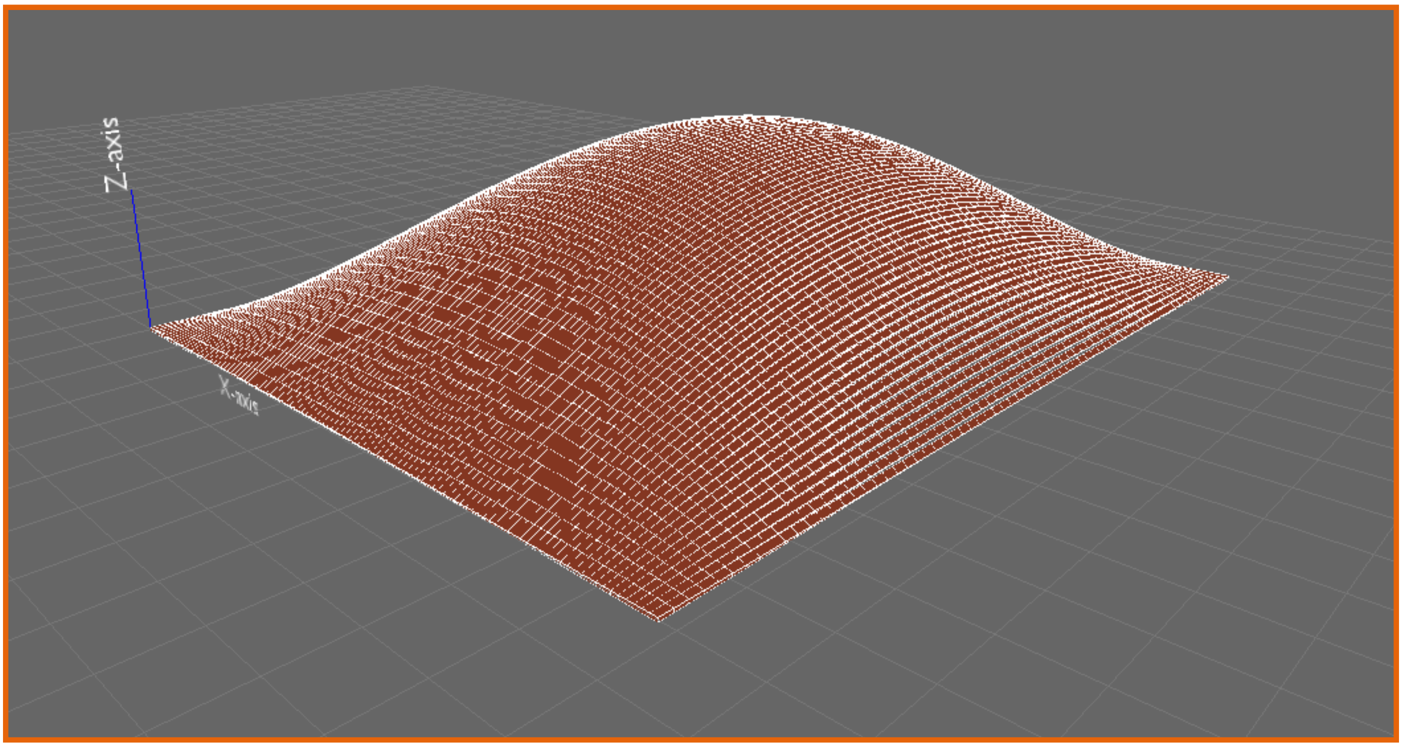
Proposal for a tool to design masonry double-curved shells

Analysis of conceptual models and generation of a masonry pattern



T.J. van Swinderen

TU Delft
Delft University of Technology

structural design lab

Part I - Main report

T.J. van Swinderen

Delft, August 2009

Delft University of Technology

Delft University of Technology
Faculty Civil Engineering and Geosciences
Masters Degree Building Engineering
Specialization Structural Design

PREFACE

The interest for structural design and computational design started during the course CT5251 Special Structures and the design-project AR0651 XXL Design+Engineering.
The combination of *special* shapes, structures and the digital world forms the basis for the decision to choose the field of computational design for the graduation project.

The final results of the Master's thesis is presented in two parts:
1. The main report, explaining all theories and information regarding the research;
2. The appendices, including the user manual.

This main report consists of:
- The introduction to the problem and a presentation of the motivation for this research (Chapter 1);
- The formulation of the research question and the topics that have been researched (Chapter 2);
- The results of and information gathered from the literature study (Chapter 3);
- The theories and methods used in the research: the Thrust Network Analysis and Catmull Rom splines (Chapter 4);
- Information about the new application, such as the user interface, procedures and options (Chapter 5);
- Aspects concerning brick in building practice (Chapter 6);
- Discussion, conclusions and recommendations (Chapter 7).

Chapter 8 contains the references.

The user manual of the application, more detailed explanation of relevant theories and other informative documents are listed in the second report 'Appendices'.

Before presenting the results, I would like to thank the members of my graduation committee for their time, assistance and for the knowledge they shared with me. During the research process and the consultations they were of great help to me, in advising, guiding and commenting.

Tom van Swinderen, August 2009

TABLE OF CONTENTS

ABSTRACT

INTRODUCTION

The design of a double-curved shell is time-consuming, because the shapes are often complex, the architect is very specific in the desired shape and the engineer has to perform an elaborate analysis. To decrease the time and increase the quality of the process of making a conceptual design, the communication-process between architect and engineer has to be improved.
Philippe Block and John Ochsendorf, active at the Masonry Research department of the M.I.T. [25], have proposed a new theory to analyse shells with, the Thrust Network Analysis (TNA) [10].
In this research an interactive design tool has been developed, with which masonry shells can be rapidly designed and analysed. The new tool uses the TNA theory, which is based on three-dimensional equilibrium of force networks.
After designing and analysing the conceptual shape of the shell, it has to be materialised. Following examples by Eladio Dieste, whose designs are the inspiration to use brick as building material, the tool is able to generate masonry patterns and takes into account manufacturability constraints for this material.

THEORIES

Thrust Network Analysis

The most relevant theory is the Thrust Network Analysis, which makes use of force polygons and the reciprocal relationship. This analysis performs a check of the force flow. Stability checks, such as buckling behaviour and displacements, are not regarded.

Force polygons

When the sides of the polygon are representations of forces in a network, the resulting polygon is regarded as a force polygon. The lengths of the sides represent the magnitude of the corresponding force. If the polygon is closed, the body on which these forces act is in equilibrium.

Reciprocal relationship

The relation between the primal and dual grid can be described as lines connected to a node in the dual grid form a polygon of forces in the primal grid and vice versa. This two-way relation is known as a reciprocal relationship. The mechanical property of reciprocal diagrams is expressed in the following theorem by Maxwell [8]:
'If forces represented in magnitude by the lines of a figure be made to act between the extremities of the corresponding lines of the reciprocal figure, then the points of the reciprocal figure will all be in equilibrium under the action of these forces.' (Figure 0.1)

Catmull Rom splines

To generate the shell surface, create the masonry pattern and make the application interactive the theory of Catmull Rom splines has been used. The theory creates curves, using four points and the angle of the curve in these points. The curve passes through all points. The factor τ determines the curvature of the line. The curvature of the masonry pattern is limited by the allowed angles between the bricks. To make sure the range of possible shapes is as big as possible, the factor τ has been set lower than the commonly used 0.5. In the current prototype it is 0.2.

THE TOOL

It has been created in Processing [24], an open source programming language and environment.
The new application consists of four main steps:
(1) Setup of the initial force network model;
(2) Analysis and design of the network model;
(3) Generate the masonry pattern, which can either be a linear pattern or a spherical pattern;
(4) Export the final model.

The initial force network model setup in step 1 has two options:
(a) Create a parameter model;
(b) Import a model and approximate it with a parameter model.

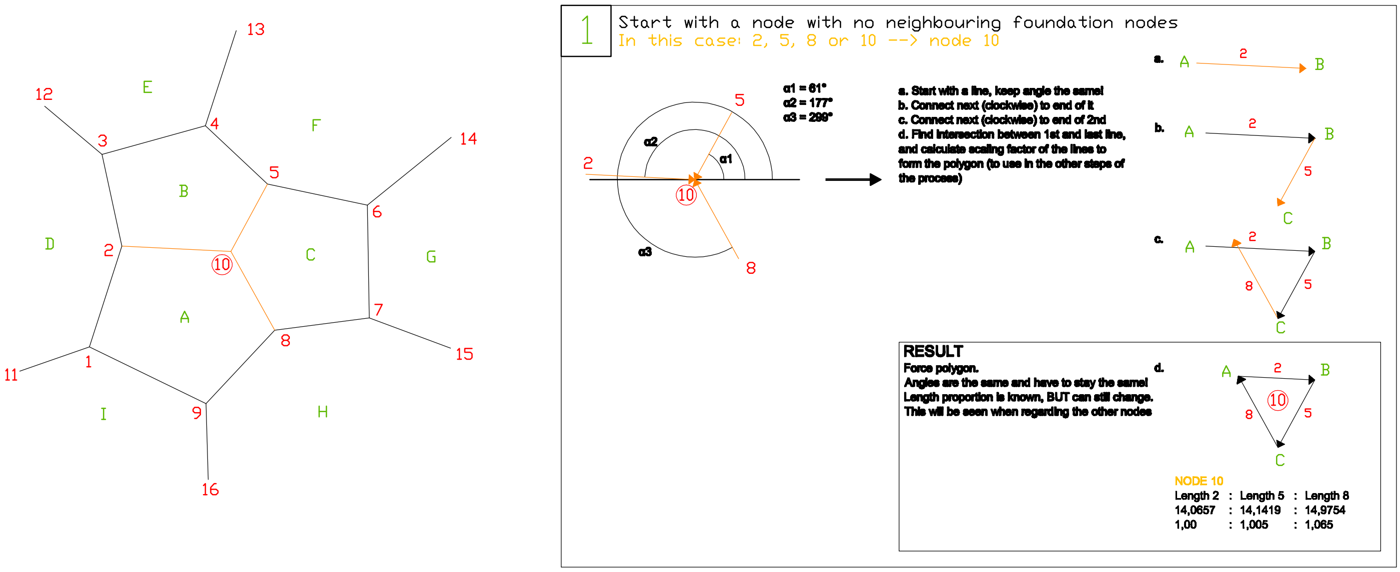


Figure 0.1 *- Example of a reciprocal relationship. Left is the network shown (primal grid) en at the right is shown how the reciprocal figure of node 10 is determined.*

The interface is designed to be user friendly (Figure 0.2, top picture). At the top of the interface the four tabs are located, that represent the four steps.

Several layers are created when the application is used, so that the results are available for all actors that are active in the continuation of the design process. For the structural engineer the force network model layer is of importance, for the architect the surface models and the contractor might be interested in the masonry pattern layer (Figure 0.2).

Companies use different software programs; Rhinoceros, Maya, 3DMax, AutoCAD and SketchUP are most commonly used. To assure widespread use the tool supports a wide range of export formats, such as:

- .dxf (for AutoCAD and 3DMax);
- .rvb (for Rhinoceros);
- .mel (for Maya);
- .rb (for SketchUP).

DISCUSSION

One of the main goals was to create an interactive tool, with which the architect and engineer can rapidly generate a conceptual shape for a masonry shell. Using the theory of TNA in combination with Catmull Rom splines has provided the right conditions to design a first prototype for this tool.

The tool performs a force flow analysis of a shell shape. The stability, such as buckling behaviour, and the displacements are not considered.

To improve the results and increase the range of possible shapes to analyse, research should be done into expanding the TNA to make it suitable for tensile forces and into designing the tool to let it form find the force network and masonry pattern automatically, instead of manually by the user.

Finally the script code must be looked at by a professional tool developer, so that the performance of the application is improved.

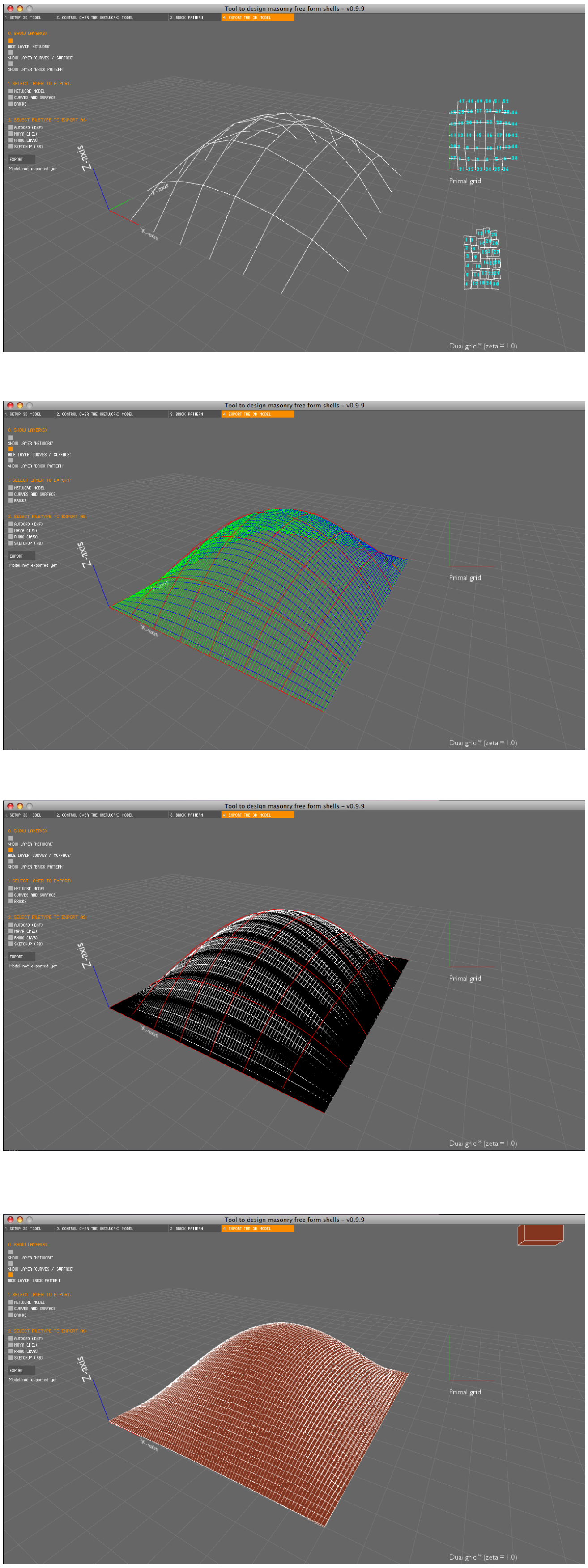
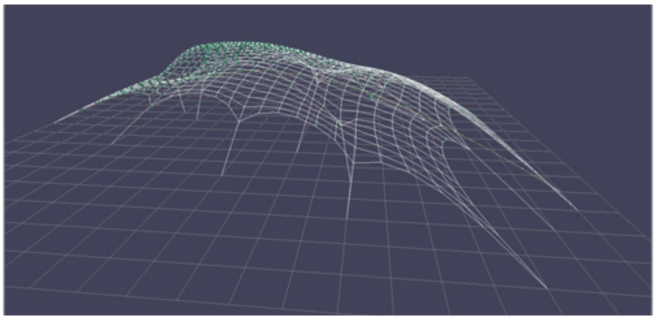


Figure 0.2 - The four layers created by the application. From top to bottom: the force network, the lines-surface, the polygon-surface and the brick pattern.

Figure 1.1 *- CADenary, form finding project developed in Processing by Axel Kilian [18], using a Particle Spring library developed by Simon Greenwold (Image courtesy of Axel Killian)*



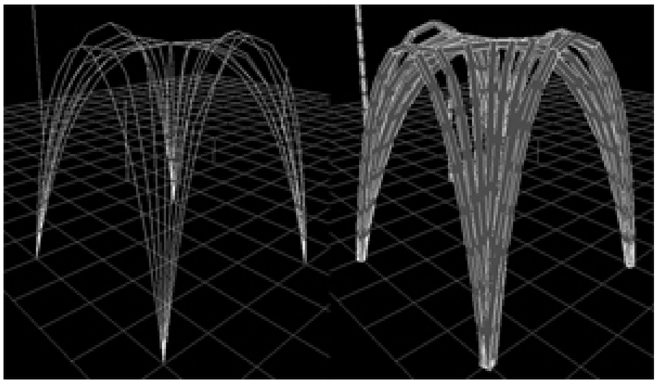
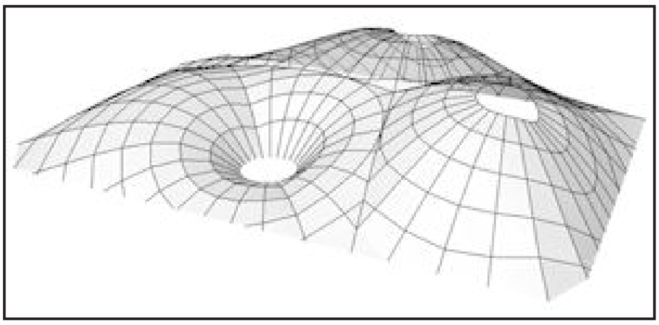
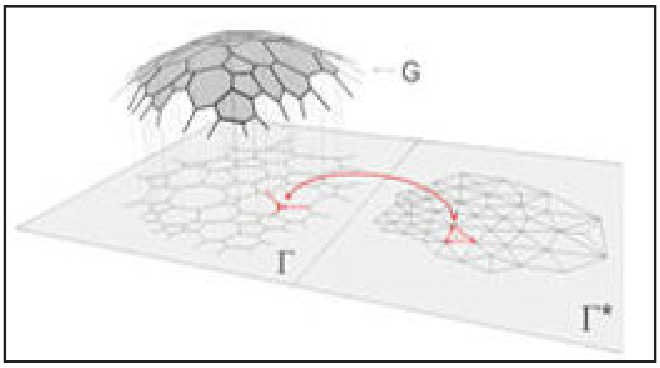


Figure 1.2 *-The theories and results of research into masonry by Philippe Block and John Ochsendorf [18] (Image courtesy of Philippe Block)*





CHAPTER 1 INTRODUCTION

The building design process is a long and complex process and many parties are active in it. One of the first phases is the conceptual design stage. Two of all participating actors during this stage are the structural engineer and the architect.
Recently the use of the computer in the design process increased significantly; both in the field of the architectural design as in the field of engineering.
In some situations these processes are combined; for instance when the design has a complex shape or when the geometry of the surface is complicated. In these cases the process of form finding is based on architectural design ánd structural analysis.

The increase in use of the computer during the design process is observed in the work field of both the structural engineer and the architect. The engineer performs finite element calculations with the computer, while the architect uses the computer to design complex shapes – often resulting in designs referred to as free form – and create renderings and digital models.
The traditional, straight forward designs, such as factory halls and houses, are functional and useful for application in residential and industrial areas. But due to their everyday appearance they tend to be less interesting for more expressive functions and fields of architecture.
Moreover buildings become higher (highrise), spans bigger (column free space) and façades more complex (challenging appearance); all to satisfy the wishes of the project initiator and to comply with the rules and need for sustainability.

However designing free form structures creates a challenge for both the architect and the structural engineer. The engineer designs structures using basic structural mechanics and sometimes with the help of physical models.
Due to the increasing complexity the engineer needs to use finite element methods and algorithms to calculate the complex and non-standard elements and structures, such as double curved surfaces.
As a result the communication with the architect tends to be more intense and proves to be more difficult and time consuming: the methods and algorithms need to be designed, tested and finally applied.
The resulting shape and structure might not satisfy the architect. As a result the design and model have to be changed and the structure has to be adapted to this new design. This whole process of optimising may take a long time.

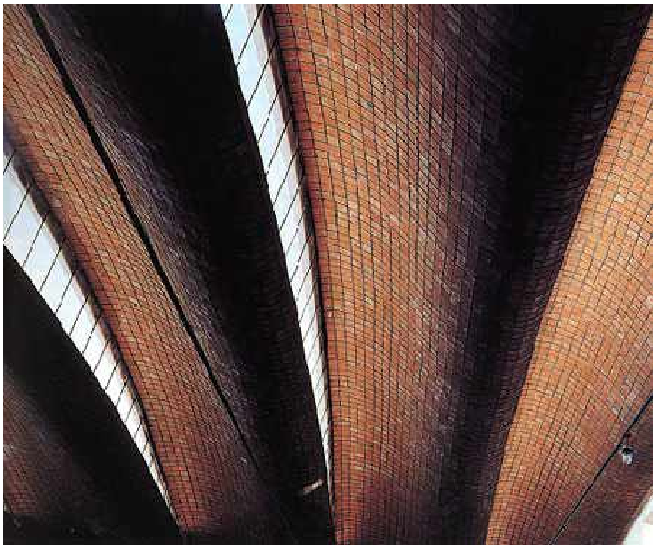
Computational design

The computer technology and capacity increased significantly last several decades. Therefore the computer is also used more often as a tool in specific phases of the design process, which is referred to as computational design.
Technological development and research into the use and abilities of materials create a wider range of possible shapes, designs and structures and make it possible to optimise designs even more. One of the results is the tendency of structures becoming more slender and at the same time more complex.
To decrease the time of the design process the communication between the architect and the structural engineer has to be faster, smoother and better.
One of the possible solutions is an application which performs the analysis faster and makes it possible to adapt and optimise the model. This way the process of designing the conceptual model is reduced to one meeting between the architect and engineer.
Several of these digital applications for free form design are already developed. These applications are mainly based on physical models or material characteristics. Examples are CADenary, the hanging model program of Axel Killian [18] (Figure 1.1) and the research into masonry structures and applications of John Ochsendorf and Philippe Block [18] (Figure 1.2).

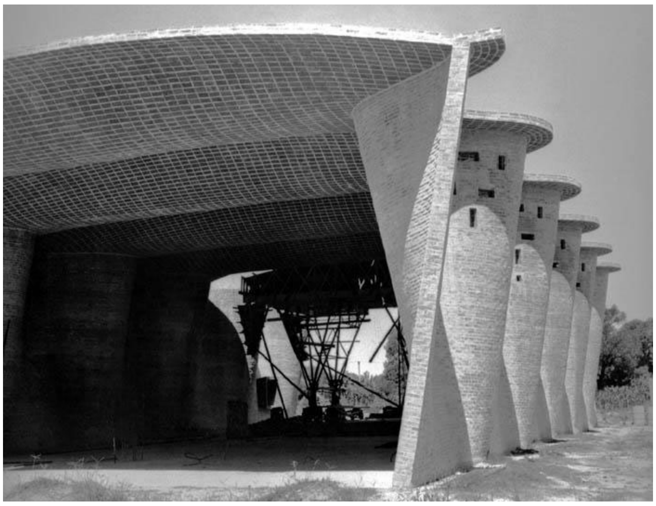
Brick

An important design aspect is the building material used for the structure. Concrete, steel and timber are commonly used

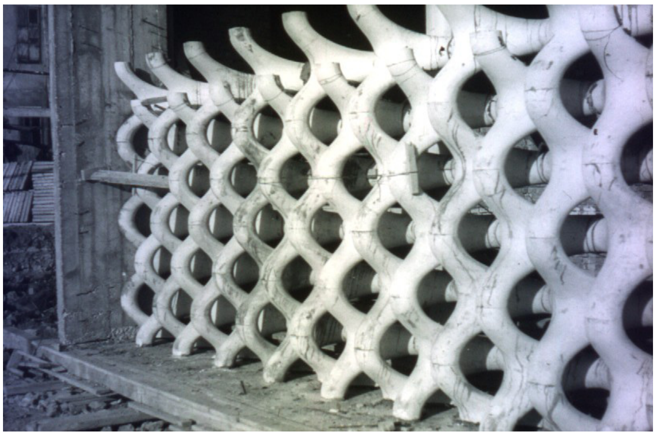
Figure 1.3 *- Several brick structures by Eladio Dieste*



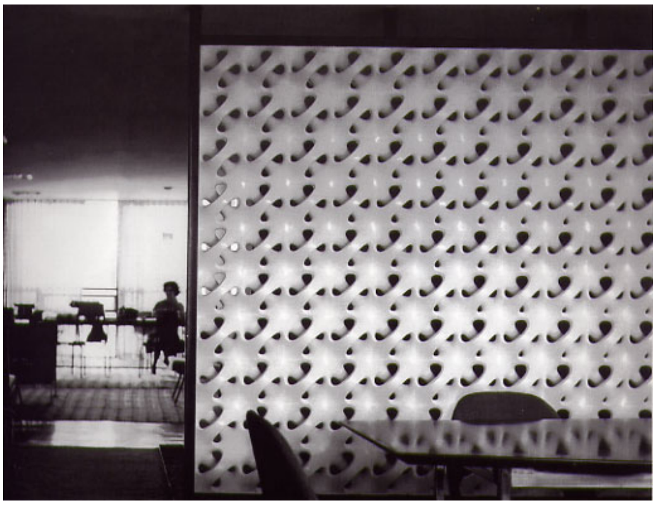
(a) *- Roof of 'Julio Herrera & Obes Warehouses', Montevideo, Uruguay. Completed in 1979.*



(b) *- 'Church of Christ the Worker', Atlantida (close to Montevideo), Uruguay. This is a photo during construction. It was completed in 1960.*



(a) *- 'Light-diffusing wall design 3 in church in Liesing', Vienna, Austria, 1952.*



(b) *- Example used in the 'Showroom of Knoll Internacional de Mexico', Mexico City, 1950.*

Figure 1.4 *- Concrete pattern designs by Erwin Hauer*

(a) *- A thin concrete shell. Service station in Deitingen-Süd, Switzerland, designed by Heinz Isler.*

(b) *- The Eden Project, Cornwall, UK. Steel grid shells by Nicholas Grimshaw.*

Figure 1.5 *- Example of a shell shape/structure:*

as material for the structural elements for most structures and buildings.

In several parts of the world, among others in the Netherlands, another material is also used and applied regularly: brick. It is used in dwellings and as facade material, although mainly in vertical elements, such as walls. A lot of knowledge has been gained by the building industry about brick and how to apply it in practice [17].

In general bricks are made out of clay. It is a natural material and therefore considered to be a sustainable material. After demolition it can be re-used as filling in for instance concrete. This is a positive characteristic of brick, taking into account all recent climate problems.

Brick performs well in building physics; the thermal insulation is good. Another advantage of brick is the freedom it gives for a structure to be built out of small elements. Theoretically all these elements can have a different size, colour and surface. However up till now brick is mainly being used in straight, vertical walls. The question raises why brick is almost never considered as a material for double curved structures, such as shells.

In other parts of the world, for instance in Uruguay, the Spanish architect and engineer Eladio Dieste created double curved structures, with a very slender appearance (Figure 1.3). Besides their beautiful appearance another remarkable aspect is the time in which they were built: between the 1950's and 1980's, long before the era of the introduction of the computer in the design process. The techniques he used are therefore interesting for further research on how to use them in the present time [2].

Architectural wishes

As mentioned before, the communication with the architect is a requirement to assure a smooth and optimised design process. Not only the shape of the building is designed by the architect; the appearance of the structure, such as the façade, is part of the design as well. Therefore the possibility to design and adapt the pattern of brick stones is a relevant research aspect to implement in the application. It offers the ability to control the appearance of the structure.

An example is the work of the Austrian designer Erwin Hauer. He created patterns and concrete elements by varying the size, position, rotation and repetition of one element (Figure 1.4).

GOAL

The aim is to design a tool that offers the possibility to enhance the design process during the conceptual design phase of a masonry double-curved shape. The chosen shape is the shell (Figure 1.5), which has a curved shape and is used as for instance covering structure of a space.

The tool assures a better communication between architect and engineer, so that the design of the model is finalized timely.

With the application the user is able to design concepts for masonry double-curved shells. The tool offers the ability to interactively adapt and finalize this model according to the wishes of the user.

The tool generates masonry patterns according to certain specified variables and parameters.

Interactivity of the application is the key aspect to include. The architect and engineer have to be able to work together and adapt the shape and get results fast. This enhances the chances of a rapid process of finding and determining the conceptual shape.

Parametric design of the pattern

The brick shapes are placed according to several desired characteristics for the structure, which are symbolized by a number of variables and parameters:

- Shape and size of the brick;
- Dimensions of the structure.

Final result

The solution is an application consisting of three parts:

- First of all it rapidly analyses the model. The input is either a model based on user-specified parameters, or an architectural shape, which is approximated with a parameter model;
- Secondly the application is interactive: the shape is manually adaptable and the application instantly performs the analysis and displays the new result;
- And finally the application will generate a masonry pattern.

The application and the result of the analysis are presented in user-friendly interface.

CHAPTER 2 RESEARCH QUESTION AND TOPICS

Introduction

In this chapter the research question (§ 2.1) and related research topics (§ 2.2) are presented.
The research question is the basis of the research. This question is subdivided into several topics, that are each used to find the answer of an aspect related to the research question. When the answer of all topics are combined, the main question should be answered to a satisfactory level.

2.1 Research question

The main title of the thesis is:
'Proposal for a tool to design masonry double-curved shells'

To clarify and describe the research completely, the title is extended with:
'To analyse concepts and generate a brick pattern'

The research question accompanying the title is formulated as:
'Is it possible to improve the speed and quality of the conceptual phase of the design process with an interactive tool, offering the ability to design and analyse masonry double-curved shells?'

To guarantee a good final result, the research question is divided into several topics. These topics are presented in the next paragraph (§ 2.2).

2.2 Research topics

An essential term in the research question is *masonry shells*. The tool is created to design these type of structures. This is only possible when information is available regarding masonry (and brick) and regarding shells.
Next step is to integrate this information in a new application. For this purpose information is required regarding the design of a software application. Moreover certain theories and algorithms are needed to implement the information about brick and shells and to make a well performing and interactive application.
The last step is to make sure the results of the application can be applied in practice. Therefore aspects such as building a brick structure and new technologies in the brick industry are looked at. The results, together with the general information about brick, are used as checks and limits for the analysis and result of the application.

In the end the following five topics have been assigned as topics for research:

1. a. The material brick;
   b. Shell structures;
2. Design of a new software application;
3. Theories and algorithms;
4. Brick structures in building practice.

Topic 1. Brick and shell structures

(a) The material brick

This topic is researched during the literature study.
Included aspects are:

- Characteristics of brick, such as strength and dimensions;
- A comparison and advantages of brick over other materials;
- Possibilities in regard to non-standard brick shapes.

(b) Shell structures

This topic is researched during the literature study.
Included aspects are:

- Force distribution systems, with the emphasis on form active structures;
- Structural mechanics of shells, such as geometry and curvature.

A Spanish architect and engineer, Eladio Dieste, designed structures in which topics (a) and (b) are combined. Therefore his work is researched as well.
The useful information regarding this topic is presented in Chapter 3.

Topic 2. Design of a new software application

This topic is investigated during the research and design stage. Relevant aspects are:

- Software related aspects, such as the ability to import and export models in certain file formats;
- The transformation of the applied theories (Chapter 4) into scripts to be able to use in the application (Topic 3);
- Design of the user interface:
  The outcome and results of the application have to be understandable and shown in a clear way. This assures a proper handling of the application during a meeting between the engineer and architect. As a result the process of form finding and obtaining a conceptual design should be better and faster.

The results are shown in Chapter 5.

Topic 3. Analysis methods and algorithms

This topic is investigated during the research and design stage. Relevant theories and methods are:

- A theory to calculate and analyse the models and designs;
- A method to generate brick patterns.

The results are shown in Chapters 4 and 5.

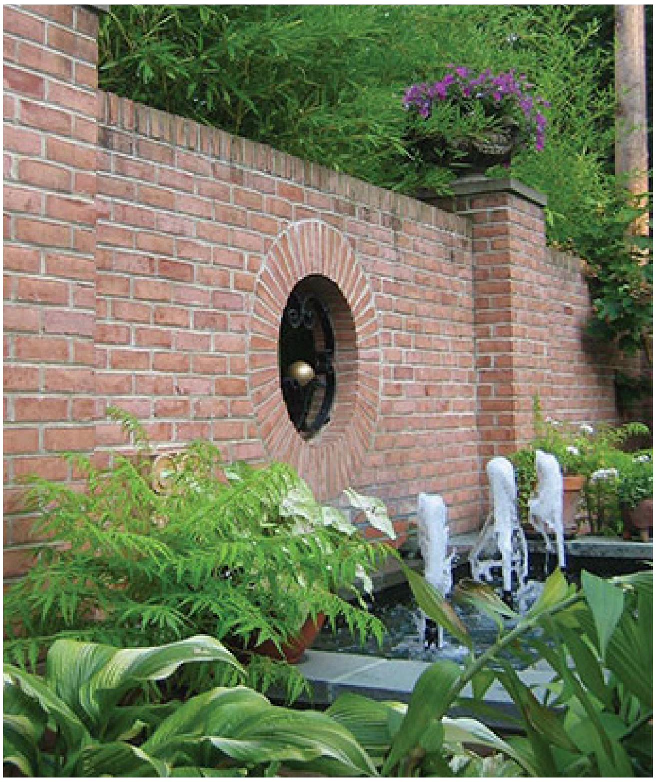
Topic 4. Brick structures in building practice

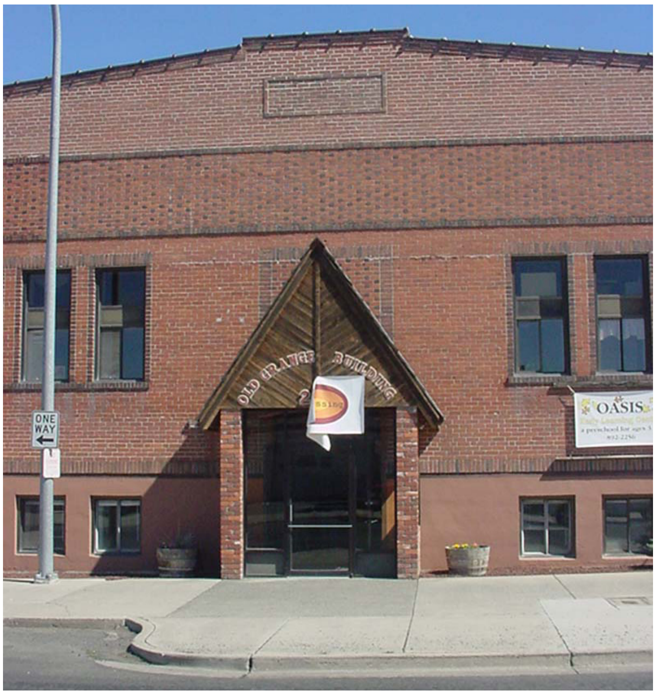
This topic is looked upon during every stage of the final thesis, because it is important to make sure the final design is realistic and buildable. Related aspects are:

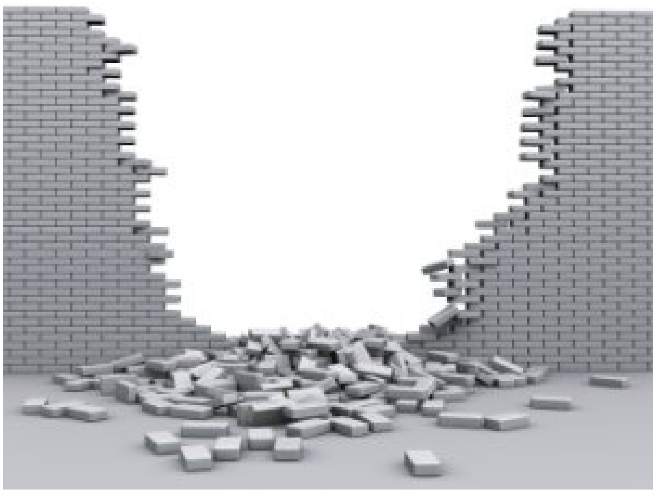
- Brick fabrication and production;
- Building physics;
- Constructing the structure:
  - Prefab elements, connected on the building site;
  - All in situ.

The results are shown in Chapter 6, together with several other aspects that were encountered during the research process.

***Figure 3.1** - Examples of brick structures*







CHAPTER 3 INFORMATION FROM THE LITERATURE STUDY

Introduction

In order to have an adequate level of information and knowledge regarding the topics involved with this research, a literature study has been performed during the first phase of the research.

Several topics have been researched:
- Brick structures;
- Shell structures;
- Free form design;
- Typology of structures;
- Form active structures;
- Computational design;
- Geometry descriptions;
- Analysis methods;
- The building practice.

The information of these topics that was considered useful in the continuation of the research is presented in this chapter.

Following the reasoning as presented in the Chapter 2 (§ 2.2) the main topics are:
- The material brick (§ 3.1);
- Eladio Dieste and his masonry structures (§ 3.2);
- Form active structures (§ 3.3);
- Shell structures (§ 3.4);
- Methods to analyse shell structures (§ 3.5);
- Computational design tools (§ 3.6).

In the end a list of the decisions, assumptions and limitations that have effect on the outcome of the research is given (§ 3.7).

3.1 The building material brick

In this paragraph information regarding the building material brick and the masonry structures made with it is presented. It is divided into subparagraphs:
- General introduction of brick, including advantages and disadvantages of it (§ 3.1.1);
- Types and dimensions of brick (§ 3.1.2);
- Structural information, such as strength and bonding properties (§ 3.1.3);
- Brick in free form designs (§ 3.1.4).

3.1.1 Introduction to brick

In Figure 3.1 some pictures of brick and brick structures are shown. Brick performs best when it is subjected to compression force, because the tensile strength of brick is very low.

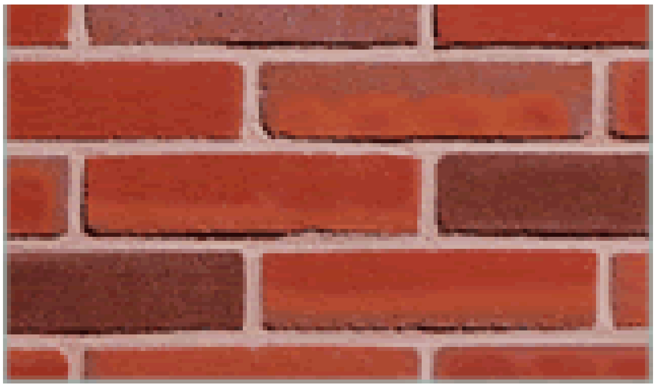
Vertical, straight elements, such as walls for houses, warehouses, factories and garden barriers, are mainly subjected to vertical loading (assuming the horizontal loading, such as wind, is transferred using the floor system or an other structure). Therefore brick is in general only used in these type of elements.

An exception are arches, although this type of structure is regarded as two dimensional as well (height and length) and can therefore be regarded as a flat plane as well.

When bricks are combined into a structural element (such as a wall) a bonding layer is used between them. The combination of bricks and bonding is referred to as *masonry* [18].

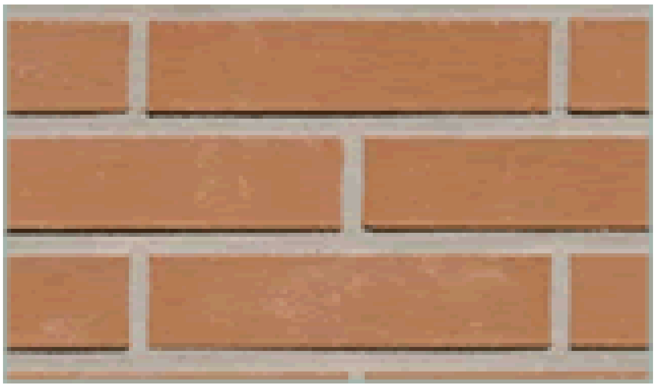
<u>ADVANTAGES</u>
- The use of materials such as brick and stone increases the thermal mass of a building, giving increased comfort in the heat of summer and the cold of winter;
- In general brick will not require painting and so can provide a structure with reduced life-cycle costs. Nevertheless an appropriate sealing of brick will reduce potential burst and failure due to frost damage;

Figure 3.2 *- Brick types*



Box shaped (Vormbak)

Handshaped (Handvorm)



Cord press (Strengpers)

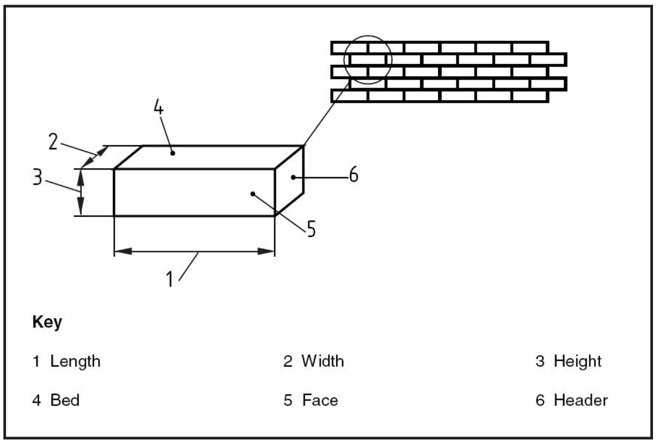


Figure 3.3 *- Dimensions and variables of brick*

- The appearance can, when well designed, create an impression of stability and durability;
- Brick is very heat resistant and thus will provide good fire protection.

