

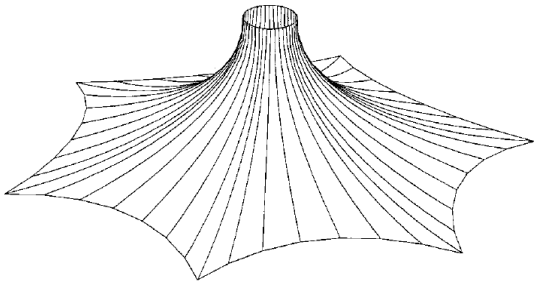


Figure C.8
Example tension structure with hoop ring at the top.

Following the cycled processes of form finding, load analysis checks, and readjustment of the form, the membrane surface must be patterned for fabrication. One approach to this is to automatically (or interactively) adjust the maximum spacings between warp control lines during form finding so that they comply with the allowable width of fabric to be used. The

triangles between adjacent warp lines can then be successively folded out onto a plane to give the cutting pattern shape. An alternative which gives better modelling of the transverse (weft) curvatures, particularly in sharply curved regions, is to use additional warp lines within each panel during form finding, and subsequently to pattern the surface by interpolating seam lines between the warp lines to achieve best use of the available fabric width. This procedure allows more accurately for the curved length across each panel, but since the panels will then consist of several bands of triangular elements along their lengths, they cannot be folded out onto a plane without imposing shearing strains in the elements. However, because the shear modulus of coated fabrics is so low the patterns obtained using the curved widths for fabrication offsets will be more accurate than using a coarse mesh which can be folded. After developing the panels onto a plane, the panels must be compensated to allow for fabrication from an unstressed state to the prestress condition of the assembled and stressed out structure. The stretch compensation values, which depend on the magnitude and ratios of the warp and weft stresses, are obtained from bi-axial testing of samples of the material batch. These bi-axial tests must allow for prestress and creep due to the effects of temperature and the cycled effects of in-service loadings. The compensations will nearly always be greatest in the weft direction (typically 3 – 5%) because this is the direction in which the fibres are crimped over the comparatively straight warp fibres. In the warp direction the compensations will be much lower (< 0.5 %), and may in fact be negative (implying cutting overlength) because of crimp interchange between the weft and warp fibres after stressing. At the ends of panels where the membrane joins stiffer elements, such as cable scallops, reduced compensations must be applied in the weft direction because of the strain incompatibility. This decompensation is graded so that full compensation is achieved at a sufficient distance into the panel equal to about one panel width.

C.5 Load Analysis of Prestressed Membranes

In form-finding of membranes shear stresses are never imposed, which is the reason for the form of Equations (C.3) and (C.5). However, they will be induced by applied loadings. Although the shear stiffness of coated fabrics is so low that the principal stresses will always be in the weave directions, shear terms must be included in order to prevent distortion of the element mesh. From Equation (C.1) the element side tensions due solely to shear stress τ are:

$$\begin{aligned} T_1 &= \frac{\tau A}{l_1 Q}\left(a_2 b_3 - a_3 b_2\right) \\ T_2 &= -\frac{\tau A \cdot b_3}{l_2 Q} \\ T_3 &= \frac{\tau A \cdot b_2}{l_3 Q} \end{aligned} \qquad \text{(C.6)}$$

After manipulation these reduce to:

$$\begin{aligned} T_1 &= \frac{\tau l_1}{2} - \frac{\tau h}{\tan \alpha_2} \\ T_2 &= -\frac{\tau l_2}{2} \\ T_3 &= \frac{\tau l_3}{2} \end{aligned} \qquad \text{(C.6)}$$

where l_i, h, α_2 are as defined in Figure C.4 (for the deformed state).

The stresses can be related to strains from the prestress state. If l_1^i and h^i are the length of side 1 and perpendicular from node 1 to side 1 in the prestress state, the strains are:

$$\varepsilon_x = \frac{l_1}{l_1^i} - 1$$
$$\varepsilon_y = \frac{h}{h^i} - 1 \qquad \text{(C.6)}$$
$$\gamma = \frac{1}{\tan\alpha_2} - \frac{1}{\tan\alpha_2^i}\left(\frac{1+\varepsilon_x}{1+\varepsilon_y}\right)$$

where α^i_2 is the angle at node 2 in the prestress state (and α_2, α^i_2 not restricted to less than 90°)

For a "non-standard" triangle the shear term is the same, but the direct strains must be related to the element distortions parallel and perpendicular to its reference axis (2-4 in Figure C.6b).
The shear stress (from the prestress state) is of secondary importance and can be taken simply as: $\tau = G \cdot \gamma$ where, since τ is a stress resultant, G is in units of kN/m (typically 1/20 of the lowest direct modulus).
The direct stress resultants are complicated by the crimp interchange effects between the weft and warp fibres and by on/off buckling or slackening in one or both of the fibre directions. To account properly for these highly non-linear effects it is necessary to consider a mechanical model of the fabric behaviour and to iterate within each element at each time step to obtain the stresses. A much simpler alternative is to use semi-orthotropic relations of the following form:

$$\sigma_x' = \sigma_x^i + E_x \cdot \varepsilon_x \;;\; \sigma_y' = \sigma_y^i + E_y \cdot \varepsilon_y$$
$$\sigma_x = \sigma_x' + E_c \cdot \varepsilon_y \;;\; \sigma_y = \sigma_y' + E_c \cdot \varepsilon_x \qquad \text{(C.6)}$$

subject to the following conditions:

$$\text{if } (\sigma_x + \sigma_y) < 0 \quad \rightarrow \; \sigma_x = \sigma_x' \;;\; \sigma_y = \sigma_y'$$
$$\text{if } (\sigma_x < 0) \quad \rightarrow \; \sigma_x = 0 \qquad \text{(C.6)}$$
$$\text{if } (\sigma_y < 0) \quad \rightarrow \; \sigma_y = 0$$

In the foregoing, E_x and E_y are the warp and weft stiffness moduli (kN/m width), and E_c is the cross stiffness associated with crimp interchange effects. These properties have to be determined from bi-axial tests of the fabric with warp and weft strains varied from the prestress state. The conditions applied to account for slackening (or wrinkling) of the membrane also assume that, because of the very low shear stiffness, the principal stress directions always align mechanically with the warp and weft fibres; wrinkling due to $\tau^2 > \sigma_x \cdot \sigma_y$ is not accounted for.
It is of course feasible to use many different types of relations which curve fit the test results, including for example polynomial functions in ε_x and ε_y, since the conditions of equilibrium and compatibility are decoupled in the analysis until convergence. But it can be found difficult to curve fit for a variety of stress states and yet obtain convergence with the best fit nonlinear properties. The simpler properties given above, however, do yield convergence and solutions can be bracketed by choosing a range of values for the elastic moduli depending on the load conditions.
Concerning numerical stability the mass components can be set according to equation (11) with stiffness values similar to equation (19) but $(E_x + \sigma_x^i)$ in place of σ_x and $(E_x + \sigma_y^i)$ for σ_y. However, since for load analyses the triangle aspect ratios are known and do not radically change (as they do in form-finding), the nodal stiffnesses can be assembled more accurately element by element.
The foregoing description of Dynamic Relaxation and its computational aspects for stressed membranes has been restricted to simplex elements. The reason for using these elements should be emphasised: that because natural stiffnesses can be described independently of rigid body movements, gross deformations are automatically accounted for. Indeed, no

stiffness matrices are set; the entire formulation being in terms of the "link" forces between nodes given by expressions C.2 (or C.4) and C.6. In this sense the procedure outlined is more akin to a finite difference approach. The same techniques cannot be used with higher order elements, such as curved isoparametric elements, because rigid body movements need to be incorporated in the formulation and transformation of element stiffnesses. This would undermine the basis of the method, and to gain greater accuracy it is more efficient to use a larger number of simplex elements or super-elements.

APPENDIX D | Natural Logarithm Method: A novel strategy for fabric form finding

The following method has been developed by the author and did not previously exist. It was based on an idea by a member of the graduation committee, ir. J.L. Coenders. He proposed exploring a mathematical analogy between so-called well-formulas and membrane geometry.

Well equations use the natural logarithmic function, or *ln*-function, to describe the shape of the piezometric surface, or groundwater table, due to groundwater flow near wells.

An interesting aspect of this equation is that the principle of superposition may be applied to describe the groundwater table in a three-dimensional area, as caused by any combination of wells, impermeable, or semi-permeable layers of soil, vertical impermeable layers and large bodies of water. However, these equations are typically used to describe the interaction between two such objects in a 2D cross-section to retain a simple mathematical model. Because we will show that, to suit the needs of this thesis, the analogy becomes rather tenuous – well formulas are already a rough approximate using natural logarithms – the term Natural Logarithm Method was coined to refer to this novel strategy, rather than some reference to groundwater tables or wells.

D.1 Introducing well formulas

Well equations were derived by Dupuit in 1863 and expanded on by Thiem in 1906 to describe groundwater tables around wells in a static situation. The Dupuit-Thiem equation gives the radial flow to a well through a confined aquifer with transmissivity *KD*:

$$h_2 - h_1 = \frac{Q_0}{2\pi KD} \ln\left(\frac{r_2}{r_1}\right) \tag{D.1}$$

Where h_1 and h_2 is the hydraulic head at points 1 and 2, at a distance of r_1 and r_2 from the well. Q_0 is the groundwater discharge from this point to the wells.
For the hydraulic head at point *M*, the equation uses the principle of superposition so that:

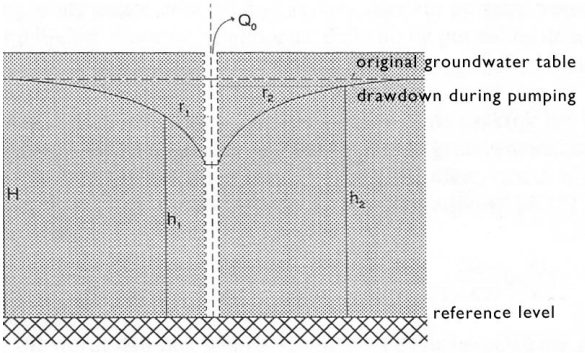


Figure D.1
A well and groundwater table indicating the variables used in the Dupuit-Thiem equation, Eqn. D.1.

$$h_M = \frac{Q_1}{2\pi KD}\ln r_1 + \frac{Q_2}{2\pi KD}\ln r_2 + C \tag{D.2}$$

Where r_1 and r_2 refer to the distances from point M to wells 1 and 2.
To include other effects on the groundwater table such as impermeable layers and large bodies of water, additional, but fictional wells are added or subtracted as illustrated in the images on this page.

