

# **POLITECNICO DI MILANO**

Department of civil and environmental engineering

Master's degree in Civil Engineering

Track Structures



## **A method for robust design of diagrid structures**

Supervisor: Prof. Capsoni Antonio

Candidate:

Lavorini Luca

ID n°: 873655

Academic year 2017/2018



# Abstract

This work aims to define a method to design robust diagrid for every load condition present in the structure environment and it focuses the attention to diagrids in high-rise buildings, in order to withstand especially lateral loads. Starting with an introduction of the various types of structure and the definition of the robustness requirement, optimization methods are introduced because of their importance in the robust design of structures. When describing the main methods, such as Michell's truss, principal stresses trajectories and topology optimization, and detailing the last one, it has been observed that they are all connected and lead to the same result, which is a kind of natural layout described by the force flow through the structure volume. Then, understanding the tall building problem, a single load case evaluation of the optimal layout is implemented on a simple example like a square box section cantilever beam under uniform transversal load orthogonal to one face and later, for the robust design, the same system is subjected to a multi-loading case with the same load that rotates around the structure. This last case is the key for robust design, because the wind impact angle is not a defined parameter, but it is the source of uncertainty that characterizes the robustness. It is then demonstrated that the robust layout follow the principal stresses trajectories of the system, a numerical example using those trajectories layout is being proposed to prove the efficiency of the approach and, at the end, some existing practical example are being listed to prove this approach.

# Contents

Abstract .....	3
1. An introduction to diagrid structural systems.....	1
2. About the idea of structural robustness .....	5
2.1. Robustness as structural collapse resistance .....	6
2.2. Robustness as best performance for every situation.....	7
3. Optimization problem history .....	9
3.1. Maxwell's theorem and Michell's trusses .....	9
3.2. Evolution of optimization methods .....	14
3.3. Principal stresses trajectories .....	14
3.4. Material distribution methods .....	15
4. Material distribution method – Topology optimization by minimum compliance design .....	17
4.1. Method description .....	18
4.2. SIMP method .....	22
4.3. Solving procedure .....	23
4.4. Mesh dependency .....	25
4.5. Analogy between different volume fractions cases.....	26
4.6. Multi-load case implementation.....	28
5. High-rise buildings mechanics.....	29
5.1. Problem introduction .....	29

5.2.	Principal stresses trajectories .....	31
5.3.	The shear-lag effect on stress distribution and principal trajectories	39
6.	Connections between principal stresses trajectories, Michell's trusses and topology optimization.....	43
6.1.	Principal stresses trajectories and topology optimization .....	43
6.2.	Topology optimization model and Michell truss.....	45
6.3.	Michell truss and principal stresses trajectories .....	46
7.	Multi-load optimization.....	49
7.1.	Model.....	49
7.2.	Single load condition.....	53
7.3.	Multi-load condition.....	58
7.4.	About vertical elements.....	59
8.	Diagrid with principal stresses trajectories configuration for robust design	61
8.1.	The web layout diagrid.....	61
8.2.	2D test .....	63
8.3.	3D test .....	67
8.4.	Nowadays examples.....	70
9.	Conclusions.....	74
10.	Appendix.....	76
10.1.	MatLab code for principal stresses trajectories with no shear-lag effect	76

10.2.	MatLab code for principal stresses trajectories with shear-lag effect	97
10.3.	Topology optimization by TOSCA with continuum-beam approach	123
10.4.	Vertical elements in the final robust diagrid .....	128
11.	Bibliography .....	133

# Figures summary

Figure 1.1 Vladimir Shukhov [1853-1939] (right), Shukhov tower in Polibino [1896] (left) .....	1
Figure 1.2 30 St Mary Axe, called "the Gherkin", London, United Kingdom, by Foster and Partners .....	3
Figure 1.3 Guangzhou TV Astronomical and Sightseeing Tower, usually called "Canton tower", Canton, China, by Information Based Architecture (IBA): Mark Hemel and Barbara Kuit .....	4
Figure 1.4 Aldar Headquarters building (right), by MZ Architects, and Capital gate (left), by RMJM, Abu Dhabi, United Arab Emirates .....	5
Figure 2.1 Ronan Point, evidence of the partial collapse (1968).....	6
Figure 3.1 James Clerk Maxwell [1831-1879] (right), reciprocal diagrams (left) of the structure, (a), and forces (b) .....	10
Figure 3.2 Anthony George Maldon Michell [1870-1959] (right), cantilever problem in a free boundary space (left) .....	12
Figure 3.3 a) principal moments pattern for a slab modulus and b) the Gatti Wool Factory floor .....	15
Figure 4.1 Simply supported beam problem.....	17
Figure 4.2 Sizing optimization solution for simply supported beam problem ..	17
Figure 4.3 Shape optimization solution for simply supported beam problem ..	18
Figure 4.4 Topology optimization solution for simply supported beam problem .....	18
Figure 4.5 General body system.....	19

Figure 4.6 Mesh dependency of the optimization problem for a simply supported beam, from a) to c) the number of finite elements increases having the same amount of volume.....	25
Figure 4.7 Comparison between layouts with different amount of volume, case a) 30%, case b) 60% and case c) 90% .....	26
Figure 4.8 Cantilever problem, with topology results of volume fraction 20% a), 40% b), 60% c) and 80% d) .....	27
Figure 5.1 3D square section cantilever problem under uniform distributed transversal load with varying direction.....	29
Figure 5.2 Section stress state for 3D square section cantilever problem under orthogonal uniform distributed transversal load, normal stress $\sigma_z$ (right) and shear stress $\tau_{xz}$ and $\tau_{yz}$ (left) .....	30
Figure 5.3 2D cantilever problem under in-plane uniform distributed transversal load (right) and section stress state (left) with normal stresses $\sigma_z$ (top) and shear stresses $\tau_{xz}$ (bottom) .....	32
Figure 5.4 Principal stresses trajectories for 2D cantilever problem under transversal uniform distributed load, tension trajectories (right) and compression trajectories (left) .....	34
Figure 5.5 Cross-section faces names (right) and impact angles (left).....	35
Figure 5.6 Principal stress trajectories for 3D cantilever problem at impact angle $0^\circ$ .....	36
Figure 5.7 Principal stress trajectories for 3D cantilever problem at impact angle $22.5^\circ$ .....	37
Figure 5.8 Principal stress trajectories for 3D cantilever problem at impact angle $45^\circ$ .....	37



Figure 5.9 Principal stress trajectories for 3D cantilever problem at impact angle 67.5° .....	38
Figure 5.10 Principal stress trajectories for 3D cantilever problem at impact angle 90° .....	38
Figure 5.11 Bold lines represent shear-lag distribution and the dotted line the Eulerian stresses distribution. The case a) is for sections close to the clamped edge, so in typical shear-lag, while the case b) is close to the free edge, so in negative shear-lag.....	40
Figure 5.12 Section stress state for 3D square section cantilever problem under orthogonal uniform distributed transversal load, normal stress $\sigma_z$ with shear-lag effect (right) and shear stress $\tau_{xz}$ and $\tau_{yz}$ (left) which remain invariant .....	40
Figure 5.13 Principal stresses trajectories for 3D cantilever problem at impact angle 0° and shear-lag effect .....	41
Figure 5.14 Principal stresses trajectories for impact angle 0°. Case a) webs for no shear-lag (left) and shear-lag (right), case b) flanges for no shear-lag (left) and shear-lag (right) .....	42
Figure 6.1 Comparison between principal stresses trajectories layout, a), and topology optimization result, b). Adjusting principal trajectories, c), a similar layout is obtained, d).....	44
Figure 6.2 Comparison between topology optimization, a), and Michell's truss b) .....	45
Figure 6.3 Comparison between topology optimization, a), and Michell's truss b) from Bendsøe and Sigmund book.....	46
Figure 6.4 Comparison between Principal stresses, a), and Michell's truss b) from obtained connecting intersection nodes between trajectories .....	47
Figure 7.1 ABAQUS model, load assignment method .....	50

Figure 7.2 ABAQUS model, load redistribution.....	51
Figure 7.3 Load dependency of symmetry .....	51
Figure 7.4 ABAQUS model, general direction load repartition.....	52
Figure 7.5 ABAQUS model, final configuration with beam elements.....	53
Figure 7.6 ABAQUS model, single load natural symmetry .....	54
Figure 7.7 ABAQUS model, SIMP results for single load case, general overview .....	54
Figure 7.8 ABAQUS model, SIMP results for single load case, web side view.	55
Figure 7.9 ABAQUS model, SIMP results for single load case, flange side view .....	55
Figure 7.10 ABAQUS model, with single load equal faces and imposed symmetry .....	56
Figure 7.11 ABAQUS model, SIMP results with single load equal faces and imposed symmetry, general overview.....	57
Figure 7.12 ABAQUS model, SIMP results with single load equal faces and imposed symmetry, side overview.....	57
Figure 7.13 ABAQUS model, general view of multi-load topology result .....	58
Figure 7.14 ABAQUS model, façade layout from multi-load topology result ...	59
Figure 8.1 Diagrid configuration for web layout, lines with starting point pitch, from left to right at H/B, H/2B, H/4B, H/8B, H/16B. Configuration without shear- lag (top) and with shear-lag (bottom).....	62
Figure 8.2 Grasshopper modelling comparison for three layouts in 2D, web layout (right), flange layout (centre), regular layout (left).....	64
Figure 8.3 Load condition and constraints conditions .....	65

Figure 8.4 Grasshopper modelling, deformed configurations .....	66
Figure 8.5 Grasshopper modelling, 3D models, web model (left), PST model (middle) and regular model (right).....	67
Figure 8.6 Grasshopper modelling, load cases, 0° (left), 45° (middle) and 90° (right).....	68
Figure 8.7 Transbay tower competition, San Francisco, California, designed by SOM.....	71
Figure 8.8 a) Principal stress trajectories layout concept, from Beghini et al [OPTIMIZATION] and b) the Kunming Junfa Dongfeng Square, Yunnan, China, designed by SOM .....	71
Figure 8.9 CITIC Financial Center, designed by SOM.....	72
Figure 8.10 Burj-Jumeira tower, Dubai, designed by SOM.....	72
Figure 10.1 Cantilever system for shear-lag computation.....	97
Figure 10.2 From Stromberg at all paper, a) is the case of a meshed rectangle under a set of transversal load, and b) is the same system but with the contribution of vertical beams at each lateral side .....	124
Figure 10.3 ABAQUS model, results from the first topology analysis without beam elements .....	125
Figure 10.4 Diagrid evaluation for columns pre-sizing.....	126
Figure 10.5 ABAQUS model, Continuum-Beam model .....	127
Figure 10.6 ABAQUS model, cylinder model .....	128
Figure 10.7 ABAQUS model, four clamped points general view .....	129
Figure 10.8 ABAQUS model, four clamped points lateral view .....	130
Figure 10.9 ABAQUS model, eight clamped points general view .....	130

Figure 10.10 ABAQUS model, eight clamped points lateral view .....	131
Figure 10.11 ABAQUS model, sixteen clamped points general view .....	131
Figure 10.12 ABAQUS model, sixteen clamped points lateral view .....	132

# 1. An introduction to diagrid structural systems

The word diagrid is a portmanteau of “*diagonal grid*” and in the field of structural engineering gives the noun to a structural framework of diagonal intersecting beams, which may be made by one or more materials. This bearing system is often used in buildings and roofs for its efficiency in term of stiffens and material costs, because it usually requires less material that another system configuration.

Nowadays, a widespread application of diagrids is in large span and high-rise buildings, where due to the large-scale system the reduced costs have been hugely highlighted.

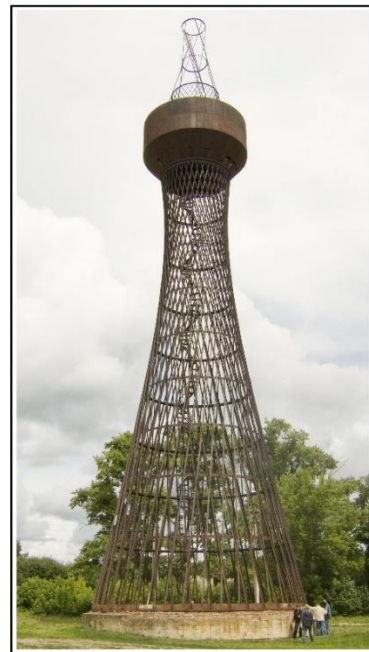


Figure 1.1 Vladimir Shukhov [1853-1939] (right), Shukhov tower in Polibino [1896] (left)

The origin of “diagonal structures”, as described by architect Ian Ritchie [47], is the work made by the Russian genius Vladimir Shukhov [1853 – 1939], who pioneered new analytical methods in many different fields. Shukhov left a considerable legacy to early Soviet Russia constructivism leading engineers to the design of hyperboloids, thin shells and tensile structures of extraordinary refinement and elegance basing on diagonal members mechanisms. One of his most famous diagrid systems, which should be considered the “father” of diagrid structures, is the Shukhov tower in Polibino, designed in 1896 and built between 1920 and 1922. It represents the essence of this kind of structure, having no vertical elements, being slender and lightweight.

Focussing on tall buildings, the study of diagrids gets a spark to develop new concepts and applications. The most influencing actions in high rise building design are the lateral loads mostly given by wind and earthquake, which are carried by exterior or interior structural systems. The diagrid is one of the most common solution for this purpose due to its duality in structural efficiency and architectural advantages, both aesthetically and functionally speaking.



*Figure 1.2 30 St Mary Axe, called "the Gherkin", London, United Kingdom, by Foster and Partners*

Commonly, the diagrid is made by diagonals elements only and doesn't require any vertical column, because the inclined members, intersecting each other, bear both shear and overturning moment effects, acting as "inclined columns" and bracing elements". Over the years, it has been presented a solution for both vertical and lateral actions, which confirm a reduction of material costs, estimated around 20% compared to conventional moment-frame structures[7].

Apart from evident costs advantages, the diagrid system allows a large use of glass façade which, form a comfort point of view, allows a generous amounts of day light into the building. Furthermore, it represents one of the most feasible solution in terms of energy efficiency and low environmental costs, which represent nowadays some of the main topics for a sustainable design.

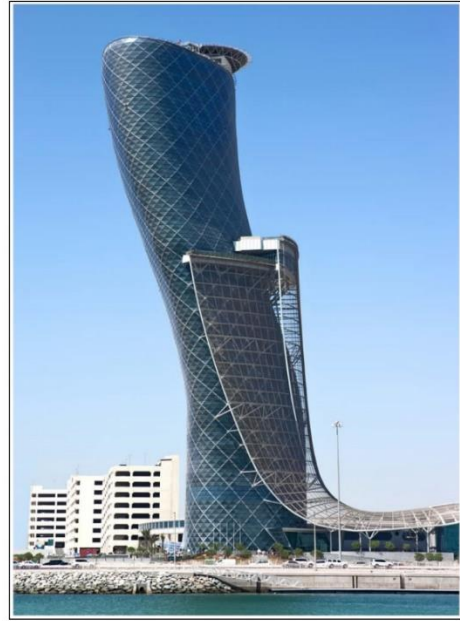


*Figure 1.3 Guangzhou TV Astronomical and Sightseeing Tower, usually called "Canton tower", Canton, China, by Information Based Architecture (IBA): Mark Hemel and Barbara Kuit*

As every structural solution, diagrids present some drawbacks or demerits. The noted complications are focussed on glasses modelling, nodes fabrication and construction. The first point is quite evident, because it involves irregular shapes and windows sequences, while the other two, that are more technical, request adequate sources.

By the way, as every structure, diagrids are not exempt from technical considerations. Their features of being versatile and aesthetically pleasant are “intrinsically” fixed into the proposed configuration, but the structural efficiency still depends on the designer ability to imagine the real environment impact during the structure life.





*Figure 1.4 Aldar Headquarters building (right), by MZ Architects, and Capital gate (left), by RMJM, Abu Dhabi, United Arab Emirates*

This work will focus on diagrid system on high-rise buildings and where the most significant environmental effect is given by lateral loads, which will be considered purely as wind actions. Being the wind and earthquake directions unpredictable, the efficiency of the diagrids is not evaluated on classical design actions only, but it takes into account the uncertainty related to loads, generating a elements layout that may have the best performance for every circumstance. This capacity is called “structural robustness”, and being related to the best performance, it is also strictly related to “optimization design approaches” which are used to determine the most efficient layout of systems.

Knowing the type of structure that is involved, the next step for a designer is to catch the philosophy that withstand behind his design choices. So, what is exactly the structural feature called “robustness” and how it exists into the structure environment.

## 2. About the idea of structural robustness

The adjective “robust” takes its origins from the Latin world, where the term “robustus”, from “robur” and earlier “robus”, has the meaning of strength, sturdy or hardy, and gives also the name to oaks trees, for their resistance and their red colour on leaves, “rubeus”, which is symbol of blood and life. As this last reference suggests, this adjective is used for subjects able to resist to different or adverse conditions far from the average one, or, in other words, to withstand against ad huge number of situations, like life presents.

Over the years the noun “robustness” is then used referring to the same root of strength and it is introduced in various knowledge fields indicating, as Woliński suggested [58], the property of a system to survive unforeseen or extraordinary exposures or circumstances that would cause failure or loss of functionality.

Almost every science defines in their fields this particular attitude in special subjects, for example in biology the idea of robustness has been used in field of evolution, meaning the persistence of a system’s characteristics after perturbation, and in morphology we define as robust species with a body-system based on strength. Moving to more practical studies, like economy, the word finds its meaning as the ability of a financial trading system to remain effective under different markets conditions, or in computer science, as the ability of a computer system to deal efficiently with errors, from the execution or from input, that should occur in every moment.

On the other hand, in more mechanical engineering field, such as civil, mechanical or aerospace, the term robustness has different meanings gravitating around the same root, and the definitions itself sometimes is one of the main issues, as Banu said [6]. Following literature and codes, two flow-ideas emerge to establish the body of such property.

## 2.1. Robustness as structural collapse resistance

In the most classical sense robustness is a structural requirement not to let the partial collapse of structure component doesn't cause or induce the global or partial collapse of the structure or using the words by Starossek and Haberland [51], it is “a desirable property of structural systems which mitigate their susceptibility to progressive or disproportionate collapse”. The need to “control” in some way the partial collapse of a structure in order to prevent a much more global functionality and safety born during the last years of the XXth century, when on 16 May 1968 the tower block Ronan Point lost one entire corner along the height due to a local explosion.



*Figure 2.1 Ronan Point, evidence of the partial collapse (1968)*

The cause was a gas explosion in the kitchen of an 18th floor flat which resulted in the collapse of the entire south-east corner of the 22-storey east London tower block, killing four people and injuring seventeen. The disaster led to a loss of public confidence in high-rise residential buildings and resulted in major changes in UK building regulations on 1970. According to Pearson and Delatte

[9], from that point British government mandated guidelines for the prevention of progressive collapses, these instructions including the requirement of fail-safe mechanism and require that under specified loading conditions a structure must remain stable with a reduced safety factor in the event of a defined structural member or portion thereof being removed. The lessons from Ronan Point changed building regulations throughout the world and over 50 years after the collapse every code around the nations presents sections which stress the idea of structural integrity and the aim to design so that structures sustain local damage with the system globally remaining stable. This requirement, at the end, takes the name of structural robustness, but even after all that safety goals it doesn't exist and unique definitions or description, neither a formula to compute it. It is strictly related to the probability of progressive collapse and the risk to special event occurrence, which allows to determinate the probability of collapse resistance, but there is no proper robustness parameter evaluation for that requirement. Mentioning the Eurocode definition, in "BS EN 1991-1-7 Actions on structures. General actions, Accidental actions" [8], it is written:

*"Robustness is the ability of a structure to withstand events like fire, explosions, impact or the consequences of human error without being damaged to an extent disproportionate to the original cause."*

## **2.2. Robustness as best performance for every situation**

Another way to define the structural robustness is related to the idea that the undamaged structure use to resist in the most optimal way to all possible load situations which were not included in the design process, like, for example in tall buildings, all possible wind blowing directions or earthquake action directions. This point of view born realizing that the world is full of uncertainties and structures need, following Dunning at al. [42], to be designed and optimized to be robust when operating in an uncertain environment. The traditional engineering approach to account for uncertainties is to employ a factor of safety. While this philosophy has produced many successful structures, the modern era

demands more efficient designs that minimize waste in order to meet pressures from both economic and environmental factors. In this context the safety factor approach may be inadequate and, taking into account, for example, the considered design loads, the load directions and their magnitude should be the source of uncertainty. Then, as Csébfalvi says [11], the investigation of the most optimal structure design, and so the most robust one, is frequently sensitive to the directional uncertainties of the applied loads. The hardest question is then, which is the optimal layout of material that guarantee the best performance to an uncertainty environment?

Summarizing those two orientations of robustness in structures, we have from one side a sufficient performance in terms of functionality in damaged situations, which is necessary for safety reasons as history has shown, and from the other the ability to behave in the most optimal way to the whole possible load situations. This last requirement is the one that matches the initial definitions of robustness and is the one that defines the guide lines for the best design, and that influences also the other robustness requirement.

The description could move on describing in detail the relations that withstand between robust design and optimal design, and the methods to investigate this last performance requirement.

### 3. Optimization problem history

The optimization problem in structural engineering usually concerns material disposition of a volume. This study is about the philosophy which manages the material distribution in a given space, standing into given system constrains and following the aim of best performances.

The investigation of the optimal design layout is a requested target nowadays in building industries, especially in high-rise buildings, due to, as Sarkisian at al. declare [35] the least amount of material goal for economic and environmental reasons. It must be noted that behind those motivations the least material amount techniques are often used because of the duality between pleasant aesthetic from an architectural point of view, and optimal efficiency.

#### 3.1. Maxwell's theorem and Michell's trusses

The theory about structural optimization born more than una century ago, when the problem itself was focussed on truss structures configurations, and it evolved till modern era in more general approaches on continua under various targets. The earlier works about optimization are attributed to Rankine (1858), who generalized the equilibrium solution of funicular systems, and Maxwell (1870), about the minimum volume of a truss layout under given loads. Both theories are based on graphical approaches and even if they started the investigation of optimality through the time it has been lost their application, till the end of the XX century. Some of the most recent applications of the works of Rankine include method for cable systems, as they are presented in Beghini et al papers [1].

At the beginning the aim of optimal structural design was to find truss structure configurations in order to get the stiffest configurations under the least amount of material. The first work which paves the way to that path was proposed by James Clerk Maxwell [1831-1879] with *“On reciprocal figures, frames, and*



*diagrams of forces*”, published in 1870 [36], where it is provide quick method to establish the minimum volume of a frame under a given set of loads and boundary conditions.

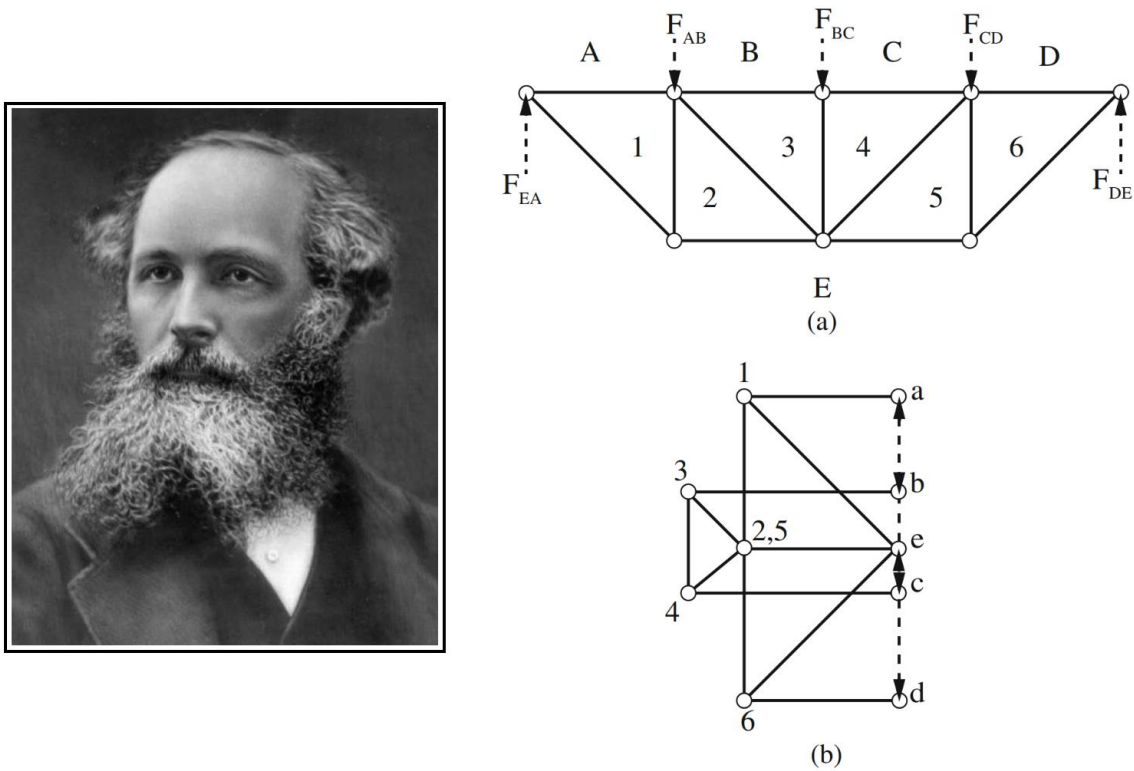


Figure 3.1 James Clerk Maxwell [1831-1879] (right), reciprocal diagrams (left) of the structure, (a), and forces (b)

As Lógó et al clearly describe [22], Maxwell in his paper proposed a theorem about the equilibrium of a series of attracting repelling centres of force and applied it to trusses, where the bars replaced the action at a distance except in the case of the external forces. Mentioning Maxwell’s words upon scientific significance of his theorem:

*“The importance of the theorem to the engineer arises from the circumstance that the strength of a piece is in general proportional to its section, so that if the strength of each piece is proportional to the stress which it has to bear, its weight will be proportional to the product of stress multiplied by the length of the piece.*

*Hence these sums of products give an estimate of the total quantity of material which must be used in sustaining tension and pressure respectively.”*

It must be noted that Maxwell uses the word “stress” for what we should term “load”.

The problem itself should be described as follows.

Taking a truss structure in equilibrium under a set of forces  $F_i$  acting at the points  $r_i$ , ( $i = 1, 2, \dots, n$ ) and dividing tension members and compression members with length, section areas and applied load respectively  $L_t, A_t, T_t$ , and  $L_c, A_c, T_c$ . Defining admissible stresses as  $f_t$  and  $f_c$ , respectively, by the principle of virtual work the optimality condition of the lightest structure is derived, and it has the volume given by:

$$V = V_c \left(1 + \frac{f_c}{f_t}\right) + \frac{1}{f_t} \sum_i F_i r_i = V_t \left(1 + \frac{f_t}{f_c}\right) - \frac{1}{f_c} \sum_i F_i r_i$$

Here  $V_t$  and  $V_c$  are the volume of all the tension and compression members respectively.

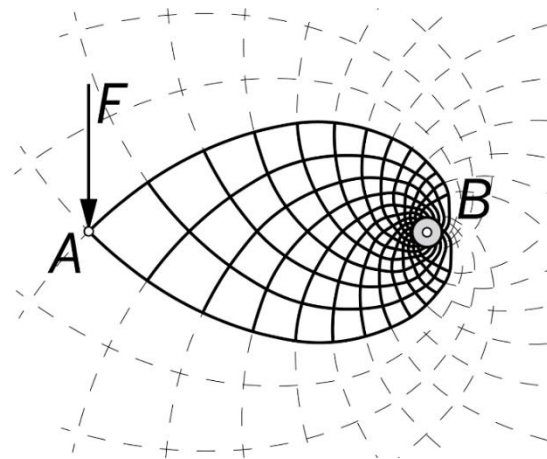
Later in 1904, Anthony George Maldon Michell [1870-1959] generalized the Maxwell’s theorem and applied it in order to establish the formulation for the optimum structural weight of a truss. Even if Maxwell has exposed clearly the need of optimal layouts, the Michell work, “*The limits of economy of material in frame-structures*” [38], is today considered the starting point of structural optimization theories.

He listed a set of conditions for a structure to be an optimum and proved the geometric restriction which determines the classes of orthogonal sets of curves along which the members of an optimum structure must lie.

The Michell problem should be described as follows.

Starting from the Maxwell's problem, referring to a set of external forces  $F_i$  acting at the points  $r_i$ , ( $i = 1, 2 \dots n$ ). Let  $D$  be a space domain which includes the points  $r_i$  and the whole of feasible space. Now, considering the all possible frameworks, trusses systems,  $S$ , contained in  $D$ , which equilibrate the forces  $F_i$  and which satisfy the stresses limiting, it is assumed that there is a framework  $S^*$  which satisfies the following condition of Michell:

*“There exists a virtual deformation of the domain  $D$  such that the strain along all members of  $S^*$  is equal to  $\pm e$ , where  $e$  is a small positive number, and where the sign agrees with the sign of the end load carried by the particular member, and further that no linear element of  $D$  has strain numerically greater than  $e$ .”*



*Figure 3.2 Anthony George Maldon Michell [1870-1959] (right), cantilever problem in a free boundary space (left)*

The statement of the theorem is that the volume  $V^*$  of  $S^*$  is less than or equal to the volume  $V$  of any of the other possible frameworks  $S$ .

$$V = \frac{f_t + f_c}{2f_t f_c} \left( \sum_t L_t T_t + \sum_c L_c T_c \right) - \frac{f_t - f_c}{2f_t f_c} \sum_i F_i r_i$$

Then, the volume  $V^*$  is derived from the principle of virtual work, where, if the virtual displacements, corresponding to Michell's statement, are  $ev_i$  at  $r_i$ , this volume  $V^*$  is:

$$V^* = \frac{f_t + f_c}{2f_t f_c} \sum_i F_i v_i - \frac{f_t - f_c}{2f_t f_c} \sum_i F_i r_i$$

The imposed deformation  $e$  imposes certain restrictions on the final layout in  $S^*$ . Citing Lógó et al paper:

“At a node of this framework the directions of the strains  $\pm e$ , which are along the lines of members of  $S^*$  and are principal directions of strain and must satisfy certain orthogonality conditions. In a three-dimensional truss, at a node with three members, there are no restrictions, if the load in members have the same sign, since in that case the virtual deformation is a pure dilatation and therefore isotropic. If one load is of opposite sign to the others, it must be at right angles to them. For a node with four members, there is again no restriction if all the loads have the same sign. If one member has an opposite load to the other three, then it must be orthogonal to them all and so forces them to lie in a plane. Finally, if the members fall into pairs with opposite-signed loads then one of these pairs must be in line and normal to the other two.”

Moving from the advantages of using the least amount of material, the most important property taking from Michell work is that when we obtain the optimal configuration, the optimal structure itself,  $S^*$ , has also the greater grade of stiffness that any other structures in the  $S$  range.

Anyway, it is noted that the original Michell formulas are not valid for different allowable stresses in tension and compression. Only in 1960 Chan [10] wrote down correctly the Michell's theorem and the validity and the critical examination of the Michell's theory can be read in Rozvany's work [48] also, almost 40 years later.

### **3.2. Evolution of optimization methods**

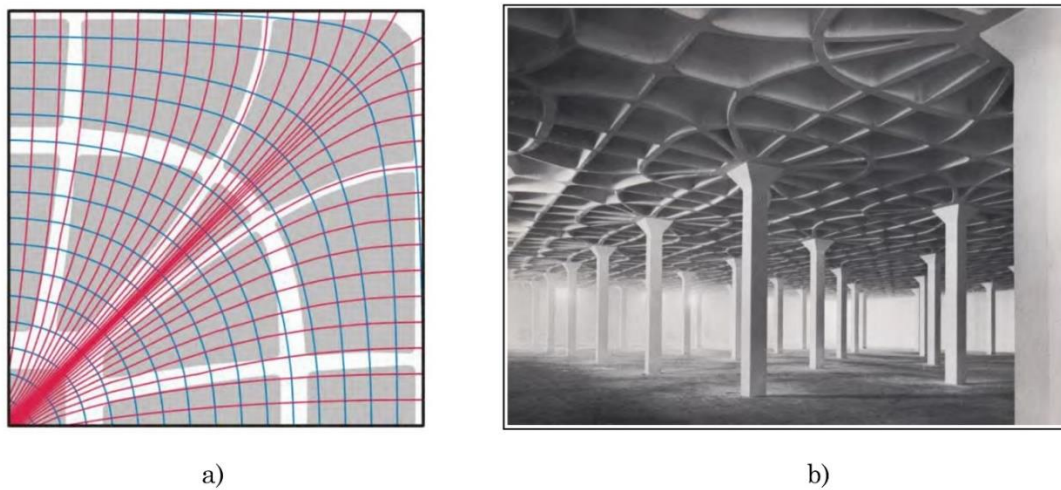
From those analytical theories applications on engineering have raised together with related problems and discussion. Firstly, in 1958, it was presented the non-uniqueness of the optimal layout, by H. L. Cox, in terms of minimum weight, since he demonstrated the existence of different configurations with the same weight. So, the optimal solutions are not unique in the case of Michell's structures. Using variational calculus Shield, later in 1973 presented some additional conditions for the stationarity of the structural volume but even with these additions it was not possible to guarantee a global optimum layout. Shield working on the Michell's structures declared that this design fails when kinematic constraints are taken into consideration, and so he chooses to propose an alternative approach which has not that limits.

The investigation about the convergence of solution and its uniqueness are of primary importance and their achievement remains more complicated if are presented multiple load cases or loading uncertainty is considered. The load can be considered as a quantity given in an interval with a certain possibility of location or/and direction, and/or magnitude, and they are selected depending on the worst effects. This topic was investigated in analytical ways by Nagtegaal and Prager in 1973.

### **3.3. Principal stresses trajectories**

Since the Cauchy tensor has been formulated, the existence of specific orientations, for every single infinitesimal element where no shear stresses exist and only axial stresses leads, was known. Those directions, *principal directions*, map a series of trajectories that define the force flow into the body from its entrance to the constraints. Studying those lines, it was noted that some natural elements follow them, for example bones pore structure in human body, and this naturally offers a guideline to use the whole material properties. There are no specific examples or a starting point of this theories, but one of the most

famous application of this approach, to obtain optimality in terms of stiffness and material economy, was the Gatti Wool Factory, in 1953, by Pier Luigi Nervi [1891-1979]. After similar works, he proposed a ribbed floor system where ribs follow principal moment directions of the slabs, around the support system made by columns. Those were isostatics lines where no torsional moment exists and intersect each other orthogonally for the hole pattern. This approach is described in detail by Halpern at all [4] where the so called “Nervi’s method” is presented in new forms.



*Figure 3.3 a) principal moments pattern for a slab modulus and b) the Gatti Wool Factory floor*

On the other hand, as it will be seen during this work, those approach with principal stresses trajectories is often used nowadays for its efficiency but also because its aesthetics.

### **3.4. Material distribution methods**

Till now it has been defined a macroscopic theory about optimization base on trusses elements, boundary conditions and loads and it allows mainly to define sizes and sections of the members itself. By the way, a microscopic approach naturally arose for shape and sizing optimizations, for and higher freedom on the form management and because the microscopic approach allows engineers

to study the relations between the microstructure itself and the macrostructure behaviour. This was for example clearly demonstrated in the paper by Cheng and Olhoff in 1981, on optimal thickness distribution for elastic plates, which started clearly in some way a marked interest in microscopic material distribution.

From the introduction of Finite Element Methods (FEM) in engineering word, optimization theories have changed their solution approach and many analytical problems, in terms of computations, have been overpassed, as Lucien Schmit, in 1960, recognized. The trend of numerical evaluation, instead of analytical one, has affirmed its position in research and design world. From that point the microstructure approach, or material distribution method, found its proper tool for numerical evaluations and experiments and enter completely in the practical word in 1988 from the first paper by Martin Philip Bendsøe et al [34], who defines a computational tool.

To complete the introduction and the continuing of the optimization analytic it is appreciable mentioning Bendsøe and Sigmund [34], from their book “*Topology optimization: theory, methods and applications*”:

*“For thin structures, that is, structures with a low fraction of available material compared to the spatial dimension of the structure, the material distribution method predicts grid- and truss-like structures. Thus, the material distribution method supplements classical analytical methods for the study of fundamental properties of grid like continua, as first treated by Michell. Applications of numerical methods to truss problems and other discrete models were first described in the early sixties but now we see that these challenging large-scale problems can be solved with specialized algorithms that use the most recent developments in mathematical programming. In its most general setting shape optimization of continuum structures should consist of a determination for every point in space if there is material in that point or not. Alternatively, for a FEM discretization every element is a potential void or structural member. In this*

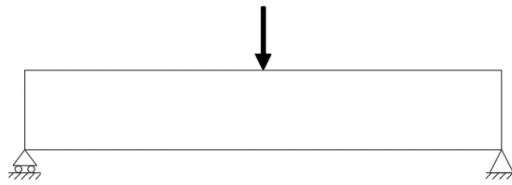
*setting the topology of the structure is not fixed a priori, and the general formulation should allow for the prediction of the layout of a structure. Similarly, the lay-out of a truss structure can be found by allowing all connections between a fixed set of nodal points as potential structural or vanishing members.”*



From the born of a such versatile approach as material distribution methods the optimization studies discovered new frontiers and more versatile solutions. Is then necessary focussing a little bit the attention on that method

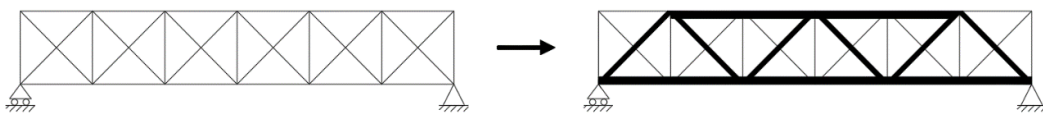
## 4. Material distribution method – Topology optimization by minimum compliance design

The three main optimization problems, which address different aspects of the structural design problem, are the sizing, shape and topology methods.



*Figure 4.1 Simply supported beam problem*

The sizing problem leads to find for example the optimal thickness distribution on a linear elastic plate or the optimal elements area in a truss structure, design variables, in order to minimize or maximize one or more physical targets, state variables, such as strain energy, maximum stress, deflection etc. The main characteristic of this approach is that the design model domain and the state variables are defined a priori and are fixed during the process.



*Figure 4.2 Sizing optimization solution for simply supported beam problem*

In the shape problem, instead of sizing, as the noun suggests the scope is the research of the optimal shape of the domain and that design space becomes a design variable.

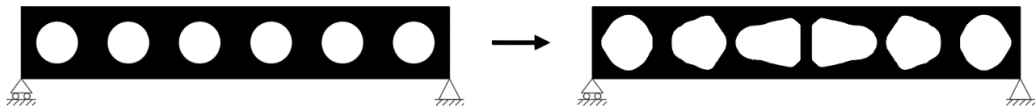


Figure 4.3 Shape optimization solution for simply supported beam problem

At the end, the topology optimization problem, which is the more versatile and general, is focussed on defining the number, shape and location of “holes” into de design domain and also its connectivity.

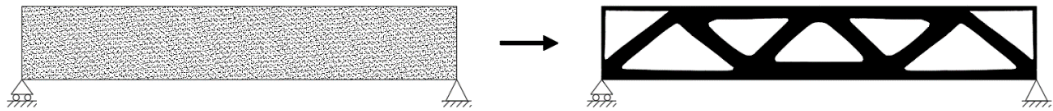


Figure 4.4 Topology optimization solution for simply supported beam problem

With this last method is possible to define the optimal layout of a structure having few information, such domain, loads and constraints, and so it represents the most versatile approach to study the nature of a mechanical system.

#### 4.1. Method description

The topology optimization problem is to determine, under a defined space region, the optimal layout, in terms of performances, standing on some known conditions, such as applied loads, supports, structure volumes and others sub requirements. So, the size, the shape and the connectivity of the final product are usually unknown or should be supposed in few cases from theories. For the purpose of this work only the simplest case of isotropic material and minimum compliance criterion is considered. With the term *compliance* we refer to the strain energy of a mechanical system, an indirectly to its displacements, so it is the inverse of the system stiffness. Indeed, minimizing the compliance of the problem is the same to maximise the stiffness of the problem and this concept will be stressed more later.

