POLITECNICO DI TORINO

Master of Science in Mechanical Engineering

Master's Degree Thesis

Modeling of an auxetic lattice structure fabricated through Additive Manufacturing

Supervisors

Prof. Aurelio SOMÀ

Prof. Fabrizio SCARPA

Candidate

Davide PELLIZZI

2019-2020

Summary

Additive Manufacturing (AM) is a cutting-edge technology that permits the fabrication of mechanical parts characterized by very complex geometries and provides also good mechanical properties to final products. However, when the component to be produced is small-sized and high-detailed AM still presents some limitations. In particular, lattice structures produced by AM show dimensional inaccuracies compared to the designed CAD model, so the component is compromised also from a structural point of view. Thus, there is a substantial discrepancy between the mechanical properties of the designed lattice structure and the fabricated one.
In an accurate design process, this difference cannot be neglected: this paper explores which are the most common defects introduced by AM and how they affect mechanical behavior. Furthermore, this work aims to create a simulation model able of including the fabrication imperfections to have a prediction of mechanical properties of the component closer to reality. Since these fabrication defects strongly depend on the design feature of every single strut, the simulation model has been developed through a script in Matlab which in turn generates a parametric code to be run in ANSYS APDL.
This study also proposes a "Design of Experiments" to detect correlations between geometrical and mechanical properties of lattices. Thanks to this approach it has been designed an auxetic lattice thought to substitute a honeycomb in a typical sandwich structure. Lastly, the discrepancy between the mechanical properties of the designed structure and those predicted for a fabricated one has been highlighted. So, to close this structural gap, an optimization of the structure has been proposed.

Acknowledgements

In primis, vorrei ringraziare i Professori Aurelio Somà e Fabrizio Scarpa per avermi concesso la possibilità di svolgere il mio periodo di tesi a Bristol e di vivere una esperienza indimenticabile che mi ha fatto crescere sotto ogni aspetto. Insieme a loro vorrei ringraziare anche il Dr. Giuliano Allegri per tutto il tempo che mi ha dedicato, guidandomi quotidianamente. I loro insegnamenti sono stati di immensa importanza per me.

Devo tutto a Mamma e Papà per avermi regalato questo importante percorso di crescita. Senza il sostegno e la fiducia che mi hanno mostrato ogni singolo giorno non avrei mai raggiunto questo traguardo. Ogni parola non sarebbe mai abbastanza per esprimere tutta la mia gratidudine nei loro confronti.

Un grazie speciale va a mio fratello, il mio porto sicuro. Complice di mille sciocchezze. Il suo affetto e i suoi consigli erano, sono e saranno fondamentali nella mia vita.

Grazie a Davide, per essere stato testimone e protagonista di ogni tappa della mia vita. L'amico che ho sempre desiderato.

Grazie a Maria, un tesoro da sempre. Un rapporto così è più unico che raro.

Grazie a Francesco, con cui ho condiviso tutti i miei anni a Strongoli e a Torino. Amico schietto e sincero di cui non posso fare a meno.

Grazie al mio collega Marco Sessa. Uomo dotato di infinita pazienza. A lui, che ha ascoltato assorbito e sopportato ogni mio sfogo senza mai giudicarmi, va grande merito di ogni mio risultato raggiunto durante questa esperienza universitaria. Una persona speciale, ma pur sempre un collega.

Grazie a Francesco, Adelina, Edoardo, Annachiara, Andrea, Luca Zocco, Gianluca e Salvatore, miei compagni di avventura diventati punti di riferimento per me. I bei ricordi del periodo universitario saranno associati inevitabilmente a loro.

Grazie ai miei meravigliosi coinquilini Roberto, Graziano, Giampiero e Federica. Non potevo desiderare compagnia migliore dopo le stressanti giornate trascorse al Politecnico.

Grazie ad Alessia, Giuseppe e Gaetano. Amici sempre distanti ma infinitamente vicini.

Infine ma non per ultimo, un grazie speciale alle mie care nonne e tutti i familiari che non hanno perso occasione per mostrarmi fiducia e affetto sincero. La loro vicinanza è stata una grande risorsa di energia durante questi anni impegnativi.

Table of Contents

List of Tables

List of Figures

Nomenclature

Acronyms

AM Additive Manufacturing

APDL Ansys Parametric Design Language

CAD Compurer Aided Design

CM Conventional Manufacturing

CM Conventional Manufacturing

DED Directed Energy Deposition

DOE Design of Experiments

EBM Electron Beam Melting

FE Finite Element

FEM Finite Element Method

L-PBF Laser Powder Bed Fusion

MLS Metal Laser Sintering

PBF Powder Bed Fusion

PDE Partial Differential Equation

RD Relative Density

SL Stereolithography

SLM Selective Laser Melting

SLS Selective Laser Sintering

Chapter 1

Introduction

Cellular solids represent a class of materials that are particularly appreciated in the engineering fields: their high strength to weight ratio make these materials very suitable for structural applications. Furthermore, with the advent of cutting-edge fabrication processes as Additive Manufacturing, these materials are becoming increasingly used in several fields, such as biomedical, automotive, and aerospace. Unlike other materials, they are unique in the customizability of mechanical properties. Since they are made of a series of cells, it has been noticed that there is a relationship between geometrical features of the latter with the mechanical properties of the whole designed structure.
The first cellular solids that appeared in engineering applications were 2D dimensional ones and the most famous are certainly honeycomb structures. However, nowadays manufacturing technologies permit to build structures with very complex geometry without affecting the production costs. So, as the years go by engineering designers try to develop continuously more sophisticated structures with the main objective of optimizing the desired mechanical properties reducing as much as possible the weight of the overall component. For this reason, many bidimensional and three-dimensional cellular solids characterized by different topologies have been developed.
In particular, 2D cellular solids present the alignment of cells only in two dimensions, while the third dimension is built by extruding the cell walls of the unit cells. The turning point has been reached when Additive Manufacturing allowed industries to substitute the cell walls with struts, providing thus even the possibility to line up the unit cells along with the third dimension. This development allows the designer to build even lighter structures, the so-called non-stochastic lattice structures.
On the other hand, the geometrical intricacy of the latter requires a higher level of accuracy of manufacturing processes. Even if this challenge is largely matched by Additive Manufacturing technologies when a thin-struts lattice must be created, many geometrical inaccuracies can occur during the fabrication process which in

turn can compromise the mechanical properties of the designed structure, thus its suitability for the intended application. For this reason, an accurate prediction of mechanical properties becomes difficult.
For this purpose, simulation models are generally processed by Finite Element analysis from the CAD model of the designed structure, thus by using solid elements. Even if this approach is quite accurate, it can be too time expensive, especially when geometrical variations caused by manufacturing processes are tried to be reproduced.
This work aims to create a simulation model made of beam elements that can also include geometrical variations for an accurate prediction of mechanical properties of the final product. Since this work has been carried out at ACCIS (*Advanced Composites Collaboration for Innovation and Science*) of the University of Bristol, a lattice structure for the replacement of a honeycomb one has been taken as a case study. Considering that this structure has been thought to be produced via EBM the original material of the honeycomb, stainless steel 316L, is kept the same also for the lattice structure. Thus, this paper does not focus on the selection of the most suitable material to be used for the optimal structure.
Furthermore, the design phase must optimize the mechanical properties only along one direction so the advantages to use an auxetic configuration are also studied. The most suitable structure has been figured out by adopting a Design of Experiment, in the way to make clear also the correlations existing between the geometrical parameters of the cells.
It is worth saying that this work has not been verified experimentally so this numerical investigation only aims to give modeling and design methods, and to highlight the issues can come from an Additive Manufacturing process.