D.2 Making the analogy

To understand how an analogy might be possible, the similarities are pointed out that exist between images of the fabric formed beam and that of the *ln*-function in three-dimensional space. The description of groundwater tables might lend itself to that of membrane geometry as well. Each 'hole' in the beam corresponds to a well, while the edges of the formwork correspond to impermeable layers or bodies of water.

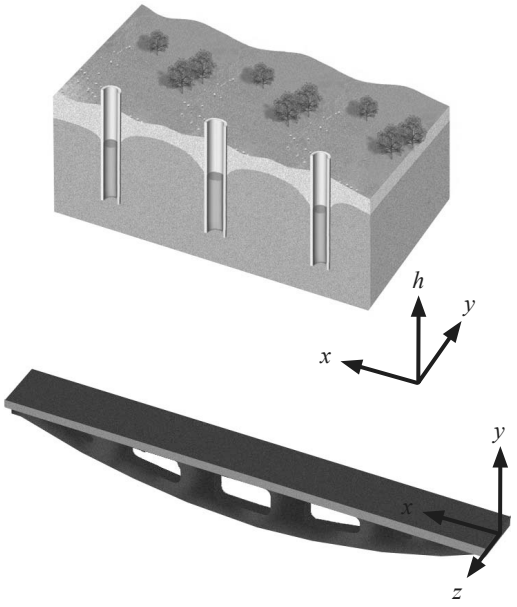


Figure D.2
The analogy between the groundwater table under influence of wells and the fabric formed beam including their typical coordinate-systems.

Exploration of the analogy starts with establishing corresponding coördinate systems. Comparing the two systems in Figure D.2. leads to the following changes in Equation D.2.

$$h = z$$

and

$$r(x,y) = x,\ y$$

It is acknowledged that the coëfficients $Q_i/2\pi KD$ are still necessary, but no longer bear a physical semblance to the problem at hand, so formula (D.2) is rewritten as:

$$z = \sum_{i=1}^{n}\left\{C_i \ln\left(\sqrt{(x-x_i)^2 + (y-y_i)^2}\right)\right\} + C \tag{D.3}$$

Where n is the number of wells in the domain. For one well, situated at the origin (0,0), the shape of function looks like Figure (D.3).

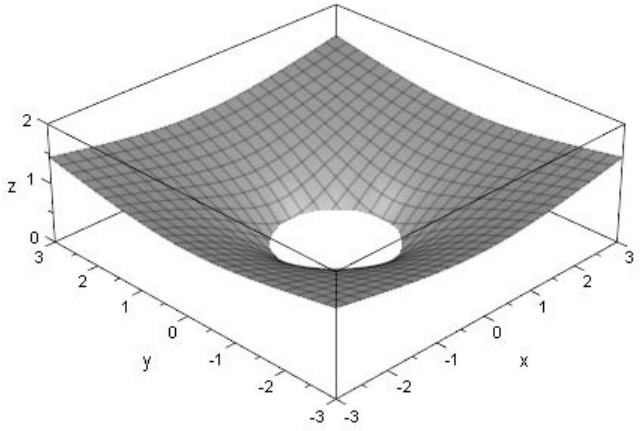


Figure D.3
Equation D.3 plotted in 3D-space in MUPAD PRO for positive z-space shows a single circular 'hole' in the function.

D.3 Creating variable hole geometries in the membrane

Due to the fact that quadrangular shapes are more likely to be used for manufacturable fabric forms, the choice is made to abandon circular shapes and to rewrite (D.3) as:

$$z = \sum_{i=1}^{n} \left\{ C_i \ln\left(|x - x_i| + |y - y_i| \right) \right\} + \tag{D.4}$$

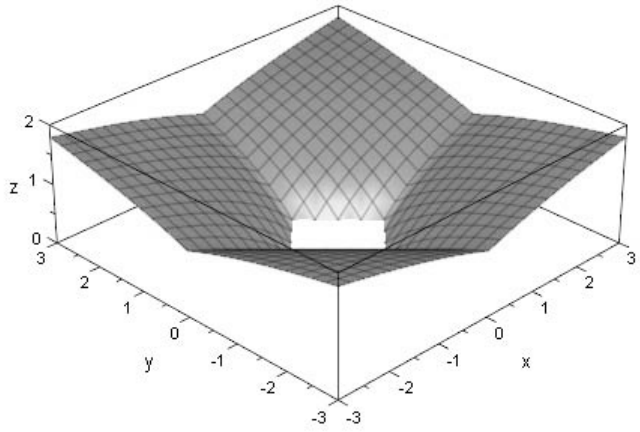


Figure D.4
Equation D.4 plotted in 3D-space in MUPAD PRO for positive z-space shows a single square 'hole' in the function.

In order to create a feasible method of describing fabric formworks, the strategy proposed is the summation (principle of superposition) of several of these ln-function, each describing a hole, or dent in the formwork. To achieve this, it should be possible to transform the rectangular hole in Figure (D.4). to describe each kind of quadrangular hole. The hole might be scaled, rotated, skewed and distorted to create any configuration.
Each of these transformations is discussed, starting with scaling.

Before starting with the various transformation operations, the reader is pointed to parameters x_i and y_i in the previous equations (D.3) and (D.4). These indicate a set of coordinates for each unique well i.

D.3.1 Scaling

Perhaps the easiest transformation, scaling is simply achieved by the introduction of coefficient C_s.

$$z = \sum_{i=1}^{n} \left\{ C_i \ln \left(\frac{|x - x_i| + |y - y_i|}{C_s} \right) \right\} + C \tag{D.4}$$

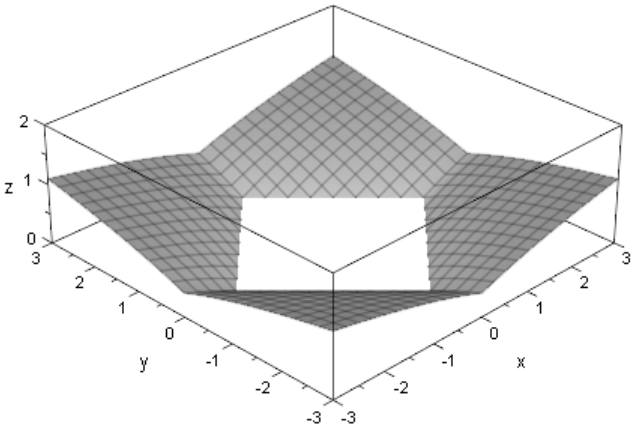


Figure D.5
Equation D.5 plotted in 3D-space in MUPAD PRO for positive z-space shows the effect of coefficient C_s to the scaling of the single square 'hole' in the function of Figure D.4.

Figure D.5 shows how the function behaves when C_s is equal to 2. The size of the cut-off rectangle at $z=0$ is twice as large in width and length.

D.3.2 Rotation

The shape also needs to be rotated over an angle φ_i, so the coordinates are recalculated from the origin of the well using formula:

$$z = \sum_{i=1}^{n} \left\{ C_i \ln |x_2| + |y_2| \right\} + C \tag{D.5}$$

with

$$x_2 = R \cdot \cos\left(\arg(x_1, y_1) - \varphi_i \right)$$

$$y_2 = R \cdot \sin\left(\arg(x_1, y_1) - \varphi_i \right)$$

$$R = \sqrt{x_1^2 - y_1^2}$$

where

$$x_1 = x - x_i$$

$$y_1 = y - y_i$$

Arg, also known as atan2, is a two-argument function that computes the arctangent of y/x given y and x, but with a range [–PI, PI]. Most computer programs offer this function using either of these designations.

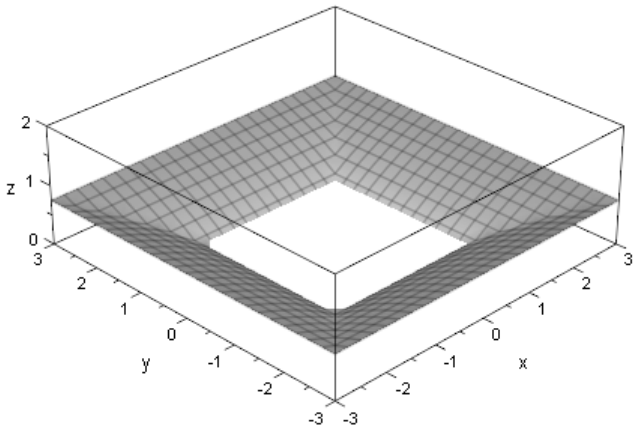


Figure D.6
Equation D.5 plotted in 3D-space in MUPAD PRO for positive z-space shows the effect of rotation φ_1 to the rotation of the single square 'hole' in the function of Figure D.4.

D.3.3 Distortions

Obviously, it is still not possible to create every quadrangle imaginable. Additional transformations should allow the possibility of skewing the shape or making a trapezoid. These transformations are shown in Figure D.7 which also show how the original diamond figure can be divided into four quadrants, each characterized by the signs of $y+x$ and $y-x$.