DISADVANTAGES

- The costs for maintenance of masonry are higher than for steel, timber and concrete structures. To replace one brick is almost impossible and therefore it is difficult to repair damage to the bonding or to any brick. Moreover masonry is more fragile for damages than other materials, since masonry structures consist of many elements and as a result many connections that can be damaged, while for instance steel grids are made out of several big elements and a concrete shell can even be considered as one big element;
- Extreme weather may cause degradation of the surface due to frost damage. If clay-based brick is used, care should be taken to select bricks suitable for the climate in question;
- Masonry must be built upon a firm foundation (usually reinforced concrete) to avoid potential settling and cracking. If expansive soils (such as clay) are present, this foundation may need to be quite elaborate;
- The high self weight of brick increases structural requirements, especially in areas regularly subjected to earthquakes.

3.1.2 Types and dimensions of brick

TYPES

The three main types of brick that are used in the brick industry are [16] (Figure 3.2):

- *Box shaped* (in Dutch: vormbak)
  A sanded, rather smooth stone with a regular surface, of which one of the flat sides has been flattened by striking.
- *Hand shaped* (in Dutch: handvorm)
  Similar to the box shaped stone, but the form is less straight and the surface is more irregular.
- *Cord press* (in Dutch: strengpers)
  A smooth, sometimes perforated stone, with cut flat sides and with surfaces that are varying between flat till sanded and smooth till very rough.

DIMENSIONS

The dimensions of bricks are standardized [18]. An internationally acknowledged modular size has been set (Table 3.1 and Figure 3.3).

Due to technological developments and the constantly increasing demands and requests from costumers and project initiators, the demand for uncommon and differentiating structures has increased. These uncommon forms are constructed with non-standard dimensions and shapes for the brick.

Moduulformaat	**190 x 90 x 50/65/90**
Waalformaat	210 x 100 x 50
Vechtformaat	210 x 100 x 40
Dikformaat	210 x 100 x 65
Rijnformaat	181 x 87 x 41
IJsselformaat	160 x 78 x 41
Kloostermop	280 x 105 x 80
Euroformaat	240 x 100 x 69

Table 3.1 *- Standard dimensions brickstones*

Generally the solution to create an *uncommon* structure with brick is found in the appearance of the stone – the texture – and by the use of non-standard shapes. In combination with different colored stones this gives a lot of possibilities regarding the appearance of a building or structure.

Besides using brick in designs for industrial and commercial buildings, architects also want to give private housing a more unique and varying appearance by using shape variations of the brick.

Some information regarding non-standard brick-shapes is found in Chapter 6.

TEXTURE

The texture of bricks is mainly determined by the method of production. Alternatively the steps during the process of giving the final shape to the stone and the post-processing gives the brick a special texture.

Brick buildings in Dhaka, Bangladesh. Design by Louis Kahn.

Crawford Municipal Art Gallery, Cork, Ireland, 2000. Design by Erick van Egeraat Architects.

Figure 3.4 *- Examples of free form designs in combination with masonry.*

3.1.3 Structural information

LIMIT STATE ANALYSIS OF MASONRY

The limit state analysis of masonry assumes the following [1]:

1. Masonry, particularly the mortar joints in masonry, has no tensile strength;
2. Compressive stress levels from loads applied to the structure are low relatively to the maximum allowed compressive (crushing) stress of masonry. Therefore material strength properties are not likely to be determining in failure analysis. However the active stresses are one of the checks for the analysis as performed in this research (§ 4.4);
3. Sliding failure does not occur.

BONDING

There are rules of bonding, which have some exceptions.[2] These specify the overlap between courses that is visible outside the wall, and also the overlap which must be made within the wall, for walls which are more than half a brick thick.

The maximum width of the bonding layer in this research has been set to 16 mm.

COMPRESSIVE STRENGTH

Masonry has a high compressive strength, but is much lower in tensile strength (twisting or stretching) unless it is reinforced.

Brickwork arches can span great distances and carry considerable loads.

The compressive strength of brick ranges from 7 - 140 N/mm^2, depending on the type of brick and function it is used for.[3] For load bearing walls in residence buildings the strength is set to 25 - 30 N/mm^2. These values are used in this research to check the active stresses in the structure.

TENSILE STRENGTH

Brickwork, like unreinforced concrete, has little tensile strength and therefore performs best when the whole structure is in compression. Where required, steel reinforcement can be introduced to increase the tensile strength.

For this research it is assumed brick can not take any tensile forces and that no reinforcement is applied. Therefore the tensile strength has been neglected and is set to 0 N/mm^2.

3.1.4 Brick in double-curved designs

An example of the possibilities of masonry in double curved structures are the designs of Eladio Dieste [2]. His methods to analyse and design masonry structures are remarkable and a good source of information for this research (§ 3.2).

Another engineer well known for his brick and stone structures is Louis Kahn (Figure 3.4, top picture) [9]. However his work is not as *slender and curved* as the work of Eladio Dieste. Therefore his work is of less interest for this research and will not be mentioned further on.

1 - Heyman, J. (1995). *The Stone Skeleton.* Cambridge: Cambridge University Press.

2 - http://www.knb-baksteen.nl/publicaties/publicatie_62.htm [19]

3 - http://www.knb-baksteen.nl/publicaties/publicatie_61.htm [19]

Figure 3.5 - *Brick structures, designed and engineered by Eladio Dieste*

(a) - *'Church of Christ the Worker', Atlantida, Uruguay, 1960.*

(b) - *Casa Dieste, Montevideo, Uruguay, 1961.*

(c) - *'Monumento homenaje en rotonda', Salto, Uruguay, 1976.*

(d) - *'Chacineria Fenix', Salto, Uruguay, 1978.*

(e) - *'Iglesia de San Pedro', Durazno, Uruguay, 1971.*

(f) - *Roof of 'Julio Herrera & Obes Warehouses', Montevideo, Uruguay. Completed in 1979.*

3.2 The masonry structures of Eladio Dieste

"The resistant virtues of the structure that we make depend on their form; it is through their form that they are stable and not because of an awkward accumulation of materials. There is nothing more noble and elegant from an intellectual viewpoint than this; resistance through form" [4]

The work of Eladio Dieste is the inspiration for this research. Double-curved structures are commonly constructed with steel, timber or concrete. However Dieste used the material brick, reinforced with steel, and the result is remarkable (Figure 3.5).

Eladio Dieste (1917-2000), both architect ánd engineer, was born in Spain. However most of his life he has lived and worked in Uruguay. He is famous for his (reinforced) brick construction techniques.
His work is based on the catenary form of the arch. He relies on the great strength of brick in compression and modifies the geometry to improve stability against buckling. This results in structures that behave as traditional vaults, but have a lightness that defies tradition and *"positions them firmly in the 20th century"*. [5]
In buildings which are enclosed, Dieste often refuses to use the walls to support the roof. He did not use models for his designs, unlike for instance Frei Otto.
A particular innovation was his Gaussian vault (§ 3.2.2), a thin-shell structure for roofs in single-thickness brick that derives its stiffness and strength from a double curvature catenary arch form that resists buckling failure. His buildings were mostly roofed with thin shell vaults constructed of brick and ceramic tiles. These forms were cheaper than reinforced concrete and did not require ribs and beams.

Several essays, which embrace the technical and philosophical aspects of his work, were written by Torrecillas (1997). In these essays explanation of deflection and stresses in double curved vaults are found.[6]

4 - Quote by Eladio Dieste in *Arquitectura y Construction* (Architecture and Construction), Torrecillas, 1997.
5 - Quote by Pedreschi in *The Engineer's Contribution to Contemporary Architecture* [2].
6 - *Pandeo de Laminas de Doble Curvatura* (Deflection in double-curvature vaults). English version: "Eladio Dieste 1943-1996, Calculation Methods"

ADVANTAGES OF REINFORCED BRICK STRUCTURES

- Brick is lighter in weight than concrete, reducing the cost of the supporting structure or foundation;
- Shorter construction period in comparison with concrete, because there is no need for hardening of the brick and only a short time of hardening of the bonding layer;
- Brick has good environmental properties, its hygroscopic nature helps to control humidity;
- Brickwork is easier to shape into double curvature forms, since the material does not have to be deformed. However form work is needed, just as it is needed for in situ concrete. Steel grids are also not complex to shape, but the grid has a bigger mesh than that of a masonry structure, since the masonry has a grid 'of bricks', while the steel structure has a grid of big planes between the steel grid.

Two generic forms of vaulted reinforced brick structures, that Dieste is well known for, are:

1. The free standing barrel vault (§ 3.2.1);
2. The Gaussian vaults (§ 3.2.2).

3.2.1 Free standing barrel vault

Dieste describes this vault as *"free-standing catenary shells without tympanums."* [7] (Figure 3.5 - c and d).
In most of the buildings the structure underneath is minimal in order to leave the roof floating, hovering above the floor below.

The barrel vault in one form can be seen as an extension of the arch, as a series of connected arches running along a line of supporting walls. These walls provide both horizontal and vertical reaction.
When the longitudinal span exceeds three times the transverse span, the dominant structural action is bending. But Dieste's vaults have a much higher longitudinal to transverse span ratio: they are designed to allow both arch action and bending action to develop.
Dieste has developed the mathematical theory to calculate the additional stresses in the transverse section of the roof.

7 - Quote by Dieste in *The Engineer's Contribution to Contemporary Architecture* [2], page 28.

Figure 3.6 *- Example of a Gaussian vault by Eladio Dieste*

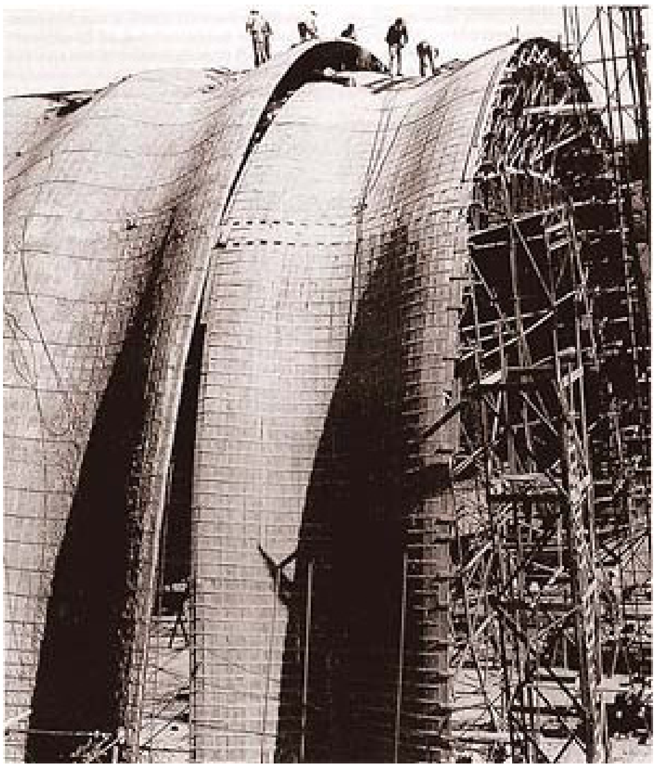
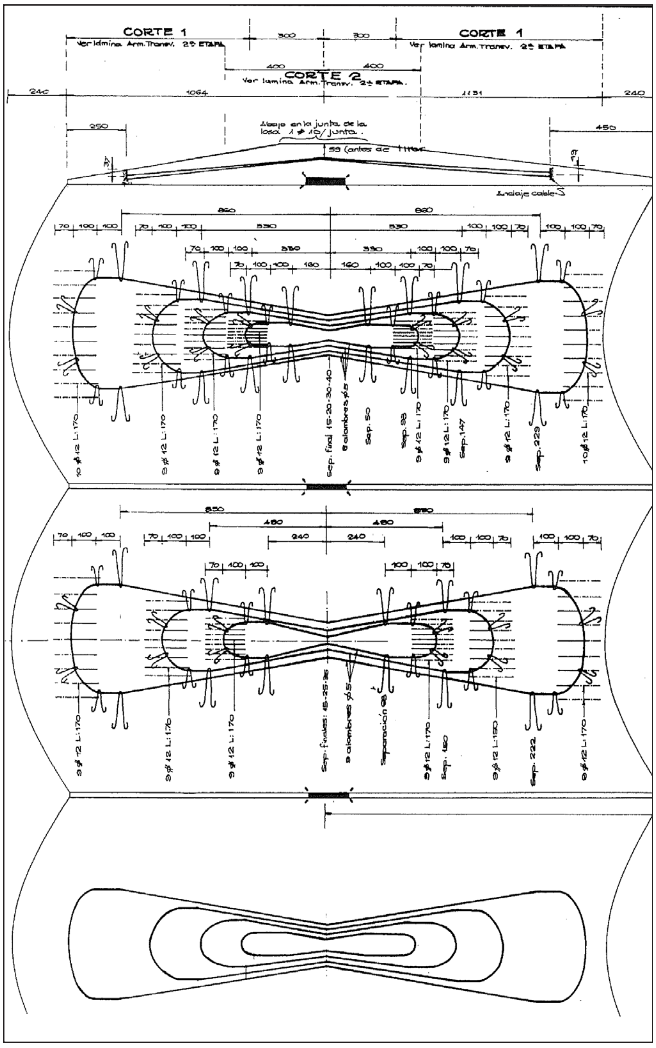
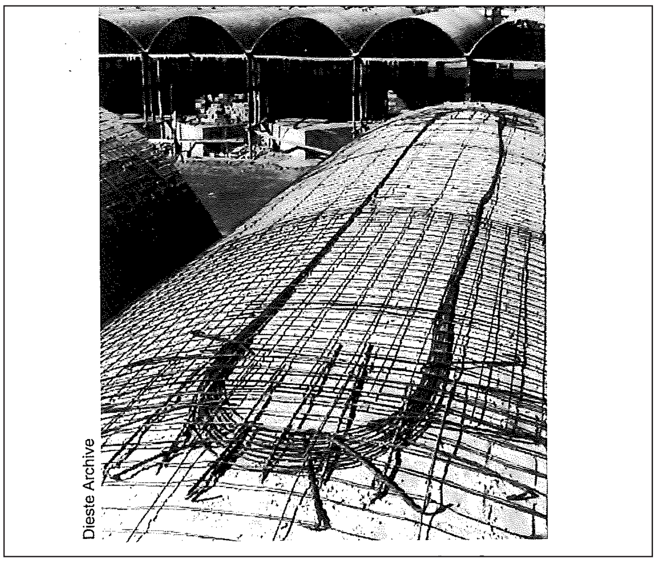


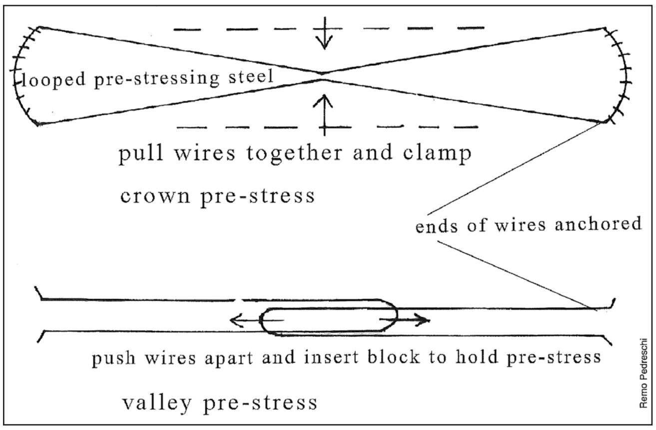
Figure 3.7 *- Reinforcement in the designs of Eladio Dieste*



(a) *- Crown reinforcement scheme, seen from the top (Image courtesy of R. Pedreschi [2]).*



(b) *- Crown reinforcement installed after the bricks have been placed, but before they are tensioned in the middle. After the tensioning a covering layer is poured over the reinforcement too protect it (Image courtesy of R. Pedreschi [2]).*



(c) *- The reinforcement is tensioned by pulling the wires towards each other in the middle of the span and anchoring them (Image courtesy of R. Pedreschi [2]).*

The technique enables the vault to be reinforced to resist these secondary stresses.[8]

The form of the vault and its ability to cantilever long distances, as used in Massaro, make it suitable for large canopy-type structures, providing shelter rather than enclosure.

REINFORCEMENT AND PRESTRESSING

Crown (top of the vault) reinforcement

Consists of looped prestressing wires (Figure 3.7). They are placed on the top of the vault once the bricks have been installed and the reinforcement between the bricks grouted in place. Each end of the loop is embedded in reinforced anchorages, tied with steel rods to the vaults. The central part of the loop remains free and rests on top of the vault. The distance between the two sides of the loop is critical. Once the anchorages have sufficient strength, the loops are pinched together at the middle point, causing them to stretch and generate inward reactions at the anchor points, pre-compressing the vault. The top of the vault is then covered with a light layer of concrete to cover and protect the cables.

Valley reinforcement

Consists of two overlapping loops. The ends of each loop are anchored into concrete and tied in the vault. A specially developed jack is placed between the overlapping cables that pushes the ends of the loop apart, stretching the cables. Once the required extension has been reached, a steel block is placed between the two loops to maintain the separation of the cables. When the jack is relaxed, the cables tighten onto the steel block, locking in the prestress force.

3.2.2 Gaussian vaults

The Gaussian vault (Figure 3.6) has evolved out of the barrel vault, extending the use of the catenary to shallower and longer spanning vaults. The shape is also used by Torroja and Candela. It is mainly used for large single-story sheds, used as warehouses, gymnasia and workshops. There are two generic variations of the Gaussian vault, thought they share similar geometric and structural roots: the long span shallow vaulted roofs and the tall curved shells.

Long span shallow vaulted roofs

Normally supported on a concrete frame or load-bearing walls.

The main structural problem is not the axial stresses themselves, but the thrust induced in such a slender structure, leading to a tendency to failure by buckling.

Solutions to prevent buckling due to own weight and too low span-thickness ratio:

- Reduce the compressive stresses, by increasing the rise of the vault;
- Increase the cross-section to make the vault stiffer. Best is to add arched ribs.

Dieste used both methods, but in a unexpected way: he used hollow clay blocks to reduce weight (to approximately two thirds of an equivalent solid concrete vault) and the shape of the structure is manipulated to provide increased resistance to buckling, without increasing the thickness. This shape can be described using a family of catenary curves of varying rises.

Tall curved shells

These structures are used for large horizontal storage silos (for grain and other bulk materials). The rise of a silo is one half of the span.

8 - Dieste, E., *Cascaras autoportantes de directriz catenaria sin timpanos* (Free-standing vaults of catenary directrix without tympanum)

(a) *- Indoor Tennis Center, Heimberg, Switzerland, 1978. A shin shell structure, designed by Heinz Isler.*

(b) *- Roof of the Olympic Stadium, Munchen, Germany, 1972. A membrane steel structure, designed by Frei Otto.*

Figure 3.8 *- Examples of form active structures*

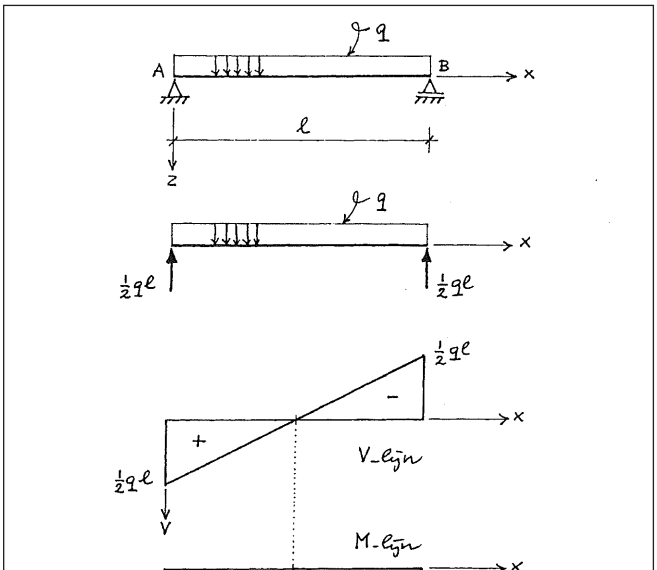


Figure 3.9 *- Example of a force distribution based on bending (Image courtesy of C. Hartsuijker, Toegepaste Mechanica Deel 1, page 395)*

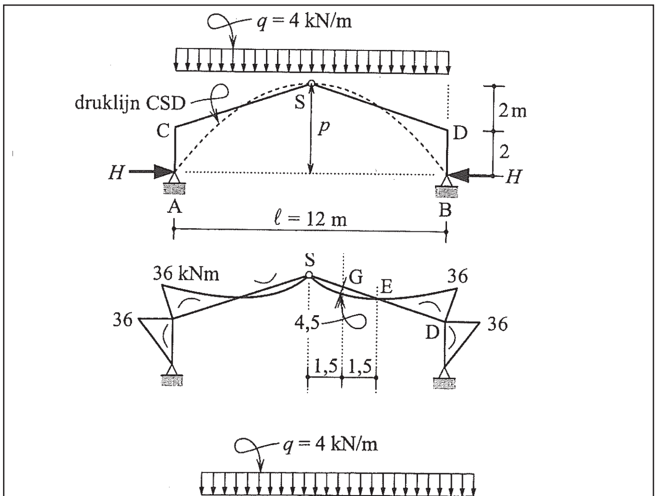


Figure 3.10 *- Example of a force distribution based on axial forces (Image courtesy of C. Hartsuijker, Toegepaste Mechanica Deel 1, page 646)*

3.3 Form active structures

The principle of *Form active structures* is used all around the world. The designs are often characterized by curvature and have an *uncommon*, *special* and *spatial* appearance (Figure 3.8).

First a short introduction is presented regarding force distribution systems.

3.3.1 Theory of force distribution systems

Two force distribution systems are considered:

- System mainly based on bending forces;
- System mainly based on axial forces.

<u>Force system based on bending force</u>

These structures are characterized by a combination of tension and compression in the cross section of an element. Due to the bending forces the element is also loaded by a shear force. Only vertical support forces are active (Figure 3.9).

An example of bending based elements are straight beams.

The loading on a beam is distributed to the support points by bending and shear forces in the element. This bending moment is a combination of tension and compression in the cross section of the element.

<u>Force system based on axial forces</u>

These structures can either be based on compressive forces or tensile forces.

In general the amount of material needed to be able to withstand the axial forces is less than for bending forces.

Besides that an important difference with the bending-system is the additional horizontal support forces (Figure 3.10).

A disadvantage of normal force structures is their shape. Especially when the function of the structure is a building. In these type of structures the volume and useful floor area are most important. Though the normal-force structures, are often curved, and especially near the supports the floor area can hardly be used.

Another disadvantage is the buckling behaviour. If the compressive force reaches a certain level, the element might fail due to buckling.

3.3.2 Theory of form active structures

Form active structures are structures in which the loading is taken by the form or the shape of the structure. In general they are non-rigid, flexible shaped in a certain way and secured at the ends.

A form active structure can support itself and is usually used to span and cover a space. They are governed by axial forces; either tensile or compressive stresses.

There are four types: cable structures, tent structures, pneumatic structures and arch structures.

The first three are mainly tensile form active structures and therefore of less interest for this research.

The arch structures are interesting, because they are compressive form active structures (Figure 3.11). They behave best when their shape is as a mirrored cable under load.

To become more familiar with these type of structures and how the theory has been applied in the building history, some engineers and architects and the method they use are given in the next paragraph.

3.3.3 Engineers designing form active structures

Several engineers and architects and their work have been looked at to obtain more insight of form active structures, such as common shapes and the historical developments throughout time. Information is given about:

- Antoni Gaudí (hanging chain models method);
- Pier Luigi Nervi (ribbed shell-structures);
- Felix Candela (thin concrete shell structures);
- Heinz Isler (inversed hanging cloth models);
- Frei Otto (tensile structures).

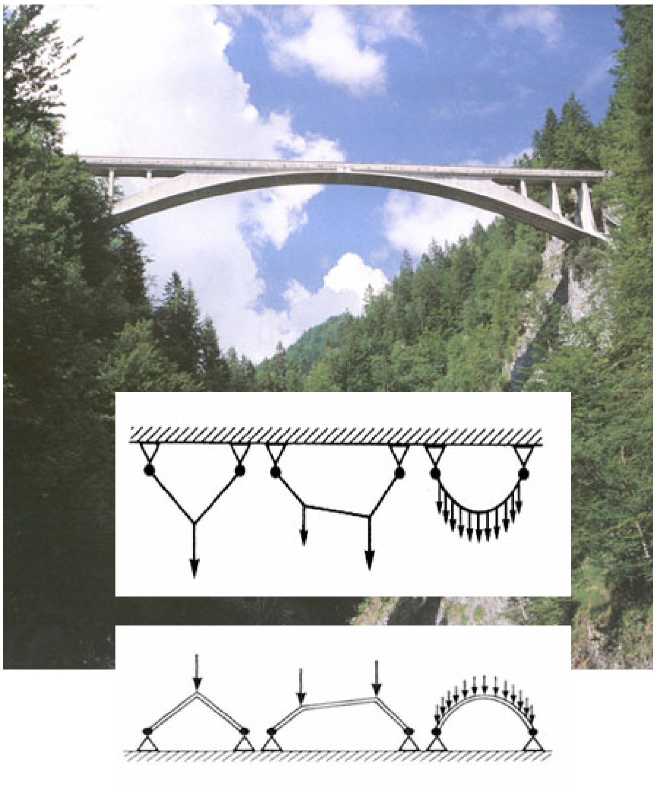
<u>MODELLING COMPRESSION BASED STRUCTURES</u>

Antoni Gaudí

Antoni Gaudi (1891 – 1979) is a Spanish engineer and architect. His work is inspired a lot by nature and was regarded as *uncommon* for the period of time he lived in. Some of his projects are the Sagrada Familia and the Colonia Guell, both located in Barcelona, Spain.

He is famous in the engineering world due to the method he used frequently in his work: the hanging chain model.

***Figure 3.11** - An example of a compressive form active structure. The Salginatobel bridge, Switzerland, Robert Maillart, 1929-1930.*



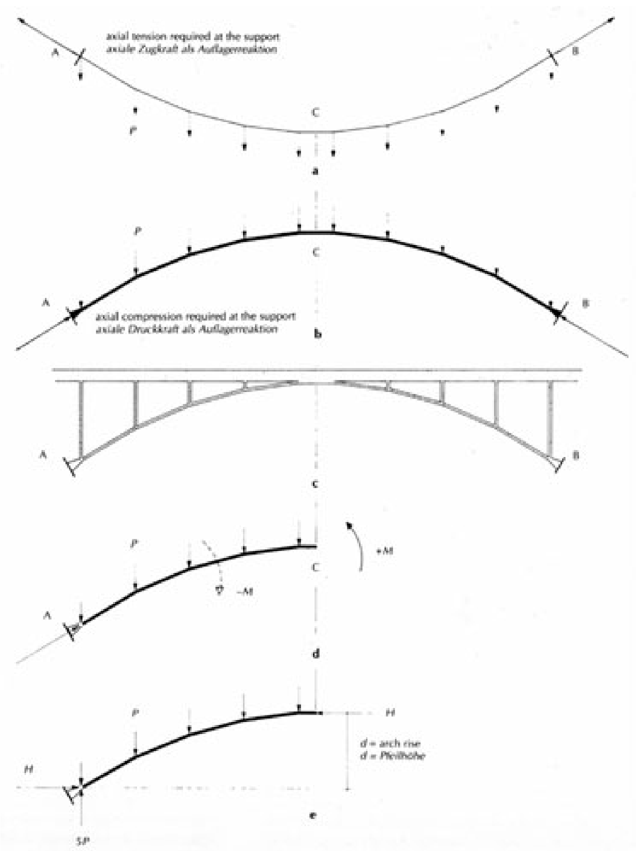


***Figure 3.12** - Hanging chain models, created by Antoni Gaudi to analyse and design structures, e.g. the 'Sagrada Familia', Barcelona, Spain.*

***Figure 3.13** - Palazetto dello Sport, a ribbed concrete shell, designed by Pier Luige Nervi (Rome, Italy, 1957)*

<u>The method of the hanging chain model (Gaudi)</u>
First chains are hung on a wooden frame. The chains can also be connected to each other to create a grid. This model will now find equilibrium under the loading of its own weight. When the shape is not moving anymore, it has found its equilibrium and only tensile forces are acting in the chains.
To obtain the compression-only model, the model is *frozen* (for instance by using a mirror or taking a photo) and the resulting model is rotated 180 degrees. In the new model only compression forces are present in the structure, instead of tensile forces (Figure 3.12).
To make the shape even more realistic, weights can be added to the chains, which represent nodal forces in the compression structure.
By playing with the length and connections of the chains he could make an architectural design. This method of designing is the exact opposite of the traditional way of structural design. Normally, the form of the building is given, and the structure is determined according to that, while in this case the structure is determined and the form follows the shape of this structure.

<u>Relevance of Gaudi's work for this research</u>:
Gaudi designed by adapting the shape and size of the structure, instead of the dimensions of the elements. He rather changes the form according to his wishes and the possibilities of the structure (form-finding), instead of calculating, dimensioning and adapting structural elements of one fixed model.
This aspect is one of the objectives of this research.

Pier Luigi Nervi
This Italian engineer (1891 – 1979) is renowned for his reinforced concrete structures.
The hangars and halls he designed in the end of the 1930's, such as the *Palazzetto dello sport* (Figure 3.13), are good examples of his common used ribbed structures.

<u>Relevance of Nervi's work for this research</u>:
The ribs can be seen as force-networks, which is one of the aspects of interest for the new application.

Felix Candela
Candela (1910 – 1997) originates from Spain, though the majority of his designs are found in Mexico.
Candela's major contribution to structural engineering was the development of thin shells made out of reinforced concrete. Reinforced concrete is extremely efficient in a *dome or shell* like shape.
He tried to solve problems by the simplest means possible. In regard to shell design, he tended to rely on the geometric properties of the shell for analysis, instead of complex mathematical means (Figure 3.14).

<u>Relevance for this research</u>:
One of the aims and assumptions of this research are thin shell structures, and if possible with no tensile forces, since reinforcement is not desirable.
Moreover one of the objectives of this research is to investigate the use of geometry to create the brick pattern and force network. Candela's designs can be a good reference in using geometry to design structures, even though his designs are not actually free form.

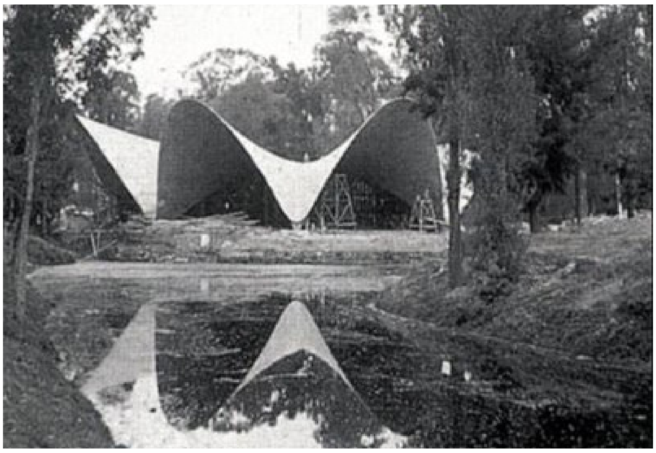


Figure 3.14 *- 'Los Manantiales Restaurant' in Xochimilco, Mexico City. A thin concrete hyper shell, designed by Felix Candela*

Heinz Isler
This engineer and architect from Switzerland (1926 - present) is famous for his designs of concrete shells.
Similar as Gaudi, observation of the natural world, where most structures have organic shapes with double curvature is very important to Isler. He tries to use mathematical formula as few as possible and therefore he approaches the challenges of each new structure by using physical modelling




Figure 3.15 *- Physical modelling by Heinz Isler*

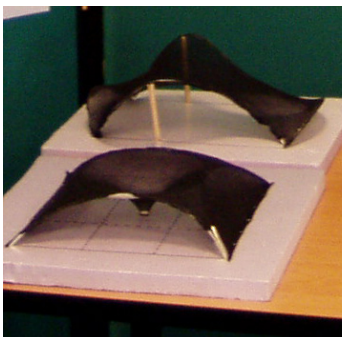
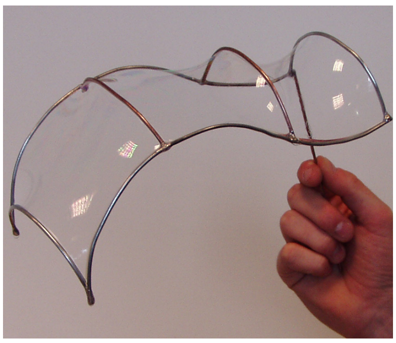




Figure 3.17 *- Examples of physical modelling, with for instance panties and soap bubbles (during assignment of the course CT5251 Special Structures)*

to determine the form and to investigate the stability of the structure.

After experimenting with pneumatic forms to create shell shapes, he discovered the, in his opinion best, method to create these shell forms: the reversed hanging-membrane model.

The method of inversed hanging membrane models (Isler)

This theory has a lot of similarities with the hanging chain model by Gaudi, with one difference: instead of creating a grid, a full surface (a shell) is produced (Figure 3.15).

The most commonly constructed type of Isler's shells is the bubble shell. (Figure 3.16) With these shells he moved away from the traditional – geometric described – shells (so that equations could be derived to calculate the forces and stresses within them) and developed a method of form-finding based on physical models.

Two projects realized with it are the *Grötzingen Open Air Theater* in Baden-Wurttemberg (Germany) and the *Heimberg Tennis Center* in Berne (Switzerland).

Relevance for this research:

His projects are good examples of thin shell structures and his inversed hanging membrane method is useful in the research, since it deals with form finding of full shell structures, which is one of the objectives of this research.

Figure 3.16 *- A service station in Deitingen-Süd (Switzerland). A bubble shell, designed by Heinz Isler*

3.4 Shell structures

An essential difference between a shell structure and a plate structure is that in the undeformed state the shell structure has curvature, while plate structures are flat. Thin shells are focused on axial forces and little bending, which is caused by the curvature of the surface. The supporting conditions of plates and beams are mainly determined by vertical forces, while for shells also horizontal forces are active.

Membrane action in a shell is primarily caused by in-plane forces, but there may be secondary forces resulting from flexural deformations. Where a flat plate acts similar to a beam with bending and shear stresses, shells are analogous to a cable, which resists loads through tensile stresses, or an arch, which resists loading through compressive stresses. The shell shape intents to eliminate tensile forces in the structure.

DIFFICULTIES

The structural behaviour of irregular curved surfaces, which have shell-like behaviour, is difficult to predict (Figure 3.19). Especially the buckling behaviour and 2nd order deformations are complex to determine.

More information about the difficulties in both designing and constructing a shell is found in literature.[9]

Shells are very efficient in carrying load. However a big disadvantage of shells is their brittle behaviour. A shell gives less warning signals when it is close to failing compared to other structures. For instance steel grids will first deform and a concrete structure will crack and the reinforcement will deform.

3.4.1 Thin shell structures

Thin-shell structures are light weight constructions and commonly based on the form active structure theory.

If it consists of prefabricated element, they are typically curved and assembled into large structures on site. Typical applications are fuselages of air planes, boat hulls and roof structures in building.

A thin shell is defined as a shell with a thickness which is relatively small compared to its other dimensions and in which deformations are not large compared to thickness. The

9 - Holgate, A., *The art of structural design*, 1986. (about the Sydney Opera House)

Figure 3.18 *- Shell structures*

(a) *- Korkeasaari Island Lookout Tower, Helsinki, Finland. Timber grid shell. Designed by Avanto Architects.*

(b) *- Sydney Opera House, Australia, 1973. Concrete frame & precast concrete ribbed roof. Engineered by Arup.*

(c) *- Roof of the 'Central Library' during construction, Tromsø, Norway, 1970. Designed by Gunnar Bøgeberg Haugen.*

(d) *- Same roof as in picture a, though now it is finished and being used. In combination with the glass facade it creates a nice composition.*

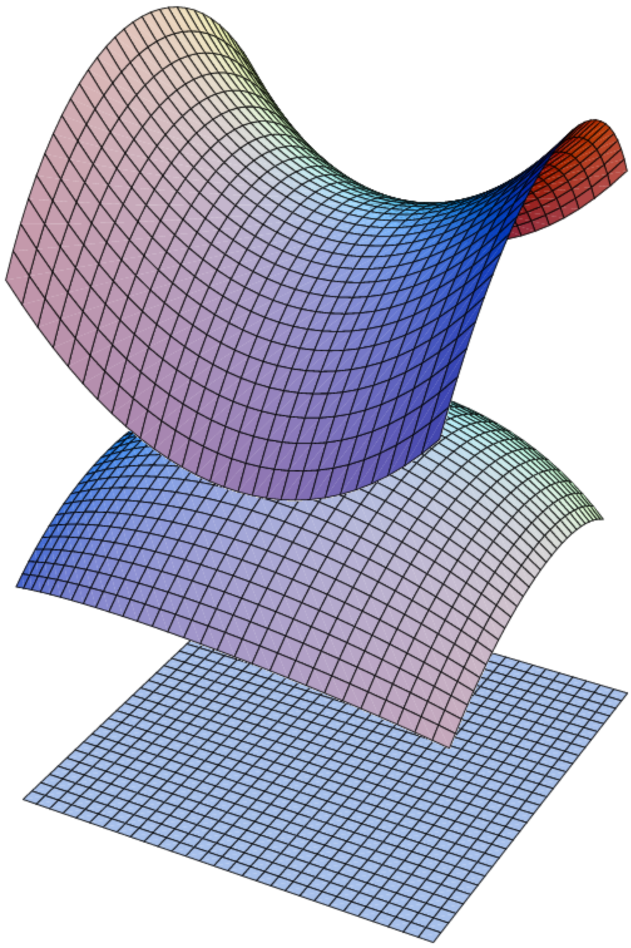


Figure 3.19 - *Shell geometry - Gaussian curvature.*
From top to bottom: anti-synclastic surface (negative curvature), synclastic surface (positive curvature) and flat surface (zero curvature)

ideal thin shell must be capable of developing both tension and compression, to be able to deal with deformations and point loadings.

3.4.2 Shell geometry

The surface of a shell is described using geometry. This reduces the complexity of the process of generating the brick pattern. To be able to describe the geometry of a surface, it is relevant to investigate the options to do this.
Below is shown how a surface can be classified by its curvature (the method of Gaussian curvature).

THE DEFINITION OF GAUSSIAN CURVATURE
Every point of a surface has two principal curvatures. The curvatures are found using the formula below, with the use of the radius of the curve.
With the curvature is measured how the surface bends by different amounts in different directions at a certain point.
The result is often presented with the Gaussian curvature: the product of two main surface curves through a point on the surface (Figure 3.19).
The formula for the Gaussian curvature (K) is: $K = \kappa_1 \cdot \kappa_2$
With: κ_1 = principal curvature 1
κ_2 = principal curvature 2

And:
$$\kappa_1 = \frac{\partial \phi_1}{\partial s} = \frac{1}{R_1}$$

With: $\partial \varphi_1$ Difference in angle 1
∂s Difference in distance on the curve of main curvature 1
R_1 Radius of the curve (distance to centerpoint)

Three cases are to be considered:
1. K > 0 → Synclastic surface (the two curvatures κ_1 and κ_2 are in *the same* direction);
2. K < 0 → Anticlastic surface (the two curvatures κ_1 and κ_2 are in *different* direction);
3. K = 0 → Zeroclastic (one of the two curvatures is zero).

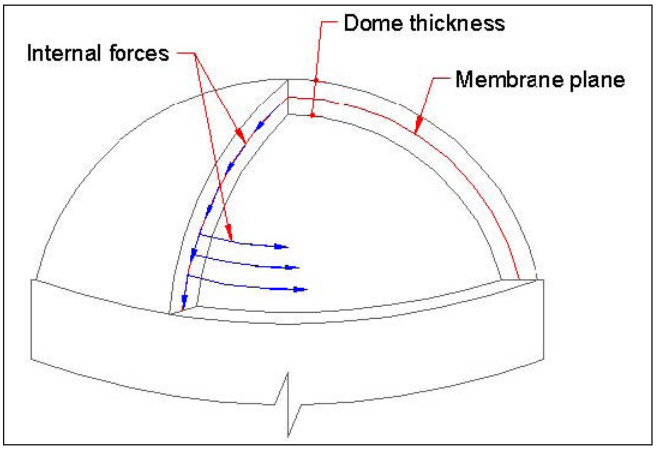
The interesting aspect the three cases of Gaussian curvature are: when the curvature is synclastic (positive), the surface is shell like and is therefore desirable in this research. When the curvature is anticlastic (negative) there are tensile forces in the *meridional* direction (. Finally when the curvature is zeroclastic, the structure is not double-curved and considered a ruled surface.