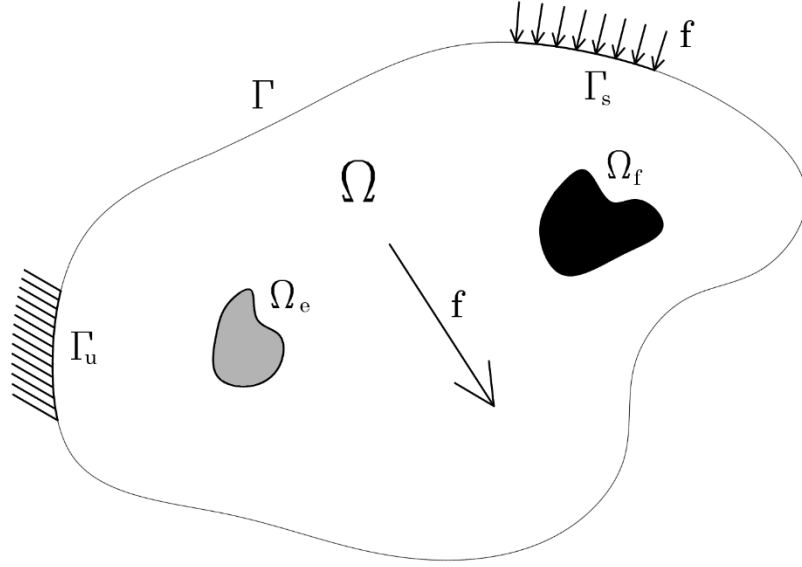


Figure 4.5 General body system

Considering a source domain  $\Omega$ , in  $\mathbf{R}^3$  or  $\mathbf{R}^2$ , of an isotropic material, in which it is contained a “target” domain  $\Omega_{mat}$ , on which volume and surface loads,  $f$ , are applied, and constraints are imposed,  $\Gamma_u$ . There should be also some void region or fixed solid region,  $\Omega_e$  and  $\Omega_f$ . The optimal design problem is defined as the investigation of the optimal stiffness tensor  $E_{ijkl}(x)$ , which can vary into the source domain. Writing the bilinear form of energy, where  $u$  is the equilibrium configuration and  $v$  the virtual displacement:

$$a(u, v) = \int_{\Omega} E_{ijkl}(x) \varepsilon_{ij}(u) \varepsilon_{kl}(v) d\Omega$$

and

$$\varepsilon_{ij}(u) = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

About the load work in linear form:

$$l(u) = \int_{\Omega} f u d\Omega + \int_{\Gamma_T} f u d\Gamma_T$$

The problem is then formulated as the investigation of the minimum strain energy or minimum compliance problem, and then it implies that the system configuration with the minimum deformation is also the stiffest in the whole possible configurations. So, working intrinsically on a maximum stiffness problem, the formulation is then written as:

$$\begin{aligned} \min_{u \in U, E} \quad & l(u) \\ \text{s. t.} \quad & a_E(u, v) = l(v) \quad \text{for all} \quad v \in U \\ & E \in E_{ad} \end{aligned}$$

Where  $U$  is the space of the kinematically admissible displacements and  $E_{ad}$  is the set of all admissible stiffness tensor for the design problem in the unknown  $\Omega_{mat}$  configuration. It must be underlining that the stiffness tensor  $E$  has zero value outside the admissible set and that the limit resource of the problem is expressed by the target volume, defines as:

$$\int_{\Omega_{mat}} 1 d\Omega \leq V = \int_{\Omega} 1 d\Omega$$

When the problem is solved by computational approach, discretizing the continuum by FE for example, the parameters previously defined are also discretized, and introducing the corresponding quantities the formulation becomes:

$$\begin{aligned} \min_{u \in U, E} \quad & f^T u \\ \text{s. t.} \quad & K(E_e)u = f \quad \text{for} \quad K = \sum_{e=1}^N K_e(E_e) \\ & E_e \in E_{ad} \end{aligned}$$

Where  $K$  is the stiffness matrix of the whole system generated by the stiffness contribution of every single element  $e = 1, \dots, N$ , which depends on the single element stiffness  $E_e$ .

Having now a discretized space in FE the next step is to determine which points of that space are material points and which are void points. For the formulated problem described above, this implies that the set of admissible stiffness tensors,  $E_{ad}$ , has the property:

$$E_{ijkl} = 1_{\Omega^{mat}} E_{ijkl}^0 \quad \text{where} \quad 1_{\Omega^{mat}} = \begin{cases} 1 & \text{if } x \in \Omega^{mat} \\ 0 & \text{if } x \in \Omega / \Omega^{mat} \end{cases}$$

$$\int_{\Omega} 1_{\Omega^{mat}} d\Omega = V_{mat} \leq V = \int_{\Omega} 1 d\Omega$$

The last relation expresses a limit on the amount of material, defined as a fraction of the initial source volume, in which the compliance must be minimized. The tensor  $E_{0ijkl}$  defines the stiffness tensor of the isotropic material and the admissible tensor under this condition results as a distributed and discrete valued design problem, or  $0-1$  problem. Anyway, to solve that problem, commonly, integer variables  $0-1$  are not suggested and then they are replaced with continuous values between the previous edges by the introduction of a factor which steers the process as much as possible to the  $0-1$  type.

The design problem is then solved modifying the stiffness matrix so that it depends continuously on a function interpreted as the density of the material, stressing the idea that the optimization is a problem of regions with and without material. For the values of the tensor that are between  $0$  and  $1$ , due to that new factor of continuity, are considered as “artificial” density values and then, in order to mitigate correctly their contribution, they are penalized in some way.

## 4.2. SIMP method

There are many methods that follow that approach, but the most popular and stable is the “*Solid Isotropic Material with Penalization*”, SIMP, or penalized proportional stiffness model, which is formulated as:

$$E_{ijkl}(x) = \rho(x)^p E_{ijkl}^0 \quad \text{where} \quad p > 1; 0 \leq \rho(x) \leq 1 \quad x \in \Omega$$

$$\int_{\Omega} \rho(x) d\Omega = V_{mat} \leq V = \int_{\Omega} 1 d\Omega$$

The “artificial density”  $\rho(x)$  is the design function, with values between 0 and 1, and  $E_{ijkl}$  represent the properties of the isotropic material as always. At the extreme of its interval,  $\rho(x)$  gives to the stiffness tensor the values:

$$E_{ijkl}(\rho = 0) = 0 \quad E_{ijkl}(\rho = 1) = E_{ijkl}^0$$

So, imaging the source space as a grid of squares, the final result is a map of white (no material), black (full material) and grey (partial material) squares, where this last typology, if the problem converges perfectly, disappear from the system.

The penalty factor  $p$  is choosing greater than 1 and mathematically speaking, for intermediate values of density, it gives for those elements an unfavourable value of stiffness compared to the full material elements. So, the choice of the penalty factor determines directly the convergence and quality of the final layout solution.

Being this a computational model, the need to find a physical meaning of its result naturally arise especially when the solution doesn't converge clearly and there are some “grey” regions that must be interpreted. To guarantee a physical

material model meaning to SIMP, there have be found limits values for penalty factors, for 2D and 3D problems:

$$p \geq \max \left\{ \frac{2}{1 - \nu^0}; \frac{4}{1 + \nu^0} \right\} \quad 2D \text{ models}$$

$$p \geq \max \left\{ 15 \frac{1 - \nu^0}{7 - 5\nu^0}; \frac{3}{2} \frac{1 - \nu^0}{1 - 2\nu^0} \right\} \quad 3D \text{ models}$$

Where  $\nu^0$  is the Poisson's ration. In the most general cases the limit value to obtain a reasonable result is for 2D cases "3" and for the 3D cases "2" but is a convention for both cases taking a penalty factor of "3" at least.

Working with FEM analysis, to complete the explanation the FE form of the final problem is given:

$$\min_{u, \rho_e} f^T u$$

$$s. t. \quad \left( \sum_{e=1}^N \rho_e^p K_e \right) u = f$$

$$\sum_{e=1}^N v_e \rho_e \leq V, \quad 0 < \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N$$

In order to avoid singularities is commonly introduced a lower bound value for the density,  $\rho_{min}$ , at  $10^{-3}$ .

### 4.3. Solving procedure

To solve the problem, an iterative procedure is proposed, where for each previously computed design and its associated displacements it update the



design variables at each point or FE, independently from any other updated at other points or FE, based on the necessary conditions of optimality.

For a general overview of the process, the topology design on material distribution method with an isotropic material the computational steps are divided in three groups:

### *Pre-processing*

Chose reference domain, load, boundary conditions, region of the process, volume fraction target, sufficiently fine meshing.

### *Optimization*

Basing on displacement finite element analysis, computing the first analysis on the uniformly distributed isotropic material the minimum compliance is solved a new configuration is obtained. The process continues till a marginal improvement target.

### *Post processing*

Interpretations of material distribution results and representation.

The reason why the displacement analysis of each element is considered is because implicitly through the equilibrium equations and relative derivatives they are computed in respect to the design variables of the problem in terms of sensitivity analysis.

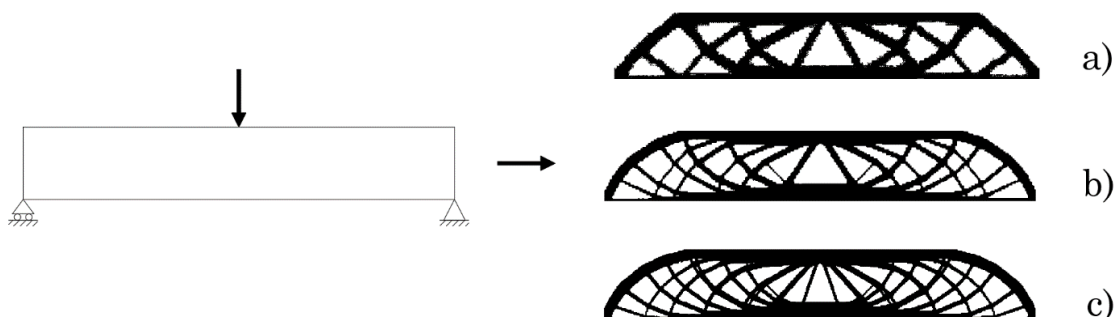
For more information about methods, programming and construction details of the methods it is suggested referring to the literature book by M. P. Bendsøe and O. Sigmund, “*Topology optimization – Theory, methods and applications*” [34].

In addition, all the following 2D topology optimization pictures have been obtained by a MatLab code given by O. Sigmund, [40], typically used for academic purposes.

#### 4.4. Mesh dependency

Some important aspects on results quality have been detected during the years focussing on some dependency of the method. Firstly, this approach depends strictly in the problem modelling, in particular to its load applications and meshing. The first influences the results, as it will be show later, mostly for symmetrical requirement output or trace of the load application on the final body. The second point, instead, is an important aspect for results purposes.

The mesh dependence of the solution is strictly related to the capacity action of the process during the analysis in such small or not part of the body and it often influences also the existence of solutions. Refining the mesh, the process can operate in a more microstructure way, ideally reached if the mesh is infinitely small, and the holes created in the domain are not large enough but subdivided in many smaller holes creating a fine curtain of elements. The reason of this choice by the process is that with the same values of volume fraction, a structure with a large number of holes, that define a kind of grid, is stiffer that one with a single hole with the same volume dimensions.



*Figure 4.6 Mesh dependency of the optimization problem for a simply supported beam, from a) to c) the number of finite elements increases having the same amount of volume*

So, increasing the mesh refinement a more detailed structure is achieved. For some purpose, this property is not desirable, like in many mechanical and aerospace projects, where producing an object with such detailed parts is impossible and inconvenient; in this case this feature is covered by filtering actions on the space of density changing, in order to manage the process in a simplest result and less detailed. On the other hand, for large scale systems, a more detailed solution is desirable, because a simpler solution like case a) in leads too much big elements, non-feasible with realizations.

#### 4.5. Analogy between different volume fractions cases

It must be observed now how the body changes his configuration depending on volume fraction and, in reason of that, how a simpler truss configuration is just the “less material layout” of a finer and more detailed configuration given by a “more material layout”.

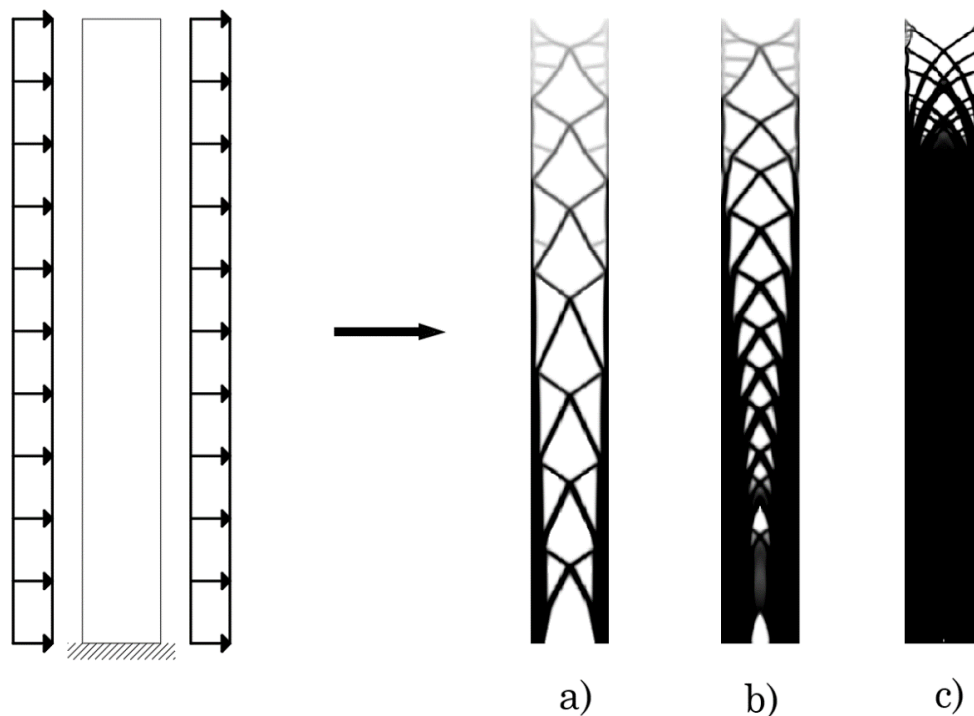
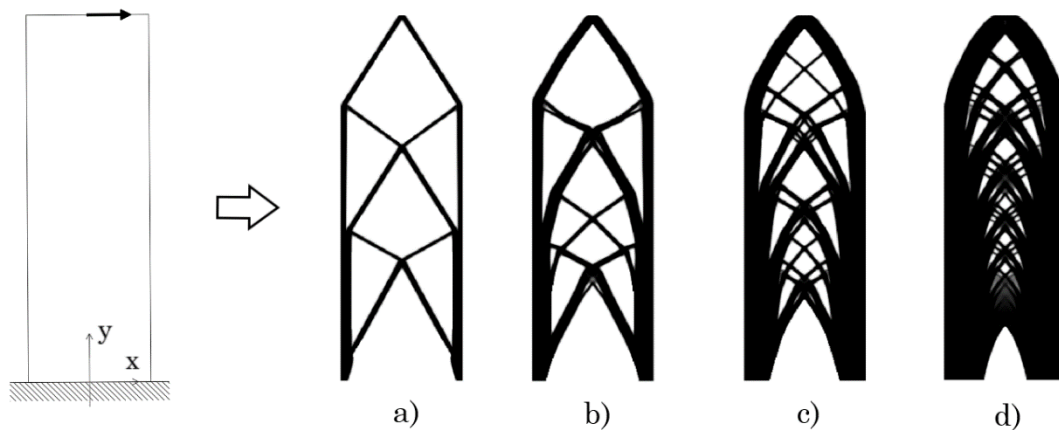


Figure 4.7 Comparison between layouts with different amount of volume, case a) 30%, case b) 60% and case c) 90%

It has been noted that decreasing the volume fraction, a simple truss with a minimum number of cells is obtained, but increasing the volume fraction the number of cells tends to grow. From a fraction of 0.3 to 0.6, considering the example in the figure, the number of cells quite duplicates, and the elements become thicker, but passing to a higher fraction like 0.9, the truss part remains more detailed. In general, also considering the case presented in figure, a more complex layout like case c) in both figures, is the detailed version of a simpler one made by a lower fraction of volume.

The previous example is affine to the structure quant will be analysed later, but to give a much clear explanation of this analogy between a truss tube and a fine diagrid let consider a stocky cantilever, so that the topology gives clearer diagrids:



*Figure 4.8 Cantilever problem, with topology results of volume fraction 20% a), 40% b), 60% c) and 80% d)*

The simple layout obtained for a fractal around 20% of the total volume is just a “primitive form” of a finer configuration that emerges for larger amount of material.

## 4.6. Multi-load case implementation

A general overview of the topology optimization problem has been given, then some more specific case study should be considered. In order to find the optimal material distribution of a robust structure, more than one load case must be considered, and the design criterion of the topology process must be modified.

The single load case of minimum compliance problem should be easily extended to a multiloading problem, where a minimization of the weighted average of the compliances of each load is computed. The continuum formulation then becomes:

$$\min_{u \in U, E} \sum_{k=1}^M w^k l^k(u^k)$$

$$s. t. \quad a_E(u^k, v) = l^k(v) \quad \text{for all } v \in U, k = 1, \dots, M$$

$$E \in E_{ad}$$

Where the index  $k$  refers to the  $k$ -th load case on the set of  $M$  load cases. The factor  $w$  is the weight. Again, for more details about the programming phase and mathematical procedures it is suggested referring to the previous mentioned test book by M. P. Bendsøe and O. Sigmund.

During those years, anyway, other methods have been implemented to catch multi-loads results, in particular to reduce the amount of computations. The most popular techniques are proposed by Kai A. James at all [26], with the dynamic aggregation process, and by Zouhour Jaouadi at all [62], by the Epsilon method. This last in particular reduce the amount of computations because bases the process considering the minimization of the load configuration with the maximum compliance value.

The following parts aim to describe an approach to determine an optimal and robust diagrid layout for high-rise buildings in order to bear the lateral loads mostly given by wind loads. To lighten the procedures an easy and simply tall building is used, but the considerations should evolve for more complicated cases. First, a high-rise buildings analysis description will be proposed.

## 5. High-rise buildings mechanics

Before starting with the design explanations, it is necessary to understand the overall mechanical behaviour of high-rise buildings.

### 5.1. Problem introduction

The problem of high-rise building design is mostly a cantilever beam problem, which is fixed at the base, and due to this fact, the system is statically determinated the main laws governing the problem are simple. So, the state of internal stresses is easily computed once the load set is defined.

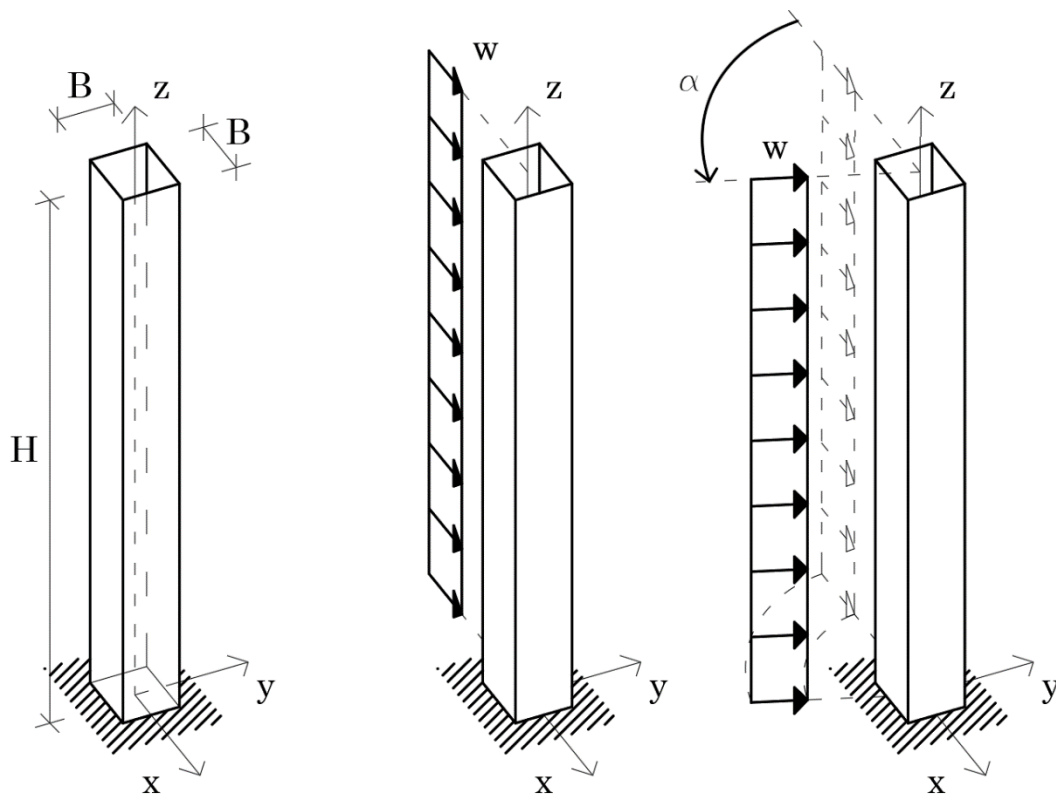


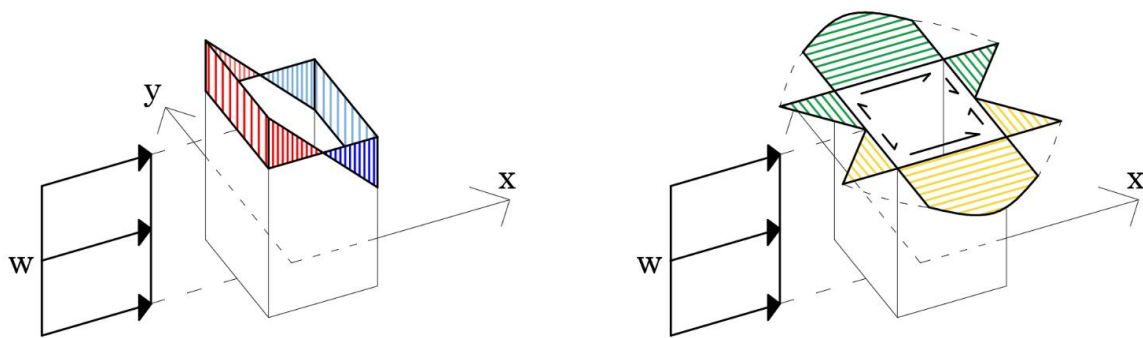
Figure 5.1 3D square section cantilever problem under uniform distributed transversal load with varying direction

The structure under analysis is a tube, height  $H$ , with square section of side  $B$  and a slenderness ratio,  $B:H$ , of  $1:8$ .

By the goal of this work, the problem becomes much simpler because is focussed on lateral load systems of the building and the transverse load analysis determine the overall response. The main lateral load considered is the wind action, which is also the variable parameter that determines the robustness problem due to its direction uncertainty. Knowing from classical literature that the wind load distribution depends in general on the wind velocity, roughness of the space around the building and transverse and tangential effect, the load is simplified as a shard of uniform distributed load enough to catch in a good way the overall behaviour.

As explained in the first part, the principal stresses trajectories are a good guide line to understand the behaviour of a structure and implicitly they suggest the material positions for maximum stiffness and minimum weight. Then, to conclude, principal stresses trajectories are taken under analysis.

As remarked, from classical structural mechanical analysis, having a thin closed section the only stresses considered are the normal stresses along  $z$  and shear stress flux for  $xz$  and  $yx$  directions, then a representation of their trend for an orthogonal wind load is proposed:



*Figure 5.2 Section stress state for 3D square section cantilever problem under orthogonal uniform distributed transversal load, normal stress  $\sigma_z$  (right) and shear stress  $\tau_{xz}$  and  $\tau_{yz}$  (left)*

Those results come from the Euler-Bernoulli beam theory which is us based on two main hypotheses:



- The cross section remains orthogonal to the beam axis during the deformation process, so there is no shear deformation.
- The cross section remains plane, so there is no warping.

In tall buildings those assumptions are not enough to describe the structure behaviour, because shear deformations and shear-lag effects occur and have a marked impact on the stress distribution

For a matter of simplicity, since the overall behaviour of the structure doesn't change with shear-lag effects, the stresses have been computed in this chapter explanation basing on Euler-Bernoulli beam theory and the case with shear-lag effect is considered later on.

The steps that follow this part begin with a 2D problem, where the building is considered as a rectangle with a transversal distributed load on one long side, and the in-plane behaviour is observed. Then, a general 3D problem is considered, and the overall deduction are extrapolated.

## **5.2. Principal stresses trajectories**

Starting from a 2D example, there is a rectangle plate with base  $B$  and height  $H$  of aspect ratio  $B:H$  equal to 1:8, so in the typical range of slender structure. The thickness of the plate is unitary, and the wind load is uniform distributed load is applied on one of the long sides of the plate. By simple static it is possible to compute the internal actions of the structure, moment and shear, with respect to the height coordinate, accordingly to the Euler-Bernoulli beam theory.

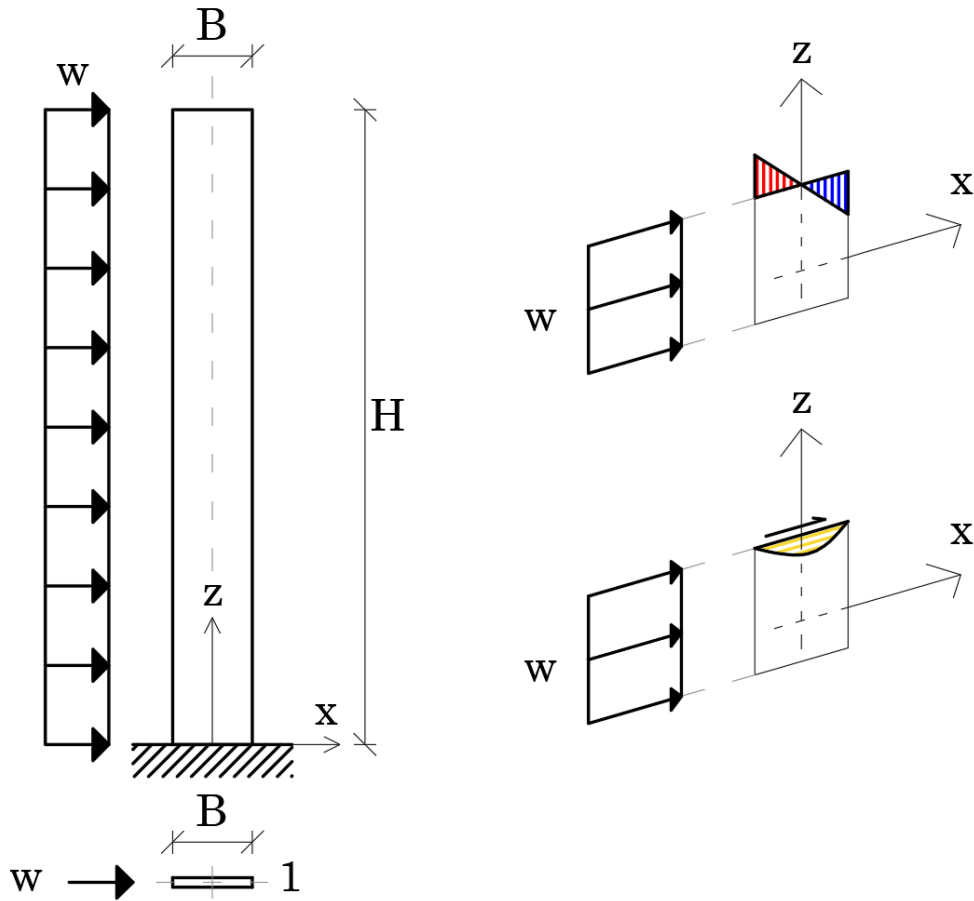


Figure 5.3 2D cantilever problem under in-plane uniform distributed transversal load (right) and section stress state (left) with normal stresses  $\sigma_z$  (top) and shear stresses  $\tau_{xz}$  (bottom)

$$M(z) = \frac{wH^2}{2} \left(1 - \frac{x}{H}\right)^2 \quad T(z) = wH \left(1 - \frac{x}{H}\right)$$

Due to the sharp difference between the thickness and the width the only relevant computed stresses are the normal stress along the vertical direction,  $\sigma_z$ , and the shear stress in the width direction,  $\tau_{zx}$ , which come from respectively flexural action and shear action. The formulas to compute them is given by Saint Venant's theory as literature often presents for a variety of cross sections.

As it has been observed before, observing principal stresses trajectories permits to understand how the structure behaves internally to transport its actions to the ground, then, rotate the system point by point in the principal plane allows to determine those paths. By the theory to search a direction where shear

disappear and normal stresses only remain, through the Mohr's circle, the principal angle for each point, with respect to vertical direction, is given by:

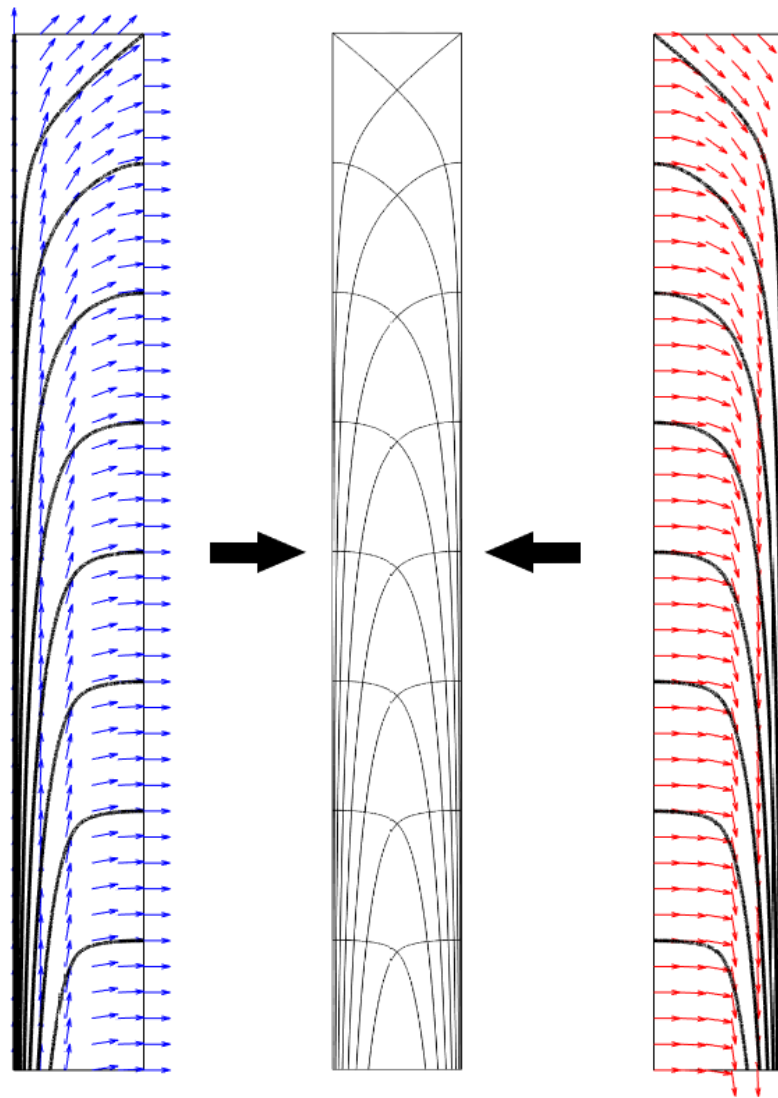
$$\tan(2\theta) = \frac{2\tau_{zx}}{\sigma_z} = \frac{2 \tan(\theta)}{1 - \tan^2(\theta)} = A(z, x)$$

Solving the second order equation with respect to the tangent of the angle it is possible to define an equation that draws the trajectory point by point of principal stresses:

$$\tan(\theta) = \frac{dx}{dz} = -\frac{1}{A} \pm \sqrt{\frac{1}{A^2} + 1}$$

The solution of the above equation leads to a set of two characteristic lines along which there is no shear and normal stresses only exist. To represent those curves a finite difference method resolution has been used, like L. Stomberg et al [28] showed, and the obtained curves should be divided into two trend groups, compression and tension. More details about principal stresses trajectories representation should be found in the appendix.

The philosophy of this diagram is to explain and show how the forces in a system "enter" into the body and move through it till the foundation, and since this is a natural force path it represents an analytical method to identify the optimal material layout. The efficiency of that approach is also explained in many literatures works [1].



*Figure 5.4 Principal stresses trajectories for 2D cantilever problem under transversal uniform distributed load, tension trajectories (right) and compression trajectories (left)*

From those graphs emerge come typical characteristic high-rise building behaviour, which are:

- Tension and compression lines meet at  $45^\circ$  close to the top. This is due to the dominant shear force at the top comparing with the bending effect, which generates the normal stresses on the section.
- The verticality of the path at the base is due to the dominant effect of the moment on the shear. For the 2D case, the lines that moves to the opposite edges become in this point vertically orientated, because the shear

becomes zero and the force move directly to the constraint at the base. In the 3D model is not exactly like that, because the shear flow diagram guarantees generally, depending on the load orientation, a non-zero shear value at the corners. Another physical justification about the increase of line density to the low-edges side, is that since the best way to sustain the overturning moment is to put material far away as much a possible from the neutral axis.

- Each group line meets the other group line at 90°, as principal stresses theory suggested.

It has been shown how in nature “diagrids” spontaneously rise from mechanical necessity of systems, and how in nature, as literature largely explains, shapes are just graph representations of forces flows.

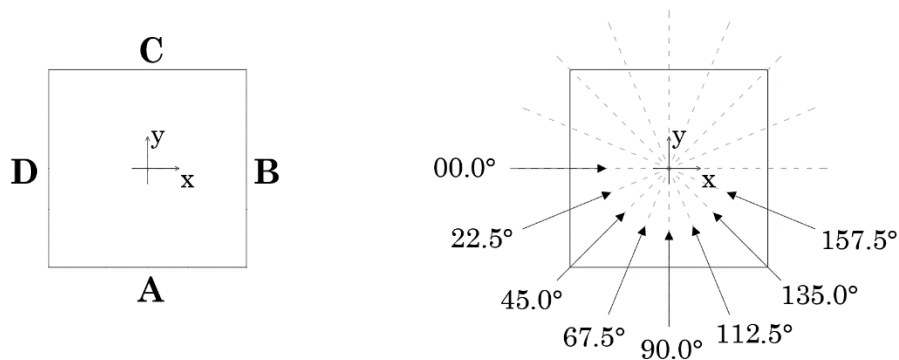
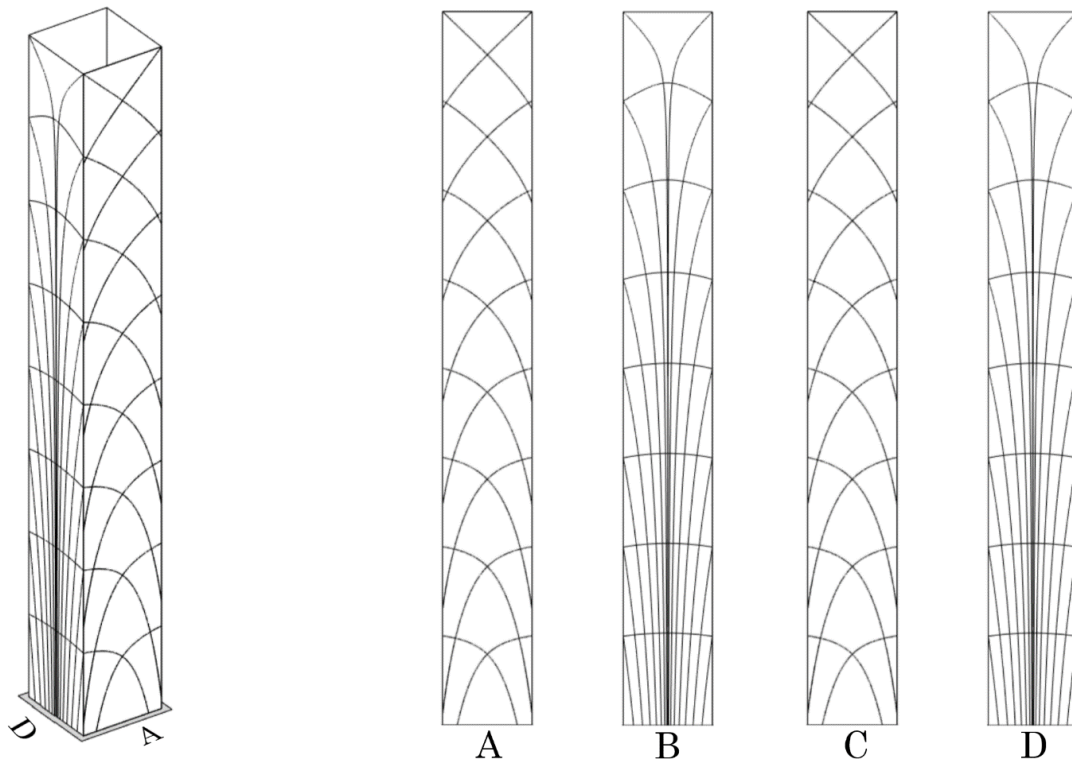


Figure 5.5 Cross-section faces names (right) and impact angles (left)

Moving in a 3D example, having a square tube cross section, the same considerations persist and considering the same load case, where the wind blows orthogonally to a face, the shear flow smooths the line trend especially close to the corners. The facades have, two by two, distinctly role in the resisting capacity of the building. The faces parallel to the wind, web elements, flow as before, carrying mainly the shear effects through its “natural diagrid”, while the orthogonal faces, flange elements, have mainly vertical lines, because their normal actions due to bending moment are prevalent. In those last faces, the smooth curvature close to the lateral edges, in contrast with the pure vertical

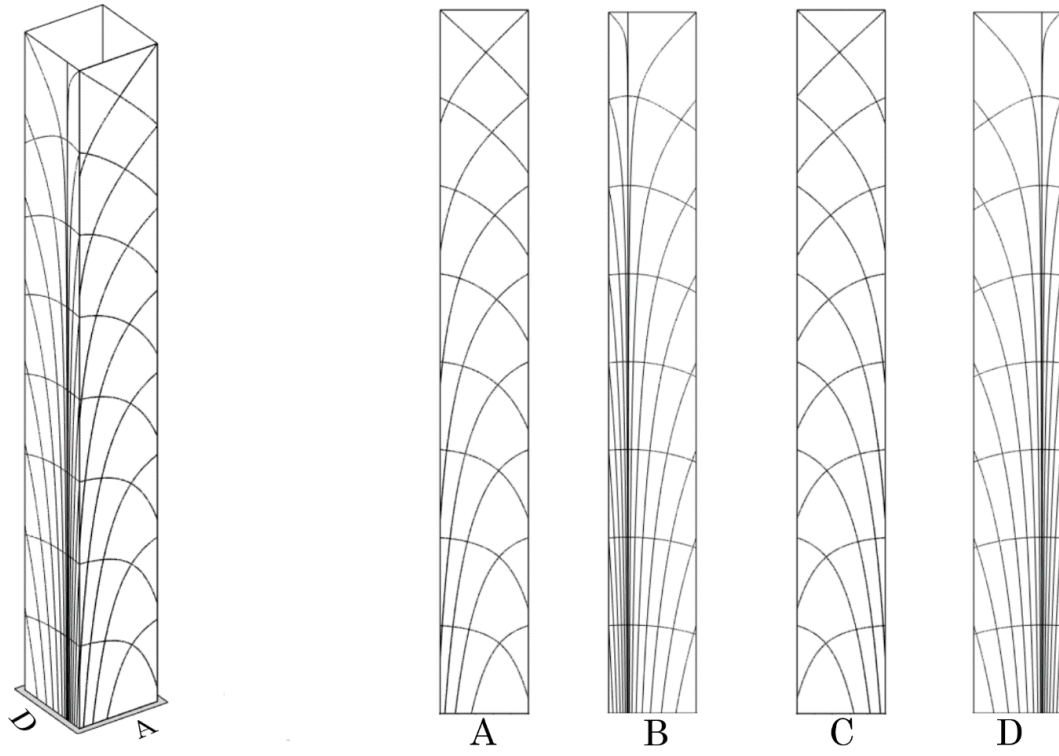
direction at the centre line, is due to the shear flow that moves from the force impact region at the middle to the edges. The following pictures gives an idea of the force path into the structure, in tension and compression meaning.



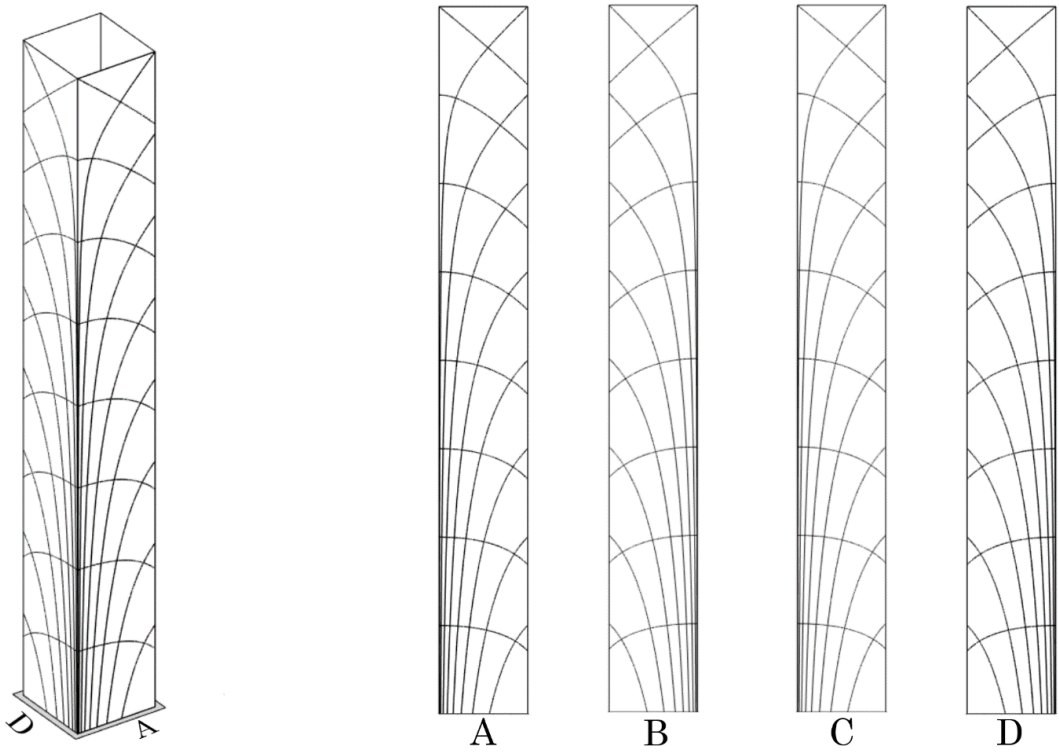
*Figure 5.6 Principal stress trajectories for 3D cantilever problem at impact angle  $0^\circ$*

It is necessary now to observe how this optimal natural layout changes when the force entrances region and force direction changes. The previous line behaviors, which are divided by mechanical reason, are now linearly combined giving new layouts.