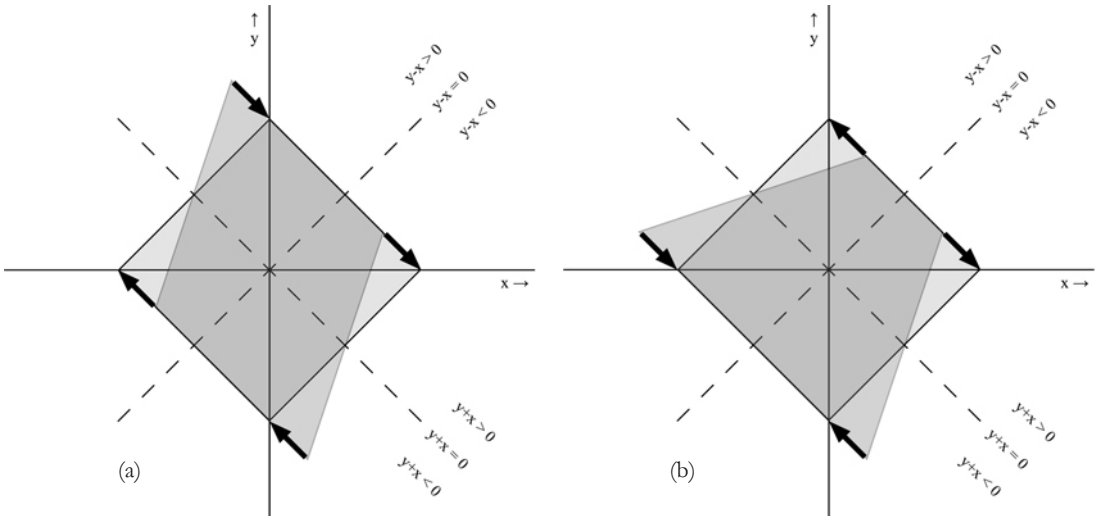


Figure D.7
The operations necessary to create skewed or trapezoidal geometries can be defined depending on the sign of the x and y-directions

To achieve the transformation of Figure D.7a, producing a skewed figure, requires the following changes to the coordinates *(x,y)*.

$$\Delta x = -\frac{1}{2} C_{s1,i} \left(y_1 + x_1 \right)$$
$$\Delta y = y_1 + \frac{1}{2} C_{s1,i} \left(y_1 + x_1 \right) \qquad \text{(D.6)}$$
$$\textit{where}$$
$$x_1 = x - x_i$$
$$y_1 = y - y_i$$

Where $C=[-1..1]$ because larger coefficient result in non-realistic shapes, where width << length, and the skewed shape is no longer dominant over the rectangular shape.

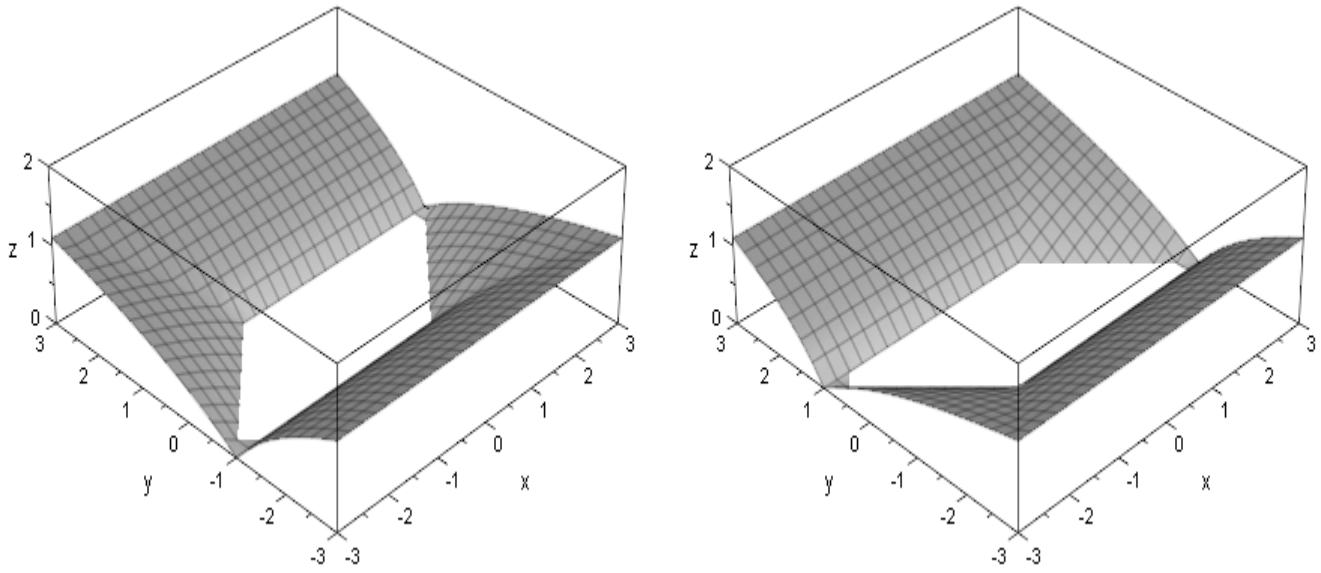


Figure D.8
Equation D.6 and D.7 plotted in 3D-space in MUPAD PRO for positive z-space shows the effect of skewing to the single square 'hole' in the function of Figure D.4.

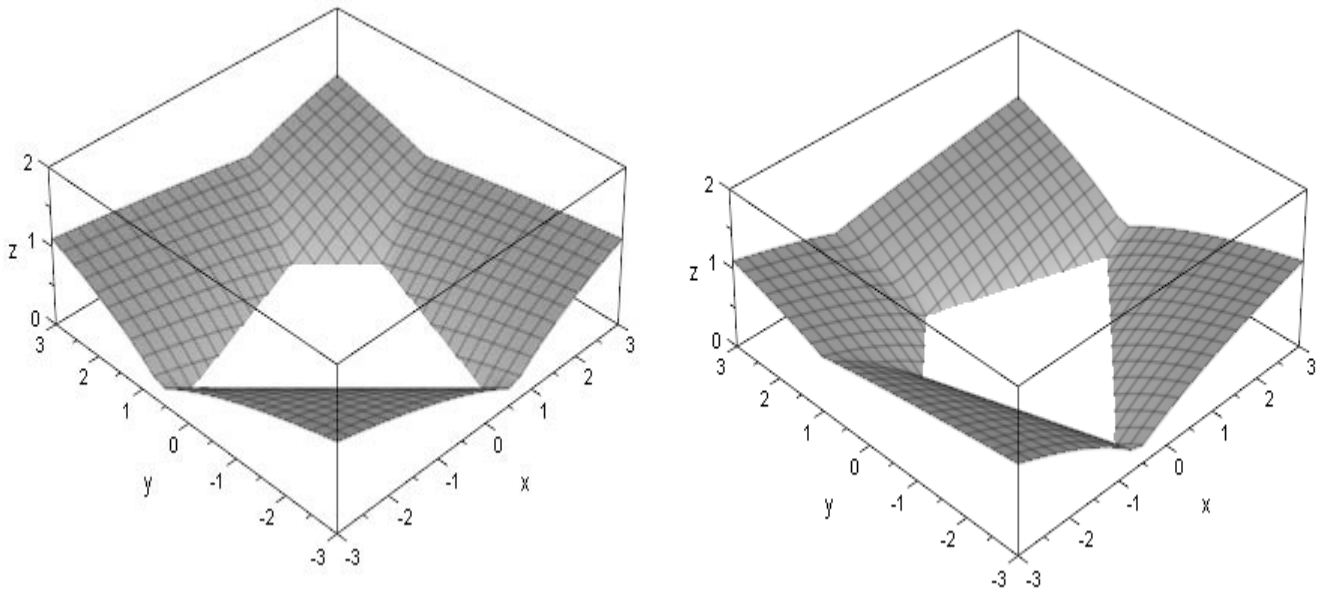
The shape also has to be skewed in the other direction, so additional terms are defined:

$$\Delta x = +\frac{1}{2} C_{s2,i} \left(y_1 - x_1 \right)$$
$$\Delta y = +\frac{1}{2} C_{s2,i} \left(y_1 - x_1 \right) \tag{D.7}$$

A similar strategy is applied to creating trapezoidal shapes as in figure D.7b. For each direction the terms for the coordinate changes are:

Figure D.9
Equation D.8 and D.9 plotted in 3D-space in MUPAD PRO for positive z-space shows how the single square 'hole' in the function of Figure D.4 can be made trapezoidal.

$$\Delta x = -\frac{C_{t1,i}}{C_s} \left(y_1 + x_1 \right) \cdot sign \left(y_1 - x_1 \right)$$
$$\Delta y = +\frac{C_{t1,i}}{C_s} \left(y_1 + x_1 \right) \cdot sign \left(y_1 - x_1 \right) \tag{D.8}$$



And in the other direction:

$$\Delta x = +\frac{C_{t2,i}}{C_s}(y_1 - x_1)\cdot sign(y_1 + x_1)$$
$$\Delta y = +\frac{C_{t2,i}}{C_s}(y_1 - x_1)\cdot sign(y_1 + x_1) \quad \text{(D.9)}$$

D.3.4 Combining all transformations

It is now possible to define this single hole as any quadrangular shape imaginable, by using a single formula which consists of multiple terms and has several driving parameters:

$$z = \sum_{i=1}^{n}\left\{C_i \ln\frac{|x_3| + |y_3|}{C_{s,i}}\right\} + C$$

with

$$x_2 = R\cdot\cos\left(\arg(x_1, y_1) - \varphi_i\right)$$
$$y_2 = R\cdot\sin\left(\arg(x_1, y_1) - \varphi_i\right)$$
$$R = \sqrt{x_1^2 - y_1^2}$$

with

$$x_3 = x_2 - \frac{1}{2}C_{s1,i}(y_2 + x_2) + \frac{1}{2}C_{s2,i}(y_2 - x_2) + \frac{C_{t2,i}}{C_{s,i}}(y_2 - x_2)\cdot sign(y_2 + x_2) - \frac{C_{t1,i}}{C_{s,i}}(y_2 + x_2)\cdot sign(y_2 - x_2)$$

$$y_3 = y_2 + \frac{1}{2}C_{s1,i}(y_2 + x_2) + \frac{1}{2}C_{s2,i}(y_2 - x_2) + \frac{C_{t2,i}}{C_{s,i}}(y_2 - x_2)\cdot sign(y_2 + x_2) + \frac{C_{t1,i}}{C_{s,i}}(y_2 + x_2)\cdot sign(y_2 - x_2)$$

where

$$x_1 = C_x(x - x_i)$$
$$y_1 = C_y(y - y_i)$$

The parameters are:

C_i	for scaling of the total function
C	for translation of the total function in z-direction
(x_i, y_i)	for the location of the center of the hole
C_x	for scaling in the x direction
C_y	for scaling in the y direction
φ_i	for rotation
$C_{s1,i}$ and $C_{s2,i}$	for skewing in both direction
$C_{t1,i}$ and $C_{t2,i}$	for trapezoidal shaping in both directions

The combination of all these transformations for one single hole will lead to shapes such as shown in Figure D.10.