3.4.3 Structural mechanics of shells

FORCE FLOW IN SHELLS
Shell like behaviour is characterized by mainly axial (compressive) forces and little bending moments.
Two important forces are (Figure 3.20):
- Meridional forces;
- Hoop forces.

BUCKLING OF SHELLS
This aspect has not been regarded in the research. It should be investigated before using the output of the new application, in practice since it is an important aspect to secure the stability of a shell.[10]

10 - Coenders, J.L., 2007. *Dictaat CT5251 Structural Design – Special structures*, 2nd ed. [1] - page 131-140.



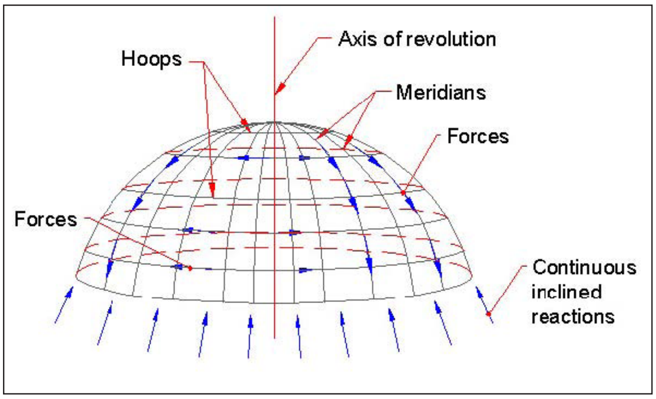


Figure 3.20 - *Shell like behaviour: meridional and hoop forces*

3.5 Structural analysing methods

Several theories and methods to analyse shell structures exist. One is discussed in this paragraph: *The membrane theory.* This theory is applied in two types of methods:

a. Analytical methods;
b. Graphical methods.

THE MEMBRANE THEORY

The membrane theory is the basis of the current methodology of dome structural analysis and provides a reasonable approximation for thin-shelled domes.[11] The predominantly load case is most often its own weight.

The membrane theory assumes the following:[12]

1. Applied loads are resisted by internal forces within the surface, which has no stiffness against bending; therefore internal forces are either pure tension or pure compression;
2. On a symmetrically and uniformly loaded dome, internal forces act perpendicular to each other in the meridional and latitudinal hoop directions;
3. Internal forces are coplanar; that is, the membrane has zero thickness;
4. The membrane plane is located along the centre line of the actual dome thickness; thus the lines of thrust must also follow the centre line.

The last assumption, which constrains the line of thrust to a two-dimensional plane along the centre line of the dome, merits discussion. In reality, many lines of thrust may lie within the finite thickness of the dome, all viable solutions for a stable structure. Furthermore, the lines of thrusts in the meridional and hoop directions may not coincide. The thickness of the structure only defines a permissible region within which a membrane solution must be found.[13]

Therefore the membrane theory operates on the lower bound principle: if one solution is found by assuming the line of thrust at the centre line that achieves stability and equilibrium, then the structure will also find its own solution.

The membrane theory remains the basis of most modern engineering methods that model the behavior of domes.

ANALYTICAL METHODS

Analytical methods utilize geometry and calculus to calculate the change of internal forces from one side of the element to another side of an infinitesimally small element of the dome. The sum of these forces must establish equilibrium in directions tangential to the meridians, normal to the dome surface, and parallel to the latitudes for the entire dome structure.[14]

Though formulae for membrane analysis of domes was introduced as early as 1858, it was not until 1926 when a mathematical theory describing the behavior of dome shells of revolution became simplified enough for practical use.[15]

GRAPHICAL METHODS

Graphical analysis provides a visual method of solving for structural equilibrium through the knowledge of building geometry and forces.

Information obtained from a graphical analysis include meridional and hoop forces along the arc of a dome, horizontal thrusts, and the deviation of the thrust line from the line of the assumed membrane in cases where no tensile capabilities of the dome structure are assumed.

- Possible for shells, made by surfaces of revolution, subjected to its own weight;
- More insight in flow of forces;
- Easy way to construct a polygon of forces. This polygon then represents the corrected line of thrust, whereby the hoop-forces correct the line of thrust of the load to coincide with the system line of the shell;
- The graphical method gives a very good result compared with a FEM calculation. Also it gives a good understanding of the mechanical behavior, besides only the numerical result.

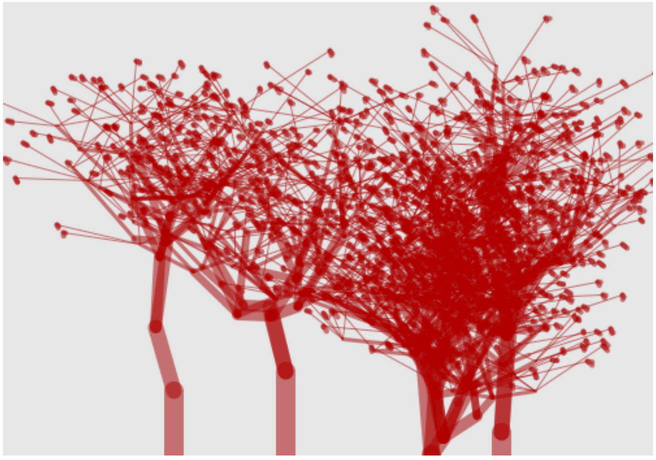
11 - Heyman, J. (1996). *Arches, Vaults and Buttress.* Hampshire, Great Britain: Variorum.

12 - Heyman, J. (1995). *The Stone Skeleton.* Cambridge: Cambridge University Press.

13 - Heyman, J. (1996). *Arches, Vaults and Buttress.* Hampshire, Great Britain: Variorum.

14 - Heyman, J. (1977). *Equilibrium of Shell Structures*. Oxford: Clarendon Press.

15 - Billington, D. P. (1982). *Thin Shell Concrete Structures.* New York: McGraw-Hill Book Co.



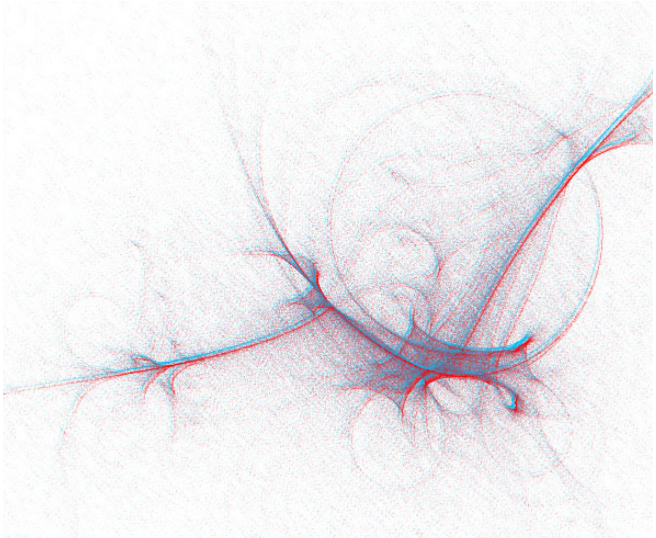
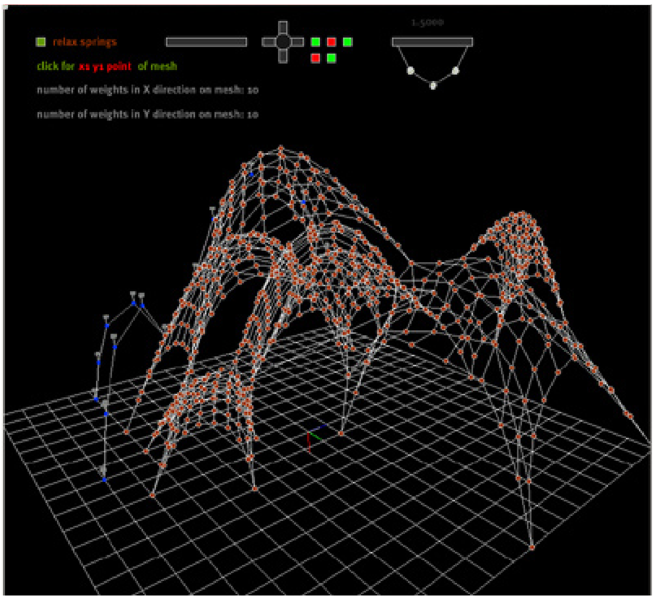
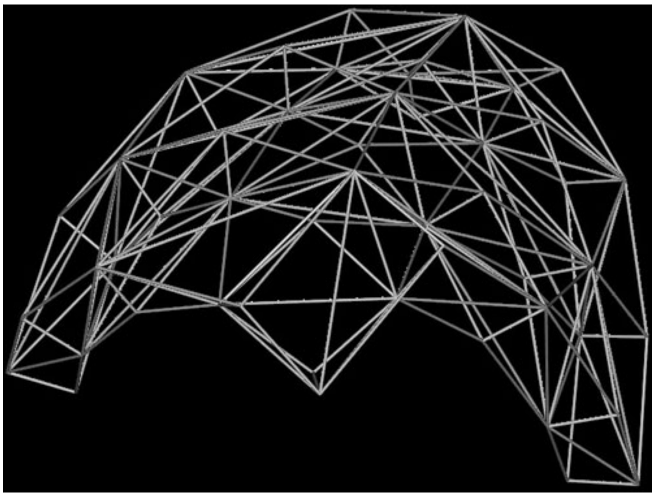


Figure 3.21 - *Possibilities with Processing and Java [23]*





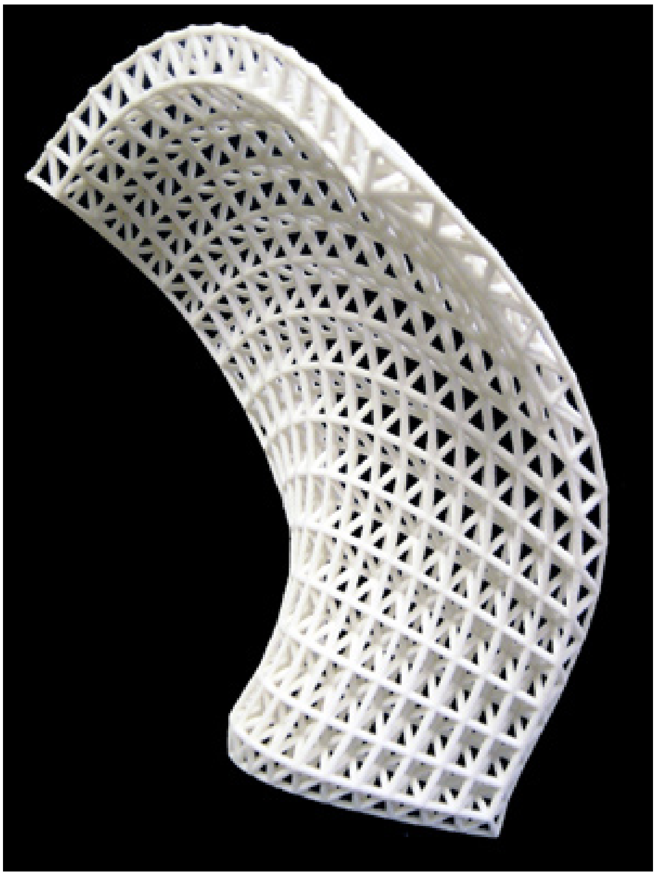


Figure 3.22 - *Results of application of algorithms (Image courtesy of A. Killian)*

3.6 Computational design: Processing, JAVA and algorithms

3.6.1 The software Processing

Processing is an open source project initiated by Casey Reas and Benjamin Fry, both formerly of the Aesthetics and Computation Group at the MIT Media Lab.

It is "*a programming language and integrated development environment (IDE) built for the electronic arts and visual design communities*", which aims to teach the basics of computer programming in a visual context, and to serve as the foundation for electronic sketchbooks. One of the stated aims of Processing is to act as a tool to get non-programmers started with programming, through the instant gratification of visual feedback (Figure 3.21). The language builds on the graphical capabilities of the Java programming language, simplifying features and creating new ones.

Reasons to use Processing

Besides the fact of designing with bricks, *interactivity* is the aspect which has to distinguish this application from other available programs. To assure the application is interactive, there are two requirements:

1. The ability to adapt the shape in a simple and fast way;
2. The new shape has to be analysed and results must be given fast, so that the concepts can be altered real time during a meeting.

Processing offers the possibilities for both requirements and for that reason it has been chosen to use during this research and to design the new application with.

3.6.2 Programming language Java

This is one of the three modes, available in Processing, and is the most flexible one, allowing complete Java programs to be written from inside the Processing Environment (as long as they're still subclasses of PApplet). This mode is for advanced users only and is not really recommended. Using this mode means that any additional tabs will no longer be inner classes, meaning that you'll have to do extra work to make them communicate properly with the host PApplet. It is not necessary to use this mode just to get features of the Java language.

```
public class MyDemo extends PApplet {
  void setup() {
    size(200, 200);
    noStroke();
    fill(0, 102, 153, 204);
  }
  void draw() {
    background(255);
    rect(width-mouseX, height-mouseY, 50, 50);
    rect(mouseX, mouseY, 50, 50);
  }
}
```

A good example is the digital hanging chain modelling tool, created by Axel Kilian: CADenary [16]. This modelling program has been made in Processing. The program is available online [25].

3.6.3 Algorithms

An algorithm is a sequence of finite instructions – steps – often used for calculation and data processing. It is a method in which a list of well-defined steps for completing a task is run through. The transition from one step to the next is not necessarily deterministic; some algorithms incorporate randomness. These are known as probabilistic algorithms.

Nowadays algorithms are most of the times connected to the use of the computer, because of the speed they offer. That way the result from an algorithm is obtained fast. When looking at algorithms from this way, they are essential to the way computers process information.

The analysis and study of algorithms is a discipline of computer science and is often practiced abstractly without the use of a specific programming language or implementation. In this sense algorithms-analysis resembles other mathematical disciplines in that it focuses on the underlying properties of the algorithm and not on the specifics of any particular implementation.

Three examples of the results, images and analysis that are obtained when algorithms are used are shown in figure 3.22.

16 - CADenary, form finding project developed in Processing by Axel Kilian, using a Particle Spring library developed by Simon Greenwold [12]

3.7 Overview of choices and assumptions and limitations

DECISIONS

- A method inspired on graphical analysis is used to analyse the model: the Thrust Network Analysis [10];

- Only shell type structures are regarded. The Gaussian curvature has to be positive;

- The program Processing is used to create a stand-alone application, which makes use of the JAVA scripting language.

ASSUMPTIONS

- The masonry can not be loaded by any tensile forces, because the tensile strength of the masonry is neglected;

- Reinforcement is not used and applied;

- Compression-only structures are designed;

- The maximum compressive strength of the masonry cross section is 30 N/mm^2. When the stress in the network is higher, a warning must be given in the application.

OBJECTIVES

- The new application has to be interactive: after any change in the design, the analysis and pattern have to adapt instantly;

- The new application has to be able to perform a partial structural analysis of a double-curved shell. However only the force flow is checked;

- The new application has to be able to create a brick pattern;

- The new application has to be a stand-alone application, with the ability to export the result, so that it can be used in the continuation of the design process.

LIMITATIONS

- The buckling behaviour of the shell shape is not regarded during the analysis;

- The displacements and accompanying second order analysis is not regarded;

- The Gaussian curvature should be positive in the whole structure. This limits the range of possible heights for every node, since it is dependent on the height of surrounding nodes.
  This check has not been implemented in the prototype of the application;

- The tensile forces in the hoop direction of the surface are not analysed and checked.

Figure 4.1 *- Examples of polygons*

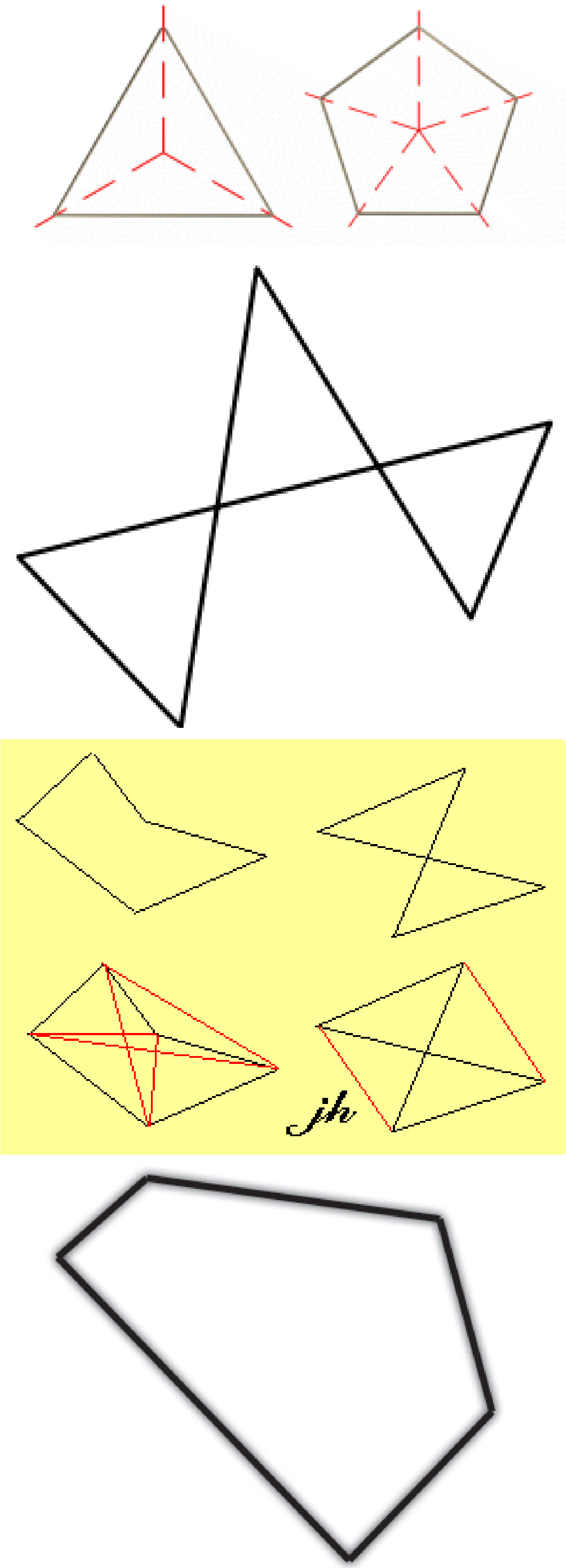


Figure 4.2 *- Polygon classified by number of sides*

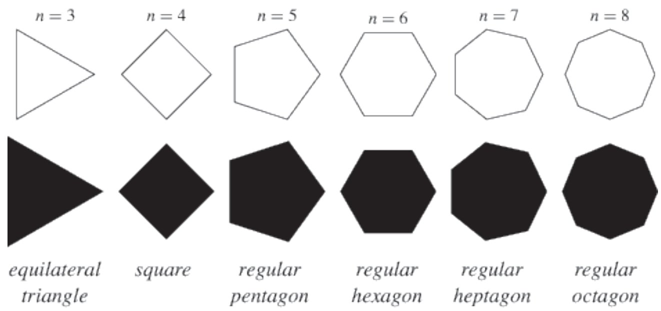


Figure 4.3 *- Polygon classified by convexity*

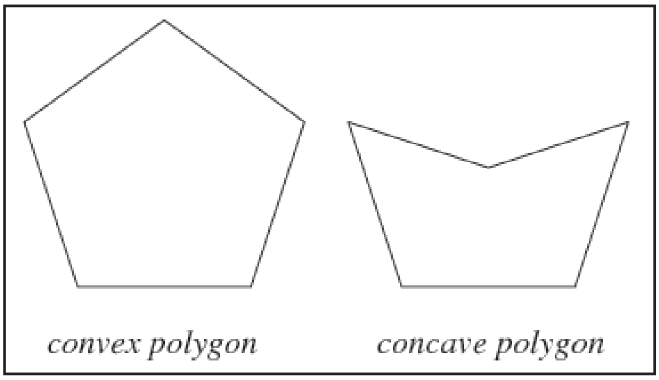
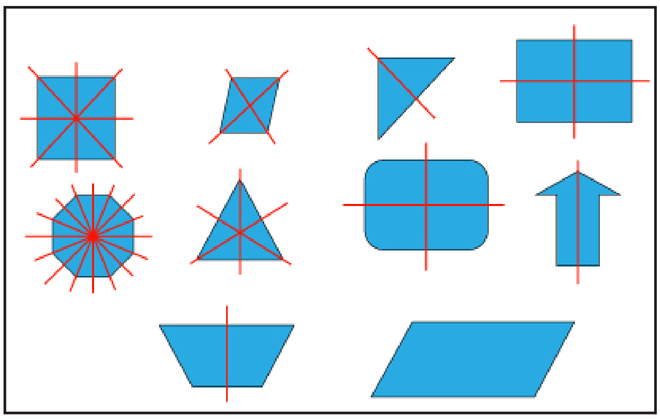


Figure 4.4 *- Polygon classified by symmetry*



CHAPTER 4 THEORIES AND METHODS

Introduction

In this chapter the theories and methods that are used in the research are presented.
One of the basic concepts is the *force polygon* (§ 4.1). Once this concept is clear the theory of the Thrust Network Analysis is discussed. After an introduction (§ 4.2) an explanation is given why and how this theory is useful for this research (§ 4.3).
The next step is to show and explain the steps taken in this theory (§ 4.4). One of the steps is to solve a linear optimization problem, for which the Simplex method is used (§ 4.5).
The next two paragraphs focus on the surface- and brick pattern-generation, in which the theory of Catmull Rom splines is used (§ 4.6 and § 4.7).
Finally in the last paragraph (§ 4.8) a list of limitations is presented, in addition to the ones presented earlier (§ 3.7).

An important remark regarding the *Thrust Network Analysis:* The majority of the material presented in this chapter is based on the article by Philippe Block and John Ochsendorf, who are the authors and creators of this theory [10].

4.1 Theory of force polygons

Two descriptions of a force polygon are:

1. "A closed polygon whose sides are vectors representing the forces acting on a body in equilibrium."
2. "The graphical representation of the internal and external forces of a structure."

<u>POLYGON</u>

The first term to discuss is *polygon*. This word derives from the Greek πολύς ('many') and γωνία (gonia), meaning 'knee' or 'angle'. Accordingly a polygon is "many angles". Today a polygon is more usually understood in terms of "many sides".
An appropriate description of the term polygon is:
"A plane figure bounded by three or more straight line segments." (Figure 4.1)

These line segments are from here-further called *sides* and the points where two lines meet are the polygon's *vertices*. If the object is not closed, it is referred to as polyline: one line consisting of several line-elements connected to each other. When the object consists of less than three sides it can not form a closed figure and is therefore not a polygon.

Polygons can be classified in different means:

1. By number of sides (Figure 4.2);
2. By convexity (Figure 4.3);
3. By symmetry (Figure 4.4).

For this research the relevant type is the *simple polygon*. For this type the sides of the polygon do not cross themselves.
The number of nodes depends on the number of lines connected to a node in the network. The symmetry and convexity of the polygon is determined by the composition of the lines network (§ 4.4).

***Figure 4.5** - The length of the line, represent the magnitude of the force in it*

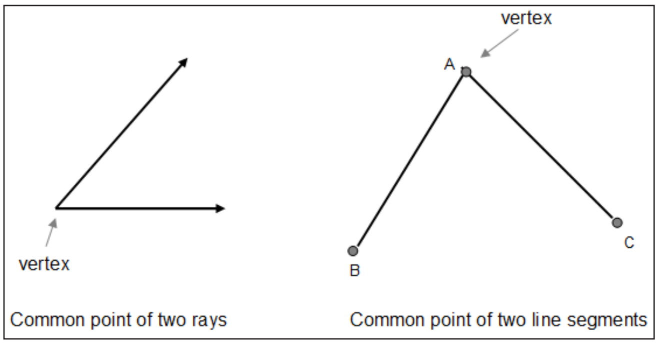
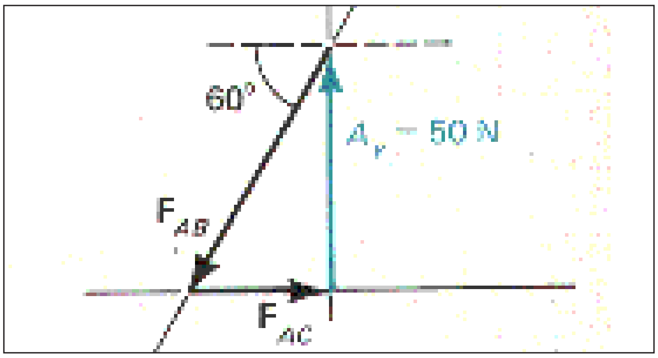


***Figure 4.6** - Constructing of one force polygon when one value is known*



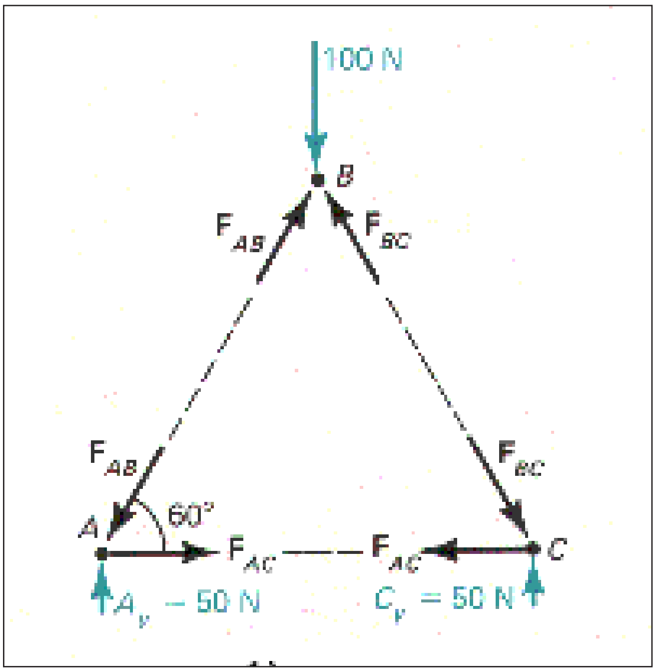


***Figure 4.7** - 2D example: a node in a network, with three lines connected to it*

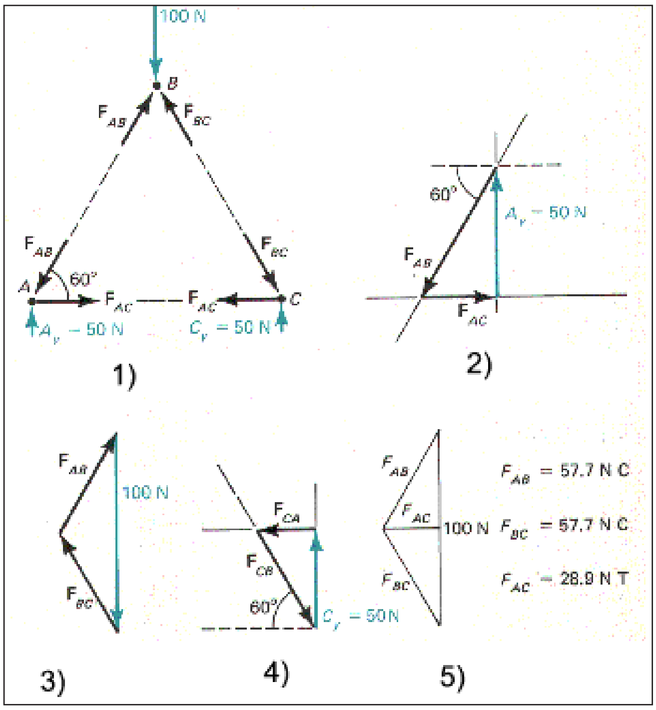


***Figure 4.8** - Closed force polygon: the related node is in equilibrium*

FORCE POLYGON

If the sides of the polygon are representations of forces in a network, the resulting closed polygon is regarded as a force polygon. The lengths of the sides represent the magnitude of the corresponding force (Figure 4.5).

Reading back the second description of a force polygon, as given above, a body in equilibrium is represented by a closed polygon. When regarding a node as a body a force polygon can be constructed for each node. This polygon consists of the loading and the forces in the line-elements connected to the node. Since equilibrium is required, the polygon has to be closed. Since the magnitude of one of the lines is known, in this case the force, the polygon can be constructed. (Figure 4.6)

2D EXAMPLE

The theory is now explained using an example in 2D space (Figure 4.7). An arch consists of several stones. Each stone is represented by a node, located in the centre of the stone. The material in between two nodes is shown as a line. To each node a force is applied, which in reality is for instance the own weight of the stone, increased with a certain live load.

The force polygon can now be created, using three lines: the direction of the loading and the direction of the two lines connected to the node (Figure 4.8). When the polygon is closed, the node is in equilibrium.

Though the arch consists of more than one node and a line is connected to two nodes. Therefore the polygons of all nodes have to be combined into one diagram. When the polygons of all nodes are in equilibrium, the whole arch is in equilibrium. The resulting diagram gives a clear insight in the size of the forces in all lines. Moreover the support forces are shown, when regarding the polygons of the border-nodes.

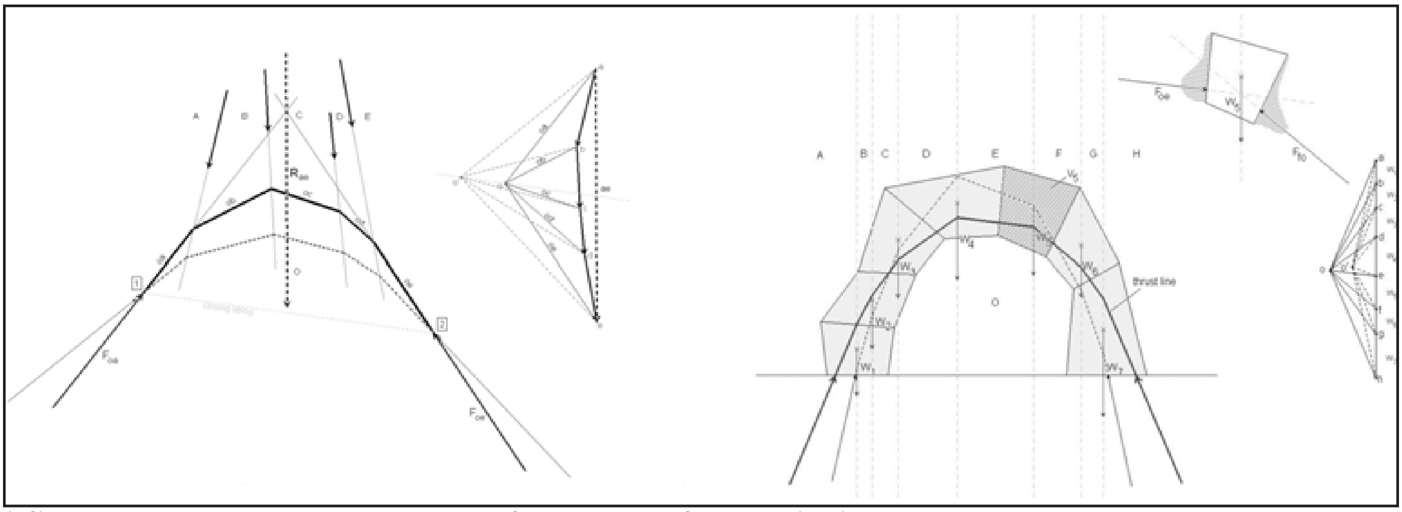
This method has been documented in the Thrust Line Analysis theory. [1]

3D APPLICATION

Further research at the Massachusetts Institute of Technology (M.I.T.) into the theory of force polygons, 3D line networks and the Thrust Line Analysis has resulted in an enhanced theory: the Thrust Network Analysis. In short, this is the 3D application of the Thrust Line Analysis; instead of regarding only two neighbouring nodes and lines, the network is extended to three dimensions. This is discussed in the next paragraphs.

1 - Block, Philippe, 2005. Available at: http://web.mit.edu/masonry/interactiveThrust/

Figure 4.9 *- Theory of the Thrust Line Analysis method (Image courtesy of Philippe Block)*



(left) Two possible compression-only equilibrium shapes for a random set of loads, and (right) an interactive thrust-line application: the user can adapt the geometry by dragging control points and the structural feedback, in the form of a thrust-line, is updated in real-time. The magnitudes of the forces in the system are visualized in the accompanying funicular polygon (right).

4.2 Introduction of the Thrust Network Analysis

The Thrust Network Analysis (TNA) is a new methodology for three-dimensional equilibrium calculations. The theory presents a methodology for generating compression-only vaulted surfaces and networks. Two important aspects are:

1. *The primal grid*, which is the planar projection of a three-dimensional grid of a compression shell;
2. *The dual grid,* which is the reciprocal figure of the primal grid.

When this relation is used in a linear optimization method (in this case the Simplex method) it *"provides a graphical and intuitive method, adopting the same advantages of techniques such as graphic statics, but offering a viable extension to fully three-dimensional problems".*[2]

<u>THRUST LINE ANALYSIS</u>

This new theory is based on the Thrust Line Analysis, which is a powerful graphical method for calculating the range of lower-bound equilibrium solutions of compression-only systems (Figure 4.9 and § 4.1).

A disadvantage of this analysis method is the limitation of application for 2D cases only, such as arches, while a shell structure has a three-dimensional force distribution.

<u>FIELDS OF APPLICATION</u>

The TNA theory is applied in several situations:

- For the analysis of vaulted historical structures, specifically in unreinforced masonry;
- To design new vaulted structures.

The last category is interesting for this final thesis research. More information regarding the benefit of the TNA for this research is found in § 4.3.

2 - Block, Philippe, 2007. *Journal of the international association for shell and spatial structures: J. IASS*, nr. 47, p.167 [10]

4.3 The benefit for this research

The TNA makes use of four main elements in the process of analysis: [3]

1. Force networks, representing possible forces in equilibrium in the structure;
2. Reciprocal diagrams, visualizing the proportional relationship of the horizontal forces in the network and providing a high level of control for the user to manipulate the force distributions in the system;
3. The use of envelopes (boundaries) defining the solution space;
4. Linear optimization, resulting in fast computation of results.

The main requirements for the new application are (§ 2.2):

1. Interactivity;
2. Fast results;
3. Compression-only structures;
4. Double curvature models.

Each of these requirements is shortly discussed to see if the main elements of TNA – as mentioned above – can be used to fulfill these requirements:

<u>INTERACTIVITY</u>

This is taken care of by point 2 (the reciprocal relationship of the diagrams). In the new application it is possible:

- To adapt the position of nodes in the 3D model;
- To change the loading;
- To add or remove lines.

So in three ways interactivity is assured:

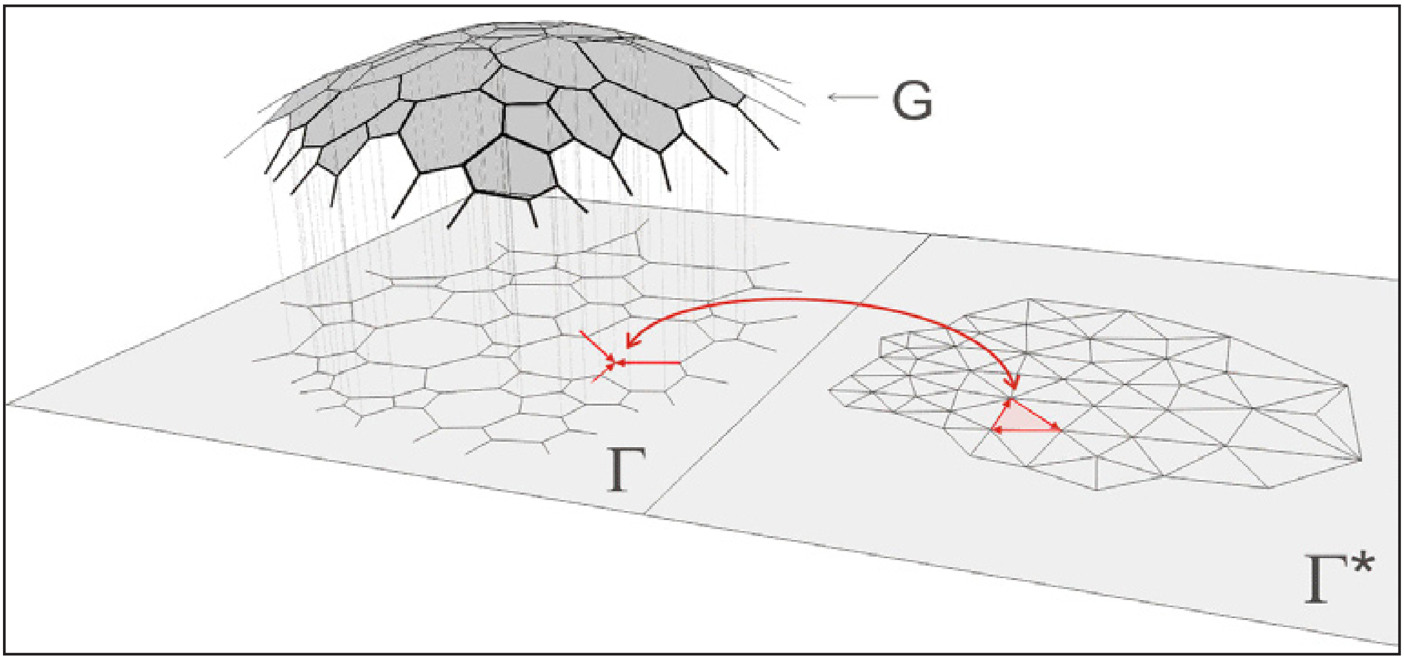
1. The ability to adapt the force distribution by adapting the scale factor (§ 4.4.5) and directly related the reciprocal figure;
2. The ability to change the force network (the 3D model or the primal grid);
3. The ability to change the loading for each node.

<u>FAST RESULTS</u>

The results are available fast due to the use of a fast linear optimization method (the Simplex method). Performing the

3 - Block, Philippe, 2007. *Journal of the international association for shell and spatial structures: J. IASS*, nr. 47, p.169 [10]

Figure 4.10 *- Relationship between force network, primal grid and dual grid (Image courtesy of Philippe Block)*



Relationship between compression shell (G), its planar projection (primal grid Γ) and the reciprocal diagram (dual grid Γ) to determine equilibrium.*

analysis and finding the solution in this research is achieved by a one-step linear optimization. This optimization is performed rapidly and so the results are available rapid.

COMPRESSION-ONLY STRUCTURES

The theory of TNA is based on compression-only structures. If tensile forces are present in the network it is not possible to obtain the primal and dual grid.

One remark: all loads are applied in the same direction; in this case vertically, as is the case for gravitational loading. Due to this wind loads are not taking into account for now.

A result of the assumption of compression-only structures is that the surface of the shape can not curl back onto itself. The 3D force networks represent load paths throughout a structure. This observation is important, since this means that in theory only these network paths have to be completely of brick and all planes between and inside the network lines can be open. These open planes can then be used to create brick patterns to comply with the architectural wishes.

DOUBLE CURVATURE MODELS

The Thrust Network Analysis is developed especially for three-dimensional models, which often have double curved surfaces.

In the next two paragraphs, the benefits from architectural point of view (§ 4.3.1) and from engineering point of view (§ 4.3.2) are explained.

4.3.1 Architecture

As long as the compression-only force network is in equilibrium, the resulting model form is realistic and applicable. Architectural *freedom* is created due to the ability to change the shape of the shell and with it the brick pattern.

The ability to adapt the brick pattern is a recommendation for further development of the application.

Two additions that would significantly increase the possibilities of the application are:

- The implementation of being able to use more than one brick shape and of irregular, non-rectangular shapes;
- The possibility to use colored bricks and create color-patterns.

These additions are not implemented during this research and are therefore recommendations for further research (Chapter 7).

4.3.2 Mechanics

Due to the use of force diagrams (the primal and dual grid) the Thrust Network Analysis gives a clear graphical representation of forces in the system (Figure 4.10 and § 4.1).