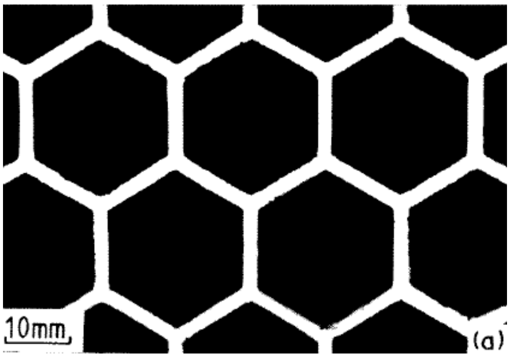
Chapter 2

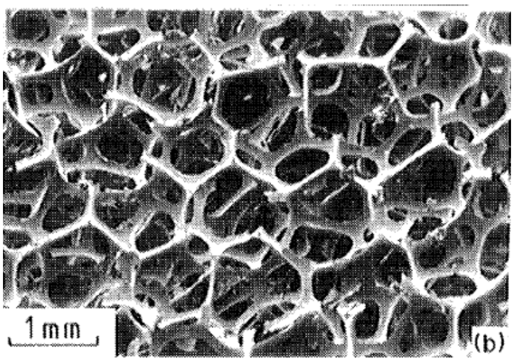
Cellular solids

2.1 Introduction

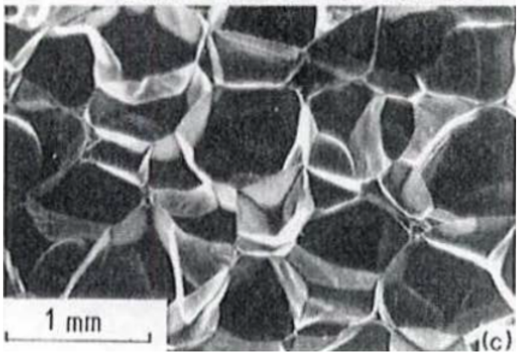
Nature has always been inspiring for humans because it offers a wide variety of different materials characterized by particular mechanical properties. Among all of them, there is a class of material deserving special attention, cellular solids. Their name derives from the Latin word *"cella"*, which means a small compartment. This is the perfect definition to give an idea of how cellular solids are shaped because they appear in fact as a series of packed cells forming a whole structure.
Despite their unusual geometry, cellular materials are quite easy to be found in nature and historically they always represented a precious resource for the human being: glaring examples are wood and cork. For instance, the former was used by Egyptians to build pyramids while the latter by the ancient Romans to make caps for wine bottles. As can be deduced from the first example, these kinds of materials show high structural capability against a low weight if compared to conventional material used in structural applications. This peculiarity meets exactly the modern engineering challenges, decrease of weight, and this is the reason why Man started to create cellular solids artificially.
The series of unit cells can develop in two dimensions as the case of *honeycomb structures* or in three dimensions as for *foams*. The latter in turn, can be open-celled if the unit cells are made only of edges and closed-celled if the faces of cells are wall-like.

As imagined so far, cellular solids can be named so only if the material shows a minimum level of porosity. In particular, this parameter is measured in terms of *relative density* $RD = \rho/\rho_s$, where ρ is the density of the cellular solid and ρ_s is the density of the bulk material from which cellular solid is made. Thus, to satisfy the announced minimum level of porosity, RD should be lower than 0.3. From what it has been said so far it is clear that the stiffness of cellular solids is much lower than that of material they are made of. However, they are appreciated because of their lightness. Furthermore, their high porosity, in particular for foams, turned out to be a great advantage in terms of absorption properties. Nowadays they are widely used as acoustic and thermal insulators but also as impact absorbers. Despite all





(a) "Two-dimensional honeycomb" (b) "three-dimensional foam with open cells"



(c) "three-dimensional foam with closed cells"

Figure 2.1: Cellular solids [1]

these advantages, foams are not preferred for structural applications because since their geometry cannot be accurately controlled, their mechanical properties cannot be customized and this represents a big limit.
As a consequence, engineers have always tried to develop personalized cellular structure but their geometric complexity has represented an obstacle for the fabrication process. With the advent of Additive Manufacturing (AM) the target of geometric complexity could be fabricated has changed radically and in turn, it has brought a revolution also in the way to conceive cellular solids. Unit cells can indeed be repeated in two and three dimensions with the possibility to give them the desired shape without affecting production costs.

In fact, nowadays man-made cellular solids are widely used in aerospace, automotive, and biomedical applications. However, the target of reducing the weight of the components is continuously pushed further leading thus to the necessity of designing even more complex geometries.
In conclusion comparing their enormous applicative potential and their current effective number of applications, further research on them has still to be made.

2.2 Unit cells

Cellular solids are interesting to study because their mechanical macro-properties strictly depend on geometrical features. In particular, *cell shape* is the most relevant factor that determines the mechanical properties of the overall structure. For example, honeycomb structures show isotropic properties because of the symmetry provided by the hexagonal unit cell. Irregularly shaped cells give anisotropy instead. Thus, two structures made of the same bulk material but with two different shapes of cells show certainly different mechanical properties. Many examples are reported in Figure 2.2. Furthermore, the shape of cells is not sufficient to roughly predict how mechanical properties are distributed along with different directions but the way they are positioned respect to each other, thus the *topology*, Figure 2.3, is also an important feature. So, a multitude of geometric solutions makes cellular solids very attractive from a design point of view.

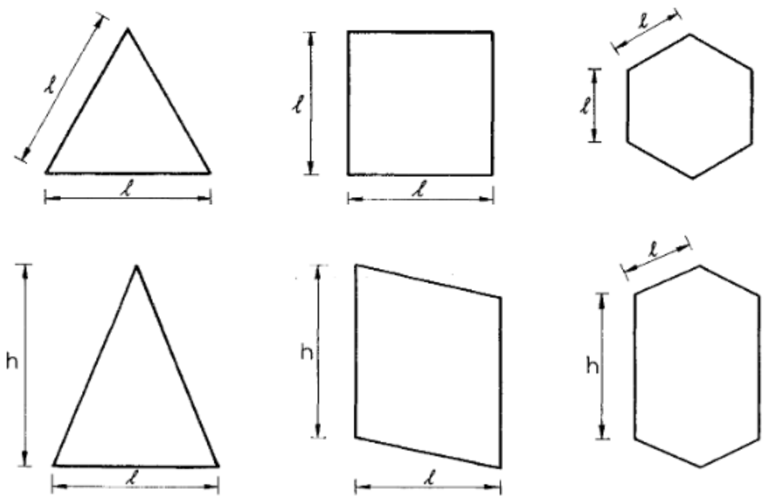


Figure 2.2: Common two-dimensional unit cell geometry [1]