To better understand that force flow behaviour other wind load action angles have been adopted,  $22.5^\circ$ ,  $45^\circ$ ,  $67.5^\circ$  and  $90^\circ$ , and due to the double symmetry of the cross section the other impact angle responses should be deduced by analogy. The principal stresses trajectories are listed as in the previous angles list:



*Figure 5.7 Principal stress trajectories for 3D cantilever problem at impact angle  $22.5^\circ$*



*Figure 5.8 Principal stress trajectories for 3D cantilever problem at impact angle  $45^\circ$*

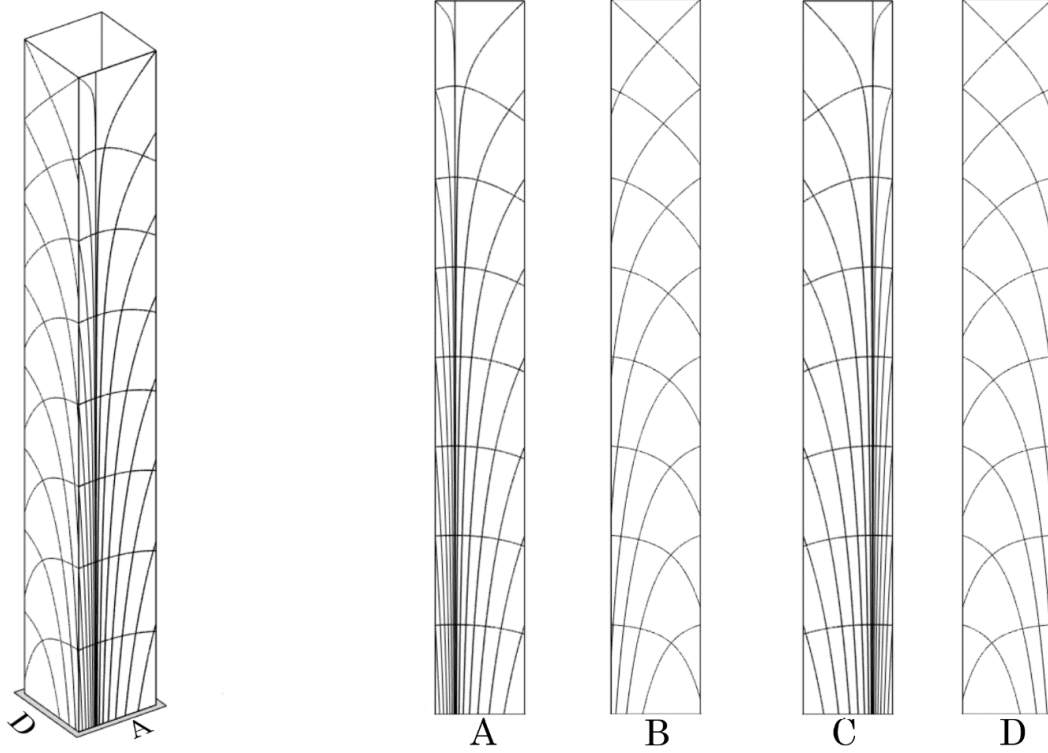


Figure 5.9 Principal stress trajectories for 3D cantilever problem at impact angle  $67.5^\circ$

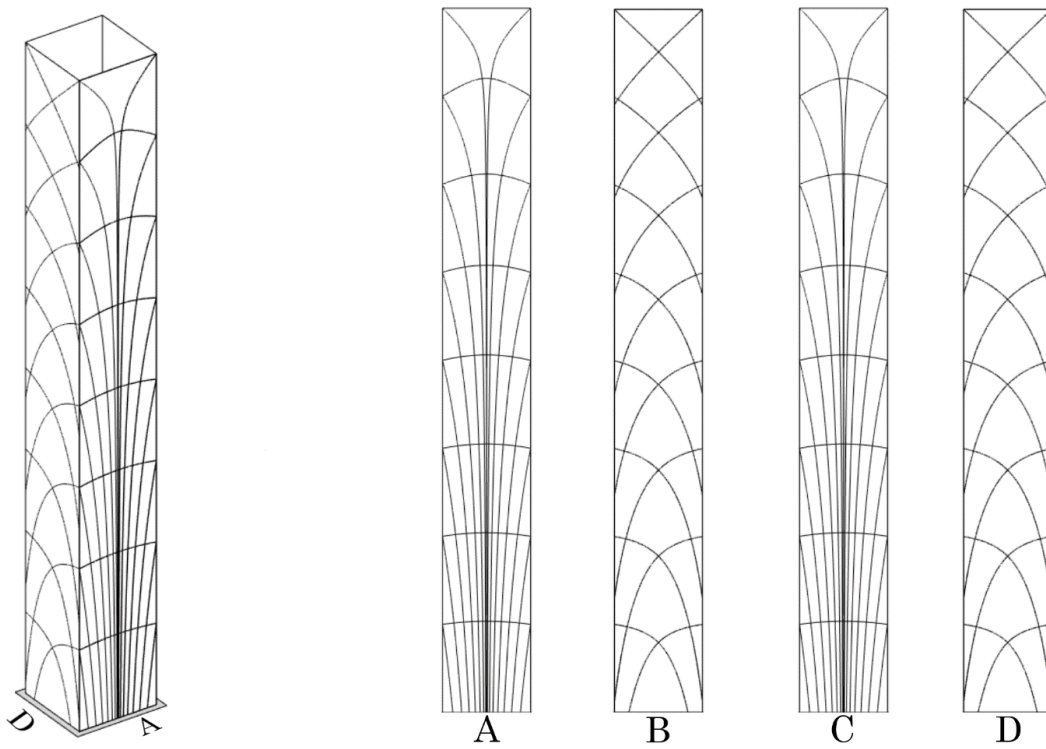


Figure 5.10 Principal stress trajectories for 3D cantilever problem at impact angle  $90^\circ$



As should be observed, since the  $0^\circ$  case, there is a strip in the façade where the force “enters” somehow into the structure and propagate its stresses like vanes into the body. Just moving this entry point the same scheme rotates around the surface maintaining the same topology. This behaviour is even more clear if we observe the same load case to a cantilever tube of circular cross section.

The curves for the given problem have been drawn by a MatLab code, which is programmed for a cantilever tube with rectangular cross section under a transversal distributed load. The code and its features are presented in the appendix at the end.

### **5.3. The shear-lag effect on stress distribution and principal trajectories**

. In our case, having a thin closed cross section, the shear-lag effect should play a relevant role and the hypothesis where sections remain plane after deformation is far from the real behaviour of this kind of structure. Following the theories of Eric Reissner [46] in his paper, the real normal stress trend, particularly into flanges, is not linear or constant, but follows a quadratic behaviour, focussing higher amount of stresses into corners than in the centre of the side. Having the same moment in the section, but with a different stress redistribution the corners elements suffer a higher effect given by loads, and this phenomenon it was called *shear-lag*.

Moreover, it was observed by Foutch and Chang in 1982 [12] an anomaly in the shear-lag redistribution also, for box cantilever beam, because after the fourth quarter of the beam, the previous stress trend inverted its behaviour. This last phenomenon is then called *negative shear-lag*.

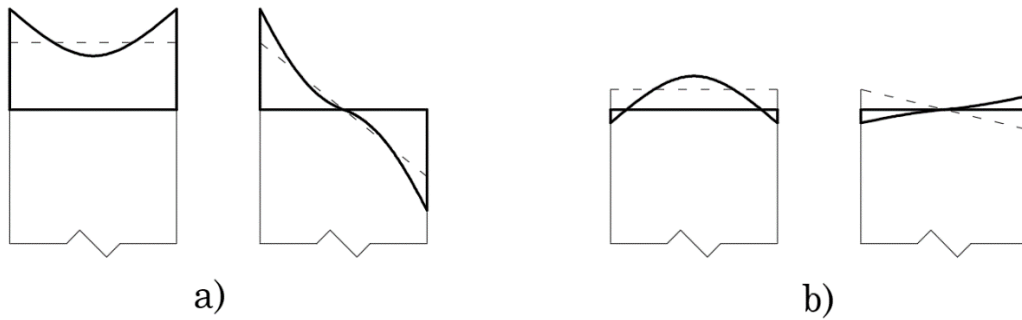


Figure 5.11 Bold lines represent shear-lag distribution and the dotted line the Eulerian stresses distribution. The case a) is for sections close to the clamped edge, so in typical shear-lag, while the case b) is close to the free edge, so in negative shear-lag

Then, the overall stresses distribution in a general cross-section becomes:

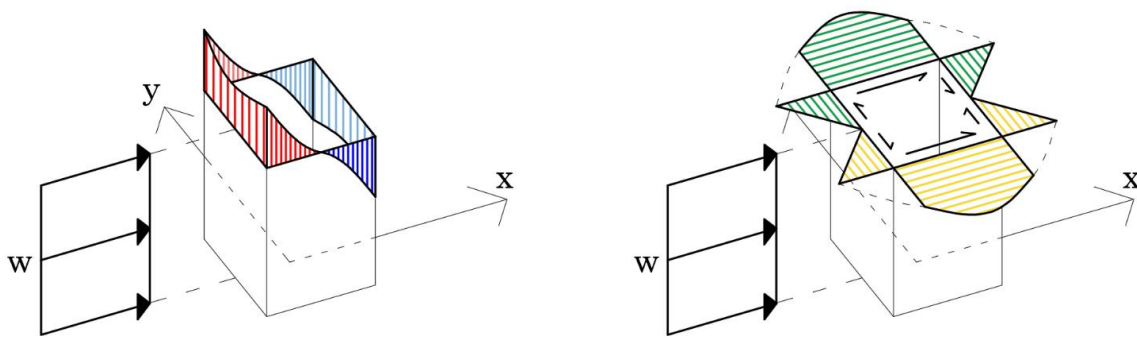
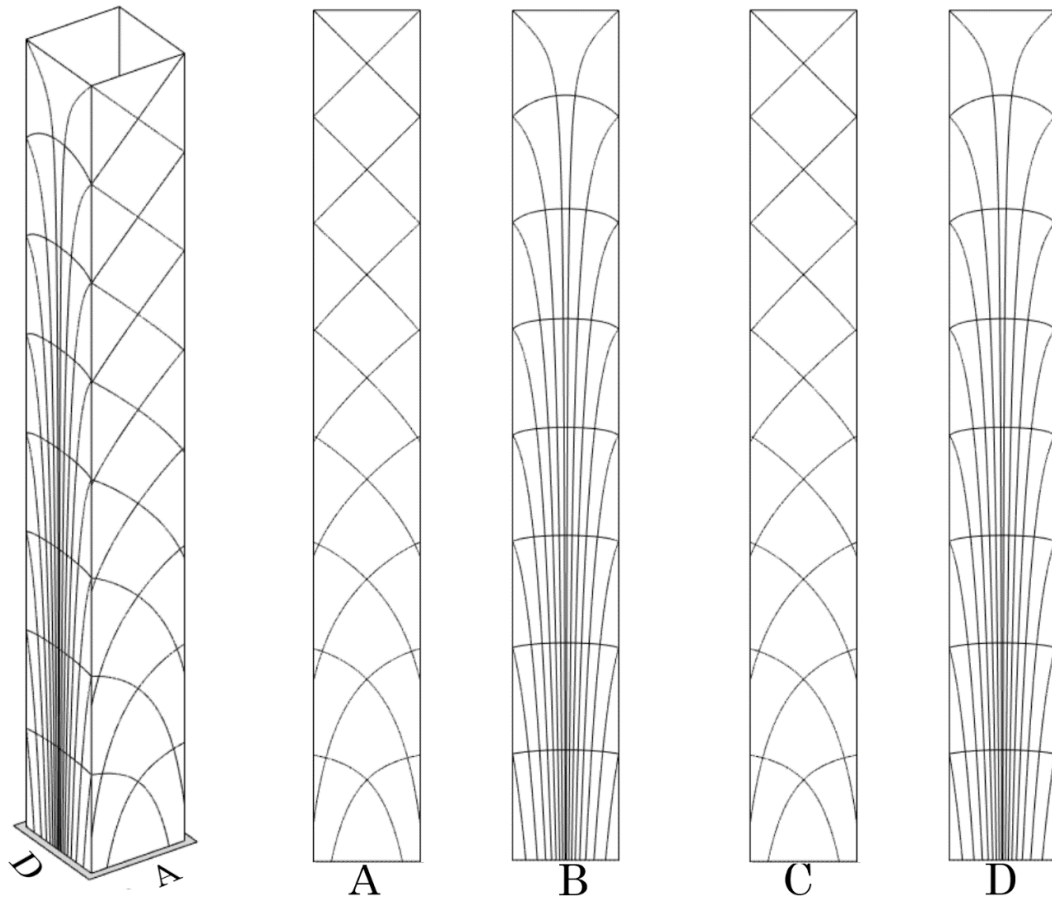


Figure 5.12 Section stress state for 3D square section cantilever problem under orthogonal uniform distributed transversal load, normal stress  $\sigma_z$  with shear-lag effect (right) and shear stress  $\tau_{xz}$  and  $\tau_{yz}$  (left) which remain invariant

For more information on this effect is suggested to refer to the Reissner papers [13] and Fouth at all [12], or to check the appendix.

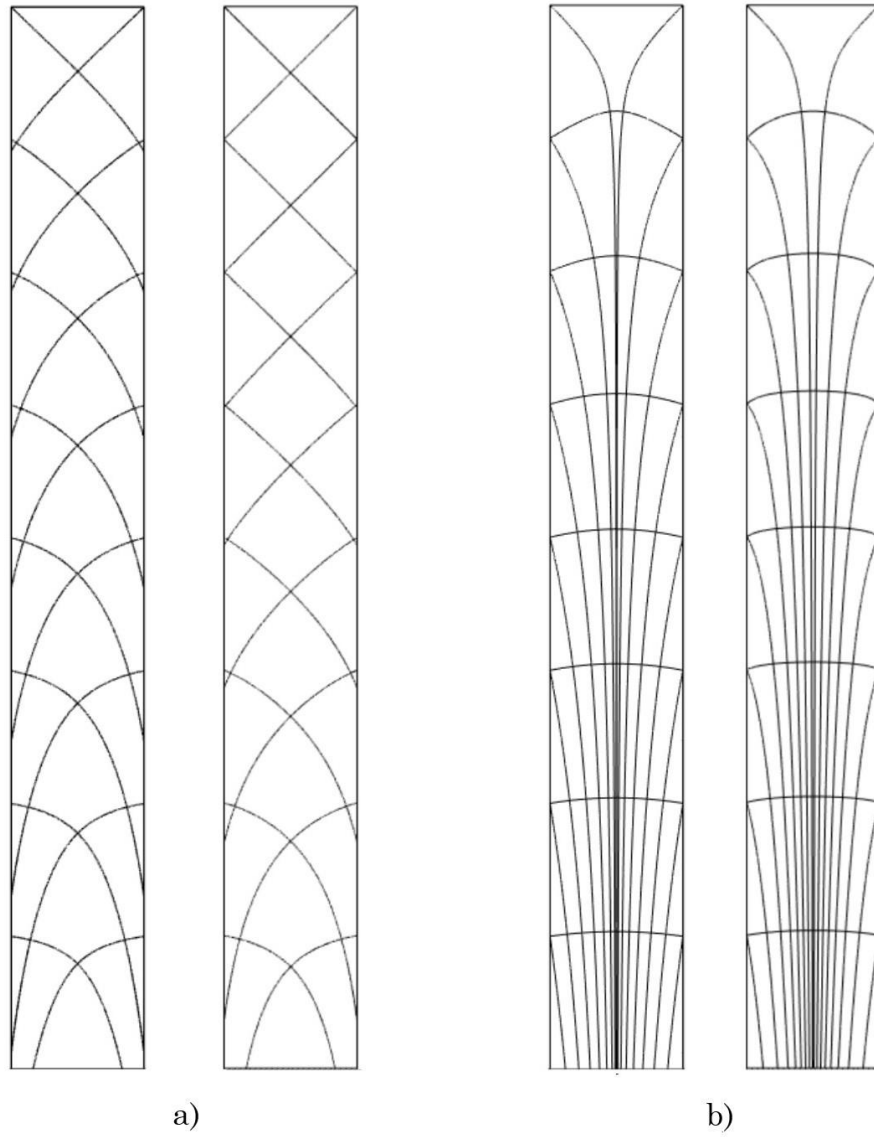
This different stress distribution affects the principal stresses trajectories and a new set of curves is then draw basing on previous approach, and later compared with the set without shear-lag. For this comparison only the load case indicated with  $0^\circ$  impact angle is taken into account.



*Figure 5.13 Principal stresses trajectories for 3D cantilever problem at impact angle  $0^\circ$  and shear-lag effect*

The most evident aspect is the dominant behaviour of the shear when the normal stresses take this new trend, in particular, the  $45^\circ$  layout becomes dominant quite immediate in the second half of the cantilever.

More considerations on shear-lag and its layout will be given in the last chapter with some numerical example and in the appendix for the theoretical demonstration.



*Figure 5.14 Principal stresses trajectories for impact angle  $0^\circ$ . Case a) webs for no shear-lag (left) and shear-lag (right), case b) flanges for no shear-lag (left) and shear-lag (right)*

Now that we have an overall view of the problem, a comparison between optimization methods to find the optimal layout is presented and later an approach for the multiload case is implemented

## **6. Connections between principal stresses trajectories, Michell's trusses and topology optimization**

The search for the perfect optimal layout is the common goal for all previous methods, which stand behind specific and distinct philosophies. Principal stresses trajectories light up to material orientations where the force flow passes through, Michell trusses found a layout of minimum volume under the condition of orthogonality between elements and topology optimization define a material volume target layout which has the minimum compliance.

Anyway, tending to the same goal of finding an optimal structure under given conditions, it should be noted that the result is one and only one which is presented in different forms. Globally observing those approach, it has been remarked that all of them bring the designer to the same solution, or to deducing it, because in topology optimization approach, in particular, the result must be interpreted.

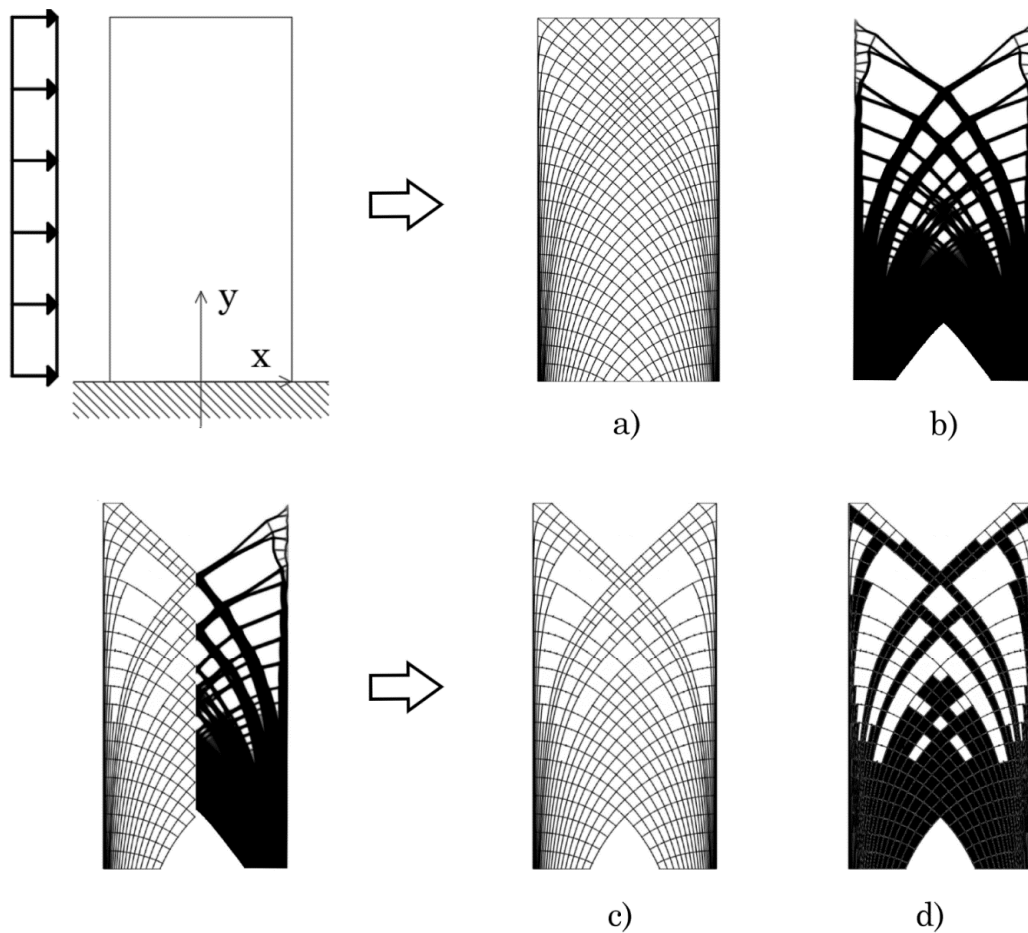
Firstly, a discussion between principal trajectories and topology will be presented, because it is more relevant for the later expositions, and secondly a comparison of the two previous approach with the Michell truss will be done. The comparison with Michell approach is just a clarification of the philosophy that the optimal truss is one and only one into a given system.

### **6.1. Principal stresses trajectories and topology optimization**

To show the analogy between principal stresses trajectories and topology optimization, it has been considered a cantilever beam under a transversal uniform distributed load, with slender ratio 1:2.

As it has been said previously, topology optimization methods under minimum compliance design tend to choose the stiffest layout, and from a practical point of view this matches perfectly with a configuration that offers the minor strain energy, similarly to a configuration with only compression and tension members.

So, it is quite immediate to suppose that the topology optimization will tend to follow the principal stresses trajectories because they already highlight an element orientation without flexural effects but with only axial action.



*Figure 6.1 Comparison between principal stresses trajectories layout, a), and topology optimization result, b). Adjusting principal trajectories, c), a similar layout is obtained, d).*

The topology result has been obtained by the MatLab code given by Sigmund [40] and the principal stresses trajectories by a code similar to the one shown in

the appendix. The configurations are based on the same philosophy, and the example should be much more clear if a finer set of curves is considered. It is clear that the stiffest truss system proposed by the topology problem is the same of principal stresses trajectories, just adjusting it by a finite number of elements. The fact that topology model doesn't suggest curved elements like the trajectories is due to a strain energy reason, because the curved beam under axial actions suffers bending effects, and it has a major strain contribute due to that part. So, following the minimum compliance principle, the straight element is more efficient in general, but, as it will be observed later in practical examples, a target value of curvature for those elements should be defined in order to accept them in the design.

## 6.2. Topology optimization model and Michell truss

Still referring to 2D examples, a different boundary condition is taken now, referring to literature and documentations already developed. Considering a beam simply supported at the opposite corners of one edge, as it was already drawn by Chan [10] in his paper, and under a point load in transversal direction at the central point of the opposite edge, the system is presented.

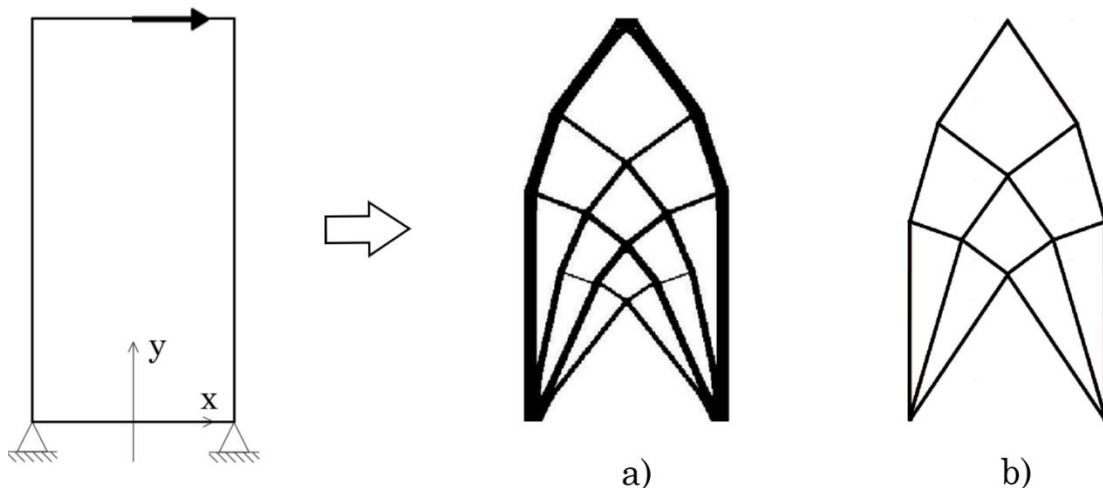


Figure 6.2 Comparison between topology optimization, a), and Michell's truss b)

This example is proposed in the work by Ole Sigmund et al [41].



By the way this analogy is also presented in literature starting from Bendsoe and Sigmund book, stressing the fact that topology method and Michell approach tend to the same stiffest truss layout.

Proposing the Michell beam problem, seen in the introduction chapters, the similarity between topology results and Michell truss is even more clear:

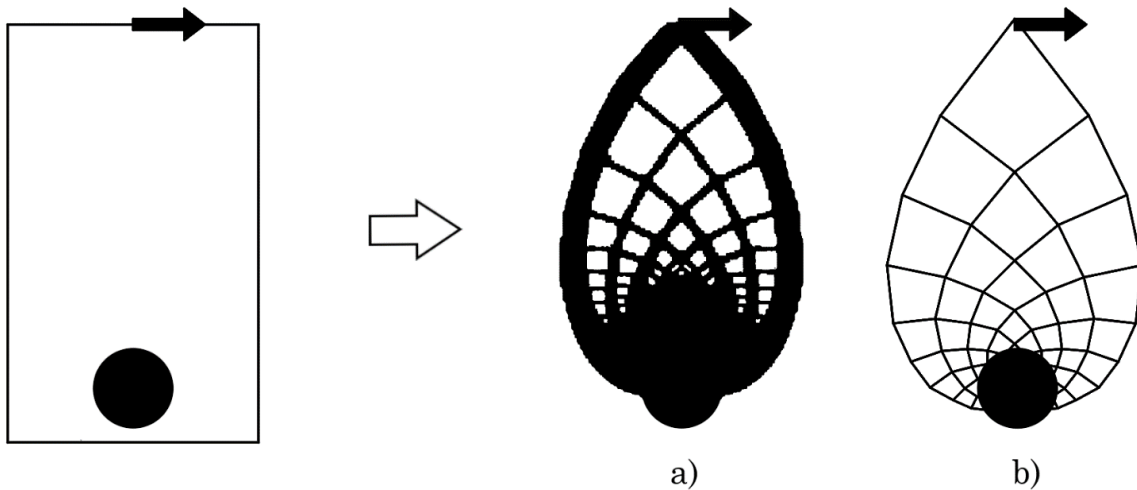
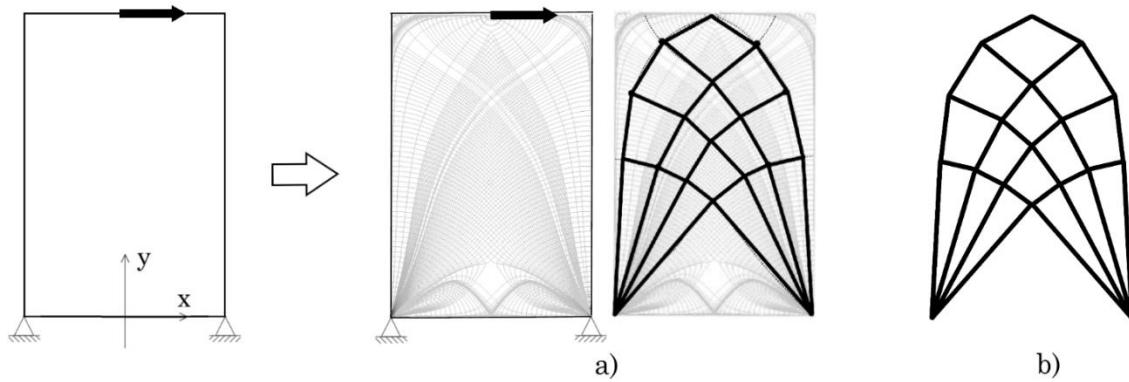


Figure 6.3 Comparison between topology optimization, a), and Michell's truss b) from Bendsoe and Sigmund book

### 6.3. Michell truss and principal stresses trajectories

Even considering the previous two cases of principal stresses trajectories and Michell's truss, which differ each other from the constraint condition, the analogy between them is evident, like at the base the verticality of the material disposition to bear the overturning moment or the orthogonality between elements. A clearer comparison is presented in Y. Li and Y. Chen paper [60], where a truss system is obtained for the two supports case following the principal stresses trajectories and emerges that its configuration is the same proposed by Michell.



*Figure 6.4 Comparison between Principal stresses, a), and Michell's truss b) from obtained connecting intersection nodes between trajectories*

The Michell's system should be even more detailed is more principal trajectories intersections being considered, but for more detail about this method it is suggested to look at Y. Li and Y. Chen paper [60].

Then, it should be suggested that the analytical approaches proposed by Michell and density method by topology tends to the natural behaviour written by principal stresses trajectories through the volume, which alluded to the idea that the optimal shape is incarcerated into the body like, following Michelangelo's philosophy, sculptures already exist into marble blocks. The principal stresses trajectories remain one of the most feasible methods to reach structural optimality in terms of stiffness.

A criterion for single load case problem has been determine, but not for multiload conditions, which is the natural condition for every mechanical system.

## 7. Multi-load optimization

To study that part, looking for the aim of robustness and observing the analogy between optimization approaches, a topology optimization which minimizes the weighted sum of the study cases compliance is implemented, as described in the initial chapters.

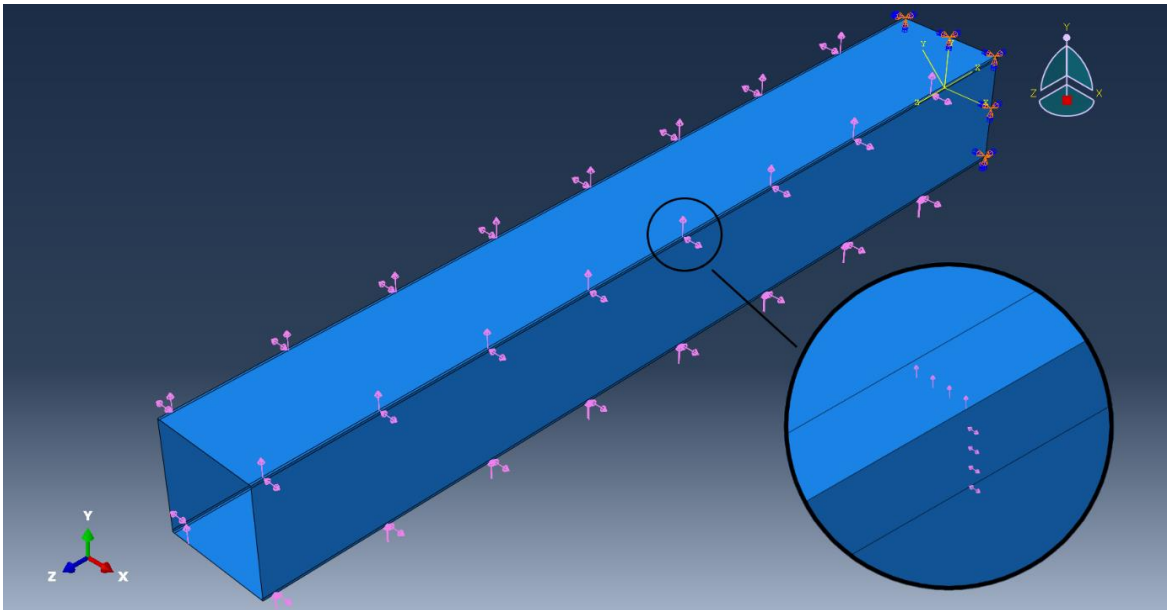
To do that the commercial software ABAQUS is adopted, which uses its own optimizer TOSCA for section, shape and topology optimizations. That programme adopts different topology method, like RAMP and condition-based approach, but for our purpose and familiarity the SIMP approach is used, in order also to compare the results with previous considerations, following the criterion of minimum compliance.

### 7.1. Model

A simplification of a tall building under wind load is created, by a square section tube, with a width-height ratio of 1:8 and fixed at the base edge, modeled by shells, meshed in square elements and the load is defined as a uniform distributed passing through the centre of the section.

About the load modeling, conditions on the model must be used in order to have a clearer result.

The program, in 3D models, doesn't allowed directly transversal distributed load so, in order to recreate the same effect it has been used a pressure force along a mesh strip thin enough to simulate correctly a distributed load line. This technique is often used in practice and is very common as it is possible check in manuals.



*Figure 7.1 ABAQUS model, load assignment method*

For a single load case, considering for example the one orthogonal to one side, it is not applied orthogonally at the middle of the given face, but it is split in two shards acting at the lateral edges along a single mesh strip. The choice to split it into two shard comes from the fact that during the optimization, the strip mesh is subjected to attract material, maybe due to the direct located deformation that the impact region implies, then if it is at the centre of the face an unreal material vertical line should occurs at the end of the computation. On the other side, applying the load on the lateral edges the same phenomenon should occurs there, but it is mitigated and justified by two aspects. The first is that due to the presence of an orthogonal mesh line, which comes from the adjacent face, a huge part of the load effect is directly transmit to the web elements without deforming the orthogonal plate. This aspect is important because considering that the process is focussed on minimum compliance, unreal or due-to-modelling strain effects impact the results. The second aspect is that at the lateral edges a columns formation is quite natural because, especially at the base, they are used to bear the overturning moment, being at the most far region of the face to the neutral axis, as observed in previous chapters.

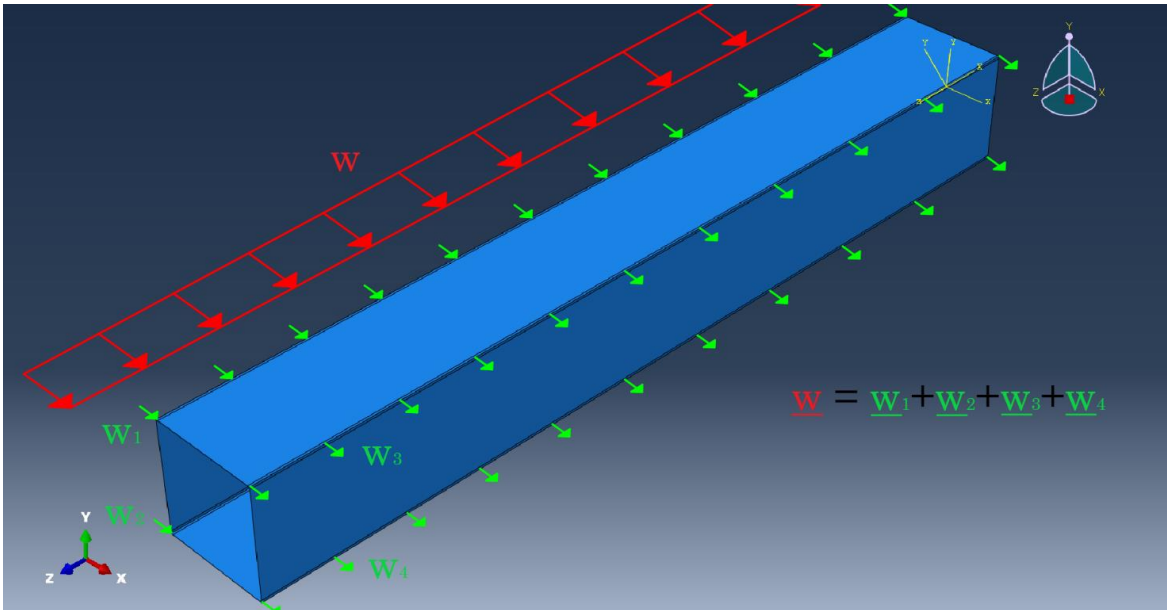


Figure 7.2 ABAQUS model, load redistribution

Another split. As the Stromberg paper suggest, helping the process to achieve a symmetrical diagrid result, the load must be split in other two parts, one pressing the surface and the other in the opposite face pulling outside.

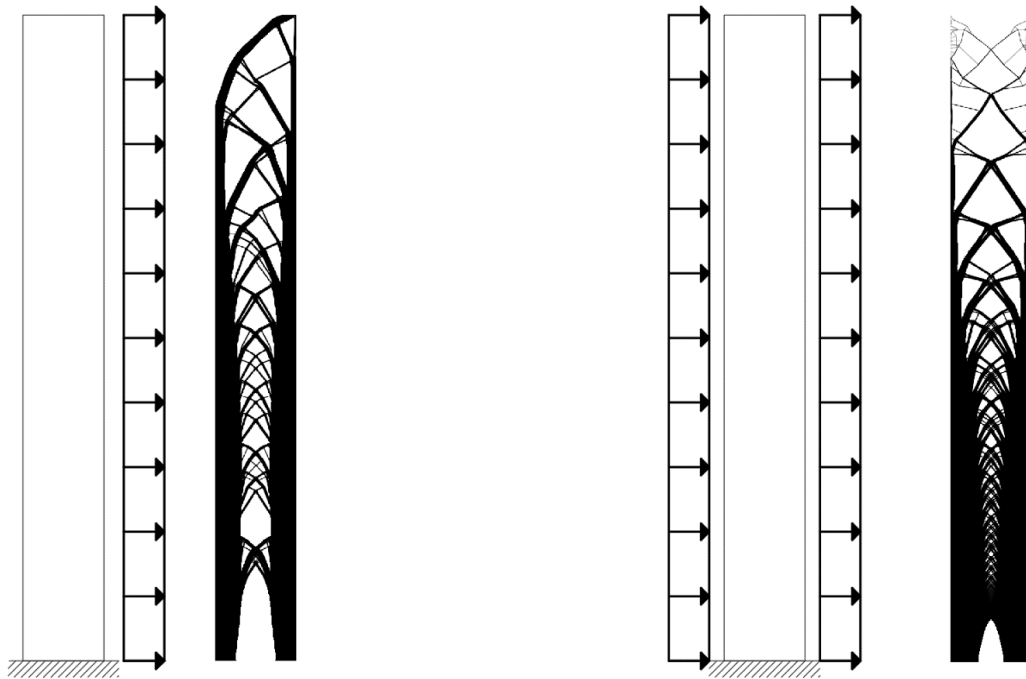


Figure 7.3 Load dependency of symmetry

The last point about loads modelling is the vector subdivision of resultant. Using pressures, which act normally to the application surface, to obtain inclined wind actions the load has been divided in two components, staying on two consecutive faces, where their resultant gives the desired effect.