Only the force flow and equilibrium of the structure is analysed. Aspects such as buckling behaviour and displacements are not taken into account.

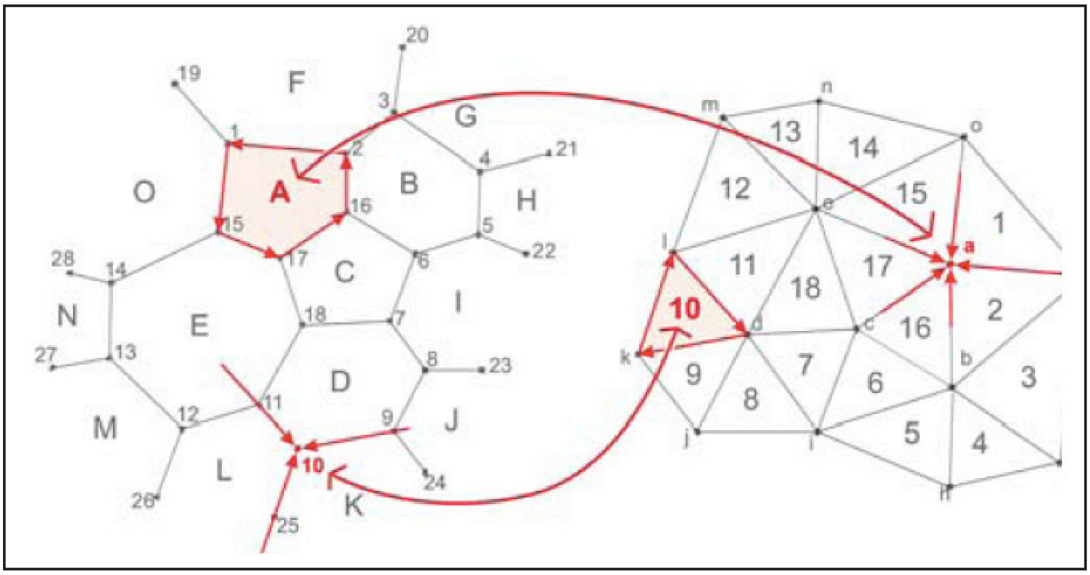


***Figure 4.11** - The two-way relation between primal and dual grid (Image courtesy of Philippe Block)*

4.4 Explanation of steps

The important steps of the TNA for this research are:

1. Construct the primal grid (§ 4.4.1);
2. Construct the dual grid (§ 4.4.2);
3. Solve the problem (§ 4.4.3).

Two other aspects concerning the TNA are the application of loading (§ 4.4.4) and the function of the scalefactor ζ (§ 4.4.5).

4.4.1 Construct the primal grid

The primal grid is the planar vertical projection of a three-dimensional grid or (force) network of a compression shell. If to compare it with an event in nature: imagine the sun exactly above the center of the grid; the shadow of the three dimensional network on the ground is the primal grid.

4.4.2 Generate the dual grid

The lines connected to a node in the primal grid, form a polygon of forces in the dual grid (Figure 4.11). A definition of *polygon of forces* is: *"The sides of a force polygon represent in magnitude and direction a system of forces in equilibrium"* [8].
The direction (angle) of each line remains the same, the length (representing the force) and position change.
To relation between the dual grid and primal grid is exactly the same though: the lines connected to a node in the dual grid form a polygon of forces in the primal grid. This two-way relation is known as a *reciprocal relationship*. The mechanical property of reciprocal diagrams is expressed in the following theorem by professor Maxwell:
'If forces represented in magnitude by the lines of a figure be made to act between the extremities of the corresponding lines of the reciprocal figure, then the points of the reciprocal figure will all be in equilibrium under the action of these forces.'[4]

The steps to visualize this reciprocal relationship are:

1. For every node in the primal grid (excluding the foundation nodes) obtain the related lines;
2. Make a polygon of these lines, by adding one to the end of the other. Start with any line and add the next, as it is the next in clockwise order around the node in the primal grid;
3. If any of the lines of the regarded node is not connected to a foundation node: Close the force polygon, by finding the intersection point of the two lines that are not connected yet;
4. Place and scale all the polygons;
5. Determine the range of possible angles for the foundation lines and set it at the midpoint of this range.

A more detailed explanation, including pictures and drawings, is found in Appendix A.

4.4.3 Solve the problem

Using the geometry of the primal grid and the dual grid, together with the nodal loading and the boundary conditions – the minimum and maximum height of each point – this problem can be solved using a one-step linear optimization. More about this optimization and finding the solution can be found in § 4.5.
The analysis is performed using several checks:

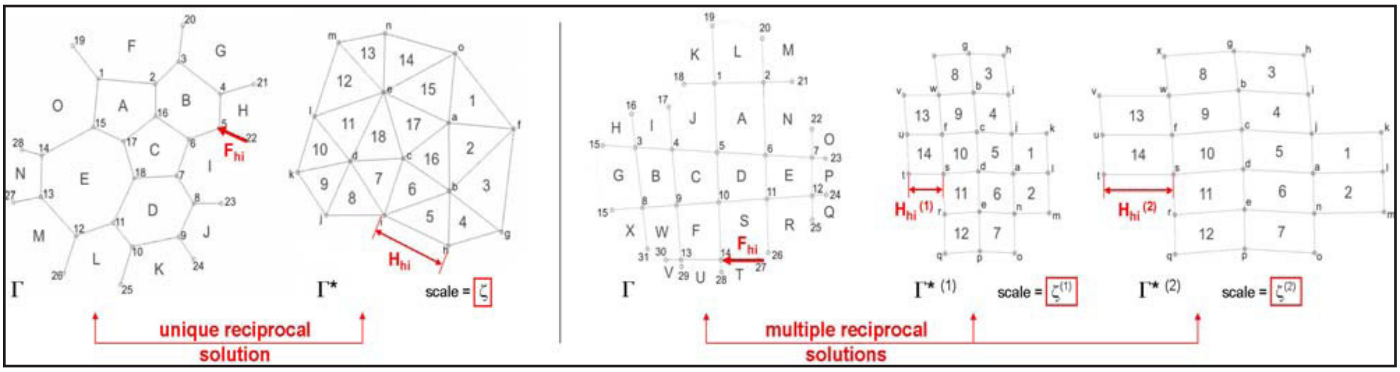
- The network lines have to lie within the brick thickness, which is represented by the minimum and maximum height of each node;
- The stress in the structure should be lower than the maximum allowed stress. The active stress must be determined by dividing the force in a network line with the masonry area related with that line;
- The maximum allowable angle between bricks, so that the bounding layer does not exceed the maximum thickness. Or in other words: to avoid gaps in the masonry pattern.

4.4.4 Nodal loading

The weights attributed to the loaded nodes come from distributing the dead load of the 3D area around those nodes. In addition to this self weight, loads such as asymmetric live loads can be applied.
An extra point of attention: in this research all loads are applied in the same direction, as is the case for gravitational loading. It is recommended to expand this option by making it possible to add loading in any direction. The loading should be divided into a horizontal and vertical part. The vertical

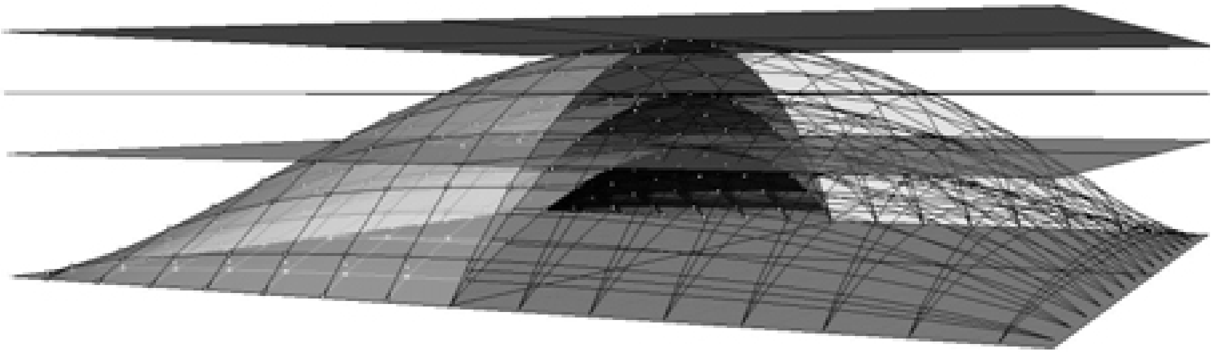
4 - Maxwell, Professor Clerk, 1864. *Philosophical Magazine*. p.258 [11]

Figure 4.12 *- Valency and the influence of the scalefactor (Image courtesy of Philippe Block)*



(a) *- Valency is 3 and the dual grid is set*

(b) *- Valency is 4 and the dual grid has several solutions. The scalefactor determines the final result*



(c) *- Decreasing the scale factor ζ of the dual grid means overall lower horizontal forces in the system and hence a deeper solution for the same set of applied loads.*

part is still used to solve the problem, though the horizontal part has to be added in the force polygon. If and how this influences the reciprocal relationship and the final result is a topic for further research.

4.4.5 Scalefactor ζ and its influence on the solution

The user can manually change the force distribution by changing the scale between the primal and dual grid. This scale factor ζ is not always of importance. This depends on the *valency* of the grid. A grid where the nodes have three lines connected to them has valency 3. The same way a grid where the nodes have four lines connected to them, has valency 4.

When a grid has valency 4 or higher it is regarded as an *indeterminate* grid. In this case, the scalefactor has influence on the result of the analysis (Figure 4.12).
Decreasing the scale factor ζ of the dual grid means overall lower horizontal forces in the system and hence a deeper (higher) solution for the same set of applied loads. Following the same reasoning, a higher scale factor causes a shallower shell and thus higher horizontal forces.

4.5 Linear optimization theory: the Simplex method

Most of the information mentioned in this paragraph has been found in a book, in which the theory of matrices and linear optimisation are explained [7]. Also two websites have been used; one dealing with linear optimization problems [21] and the other with Simplex method [22].
In the last paragraph an example is given of these steps applied to an example network. (§ 4.5.5)

4.5.1 General information

To make an analysis according to the theory of the TNA, a one-step linear optimization is used. First of all the theory of linear optimization will be shortly discussed.

Linear programming and solving
There are several techniques and theories dealing with linear optimization problems. A first distinction is the Graphical and Analytical method. The first one gives clear results, but performs best with just one or two variables in the objective function. When this function becomes more extensive and when the problem increases in number of steps needed to solve it, the Analytical methods are better. In this research the Simplex method is used.

Simplex Method
In this method the theory of Gauss-Jordan (GJ) is used. It deals with solving LP problems with pivoting. *Pivoting* uses row operations (known as Gauss-Jordan row operations) to change one matrix entry (the pivot) to “1”, and then to change all other entries in the pivot’s column into zero’s. More detailed information regarding the GJ-theory is added in the appendices [Appendix B].
However, the Simplex method has some disadvantages. For example, it requires that all variables be non-negative (≥ 0); also, all other constraints must be in ≤ form with non-negative right-hand-side (RHS) values.
More information about the steps and actual application of the Simplex method is found in the appendices [Appendix C].

Two important variables within the Simplex method are the objective function and the constraints of the problem.

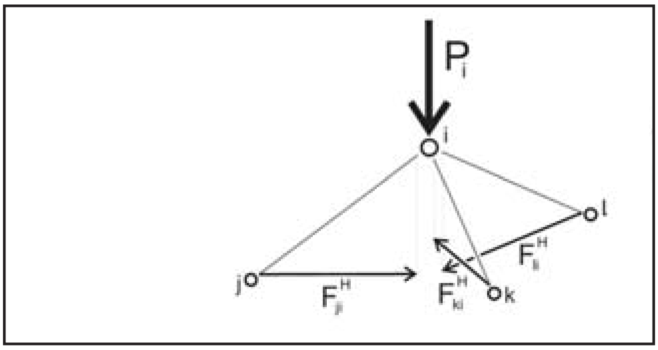


Figure 4.13 *- Equilibrium of one node in the force network (Image courtesy of Philippe Block)*

The *objective function* contains the variable which needs to be optimized. This optimization can be either to minimize or to maximize a variable.
The objective function consists of several variables and certain *constraints* may be assigned to these variables.

4.5.2 Implementation in this research

To explain the implementation in a good way an assumption has been made. This is the fact that a node with three connected network-lines is considered. When using the program it is also possible a node is connected with more than three lines.

As mentioned in the last paragraph the next step is to determine:

1. The objective function;
2. The constraints.

To be able to explain the content in a clear manor, first the constraints will be shown.
There are two constraining aspects: (1) static equilibrium in every node, and (2) a lower and upper boundary for the height of every node.

1. Static equilibrium in every node for the applied loading

A description of static equilibrium is: *'The sum of all forces acting on the object in static equilibrium must add to zero.'*
In formula-form:

$$\sum F = 0 \qquad (1)$$

The force situation of a node in the 3D diagram G consists of the applied loading P and the forces in the network-lines connected to the node (Figure 4.13). When considering vertical equilibrium, formula (1) transforms in:

$$F_{ji}^{V} + F_{ki}^{V} + F_{li}^{V} - P_i = 0$$
$$\rightarrow F_{ji}^{V} + F_{ki}^{V} + F_{li}^{V} = P_i \qquad (2)$$

However in this research the horizontal forces are most relevant, because the relation between the primal and dual grid is based on these forces. Therefore the vertical forces of (2) are now expressed in the horizontal components:

$$F_{ji}^{H} \cdot \frac{(z_i - z_j)}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} + F_{ki}^{H} \cdot \frac{(z_i - z_k)}{\sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}} + F_{li}^{H} \cdot \frac{(z_i - z_l)}{\sqrt{(x_i - x_l)^2 + (y_i - y_l)^2}} = P_i \qquad (3)$$

Next step is to use the lengths of the branches in the primal and dual grid, and the relation between the dual grid lengths and the horizontal force. First of all the length of the branch in the primal grid is H_{ij} and in the dual grid H^*_{ij}. The relation between H^*_{ij} and the corresponding horizontal force F^H_{ji} is scale factor ζ. In formula this is shown as:

$$F_{ji}^{H} = \zeta \cdot H_{ij}^{*}, \quad F_{ki}^{H} = \zeta \cdot H_{ik}^{*}, \quad F_{li}^{H} = \zeta \cdot H_{il}^{*} \qquad (4)$$

By using these relations in (3) and afterwards rearranging it so that it becomes a function of the branch lengths in both grids, the new formula is:

$$\left(\frac{H_{ij}^{*}}{H_{ij}} + \frac{H_{ik}^{*}}{H_{ik}} + \frac{H_{il}^{*}}{H_{il}}\right) \cdot z_i - \frac{H_{ij}^{*}}{H_{ij}} \cdot z_j - \frac{H_{ik}^{*}}{H_{ik}} \cdot z_k - \frac{H_{il}^{*}}{H_{il}} \cdot z_l - P_i \cdot r = 0 \qquad (5)$$

In (5) the new variable r is the inverse of the unknown scale ζ of the dual grid. By introducing constant C, (5) can be rewritten as:

$$C_i \cdot z_i + C_j \cdot z_j + C_k \cdot z_k + C_l \cdot z_l - P_i \cdot r \qquad (6)$$

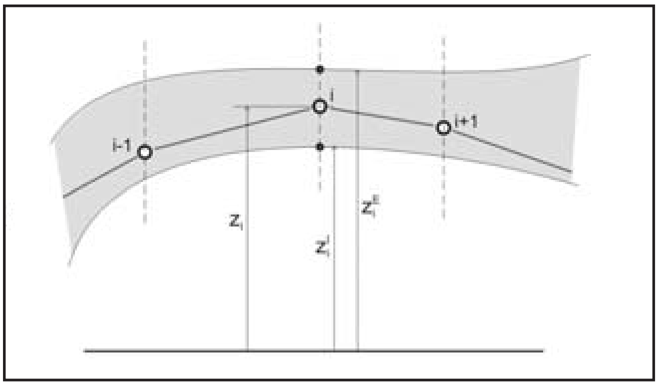


Figure 4.14 *- Boundary conditions of a node and its neighbouring nodes (Image courtesy of Philippe Block)*

Maximize/Minimize r										Matrix 1
Maximize/Minimize	K_i	z_i	$+ K_j$	z_j	$+ K_k$	z_k	$+ K_l$	z_l		
Constraints		z_i							$\leq$	z_i^E
		z_i							$\geq$	z_i^I
				z_j					$\leq$	z_j^E
				z_j					$\geq$	z_j^I
						z_k			$\leq$	z_k^E
						z_k			$\geq$	z_k^I
								z_l	$\leq$	z_l^E
								z_l	$\geq$	z_l^I

Figure 4.15 *- Start matrix when regarding only one node*

The constants C are:

$$C_i = \frac{H_{ij}^*}{H_{ij}} + \frac{H_{ik}^*}{H_{ik}} + \frac{H_{il}^*}{H_{il}}$$

$$C_j = -\frac{H_{ij}^*}{H_{ij}}$$

$$C_k = -\frac{H_{ik}^*}{H_{ik}}$$

$$C_l = -\frac{H_{il}^*}{H_{il}}$$

2. The lower and upper boundaries of the height for each node

To actually approximate the model, the height of every node needs to have some ability to be shifted up or down. This way the force-path, actual model and forces in it can be controlled. Therefore a lower and upper boundary is needed for each node (Figure 4.14). The solution has to lie within these boundaries. In formula:

$$z_i^I \le z_i \le z_i^E \qquad (7)$$

The objective function

In this research the objective function for the linear optimization problem is r – the inverse of the scale factor ζ. This variable r either has to be minimized or maximized, to obtain the envelope in which the model has to be placed. The result of this objective function is respectively the shallowest and deepest solution – still within the limits – for a chosen combination of primal and dual grid.

4.5.3 Solution for one point

Determination of the general problem for one node

The objective function is to maximize or minimize the value of r, where r is described as:

$$r = \frac{1}{P_i} \cdot (C_i \cdot z_i + C_j \cdot z_j + C_k \cdot z_k + C_l \cdot z_l)$$

$$\rightarrow r = \frac{C_i}{P_i} \cdot z_i + \frac{C_j}{P_i} \cdot z_j + \frac{C_k}{P_i} \cdot z_k + \frac{C_l}{P_i} \cdot z_l$$

Next step is to introduce a new variable – in this case K – to simplify the process to find the solution. All values in the description of K are given on forehand and are constant during the linear optimization process. They can be altered after the optimization has been performed.

$$r = K_i \cdot z_i + K_j \cdot z_j + K_k \cdot z_k + K_l \cdot z_l \qquad (8)$$

In which the constants K are described as:

$$K_i = \frac{C_i}{P_i} = \frac{\left(\frac{H_{ij}^*}{H_{ij}} + \frac{H_{ik}^*}{H_{ik}} + \frac{H_{il}^*}{H_{il}}\right)}{P_i}$$

$$K_j = \frac{C_j}{P_i} = -\frac{H_{ij}^*}{H_{ij} \cdot P_i}$$

$$K_k = \frac{C_k}{P_i} = -\frac{H_{ik}^*}{H_{ik} \cdot P_i}$$

$$K_l = \frac{C_l}{P_i} = -\frac{H_{il}^*}{H_{il} \cdot P_i}$$

Determination of the boundary conditions/constraints

Formula (7) is divided into two constraints: a) the ≤ constraints, and b) the ≥ constraint:

$$z_i^I \le z_i \le z_i^E \quad \Rightarrow$$

$$(a)\ z_i \le z_i^E$$

$$(b)\ z_i \le z_i^I$$

These constraints are applied to the considered node *I* and to the nodes connected to the force network lines corresponding with node *i*. As a result the amount of constraints is dependent on the number of lines connected to the node. In formula: $noc = 2 \cdot (1 + nol)$

In this formula *noc* is the Number of Constraints and *nol* is the Number of Lines; e.g. a node with three lines has 8 constraints and a node with four lines has 10 constraints. More information about how to make these formulas suitable for solving is found in Appendix C.

Generate the matrix

Objective function:

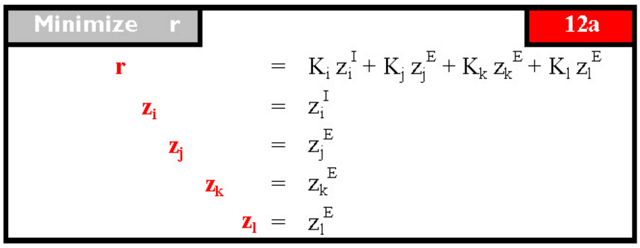
$$r = K_i \cdot z_i + K_j \cdot z_j + K_k \cdot z_k + K_l \cdot z_l$$

Constraints:

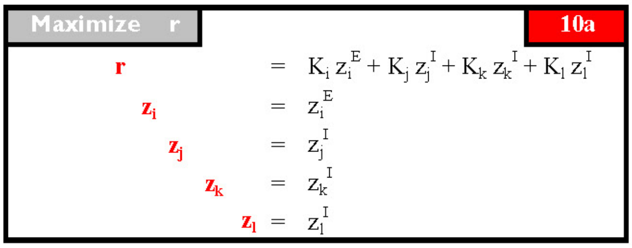
$$(a)\ z_i \le z_i^E$$

$$(b)\ z_i \le z_i^I$$

Figure 4.16 *- Final matrices when regarding only one node*

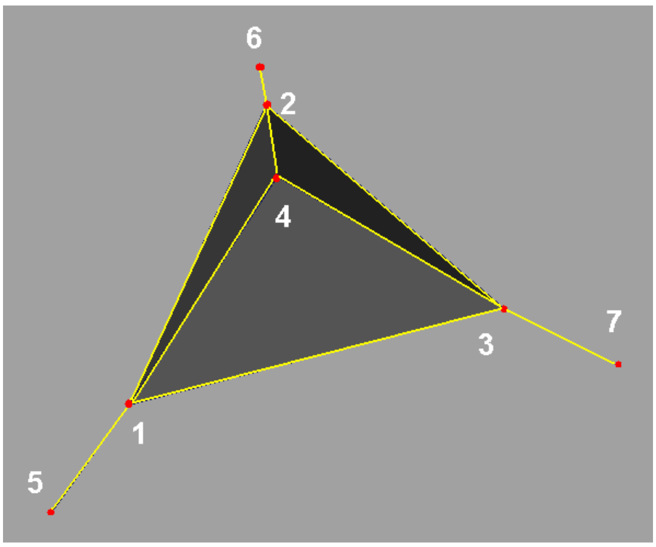


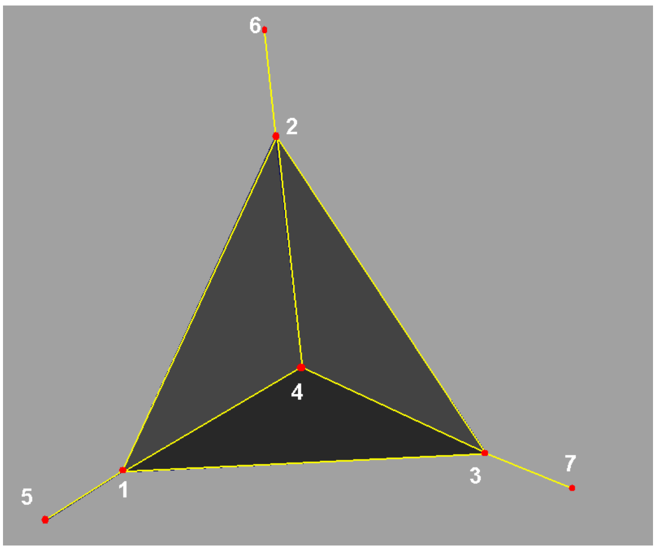
(a) *- Minimizing r*



(b) *- Maximizing r*

Figure 4.17 *- 3D model used in the example to show the Simplex method for more nodes. On the right the primal grid (topview) is shown*





The result written in matrix notation are found in Figure 4.15.

<u>Solving the matrix</u>

The important steps to solve this matrix are:

1. Write all constraints in ≤ form. This is achieved by introducing an extra variable y;
2. Write all constraints in = form. This is achieved by introducing an extra variable x;
3. Minimize all y variables until they are all 0. After this they can be eliminated from the matrix;
4. Solve the remaining matrix until r has either been maximized or minimized. Exact details of this procedure can be found in Appendix C.

The results for one single node are not special, in the sense that heights of the nodes either go to maximum or minimum allowed height (Figure 4.16). The explanation for this is the fact that only one node is regarded. As a result the height values can obtain any value – and so also the minimum and maximum boundary-values. The node is not dependent on other nodes. In other words: when maximizing r, the deepest solution is obtained, which is reached when node i has maximum height and all other nodes the minimum values. Following this reasoning, the result when minimizing r – which results in the shallowest result – is minimum height for node i and maximum height for the other nodes. This will change when the whole network is regarded (§4.5.4).

4.5.4 Solution for all points

<u>Difference with solving for one point</u>

In this case all nodes are considered, instead of only one. Consequence is that certain nodes become dependent on others. A global matrix is created, which consists of all local matrices per single node. The steps needed for this are shown below.

Another aspect that has to be considered is the fact that all foundation points – or in practical point of view: the points that are connected to the ground – have only one possible height.

<u>Additional steps</u>

1. A loop is needed to run through all nodes and create a local matrix for every node;
2. A global matrix is formed, consisting of a combination of the local matrices;
3. The global matrix is solved, as is explained before for the local matrix in § 4.5.3 and as is explained in Appendix C.

4.5.5 An example

Consider the figure shown in Figure 4.17. The structure consists of four normal nodes (1–4) and three foundation nodes (5–7). First we consider the four nodes separately and create the local objective functions.

Node 1

Nodes around node 1: 2, 3, 4 and 5

$$r_{N1} = (K_{1,\,N1}\cdot z_1) + (K_{2,N1}\cdot z_2) + (K_{3,\,N1}\cdot z_3) + (K_{4,\,N1}\cdot z_4) + (K_{5,\,N1}\cdot z_5)$$

In which $K_1 - K_5$ are:

$$K_{1,N1} = \frac{C_1}{P_1} = \frac{\left(\frac{H^*_{12}}{H_{12}} + \frac{H^*_{13}}{H_{13}} + \frac{H^*_{14}}{H_{14}} + \frac{H^*_{15}}{H_{15}}\right)}{P_1}$$

$$K_{2,N1} = \frac{C_2}{P_1} = -\frac{H^*_{12}}{H_{12}\cdot P_1}$$

$$K_{3,N1} = \frac{C_3}{P_1} = -\frac{H^*_{13}}{H_{13}\cdot P_1}$$

$$K_{4,N1} = \frac{C_4}{P_1} = -\frac{H^*_{14}}{H_{14}\cdot P_1}$$

$$K_{5,N1} = \frac{C_5}{P_1} = -\frac{H^*_{15}}{H_{15}\cdot P_1}$$

Node 2

Nodes around node 2: 1, 3, 4 and 6

$$r_{N2} = (K_{2,\,N2}\cdot z_2) + (K_{1,N2}\cdot z_1) + (K_{3,\,N2}\cdot z_3) + (K_{4,\,N2}\cdot z_4) + (K_{6,\,N2}\cdot z_6)$$

Generate global matrix to be solved:

$$\begin{pmatrix} K_{1,N1} & K_{2,N1} & K_{3,N1} & K_{4,N1} & K_{5,N1} & 0 & 0 \\ K_{1,N2} & K_{2,N2} & K_{3,N2} & K_{4,N2} & 0 & K_{6,N2} & 0 \\ K_{1,N3} & K_{2,N3} & K_{3,N3} & K_{4,N3} & 0 & 0 & K_{7,N3} \\ K_{1,N4} & K_{2,N4} & K_{3,N4} & K_{4,N4} & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{pmatrix} = \begin{pmatrix} r_{N1} \\ r_{N2} \\ r_{N3} \\ r_{N4} \end{pmatrix} = r_{total}$$

Figure 4.18 *- Global matrix to solve for r and z-values*

In which $K_1 - K_4$ and K_6 are:

$$K_{2,N2} = \frac{C_1}{P_2} = \frac{\left(\frac{H^*_{21}}{H_{21}} + \frac{H^*_{23}}{H_{23}} + \frac{H^*_{24}}{H_{24}} + \frac{H^*_{26}}{H_{26}}\right)}{P_2}$$

$$K_{1,N2} = \frac{C_1}{P_2} = -\frac{H^*_{21}}{H_{21} \cdot P_2}$$

$$K_{3,N2} = \frac{C_3}{P_2} = -\frac{H^*_{23}}{H_{23} \cdot P_2}$$

$$K_{4,N2} = \frac{C_4}{P_2} = -\frac{H^*_{24}}{H_{24} \cdot P_2}$$

$$K_{6,N2} = \frac{C_6}{P_2} = -\frac{H^*_{26}}{H_{26} \cdot P_2}$$

Node 3

Nodes around node 3: 1, 2, 4 and 7

$$r_{N3} = (K_{3,\,N3} \cdot z_3) + (K_{1,\,N3} \cdot z_1) + (K_{2,\,N3} \cdot z_2) + (K_{4,\,N3} \cdot z_4) + (K_{7,\,N3} \cdot z_7)$$

In which $K_1 - K_4$ and K_7 are:

$$K_{3,N3} = \frac{C_3}{P_3} = \frac{\left(\frac{H^*_{31}}{H_{31}} + \frac{H^*_{32}}{H_{32}} + \frac{H^*_{34}}{H_{34}} + \frac{H^*_{37}}{H_{37}}\right)}{P_3}$$

$$K_{1,N3} = \frac{C_1}{P_3} = -\frac{H^*_{31}}{H_{31} \cdot P_3}$$

$$K_{2,N3} = \frac{C_2}{P_3} = -\frac{H^*_{32}}{H_{32} \cdot P_3}$$

$$K_{4,N3} = \frac{C_4}{P_3} = -\frac{H^*_{34}}{H_{34} \cdot P_3}$$

$$K_{7,N3} = \frac{C_7}{P_3} = -\frac{H^*_{37}}{H_{37} \cdot P_3}$$

Node 4

Nodes around node 4: 1, 2 and 3

$$r_{N4} = (K_{4,\,N4} \cdot z_4) + (K_{1,\,N4} \cdot z_1) + (K_{2,\,N4} \cdot z_2) + (K_{3,\,N4} \cdot z_3)$$

In which $K_1 - K_4$ and K_7 are:

$$K_{4,N4} = \frac{C_4}{P_4} = \frac{\left(\frac{H^*_{41}}{H_{41}} + \frac{H^*_{42}}{H_{42}} + \frac{H^*_{43}}{H_{43}}\right)}{P_4}$$

$$K_{1,N4} = \frac{C_1}{P_4} = -\frac{H^*_{41}}{H_{41} \cdot P_4}$$

$$K_{2,N4} = \frac{C_2}{P_4} = -\frac{H^*_{42}}{H_{42} \cdot P_4}$$

$$K_{3,N4} = \frac{C_4}{P_4} = -\frac{H^*_{43}}{H_{43} \cdot P_4}$$

All local objective functions are known. Next step is to combine them and create the global objective functions and the belonging constraints.

Global objective function

The global objective function is the combination of all separate objective functions of all nodes, where the individual r for each node has to be the same:

$$r_{total} = r_{N1} = r_{N2} = r_{N3} = r_{N4}$$

And as a result:

$$r_{total} = r_{N1} = (K_{1,\,N1} \cdot z_1) + (K_{2,N1} \cdot z_2) + (K_{3,\,N1} \cdot z_3) + (K_{4,\,N1} \cdot z_4) + (K_{5,\,N1} \cdot z_5)$$

And:

$$r_{total} = r_{N2} = (K_{1,N2} \cdot z_1) + (K_{2,\,N2} \cdot z_2) + (K_{3,\,N2} \cdot z_3) + (K_{4,\,N2} \cdot z_4) + (K_{6,\,N2} \cdot z_6)$$

And:

$$r_{total} = r_{N3} = (K_{1,N3} \cdot z_1) + (K_{2,\,N3} \cdot z_2) + (K_{3,\,N3} \cdot z_3) + (K_{4,\,N3} \cdot z_4) + (K_{7,\,N3} \cdot z_7)$$

And:

$$r_{total} = r_{N4} = (K_{1,N4} \cdot z_1) + (K_{2,\,N4} \cdot z_2) + (K_{3,\,N4} \cdot z_3) + (K_{4,\,N4} \cdot z_4)$$

This is shown in matrix form in Figure 4.18.

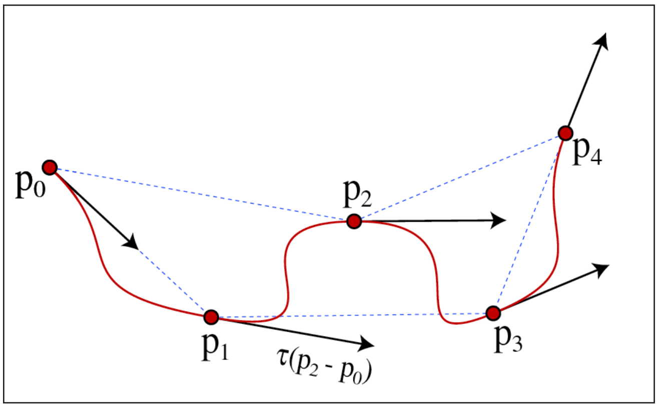


Figure 4.19 *- A Catmull-Rom spline*

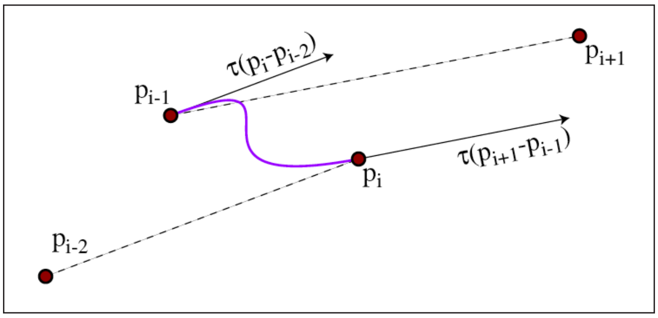


Figure 4.20 *- Catmull-Rom spline derivation*

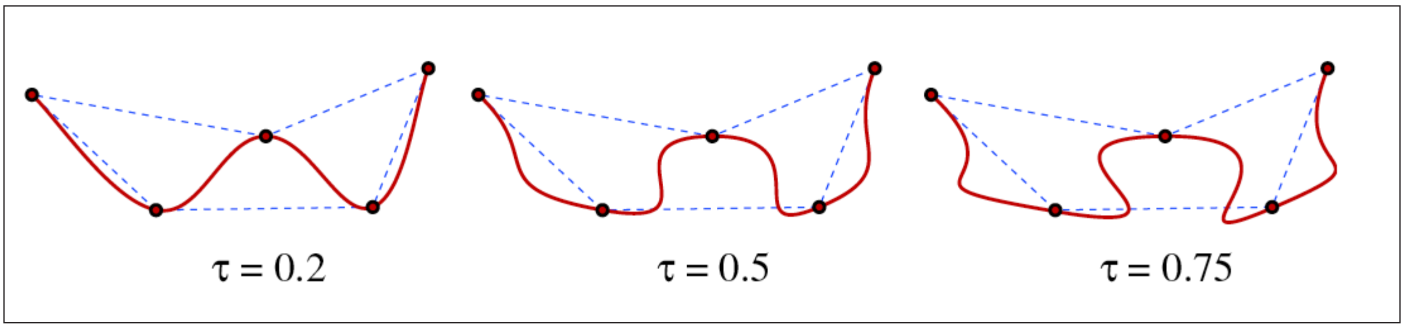


Figure 4.21 *- The effect of tension variable* τ

The steps to solve this matrix with an interation are:

1. Take a random internal node and calculate r by choosing a random value for the z-values;
2. Use the values for r and z in the calculation of the next random node;
3. If the value for on the of the Z-values is out of boundary, adapt r until it is OK and restart the iteration.

Continue this process of iteration until all values for r are equal and the Z-values are within their boundaries.

Global constraints

The constraints belonging to the objective function are:

1. $z_1 \leq z_1^E$
2. $z_1 \geq z_1^I$
3. $z_2 \leq z_2^E$
4. $z_2 \geq z_2^I$
5. $z_3 \leq z_3^E$
6. $z_3 \geq z_3^I$
7. $z_4 \leq z_4^E$
8. $z_4 \geq z_4^I$
9. $z_5 = 0$
10. $z_6 = 0$
11. $z_7 = 0$

Solution

The global matrix that has to be solved is shown in Figure 4.18. After taking all steps of the Simplex method (Appendix C) the final result is obtained.

[This linear optimization formula is not integrated in the current version of the script. The matrix is formed and the boundary conditions are calculated. Both can be exported and seen in a text file (§ 5.11)]

4.6 Theory of Catmull Rom splines

This theory is applied to determine the height and angles of each individual brick in the brick pattern.

THEORY

The theory uses the coordinates and tangent of several points to determine the curve between these points. This curve is described by a formula, consisting of the coordinates of the four points and a factor tau.

The curve always passed through these points and the curve is adapted according to the position of each of the four points.

The last remark is the reason to use Catmull Rom splines: when a position of a point is changed, the curve is changed as well and still passes all points.