It is worth saying that cells characterizing a cellular solid may not be all the same. When a structure presents all identical cells each of them is called *unit cell*, if

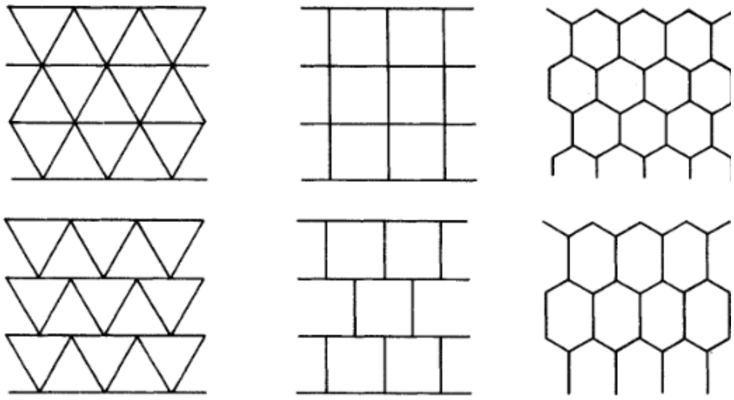


Figure 2.3: Common two-dimensional topologies [1]

not they are simply named *cells*. The first case is more common for two-dimensional cellular solids while the second one is more attributable to three-dimensional cellular solids as foams.

However in recent years, improvement in the accuracy of manufacturing technologies is allowing to design uniform three-dimensional cellular structures that take the name of *lattices*. This latter is built by replicating a three-dimensional unit cell, shown in Figure 2.4, in the space. These kinds of structures differ from honeycomb-like ones because they evolve in the third dimension through interconnected struts rather than cell walls. This solution favors even more material saving and reduction of weight as well.

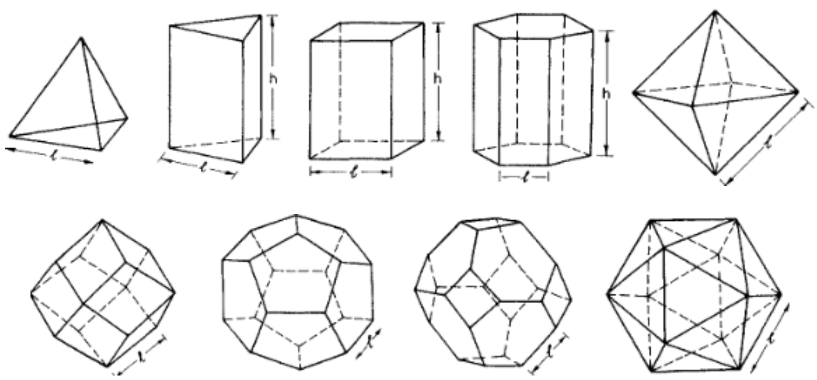


Figure 2.4: Common three-dimensional unit cells [1]

2.3 Mechanics

So far it has been seen that cellular solids can have different macro-mechanical properties depending on shapes of cells and topology of the structure. If the problem is analyzed from a microscopic point of view, cells with dissimilar geometries can even undergo completely different deformation mechanisms. On the base of this assertion, *Deshpande et al* [2] proposed a classification of lattice structures according to their dominant deformation mechanism: *bending-dominated* and *stretching-dominated*. To better understand this categorization it is worth introducing two groups of interconnected struts, as in Figure 2.5.

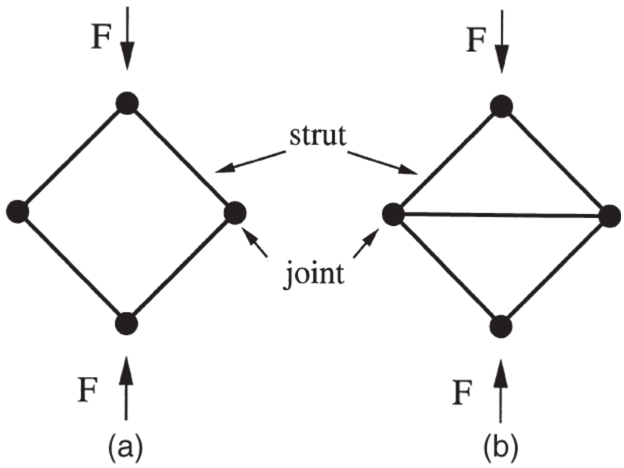


Figure 2.5: (a) Mechanism; (b) Structure [2]

If the interconnections of struts are seen as pin-joints, the assembly in Figure 2.5a under loading condition undergoes to kinematic rotations of struts so it does not have any stiffness, this is called *mechanism*. On the contrary, the assembly in Figure 2.5b under load generates reaction forces axially to struts. This is called *stretching dominated structure*.
However lattice structures do not have pin-jointed edges but they can be considered rigid. So if the rotation at the joints is locked, the struts of mechanism in Figure 2.5a start to deform by bending. The so-called *bending-dominated structure*. While the structure in Figure 2.5b, even under rigidly connected struts hypothesis, keeps giving mainly axial deformation.
To univocally determine the deformation mechanism of a 3D structure, *Deshpande et al* [2] suggest to refer to the Maxwell stability criterion:

$$M = b - 3j + 6 \tag{2.1}$$

As they explain this *"algebraic rule sets out the condition for a pin-jointed frame of b struts and j frictionless joints to be both statically and kinematically determinate ie. to just be rigid"*. Effectively this rule is not revealing if the structure is bending or stretching dominated but only whether a set of non-rigidly interconnected struts is a mechanism or a structure. However, once the category of the assembly is identified, it is possible to determine also the dominant deformation mechanism just by introducing the rigidly-jointed condition.

To give a mathematical interpretation to the problem, it is sufficient to analyze Eq.2.1:

- $M < 0$ the pin-jointed assembly is a mechanism. So under rigidly-jointed condition, it turns out to be a bending dominated-structure
- $M >= 0$ the pin-jointed assembly is a structure. So under rigidly-jointed condition, it turns out to be a stretching-dominated structure

Just for sake of clarity, let's make a brief overview to spot the mechanical differences between these two types of structures.

Bending-dominated structure

Bending dominated structures are characterized by particular topological configurations that favor the bending of cell edges. An example of open-cell foam is reported in Figure 2.6. The just mentioned image shows how axial reaction forces

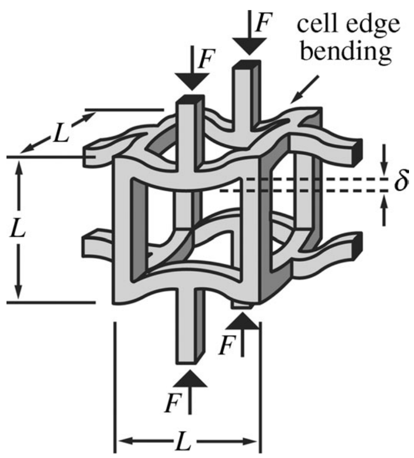


Figure 2.6: Open-cell foam [3]

of struts are exerted perpendicularly each other, then forcing the cell edges to bend. Thus the force F exerted perpendicularly to a strut of length L, with squared cross section of side t, and made of a specific material with Young modulus E_s produces a deflection of δ:

$$\delta = \frac{FL^3}{E_s I} \tag{2.2}$$

where $I = t^4/12$ is the area moment of inertia of struts. Furthermore *Ashby* [3] also found a correlation between the elastic modulus of the lattice and its relative density:

$$\frac{E}{E_s} \propto (RD)^2 \tag{2.3}$$

If the analysis is extended to all deformation field of the cell until the collapse is reached, Figure 2.7 shows the large plateau a bending-dominated structure can provide in the field of plastic deformations. Thus, considering the high deformation capability this kind of structure is very suitable for energy absorption applications.