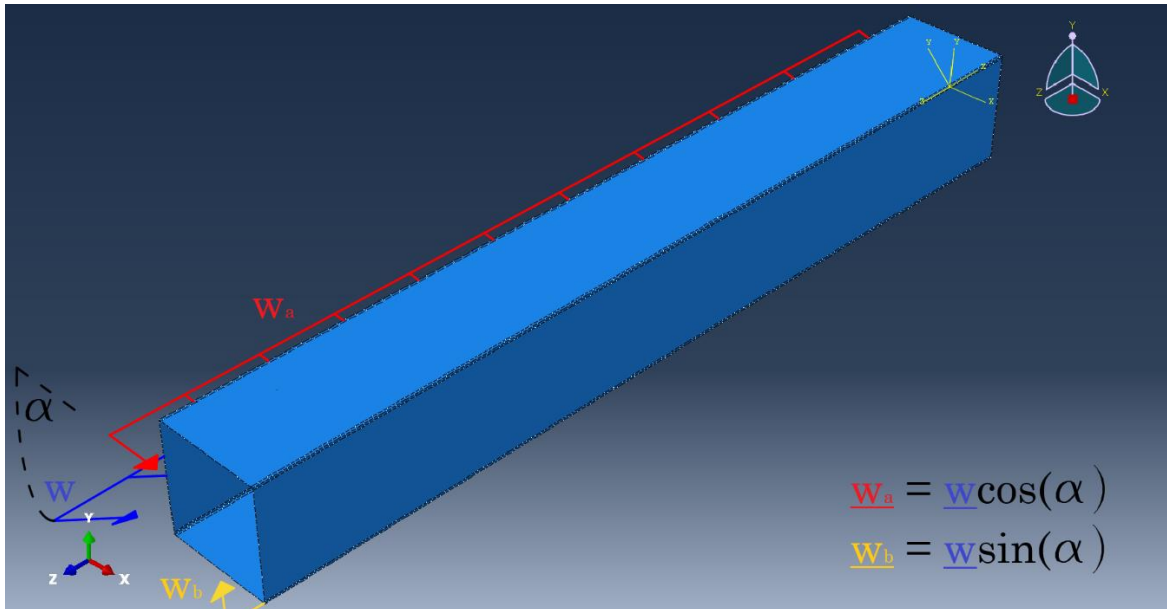
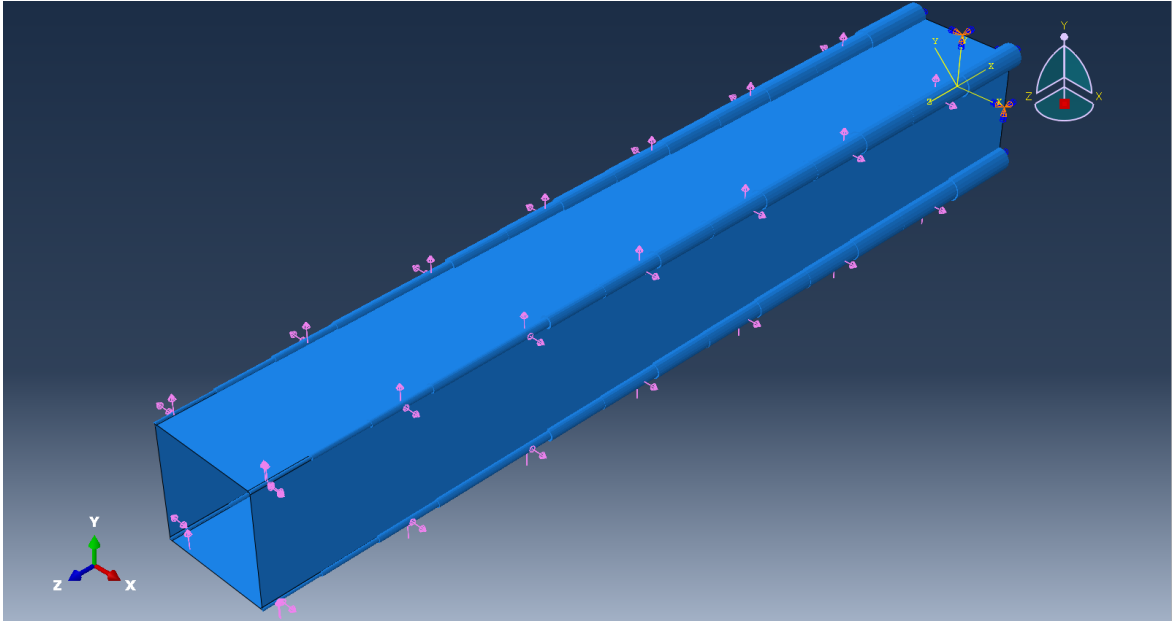


Figure 7.4 ABAQUS model, general direction load repartition

Coming back to overall modelling, following Stromberg suggestions, and in accordance with the column observations, the natural path of the topology method is to put material at the side corners creating columns elements. To avoid focussing on those vertical elements creation and to obtain a clearer diagrid system, beams elements, with variable cross section along the height, have been used and fixed at shell mesh nodes.

More details on this last ABAQUS procedure is given in the Appendix, but it is useful here to remember that the process doesn't depend on material weight or elastic properties themselves but on ratio between them, because normalized quantity enters in the computations. Moreover, the load modulus is not also important, except if more than one is considered, because in that case the relation between them is important, not the modulus value itself.



*Figure 7.5 ABAQUS model, final configuration with beam elements*

Being TOSCA a program focussed on object production and not for large scale structures, a truss tube is obtained instead of a diagrid as in the previous chapters, by the way the analogy is possible due to considerations made in previous chapters.

## **7.2. Single load condition**

In first place, a single load case analysis has been launched to observe the topology results for 3D systems. So, using a load parallel to  $x$  axis, from left to right, with the typical values of penalty factor 3, volume target of 0.3 and symmetrical conditions along both axes, the analysis gives the following result:



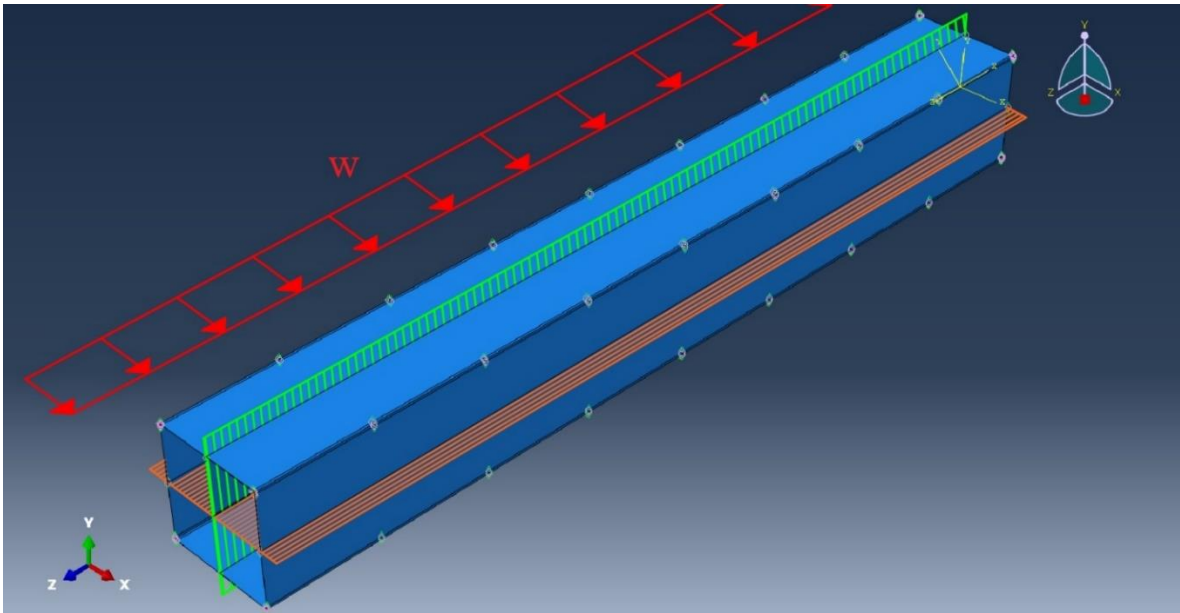


Figure 7.6 ABAQUS model, single load natural symmetry

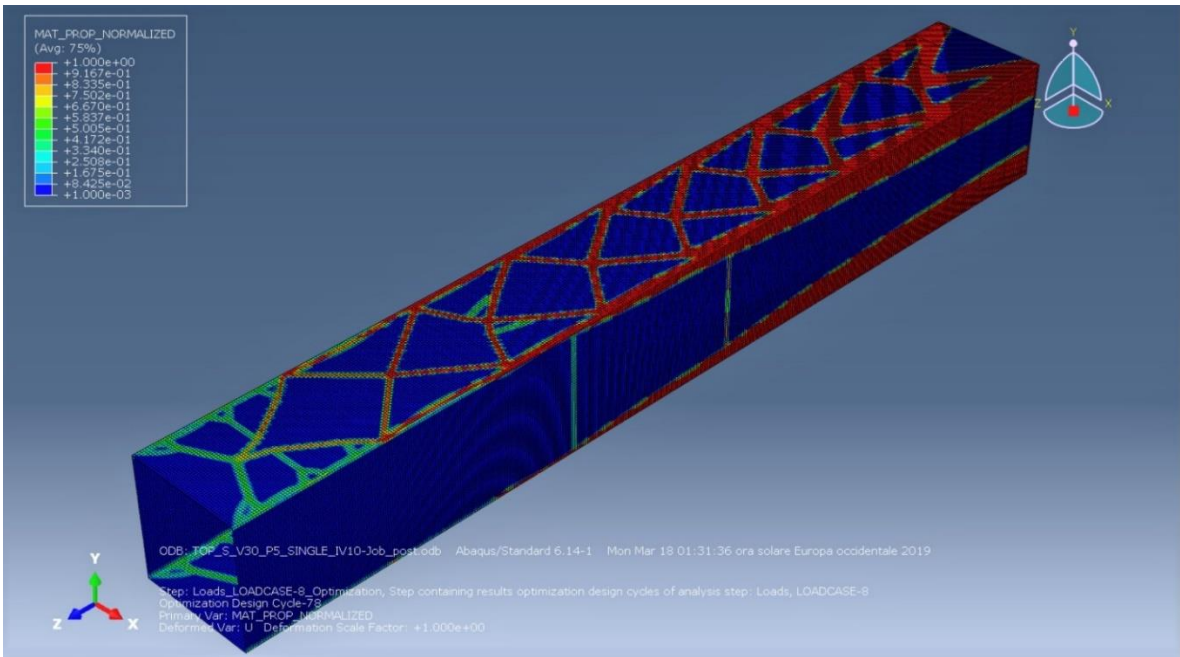


Figure 7.7 ABAQUS model, SIMP results for single load case, general overview

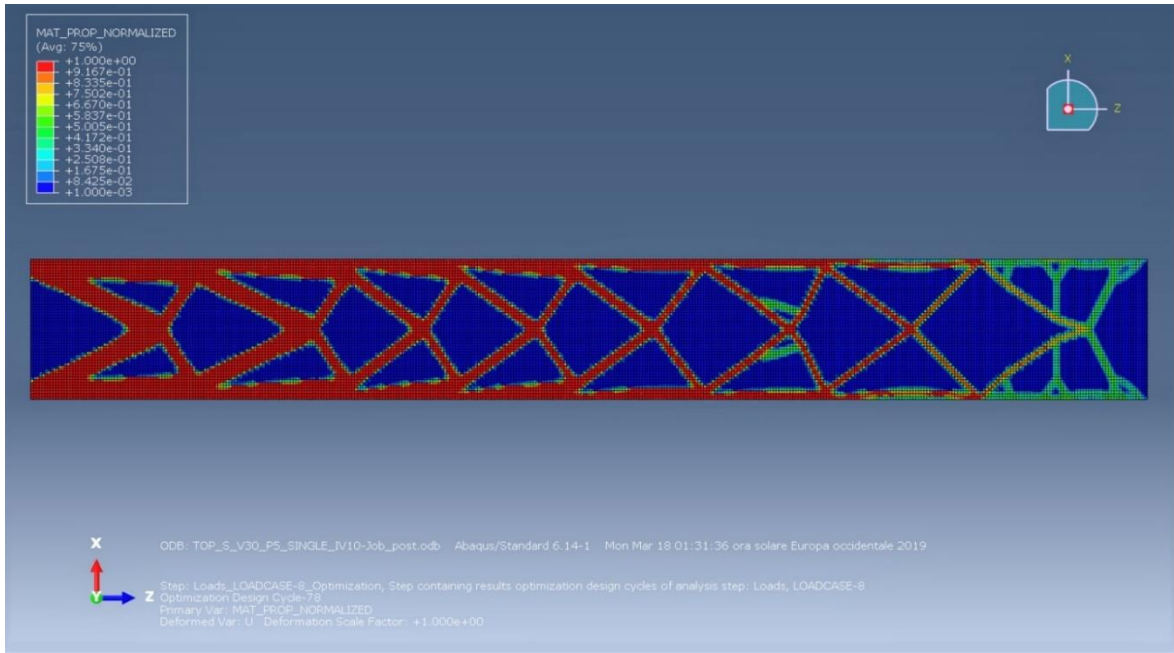


Figure 7.8 ABAQUS model, SIMP results for single load case, web side view

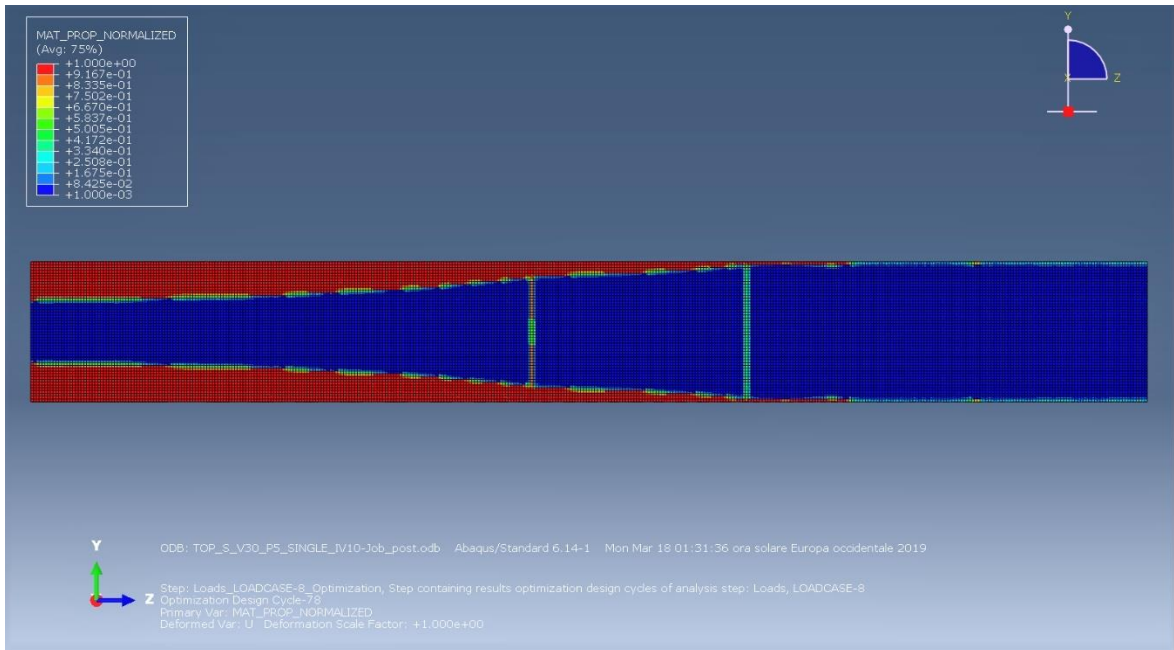
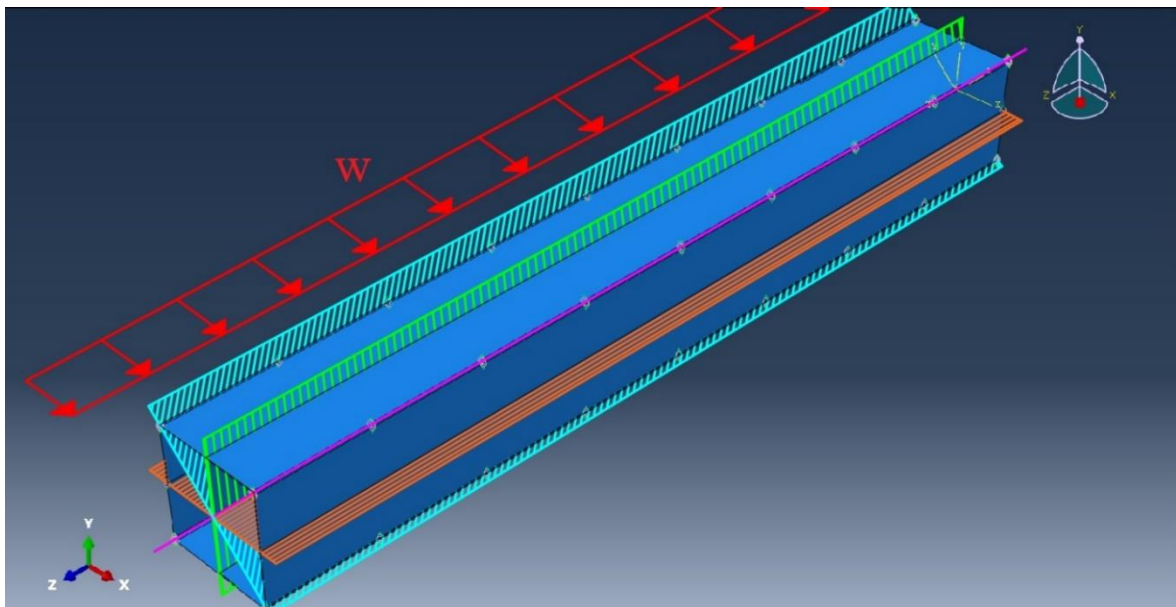


Figure 7.9 ABAQUS model, SIMP results for single load case, flange side view

As it should be observed, the layout configuration match perfectly the principal lines layout, at the flanges sides the overturning moment is carried by vertical

elements which are collocated at the external sides, due to mechanical efficiency for the neutral axis distance, and stress intensities in that part comparing with the central strips. On the web faces the diagrid follows the principal stresses trajectories as on the other sides, with  $45^\circ$  elements moving at the top and most vertical elements at the base. At the top, there is a fan-shape configuration, but it is due to computational procedures, because there is no material enough to create the structure, and the same result can also be obtained in 2D cases as seen previously. It must be noted that when we have a truss, instead of a curtain of diagrid elements, the connection nodes at the sides are not perfectly fitted with the trajectories, but they remain interpretable for our purposes.

A second test has been conducted using the same previous system but adding others two symmetry conditions along quadrant bisectors, in order to get a single façade layout in all the sides. The result is the following:



*Figure 7.10 ABAQUS model, with single load equal faces and imposed symmetry*

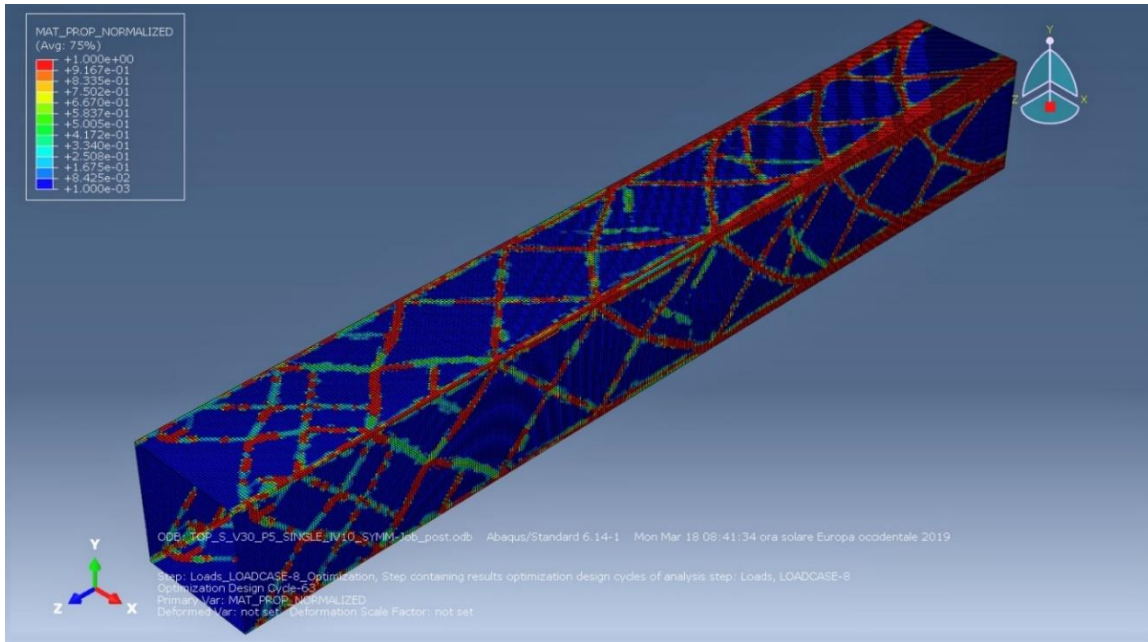


Figure 7.11 ABAQUS model, SIMP results with single load equal faces and imposed symmetry, general overview

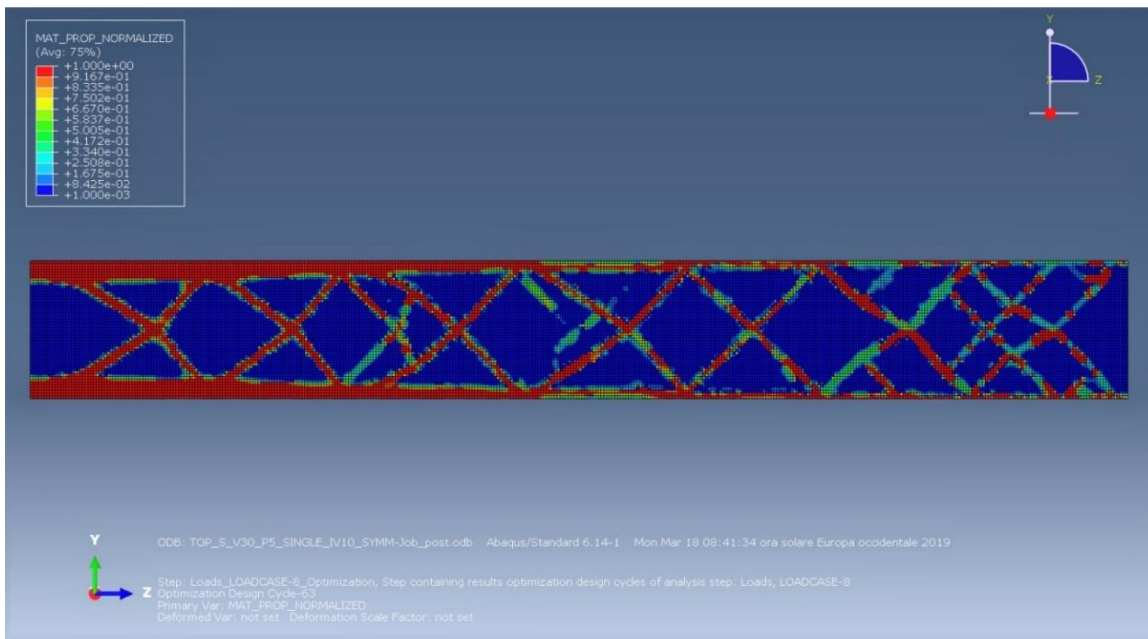


Figure 7.12 ABAQUS model, SIMP results with single load equal faces and imposed symmetry, side overview

The obtained layout suggests that the web configuration remain the most dominant and efficient in the overall layout.

### 7.3. Multi-load condition

Setting a volume fraction limit of 0.3 and under symmetrical conditions for the wind rose action, the following wind case has been considered. From the negative  $x$  side, with a counter clockwise reference, the study angles are:  $0^\circ$ ,  $22.2^\circ$ ,  $45^\circ$ ,  $67.5^\circ$ ,  $90^\circ$ ,  $112.5^\circ$ ,  $135^\circ$ ,  $157.5^\circ$ .

To interpret correctly the result and the topology paths, different test must be implemented, then the penalty factor and the maximum change design cycle.

The first test was made using a penalty factor 3 and an initial maximum change design cycle of 0.25.

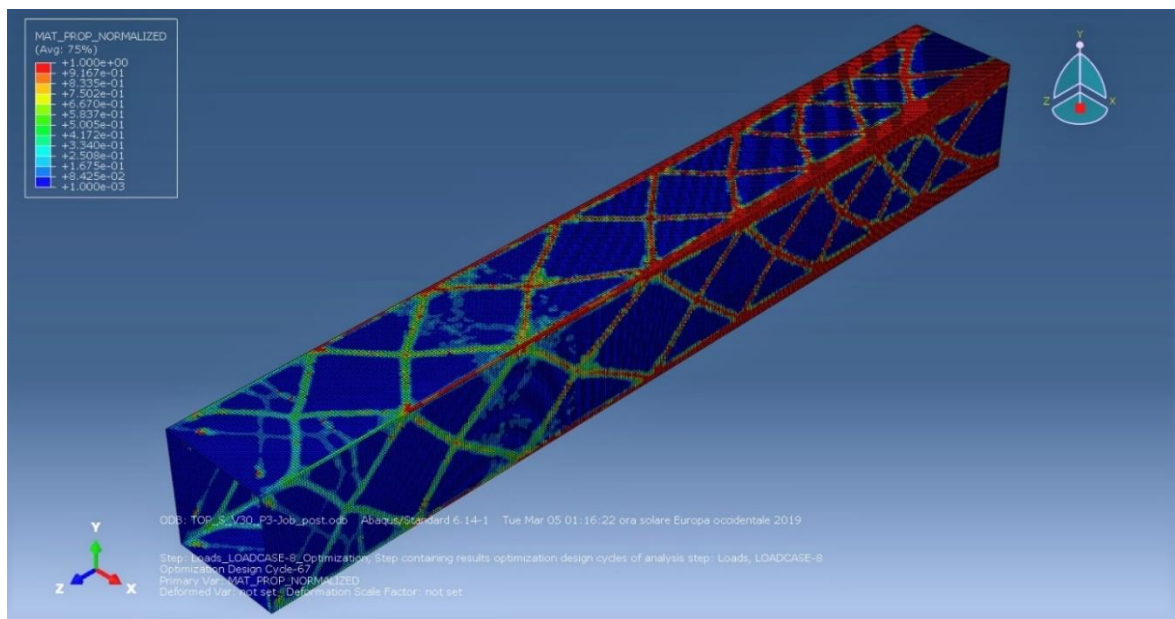


Figure 7.13 ABAQUS model, general view of multi-load topology result

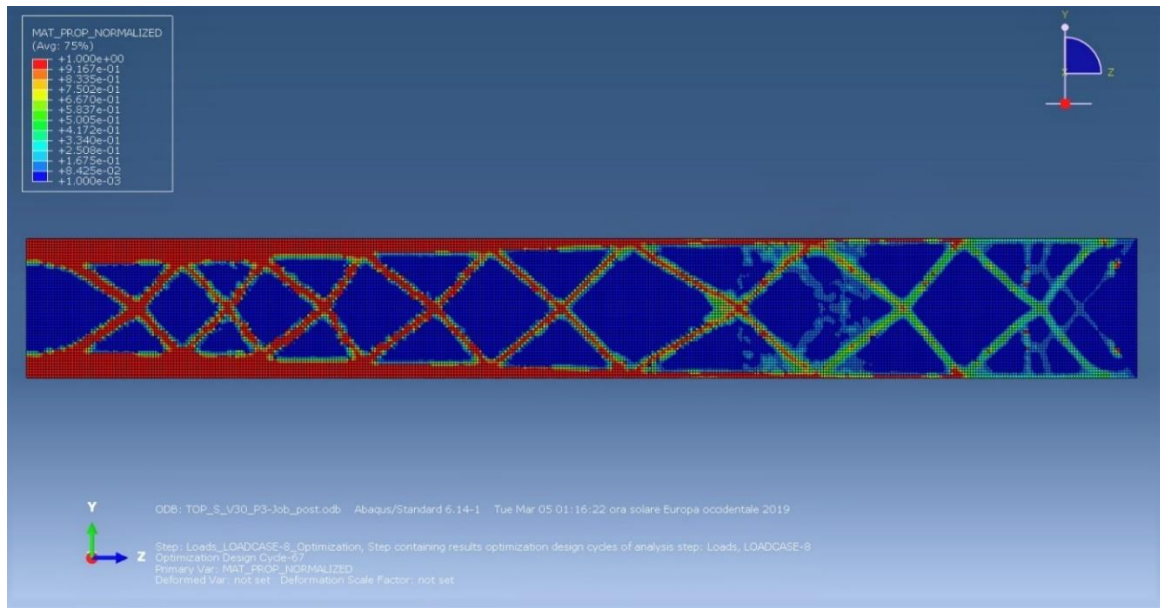


Figure 7.14 ABAQUS model, façade layout from multi-load topology result

Completing the analysis and observing the results, it should be deduced that the optimal layout configuration for all four faces is the one under a wind rose is the one gotten from orthogonal wind force at  $0^\circ$ . This should be justified making some considerations:

- The diagrid façades work on a given plane, then when the action lays on that plane it takes its maximum contribution to the bearing capacity.
- In the first step, with single load case without diagonal symmetry conditions, the web sides represent quite totality of the overall stiffness, then is a nice performant configuration.

#### 7.4. About vertical elements

The vertical elements, mostly at the base region, remain a fundamental component of the final result because, as it was clarified previously, they have the duty to sustain the overturning moment in a region where the flexural component is higher than the shear one.

About their disposition, in this case the square section gives a kind of instruction about where to collocate them, because of the singularity points of the sections which attracts columns. Bu the way their contribution is fundamental for an optimal and robust design, due to their efficiency at the base, and this is demonstrated considering another example on TOSCA described in the appendix.

It should be deduced from previous results that the most efficient layout for a diagrid is the principal stresses trajectories configuration in web members when the wind blows orthogonally to the surface. The following step will test this strategy by some easy model.



## **8. Diagrid with principal stresses trajectories configuration for robust design**

After all those considerations and tests, it is requested to prove the efficiency of this approach. The strategy consists to make some numerical example modelling the structure with Grasshopper on Rhinoceros 6, which is a versatile programme for parametric design and for easy and fast structural analysis with Karamba 3D plug-in.

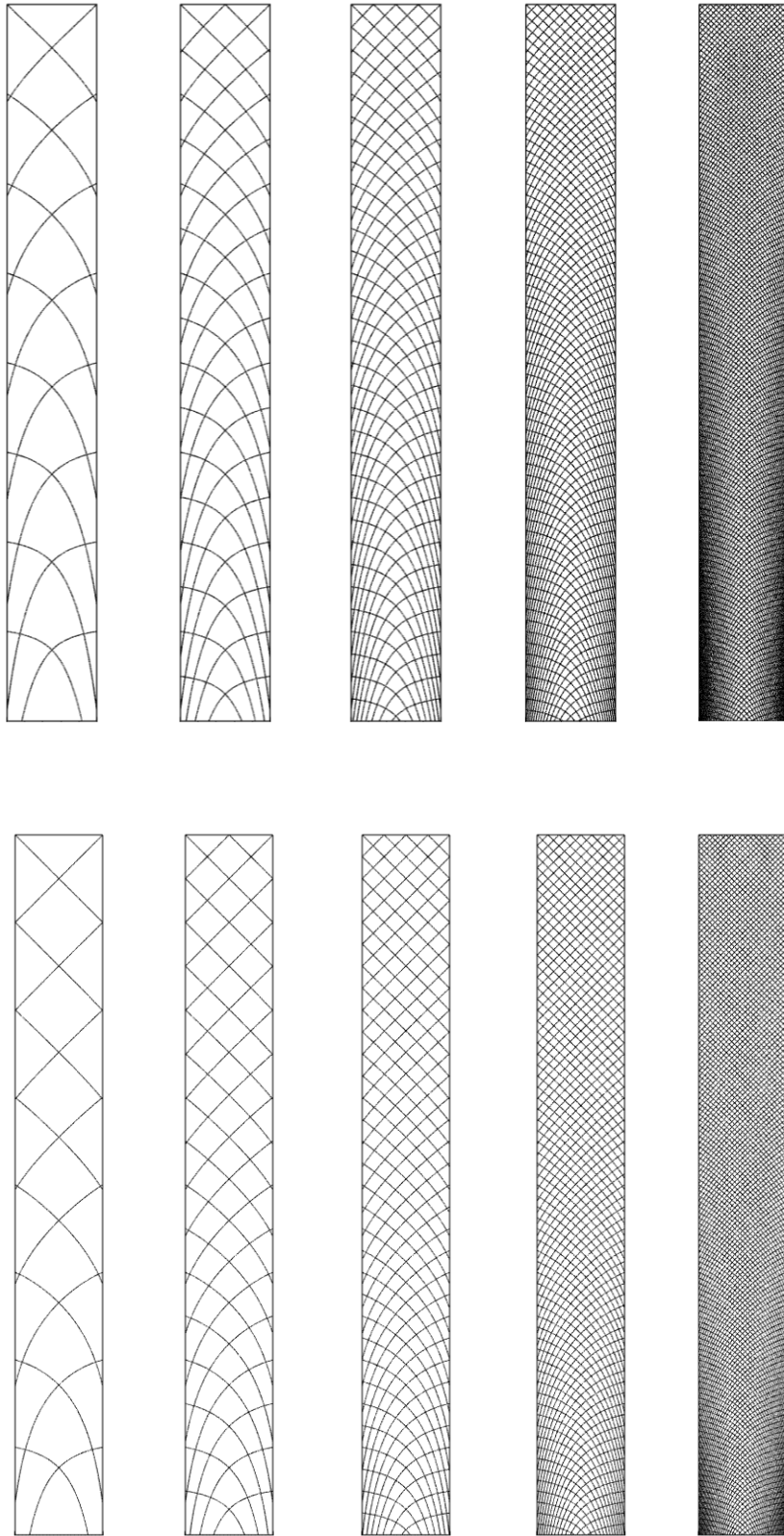
The robust layout that emerges from the study is the stress trajectories web configuration given by an orthogonal load on one face. This layout will be called “web layout” for a matter of simplicity.

### **8.1. The web layout diagrid**

The façade that follows this configuration are used in many projects nowadays, and maybe the clearest examples of this approach come from Skidmore Owings & Merrill LLP studios. They proposed this layout in different projects, because, as Beghni at all [1] say, they present a minimization of the weight, useful in construction industries, but also an aesthetically pleasant result from architectural point of view.

This last sentence follows the philosophy about how the purpose of robustness and strength is strictly linked to the idea of beauty and aesthetic through natural disposition, as flow of forces into a space.

Basing on the simple cantilever layout seen till now, it is shown how the trajectories should be manipulated to define the diagrid which may be detailed as the designer wishes. The set of diagonals made by trajectories intersecting each other forming the cells of the layout and in the following pictures a gradient definition is presented to give the evolution growth of the configuration:



*Figure 8.1 Diagrid configuration for web layout, lines with starting point pitch, from left to right at  $H/B$ ,  $H/2B$ ,  $H/4B$ ,  $H/8B$ ,  $H/16B$ . Configuration without shear-lag (top) and with shear-lag (bottom)*

The façade should be made by refined elements basing on design purposes. The choice of curved element anyway it is not suggested as topology optimization approach has shown, then strict elements are used instead of curved.

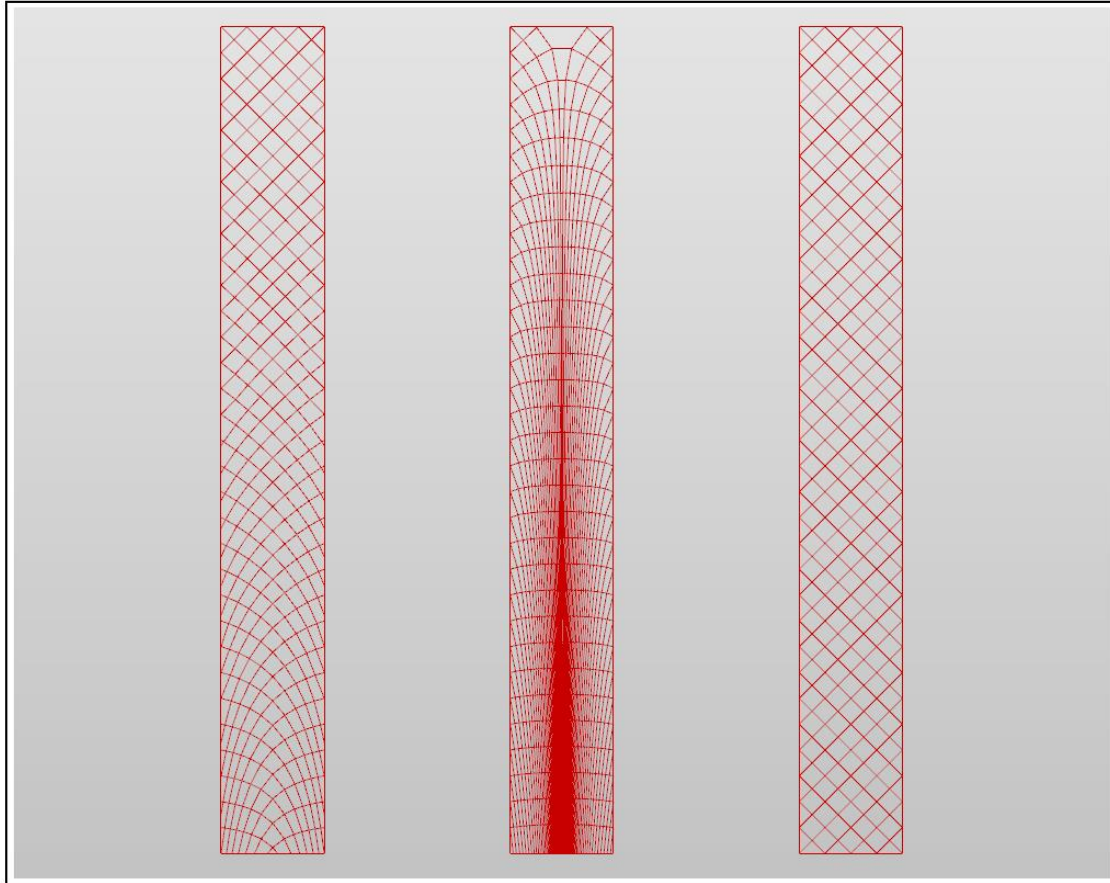
Anyway, to avoid relevant flexural effects and to maintain the curved layout, for aesthetic reasons, a refine configuration is needed, in order to obtain a free length between intersection small enough to neglect the curvature bending effects. Practically, in order to use curved element, another approach should be used. Using a “main” layout, like the configuration 2 seen before, made by thicker elements, a “sub” and smaller layout, as configuration 16 which has an admissible curved elements length, should be use combined with the first as a distributed support layout along the main elements. This implies that the main curved elements will have a series of supports along their length which guarantee the maximum curved length to neglect bending effects.

To test the efficiency of web layout the third configuration with shear-lag effect has been chosen and the steps follow:

- Comparison between web layout, flange layout and regular diagrid layout in 2D case with in plane and out of plane forces, maintaining the same material volume.
- Comparison between web layout and regular diagrid layout in a 3D cantilever problem under a rose of wind actions, maintaining the same material volume.

## **8.2. 2D test**

Modeling on Grasshopper the three diagrid layouts:

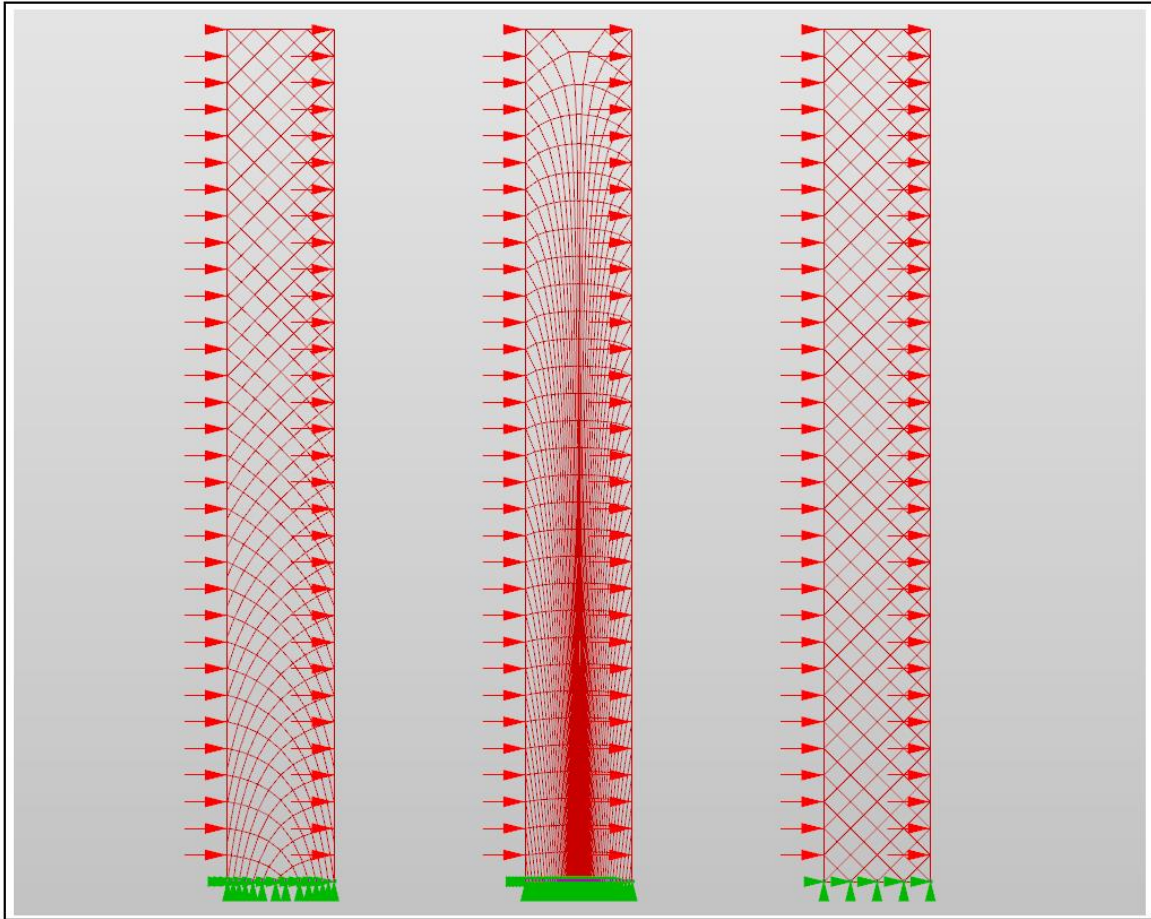


*Figure 8.2 Grasshopper modelling comparison for three layouts in 2D, web layout (right), flange layout (centre), regular layout (left)*

The aim is to compare the performances of the layouts and because of this the cross-section dimensions itself are not important but it is relevant the same used volume fraction between models and the relation between responses. The system is composed as in the previous studies, by distributed loads on the both sides and hinges at the base.