FORMULAE

To calculate the angle for each individual brick in the pattern, the theory of Catmull-Rom splines is used. [15]

Catmull-Rom splines are a family of cubic interpolating splines formulated in such way that the tangent at each point p_i (called α_i) is calculated using the tangent of the previous and next point on the spline: p_{i+1} (α_{i+1}) and p_{i-1} (α_{i-1}). (Figure 4.19)

Consider a single Catmull-Rom segment, p(s). Suppose it is defined by four control points, p_{i-2}, p_{i-1}, p_i, and p_{i+1}. (Figure 4.20)

The tangents for the non-border points (p_{i-1} and p_i) are:

$$p_{i-1} = \tau * (p_{i-2} - p_i)$$
$$p_i = \tau * (p_{i-1} - p_{i+1})$$

Note that the tangent at the border points (p_0 and p_{end}) is not clearly defined. For those points one of the two angles is not known. The tangent is set to:

$$p_0 = \tau * (p_1 - (p_1 + (p_2 - p_1)))$$
$$p_{end} = \tau * (p_{end-1} - (p_{end-1} + (p_{end-2} - p_{end-1})))$$

The formula for the curve segment between p_{i-1} and p_i is:

$$\begin{aligned} p(s) &= c_0 + c_1 \cdot u + c_2 \cdot u^2 + c_3 \cdot u^3 \\ &= \sum_{k=0}^{3} c_k \cdot u^k \end{aligned}$$

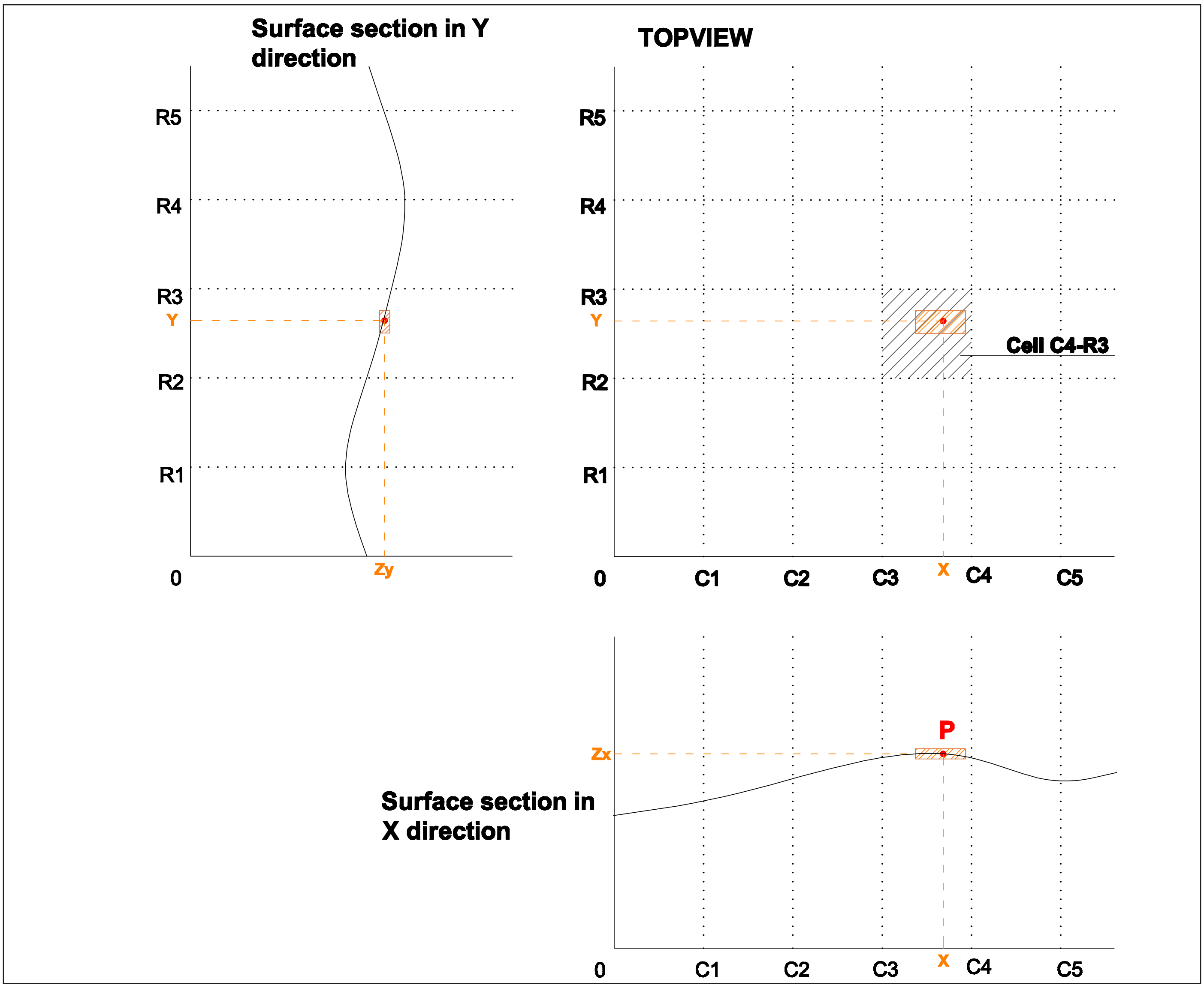


Figure 4.22 - *Point P, which is the centerpoint of a brick. It is characterisied by a Cell and 3 coordinates: x, y and z*

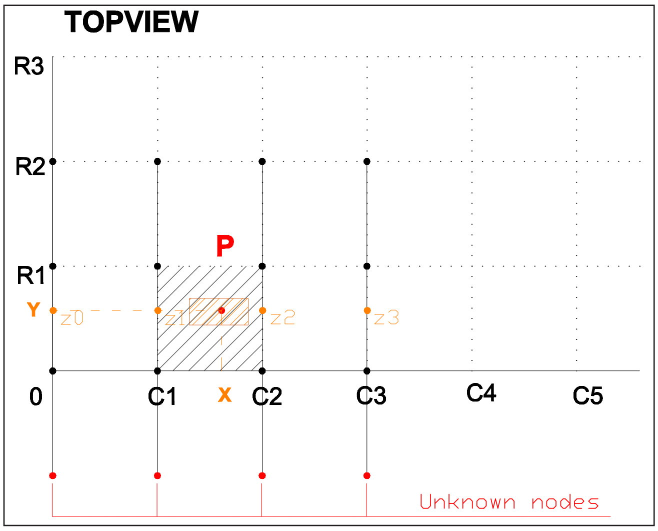


Figure 4.23 - *Unknown value for specific nodes at the border of the grid*

In which:

$$c_0 = p_{i-1}$$
$$c_1 = (-\tau)\cdot p_{i-2} + (\tau)\cdot p_i$$
$$c_2 = (2\tau)\cdot p_{i-2} + (\tau-3)\cdot p_{i-1} + (3-2\tau)\cdot p_i + (-\tau)\cdot p_{i+1}$$
$$c_3 = (-\tau)\cdot p_{i-2} + (2-\tau)\cdot p_{i-1} + (\tau-2)\cdot p_i + (\tau)\cdot p_{i+1}$$

The first derivative of this function determines the angle of the curve in that point.

PARAMTER TAU (τ)

The parameter τ is known as *tension* and it affects how sharply the curve bends at the (interpolated) control points (Figure 4.21). In general it is set to 0.5.

Though when using a brick pattern, the curvature of the surface is limited by the angle between the brick. The lower the curvature, the higher the chance the shell is '*buildable*'. Therefore it is better to lower the value of τ. The value has been set to 0.25, and can be altered only in the script code.

APPLICATION OF CATMULL ROM SPLINES IN 3D GRID

To explain the theory, some variables are introduced (Figure 4.22):

- A brick is regarded as Point, which is the center point of the brick;
- A polygon in the grid is regarded as Cell;
- The location of a Cell is a combination of a row and column.

Steps

1. Find the Cell in which the Point is located (Figure 4.22, righttop);

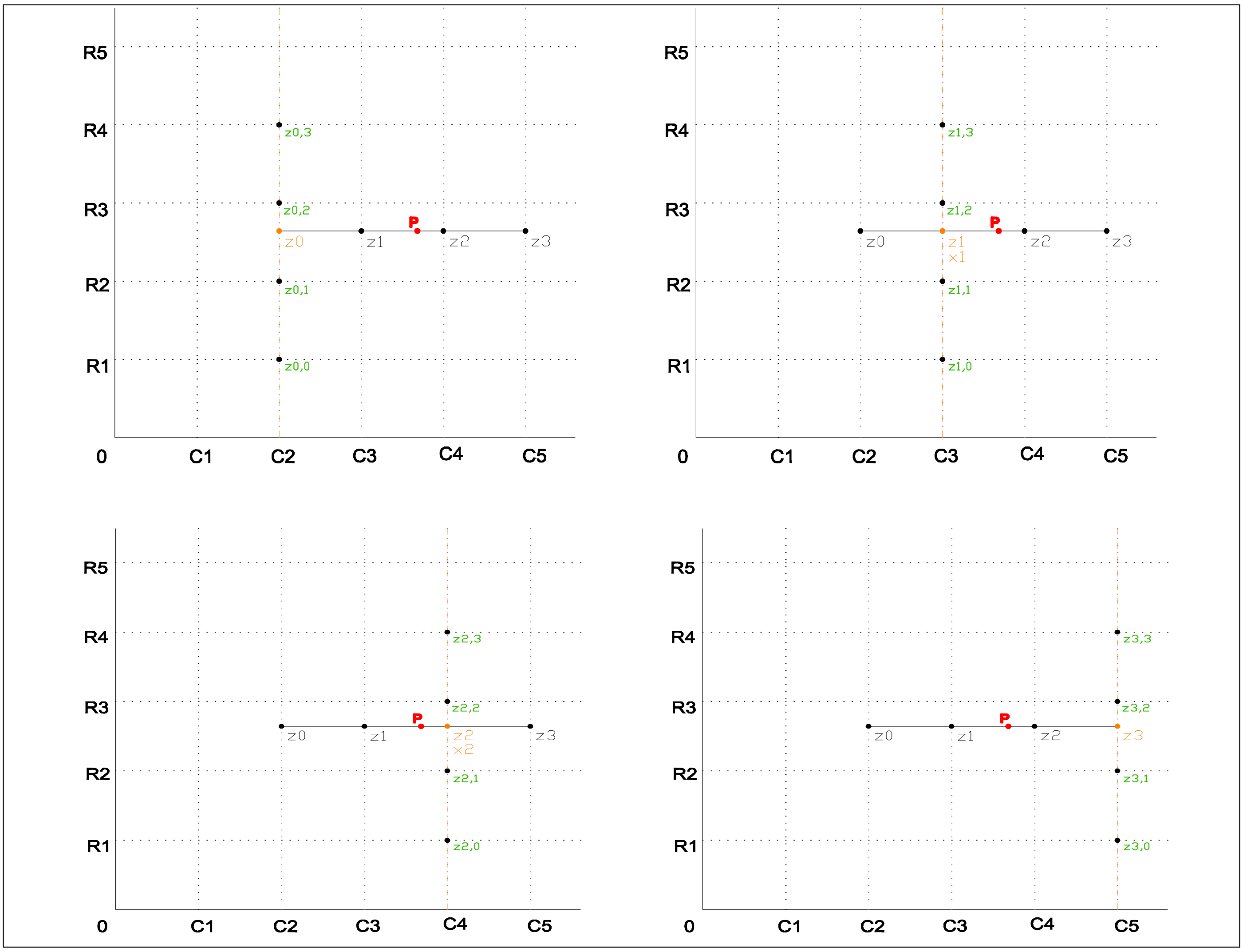


Figure 4.24 *- Determine z0, z1, z2 and z3*

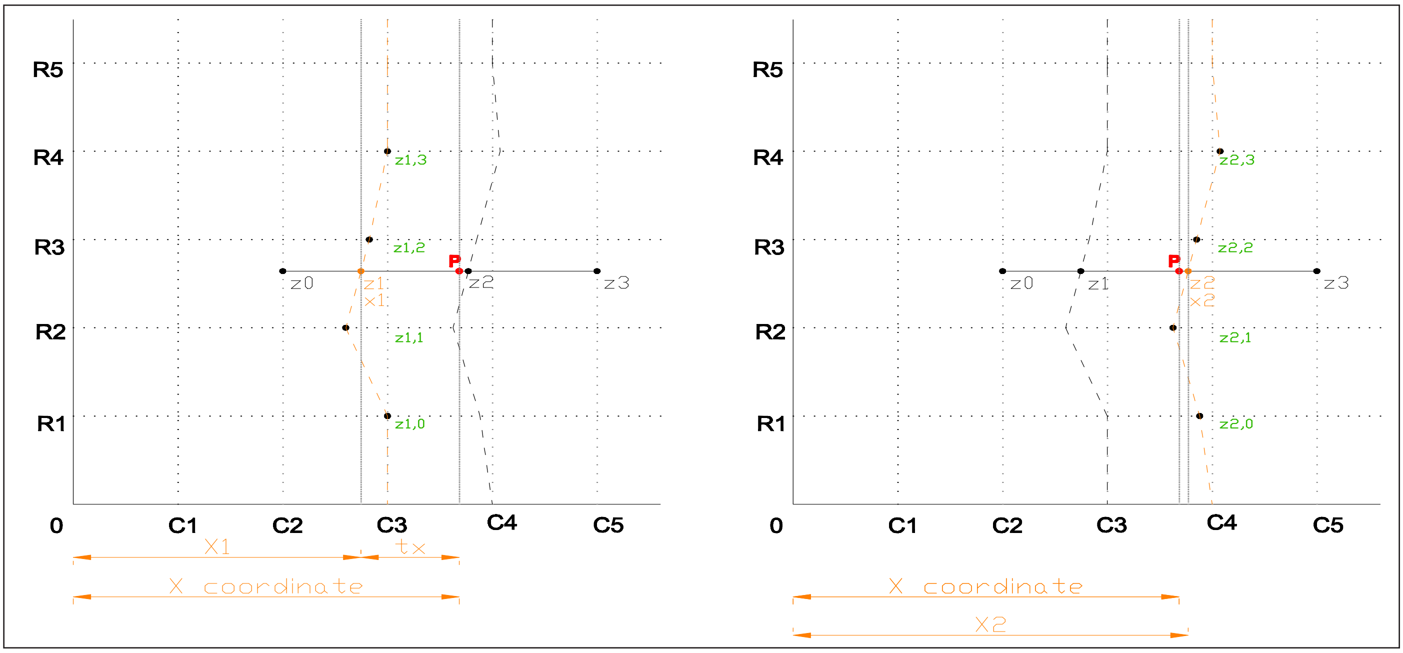


Figure 4.25 *- Determine x values for the third and second point*

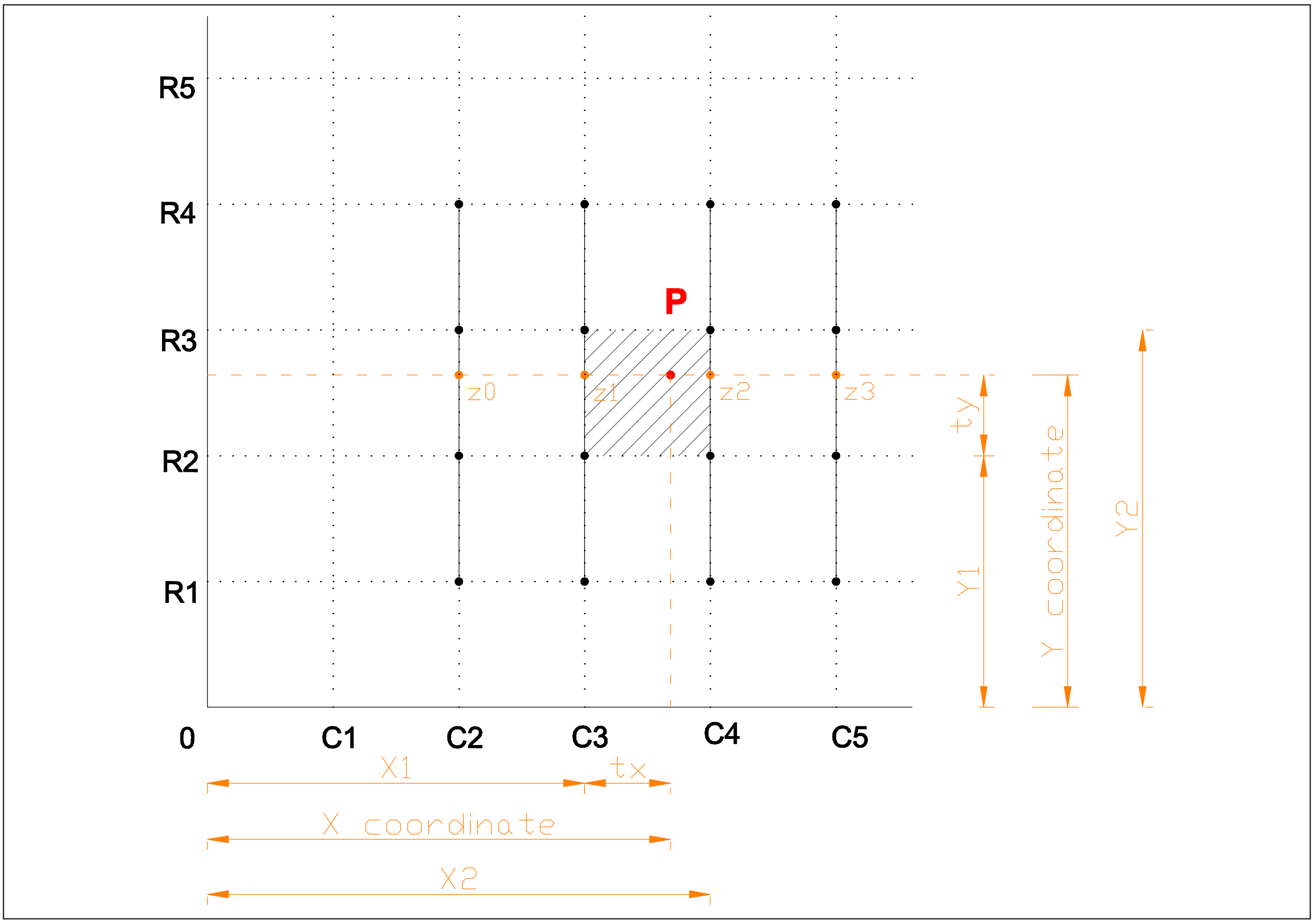


Figure 4.26 *- Determine height of point P using the four heights z0 - z3*

2. Check if there are enough neighbouring nodes (values) to calculate height and angles of the Point. If not: set a value for the missing points (Figure 4.23);
3. Determine the four z-values of the column lines around the Cell in which the Point is located (Figure 4.24);
4. Determine the x-values of the second and third point. (Figure 4.25). These values are used to determine the tx value, which is needed to calculate the height for point P;
5. With the four z-values, (step3) and the two x-values (step4) the z-value of the Point is calculated (Figure 4.26). In addition the angle of the surface in both directions must be calculated and checked. These angles are the angles of the brick in that Point (Figure 4.27).

[This theory is implemented to create the lines and surface of the shape.
The step to calculate the angles of each individual brick with this theory has not been implemented yet. In the current version of the prototype the angles are calculated using the known values of the angles between the nodes and interpolating between them. As a result the brick pattern is not following the exact surface everywhere.]

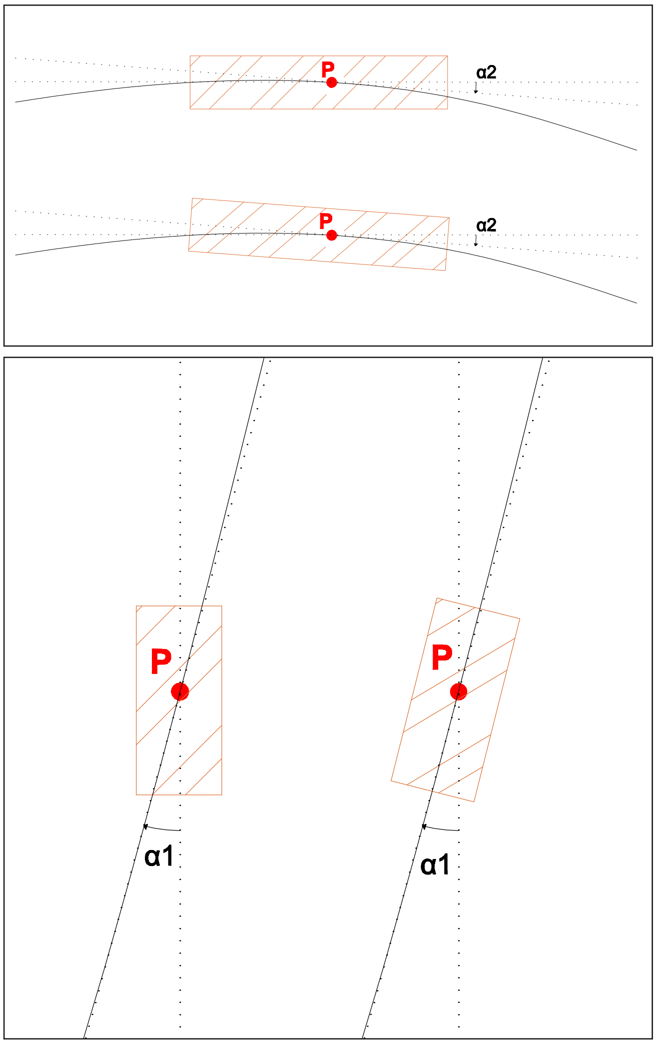
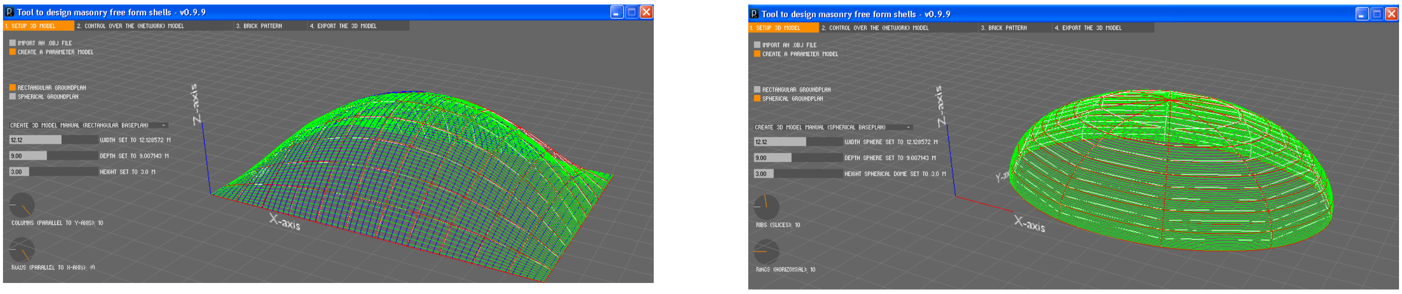


Figure 4.27 *- Angles of brick*



***(a)** - Rectangular grid* ***(b)** - Spherical grid*

***Figure 4.28** - Appearance of the two pattern techniques*

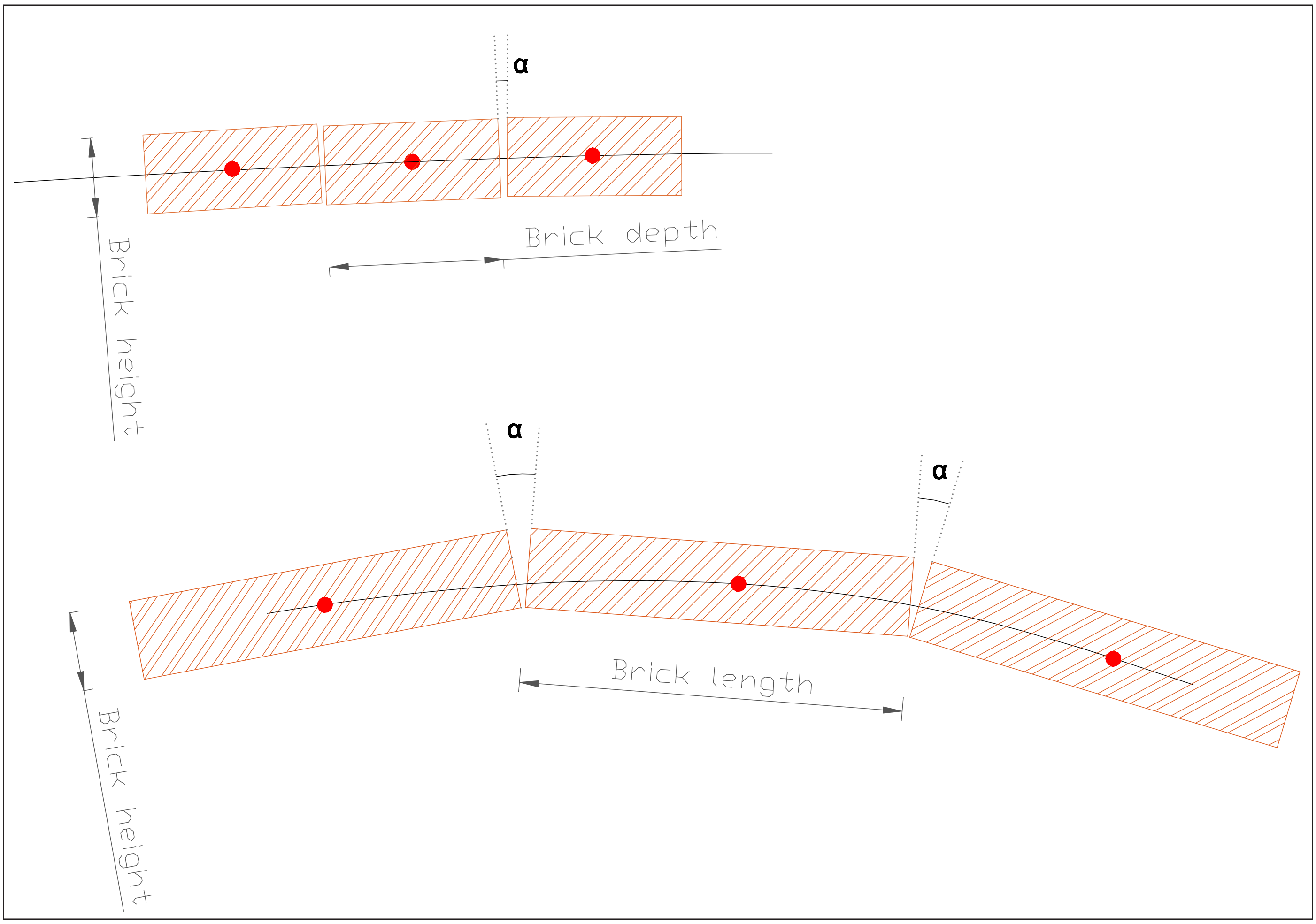


***Figure 4.29** - Angle between two bricks*

4.7 Masonry pattern generation

Two types of patterns can be generated, depending on the force network that has been chosen:
- A linear pattern (§ 4.7.1);
- A spherical pattern (§ 4.7.2).

Which pattern is chosen by the application depends mainly on the shape of the base plan: when the plan is rectangular, it is convenient to use the linear force network and the linear pattern. However when the plan is elliptical, the spherical force network and spherical pattern are more convenient. The appearance of the two options is quite different, even when the common dimension parameters - the base plan width, base plan depth and height of the shell - are the same (Figure 4.28).

4.7.1 Linear rectangular pattern

This pattern consists a grid of lines in X and Y direction.
There are two directions:
- The main direction;
- The secondary direction.

<u>MAIN DIRECTION</u>
The main direction is determined by the curvature of the surface: the direction in which the curvature is smallest is the main direction. The curvature is determined by the height-width ratio. The wider the base plan, the shallower the surface, which means a smaller curvature.
In other words: the main direction is determined by the *longer* side of the base plan.
This direction is important, because it determines in which direction the grid is generated.

<u>CHECK OF PATTERN: MAXIMUM ANGLE BETWEEN BRICKS</u>
The brick pattern has to be realistic. Therefore the bricks can not cross each other. On the other hand the gap between two bricks is also limited to 16 mm, because otherwise the bonding layer becomes too thick.
The maximum angle depends on the height of the brick. (Figure 4.29). This angle must be determined and a warning has to be given when the angle is too big. This is not yet incorporated in the prototype.

<u>STEPS</u>
Step 1
Create lines next to each other, with a spacing of one brick width. These form the centre-lines of the bricks.

Step 2
Determine the length of the lines, and divide it by the brick length. The amount of bricks is now known.

Step 3 - Loop through all main direction lines:
Step 3a
Determine the X and Y coordinate of the first brick and calculate the corresponding height and angles, using the theory of § 4.6.

Step 3b
Use the current X and Y coordinate and the maximum allowed angle between two bricks to determine the coordinates for the next brick and continue with step 3a until the whole line is filled with bricks.

Step 4
The result is a masonry pattern of long lines of brick (Figure 4.30 - left).
One disadvantage is big openings when the curvature of the surface is too high (Figure 4.31).

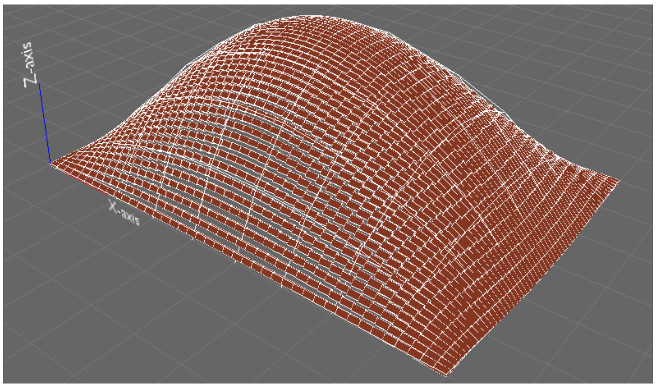
4.7.2 Spherical pattern

This pattern consists of elliptical rings, which form the centre line for the brick.

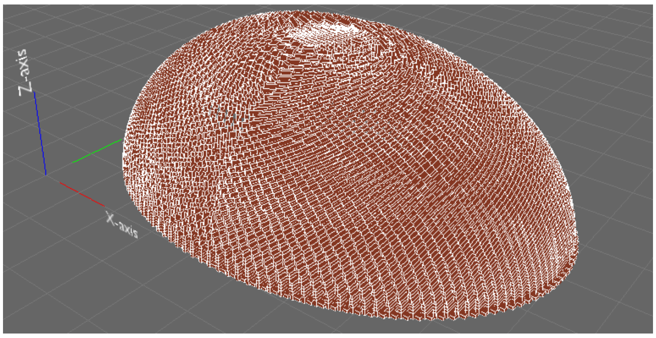
<u>STEPS</u>
Step 1
Create rings above each other with a spacing of one brick height. These form the centre-lines of the bricks.

Step 2
Determine the length of the rings, and divide it by the brick length. The amount of bricks is now known.

Step 3 - Loop through all rings:
Step 3a



(a) *- Rectangular masonry pattern*



(b) *- Spherical masonry pattern.*

Figure 4.30 *- The two types of brick patterns.*

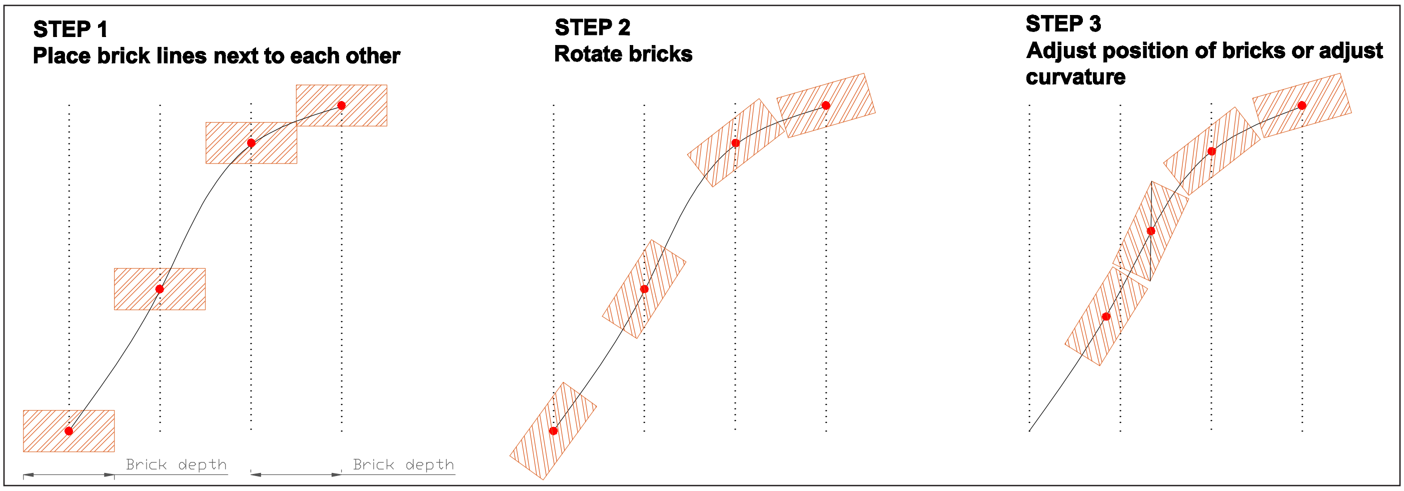


Figure 4.31 *- Proposed solution to the problem of gaps in the pattern*

Determine the X and Y coordinate of the first bricks and calculate the corresponding height and angles, using the theory of § 4.6.

Step 3b
Use the current X and Y coordinate to determine the coordinates for the next brick and continue with step 3a until the whole line is filled with bricks.

Step 4
The result is a masonry pattern of rings of brick (Figure 4.30 - right).

One disadvantage is gaps between the rings when the curvature of the surface is too high (Figure 4.31).
Another disadvantage are the top rings. Their radius is too small to be able to fit bricks in it and still make sure the angle between the bricks is not too big (Figure 4.32).
Therefore either a closing element is needed or the top of the structure is left open. The last option is not recommended, because it has a bad influence on the force flow and moreover it is often not desired when the shell covers a space.
When a closing element is used, this element does have influence on the force flow, since it has adds a loading to the top ring nodes. This loading has to be added in the analysis.

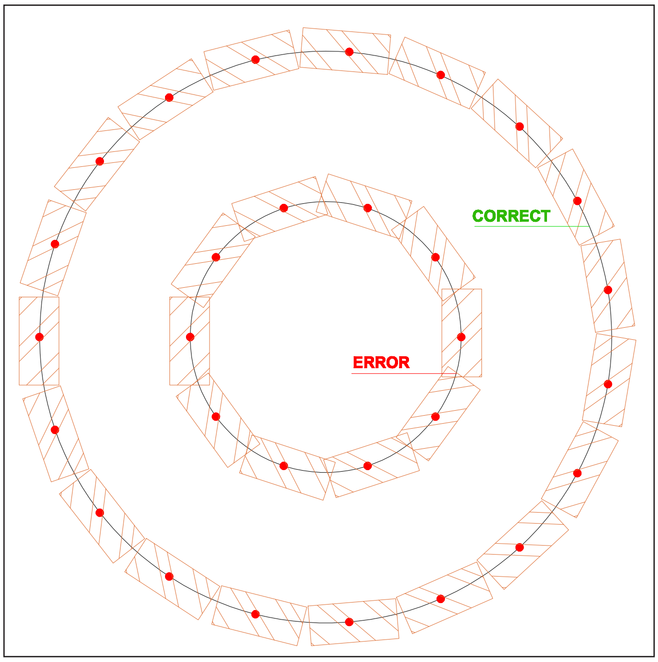


Figure 4.32 *- Problem of the top rings with small radius. The solution is a closing element that replaces these rings.*

4.8 Limitations

- When nodes are located to close in the primal grid, the result of the dual grid is unrealistic. An exact value for this limitation has not been obtained. Therefore it is recommended to use the program and change the primal grid with a *common sense*. When the dual grid is not realistic, the results of the analysis should not be used;

- The grid is limited to consist of maximum 225 nodes (15x15 grid), to assure a smooth performance and to limit the size of the matrix to calculate stresses. In theory the matrix to calculate the stresses and forces can be of unlimited size. However the analysis takes longer with increasing amount of nodes.

- The dimensions of the grid are limited to 15.0x15.0x4.0m. This assures that the limited grid size is still applicable and realistic;

- The angle between the bricks must be implemented as a check whether the masonry pattern is realistic.
  [This is not implemented yet in the the current prototype]

- The gap between the lines of brick in the linear pattern must be repositioned so that gaps in the perpendicular direction are closed.
  [This is not implemented yet in the the current prototype]

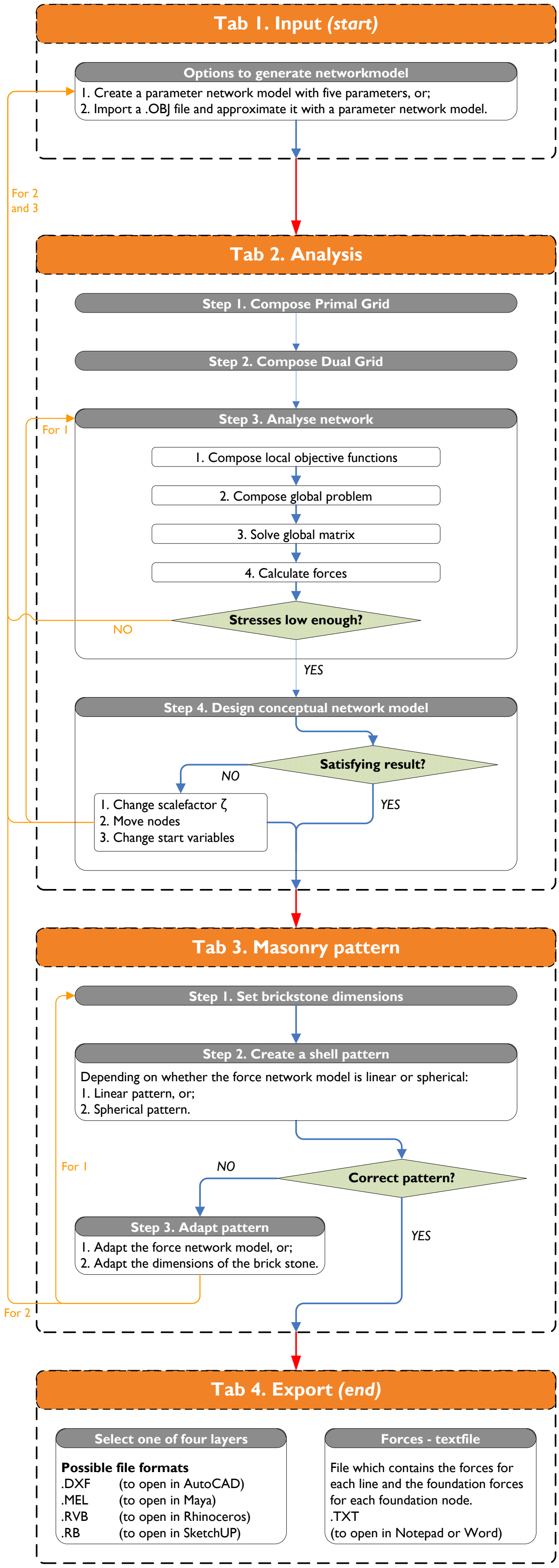


Figure 5.1 *- Main flow chart of the new application*

CHAPTER 5 THE NEW APPLICATION

Introduction

In this chapter the prototype of the new application is presented. The theories of Chapter 4 are the basis of the application.

First the main flowchart of the application is explained (§ 5.1). To operate and use the application and display the results, a user Interface (UI) has been created (§ 5.2).
The main content of the application is the 3D model. There are several options to display it (§ 5.3).
To make the application interactive, it is relevant the user can affect the result and outcome of the application. Therefore the user has to be able to change and adapt certain variables, such as the nodal loading, the load case and the position of the nodes (§ 5.4).
In the following paragraphs the main flow chart is explained more in detail:

- The input options (§ 5.5);
- The procedures within the application (§ 5.6);
- The main elements of the model and used as input for the procedures (§ 5.7);
- The output options (§ 5.8).

The next contribution to this chapter is description regarding the relation between the masonry pattern and force network (§ 5.9).
More information regarding the application is found in the *user manual* (Appendix F).
Finally the last paragraph of this chapter deals with the place of the application within the workflow and within the design process to explain how the application is used in practice (§ 5.10).

[In the most recent version of the report the current status of the application and its script is included at the end of the chapter (§ 5.11). In the other paragraphs the text concerning the status of the discussed aspect is highlighted in red]

5.1 Main flowchart for the new application

To explain the steps of the application in a clear way flowcharts are used.
First of all an overview of the steps of the application is presented in the main flowchart (Figure 5.1).
It consists of four main groups, which individually consist of their own elements and procedures. The flow chart for each individual group is given in the corresponding paragraph:

1. Start – input (§ 5.5);
2. Analysis – create the primal and dual grid and analyse the model (§ 5.6 and § 5.7);
3. Generate the masonry pattern (§ 5.7);
4. Export specific layers (§ 5.8).

The important steps in the script are:

1. Analysis (step 2 of the four steps);
2. Surface generation and masonry pattern (step 3 of the four steps).

[In the current version of the prototype the script of both of these steps contains errors. These errors and bugs are mentioned at each step that is explained in the previous and this chapter]

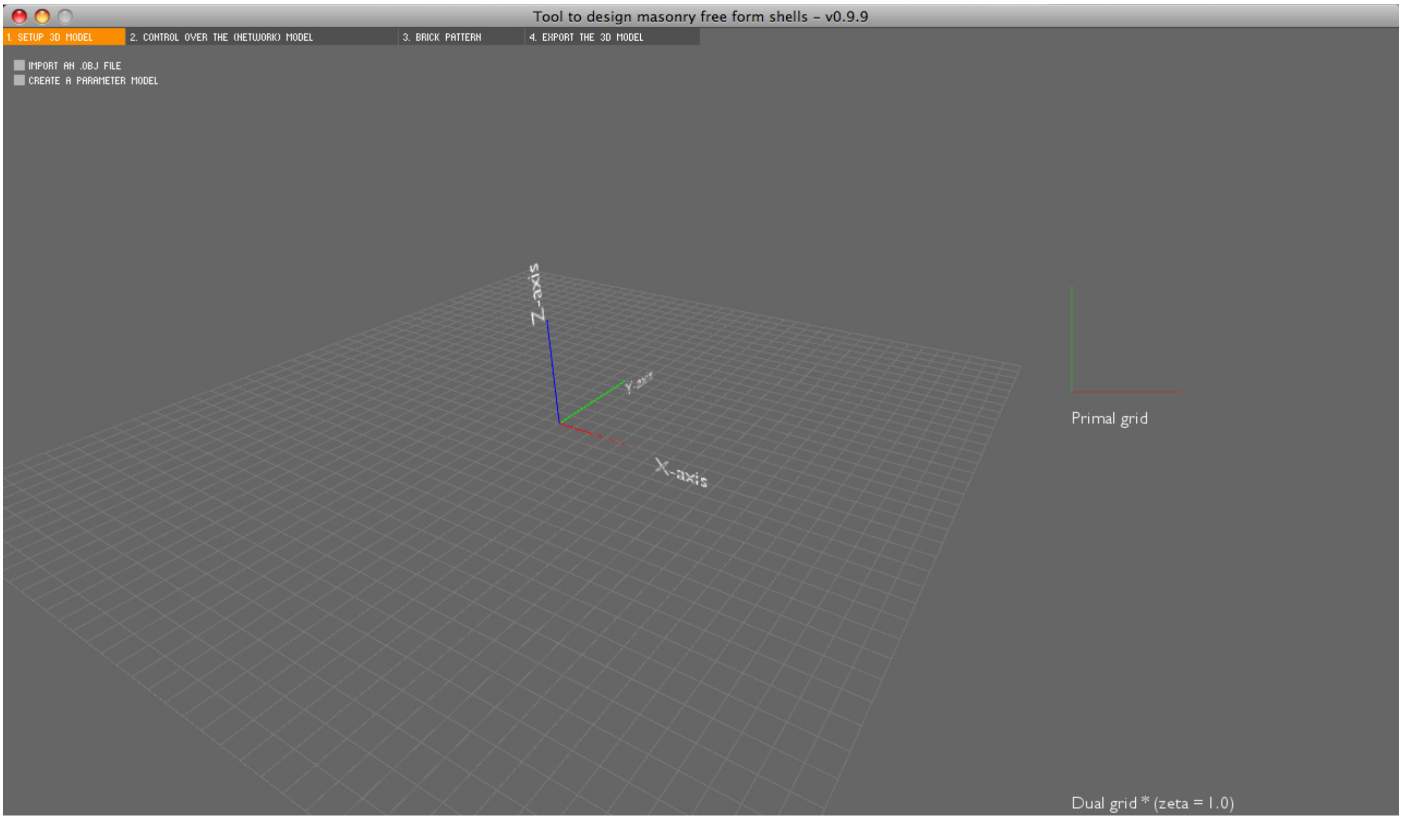


Figure 5.2 *- Current user interface*

5.2 User interface

A relevant aspect to assure a clear and easy use of the application is the user interface (UI). The appearance has to be clear. Using it has to be easy and in a logical sense. Finally the results have to be presented in such a way that they are well comprehensible.
The application uses several schemes and diagrams, that should all be well visible: when the force network is determined and analysed the primal and dual grid are visible. In other situations, for instance when files are exported, these diagrams can be turned off. This assures a good and orderly UI.