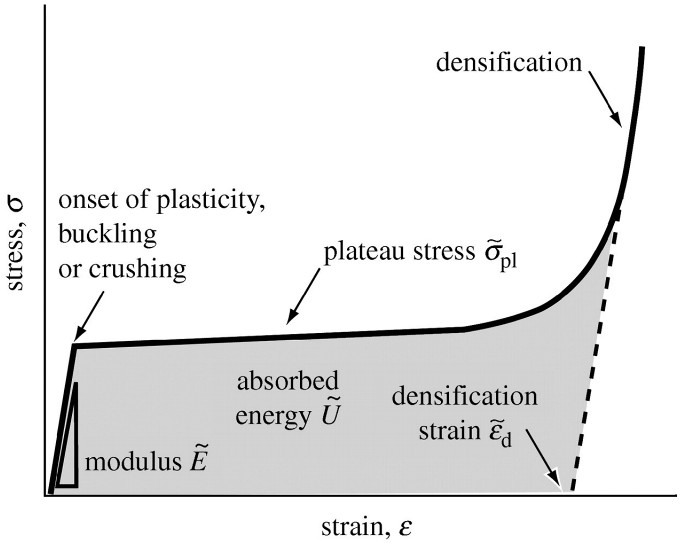


Figure 2.7: Stress-strain plot for bending dominated lattice [3]

Stretching-dominated structure

The elements characterizing a stretching-dominated structure deform mainly by stretching or compression. An example is provided in Figure 2.8. Here, there is no flexural displacement δ but Young's modulus has been found to have the following correlation with RD:

$$\frac{E}{E_s} \approx \frac{1}{3}(RD) \tag{2.4}$$

This time the most important mechanical feature can be noticed in Figure 2.9 is the high stiffness and strength shown in elastic field.

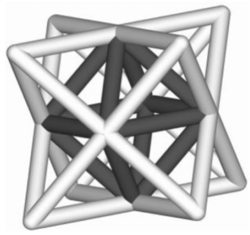


Figure 2.8: Stretching-dominated unit cell [3]

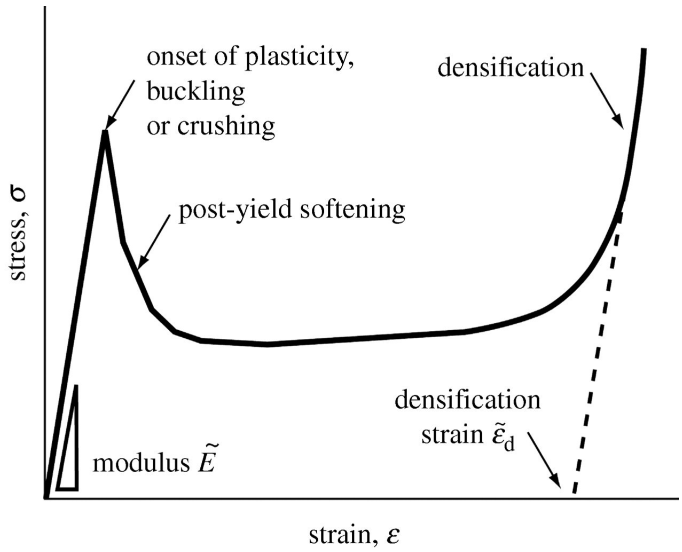


Figure 2.9: Stress-strain plot for stretching dominated lattice [3]

Comparison between stretching-bending dominated structures

In conclusion, it is possible to say that stretching-dominated structures show a higher strength and onset of plasticity compared to bending-dominated structures. But, since the former withstand very high stresses, when they reach buckling or struts collapse the whole structure undergoes a sudden softening effect. So this behavior is not suitable for high energy absorption application.
Furthermore the relationships found in Eq. 2.3 and 2.4 are plotted in Figure 2.10 in a logarithmic scale.

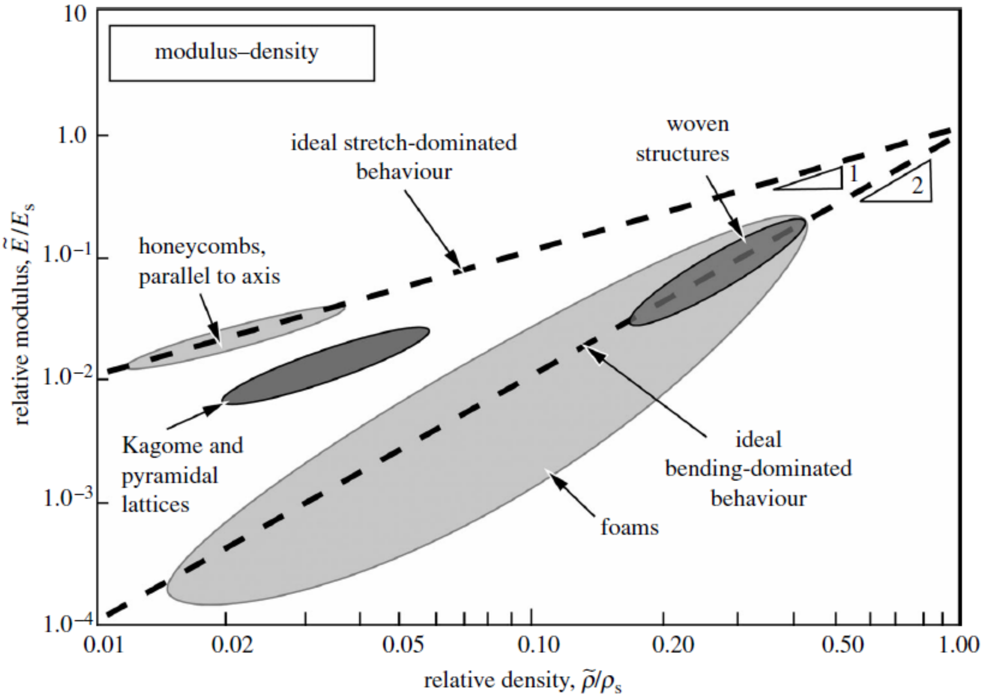
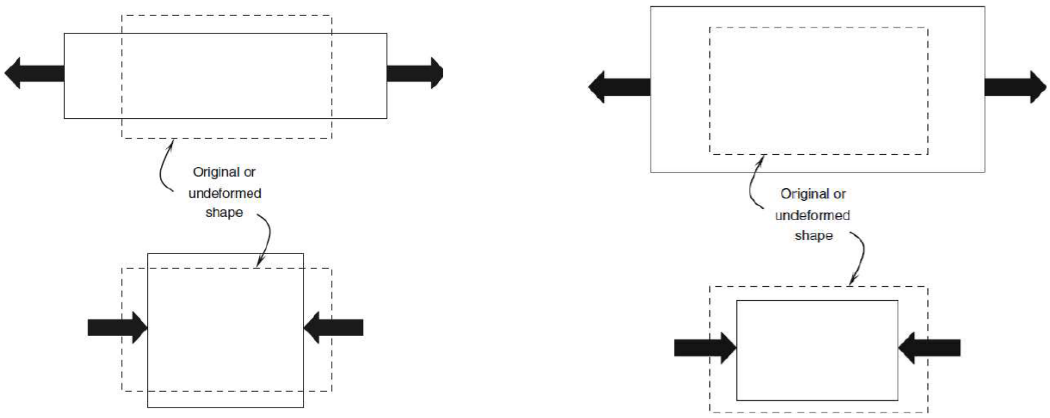