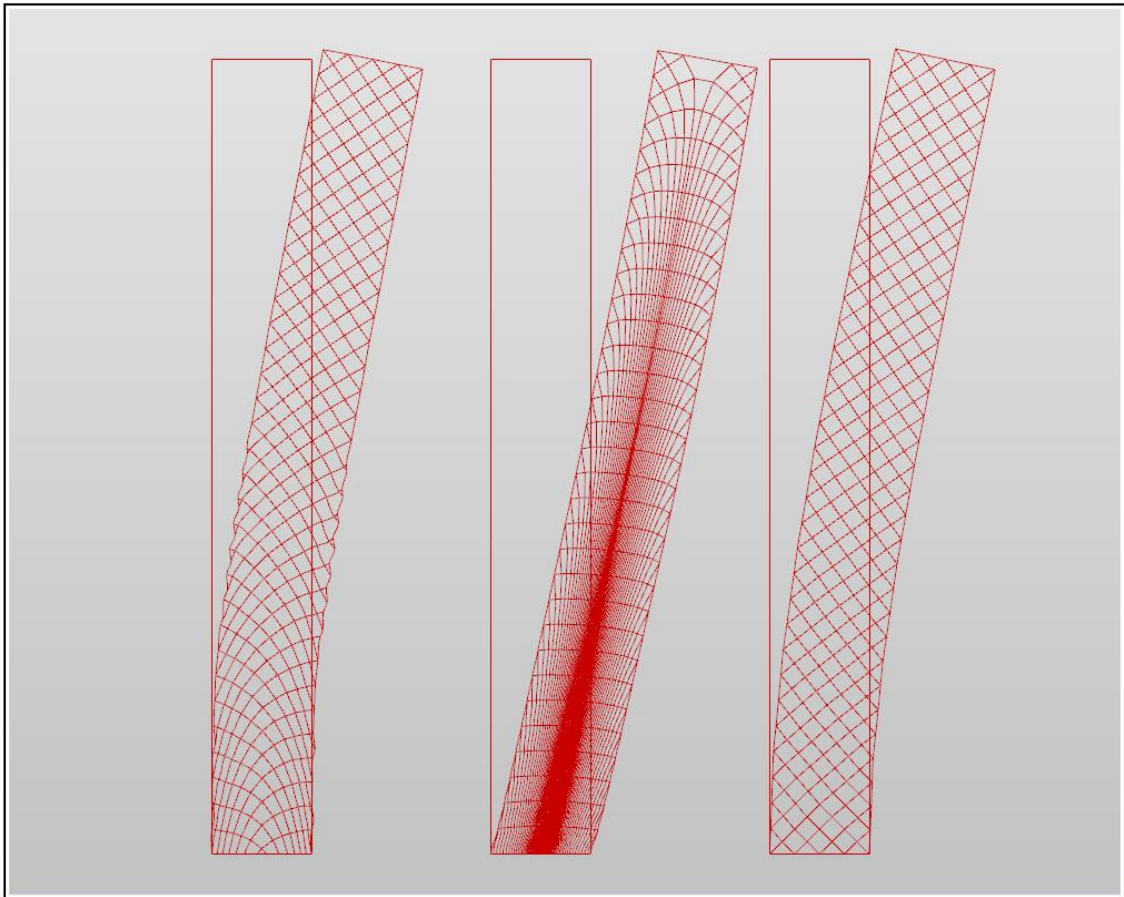
The regular layout, it has been chosen in order to have the same cells at the top of the web layout, so it has 4 cross elements per row, made by 45° inclinations.

The in-plane load system is considered.



*Figure 8.3 Load condition and constraints conditions*

Doing a first order analysis the efficiency of the web layout is clear in terms of displacement and strain energy, the following values have been obtained in comparing the results normalizing with respect to the web responses:



*Figure 8.4 Grasshopper modelling, deformed configurations*

Displacements:

- Web layout: 1.000
- Flange layout: 10.208
- Regular layout 1.120

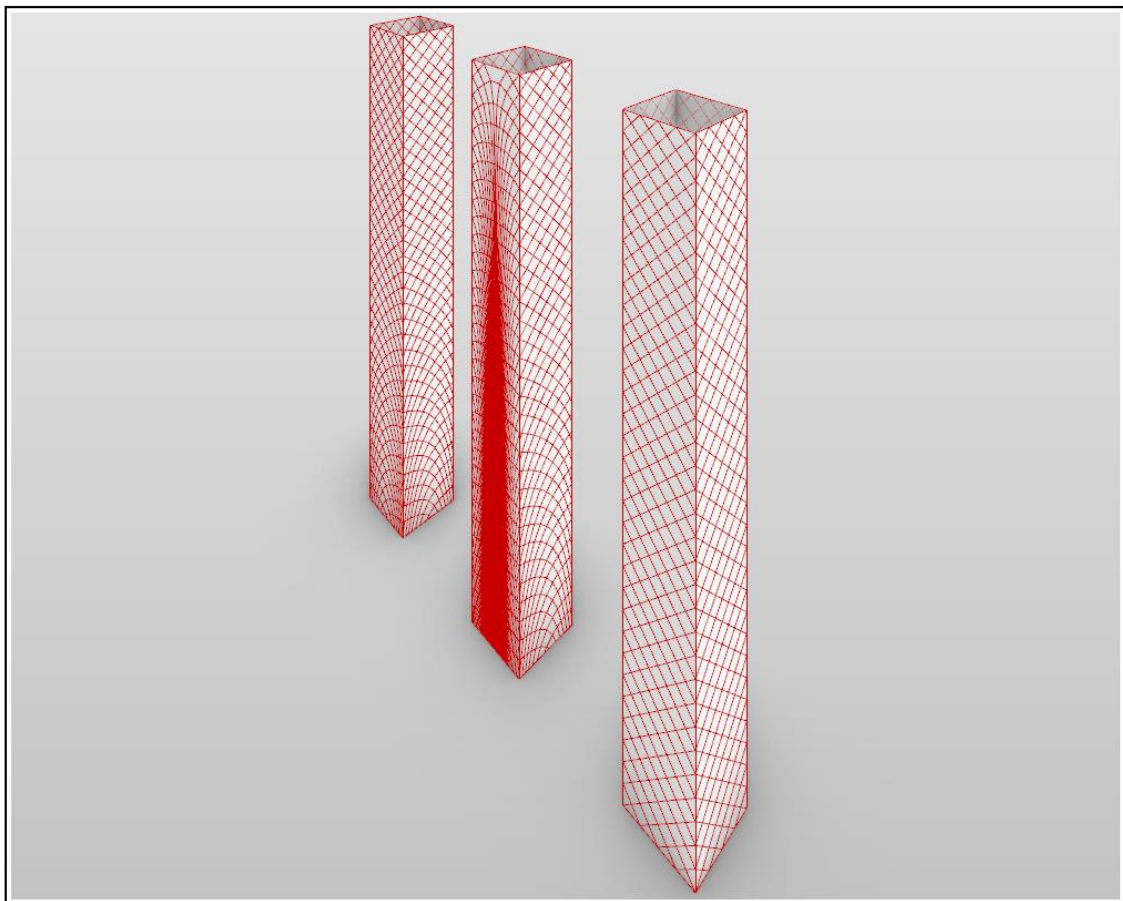
Stain energy:

- Web layout: 1.000
- Flange layout: 14.513
- Regular layout 1.223

The web layout efficiency is evident compared with the flange layout, which deforms like a shear-type frame, while with respect to the regular grid the efficiency is around 12%, for displacements, and 22% for strain energy.

### 8.3. 3D test

Passing now to a 3D case, the cantilever tube is analysed under the rose of wind actions seen before, and the results are normalized with respect to the full web configuration and compared to understand the real efficiency of the solution.



*Figure 8.5 Grasshopper modelling, 3D models, web model (left), PST model (middle) and regular model (right)*

The models are, a total web layout configuration, “web model”, a configuration that follows the principal trajectories for a  $0^\circ$  impact angle case, “PST model” and a regular grid layout model, “regular model”:

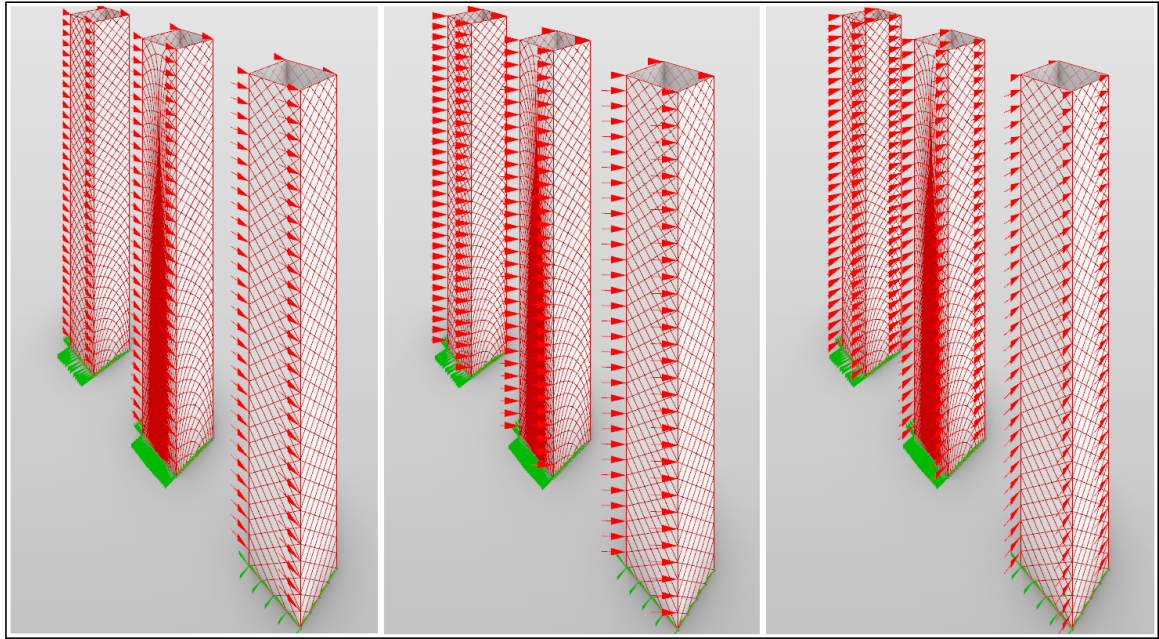
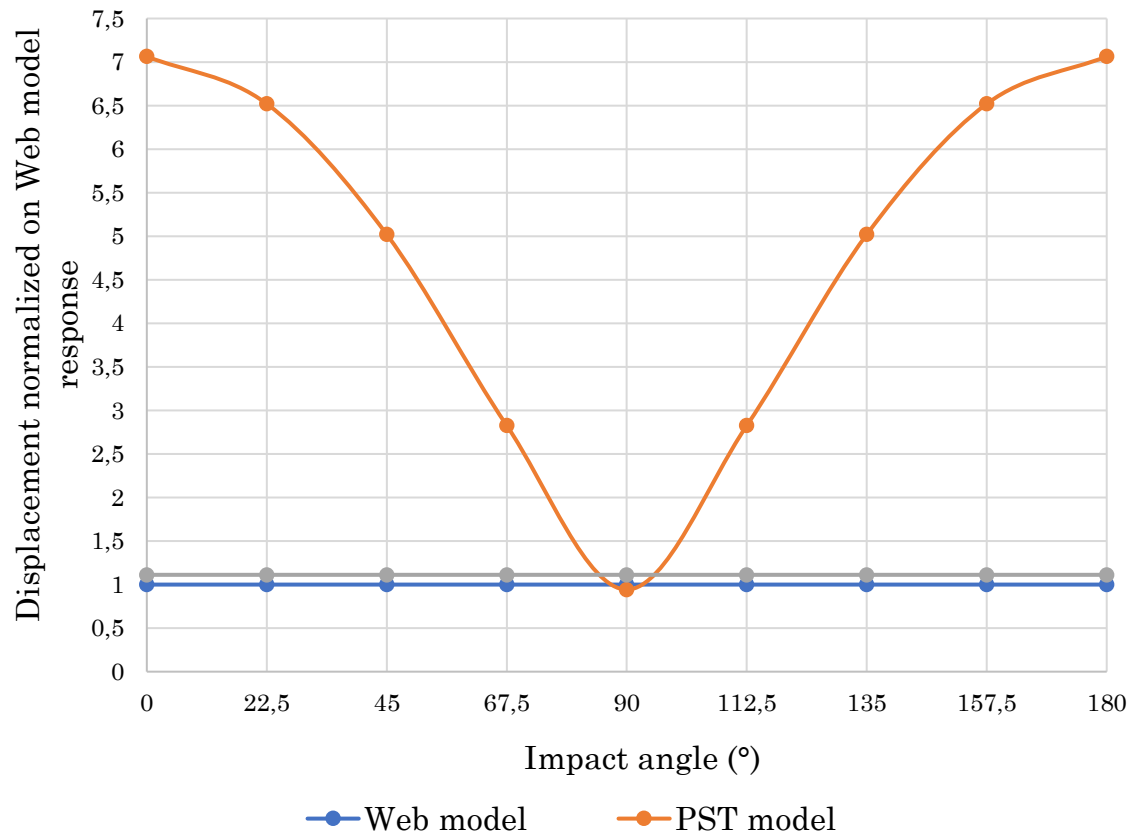


Figure 8.6 Grasshopper modelling, load cases, 0° (left), 45° (middle) and 90° (right)

Displacements:

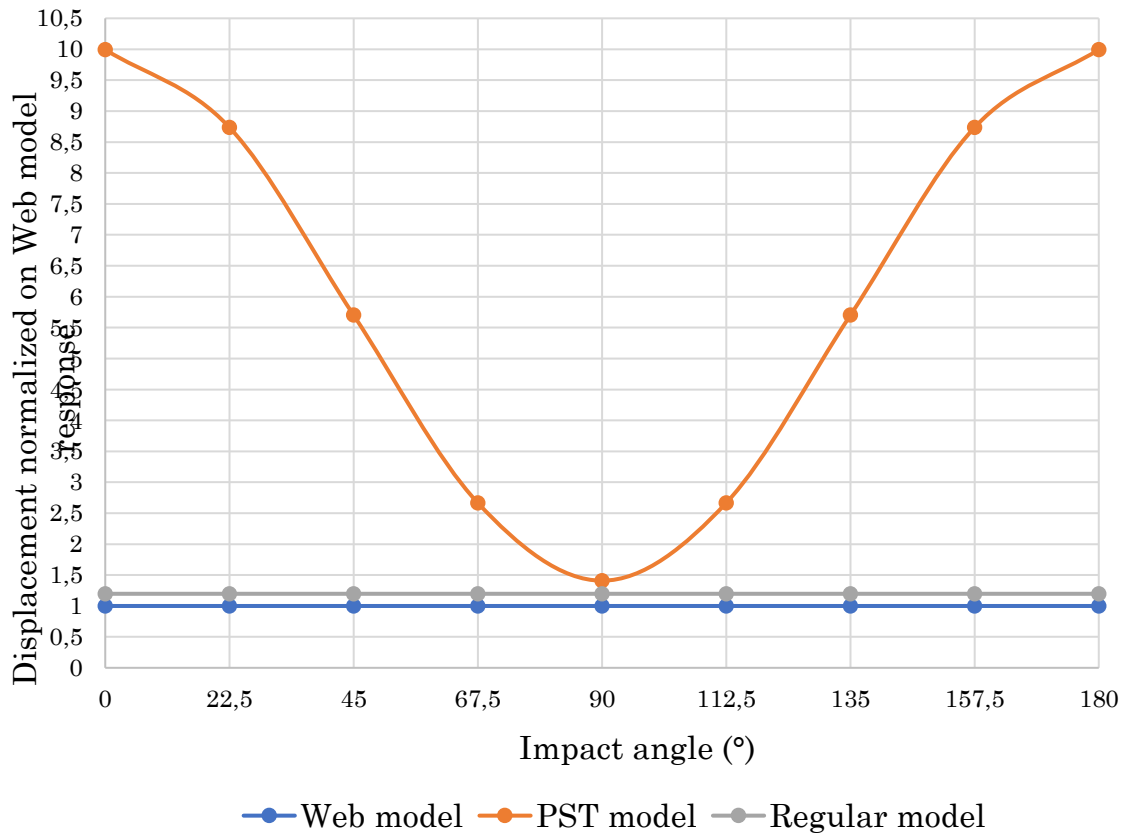
<b><i>Impact angle</i></b>	<b><i>0°</i></b>	<b><i>22.5°</i></b>	<b><i>45°</i></b>	<b><i>67.5°</i></b>	<b><i>90°</i></b>	<b><i>112.5°</i></b>	<b><i>135°</i></b>	<b><i>157.5°</i></b>	<b><i>180°</i></b>
Web model	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>
PST model	<i>7.064</i>	<i>6.520</i>	<i>5.020</i>	<i>2.826</i>	<i>0.943</i>	<i>2.826</i>	<i>5.020</i>	<i>6.520</i>	<i>7.064</i>
Regular model	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>	<i>1.112</i>





Strain energy

<b><i>Impact angle</i></b>	<b><i>0°</i></b>	<b><i>22.5°</i></b>	<b><i>45°</i></b>	<b><i>67.5°</i></b>	<b><i>90°</i></b>	<b><i>112.5°</i></b>	<b><i>135°</i></b>	<b><i>157.5°</i></b>	<b><i>180°</i></b>
Web model	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>
PST model	<i>9.994</i>	<i>8.737</i>	<i>5.701</i>	<i>2.666</i>	<i>1.409</i>	<i>2.666</i>	<i>5.701</i>	<i>8.737</i>	<i>9.994</i>
Regular model	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>	<i>1.197</i>



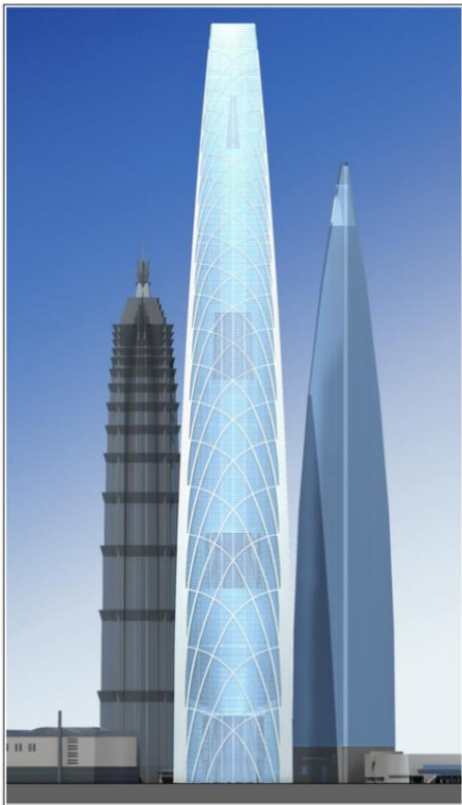
It is evident now that the web layout is the stiffest and robust in all cases. In comparison with the classical PST layout, except for the case 90° which is the optimal for this last layout, the web configuration is hugely stiffer in all the direction. Concerning the classical regular layout, it has been observed that a constant difference occurs, due to the symmetry in both cases in all wind directions, which is around 12% and 20%.

#### 8.4. Nowadays examples

This approach has been used several times by studios for lateral system in tall buildings, but the most incident contribution has been given by Skidmore, Owings & Merrill LLP (SOM), from the Transbay Tower Competition, in San Francisco, to the Burj Jumeira, in Dubai. Here we are going to present some of those examples:



Figure 8.7 Transbay tower competition, San Francisco, California, designed by SOM



a)



b)

Figure 8.8 a) Principal stress trajectories layout concept, from Beghini et al [OPTIMIZATION] and b) the Kunming Junfa Dongfeng Square, Yunnan, China, designed by SOM



*Figure 8.9 CITIC Financial Center, designed by SOM*



*Figure 8.10 Burj-Jumeira tower, Dubai, designed by SOM*

Those last two examples express clearly all the concepts seen till now. The CITIC Financial Center has got, for all its four faces, the web layout that comes

from Michell's, which, as observed, takes origin from principal stresses trajectories, and obtains a robust layout for horizontal actions.

Looking now the Burj-Jameira tower, it follows the same philosophy but thanks to a finer sub grid, can use curved elements that should be interpreted as main trajectories. As it was observed in chapter 4, the curved element is not optimal for flexural reasons. Anyway, if a finer grid is used to sustain those curved elements, creating a curtain of supports along them, the flexural effect caused by curvature being reduced considerably.

## 9. Conclusions

At the end, it has been observed that for the robust design of diagrid structures only diagonal elements are not sufficient due to the variable stress field along the body. Once the load set is properly defined, considering all possible relevant conditions, a weighted combination of the action effects is needed, and the topology optimization presents an efficient tool in this sense, producing a structure that is influenced by all loads set. Concerning the cantilever square tube previously considered, the result is a combination of stress trajectories in the configuration that represents the web behaviour of each face. This should suggest that the most efficient layout for a in-plane diagrid is the one that maximizes the in-plane resistance. Anyway, it is evident, by the analogy between every single optimization method seen in previous chapters, that every approach lays on principal stress trajectories, that are a kind of stress map that defines the bearing capacity of the body itself. For a robust design those paths must be combined as topology optimization does.

To clearly demonstrate this philosophy the next step should be implementing a topology optimization on the same system considered in this work by using a MatLab code like the one proposed by Sigmund [40] but for 3D structures, where the filter radius should be managed and finer results be obtained. This strategy may offer a defined lines paths set, instead of the tube system obtained by ABAQUS, which represents a weighted combination of principal stresses trajectories in various cases. Moreover, in order to affirm from an analytical point of view this concept a weighted result of principal stress trajectories should be obtained, probably minimizing the strain energy in each point for every loading case stress field.

The final result, by the way, must be affine, for tall buildings, to the stress hierarchy, so at the base it must be present more vertical elements for the overturning moment and moving to the top more 45° elements for the shear

effects. The structure that affirms this conclusion is the cylinder considered in the appendix, where due to its polar symmetry there is no difference of load impact around the body and still the vertical elements are required just for the base part and the diagonals moving to the top.

We can assume, following this paper that to design a structure the best way in terms of material costs and structural resistance is to follow the natural path already written into the volume subjected to a given set of loads, because the structure already exists into the material through the force flow, and the engineer must just sculpt it out.

*“You see a block, think of an image: the image is inside you only have to undress it”*

*Michelangelo Buonarroti*

## 10. Appendix

### 10.1. MatLab code for principal stresses trajectories with no shear-lag effect

The MatLab code has been developed only for a cantilever tube clamped at one edge and under a transversal distributed load directed to the barycentre of the cross-section, which is a double symmetric quadrilateral. The load can vary in direction changing the angle referred to the  $x$  axis and it is then split in two components respectively on  $x$  or  $y$  direction. Of course, the beam dimensions and slenderness may be modified in relation to purposes and scopes.

About the representation procedure for the trajectories, once the internal actions have been computed, also the stress state in every section point is computed, following for normal stresses, the constant trend at the flanges and linear at webs, and for shear stresses the flux given by the first order moment in the Jourawski's formula. For the principal stresses map, a uniform grid has been used on each side and in every node the principal stress "one" and "two", according with Mohr's theory, and the relative orientation angles have been computed. Then a vector representation by the command "*quiver*" gives the flux of the force through the body. This part was fundamental to compare the correct evaluation of the represented trajectories. For the curves, the representation starts from a boundary point which stays on the vertical line or on the top one. For every chosen  $z$  value two starting points are taken, one on the right side and the other on the left of the considered face, then for each point the principal stress one and two are computed obtaining the respective angles. From those angles, imposing an infinitesimal  $x$ , or  $y$ , step, the tangent, moving towards the other side, identifies the new  $z$  coordinate. Iteratively, till the end of the boundary, the curve is drawn and the results are accurate as much as the  $x$ , or  $y$ , step is small, and basically this is exactly a finite difference resolution method.



Thinking about the number of lines, in first place, this is equal to the ratio between height and base but if we modify the multiplier “ $n$ ” this number increase  $n$ -times on the vertical sides plus  $n$  curves from the top edge. It must be remarked that all the quantities are referred to meters, for lengths, Newtons, for forces, and grades for angles.

Here follows the explained code:

```

%% Data

% Geometry
a=25; %Side length A and C
b=25; %Side length B and D
H=200; %Height
tb=1; %Side thickness A and C
ta=1; %Side thickness B and D
faces=4; %Faces number (not changeable)

% Wind load
W=100000; %Wind load intensity, as uniform distributed
          %(this value doesn't influence the result)
teta_x=0; %Counter clockwise orientation angle from x
axis
w_x=W*cos(teta_x*pi/180); %Wind component along x
w_y=W*sin(teta_x*pi/180); %Wind component along y

% Grid subdivision for mapping
r=200; %Z subdivision
p=25; %X subdivision
q=25; %Y subdivision

% Inertia moments
Ia=2*ta*((a-2*tb)^3)/12+2*tb*b*((a-tb)/2)^2; %Inertia moment on Y
Ib=2*tb*((b-2*ta)^3)/12+2*ta*a*((b-ta)/2)^2; %Inertia moment on X

% Trajectories parameters
m=10000; %Increment along X = a/m
mm=10000; %Increment along Y = b/mm
num=32; %Starting points multiplier
          %N°Points on X(or Y) = num
          %N°Points on Z = N°Points on X(orY) * 8
e=(H/a)*num+num; %Counter for cycles

%% Faces

%Set of matrices for each face in the grid
%(All the variables are explained later)

%Face A
M_A=zeros([r+1 1]); %Moment
Vw_A=zeros([r+1 1]); %Shear by w
Vww_A=zeros([r+1 1]); %Shear by ww
N_A=zeros([r+1 1]); %Axial force

```

```

Sw_A=zeros([1 (p+1)]);           %First order moment for wind w
Sww_A=zeros([1 (p+1)]);         %First order moment for wind ww

Sigma_A=zeros([(r+1) (p+1)]);   %Normal stresses
Tau_A=zeros([(r+1) (p+1)]);     %Shear stresses
Sigma1_A=zeros([(r+1) (p+1)]);  %Principal stresses 1
Sigma2_A=zeros([(r+1) (p+1)]);  %Principal stresses 2
Fi1_A=zeros([(r+1) (p+1)]);     %Principal angle 1, with respect to Z
Fi2_A=zeros([(r+1) (p+1)]);     %Principal angle 2, with respect to Z

%Face C
M_C=zeros([(r+1) 1]);
Vw_C=zeros([(r+1) 1]);
Vww_C=zeros([(r+1) 1]);
N_C=zeros([(r+1) 1]);

Sw_C=zeros([1 (p+1)]);
Sww_C=zeros([1 (p+1)]);

Sigma_C=zeros([(r+1) (p+1)]);
Tau_C=zeros([(r+1) (p+1)]);
Sigma1_C=zeros([(r+1) (p+1)]);
Sigma2_C=zeros([(r+1) (p+1)]);
Fi1_C=zeros([(r+1) (p+1)]);
Fi2_C=zeros([(r+1) (p+1)]);

%Face B
M_B=zeros([(r+1) 1]);
Vw_B=zeros([(r+1) 1]);
Vww_B=zeros([(r+1) 1]);
N_B=zeros([(r+1) 1]);

Sw_B=zeros([1 (q+1)]);
Sww_B=zeros([1 (q+1)]);

Sigma_B=zeros([(r+1) (q+1)]);
Tau_B=zeros([(r+1) (q+1)]);
Sigma1_B=zeros([(r+1) (q+1)]);
Sigma2_B=zeros([(r+1) (q+1)]);
Fi1_B=zeros([(r+1) (q+1)]);
Fi2_B=zeros([(r+1) (q+1)]);

%Face D
M_D=zeros([(r+1) 1]);
Vw_D=zeros([(r+1) 1]);
Vww_D=zeros([(r+1) 1]);
N_D=zeros([(r+1) 1]);

Sw_D=zeros([1 (q+1)]);
Sww_D=zeros([1 (q+1)]);

Sigma_D=zeros([(r+1) (q+1)]);
Tau_D=zeros([(r+1) (q+1)]);
Sigma1_D=zeros([(r+1) (q+1)]);
Sigma2_D=zeros([(r+1) (q+1)]);
Fi1_D=zeros([(r+1) (q+1)]);
Fi2_D=zeros([(r+1) (q+1)]);

```

```

%Here the cycle start for each face, and in one step the principal
stresses
%map and the trajectories are computed
%(The parameters equal to 1 and -1 are used to adapt the cycle to every
%face orientation. More over, the "w" load is referred to the load in the
%same direction of the considered face, while the "ww" load for the
%orthogonal direction, then they change for each face.)

```

```

for ii=1:1:faces
    if ii==1;
        tw=ta;
        tww=tb;
        w=w_x;
        ww=w_y;
        sw=1;
        sww=1;
        f=1;
        lf=1;
        l=a;
        ll=b;
        u=p;
        Iw=Ia;
        Iww=Ib;
    elseif ii==2;
        tw=ta;
        tww=tb;
        w=w_x;
        ww=w_y;
        sw=-1;
        sww=1;
        f=-1;
        lf=-1;
        l=a;
        ll=b;
        u=p;
        Iw=Ia;
        Iww=Ib;
    elseif ii==3;
        tw=tb;
        tww=ta;
        w=w_y;
        ww=w_x;
        sw=1;
        sww=-1;
        f=1;
        lf=-1;
        l=b;
        ll=a;
        u=q;
        Iw=Ib;
        Iww=Ia;
    elseif ii==4;
        tw=tb;
        tww=ta;
        w=w_x;
        sw=-1;
        sww=-1;
        f=-1;
        lf=1;
        l=b;
        ll=a;
    end
end

```

```

    u=q;
    Iw=Ib;
    Iww=Ia;
end

%Grid definition for the principal stresses mapping
Sw=zeros([1 (u+1)]);
Sww=zeros([1 (u+1)]);

Sigma=zeros([(r+1) (u+1)]);
Tau=zeros([(r+1) (u+1)]);
Sigma1=zeros([(r+1) (u+1)]);
Sigma2=zeros([(r+1) (u+1)]);
Fi1=zeros([(r+1) (u+1)]);
Fi2=zeros([(r+1) (u+1)]);

%Grid
[X,Z]=meshgrid(-f*l/2:f*l/u:f*l/2,H:-H/r:0);
Y=(-lf*l/2)*ones([(r+1) (u+1)]);

%Z coordinates
for j=1:1:r+1;
    if j==1;
        z=0.0001;
    else
        z=(H/r)*(j-1);
    end

    %Internal actions
    M(j)=-w*(z^2)/2;
    Vw(j)=w*z;
    Vww(j)=ww*z;
    N(j)=((ww*(z^2)/2)*(lf*l/2))/Iw)*l*tw;

    %X or Y coordinates for the give Z coordinate
    for i=1:1:u+1;
        x=f*(-(l/2)+(l/u)*(i-1));
        Sw(i)=sw*tw*(l*l)/4+sw*tw*0.5*((l/2)^2-x^2);
        Sww(i)=sww*tw*(l/2)*x;
        Tau(j,i)=Vw(j)*Sw(i)/Iw+Vww(j)*Sww(i)/Iww;
        Sigma(j,i)=M(j)*x/Iw+N(j)/(l*tw);
        Sigma1(j,i)=Sigma(j,i)/2+sqrt((Sigma(j,i)^2)/4+Tau(j,i)^2);
        Sigma2(j,i)=Sigma(j,i)/2-sqrt((Sigma(j,i)^2)/4+Tau(j,i)^2);
        A=2*Tau(j,i)/Sigma(j,i);

        %Angle evaluation procedure
        if Sigma(j,i)==0;
            if Tau(j,i)>0;
                Fi1(j,i)=45;
                Fi2(j,i)=135;
            elseif Tau(j,i)<0;
                Fi1(j,i)=-45;
                Fi2(j,i)=-135;
            end
        elseif Tau(j,i)==0;
            if Sigma(j,i)>0;
                Fi1(j,i)=0;
                Fi2(j,i)=90;
            elseif Sigma(j,i)<0;
                Fi2(j,i)=180;
            end
        end
    end
end

```

```

        Fil(j,i)=-90;
    end
elseif Tau(j,i)>0;
    if Sigma(j,i)>0;
        Fil(j,i)=0.5*(atan(A))*180/pi;
        Fi2(j,i)=Fil(j,i)+90;
    elseif Sigma(j,i)<0
        Fil(j,i)=0.5*(atan(A))*180/pi+90;
        Fi2(j,i)=Fil(j,i)+90;
    end
elseif Tau(j,i)<0;
    if Sigma(j,i)>0;
        Fil(j,i)=0.5*(atan(A))*180/pi;
        Fi2(j,i)=Fil(j,i)-90;
    elseif Sigma(j,i)<0
        Fil(j,i)=0.5*(atan(A))*180/pi-90;
        Fi2(j,i)=Fil(j,i)-90;
    end
end
Fi1(j,i)=Fil(j,i);
Fi2(j,i)=Fi2(j,i);

%Face matrices assignment
if ii==1
    M_A(j)=M(j);
    Vw_A(j)=Vw(j);
    N_A(j)=N(j);
    Sw_A(i)=Sw(i);
    Tau_A(j,i)=Tau(j,i);
    Sigma_A(j,i)=Sigma(j,i);
    Sigma1_A(j,i)=Sigma1(j,i);
    Sigma2_A(j,i)=Sigma2(j,i);
    X_A=X;
    Y_A=Y;
    Z_A=Z;
    Fil_A(j,i)=Fil(j,i);
    Fi2_A(j,i)=Fi2(j,i);
elseif ii==2
    M_C(j)=M(j);
    Vw_C(j)=Vw(j);
    N_C(j)=N(j);
    Sw_C(i)=Sw(i);
    Tau_C(j,i)=Tau(j,i);
    Sigma_C(j,i)=Sigma(j,i);
    Sigma1_C(j,i)=Sigma1(j,i);
    Sigma2_C(j,i)=Sigma2(j,i);
    X_C=X;
    Y_C=Y;
    Z_C=Z;
    Fil_C(j,i)=Fil(j,i);
    Fi2_C(j,i)=Fi2(j,i);
elseif ii==3
    M_B(j)=M(j);
    Vw_B(j)=Vw(j);
    N_B(j)=N(j);
    Sw_B(i)=Sw(i);
    Tau_B(j,i)=Tau(j,i);
    Sigma_B(j,i)=Sigma(j,i);
    Sigma1_B(j,i)=Sigma1(j,i);
    Sigma2_B(j,i)=Sigma2(j,i);
    X_B=Y;

```

```

        Y_B=X;
        Z_B=Z;
        Fi1_B(j,i)=Fi1(j,i);
        Fi2_B(j,i)=Fi2(j,i);
    elseif ii==4
        M_D(j)=M(j);
        Vw_D(j)=Vw(j);
        N_D(j)=N(j);
        Sw_D(i)=Sw(i);
        Tau_D(j,i)=Tau(j,i);
        Sigma_D(j,i)=Sigma(j,i);
        Sigma1_D(j,i)=Sigma1(j,i);
        Sigma2_D(j,i)=Sigma2(j,i);
        X_D=Y;
        Y_D=X;
        Z_D=Z;
        Fi1_D(j,i)=Fi1(j,i);
        Fi2_D(j,i)=Fi2(j,i);
    end
end
end

%Unit vector definition for "quiver" components
CFi1=cos(pi/180*Fi1);
SFi1=sin(pi/180*Fi1);
CFi2=cos(pi/180*Fi2);
SFi2=sin(pi/180*Fi2);
R=ones(r+1,u+1);
CC1=sw*SFi1.*R;
SS1=CFi1.*R;
CC2=sw*SFi2.*R;
SS2=CFi2.*R;
O=zeros([(r+1) (u+1)]);

if ii==1
    CC1_A=CC1;
    SS1_A=SS1;
    CC2_A=CC2;
    SS2_A=SS2;
    O_A=O;
elseif ii==2
    CC1_C=CC1;
    SS1_C=SS1;
    CC2_C=CC2;
    SS2_C=SS2;
    O_C=O;
elseif ii==3
    CC1_B=CC1;
    SS1_B=SS1;
    CC2_B=CC2;
    SS2_B=SS2;
    O_B=O;
elseif ii==4
    CC1_D=CC1;
    SS1_D=SS1;
    CC2_D=CC2;
    SS2_D=SS2;
    O_D=O;
end

%Trajectories

```

```

%(Here the index "_1" or "_2" refers to the left or right side of the
%considered face, watching it from the outside to the inside.)
%(The quantities remain the same but the curves point coordinates X,Y
%and Z have been added.)

```

```

%Point coordinates
F2_1=NaN([ (m+1) (e) ]);
F2_2=NaN([ (m+1) (e) ]);
F1_1=NaN([ (m+1) (e) ]);
F1_2=NaN([ (m+1) (e) ]);
X2_1=NaN([ (m+1) (e) ]);
X2_2=NaN([ (m+1) (e) ]);
Z2_1=NaN([ (m+1) (e) ]);
Z2_2=NaN([ (m+1) (e) ]);
Y2_1=NaN([ (m+1) (e) ]);
Y2_2=NaN([ (m+1) (e) ]);
X1_1=NaN([ (m+1) (e) ]);
X1_2=NaN([ (m+1) (e) ]);
Z1_1=NaN([ (m+1) (e) ]);
Z1_2=NaN([ (m+1) (e) ]);
Y1_1=NaN([ (m+1) (e) ]);
Y1_2=NaN([ (m+1) (e) ]);

```

```

Sigma1_1=NaN([ (m+1) (e) ]);
Sigma1_2=NaN([ (m+1) (e) ]);
Sigma2_1=NaN([ (m+1) (e) ]);
Sigma2_2=NaN([ (m+1) (e) ]);

```

```

%Fake components for the evaluation
Sigma11=NaN([ (m+1) (e) ]);
Sigma22=NaN([ (m+1) (e) ]);
F1=NaN([ (m+1) (e) ]);
F2=NaN([ (m+1) (e) ]);
X1=NaN([ (m+1) (e) ]);
X2=NaN([ (m+1) (e) ]);
Z1=NaN([ (m+1) (e) ]);
Z2=NaN([ (m+1) (e) ]);
Y1=NaN([ (m+1) (e) ]);
Y2=NaN([ (m+1) (e) ]);

```

```

%Side starting point

```

```

for iii=1:1:2
    if iii==1
        x00=(-1/2)*f;
        ff=1;
    elseif iii==2
        x00=(1/2)*f;
        ff=-1;
    end

```

```

%Starting point selection

```

```

for i=1:1:e;
    if i<num+1
        z0=0.0001;
        x0=-x00-i*(1/num)*ff*f;
    else
        z0=(H/(e-num))*(i-num);
        x0=x00;
    end

```

```

M0=-w*(z0^2)/2;
Vw0=w*z0;
Vww0=ww*z0;
N0=(( (ww*(z0^2)/2)*(lf*ll/2))/Iw)*l*tw;
Sw0=sw*tw*(ll*l)/4+sw*tw*0.5*((1/2)^2-x0^2);
Sww0=sww*tw*(ll/2)*x0;
Tau=Vw0*Sw0/Iw+Vww0*Sww0/Iww;
Sigma=M0*x0/Iw+N0/(l*tw);
Sigma11(1,i)=(Sigma/2)+sqrt(((Sigma^2)/4)+Tau^2);
Sigma22(1,i)=(Sigma/2)-sqrt(((Sigma^2)/4)+Tau^2);
A0=2*Tau/Sigma;

if Sigma==0;
    if Tau>0;
        F1(1,i)=45;
        F2(1,i)=135;
    elseif Tau<0;
        F1(1,i)=-45;
        F2(1,i)=-135;
    end
elseif Tau==0;
    if Sigma>0;
        F1(1,i)=0;
        F2(1,i)=90;
    elseif Sigma<0;
        F2(1,i)=180;
        F1(1,i)=-90;
    end
elseif Tau>0;
    if Sigma>0;
        F1(1,i)=0.5*(atan(A0))*180/pi;
        F2(1,i)=F1(1,i)+90;
    elseif Sigma<0
        F1(1,i)=0.5*(atan(A0))*180/pi+90;
        F2(1,i)=F1(1,i)+90;
    end
elseif Tau<0;
    if Sigma>0;
        F1(1,i)=0.5*(atan(A0))*180/pi;
        F2(1,i)=F1(1,i)-90;
    elseif Sigma<0
        F1(1,i)=0.5*(atan(A0))*180/pi-90;
        F2(1,i)=F1(1,i)-90;
    end
end
F1(1,i)=F1(1,i);
F2(1,i)=F2(1,i);

X1(1,i)=x0;
Z1(1,i)=H-z0;
X2(1,i)=x0;
Z2(1,i)=H-z0;
Y1(1,i)=(-ll/2)*lf;
Y2(1,i)=(-ll/2)*lf;
z1=H-z0;
z2=H-z0;

if iii==1
    Sigma1_1(1,i)=Sigma11(1,i);
    Sigma2_1(1,i)=Sigma22(1,i);

```



```

        F1_1(1,i)=F1(1,i);
        X1_1(1,i)=X1(1,i);
        Z1_1(1,i)=Z1(1,i);
        Y1_1(1,i)=Y1(1,i);
        F1_2(1,i)=F2(1,i);
        X1_2(1,i)=X2(1,i);
        Z1_2(1,i)=Z2(1,i);
        Y1_2(1,i)=Y1(1,i);
elseif iii==2
        Sigma2_1(1,i)=Sigma11(1,i);
        Sigma2_2(1,i)=Sigma22(1,i);
        F2_1(1,i)=F1(1,i);
        X2_1(1,i)=X1(1,i);
        Z2_1(1,i)=Z1(1,i);
        Y2_1(1,i)=Y1(1,i);
        F2_2(1,i)=F2(1,i);
        X2_2(1,i)=X2(1,i);
        Z2_2(1,i)=Z2(1,i);
        Y2_2(1,i)=Y1(1,i);
end

if ii==1
        F2_1_A(1,i)=F2_1(1,i);
        F2_2_A(1,i)=F2_2(1,i);
        F1_1_A(1,i)=F1_1(1,i);
        F1_2_A(1,i)=F1_2(1,i);
        X2_1_A(1,i)=X2_1(1,i);
        X2_2_A(1,i)=X2_2(1,i);
        Z2_1_A(1,i)=Z2_1(1,i);
        Z2_2_A(1,i)=Z2_2(1,i);
        Y2_1_A(1,i)=Y2_1(1,i);
        Y2_2_A(1,i)=Y2_2(1,i);
        X1_1_A(1,i)=X1_1(1,i);
        X1_2_A(1,i)=X1_2(1,i);
        Z1_1_A(1,i)=Z1_1(1,i);
        Z1_2_A(1,i)=Z1_2(1,i);
        Y1_1_A(1,i)=Y1_1(1,i);
        Y1_2_A(1,i)=Y1_2(1,i);
        Sigma1_1_A(1,i)=Sigma1_1(1,i);
        Sigma1_2_A(1,i)=Sigma1_2(1,i);
        Sigma2_1_A(1,i)=Sigma2_1(1,i);
        Sigma2_2_A(1,i)=Sigma2_2(1,i);
elseif ii==2
        F2_1_C(1,i)=F2_1(1,i);
        F2_2_C(1,i)=F2_2(1,i);
        F1_1_C(1,i)=F1_1(1,i);
        F1_2_C(1,i)=F1_2(1,i);
        X2_1_C(1,i)=X2_1(1,i);
        X2_2_C(1,i)=X2_2(1,i);
        Z2_1_C(1,i)=Z2_1(1,i);
        Z2_2_C(1,i)=Z2_2(1,i);
        Y2_1_C(1,i)=Y2_2(1,i);
        Y2_2_C(1,i)=Y2_2(1,i);
        X1_1_C(1,i)=X1_1(1,i);
        X1_2_C(1,i)=X1_2(1,i);
        Z1_1_C(1,i)=Z1_1(1,i);
        Z1_2_C(1,i)=Z1_2(1,i);
        Y1_1C(1,i)=Y1_1(1,i);
        Y1_2C(1,i)=Y1_2(1,i);
        Sigma1_1_C(1,i)=Sigma1_1(1,i);
        Sigma1_2_C(1,i)=Sigma1_2(1,i);

```

```

Sigma2_1_C(1,i)=Sigma2_1(1,i);
Sigma2_2_C(1,i)=Sigma2_2(1,i);
elseif ii==3
F2_1_B(1,i)=F2_1(1,i);
F2_2_B(1,i)=F2_2(1,i);
F1_1_B(1,i)=F1_1(1,i);
F1_2_B(1,i)=F1_2(1,i);
Y2_1_B(1,i)=X2_1(1,i);
Y2_2_B(1,i)=X2_2(1,i);
Z2_1_B(1,i)=Z2_1(1,i);
Z2_2_B(1,i)=Z2_2(1,i);
X2_1_B(1,i)=Y2_1(1,i);
X2_2_B(1,i)=Y2_2(1,i);
Y1_1_B(1,i)=X1_1(1,i);
Y1_2_B(1,i)=X1_2(1,i);
Z1_1_B(1,i)=Z1_1(1,i);
Z1_2_B(1,i)=Z1_2(1,i);
X1_1_B(1,i)=Y1_1(1,i);
X1_2_B(1,i)=Y1_2(1,i);
Sigma1_1_B(1,i)=Sigma1_1(1,i);
Sigma1_2_B(1,i)=Sigma1_2(1,i);
Sigma2_1_B(1,i)=Sigma2_1(1,i);
Sigma2_2_B(1,i)=Sigma2_2(1,i);
elseif ii==4
F2_1_D(1,i)=F2_1(1,i);
F2_2_D(1,i)=F2_2(1,i);
F1_1_D(1,i)=F1_1(1,i);
F1_2_D(1,i)=F1_2(1,i);
Y2_1_D(1,i)=X2_1(1,i);
Y2_2_D(1,i)=X2_2(1,i);
Z2_1_D(1,i)=Z2_1(1,i);
Z2_2_D(1,i)=Z2_2(1,i);
X2_1_D(1,i)=Y2_1(1,i);
X2_2_D(1,i)=Y2_2(1,i);
Y1_1_D(1,i)=X1_1(1,i);
Y1_2_D(1,i)=X1_2(1,i);
Z1_1_D(1,i)=Z1_1(1,i);
Z1_2_D(1,i)=Z1_2(1,i);
X1_1_D(1,i)=Y1_1(1,i);
X1_2_D(1,i)=Y1_2(1,i);
Sigma1_1_D(1,i)=Sigma1_1(1,i);
Sigma1_2_D(1,i)=Sigma1_2(1,i);
Sigma2_1_D(1,i)=Sigma2_1(1,i);
Sigma2_2_D(1,i)=Sigma2_2(1,i);
end

```

```

%Now for each starting point the trajectory for the principal
%stress 1, tension, or principal stress 2, compression is
%evaluated.