The latest version of the user interface (Figure 5.2) is divided into four tabs (displayed in the top of the UI). These tabs correspond with the four groups, as mentioned in § 5.1:
1. Setup force network;
2. Analysis and design the conceptual shape;
3. Brick pattern;
4. Export.

The four tabs are briefly discussed. A more detailed description is found in the user manual (Appendix F).

<u>TAB 1 - SETUP FORCE NETWORK MODEL</u>
Two options are available to setup the network model:
- Create a parameter model;
- A .obj file can be imported and than manually approximated with a parameter model.

<u>TAB 2 - CONTROL</u>
The main functions are to control the appearance of the 3D model and adapt the result-dependant variables:
- At the left side information about how to pan, rotate and zoom the 3D model is displayed;
- Several buttons to show the top-, side- or front view of the 3D model;
- The option to display or hide the loading and adapt the applied loadcase;
- At the right side of the UI the primal grid (top) and in the dual grid (bottom) are displayed.

<u>TAB 3 - MASONRY PATTERN</u>
The dimension of the brick is chosen. Adapting one of the values results in a new pattern of the brick.

<u>TAB 4 - EXPORT</u>
A layer can be exported as several file formats.
There are four layers to export:
1. The force network;
2. Surface created with lines;
3. Surface created with polygons;
4. The masonry pattern.

And there are four file formats to export the data to be used in these software programs:
1. AutoCAD;
2. Maya;
3. Rhinoceros;
4. SketchUP.

5.3 Display the 3D model

The 3D model consists of several layers, which can all be selected and displayed.
These layers are:
- The force network (white lines);
- The surface displayed with curves (green/blue lines);
- The surface displayed as polygons (black planes, white lines);
- The masonry pattern (red cubes).

How to display them, switch between them, export them and adapt them is explained in the user manual (Appendix F).

5.4 Variables to adapt

The application offers several possibilities to control the model and its outcome:

1. Nodal loading for each individual node in the network (§ 5.4.1);
2. Load case (§ 5.4.2);
3. Relocate nodes in the 3D model and primal grid (§ 5.4.3);
4. Adapt the relation between the primal and dual grid: scale factor ζ (§ 5.4.4).

5.4.1 Nodal loading

This value should be determined by calculating the surface area transferred to the node. This is a rough approximation, since it is a complex matter to predict what the area for each node will be. In the current version of the program the value is set to a certain value, without using the area. The loading of each individual node can be altered according to the wishes of the user.

Another relevant aspect regarding nodal loading is the thickness of the shell. When a shape is not buildable with a shell thickness of one layer of bricks, there might be a situation in which it can be build if certain areas have a thickness of two or more layers.

Therefore two aspects – the nodal loading and the element thickness – have to be manually adaptable.

[In the current version of the prototype the initial load on every node is set to a certain value. In a realistic situation this value should be linked to the surface and thickness of the area surrounding the node.

The thickness of the shell surface can only be adapted for the complete shell. It is advised to make this possible only for selected areas of the shell as well.]

5.4.2 Load case

A certain load case, such as snow load, can be applied on the shell surface. In the application this aspect is obtained by adding a load to all nodes at once, instead of adapting the load of one individual node (§ 5.4.1).

The shape remains the same while the loading is changed. The analysis is performed with the new loading. If the network is still in equilibrium, the shape is able to resist the load case.

All load cases can be checked in the same way. If the structure fails in any load case, the shape has to be changed and again checked for all load cases.

5.4.3 Relocate nodes

To adapt the shape of the 3D model, it is possible to change the position of network nodes. Simply click and drag a node in the 3D model and see the shape being changed. The new shape, the primal grid and dual grid are instantly updated.

The analysis and brick pattern will not be updated though, until the node has been released again. (§ 5.6.2)

5.4.4 Adapt scale factor ζ

The function and importance of the scale factor ζ is discussed earlier (Chapter 4). However because this variable is manually adaptable, it is once again mentioned here.

The scale factor determines the relation and value of the forces in the network beams. When the value becomes smaller, the *steepness* of the line-element increases, which results in a higher shell structure and lower horizontal and higher vertical forces (Figure 4.12 c).

Following the same reasoning, a higher scale value results in a lower structure, higher horizontal forces and lower vertical forces.

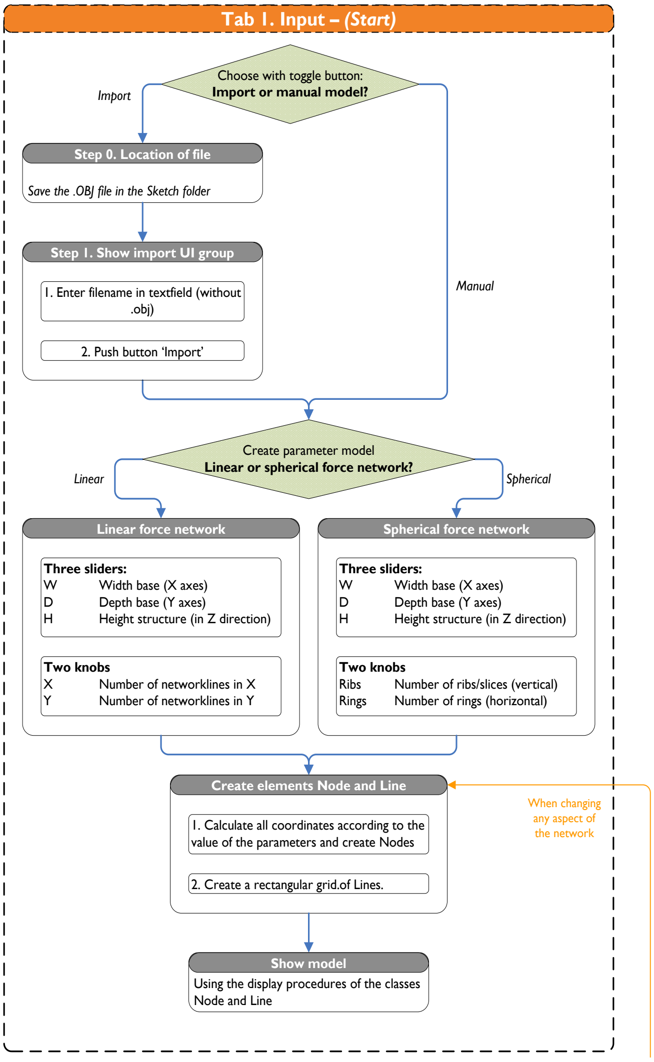


Figure 5.3 *- Flowchart of tab 1: Setup networkmodel*

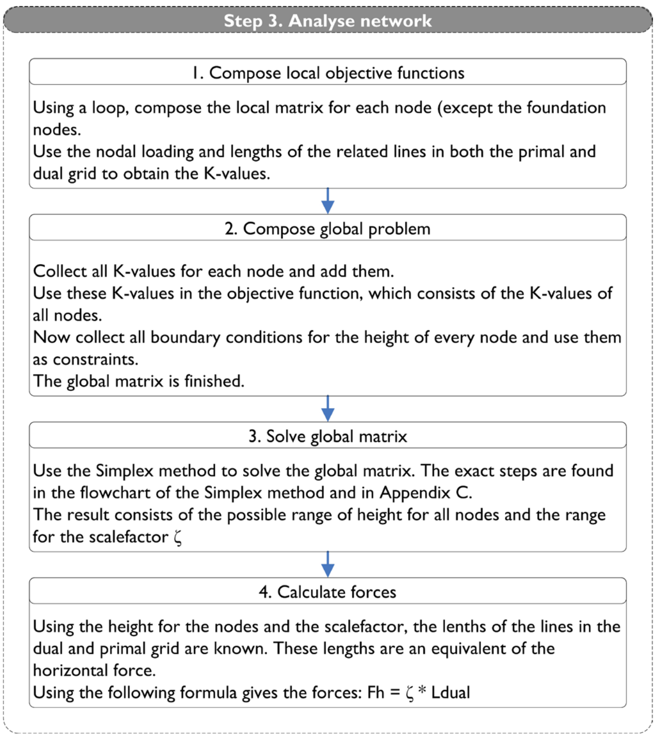


Figure 5.4 *- Flowchart of the Simplex-method procedure*

5.5 Input options

There are two options:

1. Create a parameter model (§ 5.5.1);
2. Import a model (§ 5.5.2).

(Flowchart - Figure 5.3)

5.5.1 Create a parameter model

Five parameters are used, all manually adaptable, to generate a shell shape.

The model can either be based on:

- A rectangular grid;
- A spherical grid.

The common parameters are:

- Shell height;
- Base width;
- Base depth;

For the rectangular grid the two other parameters are:

- Number of rows in X and Y direction to determine the density of the grid and the amount of compression lines.

And for the spherical grid:

- Number of bays and rings density of the grid and the amount of compression lines.

5.5.2 Import a model

The only possible file format to import is .OBJ. This format is not the optimal file format, since it only consists of vectors and it is complex to generate the correct line elements. Moreover the surfaces imported as .OBJ files are polygon surfaces and have no relation with a structural system, force distribution or network.
Therefore it is advised to use the imported model only to approximate it with a parameter model and perform the analysis on this parameter approximation model.
As a result additional steps have to be taken after importing, before an analysis can be done and a pattern generated.

5.6 Procedures

Several procedures are used in the application:

- An analysis method: the Simplex algorithm (§ 5.6.1);
- Method to relocate points of the 3D model (§ 5.6.2);
- Possibility to adapt the scale factor ζ (§ 5.6.3);
- Create a force network to begin with (§ 5.6.4).

One more procedure is the generation of the masonry pattern. Due to its significant contribution and relevance for the research, it is discussed in another paragraph (§ 5.9).

5.6.1 Implementation of the Simplex method

With the Simplex method the linear optimization problem is solved. The results are used to calculate the forces in the line elements of the force network model.
The constants of the optimization problem are a combination of characteristics of the network: the lengths of the line-elements in the primal and dual grids and the loading of each node (§ 4.4 and § 4.5).
(Flowchart: Figure 5.4)

5.6.2 Relocate points

Processing is working in 2D mouse locations, while the model is in 3D. In other words: coordinates can only be expressed in X and Y location of the mouse, while the vectors (nodes) of the 3D model have a X, Y and Z coordinate. This creates a problem: a node can be selected, but after relocating the point the new coordinates can not be determined, since it is based on the X and Y coordinate of the mouse at the moment of releasing the node after relocating it.
To make sure this method is not a complex procedure, the following is assumed, regarding moving in 3D environment:
Two procedures replace the 3D relocating. To be able to use these procedures, a node has to be selected, by clicking on it in the primal grid. The two procedures, which only work when a node is selected in the primal grid, are:

- Adapt the X and Y coordinate by dragging a node in the primal grid;
- Adapt the Z coordinate, using the UP and DOWN-keys.

Class: Node
(Description) A point, defined by three coordinates: X, Y and Z. *(Characteristics)* - X, Y and Z coordinates - Node-number i (in primal grid and polygon of dual grid) *(Procedures)* - Contains To check if a node is selected, when the mouse is clicked - AdaptLoading To adapt the loading when the + or – key is pressed while a node is selected - Display - DisplayPG The same as 'Display', but without using the Z coordinate - DisplaySelected The same as 'Display', but a different color is used, to highlight the node

Figure 5.5 *- Characteristics of Node-class*

Class: Line
(Description) Connection between two nodes *(Characteristics)* - Coordinates of the two nodes: X1, Y1, Z1, X2, Y2, Z2 - Two numbers i and j which are the node ID numbers *(Procedures)* - getLength3D To obtain the length of the line to create the brick pattern - getLengthPG To obtain the length of the line to form the polygon in the dual grid - getAngle3DX To obtain the angle of the line with the X-axis to rotate the bricks with - getAngle3DY To obtain the angle of the line with the Y-axis to rotate the bricks with - getAnglePG To obtain angle in the primal grid to create the dual grid - obtain To check if a line is clicked when the mousebutton is clicked - update To update all coordinates after moving a node - display - displayPG

Figure 5.6 *- Characteristics of Line-class*

5.6.3 Change scale factor ζ

This is a factor that influences the outcome of the Thrust Network Analysis (TNA) and therefore is relevant for the application.

The relation between the primal and dual grid is for one determined by the scale factor ζ. If the primal grid consists of polygons, formed by more than three sides, the dual grid has not got an unique solution. In this case, the scale factor ζ determines the scale of the dual grid, and as a result also determines the outcome of the force distribution. Therefore the scale factor has to be adjustable. The selected factor is directly linked to the scale of the dual grid shown in the user interface.

5.6.4 Starting force network

Block experienced the problem of finding a suitable force network for a random shape and therefore this was one of the recommendations for further research of the Thrust Network Analysis article [10]. Since in this research this aspect has not been solved, the need for an algorithm to find a starting 3D force network for an imported model remains a recommendation for further research.

5.7 Main elements

The input elements for the procedures (§ 5.6) are specific elements. These elements are the *stones* with which the application, and the corresponding models and diagrams, are build and analyzed.

The basic elements are:

- Nodes (§ 5.7.1);
- Lines (§ 5.7.2);
- Polygons (§ 5.7.3);
- Primal grid Γ (§ 5.7.4);
- Dual grid Γ* (§ 5.7.5);
- Bricks (§ 5.7.6).

5.7.1 Nodes

Characteristics

When a node is selected, it is highlighted in red, including the loading.

(Flowchart: Figure 5.5)

Constructor

Create a 3D point by determining the X,Y and Z coordinate. Add a number to it, for identifaction in the primal and dual grids.

Methods

- Select: a point in the 3D force network model or primal grid;
- Move: a point in the 3D force network model after selecting it;
- Adapt the nodal loading.

5.7.2 Lines

Characteristics

A line is given a red color when it is overloaded or is in tension. When one of the two nodes it is connected to is selected, it is highlighted in orange.

(Flowchart: Figure 5.6)

[This aspect is not working in the current version of the prototype, because the analysis is not performed completely yet, due to the problems with scaling the dual grid correct.]

Constructor

Class: Polygon

(Description)
Several nodes connected by lines to form a closed polygon

(Characteristics)
- Coordinates of a certain amount of nodes: X1, Y1… Xn-1, Yn-1, Xn, Yn
- Identification number i, which is the number of the node belonging to the polygon
- ID's of the lines forming the polygon: ID1 … IDn-1, IDn

(Procedures)
- update

To update the coordinates and position of the polygon

- display

Figure 5.7 *- Characteristics of Polygon class*

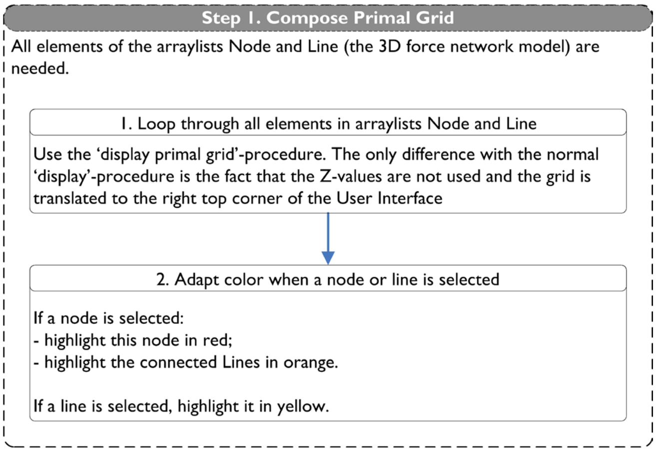


Figure 5.8 *- Flowchart of Primal grid*

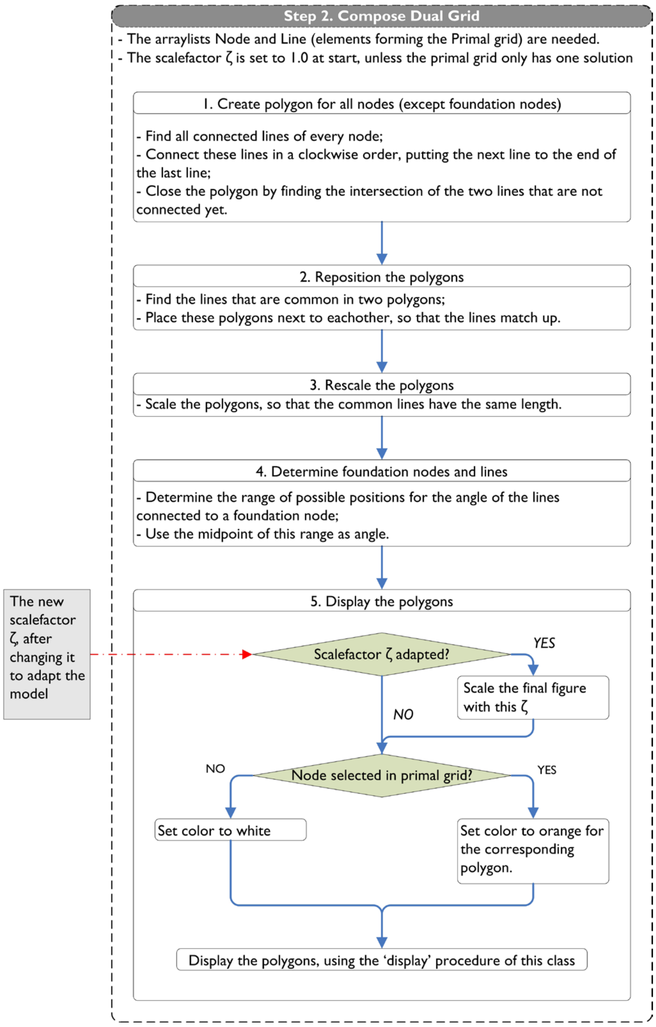


Figure 5.9 *- Flowchart of Dual grid*

Class: Brick

(Description)
3D cube

(Characteristics)
- Width W
- Height H
- Depth D
- Color colorBrick
- Coordinates of base X, Y and Z

(Procedures)
- update

- display

Figure 5.10 *- Characteristics of the Brick class*

Create a line between two Nodes. Add the numbers of these nodes as identification of the line during analysis.

Methods

- Select: a line in the 3D or primal grid and show the corresponding information.

5.7.3 Polygons

Characteristics

A polygon is highlighted in orange if the corresponding node is selected in the primal grid.
(Flowchart: Figure 5.7)

Constructor

Create a polygon in the dual grid, using the X and Y coordinates of all nodes belonging to the lines included in the polygon.
Add a number to it to correspond to the node number in the primal grid.

Methods

- Move: let the polygon be moved to the right position in the dual grid, using the procedure to create the dual grid;
- Scale: the polygon so that the dual grid is correct.

5.7.4 The Primal grid Γ

As mentioned before, the primal grid is the planar projection of the force network of the 3D model: it consists of the horizontal reflections of the force network line elements on the horizontal ground plan (X-Y plane). It can be compared with the shadows of a 3D frame on the ground, when it is being lighted from above by a diffuse light-source.
(Flowchart: Figure 5.8)

Steps

- The direction and length of the line elements of the force-network are used to determine the horizontal length of them in the primal grid;
- The primal grid is shown in the right top corner of the user interface.

Extra features

It is possible to select a node in the primal grid, if it for instance needs to be repositioned. The selected node is highlighted, including the lines connected to it. This is also applied to the 3D model.
When a beam is overloaded, the corresponding line gets a red color.

5.7.5 The Dual grid Γ^*

Starting point is the primal grid. The dual grid is the reciprocal figure of the primal grid. A *reciprocal figure* is defined as: *'Two plane figures are reciprocal when they consist of an equal number of lines, so that corresponding lines in the two figures are parallel and corresponding lines which converge to a point in one figure form a closed polygon in the other.'* [10].
More information is presented earlier in the report (§ 4.4).
(Flowchart: Figure 5.9)

Steps

- All lines related to a node in the primal grid are collected;
- Next step is to transform these lines into polygons. These polygons have to be closed to assure equilibrium;
- If the primal grid consists of polygons existing of more than three lines, the scale factor ζ is variable. It will be set to 1.0 and can be changed manually afterwards;
- The dual grid is shown in the right lower corner of the user interface.

[This aspect is working in the current version of the prototype when a polygon is made of 4 lines. When it has only 3 lines the script to form polygons and the dual grid is not working.]

Extra features

When a node is selected in the primal grid, the corresponding polygon in the dual grid is highlighted in orange. This enhances the ability to see what the effect is of any changes made to a node.

5.7.6 Brick

An assumption is that only standard brick shapes and sizes

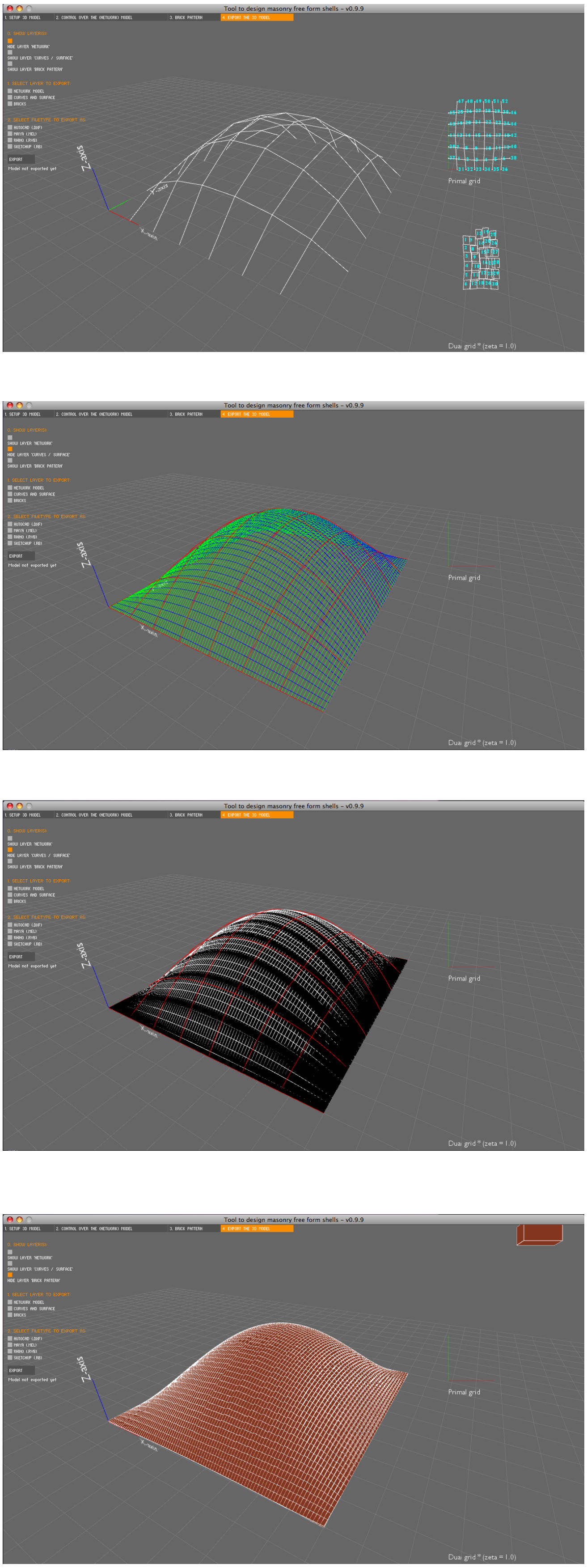


Figure 5.11 - The four layers created by the application. From top to bottom: the force network, the lines-surface, the polygon-surface and the brick pattern.

are used, which can be easily made by brick producers. Moreover only one brick shape is used to generate the masonry pattern.
(Flowchart: Figure 5.10)

Extra features

The brick pattern can be displayed or hidden in the 3D model, by toggling a button in Tab 3 - Brick pattern.
In this research only one color is used for the brick. A good addition would be the possibility to use more colors, so that patterns can be created using certain color schemes.
This has therefore been added as a recommendation for further research to extend the ability to adjust the appearance of the brick facade (Chapter 8).

5.8 Export options

5.8.1 Stresses and forces in structure

There is an option is to generate a text-file, in which the information of all lines and nodes is given.
This information consists of:

- Node number;
- Node coordinates;
- Nodal loading;
- Line lengths;
- Line stress.

The loadings in the lines connected to foundation nodes are separated from the other lines.
When designing the structure, the foundation is an important aspect, especially for shell structures. To design the foundation, the relevant forces must be known. The force has a value and a direction. The force should be divided into its horizontal and vertical component.

5.8.2 Export final model

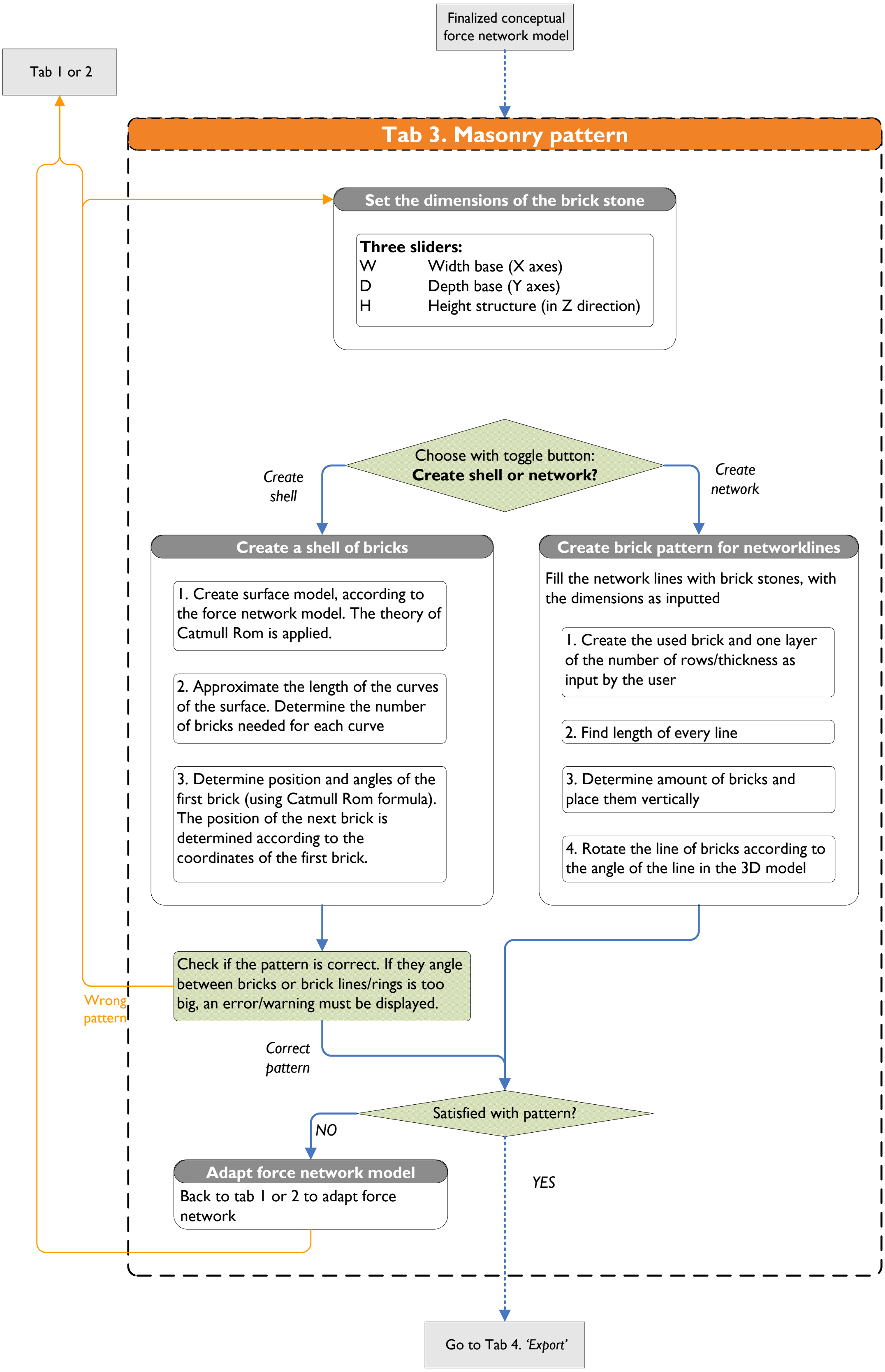
Several layers are created when the application is used, so that the results are available for all actors that are active in the continuation of the design process. For the structural engineer the force network model layer is of importance, for the architect the surface models and the contractor might be interested in the masonry pattern layer (Figure 5.11).

- The force network mode (for the structural engineer)
- The surface model, consisting of curves (for the architect)
- The surface model, consisting of polygons (for the architect);
- The brick pattern (for the brick producer and contractor).

The 3D model can be exported as four file formats:

- AutoCAD .DXF file
- Autodesk Maya .MEL file
- Rhinoceros .RVB file
- SketchUP .RB file

Figure 5.12 *- Flowchart of the Masonry pattern procedure*



All file formats make use of points, coordinates, lines or a combination of them. This explains why in all programs, except AutoCAD, the *script editor* of the program is needed to open the exported file.

[It is possible to export in the current version of the prototype, but the model is not positioned in the origin of the axis and it is rotated.]

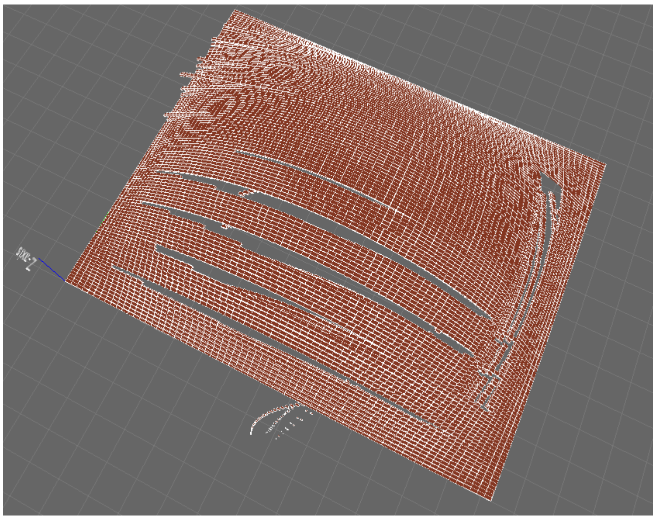


***Figure 5.13** - Failing of the masonry pattern script procedure*

5.9 Masonry pattern

To visualize the masonry shell a pattern is created. (Flow chart - Figure 5.12). The technique to generate the patterns is explained earlier (§ 4.7).

To generate a *perfect* pattern, closing the exact boundaries of the base plane, non-standard brick shapes are needed.

However an assumption is to only use one size of a standard brick and so this is not being researched further.

Non-standard forms will be an improvement for the application and has therefore been added to the recommendations (Chapter 8).

5.9.1 Range of dimensions for brick

The range of allowed dimensions of the standard brick is determined by the factory and the machinery where the bricks are produced.

After contact with brick producers [17], the following ranges have been set:

- Length 160 - 280 mm;
- Depth 75 - 120 mm;
- Height 50 - 90 mm.

The exact dimensions of the brick are determined with the sliders in tab 3 - Masonry pattern.

[The masonry pattern procedure is partly working in the current version of the prototype. At some positions of the surface the pattern-script is failing (Figure 5.13). This is caused by the decision to equalize the length of the lines in the 3d force network. As a result the related primal grid has inclined lines, resulting in curved lines from left to right and top to bottom. The script where the related nodes for the Catmull Rom theory are obtained fails in certain positions due to this curvature.]

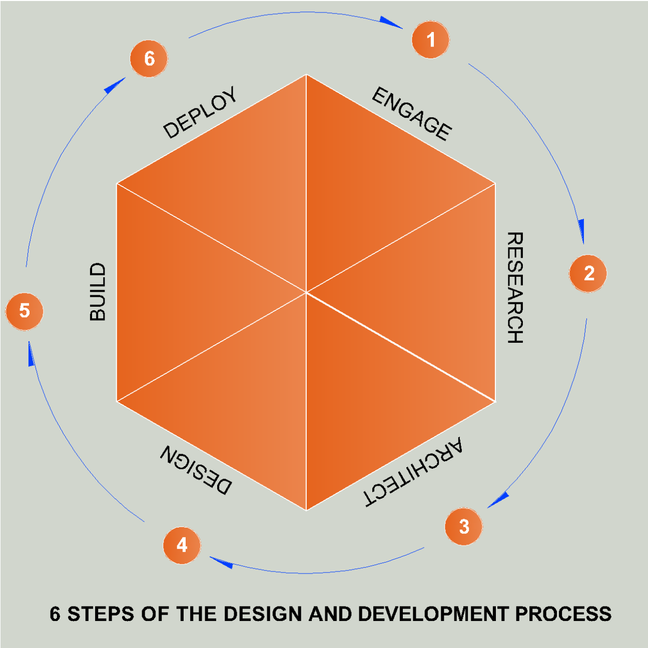


Figure 5.14 *- Design process, build up in 6 stages*

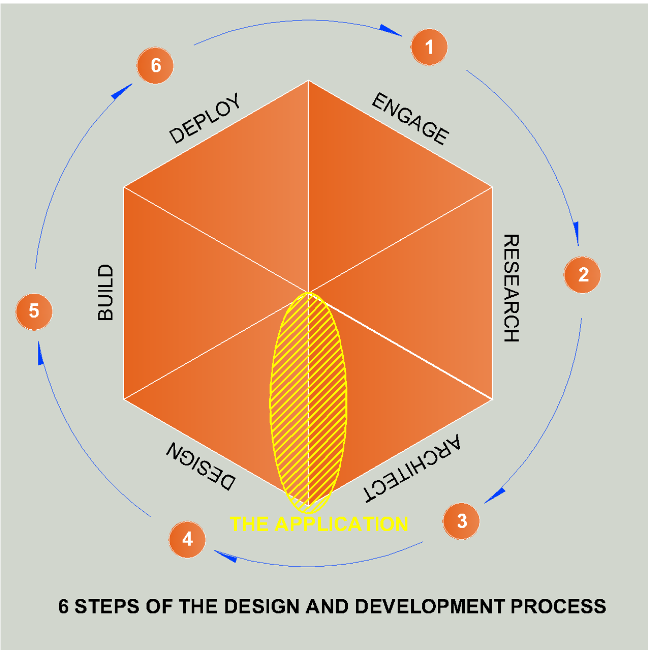


Figure 5.15 *- The yellow highlighted area is the position of the application in the design process.*

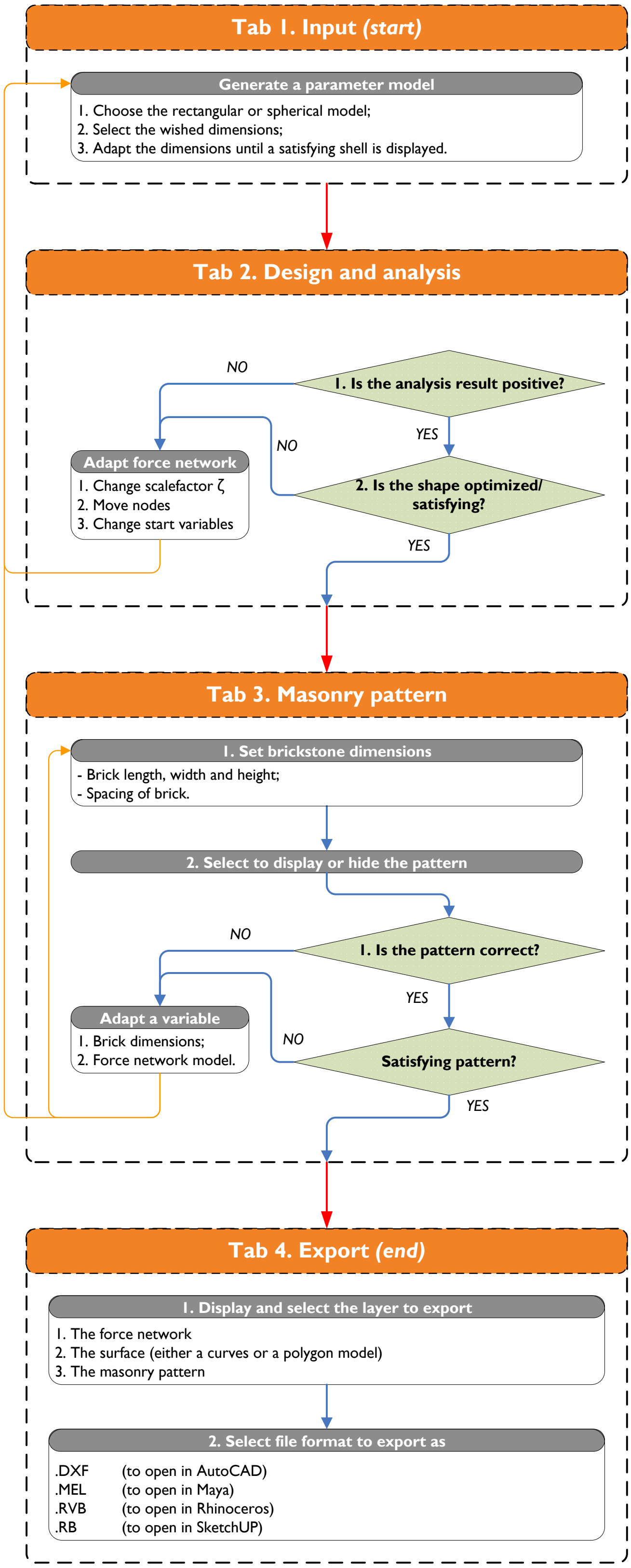


Figure 5.16 *- Workflow case 1 - the wish is a masonry shell*

5.10 Workflow of the application

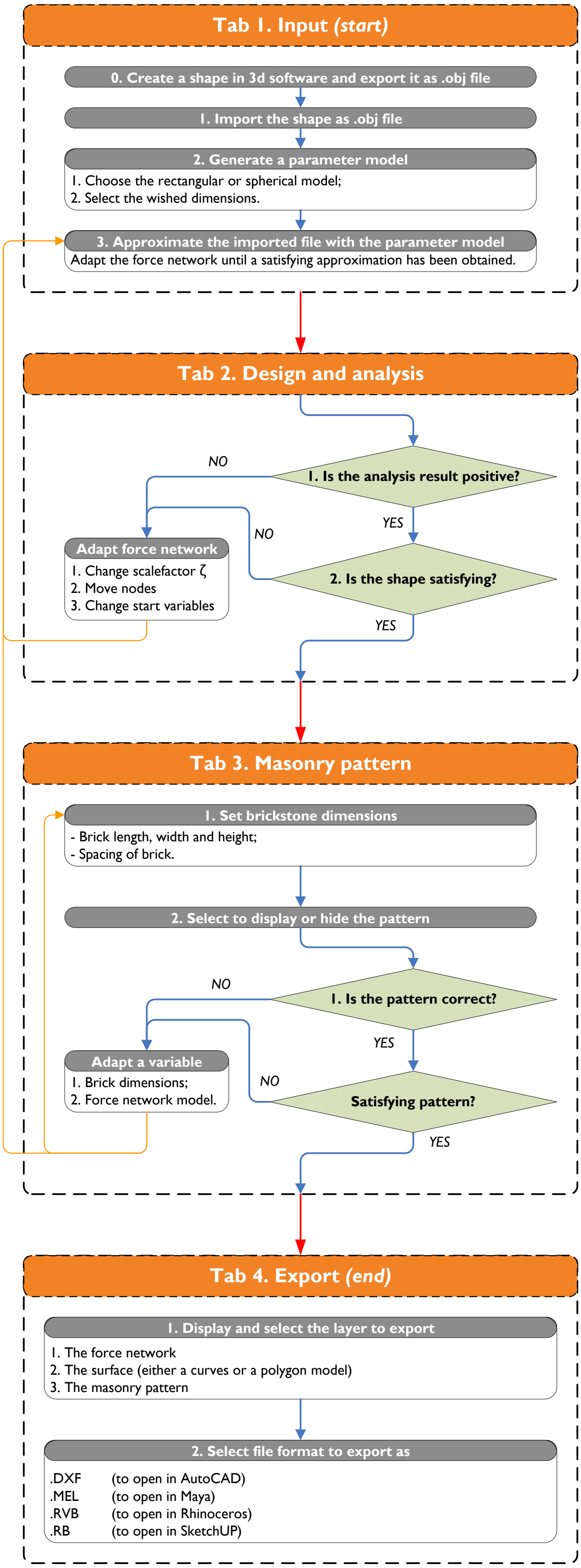


Figure 5.17 *- Workflow case 2 - Shape is known, check if masonry is possible*

<u>POSITION IN THE DESIGN PROCESS</u>

The application is used during the conceptual design stage of the design process. The whole process consists of several stages and the period of time it takes can be from a year up to several years. (Figure 5.14)

The conceptual phase of the design is located in the beginning of the *architect* and *design* stage.