Figure 2.10: Relative elastic modulus plotted against *RD* on a logarithmic scales [3]

Auxeticity

So far, it has been clearly said micro-deformation mechanism of cell edges affects the macro-mechanical properties of all structures. The relationship between microscopic properties and macroscopic ones is correlated only to geometrical features.
The difference between micro and macro is marked when unit cells are built in the way to give "auxeticity" to the structure. The term *"auxetic"* derives from the Greek word *"auxetikos"* which means "that increases" and it has been introduced in the literature of cellular solids for the first time in 1991 by Prof. Evans from the University of Exeter [4]. As shown in Figure 2.12b, when an auxetic body is under stretching load it produces an elongation also in the transversal direction. Thus the auxeticity is strictly related to the macroscopic deformation behavior of a determined body. For this purpose, it is possible to define a coefficient that measures the grade of deformability of transversal cross-section respect to that of the section lying on the plane parallel to the loading direction. This is the Poisson's ratio:

$$\nu = -\frac{\varepsilon_{trans}}{\varepsilon_{load}} \tag{2.5}$$



(a) Response of a conventional material to stretching(above) and compression (below) loads

(b) Response of an auxetic material to stretching(above) and compression (below) loads

Figure 2.11: Deformation mechanisms of a conventional (a) and auxetic material (b)

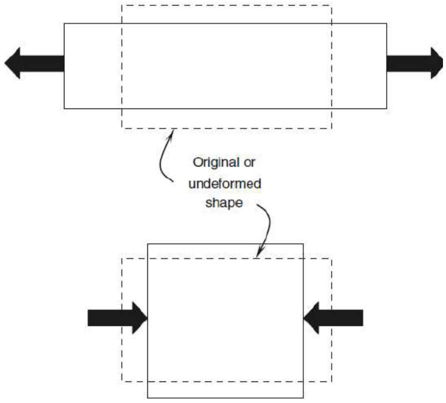
where ε_{trans} is the transversal deformation and ε_{load} is the deformation along loading direction. Furthermore, for isotropic material, it can link Elastic E and Shear modulus G as follows:

$$E = 2G(1 + \nu) \tag{2.6}$$

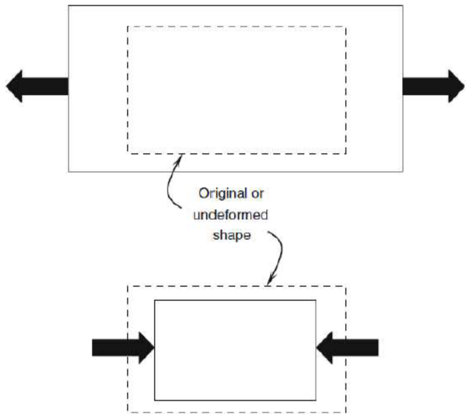
So far, it has been clearly said micro-deformation mechanism of cell edges affects the macro-mechanical properties of all structures. The relationship between microscopic properties and macroscopic ones is correlated only to geometrical features.
The difference between micro and macro is marked when unit cells are built in the way to give "auxeticity" to the structure. The term *"auxetic"* derives from the Greek word *"auxetikos"* which means "that increases" and it has been introduced in the literature of cellular solids for the first time in 1991 by Prof. Evans from the University of Exeter [4]. As shown in Figure 2.12b, when an auxetic body is under stretching load it produces an elongation also in the transversal direction. Thus the auxeticity is strictly related to the macroscopic deformation behavior of a determined body. For this purpose, it is possible to define a coefficient that measures the grade of deformability of transversal cross-section respect to that of the section lying on the plane parallel to the loading direction. This is the Poisson's ratio:

$$\nu = -\frac{\varepsilon_{trans}}{\varepsilon_{load}} \tag{2.7}$$

where ε_{trans} is the transversal deformation and ε_{load} is the deformation along loading direction. Furthermore, for isotropic material, it can link Elastic E and Shear



(a) Response of a conventional material to stretching(above) and compression (below) loads



(b) Response of an auxetic material to stretching(above) and compression (below) loads

Figure 2.12: Deformation mechanisms of a conventional (a) and auxetic material (b)

modulus G as follows:

$$E = 2G(1 + \nu) \tag{2.8}$$

Furthermore, it must be said that ν can take values between -1 and 1 and it indicates auxeticity when it reaches negative values.
Now that the macroscopic auxetic behavior is clearly explained, it is interesting to observe what can produce this peculiar deformation mechanism. This work is focused on cellular solids so let's see how this class of material can undergo auxeticity.
Looking at the honeycomb structure illustrated in Figure 2.13a, it possible to notice that the whole structure shows a typical deformation behavior. While if the cell struts are set as re-entrant as in Figure 2.13b, the structure behaves like an auxetic material. So it is evident that the orientation of cell struts determines the auxeticity of the whole structure. Obviously, the one shown in Figure 2.13 is not the only solution and actually, the negative Poisson's ratio can be achieved by using a multitude of different unit cells. Many simple two-dimensional configurations are shown in Figure 2.14.

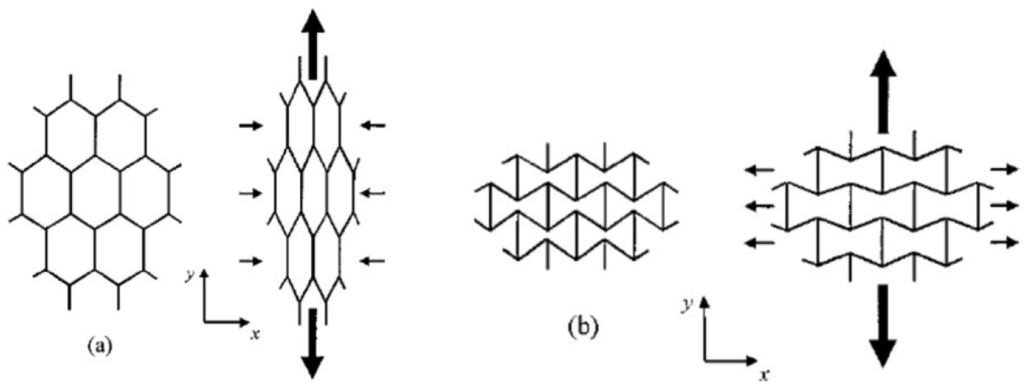


Figure 2.13: Regular honeycomb (a) and an auxetic one (b)