```

```

%Tension lines construction (Sigma 1)
%(Iterative cycle till the boundary end, based on tangent

```

point

```

%evaluation)
jj=1;
while z1>0
jj=jj+1;
if F1((jj-1),i)==0
X1(jj,i)=X1((jj-1),i);
Z1(jj,i)=Z1((jj-1),i)+H/mm;
elseif F1((jj-1),i)==180

```

```

X1(jj,i)=X1((jj-1),i);
Z1(jj,i)=Z1((jj-1),i)-H/mm;
else
X1(jj,i)=X1((jj-1),i)+f*ff*(l/m);
if F1((jj-1),i)==90;
    Z1(jj,i)=Z1((jj-1),i);
elseif F1((jj-1),i)==-90;
    Z1(jj,i)=Z1((jj-1),i);
elseif F1((jj-1),i)==270;
    Z1(jj,i)=Z1((jj-1),i);
elseif F1((jj-1),i)==-270;
    Z1(jj,i)=Z1((jj-1),i);
else
    Z1(jj,i)=Z1((jj-1),i)+ff*(l/m)/...
        tan(F1((jj-1),i)*pi/180);
end
end

%Boundary check
if Z1(jj,i)<0
    if F1((jj-1),i)==180
        Z1(jj,i)=0;
        X1(jj,i)=X1(jj-1,i);
        z1=0;
        x1=X1(jj,i);
    else
        Z1(jj,i)=0;
        X1(jj,i)=X1(jj-1,i)-f*Z1((jj-1),i)*...
            tan(F1((jj-1),i)*pi/180);
        if X1(jj,i)>1/2
            Z1(jj,i)=Z1((jj-1),i)+f*(1/2-...
                X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=1/2;
            z1=0;
            x1=X1(jj,i);
        elseif X1(jj,i)<-1/2
            Z1(jj,i)=Z1((jj-1),i)+f*((-1/2)-...
                X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=-1/2;
            z1=0;
            x1=X1(jj,i);
        else
            z1=0;
            x1=X1(jj,i);
        end
    end
elseif Z1(jj,i)>H
    if F1((jj-1),i)==0
        Z1(jj,i)=H;
        X1(jj,i)=X1(jj-1,i);
        z1=0;
        x1=X1(jj,i);
    else
        Z1(jj,i)=H;
        X1(jj,i)=X1(jj-1,i)+f*(H-Z1((jj-1),i))*...
            tan(F1((jj-1),i)*pi/180);
        if X1(jj,i)>1/2
            Z1(jj,i)=Z1((jj-1),i)+f*(1/2-...
                X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=1/2;
            z1=0;
        end
    end
end

```

```

        x1=X1(jj,i);
elseif X1(jj,i)<-1/2
    Z1(jj,i)=Z1((jj-1),i)+f*((-1/2)-...
        X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
    X1(jj,i)=-1/2;
    z1=0;
    x1=X1(jj,i);
else
    z1=0;
    x1=X1(jj,i);
end
end
elseif X1(jj,i)>1/2
if F1((jj-1),i)==90
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==-90
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==270
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==-270
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=1/2;
    z1=0;
    x1=X1(jj,i);
else
    X1(jj,i)=1/2;
    Z1(jj,i)=Z1((jj-1),i)+f*(1/2-...
        X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
    z1=0;
    x1=1/2;
end
elseif X1(jj,i)<-1/2
if F1((jj-1),i)==90
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=-1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==-90
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=-1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==270
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=-1/2;
    z1=0;
    x1=X1(jj,i);
elseif F1((jj-1),i)==-270
    Z1(jj,i)=Z1(jj-1,i);
    X1(jj,i)=-1/2;
    z1=0;
    x1=X1(jj,i);

```

```

else
    X1(jj,i)=-1/2;
    Z1(jj,i)=Z1((jj-1),i)+f*((-1/2)-...
        X1((jj-1),i))/tan(F1((jj-1),i)*pi/180);
    z1=0;
    x1=-1/2;
end
else
    Z1(jj,i)=Z1(jj,i);
    X1(jj,i)=X1(jj,i);
    z1=Z1(jj,i);
    x1=X1(jj,i);
end

Y1(jj,i)=(-11/2)*lf;

M1=-w*((H-z1)^2)/2;
V1w=w*(H-z1);
V1ww=ww*(H-z1);
N1=((ww*((H-z1)^2)/2)*(lf*11/2))/Iw*(1*tw);

S1=sw*tww*(11*1)/4+sw*tw*0.5*((1/2)^2-x1^2);
S1w1=sww*tw*(11/2)*x1;
T1=V1w*S1/Iw+V1ww*S1w1/Iww;
Sigma1=M1*x1/Iw+N1/(1*tw);

Sigma11(jj,i)=(Sigma1/2)+sqrt(((Sigma1^2)/4)+T1^2);
A1=2*T1/Sigma1;

if Sigma1==0;
    if T1>0;
        F1(jj,i)=45;
    elseif T1<0;
        F1(jj,i)=-45;
    end
elseif T1==0;
    if Sigma1>0;
        F1(jj,i)=0;
    elseif Sigma1<0;
        F1(jj,i)=-90;
    end
elseif T1>0;
    if Sigma1>0;
        F1(jj,i)=0.5*(atan(A1))*180/pi;
    elseif Sigma1<0
        F1(jj,i)=0.5*(atan(A1))*180/pi+90;
    end
elseif T1<0;
    if Sigma1>0;
        F1(jj,i)=0.5*(atan(A1))*180/pi;
    elseif Sigma1<0
        F1(jj,i)=0.5*(atan(A1))*180/pi-90;
    end
end
end
F1(jj,i)=F1(jj,i);

if iii==1
    Sigma1_1(jj,i)=Sigma11(jj,i);
    F1_1(jj,i)=F1(jj,i);
    X1_1(jj,i)=X1(jj,i);

```

```

        Z1_1(jj,i)=Z1(jj,i);
        Y1_1(jj,i)=Y1(jj,i);
    elseif iii==2
        Sigma2_1(jj,i)=Sigma11(jj,i);
        F2_1(jj,i)=F1(jj,i);
        X2_1(jj,i)=X1(jj,i);
        Z2_1(jj,i)=Z1(jj,i);
        Y2_1(jj,i)=Y1(jj,i);
    end
end

%Compression lines construction (Sigma 2)
jjj=1;
while z2>0
    jjj=jjj+1;
    if F2((jjj-1),i)==0
        X2(jjj,i)=X2((jjj-1),i);
        Z2(jjj,i)=Z2((jjj-1),i)+H/mm;
    elseif F2((jjj-1),i)==180
        X2(jjj,i)=X2((jjj-1),i);
        Z2(jjj,i)=Z2((jjj-1),i)-H/mm;
    else
        X2(jjj,i)=X2((jjj-1),i)+f*ff*(1/m);
        if F2((jjj-1),i)==90;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==-90;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==270;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==-270;
            Z2(jjj,i)=Z2((jjj-1),i);
        else
            Z2(jjj,i)=Z2((jjj-1),i)+ff*(1/m)/...
                tan(F2((jjj-1),i)*pi/180);
        end
    end
end

%Boundary check
if Z2(jjj,i)<0
    if F2((jjj-1),i)==180
        Z2(jjj,i)=0;
        X2(jjj,i)=X2(jjj-1,i);
        z2=0;
        x2=X2(jjj,i);
    else
        Z2(jjj,i)=0;
        X2(jjj,i)=X2(jjj-1,i)-f*Z2((jjj-1),i)*...
            tan(F2((jjj-1),i)*pi/180);
        if X2(jjj,i)>1/2
            Z2(jjj,i)=Z2((jjj-1),i)+f*(1/2-...
                X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
            X2(jjj,i)=1/2;
            z2=0;
            x2=X2(jjj,i);
        elseif X2(jjj,i)<-1/2
            Z2(jjj,i)=Z2((jjj-1),i)+f*((-1/2)-...
                X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
            X2(jjj,i)=-1/2;
            z2=0;
            x2=X2(jjj,i);
        else

```

```

        z2=0;
        x2=X2 (jjj, i);
    end
    end
elseif Z2 (jjj, i)>H
    if F2 ((jjj-1), i)==0
        Z2 (jjj, i)=H;
        X2 (jjj, i)=X2 (jjj-1, i);
        z2=0;
        x2=X2 (jjj, i);
    else
        Z2 (jjj, i)=H;
        X2 (jjj, i)=X2 (jjj-1, i)+f*(H-...
            Z2 ((jjj-1), i))*tan (F2 ((jjj-1), i)*pi/180);
        if X2 (jjj, i)>1/2
            Z2 (jjj, i)=Z2 ((jjj-1), i)+f*(1/2-...
                X2 ((jjj-1), i))/tan (F2 ((jjj-1), i)*pi/180);
            X2 (jjj, i)=1/2;
            z2=0;
            x2=X2 (jjj, i);
        elseif X2 (jjj, i)<-1/2
            Z2 (jjj, i)=Z2 ((jjj-1), i)+f*((-1/2)-...
                X2 ((jjj-1), i))/tan (F2 ((jjj-1), i)*pi/180);
            X2 (jjj, i)=-1/2;
            z2=0;
            x2=X2 (jjj, i);
        else
            z2=0;
            x2=X2 (jjj, i);
        end
    end
end
elseif X2 (jjj, i)>1/2
    if F2 ((jjj-1), i)==90
        Z2 (jjj, i)=Z2 (jjj-1, i);
        X2 (jjj, i)=1/2;
        z2=0;
        x2=X2 (jjj, i);
    elseif F2 ((jjj-1), i)==-90
        Z2 (jjj, i)=Z2 (jjj-1, i);
        X2 (jjj, i)=1/2;
        z2=0;
        x2=X2 (jjj, i);
    elseif F2 ((jjj-1), i)==270
        Z2 (jjj, i)=Z2 (jjj-1, i);
        X2 (jjj, i)=1/2;
        z2=0;
        x2=X2 (jjj, i);
    elseif F2 ((jjj-1), i)==-270
        Z2 (jjj, i)=Z2 (jjj-1, i);
        X2 (jjj, i)=1/2;
        z2=0;
        x2=X2 (jjj, i);
    else
        X2 (jjj, i)=1/2;
        Z2 (jjj, i)=Z2 ((jjj-1), i)+f*(1/2-...
            X2 ((jjj-1), i))/tan (F2 ((jjj-1), i)*pi/180);
        z2=0;
        x2=1/2;
    end
end
elseif X2 (jjj, i)<-1/2
    if F2 ((jjj-1), i)==90

```

```

        Z2 (jjj, i) = Z2 (jjj-1, i);
        X2 (jjj, i) = -1/2;
        z2 = 0;
        x2 = X2 (jjj, i);
    elseif F2 ((jjj-1), i) == -90
        Z2 (jjj, i) = Z2 (jjj-1, i);
        X2 (jjj, i) = -1/2;
        z2 = 0;
        x2 = X2 (jjj, i);
    elseif F2 ((jjj-1), i) == 270
        Z2 (jjj, i) = Z2 (jjj-1, i);
        X2 (jjj, i) = -1/2;
        z2 = 0;
        x2 = X2 (jjj, i);
    elseif F2 ((jjj-1), i) == -270
        Z2 (jjj, i) = Z2 (jjj-1, i);
        X2 (jjj, i) = -1/2;
        z2 = 0;
        x2 = X2 (jjj, i);
    else
        X2 (jjj, i) = -1/2;
        Z2 (jjj, i) = Z2 ((jjj-1), i) + f * ((-1/2) - ...
            X2 ((jjj-1), i)) / tan (F2 ((jjj-1), i) * pi / 180);
        z2 = 0;
        x2 = -1/2;
    end
else
    Z2 (jjj, i) = Z2 (jjj, i);
    X2 (jjj, i) = X2 (jjj, i);
    z2 = Z2 (jjj, i);
    x2 = X2 (jjj, i);
end

Y2 (jjj, i) = (-11/2) * lf;

M2 = -w * ((H-z2) ^ 2) / 2;
V2w = w * (H-z2);
V2ww = ww * (H-z2);
N2 = ((ww * ((H-z2) ^ 2) / 2) * (lf * 11/2)) / Iw * (1 * tw);

Sw2 = sw * tww * (11 * l) / 4 + sw * tw * 0.5 * ((1/2) ^ 2 - x2 ^ 2);
Sww2 = sww * tw * (11/2) * x2;
Tau2 = V2w * Sw2 / Iw + V2ww * Sww2 / Iww;
Sigma2 = M2 * x2 / Iw + N2 / (1 * tw);

Sigma22 (jjj, i) = (Sigma2 / 2) - sqrt (((Sigma2 ^ 2) / 4) + Tau2 ^ 2);
A2 = 2 * Tau2 / Sigma2;

if Sigma2 == 0;
    if Tau2 > 0;
        F2 (jjj, i) = 135;
    elseif Tau2 < 0;
        F2 (jjj, i) = -135;
    end
elseif Tau2 == 0;
    if Sigma2 > 0;
        F2 (jjj, i) = 90;
    elseif Sigma2 < 0;
        F2 (jjj, i) = 180;
    end
end

```



```

elseif Tau2>0;
    if Sigma2>0;
        F2(jjj,i)=0.5*(atan(A2))*180/pi+90;
    elseif Sigma2<0
        F2(jjj,i)=0.5*(atan(A2))*180/pi+90+90;
    end
elseif Tau2<0;
    if Sigma2>0;
        F2(jjj,i)=0.5*(atan(A2))*180/pi-90;
    elseif Sigma2<0
        F2(jjj,i)=0.5*(atan(A2))*180/pi-90-90;
    end
end
F2(jjj,i)=F2(jjj,i);

    if iii==1
        Sigma1_2(jjj,i)=Sigma22(jjj,i);
        F1_2(jjj,i)=F2(jjj,i);
        X1_2(jjj,i)=X2(jjj,i);
        Z1_2(jjj,i)=Z2(jjj,i);
        Y1_2(jjj,i)=Y2(jjj,i);
    elseif iii==2
        Sigma2_2(jjj,i)=Sigma22(jjj,i);
        F2_2(jjj,i)=F2(jjj,i);
        X2_2(jjj,i)=X2(jjj,i);
        Z2_2(jjj,i)=Z2(jjj,i);
        Y2_2(jjj,i)=Y2(jjj,i);
    end
end
end
end

%Face components selection
if ii==1
    F2_1_A=F2_1;
    F2_2_A=F2_2;
    F1_1_A=F1_1;
    F1_2_A=F1_2;
    X2_1_A=X2_1;
    X2_2_A=X2_2;
    Z2_1_A=Z2_1;
    Z2_2_A=Z2_2;
    X1_1_A=X1_1;
    X1_2_A=X1_2;
    Z1_1_A=Z1_1;
    Z1_2_A=Z1_2;
    Y1_1_A=Y1_1;
    Y1_2_A=Y1_2;
    Y2_1_A=Y2_1;
    Y2_2_A=Y2_2;
    Sigma1_1_A=Sigma1_1;
    Sigma1_2_A=Sigma1_2;
    Sigma2_1_A=Sigma2_1;
    Sigma2_2_A=Sigma2_2;
elseif ii==2
    F2_1_C=F2_1;
    F2_2_C=F2_2;
    F1_1_C=F1_1;
    F1_2_C=F1_2;
    X2_1_C=X2_1;
    X2_2_C=X2_2;

```

```

Z2_1_C=Z2_1;
Z2_2_C=Z2_2;
X1_1_C=X1_1;
X1_2_C=X1_2;
Z1_1_C=Z1_1;
Z1_2_C=Z1_2;
Y1_1_C=Y1_1;
Y1_2_C=Y1_2;
Y2_1_C=Y2_1;
Y2_2_C=Y2_2;
Sigma1_1_C=Sigma1_1;
Sigma1_2_C=Sigma1_2;
Sigma2_1_C=Sigma2_1;
Sigma2_2_C=Sigma2_2;
elseif ii==3
F2_1_B=F2_1;
F2_2_B=F2_2;
F1_1_B=F1_1;
F1_2_B=F1_2;
Y2_1_B=X2_1;
Y2_2_B=X2_2;
Z2_1_B=Z2_1;
Z2_2_B=Z2_2;
Y1_1_B=X1_1;
Y1_2_B=X1_2;
Z1_1_B=Z1_1;
Z1_2_B=Z1_2;
X1_1_B=Y1_1;
X1_2_B=Y1_2;
X2_1_B=Y2_1;
X2_2_B=Y2_2;
Sigma1_1_B=Sigma1_1;
Sigma1_2_B=Sigma1_2;
Sigma2_1_B=Sigma2_1;
Sigma2_2_B=Sigma2_2;
elseif ii==4
F2_1_D=F2_1;
F2_2_D=F2_2;
F1_1_D=F1_1;
F1_2_D=F1_2;
Y2_1_D=X2_1;
Y2_2_D=X2_2;
Z2_1_D=Z2_1;
Z2_2_D=Z2_2;
Y1_1_D=X1_1;
Y1_2_D=X1_2;
Z1_1_D=Z1_1;
Z1_2_D=Z1_2;
X1_1_D=Y1_1;
X1_2_D=Y1_2;
X2_1_D=Y2_1;
X2_2_D=Y2_2;
Sigma1_1_D=Sigma1_1;
Sigma1_2_D=Sigma1_2;
Sigma2_1_D=Sigma2_1;
Sigma2_2_D=Sigma2_2;
end
end

```

```

%% Model representation

% Principal stress trajectories
figure
axis equal

%Faces construction
patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],...
      [0,0,0,0],[0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+1])
xticks([-a/2:5:a/2])
yticks([-b/2:5:b/2])
zticks([0:5:H])
title('Principal stress Trajectories')
xlabel('Section "a" [m]')
ylabel('Section "b" [m]')
zlabel('Height [m]')

hold on

%Axis
quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],...
        [0;0;0],'k')

%Trajectories
%(for a clear representation, not all the lines must be represented for
%each face, so it is suggested to check the result and choose which lines
%set should be closed.)
line(X1_1_A,Y1_1_A,Z1_1_A,'Color','k')
line(X1_2_A,Y1_2_A,Z1_2_A,'Color','k')
line(X2_1_A,Y2_1_A,Z2_1_A,'Color','k')
line(X2_2_A,Y2_2_A,Z2_2_A,'Color','k')

line(X1_1_C,Y1_1_C,Z1_1_C,'Color','k')
line(X1_2_C,Y1_2_C,Z1_2_C,'Color','k')
line(X2_1_C,Y2_1_C,Z2_1_C,'Color','k')
line(X2_2_C,Y2_2_C,Z2_2_C,'Color','k')

line(X1_1_B,Y1_1_B,Z1_1_B,'Color','k')
line(X1_2_B,Y1_2_B,Z1_2_B,'Color','k')
line(X2_1_B,Y2_1_B,Z2_1_B,'Color','k')
line(X2_2_B,Y2_2_B,Z2_2_B,'Color','k')

line(X1_1_D,Y1_1_D,Z1_1_D,'Color','k')
line(X1_2_D,Y1_2_D,Z1_2_D,'Color','k')
line(X2_1_D,Y2_1_D,Z2_1_D,'Color','k')
line(X2_2_D,Y2_2_D,Z2_2_D,'Color','k')

%Principal stresses map

% Tension state
figure
axis equal

```

```

patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],[0,0,0,0],...
    [0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+1])
xticks([-a/2:5:a/2])
yticks([-b/2:5:b/2])
zticks([0:5:H])
title('Principal stress direction - Tension state')
xlabel('Section "a" [m]')
ylabel('Section "b" [m]')
zlabel('Height [m]')

hold on

quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],...
    [0;0;0],'k')
quiver3(X_A,Y_A,Z_A,CC1_A,O_A,SS1_A,0.5,'b')
quiver3(X_C,Y_C,Z_C,CC1_C,O_C,SS1_C,0.5,'b')
quiver3(X_B,Y_B,Z_B,O_B,CC1_B,SS1_B,0.5,'b')
quiver3(X_D,Y_D,Z_D,O_D,CC1_D,SS1_D,0.5,'b')

line(X1_1_A,Y1_1_A,Z1_1_A,'Color','k')
line(X2_1_A,Y2_1_A,Z2_1_A,'Color','k')

line(X1_1_C,Y1_1_C,Z1_1_C,'Color','k')
line(X2_1_C,Y2_1_C,Z2_1_C,'Color','k')

line(X1_1_B,Y1_1_B,Z1_1_B,'Color','k')
line(X2_1_B,Y2_1_B,Z2_1_B,'Color','k')

line(X1_1_D,Y1_1_D,Z1_1_D,'Color','k')
line(X2_1_D,Y2_1_D,Z2_1_D,'Color','k')

% Compression state
figure
axis equal
patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],[0,0,0,0],...
    [0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+1])
xticks([-a/2:5:a/2])
yticks([-b/2:5:b/2])
zticks([0:5:H])
title('Principal stress direction - Compression state')
xlabel('Section [m]')
ylabel('Section [m]')
zlabel('Height [m]')

```

```

hold on

quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],...
        [0;0;0], 'k')
quiver3(X_A,Y_A,Z_A,CC2_A,O_A,SS2_A,0.5, 'r')
quiver3(X_C,Y_C,Z_C,CC2_C,O_C,SS2_C,0.5, 'r')
quiver3(X_B,Y_B,Z_B,O_B,CC2_B,SS2_B,0.5, 'r')
quiver3(X_D,Y_D,Z_D,O_D,CC2_D,SS2_D,0.5, 'r')

line(X1_2_A,Y1_2_A,Z1_2_A, 'Color', 'k')
line(X2_2_A,Y2_2_A,Z2_2_A, 'Color', 'k')

line(X1_2_C,Y1_2_C,Z1_2_C, 'Color', 'k')
line(X2_2_C,Y2_2_C,Z2_2_C, 'Color', 'k')

line(X1_2_B,Y1_2_B,Z1_2_B, 'Color', 'k')
line(X2_2_B,Y2_2_B,Z2_2_B, 'Color', 'k')

line(X1_2_D,Y1_2_D,Z1_2_D, 'Color', 'k')
line(X2_2_D,Y2_2_D,Z2_2_D, 'Color', 'k')

```

## 10.2. MatLab code for principal stresses trajectories with shear-lag effect

The previous code has been modified considering the shear-lag effect as it was exposed by Reissnes in his works [13].

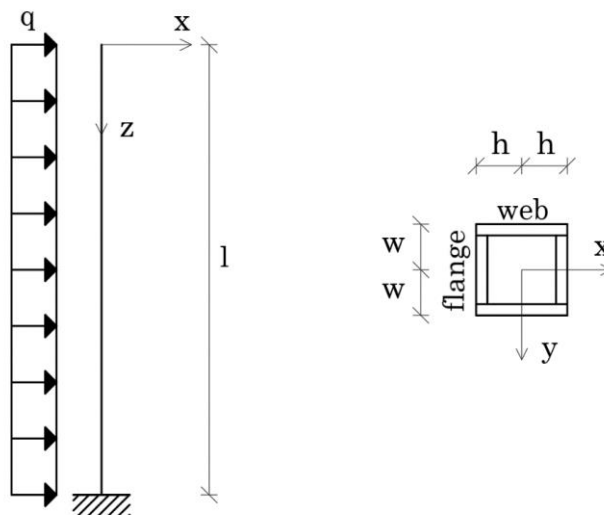


Figure 10.1 Cantilever system for shear-lag computation

Briefly, following his papers, for the cantilever system analysed till now, with a uniform distributed load orthogonal to one face, the kinematic fields for webs and flanges are:

Webs kinematic field:

$$\begin{cases} (x) & v(x, y, z) = v(z) \\ (z) & u(x, y, z) = -v'(z) x \end{cases}$$

Flanges kinematic field:

$$\begin{cases} (x) & v(x, y, z) = v(z) \\ (z) & u(x, y, z) = \pm h[-v'(z) - U(z)\Psi(y)] \end{cases} \quad \text{and} \quad \Psi(y) = 1 - \left(\frac{y}{w}\right)^2$$

Where  $\Psi$  define the parabolic profile of the stress state. Then, the moment of inertia given by web system and flange system are defined as:

The procedure continues extend as a potential energy problem defining the quantities given by webs and flanges and the external work given by the loads:

Load external work:

$$\begin{aligned} \Pi_q &= - \int_l qv(z)dz \quad \text{where} \quad -q = \frac{d^2M(z)}{dz^2} \\ &= \int_l \frac{d^2M(z)}{dz^2} v(z)dz = \dots = \int_l M(z)v''(z)dz \end{aligned}$$

Webs potential energy:

$$\Pi_w = \frac{1}{2} \int_l EI_w v''^2(z) dz$$

Flanges potential energy:

$$\Pi_f = \frac{1}{2} \iiint 2[E\varepsilon_z^2(x, y, z) + G\gamma^2(x, y, z)] dx dy dz$$

Where

$$\begin{cases} \varepsilon_z = \frac{\partial u}{\partial z} = \bar{\tau}h[-v^{II}(z) - U^I(z)\Psi(y)] \\ \gamma = \frac{\partial u}{\partial y} = \bar{\tau}h \cdot 2 \frac{y}{w} \end{cases}$$

So

$$\begin{aligned} &= \frac{1}{2} \iiint 2 \left\{ Eh^2 \left[ v^{II^2}(z) - U^{I^2}(z)\Psi^2(y) + 2v^{II}(z)U^I(z)\Psi(y) \right] + G \left( \frac{2hy}{w^2} \right)^2 \right\} dx dy dz \\ &= \dots \\ &= \frac{1}{2} \int_l EI_f \left[ v^{II^2}(z) + \frac{4}{3}v^{II}(z)U^I(z) + \frac{8}{15}U^{I^2}(z) \right] + GI \frac{1}{3} \left( \frac{2U(z)}{w} \right)^2 dz \end{aligned}$$

Total potential energy:

$$\Pi = \Pi_q + \Pi_w + \Pi_f$$

Finding the minimum to solve the problem, the derivative on  $z$  is posed equal to zero:

$$\begin{aligned} \partial \Pi = 0 &= \frac{1}{2} \int_l EI v^{II} \delta v^{II} + \frac{2}{3} EI_f v^{II} \delta U^I + \frac{2}{3} EI_f U^I \delta v^{II} + \frac{8}{15} EI_f U^I \delta U^I + \frac{4}{3} GI_f \frac{U}{w^2} \delta U \\ &+ M \delta v^{II} dz = \dots \end{aligned}$$

Getting the equations:

$$\begin{cases} v^{II} + \frac{2I_f}{3l} U^I + \frac{M}{EI} = 0 & \forall \delta v^{II} \\ v^{III} + \frac{4}{5} U^{II} - 2 \frac{G}{E} \frac{U}{w^2} = 0 & \forall \delta U \end{cases} \quad \text{boundary conditions: } \begin{cases} \left( \frac{d^2 M}{dz^2} \right) \delta v^I \Big|_0^l = 0 \\ \left( v^{II} + \frac{4}{5} U^I \right) \delta U \Big|_0^l = 0 \\ \left( \frac{dM}{dz} \right) \delta v^{II} \Big|_0^l = 0 \end{cases}$$

Now, finding  $U'$  from the first equation and computing  $U''$ , the primitive  $U$  is determined from the second equation depending on  $z$  only. Now  $U'$  is computed again from the new expression and substitute again in the first equation, obtaining then the single expression:

$$v^{II} - \frac{1}{k^2} v^{IV} = -\frac{M}{EI} + \frac{n}{k^2} \frac{M^{II}}{EI} \quad \text{where} \quad \begin{cases} n = \frac{1}{1 - \frac{5I_f}{6I}} \\ k = \frac{1}{w} \sqrt{\frac{5}{2} n \frac{G}{E}} \end{cases}$$

From this expression, the second derivative of the transverse displacement should be computed for every load condition, because the only load dependent parameter is the moment  $M$ . So, for a given load condition, if an expression of the moment, depending on  $z$  coordinate is obtained the differential equation is solved.

For this purpose, a uniform distributed load is considered and starting with the  $z$  coordinate from the top the moment takes the expression:

$$M(z) = M_0 \left( \frac{z}{l} \right)^2 \quad \text{where} \quad M_0 = q \frac{l^2}{2}$$

And

$$v^{II} - \frac{1}{k^2} v^{IV} = -\frac{M_0}{EI} \left[ \left( \frac{z}{l} \right)^2 - \frac{2(n-1)}{(kl)^2} \right]$$

The differential solution:



$$v^{II} = \frac{M_0}{EI} \left[ C_1 \sinh(kz) + C_2 \cosh(kz) - \left(\frac{z}{l}\right)^2 + \frac{2(n-1)}{(kl)^2} \right]$$

Written also as:

$$v^{III} = \frac{M_0}{EI} \left[ C_1 k \cosh(kz) + C_2 k \sinh(kz) - 2 \frac{z}{l^2} \right]$$

The reason why two derivatives are considered depends on the boundary conditions used to determine the constants. For the free edge the second derivative is involved, while for the clamped edge the third one is required.

Free edge:

$$v^{II}|_0 = 0 = \frac{M_0}{EI} \left[ C_1 \sinh(k0) + C_2 \cosh(k0) - \left(\frac{0}{l}\right)^2 + \frac{2n}{(kl)^2} \right]$$

and 
$$C_2 = -\frac{2(n-1)}{(kl)^2}$$

Clamped edge:

$$v^{III}|_0 = \frac{M_0}{EI} \left[ C_1 k \cosh(kl) - \frac{2(n-1)}{(kl)^2} k \sinh(kl) - 2 \frac{1}{l} \right] = -n \frac{M_0}{EI} 2 \frac{1}{l}$$

And 
$$C_1 = \frac{2(n-1)}{(kl)^2} \left[ \frac{\sinh(lk) - kl}{\cosh(lk)} \right]$$

Finally, the expression for the rotation  $v^{II}$  is given and is equal to:

$$v^{II} = -\frac{M_0}{EI} \left\{ \left(\frac{z}{l}\right)^2 + \frac{2(n-1)}{(kl)^2} \left[ (\cosh(kz) - 1) - \frac{\sinh(lk) - kl}{\cosh(lk)} \sinh(kz) \right] \right\}$$

From this the normal stress state is computed by linear elastic relations:

$$U^I = -\frac{3}{2} \frac{I}{I_f} \left[ v^{II} + M_0 \left( \frac{z}{l} \right)^2 \frac{1}{EI} \right]$$

So:

$$\sigma_{web} = -Exv^{II} \quad \text{and} \quad \sigma_{flange} = \pm Eh[v^{II} + U^I \Psi]$$

Now the stress state given by shear-lag contribution has been achieved, so the code should be modified and implemented.