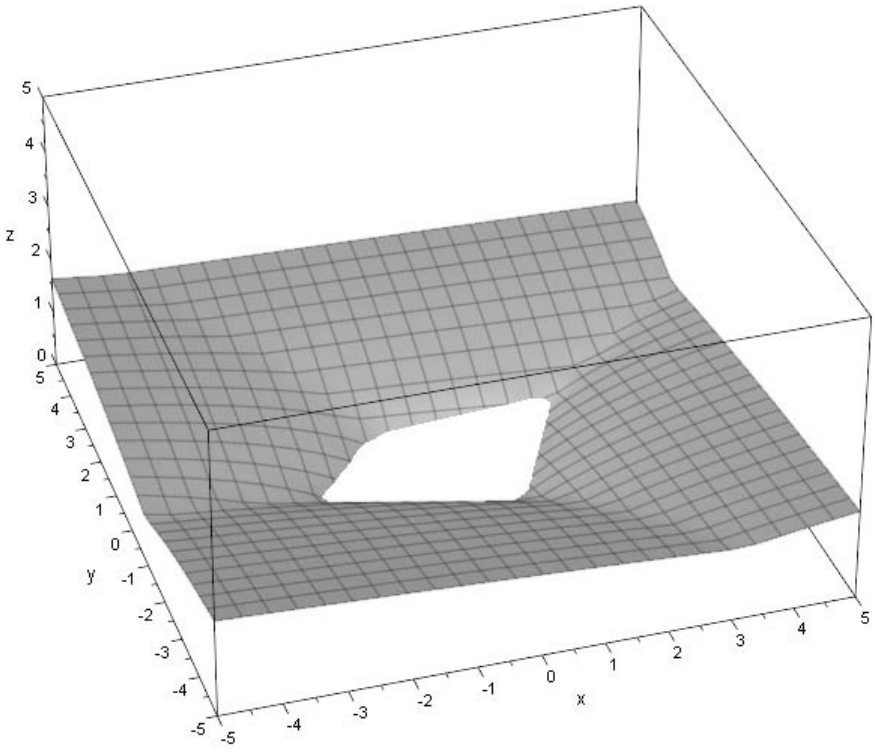


Figure D.10
The combination of all transformations can lead to arbitrary quadrangulars such as this one.

D.4 Applying the NLM to approximate fabric formed geometries

The final remaining part of this method is the definition of the domain for which the function is plotted. This requires that boundaries are defined for x,y and z, for example by defining ranges or functions that bound the graph. This bounded part of the graph is that which represents the geometry of the fabric.

Figure D.11
Photograph of a fabric formed beam cast at C.A.S.T. in Canada.

A test case was devised by iteratively attempting to approximate the shape of the fabric formed beam in Figure D.11. The MUPAD PRO code on the following page shows the implementation of the Natural Logarithm Method for this case, with Figure D.12 visualizing the result compared to a traced image of the photograph in Figure D.11.

```
Mat := Dom::Matrix():
COEF:=Mat([
[0],
[1,1,2.5,0.25*PI,1.86,0.96,0.2,0,0,0,0.3],
[1,1,3.5,0.25*PI,3.63,0.81,0,-0.2,0,0.1,0.8],
[1,1,3,0.25*PI,6,0.72,0,0,0,0,1.0],
[1,1,3.5,0.25*PI,12-3.63,0.81,0,0.2,0,-0.1,0.8],
[1,1,2.5,0.25*PI,12-1.86,0.96,-0.2,0,0,0,0.3],
[0]
])
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2.5 & 0.25\cdot\pi & 1.86 & 0.96 & 0.2 & 0 & 0 & 0 & 0.3 \\ 1 & 1 & 3.5 & 0.25\cdot\pi & 3.63 & 0.81 & 0 & -0.2 & 0 & 0.1 & 0.8 \\ 1 & 1 & 3 & 0.25\cdot\pi & 6 & 0.72 & 0 & 0 & 0 & 0 & 1.0 \\ 1 & 1 & 3.5 & 0.25\cdot\pi & 8.37 & 0.81 & 0 & 0.2 & 0 & -0.1 & 0.8 \\ 1 & 1 & 2.5 & 0.25\cdot\pi & 10.14 & 0.96 & -0.2 & 0 & 0 & 0 & 0.3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

```
n:=7:
r:=[0$n]:
time((for i from 2 to n-1 do
       C_scale  := COEF[i,1]:
       Cx       := COEF[i,2]:
       Cy       := COEF[i,3]:
       A_rotate := COEF[i,4]:
       X_loc    := COEF[i,5]:
       Y_loc    := COEF[i,6]:
       C_skew1  := COEF[i,7]:
       C_skew2  := COEF[i,8]:
       C_trap1  := COEF[i,9]:
       C_trap2  := COEF[i,10]:
       radius   := COEF[i,11]:

       x_1 := Cx*x-X_loc*Cx:
       y_1 := Cy*y-Y_loc*Cy:
       R := sqrt((x_1)^2+(y_1)^2):
       x_2 := R*cos(arg(x_1,y_1)-A_rotate):
       y_2 := R*sin(arg(x_1,y_1)-A_rotate):

       x_3 := x_2
              -(0.5*C_skew1*(x_2+y_2))
              +(0.5*C_skew2*(y_2-x_2))
              -(C_trap1*(x_2+y_2)*sign(y_2-x_2))
              +(C_trap2*(y_2-x_2)*sign(y_2+x_2)):
       y_3 := y_2
              +(0.5*C_skew1*(x_2+y_2))
              +(0.5*C_skew2*(y_2-x_2))
              +(C_trap1*(x_2+y_2)*sign(y_2-x_2))
              +(C_trap2*(y_2-x_2)*sign(y_2+x_2)):

       f[i] := ln(abs(x_3)+abs(y_3)-radius)-ln(abs(x_3)+abs(y_3)-radius+1);
       c[i] := 1/n:
end_for)):

upper_bound:=1.275:
upper_diff:=1:
f[1] := ln(y-upper_bound-upper_diff)-ln(y-upper_bound);
c[1] := 1/n:

height := 1.05:
L := 12:
C1     := 4*height/(L^2):
C2     := 4*height/L:
lower_function:=C1*(x)^2-C2*x:
lower_bound:=y-lower_function-height:
lower_diff:= 1:
f[n] := -ln(abs(lower_bound))+ln(abs(lower_bound+lower_diff));
c[n] := 1/n:
```

$$\ln(y-2.275)-\ln(y-1.275)$$

$$\ln(\left|-x^2\cdot 0.02916666667+0.35\cdot x+y-0.05\right|)-\ln(\left|-x^2\cdot 0.02916666667+0.35\cdot x+y-1.05\right|)$$