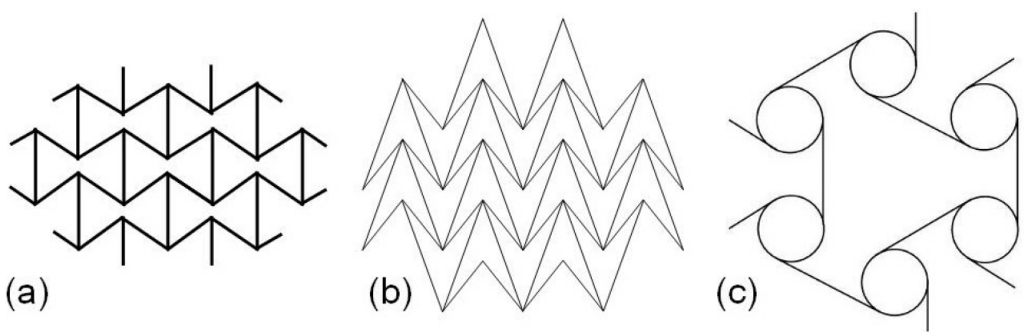


Figure 2.14: Two dimensional auxetic topologies: re-entrant hexagonal (a), arrowhead (b), and chiral (c) [5]

Chapter 3

Manufacturing processes

Lattice structures are characterized by very complex geometrical architectures, so a very high resolution is required to fabricate them adequately. The most cutting edge technology able of producing parts regardless of their particular geometrical shapes is AM.

3.1 Additive Manufacturing

Additive technologies are defined as processes that create three-dimensional parts by depositing material layer by layer. This manufacturing procedure allows the production of components without process planning. In particular, through AM it is possible to fabricate parts by following mathematical models implemented from Computer-Aided Design (CAD) data.
This technology has made its debut in 1986 when Chuck Hull discovered that "by exposing UV-curable materials to scanning laser, solid polymers patterns could be produced" [6]. So he invented the first 3D printing technology, Stereolithography (SL) (a schematic representation is shown in Figure 3.1). In the beginning, it was used for "Rapid Prototyping" because it could work only polymeric materials but this solution was appreciated by industries because prototypes production allowed them to give shape to ideas and provide evaluations during the development phase. This way of building objects has been continuously improved over the years so many new manufacturing machines have been invented and manufacturing processes optimized. In particular, nowadays 3D printing is even able to create metallic and ceramic parts with good mechanical properties. So the rule of Additive Manufacturing passed from a prototyping procedure to an actual fabrication process of final products.
The advantages taken from this change are several and as can be observed in Figure 3.2 the most important is the ability to accomplish very complex geometries

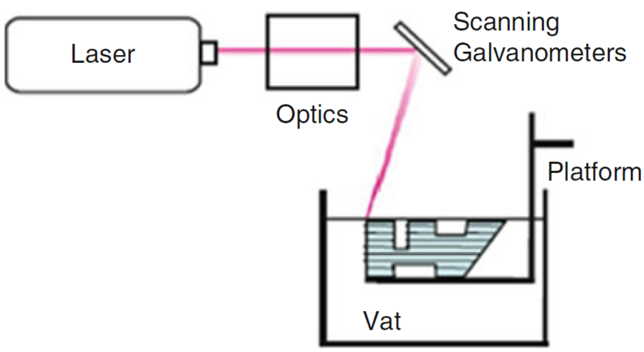


Figure 3.1: Schematic diagram of SL [6]

without increasing fabrication costs. On the other hand, this technology can not

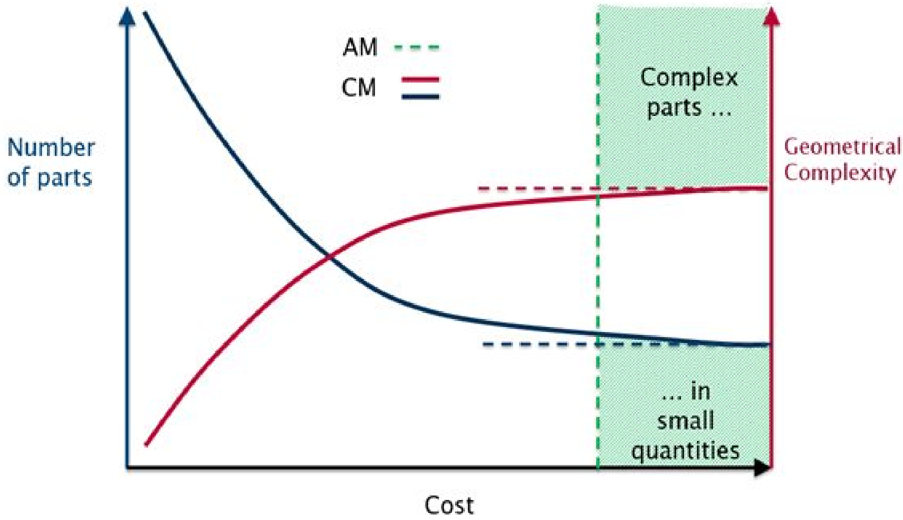


Figure 3.2: Comparison of fabrication costs between Conventional Manufacturing (CM) and AM technologies as a function of geometrical complexity and number of the parts to be produced

replace completely the CM technologies because it has two big drawbacks. The first one is the machine speed is still limited, so the production process is time expensive. The second one is the possibility to handle only a few materials and most of the time, the addition of new material with an AM technology is more expensive than doing it with conventional technology.
So, in summary, Additive Manufacturing is very suitable for the fabrication of small-sized components with very complex geometry but it is not satisfying for a long series of production because too slow.

As indicated by ASTM there are several layer-by-layer building-based processes such as Vat Photopolymerization, Powder Bed Fusion, Extrusion-Based, Material Jetting, Binder Jetting, Sheet Lamination, Direct Energy Deposition, and Direct Write. All the listed AM processes have different advantages and limitations. So the choice of the correct fabrication process is strictly related to the material that must be used, the geometrical resolution required, and the desired mechanical properties of the final product.
However, this work aims to design a lattice structure made of steel so only an overview of metal AM technologies is presented.

3.2 Powder Bed Fusion

Powder bed fusion (PBF) is one of the most used AM processes. In the beginning, this technology has been invented for the production of polymeric parts, with the Selective Laser Sintering machine (SLS), but later it has been developed the Metal Laser Sintering (MLS) machine able of treating metals.
All PBF technologies follow the same building procedure: the selective creation of micro-fusions between powder particles through a thermal source and the deposition of a new powder layer on a powder bed. A scheme of a generic PBF is represented in Figure 3.3 The production process consists of many steps:

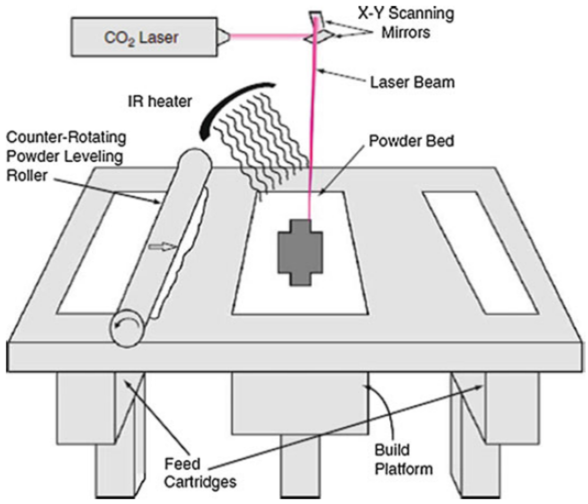


Figure 3.3: Scheme of PBF process [6]