The new code looks like:

```
%% Data

% Geometry
a=25;
b=25;
H=200;
tb=1;
ta=1;
faces=4;

% Wind load
W=10000;
teta_x=0;
w_x=W*cos(teta_x*pi/180);
w_y=W*sin(teta_x*pi/180);

% Grid subdivision for mapping
r=200;
p=25;
q=25;

% Inertia moments
%(Comparing with the other code here a distinction between flanges and
%webs must be considered following Reissner formulations.)
Iwa=2*ta*((a-2*tb)^3)/12;
Ifb=2*tb*b*((a-tb)/2)^2;
Iwb=2*tb*((b-2*ta)^3)/12;
Ifa=2*ta*a*((b-ta)/2)^2;

Ia=2*ta*((a-2*tb)^3)/12+2*tb*b*((a-tb)/2)^2;
Ib=2*tb*((b-2*ta)^3)/12+2*ta*a*((b-ta)/2)^2;

% Trajectories parameters
m=10000;
mm=10000;
num=4;
e=(H/a)*num+num;
```

```

% Material properties
% (In this case those parameters are required)
% Steel
E=0.21;    %[N/m^2]
G=0.081;   %[N/m^2]

%% Faces

%Face A
M_A=zeros([ (r+1) 1]);
Vw_A=zeros([ (r+1) 1]);
Vww_A=zeros([ (r+1) 1]);
N_A=zeros([ (r+1) 1]);

Sw_A=zeros([1 (p+1)]);
Sww_A=zeros([1 (p+1)]);

Sigma_A=zeros([ (r+1) (p+1)]);
Tau_A=zeros([ (r+1) (p+1)]);
Sigma1_A=zeros([ (r+1) (p+1)]);
Sigma2_A=zeros([ (r+1) (p+1)]);
Fi1_A=zeros([ (r+1) (p+1)]);
Fi2_A=zeros([ (r+1) (p+1)]);

%Face C
M_C=zeros([ (r+1) 1]);
Vw_C=zeros([ (r+1) 1]);
Vww_C=zeros([ (r+1) 1]);
N_C=zeros([ (r+1) 1]);

Sw_C=zeros([1 (p+1)]);
Sww_C=zeros([1 (p+1)]);

Sigma_C=zeros([ (r+1) (p+1)]);
Tau_C=zeros([ (r+1) (p+1)]);
Sigma1_C=zeros([ (r+1) (p+1)]);
Sigma2_C=zeros([ (r+1) (p+1)]);
Fi1_C=zeros([ (r+1) (p+1)]);
Fi2_C=zeros([ (r+1) (p+1)]);

%Face B
M_B=zeros([ (r+1) 1]);
Vw_B=zeros([ (r+1) 1]);
Vww_B=zeros([ (r+1) 1]);
N_B=zeros([ (r+1) 1]);

Sw_B=zeros([1 (q+1)]);
Sww_B=zeros([1 (q+1)]);

Sigma_B=zeros([ (r+1) (q+1)]);
Tau_B=zeros([ (r+1) (q+1)]);
Sigma1_B=zeros([ (r+1) (q+1)]);
Sigma2_B=zeros([ (r+1) (q+1)]);
Fi1_B=zeros([ (r+1) (q+1)]);
Fi2_B=zeros([ (r+1) (q+1)]);

%Face D
M_D=zeros([ (r+1) 1]);
Vw_D=zeros([ (r+1) 1]);

```

```

Vww_D=zeros ([ (r+1) 1]);
N_D=zeros ([ (r+1) 1]);

Sw_D=zeros ([1 (q+1)]);
Sww_D=zeros ([1 (q+1)]);

Sigma_D=zeros ([ (r+1) (q+1)]);
Tau_D=zeros ([ (r+1) (q+1)]);
Sigma1_D=zeros ([ (r+1) (q+1)]);
Sigma2_D=zeros ([ (r+1) (q+1)]);
Fi1_D=zeros ([ (r+1) (q+1)]);
Fi2_D=zeros ([ (r+1) (q+1)]);

%(With respect to the other code here have been added other parameters
which
%come from Reissner formulation, as n and k)
for ii=1:1:faces
    if ii==1;
        tw=ta;
        tww=tb;
        w=w_x;
        ww=w_y;
        sw=1;
        sww=1;
        f=1;
        lf=1;
        l=a;
        ll=b;
        u=p;
        Ifw=Ifb;
        Ifww=Ifa;
        Iw=Ia;
        Iww=Ib;
        nw=1/(1-(5/6)*(Ifw/Iw));
        kw=1/(ll/2)*sqrt((5/2)*nw*G/E);
        nww=1/(1-(5/6)*(Ifww/Iww));
        kww=1/(l/2)*sqrt((5/2)*nww*G/E);
    elseif ii==2;
        tw=ta;
        tww=tb;
        w=w_x;
        ww=w_y;
        sw=-1;
        sww=1;
        f=-1;
        lf=-1;
        l=a;
        ll=b;
        u=p;
        Ifw=Ifb;
        Ifww=Ifa;
        Iw=Ia;
        Iww=Ib;
        nw=1/(1-(5/6)*(Ifw/Iw));
        kw=1/(-ll/2)*sqrt((5/2)*nw*G/E);
        nww=1/(1-(5/6)*(Ifww/Iww));
        kww=1/(l/2)*sqrt((5/2)*nww*G/E);
    elseif ii==3;
        tw=tb;
        tww=ta;
        w=w_y;

```

```

ww=w_x;
sw=1;
sww=-1;
f=1;
lf=-1;
l=b;
ll=a;
u=q;
Ifw=Ifa;
Ifww=Ifb;
Iw=Ib;
Iww=Ia;
nw=1/(1-(5/6)*(Ifw/Iw));
kw=1/(ll/2)*sqrt((5/2)*nw*G/E);
nww=1/(1-(5/6)*(Ifww/Iww));
kww=1/(l/2)*sqrt((5/2)*nww*G/E);
elseif ii==4;
tw=tb;
tww=ta;
ww=w_x;
sw=-1;
sww=-1;
f=-1;
lf=1;
l=b;
ll=a;
u=q;
Ifw=Ifb;
Ifww=Ifa;
Iw=Ib;
Iww=Ia;
nw=1/(1-(5/6)*(Ifw/Iw));
kw=1/(-ll/2)*sqrt((5/2)*nw*G/E);
nww=1/(1-(5/6)*(Ifww/Iww));
kww=1/(l/2)*sqrt((5/2)*nw*G/E);
end

%Grid definition for the principal stresses mapping
Sw=zeros([1 (u+1)]);
Sww=zeros([1 (u+1)]);

Sigma=zeros([(r+1) (u+1)]);
Tau=zeros([(r+1) (u+1)]);
Sigma1=zeros([(r+1) (u+1)]);
Sigma2=zeros([(r+1) (u+1)]);
Fi1=zeros([(r+1) (u+1)]);
Fi2=zeros([(r+1) (u+1)]);

%Grid
[X,Z]=meshgrid(-f*1/2:f*1/u:f*1/2,H:-H/r:0);
Y=(-lf*ll/2)*ones([(r+1) (u+1)]);

%Z coordinates
for j=1:1:r+1;
if j==1;
z=0.0001;
else
z=(H/r)*(j-1);
end

```

```

%Internal actions
Mw0=-w*(H^2)/2;
Mww0=-ww*(H^2)/2;
Mw(j)=Mw0*(z/H)^2;
Mww(j)=Mww0*(z/H)^2;
Vw(j)=w*z;
Vww(j)=ww*z;
N(j)=((ww*(z^2)/2)*(lf*ll/2))/Iw)*1*tw;

%X or Y coordinates for the give Z coordinate
%(It must be noted how the stresses are computed starting from
%deflection, v, and its derivatives, v2.)
for i=1:l:u+1;
    x=f*(-(l/2)+(l/u)*(i-1));
    Sw(i)=sw*tww*(ll*1)/4+sw*tw*0.5*((l/2)^2-x^2);
    Sww(i)=sww*tw*(ll/2)*x;
    v2_w=-Mw0/(E*Iw)*(z/H)^2+(cosh(kw*z)-1-((sinh(kw*H)-kw*H)...
        /cosh(kw*H))*sinh(kw*z))*((2*(nw-1))/((kw*H)^2));
    v2_ww=-Mww0/(E*Iww)*(z/H)^2+(cosh(kww*z)...
        -1-((sinh(kww*H)-kww*H)/cosh(kww*H))*...
        sinh(kww*z))*((2*(nww-1))/((kww*H)^2));
    Uww=(-3/2)*(Iww/Iww)*(v2_ww+Mww(j)/(E*Iww));
    Tau(j,i)=Vw(j)*Sw(i)/Iw+Vww(j)*Sww(i)/Iww;
    Sigma(j,i)=-E*x*v2_w-(lf*(-ll)/2)*E*(v2_ww+Uww*(1-
(2*x/l)^2));
    Sigma1(j,i)=Sigma(j,i)/2+sqrt((Sigma(j,i)^2)/4+Tau(j,i)^2);
    Sigma2(j,i)=Sigma(j,i)/2-sqrt((Sigma(j,i)^2)/4+Tau(j,i)^2);
    A=2*Tau(j,i)/Sigma(j,i);

%Angle evaluation procedure
if Sigma(j,i)==0;
    if Tau(j,i)>0;
        Fi1(j,i)=45;
        Fi2(j,i)=135;
    elseif Tau(j,i)<0;
        Fi1(j,i)=-45;
        Fi2(j,i)=-135;
    end
elseif Tau(j,i)==0;
    if Sigma(j,i)>0;
        Fi1(j,i)=0;
        Fi2(j,i)=90;
    elseif Sigma(j,i)<0;
        Fi2(j,i)=180;
        Fi1(j,i)=-90;
    end
elseif Tau(j,i)>0;
    if Sigma(j,i)>0;
        Fi1(j,i)=0.5*(atan(A))*180/pi;
        Fi2(j,i)=Fi1(j,i)+90;
    elseif Sigma(j,i)<0
        Fi1(j,i)=0.5*(atan(A))*180/pi+90;
        Fi2(j,i)=Fi1(j,i)+90;
    end
elseif Tau(j,i)<0;
    if Sigma(j,i)>0;
        Fi1(j,i)=0.5*(atan(A))*180/pi;
        Fi2(j,i)=Fi1(j,i)-90;
    elseif Sigma(j,i)<0
        Fi1(j,i)=0.5*(atan(A))*180/pi-90;
        Fi2(j,i)=Fi1(j,i)-90;
    end
end
end

```

```

end
end
Fi1(j,i)=Fi1(j,i);
Fi2(j,i)=Fi2(j,i);

%Face matrices assignment
if ii==1
    Mw_A(j)=Mw(j);
    Mww_A(j)=Mww(j);
    Vw_A(j)=Vw(j);
    N_A(j)=N(j);
    Sw_A(i)=Sw(i);
    Tau_A(j,i)=Tau(j,i);
    Sigma_A(j,i)=Sigma(j,i);
    Sigma1_A(j,i)=Sigma1(j,i);
    Sigma2_A(j,i)=Sigma2(j,i);
    X_A=X;
    Y_A=Y;
    Z_A=Z;
    Fi1_A(j,i)=Fi1(j,i);
    Fi2_A(j,i)=Fi2(j,i);
elseif ii==2
    Mw_C(j)=Mw(j);
    Mww_C(j)=Mww(j);
    Vw_C(j)=Vw(j);
    N_C(j)=N(j);
    Sw_C(i)=Sw(i);
    Tau_C(j,i)=Tau(j,i);
    Sigma_C(j,i)=Sigma(j,i);
    Sigma1_C(j,i)=Sigma1(j,i);
    Sigma2_C(j,i)=Sigma2(j,i);
    X_C=X;
    Y_C=Y;
    Z_C=Z;
    Fi1_C(j,i)=Fi1(j,i);
    Fi2_C(j,i)=Fi2(j,i);
elseif ii==3
    Mw_B(j)=Mw(j);
    Mww_B(j)=Mww(j);
    Vw_B(j)=Vw(j);
    N_B(j)=N(j);
    Sw_B(i)=Sw(i);
    Tau_B(j,i)=Tau(j,i);
    Sigma_B(j,i)=Sigma(j,i);
    Sigma1_B(j,i)=Sigma1(j,i);
    Sigma2_B(j,i)=Sigma2(j,i);
    X_B=Y;
    Y_B=X;
    Z_B=Z;
    Fi1_B(j,i)=Fi1(j,i);
    Fi2_B(j,i)=Fi2(j,i);
elseif ii==4
    Mw_D(j)=Mw(j);
    Mww_D(j)=Mww(j);
    Vw_D(j)=Vw(j);
    N_D(j)=N(j);
    Sw_D(i)=Sw(i);
    Tau_D(j,i)=Tau(j,i);
    Sigma_D(j,i)=Sigma(j,i);
    Sigma1_D(j,i)=Sigma1(j,i);
    Sigma2_D(j,i)=Sigma2(j,i);

```

```

        X_D=Y;
        Y_D=X;
        Z_D=Z;
        Fi1_D(j,i)=Fi1(j,i);
        Fi2_D(j,i)=Fi2(j,i);
    end
end
end

%Unit vector definition for "quiver" components
CFi1=cos(pi/180*Fi1);
SFi1=sin(pi/180*Fi1);
CFi2=cos(pi/180*Fi2);
SFi2=sin(pi/180*Fi2);
R=ones(r+1,u+1);
CC1=sw*SFi1.*R;
SS1=CFi1.*R;
CC2=sw*SFi2.*R;
SS2=CFi2.*R;
O=zeros([r+1 u+1]);

if ii==1
    CC1_A=CC1;
    SS1_A=SS1;
    CC2_A=CC2;
    SS2_A=SS2;
    O_A=O;
elseif ii==2
    CC1_C=CC1;
    SS1_C=SS1;
    CC2_C=CC2;
    SS2_C=SS2;
    O_C=O;
elseif ii==3
    CC1_B=CC1;
    SS1_B=SS1;
    CC2_B=CC2;
    SS2_B=SS2;
    O_B=O;
elseif ii==4
    CC1_D=CC1;
    SS1_D=SS1;
    CC2_D=CC2;
    SS2_D=SS2;
    O_D=O;
end

%Trajectories

%Point coordinates
F2_1=NaN([m+1 e]);
F2_2=NaN([m+1 e]);
F1_1=NaN([m+1 e]);
F1_2=NaN([m+1 e]);
X2_1=NaN([m+1 e]);
X2_2=NaN([m+1 e]);
Z2_1=NaN([m+1 e]);
Z2_2=NaN([m+1 e]);
Y2_1=NaN([m+1 e]);
Y2_2=NaN([m+1 e]);
X1_1=NaN([m+1 e]);

```



```

X1_2=NaN([ (m+1) (e) ]);
Z1_1=NaN([ (m+1) (e) ]);
Z1_2=NaN([ (m+1) (e) ]);
Y1_1=NaN([ (m+1) (e) ]);
Y1_2=NaN([ (m+1) (e) ]);

Sigma1_1=NaN([ (m+1) (e) ]);
Sigma1_2=NaN([ (m+1) (e) ]);
Sigma2_1=NaN([ (m+1) (e) ]);
Sigma2_2=NaN([ (m+1) (e) ]);

%Fake components for the evaluation
F1=NaN([ (m+1) (e) ]);
F2=NaN([ (m+1) (e) ]);
X1=NaN([ (m+1) (e) ]);
X2=NaN([ (m+1) (e) ]);
Z1=NaN([ (m+1) (e) ]);
Z2=NaN([ (m+1) (e) ]);
Y1=NaN([ (m+1) (e) ]);
Y2=NaN([ (m+1) (e) ]);
Sigma11=NaN([ (m+1) (e) ]);
Sigma22=NaN([ (m+1) (e) ]);

%Side starting point
for iii=1:1:2
    if iii==1
        x00=(-1/2)*f;
        ff=1;
    elseif iii==2
        x00=(1/2)*f;
        ff=-1;
    end

%Starting point selection
for i=1:1:e;
    if i<num+1
        z0=0.0001;
        x0=-x00-i*(1/num)*ff*f;
    else
        z0=(H/(e-num))*(i-num);
        x0=x00;
    end

Mw0=-w*(H^2)/2;
Mww0=-ww*(H^2)/2;
Mw=Mw0*(z0/H)^2;
Mww=Mww0*(z0/H)^2;
Vw=w*z0;
Vww=ww*z0;
N=((ww*(z0^2)/2)*(lf*ll/2))/Iw)*1*tw;
Sw0=sw*tww*(ll*1)/4+sw*tw*0.5*((1/2)^2-x0^2);
Sww0=sww*tw*(ll/2)*x0;
v2_w=-Mw0/(E*Iw)*((z0/H)^2+(cosh(kw*z0)-1-((sinh(kw*H)-
kw*H)...
/cosh(kw*H))*sinh(kw*z0))*((2*(nw-1))/((kw*H)^2)));
v2_ww=-Mww0/(E*Iww)*((z0/H)^2+(cosh(kww*z0)-1-...
((sinh(kww*H)-kww*H)/cosh(kww*H))*sinh(kww*z0))...
*((2*(nww-1))/((kww*H)^2)));
Uww=(-3/2)*(Iww/Ifw)* (v2_ww+Mww/(E*Iww));
Tau=Vw*Sw0/Iw+Vww*Sww0/Iww;

```

```

Sigma=-E*x0*v2_w-(1f*(-l1)/2)*E*(v2_ww+Uww*(1-(2*x0/l)^2));
Sigma11(1,i)=(Sigma/2)+sqrt(((Sigma^2)/4)+Tau^2);
Sigma22(1,i)=(Sigma/2)-sqrt(((Sigma^2)/4)+Tau^2);
A=2*Tau/Sigma;

if Sigma==0;
    if Tau>0;
        F1(1,i)=45;
        F2(1,i)=135;
    elseif Tau<0;
        F1(1,i)=-45;
        F2(1,i)=-135;
    end
elseif Tau==0;
    if Sigma>0;
        F1(1,i)=0;
        F2(1,i)=90;
    elseif Sigma<0;
        F2(1,i)=180;
        F1(1,i)=-90;
    end
elseif Tau>0;
    if Sigma>0;
        F1(1,i)=0.5*(atan(A))*180/pi;
        F2(1,i)=F1(1,i)+90;
    elseif Sigma<0
        F1(1,i)=0.5*(atan(A))*180/pi+90;
        F2(1,i)=F1(1,i)+90;
    end
elseif Tau<0;
    if Sigma>0;
        F1(1,i)=0.5*(atan(A))*180/pi;
        F2(1,i)=F1(1,i)-90;
    elseif Sigma<0
        F1(1,i)=0.5*(atan(A))*180/pi-90;
        F2(1,i)=F1(1,i)-90;
    end
end
end
F1(1,i)=F1(1,i);
F2(1,i)=F2(1,i);

X1(1,i)=x0;
Z1(1,i)=H-z0;
X2(1,i)=x0;
Z2(1,i)=H-z0;
Y1(1,i)=(-l1/2)*1f;
Y2(1,i)=(-l1/2)*1f;
z1=H-z0;
z2=H-z0;

if iii==1
    Sigma1_1(1,i)=Sigma11(1,i);
    Sigma2_1(1,i)=Sigma22(1,i);
    F1_1(1,i)=F1(1,i);
    X1_1(1,i)=X1(1,i);
    Z1_1(1,i)=Z1(1,i);
    Y1_1(1,i)=Y1(1,i);
    F1_2(1,i)=F2(1,i);
    X1_2(1,i)=X2(1,i);
    Z1_2(1,i)=Z2(1,i);
    Y1_2(1,i)=Y1(1,i);

```

```

elseif iii==2
    Sigma2_1(1,i)=Sigma11(1,i);
    Sigma2_2(1,i)=Sigma22(1,i);
    F2_1(1,i)=F1(1,i);
    X2_1(1,i)=X1(1,i);
    Z2_1(1,i)=Z1(1,i);
    Y2_1(1,i)=Y1(1,i);
    F2_2(1,i)=F2(1,i);
    X2_2(1,i)=X2(1,i);
    Z2_2(1,i)=Z2(1,i);
    Y2_2(1,i)=Y1(1,i);
end

if ii==1
    F2_1_A(1,i)=F2_1(1,i);
    F2_2_A(1,i)=F2_2(1,i);
    F1_1_A(1,i)=F1_1(1,i);
    F1_2_A(1,i)=F1_2(1,i);
    X2_1_A(1,i)=X2_1(1,i);
    X2_2_A(1,i)=X2_2(1,i);
    Z2_1_A(1,i)=Z2_1(1,i);
    Z2_2_A(1,i)=Z2_2(1,i);
    Y2_1_A(1,i)=Y2_1(1,i);
    Y2_2_A(1,i)=Y2_2(1,i);
    X1_1_A(1,i)=X1_1(1,i);
    X1_2_A(1,i)=X1_2(1,i);
    Z1_1_A(1,i)=Z1_1(1,i);
    Z1_2_A(1,i)=Z1_2(1,i);
    Y1_1_A(1,i)=Y1_1(1,i);
    Y1_2_A(1,i)=Y1_2(1,i);
    Sigma1_1_A(1,i)=Sigma1_1(1,i);
    Sigma1_2_A(1,i)=Sigma1_2(1,i);
    Sigma2_1_A(1,i)=Sigma2_1(1,i);
    Sigma2_2_A(1,i)=Sigma2_2(1,i);
elseif ii==2
    F2_1_C(1,i)=F2_1(1,i);
    F2_2_C(1,i)=F2_2(1,i);
    F1_1_C(1,i)=F1_1(1,i);
    F1_2_C(1,i)=F1_2(1,i);
    X2_1_C(1,i)=X2_1(1,i);
    X2_2_C(1,i)=X2_2(1,i);
    Z2_1_C(1,i)=Z2_1(1,i);
    Z2_2_C(1,i)=Z2_2(1,i);
    Y2_1_C(1,i)=Y2_2(1,i);
    Y2_2_C(1,i)=Y2_2(1,i);
    X1_1_C(1,i)=X1_1(1,i);
    X1_2_C(1,i)=X1_2(1,i);
    Z1_1_C(1,i)=Z1_1(1,i);
    Z1_2_C(1,i)=Z1_2(1,i);
    Y1_1C(1,i)=Y1_1(1,i);
    Y1_2C(1,i)=Y1_2(1,i);
    Sigma1_1_C(1,i)=Sigma1_1(1,i);
    Sigma1_2_C(1,i)=Sigma1_2(1,i);
    Sigma2_1_C(1,i)=Sigma2_1(1,i);
    Sigma2_2_C(1,i)=Sigma2_2(1,i);
elseif ii==3
    F2_1_B(1,i)=F2_1(1,i);
    F2_2_B(1,i)=F2_2(1,i);
    F1_1_B(1,i)=F1_1(1,i);
    F1_2_B(1,i)=F1_2(1,i);
    Y2_1_B(1,i)=X2_1(1,i);

```

```

Y2_2_B(1,i)=X2_2(1,i);
Z2_1_B(1,i)=Z2_1(1,i);
Z2_2_B(1,i)=Z2_2(1,i);
X2_1_B(1,i)=Y2_1(1,i);
X2_2_B(1,i)=Y2_2(1,i);
Y1_1_B(1,i)=X1_1(1,i);
Y1_2_B(1,i)=X1_2(1,i);
Z1_1_B(1,i)=Z1_1(1,i);
Z1_2_B(1,i)=Z1_2(1,i);
X1_1_B(1,i)=Y1_1(1,i);
X1_2_B(1,i)=Y1_2(1,i);
Sigma1_1_B(1,i)=Sigma1_1(1,i);
Sigma1_2_B(1,i)=Sigma1_2(1,i);
Sigma2_1_B(1,i)=Sigma2_1(1,i);
Sigma2_2_B(1,i)=Sigma2_2(1,i);
elseif ii==4
    F2_1_D(1,i)=F2_1(1,i);
    F2_2_D(1,i)=F2_2(1,i);
    F1_1_D(1,i)=F1_1(1,i);
    F1_2_D(1,i)=F1_2(1,i);
    Y2_1_D(1,i)=X2_1(1,i);
    Y2_2_D(1,i)=X2_2(1,i);
    Z2_1_D(1,i)=Z2_1(1,i);
    Z2_2_D(1,i)=Z2_2(1,i);
    X2_1_D(1,i)=Y2_1(1,i);
    X2_2_D(1,i)=Y2_2(1,i);
    Y1_1_D(1,i)=X1_1(1,i);
    Y1_2_D(1,i)=X1_2(1,i);
    Z1_1_D(1,i)=Z1_1(1,i);
    Z1_2_D(1,i)=Z1_2(1,i);
    X1_1_D(1,i)=Y1_1(1,i);
    X1_2_D(1,i)=Y1_2(1,i);
    Sigma1_1_D(1,i)=Sigma1_1(1,i);
    Sigma1_2_D(1,i)=Sigma1_2(1,i);
    Sigma2_1_D(1,i)=Sigma2_1(1,i);
    Sigma2_2_D(1,i)=Sigma2_2(1,i);
end

%Tension lines construction (Sigma 1)
jj=1;
while z1>0
    jj=jj+1;
    if F1((jj-1),i)==0
        X1(jj,i)=X1((jj-1),i);
        Z1(jj,i)=Z1((jj-1),i)+H/mm;
    elseif F1((jj-1),i)==180
        X1(jj,i)=X1((jj-1),i);
        Z1(jj,i)=Z1((jj-1),i)-H/mm;
    else
        X1(jj,i)=X1((jj-1),i)+f*ff*(1/m);
        if F1((jj-1),i)==90;
            Z1(jj,i)=Z1((jj-1),i);
        elseif F1((jj-1),i)==-90;
            Z1(jj,i)=Z1((jj-1),i);
        elseif F1((jj-1),i)==270;
            Z1(jj,i)=Z1((jj-1),i);
        elseif F1((jj-1),i)==-270;
            Z1(jj,i)=Z1((jj-1),i);
        else
            Z1(jj,i)=Z1((jj-1),i)+ff*(1/m)/...
                tan(F1((jj-1),i)*pi/180);
        end
    end
end

```

```

end
end

%Boundary check
if Z1(jj,i)<0
    if F1((jj-1),i)==180
        Z1(jj,i)=0;
        X1(jj,i)=X1(jj-1,i);
        z1=0;
        x1=X1(jj,i);
    else
        Z1(jj,i)=0;
        X1(jj,i)=X1(jj-1,i)-f*Z1((jj-1),i)...
            *tan(F1((jj-1),i)*pi/180);
        if X1(jj,i)>l/2
            Z1(jj,i)=Z1((jj-1),i)+f*(l/2-X1((jj-
1),i))/...
                tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=l/2;
            z1=0;
            x1=X1(jj,i);
        elseif X1(jj,i)<-l/2
            Z1(jj,i)=Z1((jj-1),i)+f*((-l/2)-X1((jj-
1),i))...
                /tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=-l/2;
            z1=0;
            x1=X1(jj,i);
        else
            z1=0;
            x1=X1(jj,i);
        end
    end
elseif Z1(jj,i)>H
    if F1((jj-1),i)==0
        Z1(jj,i)=H;
        X1(jj,i)=X1(jj-1,i);
        z1=0;
        x1=X1(jj,i);
    else
        Z1(jj,i)=H;
        X1(jj,i)=X1(jj-1,i)+f*(H-Z1((jj-1),i))...
            *tan(F1((jj-1),i)*pi/180);
        if X1(jj,i)>l/2
            Z1(jj,i)=Z1((jj-1),i)+f*(l/2-X1((jj-
1),i))/...
                tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=l/2;
            z1=0;
            x1=X1(jj,i);
        elseif X1(jj,i)<-l/2
            Z1(jj,i)=Z1((jj-1),i)+f*((-l/2)-X1((jj-
1),i))...
                /tan(F1((jj-1),i)*pi/180);
            X1(jj,i)=-l/2;
            z1=0;
            x1=X1(jj,i);
        else
            z1=0;
            x1=X1(jj,i);
        end
    end
end
end

```

```

end
elseif X1(jj,i)>1/2
if F1((jj-1),i)==90
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==-90
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==270
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==-270
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=1/2;
z1=0;
x1=X1(jj,i);
else
X1(jj,i)=1/2;
Z1(jj,i)=Z1((jj-1),i)+f*(1/2-X1((jj-1),i))*...
/tan(F1((jj-1),i)*pi/180);
z1=0;
x1=1/2;
end
elseif X1(jj,i)<-1/2
if F1((jj-1),i)==90
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=-1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==-90
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=-1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==270
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=-1/2;
z1=0;
x1=X1(jj,i);
elseif F1((jj-1),i)==-270
Z1(jj,i)=Z1(jj-1,i);
X1(jj,i)=-1/2;
z1=0;
x1=X1(jj,i);
else
X1(jj,i)=-1/2;
Z1(jj,i)=Z1((jj-1),i)+f*((-1/2)-X1((jj-1),i))*...
/tan(F1((jj-1),i)*pi/180);
z1=0;
x1=-1/2;
end
end
else
Z1(jj,i)=Z1(jj,i);
X1(jj,i)=X1(jj,i);
z1=Z1(jj,i);

```

```

        x1=X1 (jj, i);
end

Y1 (jj, i)=(-ll/2) *lf;

z11=H-z1;
Mw1=-w* (H^2) /2;
Mww1=-ww* (H^2) /2;
Mw1=Mw1* (z11/H) ^2;
Mww1=Mww1* (z11/H) ^2;
Vw1=w*z11;
Vww1=ww*z11;
N1=(( (ww* (z11^2) /2) * (lf*ll/2) ) /Iw) *1*tw;
Sw1=sw*tww* (ll*1) /4+sw*tw*0.5* ( (1/2) ^2-x1^2) ;
Sww1=sww*tw* (ll/2) *x1;
v2_w1=-Mw1/ (E*Iw) * ( (z11/H) ^2+ (cosh (kw*z11) ...
-1- ( (sinh (kw*H) -kw*H) /cosh (kw*H) ) *...
sinh (kw*z11) ) * ( (2* (nw-1) ) / ( (kw*H) ^2) ) );
v2_ww1=-Mww1/ (E*Iww) * ( (z11/H) ^2+ (cosh (kww*z11) ...
-1- ( (sinh (kww*H) -
kww*H) /cosh (kww*H) ) *sinh (kww*z11) ) ...
* ( (2* (nww-1) ) / ( (kww*H) ^2) ) );
Uww1=(-3/2) * (Iww/Ifw) * (v2_ww1+Mww1/ (E*Iww) ) ;
Tau1=Vw1*Sw1/Iw+Vww1*Sww1/Iww;
Sigma1=-E*x1*v2_w1- (lf* (-ll) /2) *E* (v2_ww1+Uww1* (1-
(2*x1/l) ^2) );

Sigma11 (jj, i)= (Sigma1/2) +sqrt ( ( (Sigma1^2) /4) +Tau1^2) ;
A1=2*Tau1/Sigma1;

if Sigma1==0;
    if Tau1>0;
        F1 (jj, i)=45;
    elseif Tau1<0;
        F1 (jj, i)=-45;
    end
elseif Tau1==0;
    if Sigma1>0;
        F1 (jj, i)=0;
    elseif Sigma1<0;
        F1 (jj, i)=-90;
    end
elseif Tau1>0;
    if Sigma1>0;
        F1 (jj, i)=0.5* (atan (A1) ) *180/pi;
    elseif Sigma1<0
        F1 (jj, i)=0.5* (atan (A1) ) *180/pi+90;
    end
elseif Tau1<0;
    if Sigma1>0;
        F1 (jj, i)=0.5* (atan (A1) ) *180/pi;
    elseif Sigma1<0
        F1 (jj, i)=0.5* (atan (A1) ) *180/pi-90;
    end
end
F1 (jj, i)=F1 (jj, i);

if iii==1
    Sigma1_1 (jj, i)=Sigma11 (jj, i);
    F1_1 (jj, i)=F1 (jj, i);
end

```

```

        X1_1(jj,i)=X1(jj,i);
        Z1_1(jj,i)=Z1(jj,i);
        Y1_1(jj,i)=Y1(jj,i);
    elseif iii==2
        Sigma2_1(jj,i)=Sigma11(jj,i);
        F2_1(jj,i)=F1(jj,i);
        X2_1(jj,i)=X1(jj,i);
        Z2_1(jj,i)=Z1(jj,i);
        Y2_1(jj,i)=Y1(jj,i);
    end
end

%Compression lines construction (Sigma 2)
jjj=1;
while z2>0
    jjj=jjj+1;
    if F2((jjj-1),i)==0
        X2(jjj,i)=X2((jjj-1),i);
        Z2(jjj,i)=Z2((jjj-1),i)+H/mm;
    elseif F2((jjj-1),i)==180
        X2(jjj,i)=X2((jjj-1),i);
        Z2(jjj,i)=Z2((jjj-1),i)-H/mm;
    else
        X2(jjj,i)=X2((jjj-1),i)+f*ff*(1/m);
        if F2((jjj-1),i)==90;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==-90;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==270;
            Z2(jjj,i)=Z2((jjj-1),i);
        elseif F2((jjj-1),i)==-270;
            Z2(jjj,i)=Z2((jjj-1),i);
        else
            Z2(jjj,i)=Z2((jjj-1),i)+ff*(1/m)/...
                tan(F2((jjj-1),i)*pi/180);
        end
    end
end

%Boundary check
if Z2(jjj,i)<0
    if F2((jjj-1),i)==180
        Z2(jjj,i)=0;
        X2(jjj,i)=X2(jjj-1,i);
        z2=0;
        x2=X2(jjj,i);
    else
        Z2(jjj,i)=0;
        X2(jjj,i)=X2(jjj-1,i)-f*Z2((jjj-1),i)*...
            tan(F2((jjj-1),i)*pi/180);
        if X2(jjj,i)>1/2
            Z2(jjj,i)=Z2((jjj-1),i)+f*(1/2-...
                X2((jjj-1),i)/tan(F2((jjj-1),i)*pi/180);
            X2(jjj,i)=1/2;
            z2=0;
            x2=X2(jjj,i);
        elseif X2(jjj,i)<-1/2
            Z2(jjj,i)=Z2((jjj-1),i)+f*((-1/2)-...
                X2((jjj-1),i)/tan(F2((jjj-1),i)*pi/180);
            X2(jjj,i)=-1/2;
            z2=0;
            x2=X2(jjj,i);
        end
    end
end

```



```

else
    z2=0;
    x2=X2(jjj,i);
end
end
elseif Z2(jjj,i)>H
if F2((jjj-1),i)==0
    Z2(jjj,i)=H;
    X2(jjj,i)=X2(jjj-1,i);
    z2=0;
    x2=X2(jjj,i);
else
    Z2(jjj,i)=H;
    X2(jjj,i)=X2(jjj-1,i)+f*(H-Z2((jjj-1),i))*...
        tan(F2((jjj-1),i)*pi/180);
if X2(jjj,i)>1/2
    Z2(jjj,i)=Z2((jjj-1),i)+f*(1/2-...
        X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
    X2(jjj,i)=1/2;
    z2=0;
    x2=X2(jjj,i);
elseif X2(jjj,i)<-1/2
    Z2(jjj,i)=Z2((jjj-1),i)+f*((-1/2)-...
        X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
    X2(jjj,i)=-1/2;
    z2=0;
    x2=X2(jjj,i);
else
    z2=0;
    x2=X2(jjj,i);
end
end
elseif X2(jjj,i)>1/2
if F2((jjj-1),i)==90
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==-90
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==270
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==-270
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=1/2;
    z2=0;
    x2=X2(jjj,i);
else
    X2(jjj,i)=1/2;
    Z2(jjj,i)=Z2((jjj-1),i)+f*(1/2-...
        X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
    z2=0;
    x2=1/2;
end
elseif X2(jjj,i)<-1/2

```

```

if F2((jjj-1),i)==90
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=-1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==-90
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=-1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==270
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=-1/2;
    z2=0;
    x2=X2(jjj,i);
elseif F2((jjj-1),i)==-270
    Z2(jjj,i)=Z2(jjj-1,i);
    X2(jjj,i)=-1/2;
    z2=0;
    x2=X2(jjj,i);
else
    X2(jjj,i)=-1/2;
    Z2(jjj,i)=Z2((jjj-1),i)+f*((-1/2)-...
        X2((jjj-1),i))/tan(F2((jjj-1),i)*pi/180);
    z2=0;
    x2=-1/2;
end
else
    Z2(jjj,i)=Z2(jjj,i);
    X2(jjj,i)=X2(jjj,i);
    z2=Z2(jjj,i);
    x2=X2(jjj,i);
end

Y2(jjj,i)=(-11/2)*1f;

z22=H-z2;
Mw2=-w*(H^2)/2;
Mww2=-ww*(H^2)/2;
Mw2=Mw2*(z22/H)^2;
Mww2=Mww2*(z22/H)^2;
Vw2=w*z22;
Vww2=ww*z22;
N2=((w*(z22^2)/2)*(1f*11/2))/Iw)*1*tw;
Sw2=sw*tw*(11*1)/4+sw*tw*0.5*((1/2)^2-x2^2);
Sww2=sww*tw*(11/2)*x2;
v2_w2=-Mw2/(E*Iw)*((z22/H)^2+(cosh(kw*z22)...
    -1-((sinh(kw*H)-kw*H)/cosh(kw*H))...
    *sinh(kw*z22))*((2*(nw-1))/(kw*H)^2));
v2_ww2=-Mww2/(E*Iww)*((z22/H)^2+(cosh(kww*z22)...
    -1-((sinh(kww*H)-kww*H)/cosh(kww*H))...
    *sinh(kww*z22))*((2*(nww-1))/(kww*H)^2));
Uww2=(-3/2)*(Iww/Ifw)*v2_ww2+Mww2/(E*Iww);
Tau2=Vw2*Sw2/Iw+Vww2*Sww2/Iww;
Sigma2=-E*x2*v2_w2-(1f*(-11)/2)*E*(v2_ww2+Uww2*(1-
(2*x2/1)^2));

Sigma22(jjj,i)=(Sigma2/2)-sqrt(((Sigma2^2)/4)+Tau2^2);
A2=2*Tau2/Sigma2;

if Sigma2==0;

```

```

        if Tau2>0;
            F2(jjj,i)=135;
        elseif Tau2<0;
            F2(jjj,i)=-135;
        end
    elseif Tau2==0;
        if Sigma2>0;
            F2(jjj,i)=90;
        elseif Sigma2<0;
            F2(jjj,i)=180;
        end
    elseif Tau2>0;
        if Sigma2>0;
            F2(jjj,i)=0.5*(atan(A2))*180/pi+90;
        elseif Sigma2<0
            F2(jjj,i)=0.5*(atan(A2))*180/pi+90+90;
        end
    elseif Tau2<0;
        if Sigma2>0;
            F2(jjj,i)=0.5*(atan(A2))*180/pi-90;
        elseif Sigma2<0
            F2(jjj,i)=0.5*(atan(A2))*180/pi-90-90;
        end
    end
    F2(jjj,i)=F2(jjj,i);

    if iii==1
        Sigma1_2(jjj,i)=Sigma22(jjj,i);
        F1_2(jjj,i)=F2(jjj,i);
        X1_2(jjj,i)=X2(jjj,i);
        Z1_2(jjj,i)=Z2(jjj,i);
        Y1_2(jjj,i)=Y2(jjj,i);
    elseif iii==2
        Sigma2_2(jjj,i)=Sigma22(jjj,i);
        F2_2(jjj,i)=F2(jjj,i);
        X2_2(jjj,i)=X2(jjj,i);
        Z2_2(jjj,i)=Z2(jjj,i);
        Y2_2(jjj,i)=Y2(jjj,i);
    end
end
end
end

%Face components selection
if ii==1
    F2_1_A=F2_1;
    F2_2_A=F2_2;
    F1_1_A=F1_1;
    F1_2_A=F1_2;
    X2_1_A=X2_1;
    X2_2_A=X2_2;
    Z2_1_A=Z2_1;
    Z2_2_A=Z2_2;
    X1_1_A=X1_1;
    X1_2_A=X1_2;
    Z1_1_A=Z1_1;
    Z1_2_A=Z1_2;
    Y1_1_A=Y1_1;
    Y1_2_A=Y1_2;
    Y2_1_A=Y2_1;
    Y2_2_A=Y2_2;

```

```

Sigma1_1_A=Sigma1_1;
Sigma1_2_A=Sigma1_2;
Sigma2_1_A=Sigma2_1;
Sigma2_2_A=Sigma2_2;
elseif ii==2
F2_1_C=F2_1;
F2_2_C=F2_2;
F1_1_C=F1_1;
F1_2_C=F1_2;
X2_1_C=X2_1;
X2_2_C=X2_2;
Z2_1_C=Z2_1;
Z2_2_C=Z2_2;
X1_1_C=X1_1;
X1_2_C=X1_2;
Z1_1_C=Z1_1;
Z1_2_C=Z1_2;
Y1_1_C=Y1_1;
Y1_2_C=Y1_2;
Y2_1_C=Y2_1;
Y2_2_C=Y2_2;
Sigma1_1_C=Sigma1_1;
Sigma1_2_C=Sigma1_2;
Sigma2_1_C=Sigma2_1;
Sigma2_2_C=Sigma2_2;
elseif ii==3
F2_1_B=F2_1;
F2_2_B=F2_2;
F1_1_B=F1_1;
F1_2_B=F1_2;
Y2_1_B=X2_1;
Y2_2_B=X2_2;
Z2_1_B=Z2_1;
Z2_2_B=Z2_2;
Y1_1_B=X1_1;
Y1_2_B=X1_2;
Z1_1_B=Z1_1;
Z1_2_B=Z1_2;
X1_1_B=Y1_1;
X1_2_B=Y1_2;
X2_1_B=Y2_1;
X2_2_B=Y2_2;
Sigma1_1_B=Sigma1_1;
Sigma1_2_B=Sigma1_2;
Sigma2_1_B=Sigma2_1;
Sigma2_2_B=Sigma2_2;
elseif ii==4
F2_1_D=F2_1;
F2_2_D=F2_2;
F1_1_D=F1_1;
F1_2_D=F1_2;
Y2_1_D=X2_1;
Y2_2_D=X2_2;
Z2_1_D=Z2_1;
Z2_2_D=Z2_2;
Y1_1_D=X1_1;
Y1_2_D=X1_2;
Z1_1_D=Z1_1;
Z1_2_D=Z1_2;
X1_1_D=Y1_1;
X1_2_D=Y1_2;

```

```

X2_1_D=Y2_1;
X2_2_D=Y2_2;
Sigma1_1_D=Sigma1_1;
Sigma1_2_D=Sigma1_2;
Sigma2_1_D=Sigma2_1;
Sigma2_2_D=Sigma2_2;
end

end

%% Model representation

% Principal stress trajectories
figure
axis equal
%Faces construction
patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],[0,0,0,0],...
      [0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+10])
xticks([-a/2:1:a/2])
yticks([-b/2:1:b/2])
zticks([0:1:H])
title('Principal stress direction - Tension state')
xlabel('Section "a" [m]')
ylabel('Section "b" [m]')
zlabel('Height [m]')

hold on

%Axis
quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],...
        [0;0;0],'k')

line(X1_1_A,Y1_1_A,Z1_1_A,'Color','k')
line(X1_2_A,Y1_2_A,Z1_2_A,'Color','k')
line(X2_1_A,Y2_1_A,Z2_1_A,'Color','k')
line(X2_2_A,Y2_2_A,Z2_2_A,'Color','k')

line(X1_1_C,Y1_1_C,Z1_1_C,'Color','k')
line(X1_2_C,Y1_2_C,Z1_2_C,'Color','k')
line(X2_1_C,Y2_1_C,Z2_1_C,'Color','k')
line(X2_2_C,Y2_2_C,Z2_2_C,'Color','k')

line(X1_1_B,Y1_1_B,Z1_1_B,'Color','k')
line(X1_2_B,Y1_2_B,Z1_2_B,'Color','k')
line(X2_1_B,Y2_1_B,Z2_1_B,'Color','k')
line(X2_2_B,Y2_2_B,Z2_2_B,'Color','k')

line(X1_1_D,Y1_1_D,Z1_1_D,'Color','k')
line(X1_2_D,Y1_2_D,Z1_2_D,'Color','k')
line(X2_1_D,Y2_1_D,Z2_1_D,'Color','k')

```

```

line(X2_2_D,Y2_2_D,Z2_2_D,'Color','k')

% Tension state
figure
axis equal
patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],[0,0,0,0],[0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+10])
xticks([-a/2:10:a/2])
yticks([-b/2:10:b/2])
zticks([0:10:H])
title('Principal stress direction - Tension state')
xlabel('Section "a" [m]')
ylabel('Section "b" [m]')
zlabel('Height [m]')

hold on

quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],[0;0;0], 'k')
quiver3(X_A,Y_A,Z_A,CC1_A,O_A,SS1_A,0.5,'b')
quiver3(X_C,Y_C,Z_C,CC1_C,O_C,SS1_C,0.5,'b')
quiver3(X_B,Y_B,Z_B,O_B,CC1_B,SS1_B,0.5,'b')
quiver3(X_D,Y_D,Z_D,O_D,CC1_D,SS1_D,0.5,'b')

line(X1_1_A,Y1_1_A,Z1_1_A,'Color','k')
line(X2_1_A,Y2_1_A,Z2_1_A,'Color','k')

line(X1_1_C,Y1_1_C,Z1_1_C,'Color','k')
line(X2_1_C,Y2_1_C,Z2_1_C,'Color','k')

line(X1_1_B,Y1_1_B,Z1_1_B,'Color','k')
line(X2_1_B,Y2_1_B,Z2_1_B,'Color','k')

line(X1_1_D,Y1_1_D,Z1_1_D,'Color','k')
line(X2_1_D,Y2_1_D,Z2_1_D,'Color','k')

% Compression state
figure
axis equal
patch([-a/2-2,a/2+2,a/2+2,-a/2-2],[-b/2-2,-b/2-2,b/2+2,b/2+2],[0,0,0,0],...
[0.85 0.85 0.85]);
s_A=patch([-a/2,a/2,a/2,-a/2],[-b/2,-b/2,-b/2,-b/2],[0,0,H,H],'w');
s_B=patch([a/2,a/2,a/2,a/2],[-b/2,b/2,b/2,-b/2],[0,0,H,H],'w');
s_C=patch([a/2,-a/2,-a/2,a/2],[b/2,b/2,b/2,b/2],[0,0,H,H],'w');
s_D=patch([-a/2,-a/2,-a/2,-a/2],[b/2,-b/2,-b/2,b/2],[0,0,H,H],'w');
xlim([-a/2-10 (a/2)+10])
ylim([-b/2-10 (b/2)+10])
zlim([-1 H+10])
xticks([-a/2:10:a/2])
yticks([-b/2:10:b/2])
zticks([0:10:H])
title('Principal stress direction - Compression state')

```

```

xlabel('Section [m]')
ylabel('Section [m]')
zlabel('Height [m]')

hold on

quiver3(zeros(3,1),zeros(3,1),zeros(3,1),[a/2+10;0;0],[0;b/2+10;0],...
        [0;0;0], 'k')
quiver3(X_A,Y_A,Z_A,CC2_A,O_A,SS2_A,0.5, 'r')
quiver3(X_C,Y_C,Z_C,CC2_C,O_C,SS2_C,0.5, 'r')
quiver3(X_B,Y_B,Z_B,O_B,CC2_B,SS2_B,0.5, 'r')
quiver3(X_D,Y_D,Z_D,O_D,CC2_D,SS2_D,0.5, 'r')

line(X1_2_A,Y1_2_A,Z1_2_A, 'Color', 'k')
line(X2_2_A,Y2_2_A,Z2_2_A, 'Color', 'k')

line(X1_2_C,Y1_2_C,Z1_2_C, 'Color', 'k')
line(X2_2_C,Y2_2_C,Z2_2_C, 'Color', 'k')

line(X1_2_B,Y1_2_B,Z1_2_B, 'Color', 'k')
line(X2_2_B,Y2_2_B,Z2_2_B, 'Color', 'k')
line(X1_2_D,Y1_2_D,Z1_2_D, 'Color', 'k')
line(X2_2_D,Y2_2_D,Z2_2_D, 'Color', 'k')

```

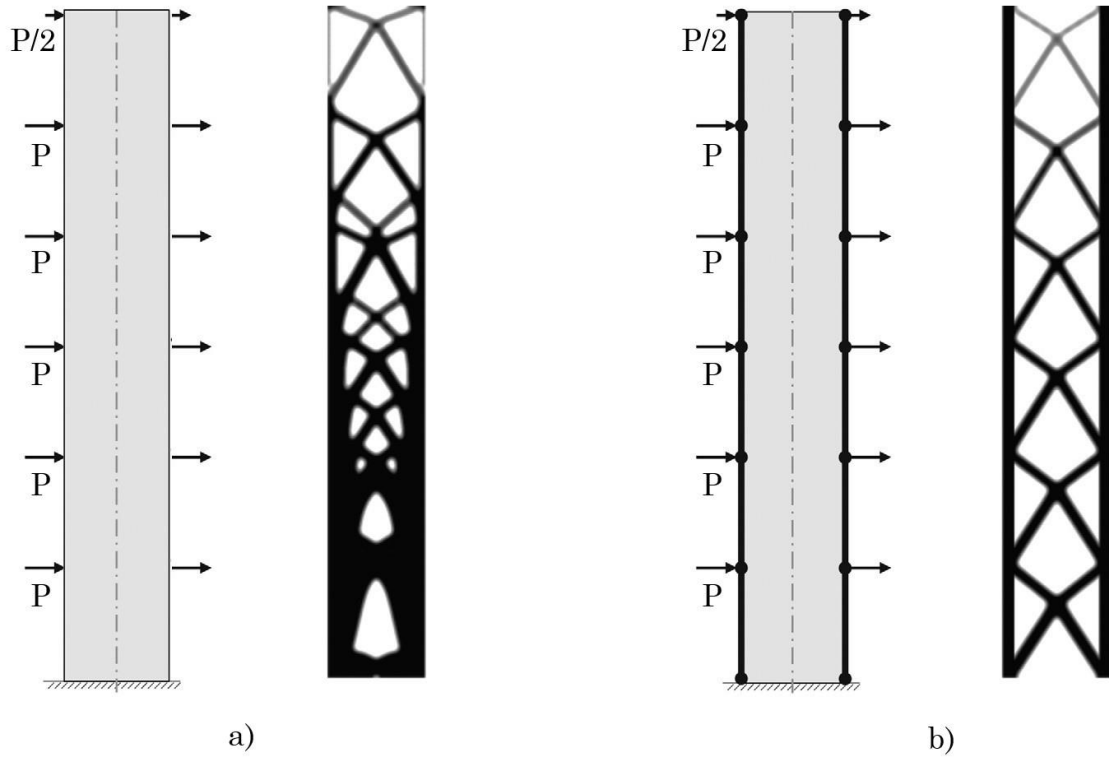
### 10.3. Topology optimization by TOSCA with continuum-beam approach

To implement the topology optimization process in TOSCA, it has been chosen to follow the approach proposed by L. L. Stromberg at all [30]. The topology optimization process itself is strictly dependent on many factors, from the load application to the mesh subdivision, as it was explained in Chapter 4, so, the result should vary or being affected by modeling choices.