```
Mat := Dom::Matrix():
C:=Mat([[ c[1],c[2],c[3],c[4],c[5],c[6],c[7] ]]);
LN:=Mat([ f[1],f[2],f[3],f[4],f[5],f[6],f[7] ]);
hm:=Mat(C*LN);
hm:=hm[1,1]:
f := piecewise([y > (lower_function+height), hm]):
plotfunc3d(f(x, y), x = 0..12, y = 0..1.27, ZRange= -0.4 .. 0.25, Mesh = [120, 20], Scaling =
Constrained, Width = 240 )
```

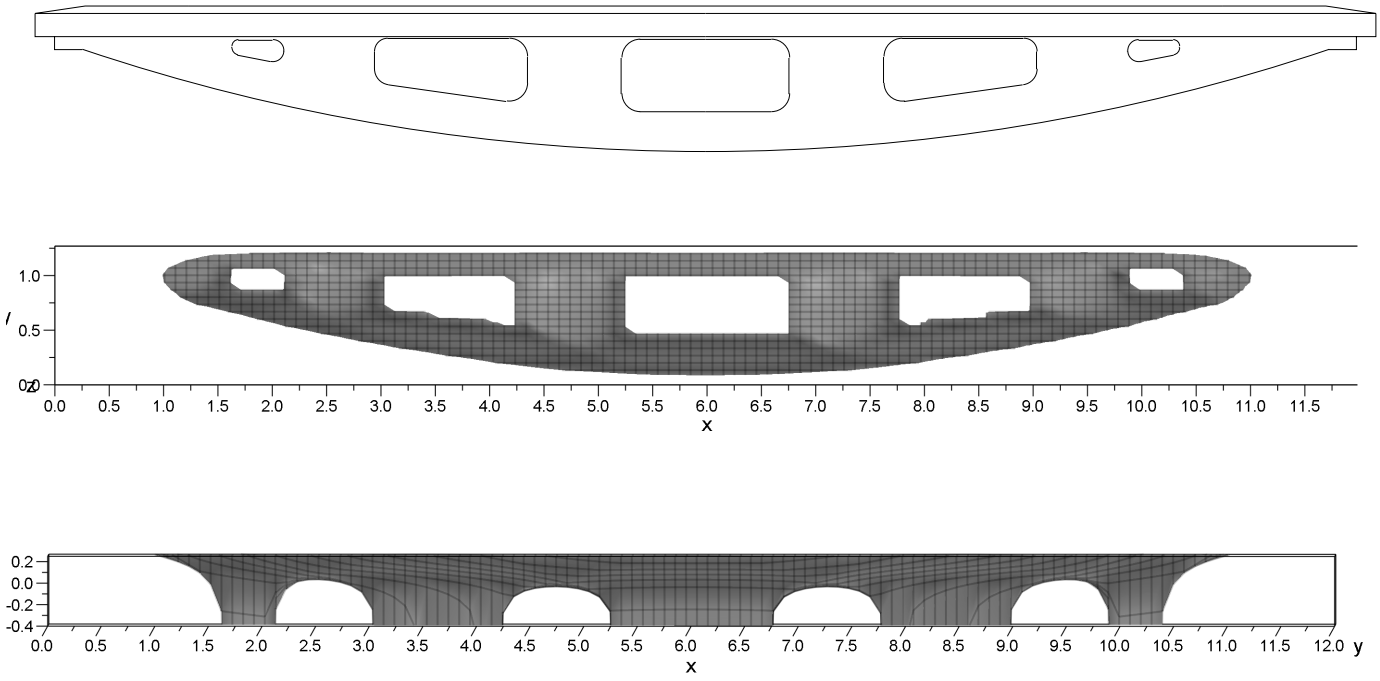


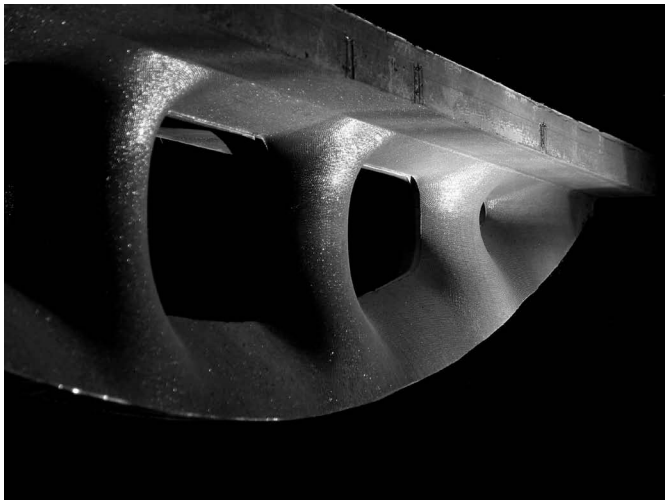
Figure D.12
The result of the code in Section D.2.7 compared to an outline of the fabric formed beam of Figure D.11.

D.5 Conclusions and recommendations

The comparison in Figure D.2 shows that an approximation of a complex geometry is possible using the Natural Logarithm Method. Unfortunately, the computational load of the method became extremely high. While the first tests of single holes took mere seconds to compute and visualize, the approximation of the fabric formed beam took over 6 minutes to calculate. This meant that for the purposes of this thesis, this method would be too demanding as it was expected that such geometries would have to be generated many times over. The reason is that each feature of the geometry i.e. hole, or boundary, adds several terms to the entire formula, making computation increasingly complex.
The initial premise was that computation would be fast at the expense of realism and accuracy of the geometry. However, the final test revealed that no significant advantage was gained in terms of computational time. Therefore, in spite of the five weeks of development time, the decision was made to abandon this method in favor of existing form finding algorithms.

If this method were to be developed further, two recommendations would be:

- To investigate the calculation of the scaling coëfficient C_i and translation coëfficient C, as they are also dependant on the total number and composition of ln-terms. E.g. adding another hole to the formula will affect the geometry at every point and will change its relative position to the boundaries. This has to be compensated by these coëfficients if the user is to have control over the method.
- To compare the approximate geometry with that of the actual fabric geometries. This would provide insight in the margins of error inherent to this kind of approximation and could offer ways to improve the method as well.



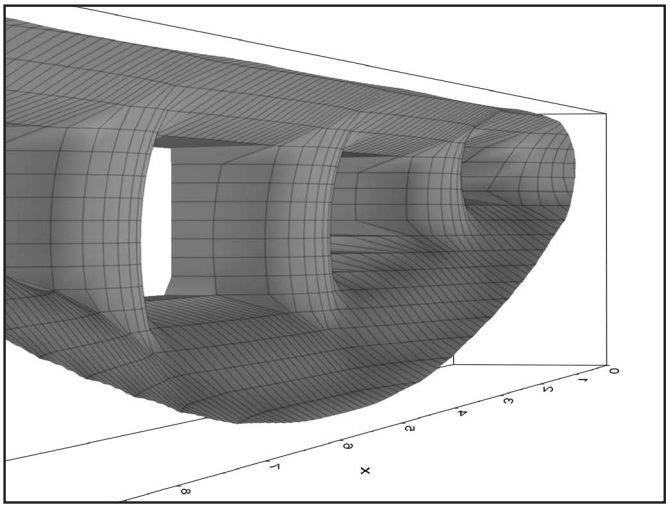
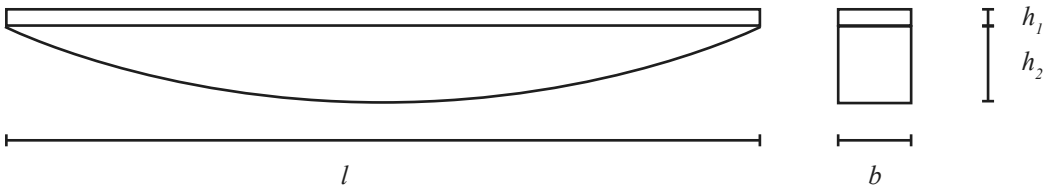


Figure D.13
The result of the code in Section D.2.7 compared to an outline of the fabric formed beam of Figure D.11.

APPENDIX E | MAPLE equations for a parabolically shaped beam

The following equations apply to the figure below where the height of the rectangular and parabolically shaped part of the beam are variables. The equations, written in MAPLE, use energy principles from the theory of elasticity. The result are coefficients *ut* and *a*, describing the amount of deflection in such a beam due to a distributed load, together offering a general equation for this case. If h_2 equals zero, the sum of *ut* and a is 5/384, which should be familiar to most structural engineers. The equations were used for the comparison of fabric formed results with this type of beam in Chapter 8.




```
> restart:
> h:=h1+h2*(1-2*x/L)*(1+2*x/L);
```

$$h := h1 + h2\left(1 - \frac{2x}{L}\right)\left(1 + \frac{2x}{L}\right)$$

```
> u:=ut*(1-2*x/L)*(1+2*x/L):
> k:=-diff(u,x,x):
> II:=1/12*b*h^3:
> M:=E*II*k:
> Evv:=int(1/2*M*k,x=-L/2..L/2);
```

$$Evv := -\frac{8}{21}\frac{E\,b\,h2^3\,ut^2}{L^3} + \frac{8}{5}\frac{E\,b\,(h1+h2)\,h2^2\,ut^2}{L^3} - \frac{8}{3}\frac{E\,b\,(h1+h2)^2\,h2\,ut^2}{L^3} + \frac{8}{3}\frac{E\,b\,(h1+h2)^3\,ut^2}{L^3}$$

```
> Epl:=-int(q*u,x=-L/2..L/2);
```

$$Epl := -\frac{2}{3}\,q\,ut\,L$$

```
> Epot:=Evv+Epl;
```

$$Epot := -\frac{8}{21}\,\frac{E\,b\,h2^3\,ut^2}{L^3} + \frac{8}{5}\,\frac{E\,b\,(h1+h2)\,h2^2\,ut^2}{L^3} - \frac{8}{3}\,\frac{E\,b\,(h1+h2)^2\,h2\,ut^2}{L^3} + \frac{8}{3}\,\frac{E\,b\,(h1+h2)^3\,ut^2}{L^3} - \frac{2}{3}\,q\,ut\,L$$

```
> diff(Epot,ut);
```

$$-\frac{16}{21}\,\frac{E\,b\,h2^3\,ut}{L^3} + \frac{16}{5}\,\frac{E\,b\,(h1+h2)\,h2^2\,ut}{L^3} - \frac{16}{3}\,\frac{E\,b\,(h1+h2)^2\,h2\,ut}{L^3} + \frac{16}{3}\,\frac{E\,b\,(h1+h2)^3\,ut}{L^3} - \frac{2}{3}\,q\,L$$

```
> ut:=solve(diff(Epot,ut)=0,ut);
```

$$ut := \frac{35}{8}\,\frac{q\,L^4}{E\,b\,\left(56\,h2^2\,h1 + 16\,h2^3 + 70\,h2\,h1^2 + 35\,h1^3\right)}$$

```
> restart:
> # h1:=0: L:=9000:
> # h2:=0:
> h:=h1+h2*(1-2*x/L)*(1+2*x/L);
```

$$h := h1 + h2\left(1-\frac{2\,x}{L}\right)\left(1+\frac{2\,x}{L}\right)$$

```
> # plot(h,x=-L/2..L/2);
> u:=ut*(1-2*x/L)*(1+2*x/L) + a*(1-2*x/L)^2*(1+2*x/L)^2;
```

$$u := ut\left(1-\frac{2\,x}{L}\right)\left(1+\frac{2\,x}{L}\right) + a\left(1-\frac{2\,x}{L}\right)^2\left(1+\frac{2\,x}{L}\right)^2$$

```
> k:=-diff(u,x,x):
> II:=1/12*b*h^3;
```

$$II := \frac{1}{12}\,b\left(h1 + h2\left(1-\frac{2\,x}{L}\right)\left(1+\frac{2\,x}{L}\right)\right)^3$$

```
> M:=E*II*k:
> Evv:=int(1/2*M*k,x=-L/2..L/2):
> Epl:=-int(q*u,x=-L/2..L/2):
> Epot:=Evv+Epl:
> eq1:=diff(Epot,ut)=0;
```

$$eq1 := \frac{32}{9}\,\frac{E\,b\,h2^3\,a}{L^3} + \frac{1}{448}\left(-\frac{6144\,E\,b\,(h1+h2)\,h2^2\,a}{L^{10}}\right.$$

$$-\frac{128}{3}\,\frac{E\,b\,h2^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^8}\Bigg)L^7+\frac{1}{80}\left(\frac{1536\,E\,b\,(h1+h2)^2\,h2\,a}{L^8}\right.$$

$$\left.+\frac{32\,E\,b\,(h1+h2)\,h2^2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^6}\right)L^5+\frac{1}{12}\left(-\frac{128\,E\,b\,(h1+h2)^3\,a}{L^6}-\right.$$

$$\left.\frac{8\,E\,b\,(h1+h2)^2\,h2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^4}\right)L^3+\frac{2}{3}$$

$$\frac{E\,b\,(h1+h2)^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L}-\frac{2}{3}\,q\,L=0$$