- A layer of powder particles is brought from feed cartridges to the build platform by means of a counter-rotating powder leveling roller

- CO_2 Laser selectively heat up particles of powder to create micro-fusion between each other
- the building platform is lowered down by the thickness corresponding to that of the desired layer

this sequence is repeated many times as needed to produce the all three-dimensional component.
It is worth saying that the quality of the final product depends on the conditions at which the fabrication process is carried out. These can be regulated by setting adequately the process parameters. These latter are many and they are related mainly to the thermal source, powder, scanning system and temperature. They are not completely independent of each other so an optimal set up should be found.
After the building process, post-processing is necessary to clean up the part from the unbound powder that supports the structure during the fabrication.
Laser Powder Bed Fusion (L-PBF) can produce very complex geometries with a high-resolution level and it is a very flexible technology because it is compatible with a multitude of materials, for example, all weldable metals.
Another pro of PBF is the reduction of waste material since the support of the structure during the building phase is represented by the same powder that in turn can be re-used.

Among the most diffused PBF technologies, MLS and Electron Beam Melting (EBM) are those deserving particular attention.

3.2.1 Metal Laser Sintering

MLS follows perfectly the PBF process. The typical machine performing MLS is built the same way of that represented in Figure 3.3 where the thermal source emits a laser beam which is focused by a set of lenses and projected on powder bed through scanning mirrors.
The materials mainly processed by this process are Alluminium, Titanium, Cobalt, Chrome, and Nickel alloys and steel. Furthermore the high level of resolution of this technology provides to the final product good mechanical properties. In fact, it is widely used to create biomedical implants.

3.2.2 Electron Beam Melting

Electron Beam Melting (EBM) has been commercialized for the first time in 2001 by Arcam AB and today it is a widely used process. A representation of an EBM machine is reported in Figure 3.4 Unlike other PBF technologies, EBM uses an electron beam to melt the metal powder particles. This means that it can process

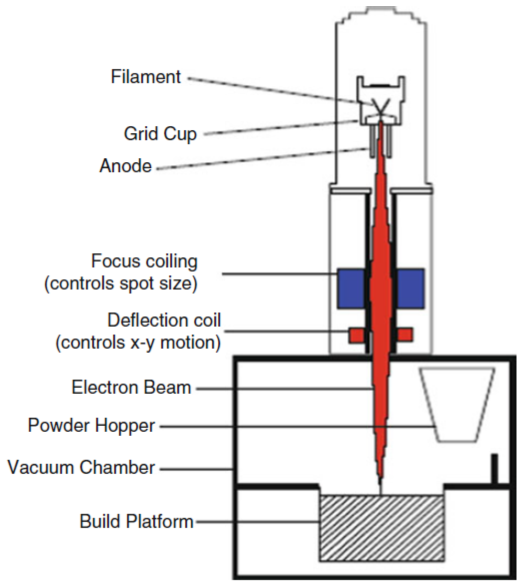


Figure 3.4: "Schematic of an EBM apparatus"[6]

only conductor materials, like metals.
On the other hand, except for the system providing the thermal source and the environment working condition, the EBM technology performs the same building steps of a PBF process: metal powder is distributed on a building table and it is selectively hit by an electron beam. Specifically, the electron beam is generated by heating a filament to 2500°C. When a voltage difference is applied to the filament, it reacts emanating electrons which in turn are accelerated by an anode. Then, a magnetic lens makes the electrons converging in a beam and it even controls its direction. The electron beam can develop up to $4kW$ of power. When the electrons point the powder layer, the metal particles start to heat up by the absorption of photons. So the kinetic energy of electrons is transformed in thermal energy for powder. To make this process possible it is necessary to conduct electric current. This is the reason why only metal powder can be treated, thus EBM is not suitable for the production of polymeric components.
The main peculiarity of EBM is that it builds the part in a vacuum chamber and this takes a lot of advantages. In particular, this protected environment allows the powder to not be subjected to oxidation so the exceeding powder removed in post-processing can be used again. Thus, material saving.
Since the Electron beam generation occurs at high-efficiency and the beam can be controlled very rapidly, this technology is also time-saving. However, due to many conductive problems that can happen in particles, the component produced by EBM is not so high-detailed as those produced by MLS. Also here, to refine

the quality of the product post-processing operations can be made, like machining operations or heat treatments.
Despite all these pros, an EBM machine is very expensive so this technology is restricted only to companies having a huge amount of money to invest in it.
Just to give a clear panoramica the main differences between MLS and EBM are spotted in Table 3.1

Table 3.1: "Differences between EBM and MLS" [6]

Characteristic	Electron beam melting	Metal laser sintering
Thermal source	Electron beam	Laser
Atmosphere	Vacuum	Inert gas
Scanning	Deflection coils	Galvanometers
Energy absorption	Conductivity-limited	Absorptivity-limited
Powder preheating	Use electron beam	Use infrared or resistive heaters
Scan speeds	Very fast, magnetically driven	Limited by galvanometer inertia
Energy costs	Moderate	High
Surface finish	Moderate to poor	Excellent to moderate
Feature resolution	Moderate	Excellent
Materials	Metals (conductors)	Polymers, metals and ceramics
Powder particle size	Medium	Fine

3.3 Directed Energy Deposition

Directed Energy Deposition (DED) is a manufacturing process that uses a focused heat source to melt powder or wire which is straight deposited, through a nozzle, in a specific point of the building basement. As stated by Gibson et al [6]: *"DED processes are not used to melt a material that is pre-laid in a powder bed but to melt materials as they are being deposited."*
Even if this technology can process different materials, it is mainly used to treat metals. In this last case this system is referred as "metal deposition". Usually this process is used to repair existing components or to produce in series different parts, because these operations can not be performed with PBF technologies. However it is also possible to produce three-dimensional complex geometries, but support material and a multi-axis deposition head are required.
Lastly, in most laser applications there is a moving system allowing relative motion between beam and part. Usually, it is preferred to move only one of the two.