Using the same example proposed by Stromberg in his paper, it was observed that in a 2D system the topology process tends to add material at vertical sides, especially close to the base, in order to obtain vertical elements as distant as possible from the axis to carry the overturning moment.

This creates a huge amount of material in that part subtracting resources for the creation of the diagrid in the other parts of the structure, leading to not feasible results.

It is marked now that with those approaches on topology optimization must be interpreted to obtain guide lines for design purposes, then a large amount of material at base columns hide elements node intersections and elements layouts.



*Figure 10.2 From Stromberg at all paper, a) is the case of a meshed rectangle under a set of transversal load, and b) is the same system but with the contribution of vertical beams at each lateral side*

To avoid this behaviour Stromberg at all, propose to add directly to the mesh body beam elements, with a proper stiffness, at the sides where this material amount lays. Those contributions set free the topology process to focus only on the diagrid construction, then without adding material for vertical elements. As it is shown in this approach leads to a clearer result which permits to identify all elements and their connections.

The same procedure has been used in ABAQUS for the modeling of the system, but some difference from Stromberg at all [30] was taken.



Being the process sensible to loads modelling, like in the paper, using point loads, the application point itself is already for the system a connection points which affects the final element disposition, so the number of cells and their dimensions is affected. Then, a distributed load is still maintained.

The beam elements must be set with appropriate cross sections in order to have enough stiffness at the base, but not too much to neglect the diagrid contribution.

More details on computational procedures and point load cases description refer to Stromberg at all [30].

To start, the dimensions of the column diameter must be obtained and then test them in the TOSCA model. To di that a tube truss model in 2D is solver considering resulting point loads on external notes, and then the required area for the maximum element capacity is given.

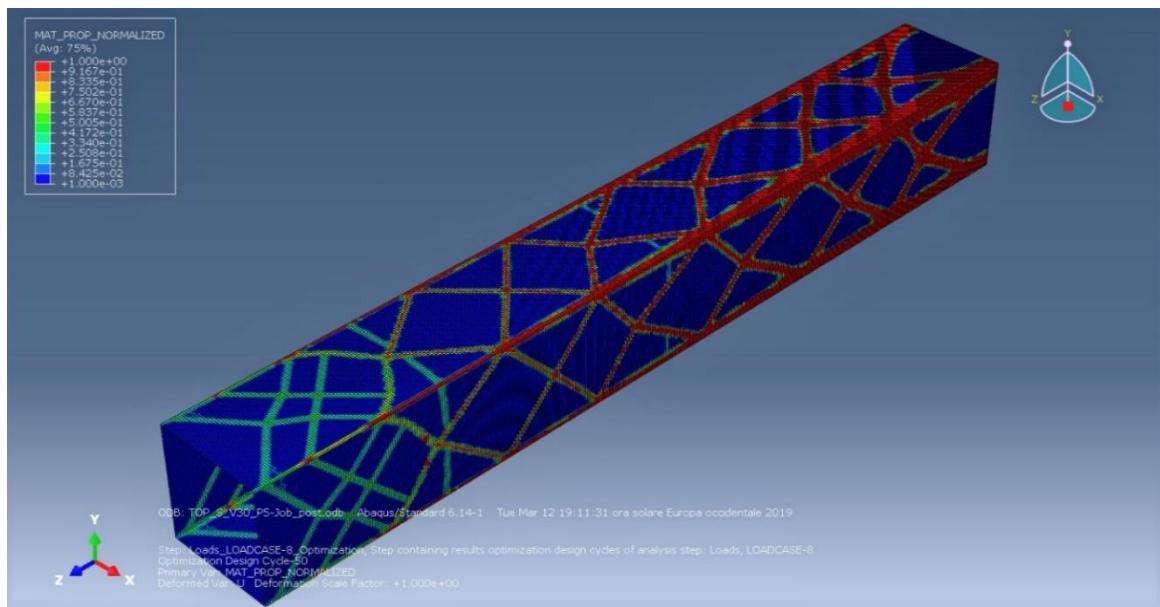


Figure 10.3 ABAQUS model, results from the first topology analysis without beam elements

The diagrid is taken from a first ABAQUS topology analysis without beam elements, in multi-loading condition. Making some test, varying penalty factor

and density interval action, the most feasible result is taken from a penalty factor of 5.

Before proceeding, it has been noted that the difference between a penalty factor of 5 and 3 it the number of cross-elements. In the first case it should be said that 7 cross points are needed, while in the other case 8, so the overall configuration change, but the final column element dimension, under the same set load, was observed to remain the same.

From the diagrid façade modulus the truss system has been obtained and solved obtaining the normal stresses in each element. About the truss system, in the topology procedures hinges are not considered, so this is just a simplification which still leads to correct results.

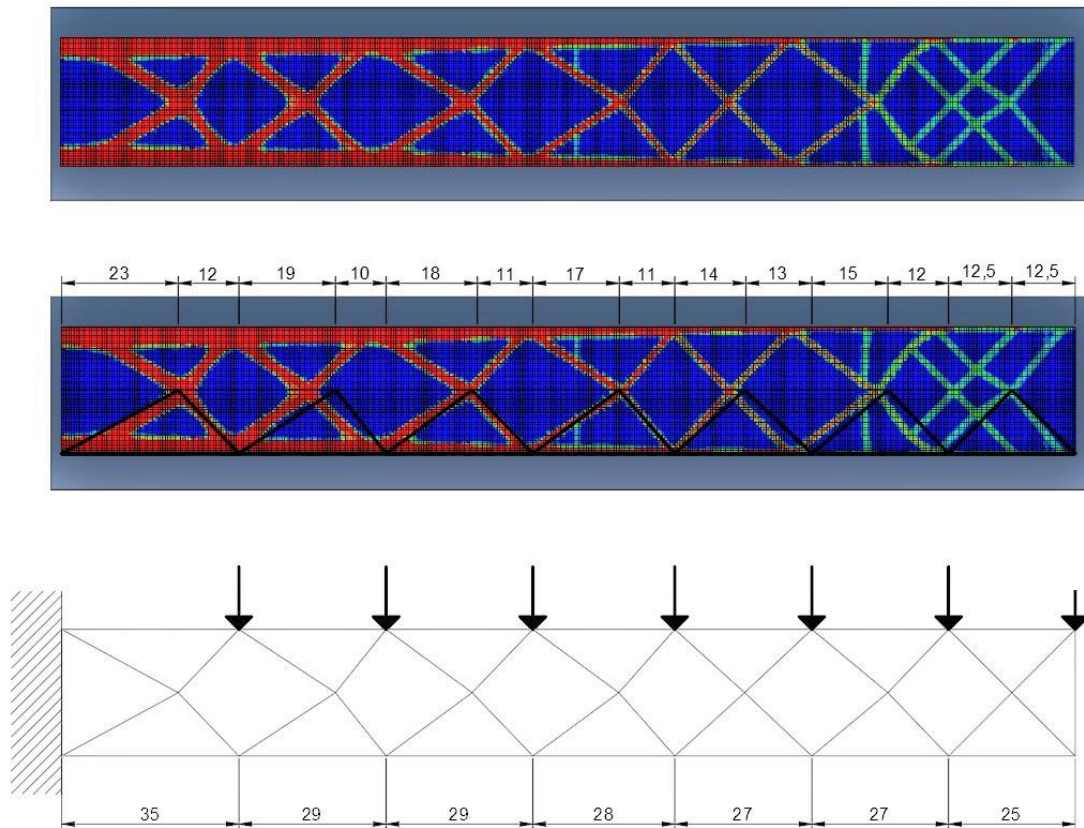
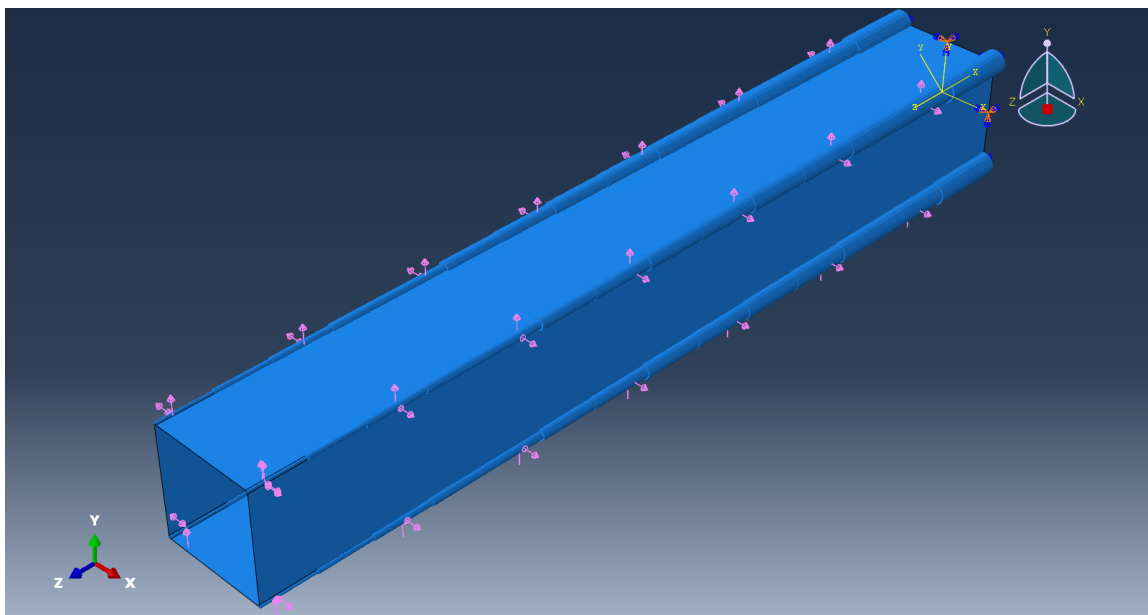


Figure 10.4 Diagrid evaluation for columns pre-sizing

The columns now are dimensioned till the material limit. It must be noted that there is no mention about material name and properties. This because all those procedures are a matter of geometries and proportions between loads, so the material yield stress limit determines the cross-section dimension and not the proportions between the cross-sections of the totality of the elements. Only after the layout is obtained is possible to verify and check all limits with criterions and material properties.

Having a column dimensions, they are modeled on ABAQUS and attached to the corner sides of the cantilever and the analysis is implemented again considering this addition to the overall strain energy but not in the volume fraction. It is observed also that the process tends to converge easily than in the other cases.



*Figure 10.5 ABAQUS model, Continuum-Beam model*

About the cross-section dimensions, their proportion is given by the truss solving, but their relationship with the mesh system must be adjusted during the analysis, running one time, observing if the process still put material in the columns and then increasing the sections. Sometimes they should be reduced in order not to affect, as it was said before, the result.

## 10.4. Vertical elements in the final robust diagrid

As it was said in the introduction chapter, the diagrid in the classical meaning doesn't have vertical elements but it is made by diagonals only. By the way, it has been observed that in the 3D model, as in the 2D, of the cantilever, vertical elements occur in the corners of the cross section and they have a marked presence close to the base. The mechanical reason is the large amount of normal stresses in those points compared to shear stresses, so a geometric effect influences the diagrid final layout. The second reason should be related to the necessity to transfer the stresses, split when they arrive to the corner, to the base.

Anyway, even if those geometrical features affect the presence of those elements, they are still needed at the case part, for the reason said before, so, the diagrid made by diagonal only has no much performance at the base than one with vertical elements there. To demonstrate the natural development of vertical elements an ABAQUS test with a tube with circular cross section has been implemented.

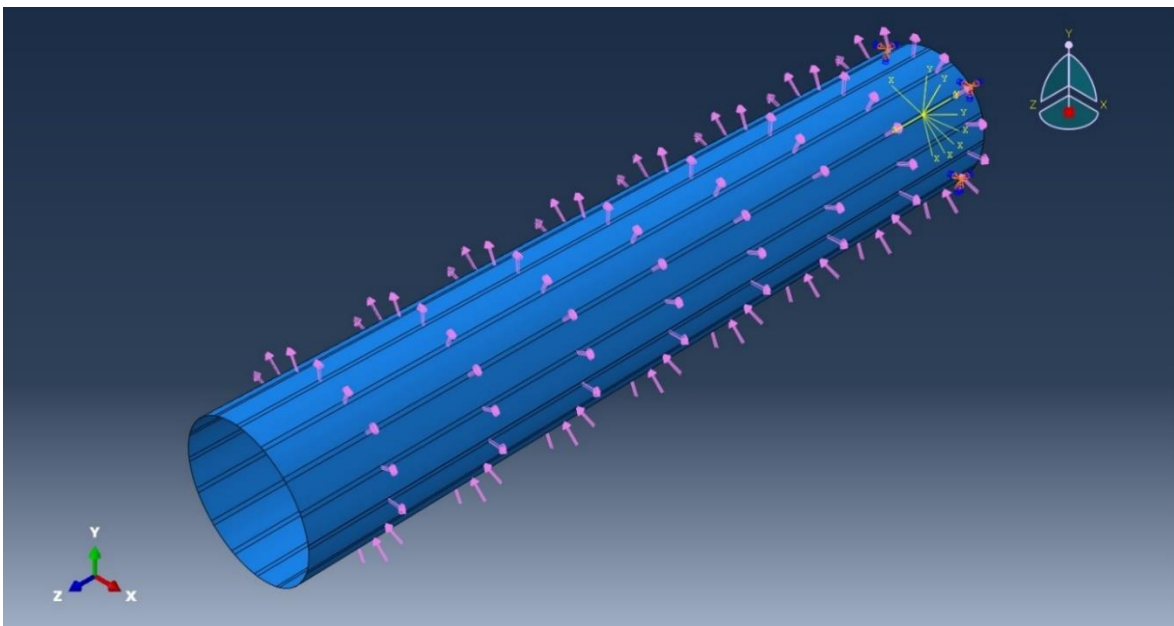


Figure 10.6 ABAQUS model, cylinder model

The same load cases of the square section case have been used, and moreover, only some points, or small local regions, should be clamped. This last condition is due to the absence of particular points from which starting the diagrid if all the edge is clamped, and then, how tests was observed by tests, the result is still the same cylinder but with a reduce height in order to achieve the given volume fraction.

Three constraint cases have been considered, 4 clamped points, 8 clamped point and 16 clamped points.

The scope of this test is not to define a clear diagrid but to observe the nature of the vertical element behaviour.

Moreover, the green vertical elements presented at the top of the tube are a mesh depending effect of the model, due to the mesh strips made for the loads.

The results are listed below.

Four clamped points:

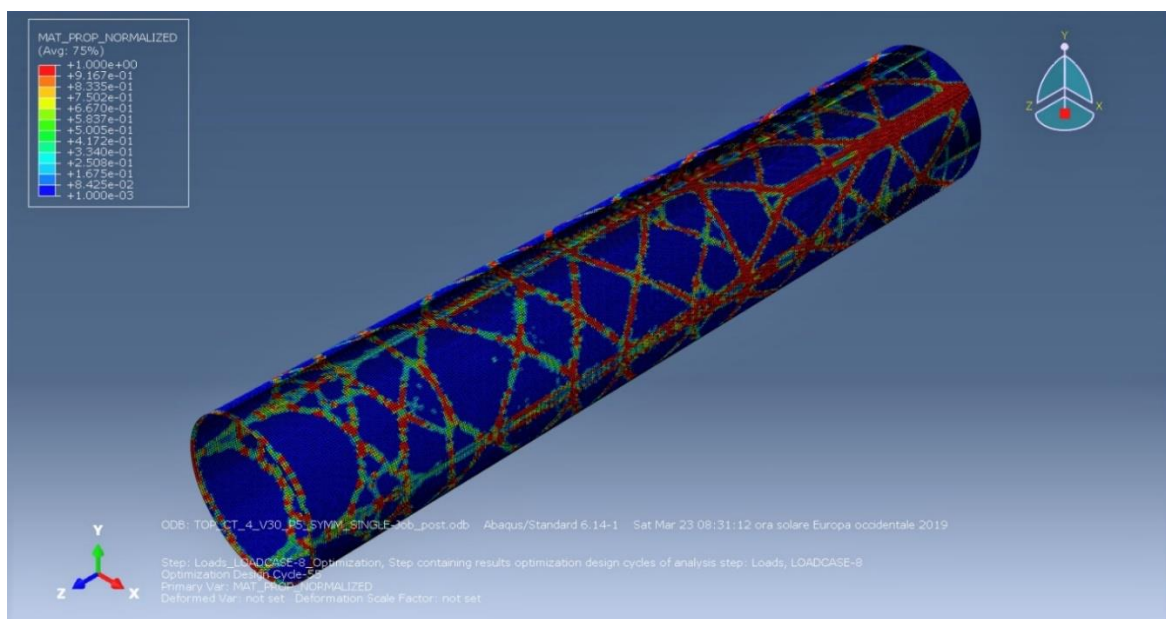


Figure 10.7 ABAQUS model, four clamped points general view

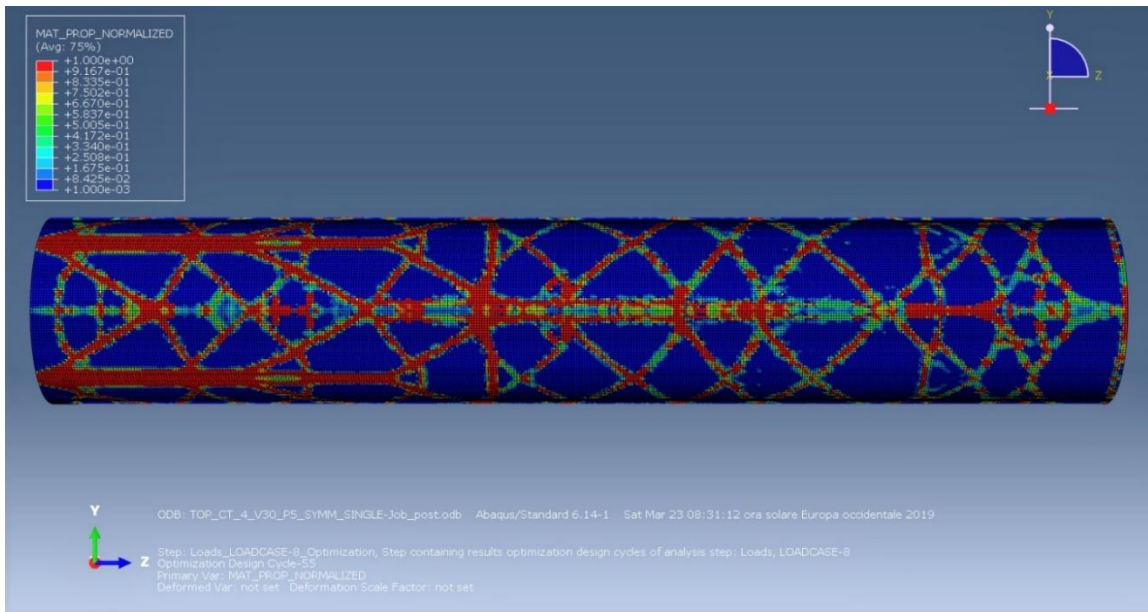


Figure 10.8 ABAQUS model, four clamped points lateral view

Eight clamped points:

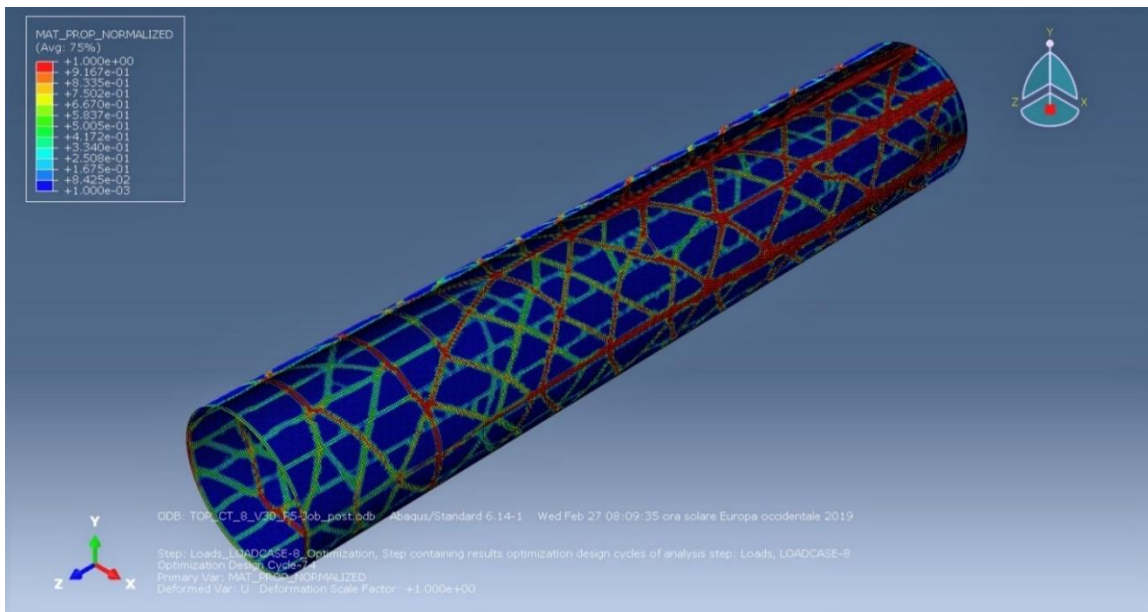


Figure 10.9 ABAQUS model, eight clamped points general view

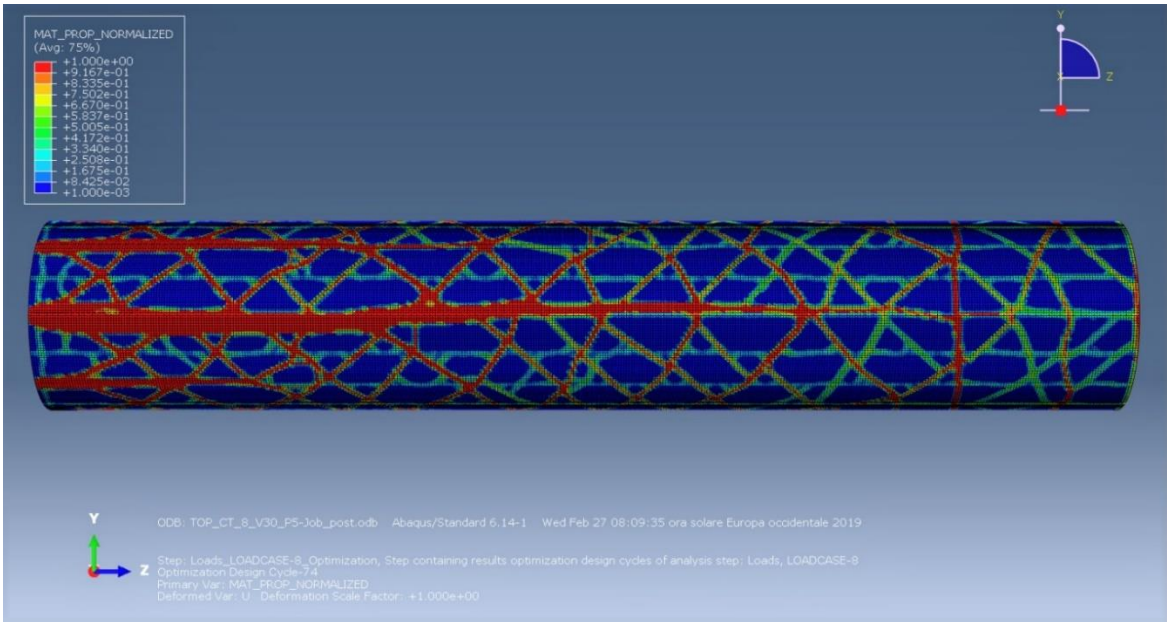


Figure 10.10 ABAQUS model, eight clamped points lateral view

Sixteen clamped points:

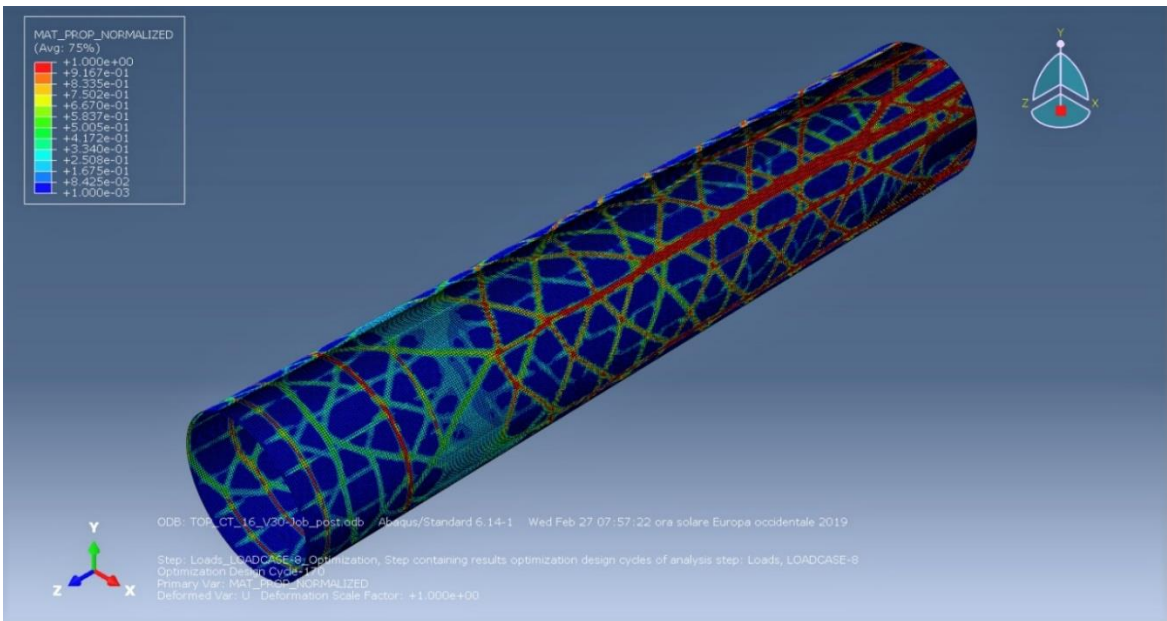


Figure 10.11 ABAQUS model, sixteen clamped points general view

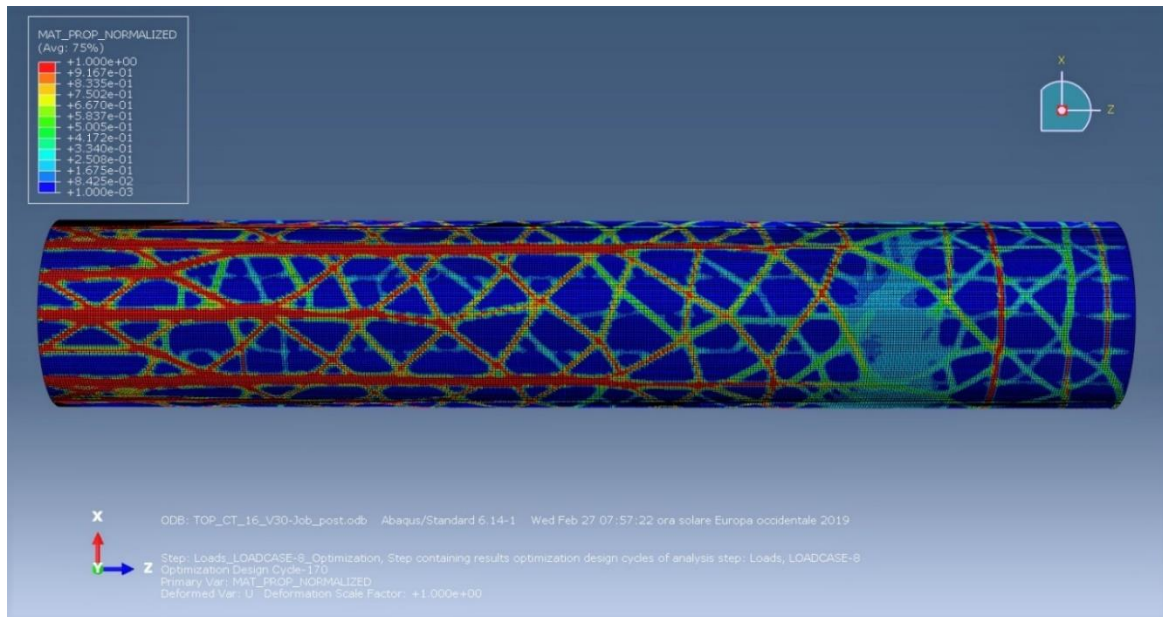


Figure 10.12 ABAQUS model, sixteen clamped points lateral view

It is clear that from each constraint points rise vertical elements till it is necessary to sustain the overturning moment, and from a give height, when the shear becomes influencer, diagonals rise.

If the principal stresses trajectories are considered, this behaviour of the tube doesn't surprise. Considering the tension curves obtained from the square tube, they both form the entrance point of the load vertically and then they move around the body moving form a  $0^\circ$  to a  $45^\circ$ , and at the end to  $90^\circ$ . So, if a polar symmetry is considered the curves result are a set of lines that rise vertically from the base and gradually come at  $45^\circ$  till the top.

This conclusion about diagrid circular tube layout is deduced also from M. Sarkisian at all [35] for the lateral system cables layout around the tube shape of Al Sharq Tower.

It is henceforth concluded that for diagrid systems, the presence of vertical elements at the base is required for an optimal efficiency.



## 11. Bibliography

- [1] A. Beghini, M. Sarkisian. (s.d.). Geometry optimization in structural design.
- [2] A. Pugsley, O. Saunders. (s.d.). Collapse of flats at Ronan Point, Canning Town.
- [3] Allinger, P. (s.d.). The TOSCA-Topology optimisation, mesh adaption and manufacturing restrictions -.
- [4] Allison B. Halpern, David P. Billington, Sigrid Adriaenssens. (2013). The Ribbed Floor Slab Systems of Pier Luigi Nervi. *"BEYOND THE LIMITS OF MAN"*. J.B. Obrebski and R. Tarczewski.
- [5] B. T. Dat, A. Traykov, M. Traykova. (s.d.). Shear-lag effect and its effect on the design of high-rise buildings.
- [6] Banu, O. M. (s.d.). Structural robustness provisions in modern design codes and regulations.
- [7] Bhathee, V. (s.d.). Diagrid structural system.
- [8] BS. (s.d.). CEN EN 1991-1-7 - Eurocode 1 - Actions on structures, Part 1-7: General actions-Accidental actions.
- [9] C. Pearson, N. Delatte. (s.d.). Ronan Point apartment tower collapse and its effect on building Codes.
- [10] Chan, A. S. (1960, December). The Design of Michell Optimum Structure, s. *Reports and Memoranda No. 3303*.

- [11] Csébfalvi, A. (s.d.). Structural optimization under uncertainty in loading directions, benchmark results.
- [12] D. A. Foutch, P. C. Chang. (1982, July). A shear lag anomaly.
- [13] E. Reissner, F. B. Hildebran. (s.d.). Least-work analysis on the problem of shear-lag in box beams.
- [14] E. S. Cavaco, J. R. Casas, L. A. C. Neves, A. E. Huespe. (s.d.). Robustness of corroded reinforced concrete structures a structural performance approach.
- [15] G. I. N Rozvany, T. Lewiński. (s.d.). Topology optimization in structural and continuum mechanics - 2014.
- [16] G. I. N. Rozvany, T. Sokół, V. Pomezanski, Z. Gaspar. (s.d.). Extension of Michell's classical (1904) truss topology optimization theory to multiple load condition.
- [17] G. Kharmanda, N. Olhoff, A. Mohamed, M. Lemaire. (s.d.). Reliability-based topology optimization .
- [18] Ghandriz, T. (s.d.). An algorithm for for structural topology optimization for multibody systems.
- [19] Guy Bouchitté, W. G. (2008). Michell Trusses and Lines of Principal Action. In *Mathematical Models and Methods in Applied Sciences Vol. 18 No. 9* (p. 1571–1603). World Scientific Publishing Company.
- [20] Husainie, S. N. (s.d.). Weight optimization of a landing gear steering collar using Tosca in Abaqus.

- [21] J. Liu, G. Wen, X. Chen. (s.d.). Topology optimization of continuum structures with uncertainty in loading direction.
- [22] J. Lógó, B. Balogh, E. Pintér. (s.d.). Topology optimization considering multiple loading.
- [23] J.J. Kingman, K. D. Tsavdaridis, V. V. Toropov. (s.d.). Applications of topology optimization in structural engineering.
- [24] Jacot, B. (s.d.). Three dimensional optimal Michell structures.
- [25] Johnsen, S. (s.d.). Structural topology optimization, basic theory, methods and applications.
- [26] K. A. James, J. S. Hansen, J. R.R.A. Martins. (s.d.). Structural topology optimization for multiple load cases using a dynamic aggregation technique.
- [27] K. Brittain, M. Silva, D. A. Tortorelli. (s.d.). Minmax topology optimization.
- [28] L. L. Stromberg, A. Beghini, W. F. Baker, G. H. Paulino. (s.d.). Application of layout and topology optimization using pattern gradation for the conceptual design of buildings.
- [29] L. L. Stromberg, A. Beghini, W. F. Baker, G. H. Paulino. (s.d.). Design of structural braced frames using topology optimization - SLIDES.
- [30] L. L. Stromberg, A. Beghini, W. F. Baker, G. H. Paulino. (s.d.). Topology optimization for braced frames, combining continuum and beam-column elements.

- [31] L. L. Stromberg, A. Beghini, W. F. Baker, G. H. Paulino, A. Mazurek. (s.d.). A new frontier in modern architecture, optimal structural topologies.
- [32] Larsson, R. (s.d.). Methodology for topology and shape optimization application to a Rear Lower Control Arm.
- [33] Leonard, J. (s.d.). Investigation of shear-lag effect in high-rise buildings with diagrid system.
- [34] M. P. Bendsøe, O. S. (s.d.). *Topology Optimization: Theory, Methods and Applications*. Denmark: Springer.
- [35] M. Sarkisian, E. Long, C. S. Doo. (s.d.). Optimization tools for the design of structures.
- [36] Maxwell, C. (1870, February). On Reciprocal Figures, Frames and Diagrams of Forces.
- [37] Michell, A. G. (1904, November). The Limits of Economy of Material in Frame-structures. p. 589-597.
- [38] Mitchell, T. (s.d.). A limit of economy of material in shell structures.
- [39] N. D. Lagaros, L. D. Psarras, M. Papadrakakis, G. Panagiotou. (s.d.). Optimum design of steel structures with web openings.
- [40] O. Sigmund, E. Andreassen, A. Clausen, M. Schevenels, B. s. Lazarov. (s.d.). Efficient topology optimization in MATLAB using 88 lines of code.
- [41] O. Sigmund, N. Aage, E. Audreassen. (s.d.). On the (non)-optimality of Michell structures .

- [42] P. D. Dunning, H. A. Kim, G. Mullineux. (s.d.). Introducing loading uncertainty in topology optimization.
- [43] P. Dewhurst, D. G. (2009). Three-dimensional cylindrical truss structures:. *WIT Transactions on The Built Environment, Vol 106*.
- [44] P. Lee, E. Long, D. Shook, M. Sarkisian. (2010). Organic and natural forms in building design. *Research Paper*.
- [45] R. S. Iwamura, A. R. De Faria. (s.d.). Topology optimization of multiple load case structures.
- [46] Reissner, E. (s.d.). Analysis of a shear lag in box beams by the principle of minimum potential energy.
- [47] Ritchie, I. (2012). Diagonal Architecture : Diagrid Structures.
- [48] Rozvany, G. I. (s.d.). Aims, scope, methods, history and unified terminology of computer-aided topology optimization in structural mechanics.
- [49] S. Daynes, S. Feih, W. F. Lu, J. Wei. (2017, pril). Optimization of functionally grade lattice structures.
- [50] Sigmund, O. (s.d.). On convergence speedup in topology optimization.
- [51] U. Starossek, M. Haberland. (s.d.). Approaches to measures of structural robustness.
- [52] V. O. Pomezanski, O. Querin, G. I. N. Rozvany. (s.d.). CO-SIMP extended SIMP algorithm with direct COrner COntact COntrol.

- [53] W. Baker, A. Beghini, J. Carrion, C. Tort. (s.d.). Studies in structural optimization.
- [54] W. F. Baker, L. L. Beghini, A. Mazurek, J. Carrion, A. Beghini. (s.d.). Maxwell's reciprocal diagrams and discrete Michell frames.
- [55] W. F. Baker, L. L. Beghini, A. Mazurek, J. Carrion, A. Beghini. (s.d.). Maxwell's reciprocal diagrams and discrete Michell frames.
- [56] William F Baker, Alessandro Beghini, Arkadiusz Mazurek. (2012). Applications of Structural Optimization in Architectural Design. ResearchGate.
- [57] Wojciechowski, M. T. (2016). *Determining Optimal Geometries of Plane Stress Truss Structures using Numerically Mapped Principal Stress Trajectories*. Raleigh.
- [58] Woliński, S. (s.d.). Defining of the structural robustness.
- [59] X. Peng, J. Li, S. Jiang, Z. Liu. (s.d.). Robust topology optimization of continuum structures with loading uncertainty using a perturbation method.
- [60] Y. Lee, Y. Chen. (s.d.). Beam structure optimization for additive manufacturing based.
- [61] Yoshida, S. (s.d.). Michell truss: shape optimization vs. analytical solution.
- [62] Z. Jaouadi, T. Lahmer. (s.d.). Topology optimization of structures subjected to multiple load cases by introducing the Epsilon constraint method.