After the initiation of the design (*Engage*) a list of demands and wishes is made (*Research*). The architect makes some sketches according to these wishes (*Architect*) and he is now interested in an engineering design (*Design*). At this moment it is time to use the application. (Figure 5.15, yellow section)

Together with the architect a meeting is attended, during which the conceptual design is completed.

<u>WORKFLOW OF THE APPLICATION</u>

The application is used in two situations:

1. The architect knows he is going to use a masonry shell and knows the rough dimensions;
2. The architect wants to create a shell structure and wonders if it is possible in masonry.

The difference between these situations is mainly the starting model. The other steps of the work flows are the same for both cases.

Case 1 (Figure 5.16)

The workflow starts in Tab 1 with a parameter model. The starting model is a parameter model, of which the parameters are adjustable according to the wishes of the architect.

Case 2 (Figure 5.17)

The workflow starts in Tab 1 with an import model (.obj format). This model is manually approximated with a parameter model.

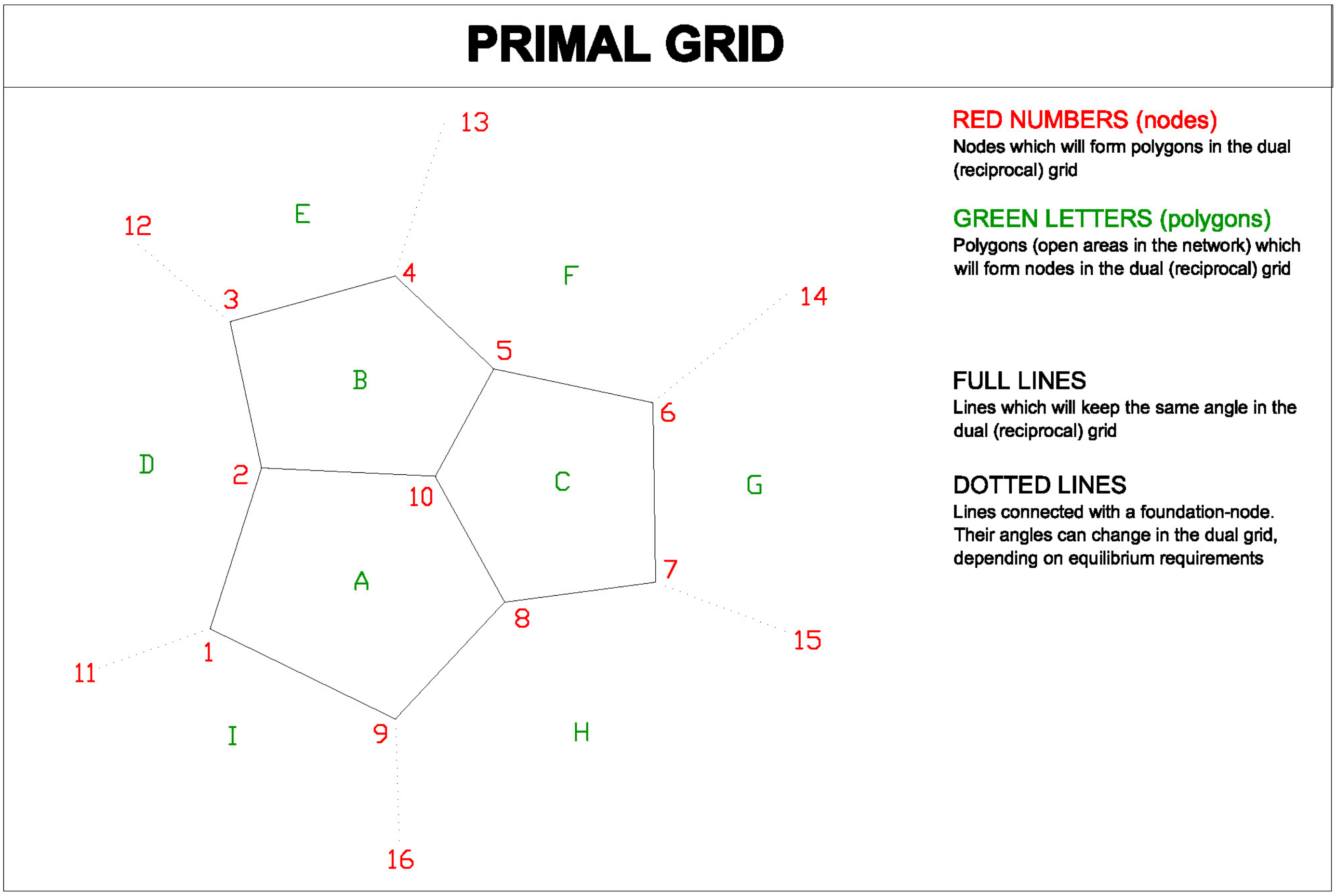


Figure 5.18 *- Imaginary primal grid, used to support the explanation of several aspects of the script.*

5.11 Current status of the script

Important topics for further research and development:

1. Generation of the polygons;
2. Generation of the dual grid;
3. Generation of the masonry pattern;
4. Performing the analysis;
5. Exporting the model;
6. Nodal loading;
7. Angle check of the brick stones;
8. Gaps in the masonry pattern.

These topics are explained in further detail. The explanation consists of a short description of the procedure as it has been designed and is intended to work, followed by the problems with this script and finally the proposed solution is given.
In most situations the explanation is supported by an example. These examples are all based on an imaginary example primal grid (Figure 5.18).

1. GENERATION OF THE POLYGONS

Procedure to generate the polygons:
All the nodes are regarded using a loop. At this point of the script, all nodes are still independent of each other and so for each node the same procedure is performed.

After a node has been selected in the loop, the following steps are taken:

1. Find the lines related to the node, by checking whether the node number is one of the two node numbers of a line. This is done for all lines, by looping through the array of all lines.
2. List the related lines according to their angle;
3. Start with the first line of the list as obtained in step 2 and connect the second line of the list to the end of the first line (Figure 5.19, step a-b);
4. Continue as step 3, until all lines have been connected (Figure 5.19, step c).

It is expected that the endpoint of the last line and the starting point of the first line do not share the same coordinates. The result is either an open polygon, or one where lines intersect. This is not acceptable for application in the TNA. Therefore the point of intersection of the first and last line has to be determined.
This is done as last step of the procedure to generate the polygon:

5. Find the coordinates of the point of intersection of the first and last line, to ensure a closed polygon (Figure 5.19, step d).

Description of the problem:
The script is not performing well when a combination of polygons of 3 and 4 sides is used. In the current version of the program it is advised to only use grids consisting of polygons of 4 sides. The script to form the polygons performs well and therefore the reason why the script does not work for polygons for a grid of a combination of polygons with less or more than 4 sides

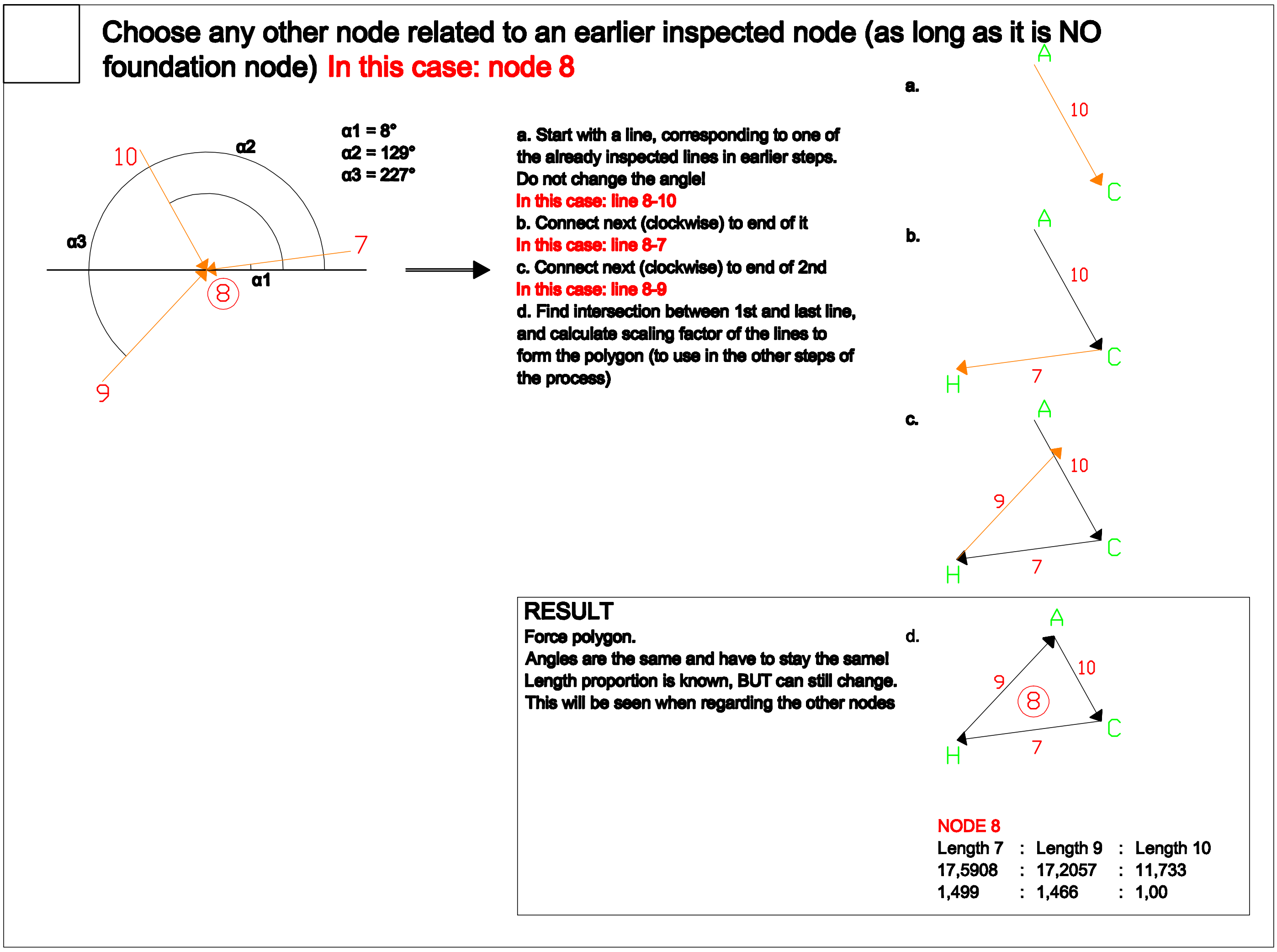


Figure 5.19 - Procedure to construct a polygon (in this case for node 8 of figure 5.18)

has not been found
It is thought that it should be found at the point in the script where the dual grid is formed and the related polygons are scaled (see *2. Generation of the dual grid*).

Solution to this problem:
Review the script by an expert in scripting.

2. GENERATION OF THE DUAL GRID

The dual grid is formed using several loops and within each of them a procedure is followed (see below).

Procedure within a loop (applied to all polygons):

1. A polygon is selected *(example: polygon 8, belonging to node 8)* (Figure 5.18)
2. The polygons connected to the selected polygon are searched for *(example: polygons 7, 9 and 10)* (Figure 5.20)
3. The cmmon line of each related polygon is scaled to the length of the corresponding line of the selected polygon. The length of the lines is calculated according to the Euclidean length, which is based on the theorem of Pythagoras. The scalefactor is the product of these two lengths *(example: polygons 8 and 10 are regarded. The scalefactor is L1/L2)* (Figure 5.21)
4. The polygon related to the common (scaled) line is scaled by the corresponding scalefactor (Figure 5.22)
5. The related polygon is moved to the right position of the selected polygon (Figure 5.23)

Description of the problem:
The final dual grid consists of polygons with several scalefactors. This is caused by the fact that a polygon of four and more sides is scaled either in one (Figure 5.24a) or in all directions (Figure 5.24b). The resulting new shape of the polygon is different (Figure 5.24c).

The way of scaling, in combination with the choice to loop through the polygons using their ID-number, causes the dual grid to have different scalefactors (Figure 5.25).

Solution to this problem:
The solution is to use the method as described by P. Block [16], using the solving of two matrix problems (§4.2.3 of [16]).
The script of this solution method is an aspect that needs to be researched further.

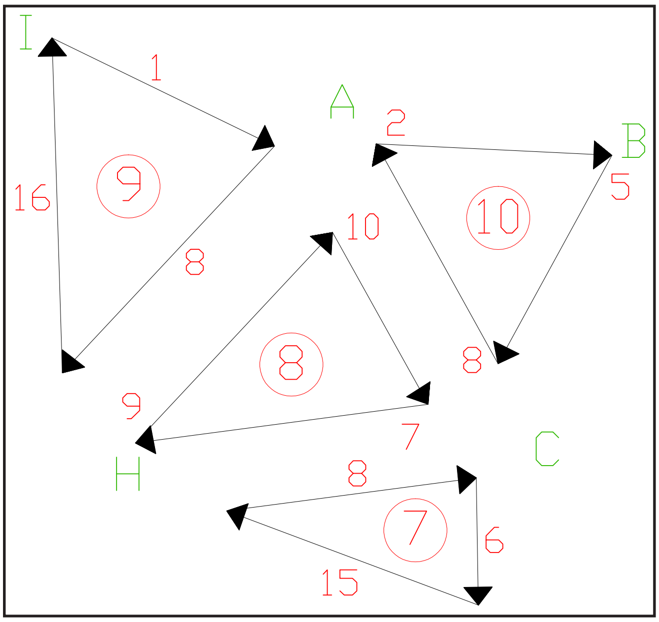


Figure 5.20 - *Step 2 of the loop to form the dual grid: the related polygons of the selected polygon are searched for.*

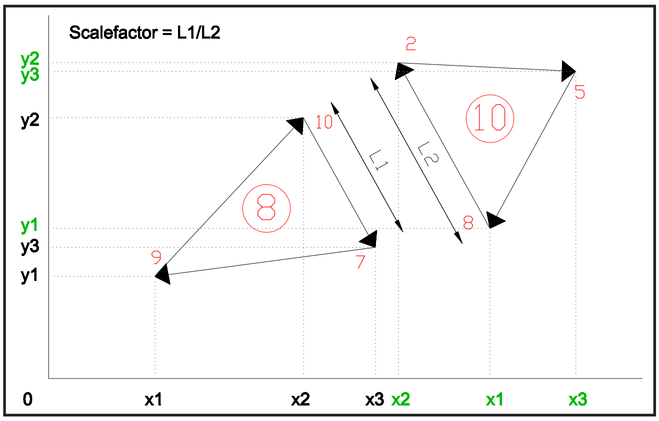


Figure 5.21 - *Step 3 of the loop to form the dual grid: the scalefactor of the related polygons is determined*

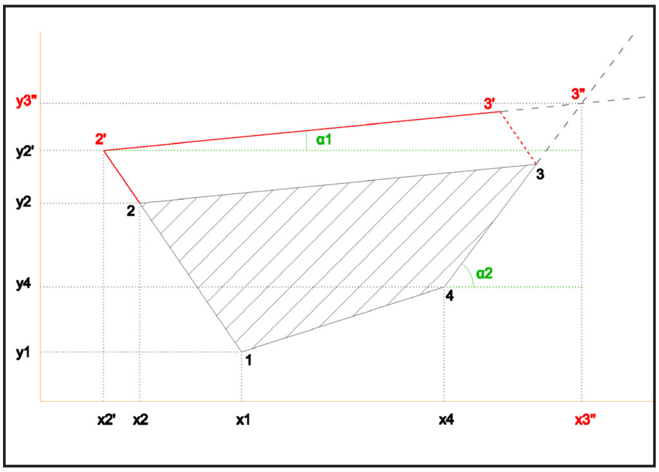


Figure 5.24a - *Scaling of a polygon in 1 direction*

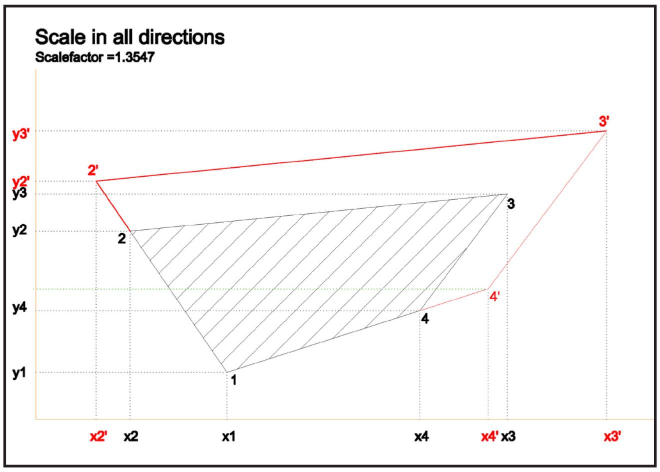


Figure 5.24b - *Scaling of a polygon in all directions*

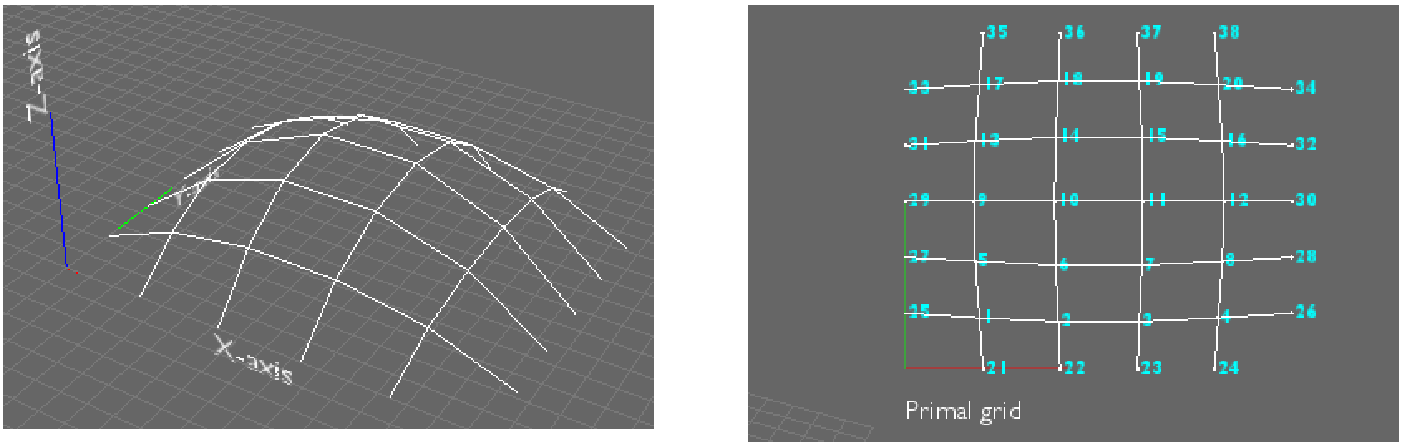


Figure 5.25 - *Screenshot. From left to right: 3D, primal grid and dual grid. In the last figure is displayed that there are different scale factors.*

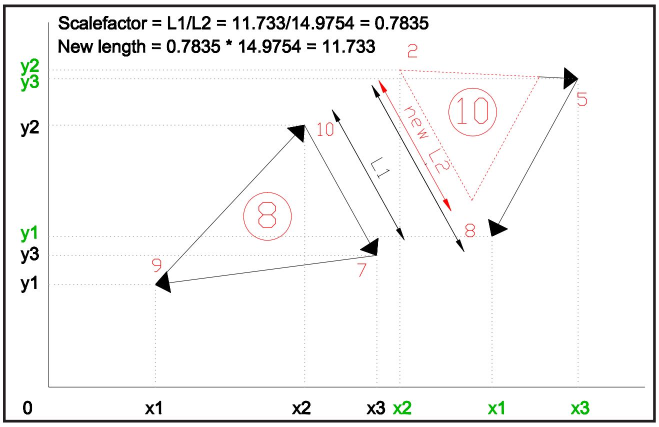


Figure 5.22 - *Step 4 of the loop to form the dual grid: the scalefactor is applied to the whole polygon (in this case for polygon10)*

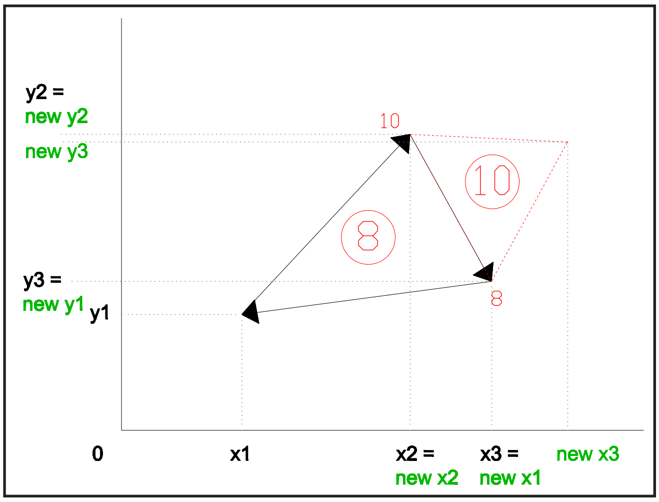


Figure 5.23 - *Step 5 of the loop to form the dual grid: the polygon is translated to the right position (in this case polygon 10 is translated)*

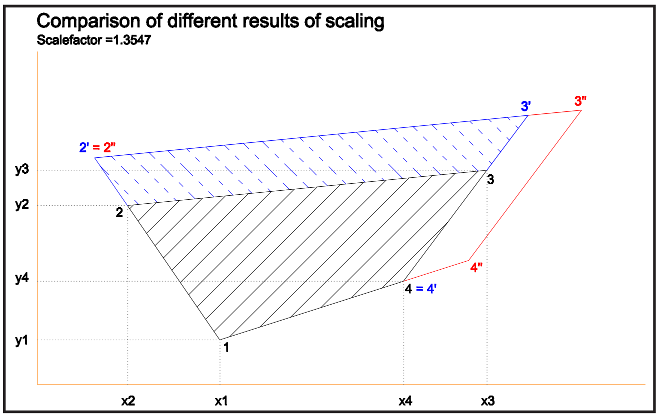
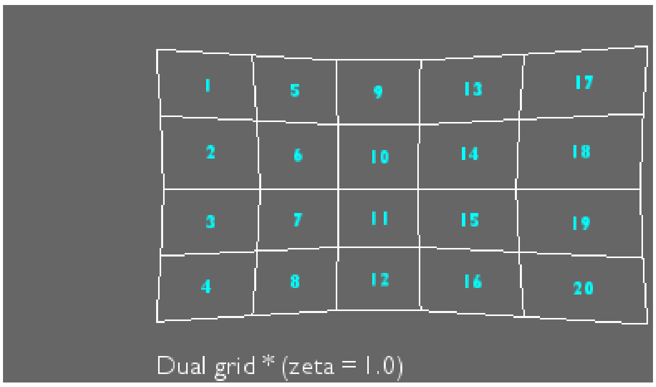
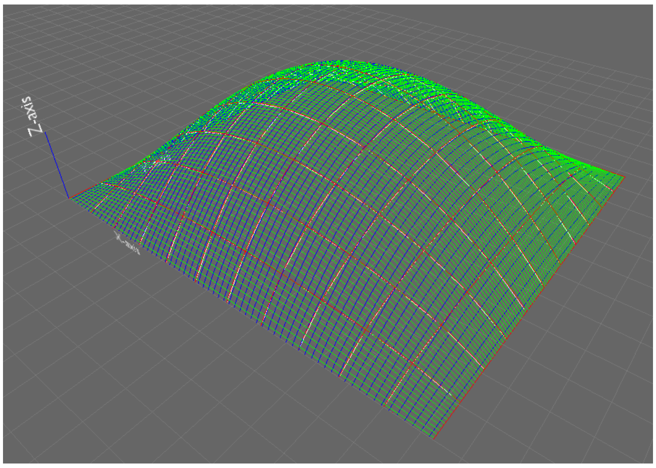


Figure 5.24c - *Comparison of scaling in 1 and all directions*





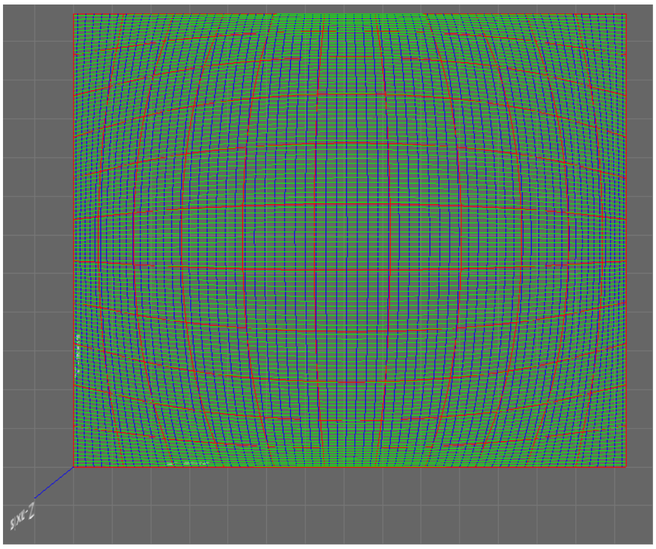
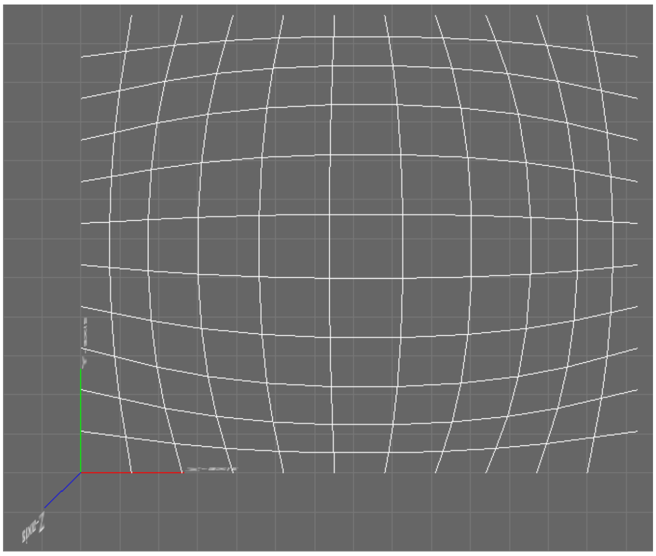


Figure 5.26 - *The lines on the surface of the shell, which are used as the centerlines for the bricks to be placed on. Above the 3D view is shown, at the left the topview.*



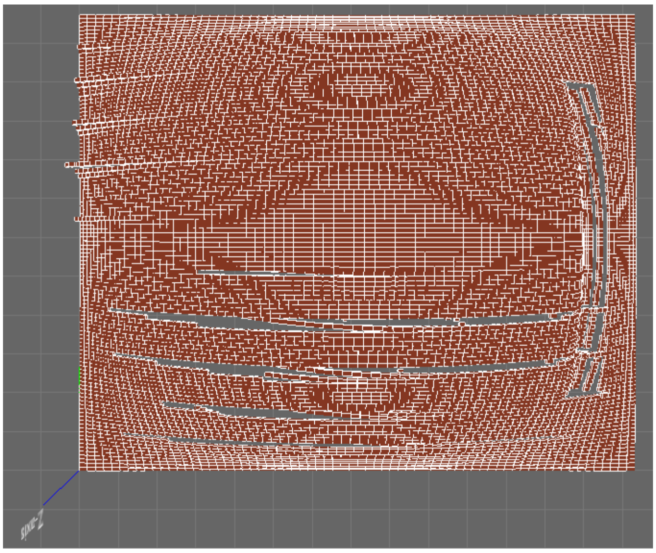
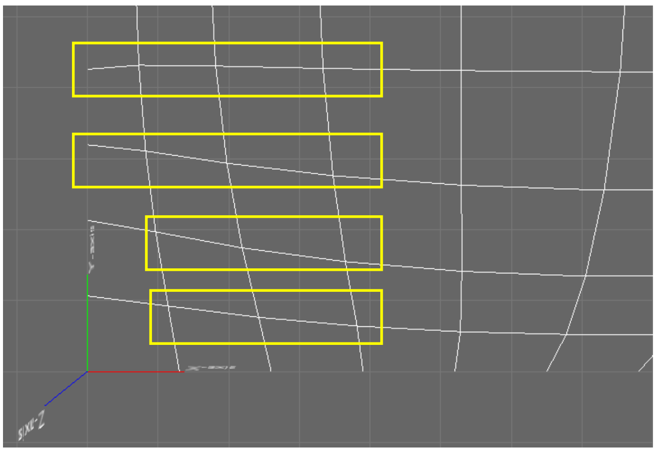


Figure 5.27 - *Failure of the masonry pattern script at several locations in the pattern.*



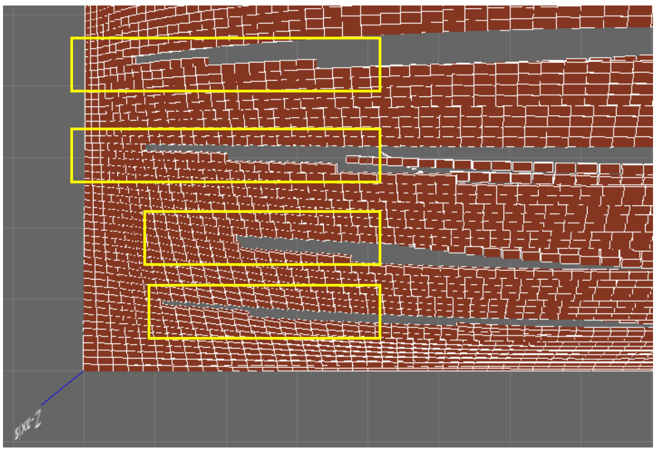


Figure 5.28 - *In yellow the beginning of the failure of the masonry pattern is shown.*

3. GENERATION OF THE MASONRY PATTERN

The masonry pattern procedure is partly working in the current version of the prototype.

Procedure to create the masonry pattern:

1. Draw 'lines' every brickwidth (Figure 5.26)
2. Approximate the length of these lines, using the coordinates of the force network and assuming straight lines between these known points;
3. Determine the amount of bricks per line by dividing the length with the brick length.

Procedure to find coordinates and angles of one brick:
(this procedure is run for every 'line' of bricks)

1. Determine the coordinates of the center point of the brick using the coordinates of the brick before it;
2. Determine the angles of the brick using the theory of Catmull Rom splines, by using the coordinates in the derivative of the Catmull Rom formula;
3. Store the values, so that they can be used in the continuation of the script.

Description of the problem:

At some positions of the surface the pattern-script is failing. (Figure 5.27).

This is caused by the shape of the primal grid. This shape is a result of the decision to equalize the length of the lines in the 3D force network. As a result the related primal grid has 'angled' lines. The pattern-script fails where the related nodes for the Catmull Rom theory are obtained. (Figure 5.28). In the yellow blocks the starting point of the failure is shown. The remaining of the bricks of that masonry 'line' is a continuation of this failure at the beginning.

Moreover the aspect of assuming a certain value for a point at the border of the surface is questionable. The best situation would be when these border points are related to the angle of the surrounding surface around these points. This has not been scripted and so is not part of the current version of the program.

Solution to this problem:

Review the script by an expert in scripting and add the missing aspects as mentioned above.

4. PERFORMING THE ANALYSIS

A matrix is generated using the theory explained in Appendix C. First the matrix values are determined (Figure 5.30). Another matrix is created with the boundaries for the height of all nodes (Figure 5.31). After the analysis has been completed, the values for the inverse of the scalefactor are put in a matrix as well (Figure 5.32).

Description of the problem:

The problem should be solved using a matrix optimization [16]. The consensus was that this could be achieved by an iteration as well, but it was proven this was not a right assumption.

In combination with the problems to obtain the lengths of the branches in the dual grid, this results in an incomplete analysis.

Solution to this problem:

The solution is to use the method as described by P. Block [16].

The script of this solution method is an aspect that needs to be researched further.

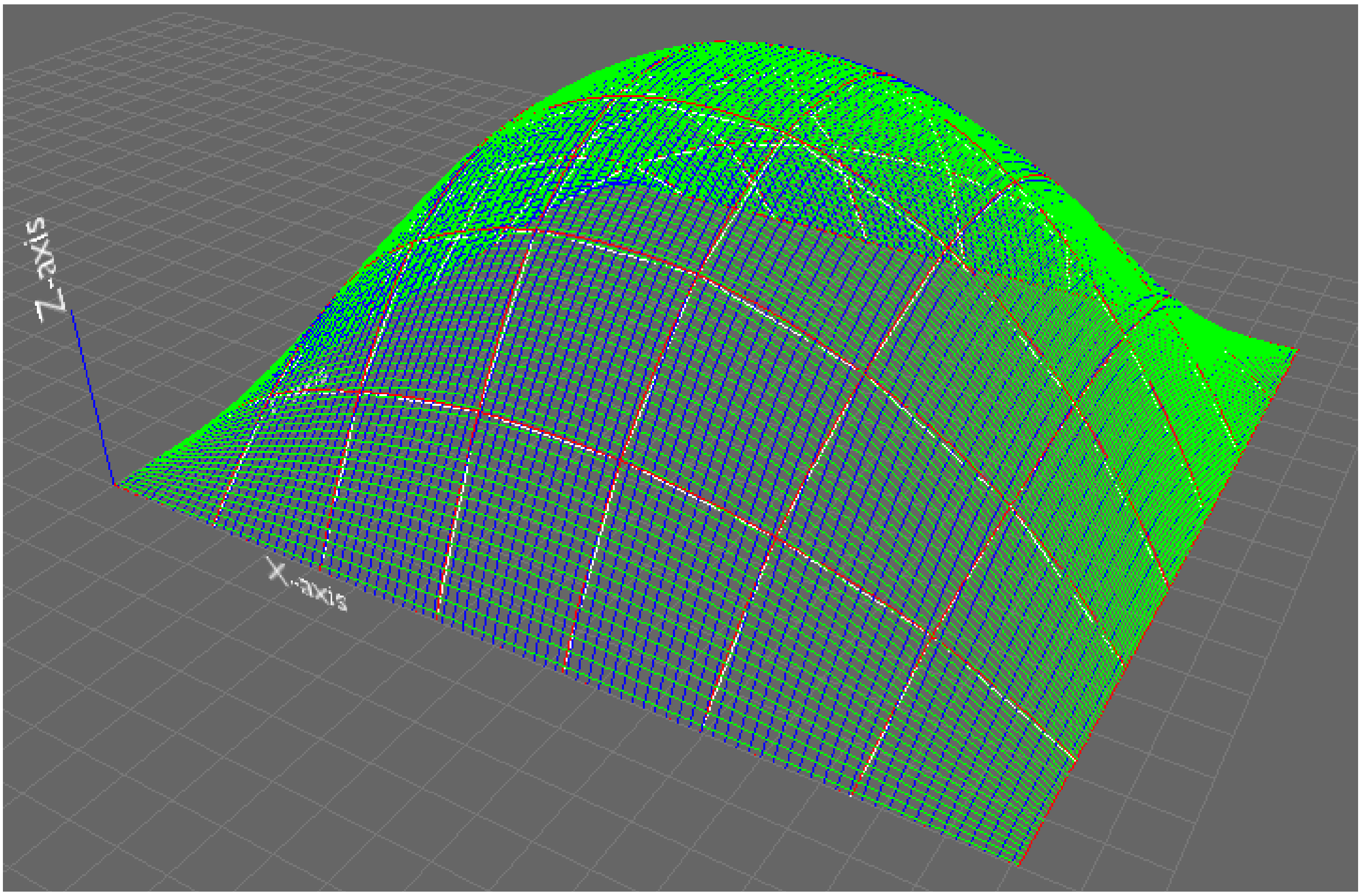


Figure 5.29 - *Force network model of which the text files are shown in figures 5.30 - 5.32. From left to right: 3D view (including brick center lines), primal grid and dual grid.*

Matrix.txt - Notepad

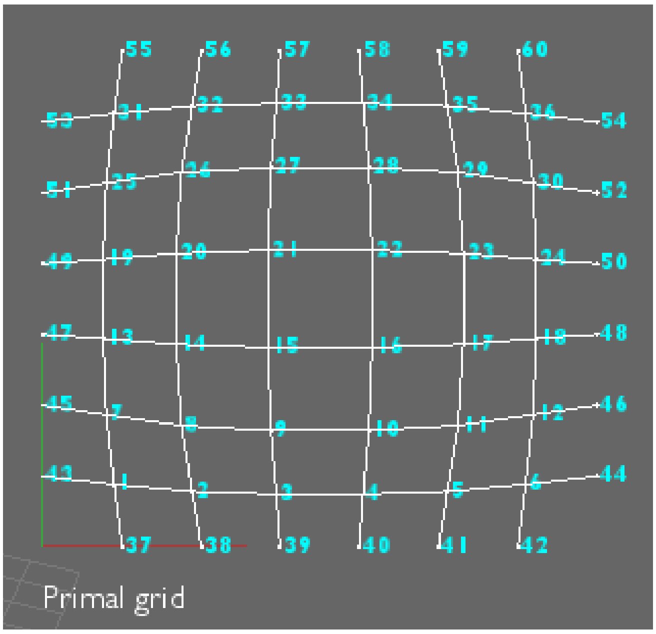
File Edit Format View Help

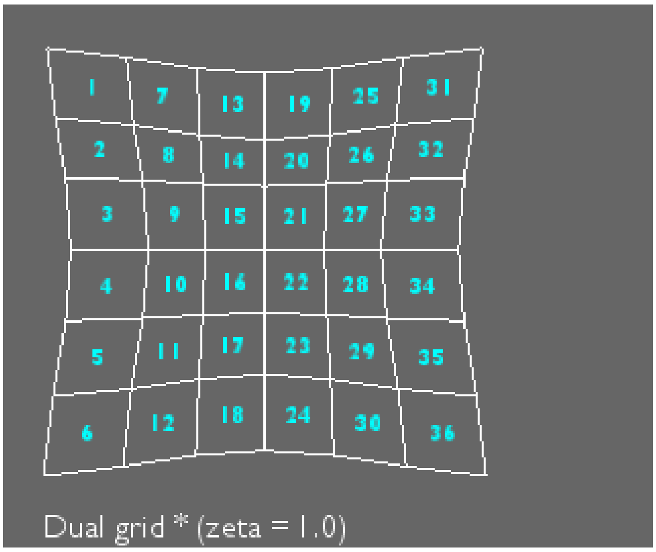
MATRIX OF K-VALUES

Number of rows = 6
Number of columns = 6
Width base = 18.1 [m]
Depth base = 16.2 [m]
Height model = 7.431143 [m]

Columns / Rows	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	1,992	-0,473	0	0	0	0	-0,460	0	0	0	0	0	0	0	0	0	0	0	0
2	-0,413	1,581	-0,379	0	0	0	0	-0,343	0	0	0	0	0	0	0	0	0	0	0
3	0	-0,474	2,207	-0,465	0	0	0	0	-0,560	0	0	0	0	0	0	0	0	0	0
4	0	0	-0,409	1,928	-0,416	0	0	0	0	-0,487	0	0	0	0	0	0	0	0	0
5	0	0	0	-0,449	2,121	-0,488	0	0	0	0	-0,520	0	0	0	0	0	0	0	0
6	0	0	0	0	-0,428	1,936	0	0	0	0	0	-0,479	0	0	0	0	0	0	0
7	-0,483	0	0	0	0	0	1,909	-0,439	0	0	0	0	-0,432	0	0	0	0	0	0
8	0	-0,348	0	0	0	0	-0,370	1,252	-0,283	0	0	0	0	-0,251	0	0	0	0	0
9	0	0	-0,541	0	0	0	0	-0,338	1,569	-0,291	0	0	0	0	-0,400	0	0	0	0
10	0	0	0	-0,434	0	0	0	0	-0,236	1,264	-0,274	0	0	0	0	-0,320	0	0	0
11	0	0	0	0	-0,385	0	0	0	0	-0,245	1,242	-0,328	0	0	0	0	-0,285	0	0
12	0	0	0	0	0	-0,627	0	0	0	0	-0,509	2,347	0	0	0	0	0	-0,562	0
13	0	0	0	0	0	0	-0,371	0	0	0	0	0	1,586	-0,379	0	0	0	0	-0,360
14	0	0	0	0	0	0	0	-0,221	0	0	0	0	-0,328	0,985	-0,250	0	0	0	0
15	0	0	0	0	0	0	0	0	-0,297	0	0	0	0	-0,251	1,004	-0,211	0	0	0
16	0	0	0	0	0	0	0	0	0	-0,384	0	0	0	0	-0,276	1,304	-0,328	0	0
17	0	0	0	0	0	0	0	0	0	0	-0,344	0	0	0	0	-0,296	1,327	-0,395	0
18	0	0	0	0	0	0	0	0	0	0	0	-0,415	0	0	0	0	-0,374	1,663	0
19	0	0	0	0	0	0	0	0	0	0	0	0	-0,330	0	0	0	0	0	1,455
20	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,295	0	0	0	0	-0,522
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,370	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,348	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,279	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,567	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0,333
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 5.30 - *Part of the text file of the matrix that needs to be solved*





Heights.txt - Notepad

NODE HEIGHTS AND BOUNDARIES

Node number	Lower boundary [m]	Height z [m]	Upper boundary [m]
1	1,608	1,617	1,625
2	2,626	2,634	2,643
3	3,145	3,154	3,162
4	3,145	3,154	3,162
5	2,626	2,634	2,643
6	1,608	1,617	1,625
7	2,647	2,655	2,664
8	4,511	4,519	4,527
9	5,548	5,556	5,565
10	5,548	5,556	5,565
11	4,511	4,519	4,527
12	2,647	2,655	2,664
13	3,182	3,190	3,198
14	5,586	5,595	5,603
15	7,002	7,010	7,019
16	7,002	7,010	7,019
17	5,586	5,595	5,603
18	3,182	3,190	3,198
19	3,182	3,190	3,198
20	5,586	5,595	5,603
21	7,002	7,010	7,019
22	7,002	7,010	7,019
23	5,586	5,595	5,603
24	3,182	3,190	3,198
25	2,647	2,655	2,664
26	4,511	4,519	4,527
27	5,548	5,556	5,565
28	5,548	5,556	5,565
29	4,511	4,519	4,527
30	2,647	2,655	2,664
31	1,608	1,617	1,625
32	2,626	2,634	2,643
33	3,145	3,154	3,162
34	3,145	3,154	3,162
35	2,626	2,634	2,643
36	1,608	1,617	1,625
37	0	0	0
38	0	0	0
39	0	0	0
40	0	0	0
41	0	0	0
42	0	0	0
43	0	0	0
44	0	0	0
45	0	0	0
46	0	0	0
47	0	0	0
48	0	0	0
49	0	0	0
50	0	0	0
51	0	0	0
52	0	0	0
53	0	0	0
54	0	0	0
55	0	0	0
56	0	0	0
57	0	0	0
58	0	0	0
59	0	0	0
60	0	0	0

Figure 5.31 - *Text file of the height limits for all nodes*

Objective functions R.txt - Notepad

NODE OBJECTIVE FUCTIONS R

Node number	Lower boundary [-]	Final R [-]	Upper boundary [-]
1	0,728	0,752	0,777
2	0,729	0,751	0,774
3	1,102	1,133	1,163
4	0,961	0,988	1,015
5	1,003	1,033	1,063
6	0,707	0,731	0,755
7	0,898	0,925	0,953
8	0,760	0,781	0,802
9	1,043	1,069	1,096
10	0,840	0,861	0,882
11	0,755	0,775	0,796
12	1,090	1,123	1,157
13	0,784	0,806	0,829
14	0,657	0,674	0,690
15	0,771	0,788	0,805
16	1,002	1,023	1,045
17	0,878	0,900	0,922
18	0,806	0,830	0,853
19	0,719	0,739	0,760
20	1,045	1,071	1,097
21	1,171	1,196	1,222
22	1,107	1,131	1,155
23	0,837	0,858	0,879
24	1,136	1,170	1,203
25	0,693	0,714	0,735
26	0,904	0,929	0,954
27	0,772	0,791	0,811
28	0,806	0,826	0,847
29	0,943	0,969	0,994
30	0,855	0,881	0,908
31	0,715	0,739	0,763
32	0,676	0,697	0,718
33	1,268	1,303	1,339
34	1,032	1,061	1,090
35	0,922	0,949	0,977
36	0,634	0,656	0,677

The lowest minimum r is: 0.6344559
The highest minimum r is: 1.2679008
The lowest maximum r is: 0.6769748
The highest maximum r is: 1.3390043

The minimum value of the top boundary R-values (0.6769748) is LOWER than the maximum
NO solution is possible!!