```
> eq2:=diff(Epot,a)=0;
```

$$eq2:=-\frac{192}{11}\,\frac{E\,b\,h2^3\,a}{L^3}$$

$$+\frac{1}{2304}\left(\frac{147456\,E\,b\,(h1+h2)\,h2^2\,a}{L^{12}}+\frac{16384\,E\,b\,h2^3\,a}{L^{12}}\right.$$

$$\left.+\frac{1024\,E\,b\,h2^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^{10}}\right)L^9+\frac{1}{448}\left(-\frac{36864\,E\,b\,(h1+h2)^2\,h2\,a}{L^{10}}-\right.$$

$$\left.\frac{12288\,E\,b\,(h1+h2)\,h2^2\,a}{L^{10}}-\frac{768\,E\,b\,(h1+h2)\,h2^2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^8}\right.$$

$$\left.-\frac{256}{3}\,\frac{E\,b\,h2^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^8}\right)L^7+\frac{1}{80}\left(\frac{3072\,E\,b\,(h1+h2)^3\,a}{L^8}\right.$$

$$+\frac{3072\,E\,b\,(h1+h2)^2\,h2\,a}{L^8}+\frac{192\,E\,b\,(h1+h2)^2\,h2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^6}$$

$$+\frac{64\,E\,b\,(h1+h2)\,h2^2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^6}\Bigg)L^5+\frac{1}{12}\left(-\frac{256\,E\,b\,(h1+h2)^3\,a}{L^6}-\right.$$

$$\left.\frac{16\,E\,b\,(h1+h2)^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^4}-\frac{16\,E\,b\,(h1+h2)^2\,h2\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L^4}\right)$$

$$L^3+\frac{4}{3}\,\frac{E\,b\,(h1+h2)^3\left(\frac{8\,ut}{L^2}+\frac{16\,a}{L^2}\right)}{L}-\frac{8}{15}\,q\,L=0$$

```
> solve({eq1,eq2},{ut,a});
```

$$\left\{ut=\frac{21}{8}\right.$$

$$\left(q\,L^4\left(224\,h2^3+1188\,h1^2\,h2+792\,h1\,h2^2+1155\,h1^3\right)\right)\Big/\left(E\,b\right.$$

$$\left(9216\,h1\,h2^5+36864\,h1^2\,h2^4+86016\,h1^3\,h2^3+116424\,h1^4\,h2^2+1024\,h2^6\right.$$

$$\left.+24255\,h1^6+83160\,h1^5\,h2\right)\Big),\ a=\frac{231}{32}\left(\left(105\,h1^3-72\,h1\,h2^2-32\,h2^3\right)q\,L^4\right)\Big/($$

$$E\,b\left(9216\,h1\,h2^5+36864\,h1^2\,h2^4+86016\,h1^3\,h2^3+116424\,h1^4\,h2^2+1024\,h2^6\right.$$

$$\left.\left.+24255\,h1^6+83160\,h1^5\,h2\right)\right)\Bigg\}$$

APPENDIX F | ANSYSscript code for ESO 1.3 and BESO 2.05

Code in this Appendix was written in NOTEPAD for direct input in finite element program ANSYS. The version numbers 1.3 and 2.05 refer to development stages of this project and have no general meaning. The code was written by the author himself based on sources mentioned throughout Chapter 2 and Appendix A. These specific versions are generally the same as existing versions of both ESO and BESO, with the exception of the adapted Removal Rate of Volume in BESO, as described in Section 2.5. The first piece of code, describing ESO, inputs the parameters for a spaghetti-bridge building contest at this faculty in 2007.

```
FINISH
/CLEAR
/PLOPTS,LEG1,OFF
/PLOPTS,FRAME,OFF
/PLOPTS,MINM,OFF
/FILNAME,ESO_MICHELL,0
/GST,OFF
/SHOW,PNG

*SET,WIDTH,100
*SET,HEIGHT,4500
*SET,LENGTH,4500
*SET,ELEMENTSIZE,100

/PREP7
BLC4,0,0,WIDTH,HEIGHT,LENGTH
ET,1,SOLID45
MP,EX,1,100000
MP,PRXY,1,0.3
ESIZE,ELEMENTSIZE
VMESH,ALL
FINISH

/SOLU
ANTYPE,STATIC,NEW
NROPT, FULL
NSEL,S,LOC,Y,0,
NSEL,R,LOC,Z,LENGTH,
F,ALL,FY,-1000,
LSEL,S,LOC,Y,0,
LSEL,R,LOC,Z,0,
NSLL,S,1
D,ALL,UY,
D,ALL,UZ,
LSEL,S,LOC,Y,0,
```

```
LSEL,R,LOC,Z,0,
NSLL,S,1
D,ALL,UY,
D,ALL,UZ,
ESEL,S,LIVE
NSEL,ALL
*SET,RRINIT,1
*SET,ER,0.5
*SET,RREND,40
*SET,RR,RRINIT
*SET,I,1
*SET,N,((RREND-RRINIT)+ER)/ER
*DIM,OUTPUT,TABLE,N,5,1
*SET,SMIN,1

*DO, RR, RRINIT, RREND, ER
   /TITLE, ESO MICHELL TRUSS     [ STEP NO. %I% WITH REJECTION RATIO RR = %RR% ]
   /SOLU
   SOLVE

   /POST1
   ETABLE,VONMISES,S,EQV
   ESORT,ETAB,VONMISES,0,1,,
   *GET,SMAX,SORT,,MAX
   *GET,SMIN,SORT,,MIN
   *SET,OUTPUT(I,1),I
   *SET,OUTPUT(I,2),RR
   *SET,OUTPUT(I,3),SMAX
   *SET,OUTPUT(I,4),SMIN
   ETABLE,HEIGHT_Y,CENT,Y
   ESORT,ETAB,HEIGHT_Y,0,1,,
   ETABLE,LENGTH_Z,CENT,Z
   ESORT,ETAB,LENGTH_Z,0,1,,
   ESEL,S,ETAB,VONMISES,0,SMAX*RR/100,,
   FINISH

   /SOLU
   EKILL,ALL
   ESELL,ALL
   FINISH

   /POST1
   ESEL,S,LIVE
   *GET,ELEMENTCOUNT,ELEM,,COUNT
   *SET,VOLUME,ELEMENTCOUNT*(ELEMENTSIZE**3)/(10**9)
   *SET,OUTPUT(I,5),VOLUME

   /VIEW,1,1,0,0
   PLESOL,S,EQV
   FINISH

   *SET,RR,RR+ER
   *SET,I,I+1
*ENDDO

/AXLAB,Y,SMAX [N/MM2]
/XRANGE,0,I
*VPLOT,,OUTPUT(1,3)
/SHOW,CLOSE,
```

The following piece of code describes a BESO algorithm for a simply supported beam with a distributed load with supports at the top corner as seen in Figure 2.22 of Chapter 2.

```
FINISH
/CLEAR
/PLOPTS,LEG1,OFF
/PLOPTS,FRAME,OFF
/PLOPTS,MINM,OFF
/FILNAME,BESO,0
/GST,OFF
/SHOW,PNG,
/UNITS,SI

*SET,WIDTH,1000         ! WIDTH [MM]
*SET,HEIGHT,20000       ! HEIGHT
*SET,LENGTH,140000      ! LENGTH
*SET,E,500
*SET,SYMM_X, 0.5*WIDTH
*SET,SYMM_Z, 0.5*LENGTH

/PREP7                  ! ENTER THE PREPROCESSOR
K,1,0,0,0               ! CORNERS OF TOTAL VOLUME
K,2,0.5*WIDTH,0,0
K,3,0.5*WIDTH,HEIGHT,0
K,4,0,HEIGHT, 0
K,5,0,0,0.5*LENGTH
K,6,0.5*WIDTH,0,0.5*LENGTH
K,7,0.5*WIDTH,HEIGHT,0.5*LENGTH
K,8,0,HEIGHT,0.5*LENGTH
V,1,2,3,4,5,6,7,8       ! VOLUME DEFINED BY CORNERS

ET,1,SOLID45            ! ELEMENT TYPE
MP,EX,1,210000          ! DEFINE STIFFNESS
MP,PRXY,1,0.3           ! DEFINE POISSON
MP,DENS,1,2.5E-6        ! DEFINE DENSITY [KG/MM3]
ESIZE,E                 ! MESH SIZE
VMESH,ALL               ! MESH AREA
FINISH

/SOLU                   ! ENTER SOLUTION PHASE
ANTYPE,STATIC,NEW       ! STATIC ANALYSIS
NLGEOM,ON
NROPT,FULL
FINISH

/PREP7
KSEL,S,KP,,4
NSLK,S
D,ALL,UY,0,
D,ALL,UZ,0,
SFA,4,,PRES,0.0001      ! IN [N/MM2]
SBCTRAN
NSEL,S,LOC,Z,SYMM_Z
DSYM,SYMM,Z
NSEL,S,LOC,X,SYMM_X
DSYM,SYMM,X
FINISH
```

```
! ----------------- !
! MAIN OPTIMIZATION !
! ----------------- !

*SET,RRV,0.05
*SET,RRVEND,0.20
*SET,ER,0.05
*SET,ERROR,0.0001
*SET,BODYCOUNT,0
*SET,RESURRECTION,0
*SET,NET_CHANGE_LIFE,0
*SET,RESURRECTION_OLD,RESURRECTION
*SET,BODYCOUNT_OLD,BODYCOUNT
*SET,OUTPUT_ROWS, 500
*DIM,OUTPUT,TABLE,OUTPUT_ROWS,15,1

*DO,ROW,1,OUTPUT_ROWS,1
   *TAXIS,OUTPUT(ROW,1),1,ROW
*ENDDO

*DO,COLUMN,1,15,1
   *TAXIS,OUTPUT(1,COLUMN),2,COLUMN
*ENDDO

*DIM,OUTPUT_SORTED,TABLE,OUTPUT_ROWS,2,1
NSEL,ALL
*GET,ALLNODES,NODE,,COUNT
*DIM,SENSITIVITY,TABLE,ALLNODES,6,1,NODE,ALPHA
*DIM,ORTHO_COORD_K1,ARRAY,6,3
*DIM,ORTHO_COORD_K2,ARRAY, 6,3
*DIM,NEAREST_NODES_K1,ARRAY,6,1
*DIM,NEAREST_NODES_K2,ARRAY, 6,1
*DIM,NEAREST_NODES_K1_TEST,ARRAY, 6,1
*DIM,NEAREST_NODES_K2_TEST,ARRAY, 6,1
*DIM,ALPHA_K0,ARRAY,6,1
*DIM,ALPHA_K0_CHECK,ARRAY,6,1
ESEL,ALL
*GET,ALLELEMENTS,ELEM,,COUNT
*DIM,SENS_ELEM_SORTED,TABLE,ALLELEMENTS,8,1,ELEMENT,ALPHA_E
*DIM,SENS_ELEM,TABLE,ALLELEMENTS,8,1,ELEMENT,ALPHA_E

*IF,ER,GT,0,THEN
   *SET, NRRV, NINT(((1-RRV)+ER)/ER)
*ELSE
   *SET, NRRV, 1
*ENDIF

*DIM,TOTAL,TABLE,NRRV, 2,1,
*TAXIS, TOTAL(1,1),2,1,2
*SET,P,1

*DO,RRV,RRV,RRVEND,ER
   *SET,V,0.5*WIDTH*HEIGHT*0.5*LENGTH/(1000**3)
   *SET, OSCILLATION, 0
   *SET,I,1
   *SET,PI,1

   *DOWHILE,I
      /TITLE, STEP %I% RRV=%RRV% PI=%PI_NEW%
      *IF, V, EQ, 0, EXIT
```

```
        /SOLU
        ESEL, S, LIVE
        NSLE,S, ALL
        SOLVE
        FINISH

        /POST1
        ETABLE, VONMISES,S,EQV
        ETABLE,STRAINENERGYDENSITY ,SEND, ELASTIC
        ETABLE,STRAINENERGY ,SENE
        ESORT,ETAB,VONMISES, 0,1,,
        *GET,SMAX,SORT,,MAX
        *GET,SMIN,SORT,,MIN
        *SET,OUTPUT(I,1),I
        *SET,OUTPUT(I,2),RRV
        *SET,OUTPUT(I,3),SMAX
        *SET,OUTPUT(I,4),SMIN
        ESEL,S, LIVE
        *GET, LIVEELEMENTS,ELEM,,COUNT
        *SET,OUTPUT(I,5),V
        *SET,OUTPUT(I,6),LIVEELEMENTS
        *SET,OUTPUT(I,7),BODYCOUNT
        *SET,OUTPUT(I,8),RESURRECTION
        *SET,OUTPUT(I,9),NET_CHANGE_LIFE
        SAVE,STEP%I%,DB

! ---------------------------------------------- !
! CALCULATE SENSITIVITY ALPHA_K FOR ALL LIVE NODES !
! ---------------------------------------------- !

        ESEL, S, LIVE
        NSLE,S,ALL
        *GET,NEXTLIVENODE,NODE,,NUM,MIN
        NSEL,S,NODE,,NEXTLIVENODE
        *TAXIS, SENSITIVITY(1,1),2,1,2,3,4,5,6

        *DO, INDEX, 1, ALLNODES
           *TAXIS,SENSITIVITY(INDEX,1),1,INDEX
        *ENDDO

        *SET, K, 1
        FINISH

        /POST1

        *DOWHILE,NEXTLIVENODE
           ESLN,S,0,ALL
           ETABLE,ALPHA_I,SENE
           ETABLE, VOLUME, VOLU
           SEXP, ALPHA_S, VOLUME, ALPHA_I, 1, 1
           SSUM
           *GET, SUMALPHA_S, SSUM,,ITEM,ALPHA_S
           *GET, SUMVOLUME, SSUM,,ITEM,VOLUME
           *GET,X,NODE,NEXTLIVENODE,LOC,X,
           *SET, SENSITIVITY(NEXTLIVENODE,4),X/1000
           *GET,Y,NODE,NEXTLIVENODE,LOC,Y,
           *SET, SENSITIVITY(NEXTLIVENODE,5),Y/1000
           *GET,Z,NODE,NEXTLIVENODE,LOC,Z,
           *SET, SENSITIVITY(NEXTLIVENODE,6),Z/1000
           *SET, ALPHA_K, (SUMALPHA_S/SUMVOLUME)
```

```
            *SET,SENSITIVITY(NEXTLIVENODE,3),3
            *SET, SENSITIVITY(NEXTLIVENODE,1),ALPHA_K
            ESEL, S, LIVE
            NSLE, S, ALL
            *GET, NEXTLIVENODE, NODE, NEXTLIVENODE, NXTH
            *IF,NEXTLIVENODE,EQ,0,EXIT
            NSEL,S,NODE,,NEXTLIVENODE

            *SET,K,K+1
        *ENDDO
        FINISH

! ---------------------------------------------- !
! CALCULATE SENSITIVITY ALPHA_K FOR ALL DEAD NODES !
! ---------------------------------------------- !

        ESEL, S, LIVE
        NSLE, S, ALL
        ESLN, S, 0, ALL
        NSLE, ALL
        ESEL, S, LIVE
        NSLE, U, ALL
        *GET, DEADNODES, NODE,, COUNT
        *IF, DEADNODES, GT, 0, THEN
            *GET, NEXTDEADNODE, NODE, , NUM, MIN
            NSEL, S, NODE, ,NEXTDEADNODE

            *DOWHILE,NEXTDEADNODE
                *GET,X,NODE,NEXTDEADNODE,LOC,X,
                *SET, SENSITIVITY(NEXTDEADNODE,4),X/1000
                *GET,Y,NODE,NEXTDEADNODE,LOC,Y,
                *SET, SENSITIVITY(NEXTDEADNODE,5),Y/1000
                *GET,Z,NODE,NEXTDEADNODE,LOC,Z,
                *SET, SENSITIVITY(NEXTDEADNODE,6),Z/1000

                *IF, X, EQ, SYMM_X, THEN
                    *SET, LOC_X1, X-E
                    *SET, LOC_X2, X-2*E
                *ELSEIF, X, EQ, SYMM_X-E, THEN
                    *SET, LOC_X1, X+E
                    *SET, LOC_X2, X
                *ELSE
                    *SET, LOC_X1, X+E
                    *SET, LOC_X2, X+2*E
                *ENDIF

                *IF, Z, EQ, SYMM_Z, THEN
                    *SET, LOC_Z1, Z-E
                    *SET, LOC_Z2, Z-2*E
                *ELSEIF, Z, EQ, SYMM_Z-E, THEN
                    *SET, LOC_Z1, Z+E
                    *SET, LOC_Z2, Z
                *ELSE
                    *SET, LOC_Z1, Z+E
                    *SET, LOC_Z2, Z+2*E
                *ENDIF

                *SET, ORTHO_COORD_K1(1,1), LOC_X1, X-E,     X,       X,       X,       X
                *SET, ORTHO_COORD_K1(1,2), Y,      Y,       Y+E,     Y-E,     Y,       Y
                *SET, ORTHO_COORD_K1(1,3), Z,      Z,       Z,       Z,       LOC_Z1,  Z-E
                *SET, ORTHO_COORD_K2(1,1), LOC_X2, X-2*E,   X,       X,       X,       X
```

```
*SET, ORTHO_COORD_K2(1,2), Y,      Y,       Y+2*E,   Y-2*E,   Y,       Y
*SET, ORTHO_COORD_K2(1,3), Z,      Z,       Z,       Z,       LOC_Z2,  Z-2*E
NSEL, ALL
*MOPER, NEAREST_NODES_K1, ORTHO_COORD_K1, NNEAR, 0.5*E
*MOPER, NEAREST_NODES_K1_TEST, ORTHO_COORD_K1, NNEAR, 0.5*E
*SET, B, 1

*DO, B, 1, 6, 1

   *IF, NEAREST_NODES_K1(B,1), EQ, 0, THEN
      *SET, NEAREST_NODES_K1(B,1), 0
   *ELSE
      *SET, NEAREST_NODES_K1(B,1), SENSITIVITY(NEAREST_NODES_K1(B,1),1)
   *ENDIF

*ENDDO

*MOPER, NEAREST_NODES_K2, ORTHO_COORD_K2, NNEAR, 0.5*E
*MOPER, NEAREST_NODES_K2_TEST, ORTHO_COORD_K2, NNEAR, 0.5*E
*SET, C, 1

*DO, C, 1, 6, 1

   *IF, NEAREST_NODES_K2(C,1), EQ, 0, THEN
      *SET, NEAREST_NODES_K2(C,1), 0
   *ELSE
      *SET, NEAREST_NODES_K2(C,1), SENSITIVITY(NEAREST_NODES_K2(C,1),1)
   *ENDIF

*ENDDO

*VOPER, ALPHA_K0_CHECK, NEAREST_NODES_K1 ,MULT, NEAREST_NODES_K2
*VOPER, ALPHA_K0, NEAREST_NODES_K1 ,ADD, NEAREST_NODES_K1
*VOPER, ALPHA_K0, ALPHA_K0 ,SUB, NEAREST_NODES_K2
*SET, SUMALPHA_K0,0
*SET, NUM, 0
*SET, D, 1

*DO, D, 1, 6, 1

   *IF, ALPHA_K0_CHECK(D,1), GT, 0, THEN
      *SET, SUMALPHA_K0, SUMALPHA_K0+ALPHA_K0(D,1)
      *SET, NUM, NUM+1
   *ELSE
   *ENDIF

*ENDDO

*IF,NUM,GT,0,THEN
   *SET,AVG_ALPHA_K0,(SUMALPHA_K0/NUM)
   *SET,SENSITIVITY(NEXTDEADNODE,2),AVG_ALPHA_K0
   *SET,SENSITIVITY(NEXTDEADNODE,3),1
*ELSE
   *SET,SENSITIVITY(NEXTDEADNODE,2),0
   *SET,SENSITIVITY(NEXTDEADNODE,3),2
*ENDIF

ESEL, S, LIVE
NSLE, S, ALL
ESLN, S, 0, ALL
NSLE, ALL
```

```
               ESEL, S, LIVE
               NSLE, U, ALL
               *GET, NEXTDEADNODE, NODE, NEXTDEADNODE, NXTH
               *IF,NEXTDEADNODE,EQ,0,EXIT
               NSEL, S, NODE, ,NEXTDEADNODE

            *ENDDO

         *ELSE
         *ENDIF

         *SET, NEXTEACHNODE, ALLNODES

         *DOWHILE, NEXTEACHNODE

            *IF, SENSITIVITY(NEXTEACHNODE,1), EQ, 0, THEN
               *SET, SENSITIVITY(NEXTEACHNODE,1), SENSITIVITY(NEXTEACHNODE,2)
            *ELSE
            *ENDIF

            *SET, NEXTEACHNODE, NEXTEACHNODE-1

         *ENDDO

! ---------------------------------------------- !
! CALCULATE SENSITIVITY ALPHA_E FOR ALL ELEMENTS !
! ---------------------------------------------- !