Figure 5.32 - *Text file of the final results for R, the inverse of the scalefactor ζ*

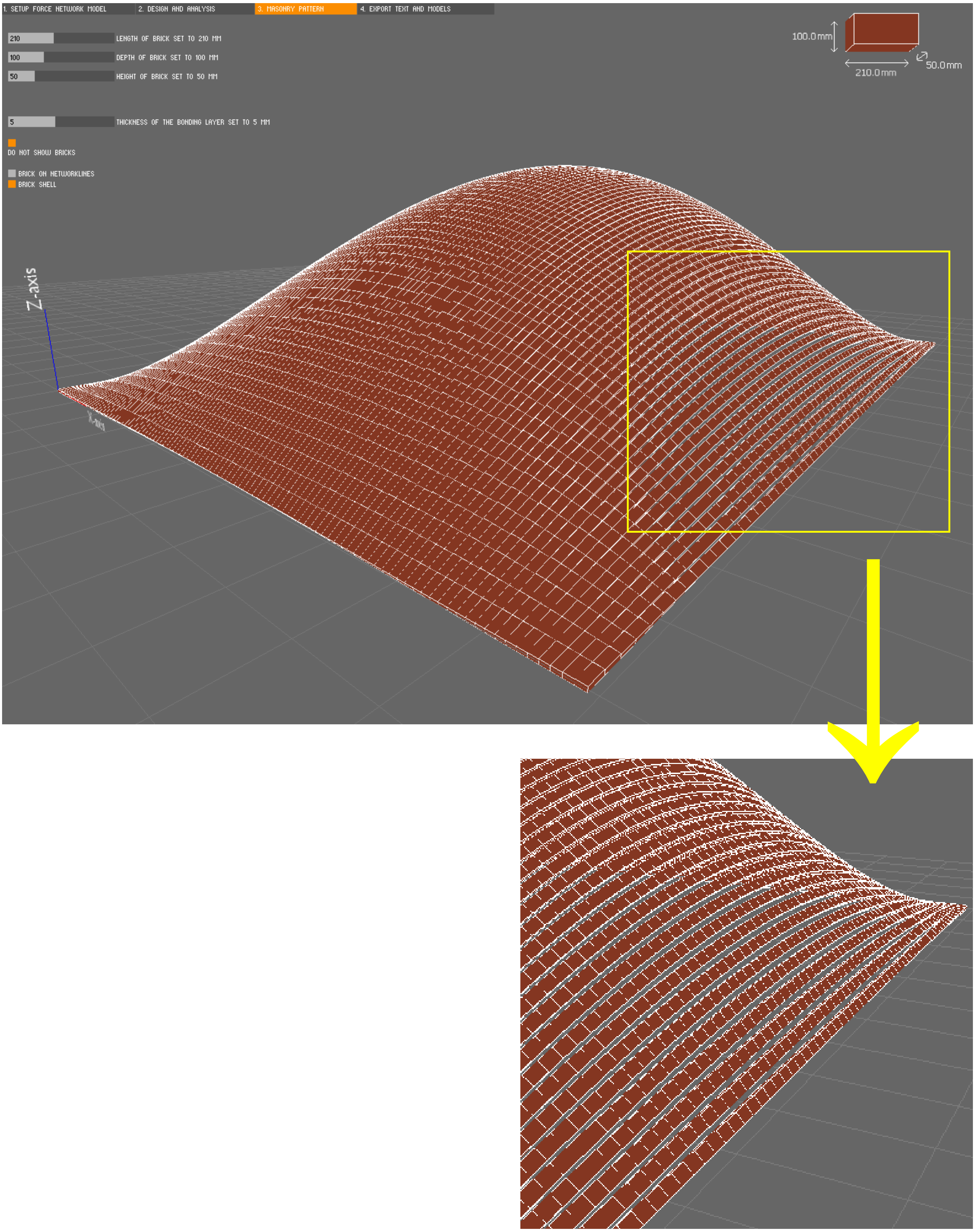


Figure 5.33 - *Gaps in perpendicular direction of the brick lines*

5. <u>EXPORT LAYERS OF THE MODEL</u>

It is possible to export several layers in the current version of the prototype, but in some occasions the model is not positioned in the origin of the axis and it is rotated.
This has to be reviewed in future research.

6. <u>NODAL LOADING</u>

In the current version of the prototype the initial load on every node is set to a certain value. In a realistic situation this value should be linked to the surface and thickness of the area surrounding the node.
The thickness of the shell surface can only be adapted for the complete shell. It is advised to make this possible only for selected areas of the shell as well.
More regarding this aspect is found in the report of P. Block [16].

7. <u>ANGLE CHECK</u>

The angle between the bricks must be implemented as a check whether the masonry pattern is realistic (§4.7).
This check consists of two parts:

1. Check of the angle of the selected brick and the neighbouring bricks within the centerline;
2. Check of the angle of the selected brick and the neighbouring brick in the neighbouring 'line' of bricks.

These checks are to be implemented in future research.

8. <u>GAPS IN THE MASONRY PATTERN</u>

The gaps between the lines of brick in the linear pattern must be repositioned so that the gaps in perpendicular direction are closed (Figure 5.33).
This extra step is to be implemented in future research.

Figure 6.1 *- Color variety in brick*

CHAPTER 6 PRACTICAL ASPECTS OF BRICK STRUCTURES

Introduction

In this chapter the aspects concerning brick structures in the building practice are discussed. During and after the design process an important aspect is the production and fabrication of brick. Which shapes are possible, which colors, which textures, etc (§ 6.1). Another aspects are the characteristics of brick concerning building physics, like water, fire resistance and insulation (§ 6.2). Last aspect for discussion is the construction and production of the structure. There are two options – in situ or prefab – but which is best to use for these brick 3D models (§ 6.3).
Several technological developments, discovered during research in the brick industry, are shortly discussed (§ 6.4).
The last paragraph shows an interesting alternative that was encountered during the course of the research (§ 6.5).

6.1 Brick fabrication and production

6.1.1 Non-standard brick forms

In a brick-factory everything happens automatic and on an assembly line. All machines for pouring, cutting and transport are restricting dramatic changes in size and shape of the brick. To keep the costs for the material low, it is therefore needed to stay within the limits, given by the machines in the factories.
The ranges of the dimensions which are used in the application are:

- Width (or length) 160 - 280 mm
- Height 75 - 120 mm
- Depth 50 - 90 mm

The costs for non-standard bricks are up to five times higher than for standard brick shapes and dimensions. This is caused by the need for manual labor to produce non-standard bricks. Moreover this process requires more time.

6.1.2 Texture

As mentioned in § 3.4 there are three possible ways of fabricating brick, regarding the texture. The choice of texture in this research is of small importance. Other aspects, like the color and shape of the bricks and actual design of the structure, are of bigger influence on the actual appearance, and so the attention will be focused on these aspects.

6.1.3 Color

The color of a structure and the patterns which can be made with it are of big influence on the actual appearance (Figure 6.1). Brick can be delivered in several colors, e.g. white, yellow, red and brown. The choice of color – for instance a light or darker one – affects the appearance of a building, but even more when several colors are applied in one structure – in other words: applying color patterns. As long as the

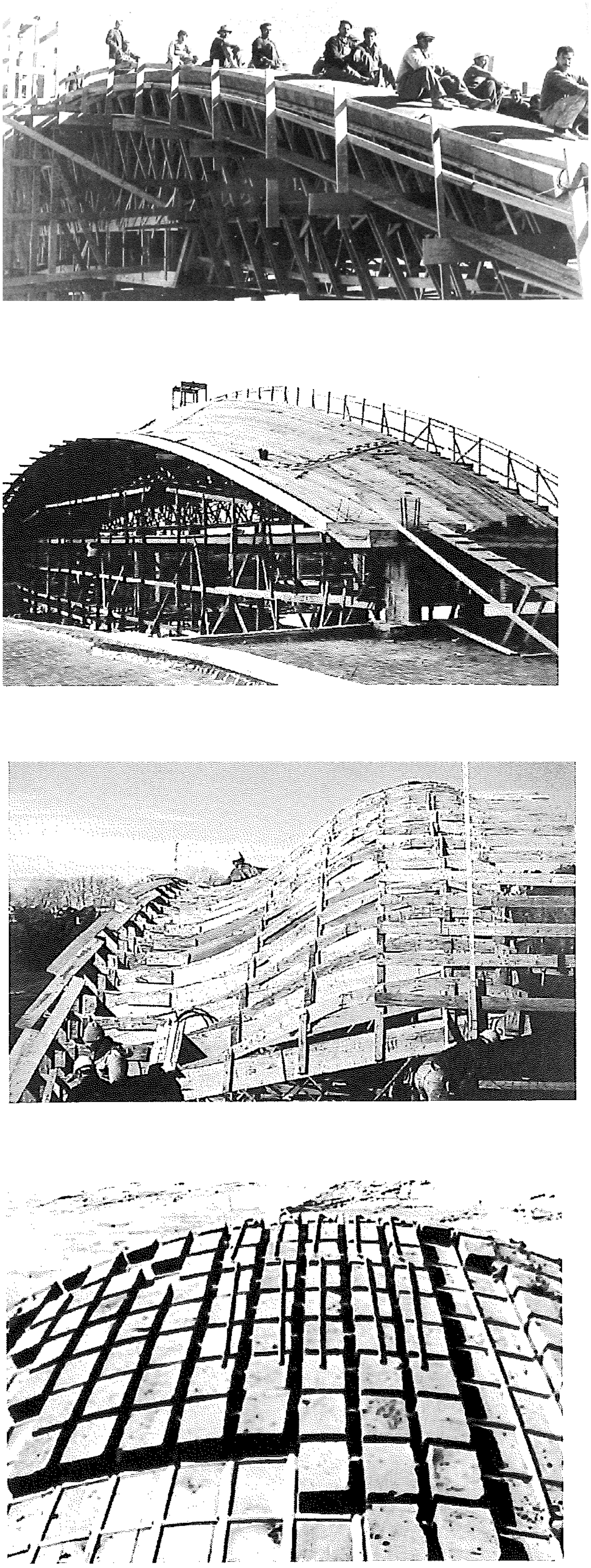


Figure 6.2 *- Wooden formwork to construct the structures of Eladio Dieste*

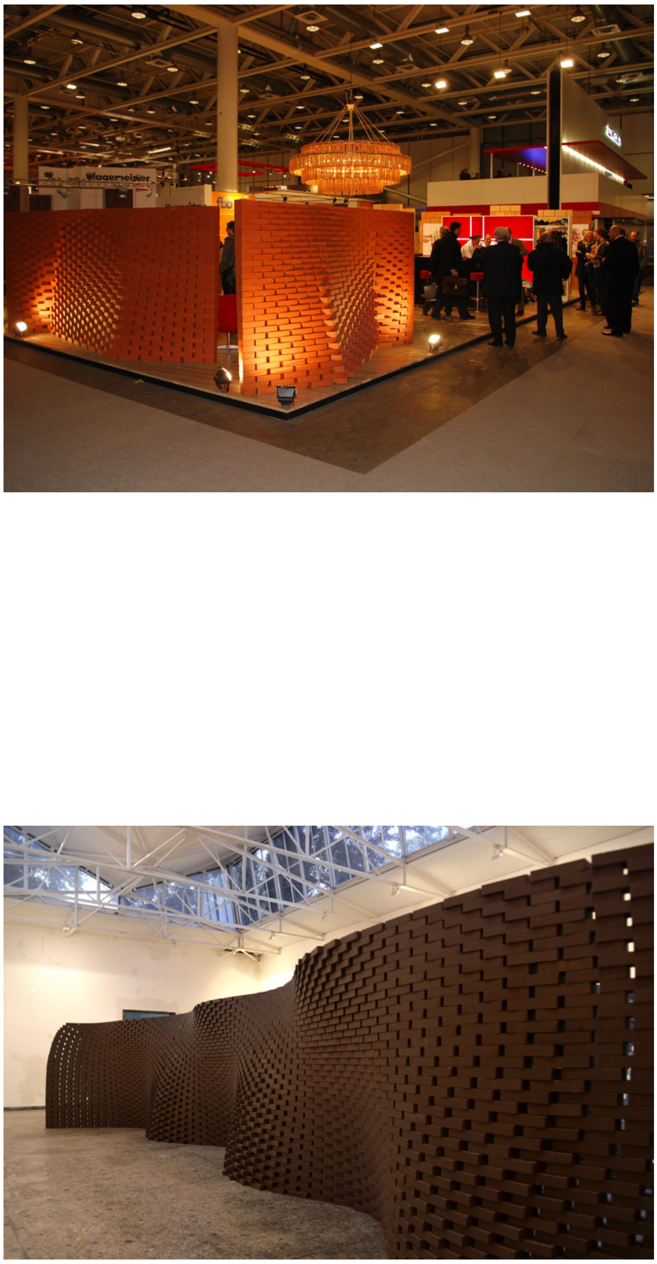


Figure 6.3 *- Brick patterns created by a robot laying the stones [19].*

chosen colors of the brick are commonly used, it should not be a problem to include several colors in the design of one structure.

6.1.4 Bonding material

As mentioned in § 3.4 there is a new bonding material making its entrance in the building practice: glue. Bonding layers are a lot thinner this way and the appearance of the building changes.
One of the functions of bonding material though is to apply curvature in brick structures. If this layer is made thinner, the angle of curvature per brick also decreases. This is exactly what is not wanted for this research. As a consequence this development will not be taken into consideration in this research.

6.2 Building physics characteristics

The main function for the models designed with the application are for now pavilions and other non-internal climate buildings. Though when a model is intended for an internal climate space, the aspect of building physics is very important. Aspects such as water penetration, fire resistance and insulation have to be checked and included in the design. Therefore to make the application better and more useful it is recommended to research how the building physics can be integrated in the application.

6.3 Building the structures

6.3.1 Prefab elements in fabric

In the factory prefab elements of brick are produced, in the same way as is done for concrete prefab elements. One of the options is to create a mould – made out of for instance plastic or concrete – in which the bricks are being placed. This way also even reinforcement could be added, which would increase the possibilities of shapes and models.
An example from building practice is the AKA-Blade System, used by CRH and other producers.[1] This system does not apply brick as a load bearing element though. It is only used as facade and appearance element. Therefore it is only used as an example of how a mould could be produced.

6.3.2 On site (with formwork)

Every brick has to be positioned in the designed pattern on site. To be able to do this, wooden formwork is needed, since the structure will not be able to resist any loading – even not its own weight – until it is completed. The buildings of Eladio Dieste were build using this technique (Figure 6.2).
This gives the impression as if the structure has to be built twice: the designed brick structure and the form work. With new techniques of laying bricks though – like the robot technique in Switzerland (§ 6.3.3) – it might not be needed to use form work for the whole structure.
Another option is to create a surface with the application, on which the brick pattern in shown. When this pattern and surface is approximated with a plastic mould, the bricks can be placed on their correct positions.
This surface has to be '*hold*' into position very precisely though and the technique to create 3D plastic moulds is still very expensive.

6.3.3 Laying patterns using a robot system

Gramazio & Kohler [19], together with the Architektur und Digitale Fabrikation at the E.T.H. Zurich, have done and are still in the process of developing a robot system to lay bricks in a designed pattern. There is a working model and several test projects are produces with it (Figure 6.3). More information is found in the next paragraph.

1 - http://www.crhclayproducts.com/

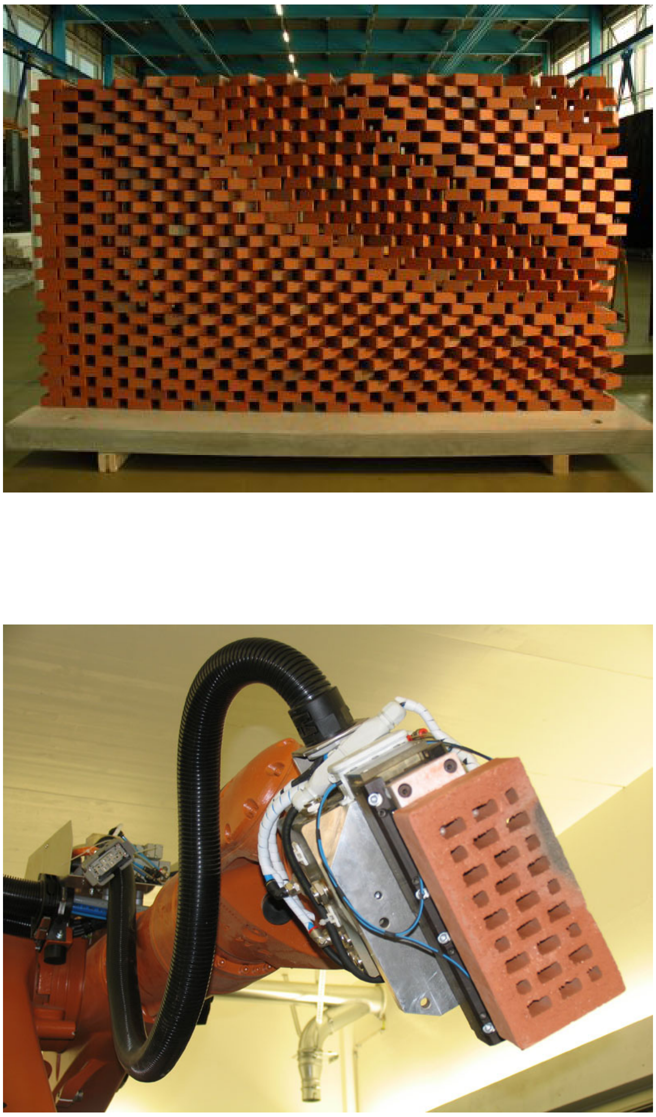


***Figure 6.4** - A brick pattern (upper picture) layed by the robot (bottom picture) created by the E.T.H. and Gramazio & Kohler [20] (Image courtesy of Gramazio & Kohler)*

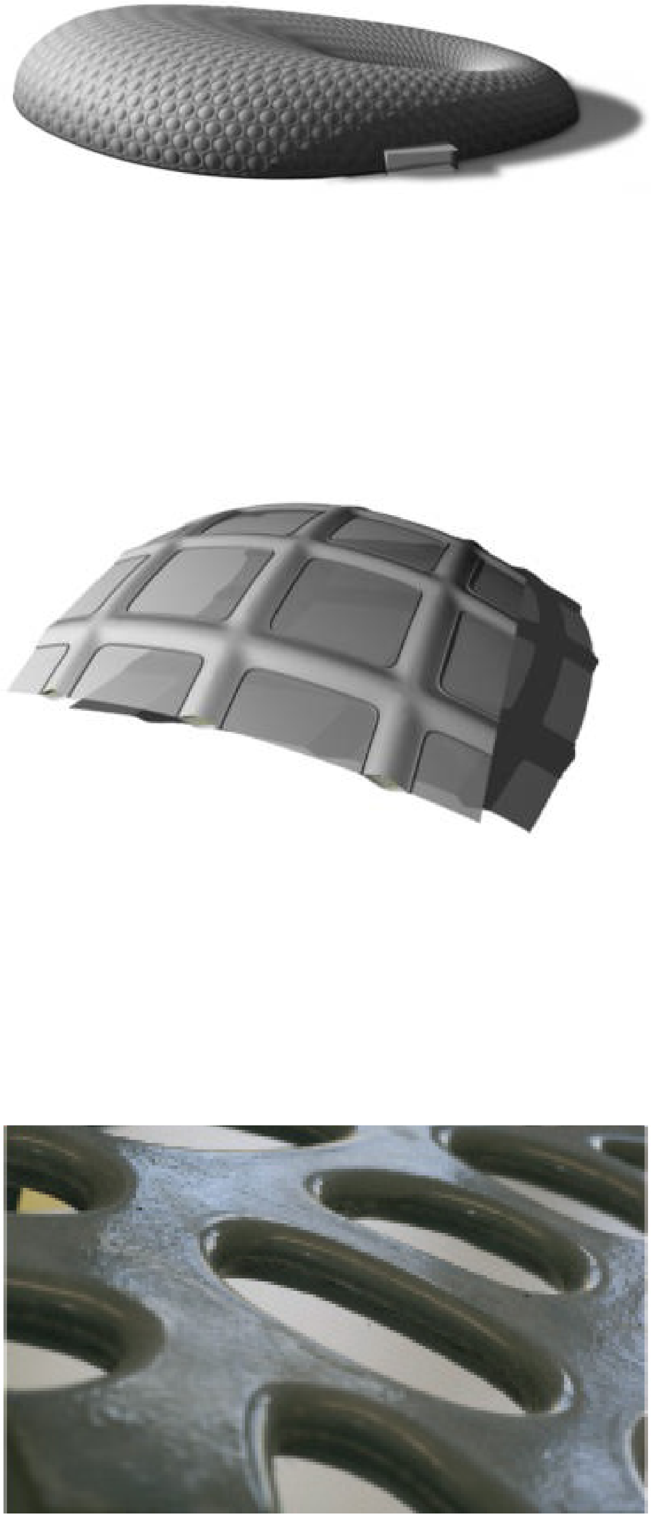


***Figure 6.5** - Concrete structures. A new technique created by GTecz (MBGT - Membrane Concrete Grid Shell) (Image courtesy of GTecz).*

6.4 Developments in the brick industry

One of the reasons for the increased possibilities of masonry and structures made of brick are technological developments and research. Three of these developments are discussed:

1. Glue as bonding material (§ 6.4.1);
2. Cutting brick stones in special shapes (§ 6.4.2);
3. A robot laying bricks in a computerized pattern (§ 6.4.3).

6.4.1 Glue as bonding material

Glue offers more strength as bonding material and for that reason could be of interest for this research. It needs a thinner bonding layer, about 2 - 5 mm, instead of 10 - 15mm, which is normal for mortar bonding.
On the other hand the bonding layer is needed to create the curvature in the structure. Moreover the strength of the masonry is determined by the weakest element of the combination of brick and bonding. The bonding layer has a comparable compressive strength as brick.
Due to these reasons glue is not considered as bonding material.

6.4.2 Cutting brick stones with a computer

Curved surfaces are difficult to make with only one specific brick-shape. In the case of a double curved surface this is even more complex and can even be considered impossible. Therefore it is important that several sizes of bricks can be used and if needed some unique shaped 'closing bricks'. This is possible, due to the cutting brick machines, which are becoming faster and more accurate.

6.4.3 Robot laying brick stones in a pattern

In Switzerland at the University of Zurich (E.T.H.) the department of Architecture together with Gramazio & Kohler [20] is currently researching the use of robots to lay patterns of bricks (Figure 6.4).
A disadvantage of the system in its current state is the requirement of having to lay the bricks in the right order, to assure the robot lays the right brick at the right position.
Considering the current capacities of the system, it is not a possibility to use for construction.

6.5 Interesting alternative

<u>STRUCTURE OF CONCRETE WITH A COVER OF BRICK</u>

Instead of a complete brick structure, concrete is used for the load bearing network structure with an external layer of brick in a certain pattern.
A German company, GTecz, has developed a new type of fluid Ultra High Performance Concrete [21]. They developed a technique, called Membrane Concrete Grid Shell (MBGT), with which networks shells of *liquid* concrete are created by using membrane in double layers (Figure 6.5).

This alternative combines the new possibilities of concrete with the classical appearance of masonry and therefore is an interesting option.

CHAPTER 7 CONCLUSION AND RECOMMENDATIONS

Introduction

In this chapter the conclusions and recommendations are presented.
The current results of the research are satisfying considering the aims and goals (Chapter 1 and 2). However several aspects concerning the application are up for discussion (§ 7.1). After this discussion the conclusions concerning the achievements are given (§ 7.2).
Before the application can be applied in the design process more research is required. Therefore several recommendations regarding aspects for further research are advised (§ 7.3).

7.1 Discussion

One of the main goals was to create an interactive tool, with which the architect and engineer can rapidly generate a conceptual shape for a masonry shell. Using the theory of Thrust Network Analysis [10] in combination with Catmull Rom splines, has provided the right conditions to design a first prototype for this tool.

<u>SHELL SHAPE</u>

Shell structures have a long history in technical development, materials and designs. Numerous shells have been designed and the material used to build them ranges from concrete and steel to timber. The range of buildable shapes is big, especially when several materials are combined, such as reinforced concrete.
However the shell designs in this research are bounded to *compression-only* and therefore the freedom and range of possible shapes and designs is reduced significantly.
One of the earliest goals for this research was *free form designing*. However during the research process this principle had to be moderated and eventually dropped. Two early made choices are the cause:

1. The choice to use the Thrust Network Analysis, which is based on compression only structures, and;
2. The assumption of no reinforcement, which again implies compression only structures.

Due to this the end product is not actually producing free form designs. The possible angles and curvatures are restricted and so the range of shapes is not as big as hoped for. Instead the designs are double-curved shapes.

<u>FUNCTION OF THE SHELL DESIGN</u>

An assumption was to take pavilions into account only. One of the reasons for this was the ability to neglect building physics in the design process.

THE TOOL

One of the aims of the research was to create a stand alone application, with the following abilities:

- To import and start with an earlier designed shape, in 3D software such as Rhinoceros;
- To export and use the result in the further design process.

The tool performs a force flow analysis of a shell shape. The stability, such as buckling behaviour, and the displacements have not been considered.

FEASIBILITY

The final goal of every design is to build and use it. Therefore it is important to consider the aspect of feasibility. From the point of view of *masonry* and *shells* the following aspects have to be looked at:

- Constructing the double-curved shell;
- Technical developments and alternatives.

Constructing the shell

- To be able to construct a shell, formwork is required;
- Laying bricks is labor intensive, especially if the pattern is not common. Therefore specialised bricklayers are needed to create the masonry shell.

Developments and alternatives

- Recent development: a robot laying bricks;
- An alternative is to combine materials: concrete as structural network and brick as cover (in- and/or outside the concrete structure).

The construction process of masonry shells is more labour intensive, expensive and time consuming in comparison with traditional brickwork as used in vertical wall-type structures.

However the result has a unique character, it stands out from its surroundings and it is better looking. The latter is a subjective matter and is therefore not a valid argument in the decision whether to use this construction and designing method.

REFLECTION

The prototype is a good start and has the potential to be developed in what it was meant to be. Research has been done in which elements and options are required for the application and an overview of this has been presented. It has been attempted to give the basic information regarding brick and analysis theories (TNA, Simplex method, Catmull Rom splines) and to integrate these in a new application. The user interface is developed and is considered to be easy in use and clear in displaying its content.

However there is one important remark concerning the script: it needs more development to make the tool complete before it can be used in practice. At the start of the research the level of scripting knowledge was low and as a consequence a big part of the research time was invested in obtaining this knowledge. Nevertheless in the end the knowledge is still not at that level, with which an application can be developed in a smooth and fast process. Unfortunately as a result certain aspects were not scripted and integrated in this prototype.

7.2 Conclusions

The conclusions have been split into two categories:

- Programming aspects;
- Technical aspects.

PROGRAMMING ASPECTS

DOUBLE CURVED SHELLS

With the prototype double-curved masonry shells are analysed and a brick pattern is generated. The range of possible shell shapes is not as big as was aimed for at the start of the research.

The analysis part of the program is partly working in the current version of the tool due to problems with the generation of the dual grid and with performing the analysis. To make the analysis step fully functional the theory regarding the use of several matrices [16] has to be implemented.

The script to generate the masonry pattern is partly working, but fails at certain places of the pattern. This is solved by reviewing the script by an professional.

INTERACTIVE

The application offers a good interactivity between the users and the application and the results. A shell based on five parameters can be created. The user has the ability to change the position of nodes in the model and adapt analysis-variables, such as the loading. Several layers can be exported, such as the masonry pattern for the producer, the surface model for architectural renderings and the force network for further analysis by the engineer.

CONNECTION WITH OTHER SOFTWARE

The ability to import files is reached to a certain extent, where the ability to export is incorporated nearly completely.

Import

In the current version of the application it is possible to import an .obj file. When it is imported, it can be used as shape to approximate with a parametric model.

Export

Each layer created and used in the application is exportable. To assure the layers are usable in other programs, such as AutoCAD and Maya, several file formats are available. The export files are based on vertex and line models.

TECHNICAL ASPECTS

THRUST NETWORK ANALYSIS

The Thrust Network Analysis is a good theory to make a force flow analysis of compression only network structures. Therefore it is useful for the application. It offers a fast method to make an accurate analysis of the force flow for the conceptual design stage.

A disadvantage of the theory is the compression-only boundary. To extend the possibilities of the application, the theory should be researched into the option of adding tension.

CATMULL ROM SPLINES

This theory is used to create the surface and pattern of the shell. It uses the location of four neighbouring nodes to determine the location and angle of a specific point. It is a fast method and in combination with the factor τ the curvature is limited. Therefore this theory is perfect to use in this research.

7.3 Recommendations

The recommendations have been split into two categories:

- Programming aspects;
- Technical aspects.

PROGRAMMING ASPECTS

FILE FORMATS

Import

To increase the functionality of the application it should become possible to import more file formats, for instance NURBS models from Rhinoceros or Maya.

Export

An aspect that is not working in all cases is to export the model without rotation and placed in the centre of the coordinate system.

FORCE NETWORK

Moreover the application requires the ability to create a suitable force network for the imported shape(s).

BRICK PATTERN

The script to generate the brick pattern needs more research. In the current situation each brick is treated as an individual element, while it is intended that all of them form one unity. The method to calculate the angles of each brick individually has to be looked at further, to improve the resulting pattern. The pattern has to be optimised to assure there are no gaps.

IMPROVE THE FREEDOM OF DESIGN AND FORM FINDING

To extend the freedom of designing and form finding, several options should be added and integrated:

- The option to use more than one shape of brick in a design and to use non-standard shaped bricks;
- The option to use more than one color of brick in a design, so that color-patterns can be applied.

IMPROVE THE FUNCTIONALITY OF THE APPLICATION

The script code of the application must be looked at by a professional tool developer, so that the performance of the application is improved.

Moreover the performance of the application is improved when this option is implemented:

- The ability to move around the model in the X-Y plane.

TECHNICAL ASPECTS

TENSILE FORCES

Compression-only structures restrict the freedom in form finding, especially in free form design.

To make it possible to design shapes in which tensile forces are active, more research needs to be done in improving the application and theories.

BUCKLING OF SHELL STRUCTURES

This aspect is not looked at in the research. It should be researched before actually using an output of the application, since it is an important factor to check before constructing and using a shell.

NODAL LOADING

The nodal loading is determined manually. To improve the results and make them more realistic, the nodal loading needs to be determined by the loading of the surface area related to the specific node.

SHELL THICKNESS

When a shape is not buildable with one layer of bricks, a situation may exist where it can be built but with more layers in certain areas. The implementation of this option will be an improvement for the application.

BUILDING PHYSICS

To be able to use the models, proposed by the application, in other functions besides pavilions, such as roofing of internal-climate spaces, one needs the building physics during the design process.

For this the building physics of brick structures and the related codes and demands needs to be integrated in the application.

CHAPTER 8 REFERENCES

BOOKS

[1] Coenders, J.L., 2007. *Dictaat CT5251 Structural Design – Special structures*, 2nd ed.

[2] Pedreschi, R., 2000. *The Engineer's Contribution to Contemporary Architecture: Eladio Dieste*.

[3] Chilton, J., 2000. *The Engineer's Contribution to Contemporary Architecture: Heinz Isler*.

[4] Pottman, H. Asperl, A. Hofer, M. & Kilian, A, 2007. *Architectural Geometry*, Bentley.

[5] Reas, C. & Fry, B., 2007. *Processing: A Programming Handbook for Visual Designers and Artists*

[6] Kanellos, A., 2007. *Topological Self-Organisation: Using a particle-spring system simulation to generate structural space-filling lattices* [internet].
Available at: http://eprints.ucl.ac.uk/4985/
[Accessed September 2008]

[7] Horssen, W.T. van & Burgh, A.H.P. van der, 1999. *Inleiding Matrixrekening en Lineaire Optimalisering*

[8] Cremona, L., 2007. *Graphical Statics - Graphical Calculus and Reciprocal Figures in Graphical Statics*.

[9] Kahn, L., *Attributed to Louis Kahn (1903-1974)*

ARTICLES

[10] Block, P. and Ochsendorf, J., 2007. *Thrust Network Analysis: A new methodology for three-dimensional equilibrium*, Journal of the International Association for Shell and Spatial Structures, no. **48**, pp. 167-173.

[11] Maxwell, J.C., 1864. *On Reciprocal Figures and Diagrams of Forces*.

[12] Kilian, A. and Ochsendorf, J., 2005. *Particle-Spring Systems for structural form-finding*, Journal of the international association for shell and spatial structures, no. **148**, pp. 77-84

[13] Chak, D. Galbraith, M. and Kilian, A., 2002. *CatenaryCAD: An Architectural Design Tool - Final Project Report* [internet].
Available at: http://designexplorer.net/newscreens/cadenarytool/final_paper.pdf
[Accessed July 2008].

[14] Kilian, A., 2004. *Linking Hanging Chain Models to Fabrication*. For ACADIA conference.

[15] Catmull, E. and Rom, R., 1974. *A class of local interpolating splines*. In Computer Aided Geometric Design, R. E. Barnhill and R. F. Reisenfeld, Eds. Academic Press, New York, pp. 317–326.

[16] Block, P., 2009. *Thrust Network Analysis: Exploring Three-dimensional Equilibrium*.

WORLD WIDE WEB

BRICK AND MASONRY

[17]
Wienerberger. [Online] (Updated 2009).
http://www.wienerberger.nl/
[Accessed June 2008].

[18]
The International Union of Bricklayers and Allied Craftworkers. [Online] (Updated 2009).
Available at: http://www.bacweb.org/
[Accessed June 2008].

[19]
Koninklijk Verbond van Nederlands Baksteenfabrikanten. [Online] (Updated 2009).
Available at: http://www.knb-baksteen.nl/
[Accessed June 2008].

[20]
Gramazio & Kohler. [Online] (Updated 2009).
Available at: http://www.gramaziokohler.com/
[Accessed July 2008].

[21]
GTecz. [Online] (Updated 2009).
Available at: http://www.gtecz.com/
[Accessed May 2009].

THEORIES AND METHODS

[22]
Richland Community College, 2003. *Gauss Jordan Elimination Through Pivoting*. [Online] (Updated 7 January 2009).
Available at: http://people.richland.edu/james/lecture/m116/matrices/pivot.html
[Accessed October 2008].

[23]
Dr. Hossein Arsham, 1994. *The Classical Simplex Method.* [Online] (Updated 2009)
Available at: http://home.ubalt.edu/ntsbarsh/Business-stat/opre/partIV.htm
[Accessed October 2008].

COMPUTATIONAL DESIGN

[24]
Fry, B. and Reas, C., 2001. *Processing*. [Online] (Updated 14 May 2009).
Available at: http://processing.org/
[Accessed September 2008 - May 2009].

[25]
Kilian, A., 1998. *Designexplorer*. [Online] (Updated 2006).
Available at: http://designexplorer.net/
[Accessed May 2008].

[26]
Masonry Research Group (M.I.T.), 2006. *Masonry at MIT*. [Online].
Available at: http://web.mit.edu/masonry/
[Accessed July - September 2008].