         *TAXIS, SENS_ELEM(1,1),2,1,2,3,4,5,6,7,8
         *TAXIS, SENS_ELEM_SORTED(1),2,1,2,3,4,5,6,7,8
         *SET, TELLER_A, 1

         *DO, TELLER_A, 1, ALLELEMENTS, 1
            *TAXIS, SENS_ELEM(TELLER_A,1),1,TELLER_A
            *TAXIS, SENS_ELEM_SORTED(TELLER_A,1),1,TELLER_A
            *GET, CENT_X, ELEM, TELLER_A, CENT, X
            *SET, SENS_ELEM(TELLER_A,6),X/1000
            *GET, CENT_Y, ELEM, TELLER_A, CENT, Y
            *SET, SENS_ELEM(TELLER_A,7),Y/1000
            *GET, CENT_Z, ELEM, TELLER_A, CENT, Z
            *SET, SENS_ELEM(TELLER_A,8),Z/1000
            *SET, SENS_ELEM(TELLER_A,1), 0
            *SET, SENS_ELEM(TELLER_A,2), 0
            *SET, SENS_ELEM(TELLER_A,3), 0
            *SET, SENS_ELEM(TELLER_A,4), 0
            *SET, SENS_ELEM(TELLER_A,5), 1
         *ENDDO

         ESEL,S,LIVE
         NSLE,S,ALL
         ESLN,S,0,ALL
         *GET,NEXTELEM,ELEM,,NUM,MIN
         ESEL,S,ELEM,,NEXTELEM

         *DOWHILE, NEXTELEM
            *GET,ALIVE,ELEM, NEXTELEM, ATTR, LIVE

         *IF, ALIVE, EQ, 1, THEN
               *GET, CENT_Y, ELEM, NEXTELEM, CENT, Y

               *IF, CENT_Y, GT, HEIGHT-3*E, THEN
```

```
                *SET, SENS_ELEM(NEXTELEM,3),0
                *SET, SENS_ELEM(NEXTELEM,4),1
            *ELSE
                *SET, SENS_ELEM(NEXTELEM,3),1
                *SET, SENS_ELEM(NEXTELEM,4),0
            *ENDIF

        *ELSE
            *SET, SENS_ELEM(NEXTELEM,3),0
            *SET, SENS_ELEM(NEXTELEM,4),0
        *ENDIF

        NSLE, S, ALL
        *GET,NEXTNODE,NODE,,NUM,MIN
        NSEL,S,NODE,,NEXTNODE
        *SET, SUMALPHA_N, 0
        *SET, N, 0
        *SET, N_TOTAL, 0

        *DOWHILE, NEXTNODE
            *SET, ALPHA_N, SENSITIVITY(NEXTNODE,1)
            *SET, SUMALPHA_N, SUMALPHA_N + ALPHA_N
            *SET, N, N+1
            *SET, N_TOTAL, N_TOTAL+1
            ESEL, S, ELEM, ,NEXTELEM
            NSLE, S, ALL
            *GET, NEXTNODE, NODE, NEXTNODE, NXTH
            *IF, NEXTNODE, EQ, 0, EXIT
            *SET, N, N-1
            NSEL, S, NODE, , NEXTNODE
        *ENDDO

        *SET, ALPHA_E, SUMALPHA_N/N_TOTAL
        *SET, SENS_ELEM(NEXTELEM,1),NEXTELEM
        *SET, SENS_ELEM(NEXTELEM,2),ALPHA_E
        ESEL, S, LIVE
        NSLE, S, ALL
        ESLN, S, 0, ALL
        *GET, NEXTELEM, ELEM, NEXTELEM, NXTH
        *IF, NEXTELEM, EQ, 0, EXIT
        ESEL, S, ELEM, , NEXTELEM

    *ENDDO

! ------------------------------------------ !
! CALCULATE THRESHOLD ALPHA_E FOR NEW SELECTION !
! ------------------------------------------ !

    ESEL, S, LIVE
    *MOPER, SENS_ELEM_SORTED, SENS_ELEM, SORT, SENS_ELEM(1,2)
    *SET, REMOVED_VOLUME, NINT(RRV*LIVEELEMENTS)
    *SET, J, 2

    *DOWHILE, J
        SENS_ELEM(J,5)=SENS_ELEM(J,5)+SENS_ELEM(J-1,5)
        SENS_ELEM(J,3)=SENS_ELEM(J,3)+SENS_ELEM(J-1,3)

        *IF, SENS_ELEM(J,3), EQ, REMOVED_VOLUME, AND, SENS_ELEM(J,2), NE, 0, THEN
            *SET, ALPHA_TH, SENS_ELEM(J,2)
            *SET, ALPHA_TH_ELEM, SENS_ELEM(J,1)
            *EXIT
```

```
            *ELSE
            *ENDIF
            *IF, J, EQ, ALLELEMENTS, THEN
               *SET, ALPHA_TH, SENS_ELEM(J,2)

               *SET, ALPHA_TH_ELEM, SENS_ELEM(J,1)
               *EXIT
            *ENDIF

            *SET, J, J+1

         *ENDDO

         *MOPER, SENS_ELEM_SORTED, SENS_ELEM, SORT, SENS_ELEM_SORTED(1,1)

   ! --------------------------------!
   ! SELECTION OF ADDITION AND REMOVAL !
   ! -------------------------------- !

         *SET, RESURRECTION, 0
         *SET, BODYCOUNT, 0
         ESEL, ALL
         *GET, NEXTELEM2, ELEM, , NUM, MIN
         ESEL, S, ELEM, ,NEXTELEM2

         *DOWHILE, NEXTELEM2
            *GET,ALIVE,ELEM, NEXTELEM2, ATTR, LIVE

            *IF, SENS_ELEM(NEXTELEM2,2), LT, ALPHA_TH, THEN

               *IF, SENS_ELEM(NEXTELEM2,4), EQ, 1, THEN
               *ELSE
                  /SOLU
                  EKILL, ALL
               *ENDIF

            *ELSE
               /SOLU
               EALIVE, ALL
            *ENDIF

            FINISH
            *GET,DEATH,ELEM, NEXTELEM2, ATTR, LIVE

            *IF, ALIVE-DEATH, EQ, -2, THEN
               *SET, RESURRECTION, RESURRECTION+1
            *ELSEIF, ALIVE-DEATH, EQ, 2, THEN
               *SET, BODYCOUNT, BODYCOUNT+1
            *ELSE
            *ENDIF

            ESEL, ALL
            *GET, NEXTELEM2, ELEM, NEXTELEM2, NXTH
            *IF, NEXTELEM2, EQ, 0, EXIT
            ESEL, S, ELEM, , NEXTELEM2

         *ENDDO

         ESEL,S, LIVE
         *GET, LIVEELEMENTS_NEW,ELEM,,COUNT
         *SET, NET_CHANGE_LIFE, LIVEELEMENTS_NEW-LIVEELEMENTS
```

```
! --------------------- !
! CALCULATE OSCILLATION !
! --------------------- !

       *IF, RESURRECTION, EQ, BODYCOUNT_OLD, THEN


          *IF, BODYCOUNT, EQ, RESURRECTION_OLD, THEN
             *SET, OSCILLATION, OSCILLATION+1
          *ELSE
             *SET, OSCILLATION, 0
          *ENDIF

       *ELSE
          *SET, OSCILLATION, 0
       *ENDIF

       *IF, OSCILLATION, EQ, 3, CYCLE
       *SET, RESURRECTION_OLD, RESURRECTION
       *SET, BODYCOUNT_OLD, BODYCOUNT

! ---------------------------------------- !
! CALCULATE PERFORMANCE INDEX AND CONVERGENCE !
! ---------------------------------------- !

       *SET,V,LIVEELEMENTS_NEW
       ESEL, S, LIVE
       *IF, V, LE, 0, EXIT
       *SET, PI_NEW, 1/(4*S*4*V)
       *SET,ERROR_I,ABS(PI_NEW-PI)/PI_NEW
       *SET, PI, PI_NEW
       *SET,OUTPUT(I,10),PI
       *SET,OUTPUT(I,11),ERROR_I
       *SET,OUTPUT(I,13),REMOVED_VOLUME
       *SET,OUTPUT(I,14),ALPHA_TH
       *SET,OUTPUT(I,15),ALPHA_TH_ELEM
       *IF, ERROR_I, LE, ERROR, EXIT

       /POST1
       ETABLE, VONMISES,S,EQV
       ESORT,ETAB,VONMISES, 0,1,,
       *GET,SMAX_NEW,SORT,,MAX
       FINISH

       *IF, SMAX_NEW, GT, 2*SMAX, EXIT
       *SET, I, I+1

   *ENDDO

   /AXLAB,Y,PERFORMANCE INDEX
   /XRANGE,0,I
   *VPLOT,,OUTPUT(1,10)
   *MOPER,OUTPUT_SORTED, OUTPUT, SORT, OUTPUT(1,11)
   *TAXIS, TOTAL(P,1),1,P

   *SET, Q, 1
   *DOWHILE, Q
      *IF, OUTPUT(Q,11), EQ,0, THEN
      *SET, Q, Q+1
      *ELSE
      *EXIT
```

```
        *ENDIF
    *ENDDO

    *SET,TOTAL(P,1),OUTPUT(Q,5)
    *SET,TOTAL(P,2),OUTPUT(Q,10)
    *MOPER, OUTPUT_SORTED, OUTPUT, SORT, OUTPUT_SORTED(1,1)
    *SET, P, P+1

*ENDDO

/AXLAB,X,RRV LOOP [N]
/AXLAB,Y,PERFORMANCE INDEX
/XRANGE,0,P
*VPLOT,,TOTAL(1,2)

ESEL, S, LIVE
/SHOW,CLOSE,
```