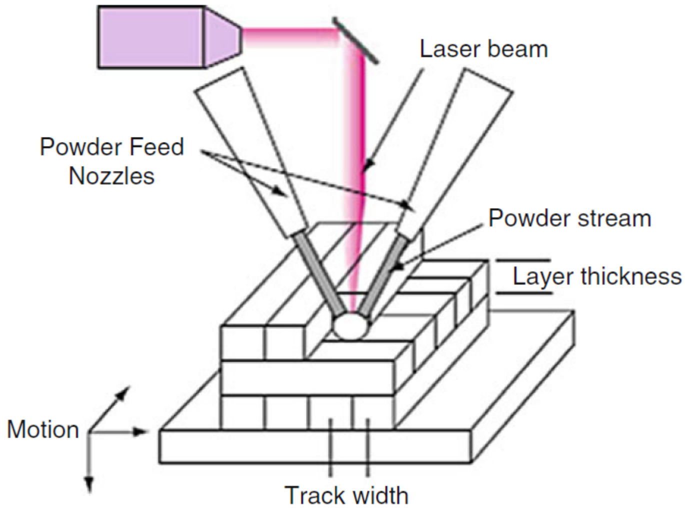


Figure 3.5: Representation of a laser powder DED system [6]

3.4 Influence of manufacturing process on lattice structures

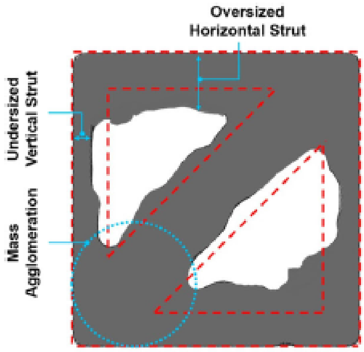
As seen so far, cellular solids can be developed along two or three dimensions. In both cases, the overall geometry may be very complex so cellular solids are very difficult to fabricate through conventional solutions. This is the main reason why the use of cellular solids is spreading across the engineering world for a not long time.
With the advent of Additive Manufacturing the fabrication of lattice structures is becoming increasingly easier. As a consequence, this technology is the most used approach to fabricate cellular solids, even because it does not need complicated process planning. However, sometimes lattice structures can have a too complex geometry and require an exaggerated level of accuracy, especially in case of micro-structures.
The high level of geometric complexity and some limitations of manufacturing processes make a fabricated lattice structure predisposed to present inaccuracies. It has been realized that these imperfections caused by fabrication operations alter the mechanical properties of the manufactured product. Thus the mechanical properties of a fabricated lattice structure can differ from those predicted in the design phase performed in a simulation environment. Indeed, these deviations can make the manufactured component not suitable for the desired application.
For this reason, it is worth doing an overview of how AM fabrication may alter geometrically the final product respect to the CAD model. Since this work aims to design a metallic lattice structure, the review focuses on PBF processes: MLS and

EBM.

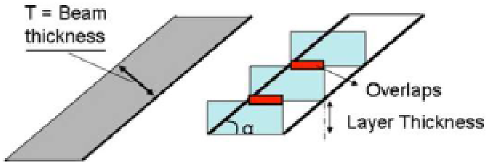
3.4.1 Lattice structures fabricated by PBF

Metallic lattices produced by PBF are prone to geometrical alterations of the structure mainly because of the thermal cycle the metal powder particles undergo during the fabrication process.
So the quality of the final product is dependent on the process parameters and a classification of them can be found in the paragraph dedicated to the introduction to PBF processes.
As a matter of fact, thermal source parameters are linked to the quality fusion of powder particles. Scan parameters determine how particles are fused. Powder parameters affect the absorption capability of particles. Temperature-related parameters influence the repeatability of the manufacturing process [7]. The main variations that a manufactured lattice shows are dimensional inaccuracies, porosity, surface irregularities.
Considering the high-detailed requirements of lattice micro-structures and the accuracy limitations of many PBF solutions, dimensional inaccuracy is one of the most diffused variations caused by fabrication processes. Dimensional variations appear differently depending on the considered lattice features. For example, struts undergo strong dimensional imperfections. One reason why this occurs is attributable to the sensitivity of metallic particles to rapid temperature changes. In particular, the high cool-down rate of the melted pool can cause shrinkages and consequent deviation of struts (Figure 3.6a). Another reason could be the inclination of the strut respect to the building direction. One of the first studies on this effect has been conducted by Cansizoglu et al. by using EBM technology [8]. As illustrated in Figure 3.6b, they attributed the phenomenon to the overlap of layers. If the thickness of deposited material is not so small compared to the designed strut dimension, rough struts can be obtained. The smaller is the overlap between layers the smaller is the resistant strut section with a consequent loss of mechanical properties. So to not compromise too much the strength of the strut, they suggest to never have a strut inclination angle lower than 20°.
A deeper study on characterization of struts fabricated by EBM has been carried out by M.Suard [10] who pointed out not only the dependency of the variation of the strut diameter on the inclination angle but even its section shape, Figure3.7. Another important geometrical issue, instead, can be spotted in the vicinity of the nodes. Since it is the interconnection point of the struts, here an extra deposition of material commonly occurs as in Figure 3.6a.

Porosity appears in the form of voids with a typical size of the order of µm. Usually, pores are generated as a consequence of gas trapping during the solidification phase

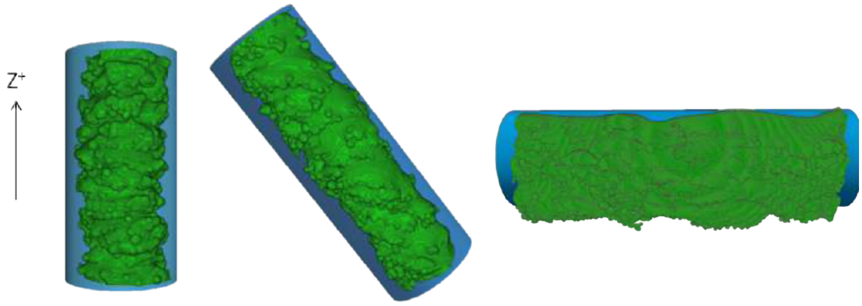


(a) Focus on localized dimensional inaccuracies [9]

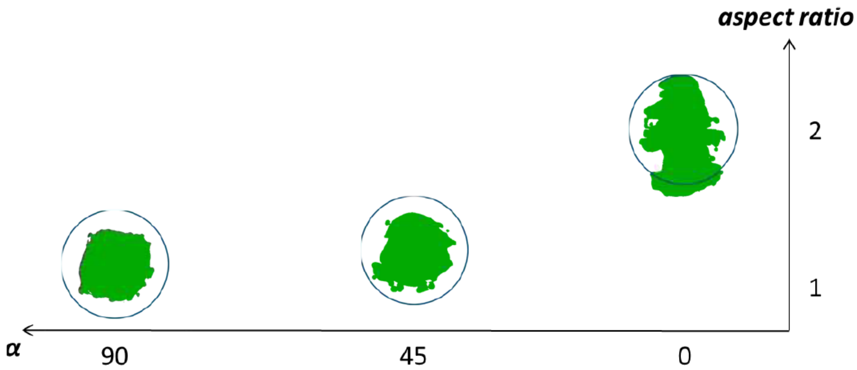


(b) Schematic of a inclined strut fabricated by EBM [8]

Figure 3.6: Representation of typical dimensional inaccuracies



(a) "Difference between designed (blue) and fabricated (green) strut"



(b) Strut shape variation against inclination angle

Figure 3.7: Dimensional struts inaccuracies for different inclination angles [10]

of the fused metallic material, of initial imperfections of the powder particles, and of scarce provided thermal energy. As illustrated in Figure 3.8, Amani et al. [11]

noticed that for a thin-strut lattice produced by Selective Laser Melting (SLM) , pores dimensions are larger in the nodes because of lack of fusions and smaller in the struts because of entrapped gas. Furthermore, these voids represent stress concentrations and the may become very dangerous under cyclic loads because they promote nucleation of cracks that can drastically reduce the fatigue life. The last

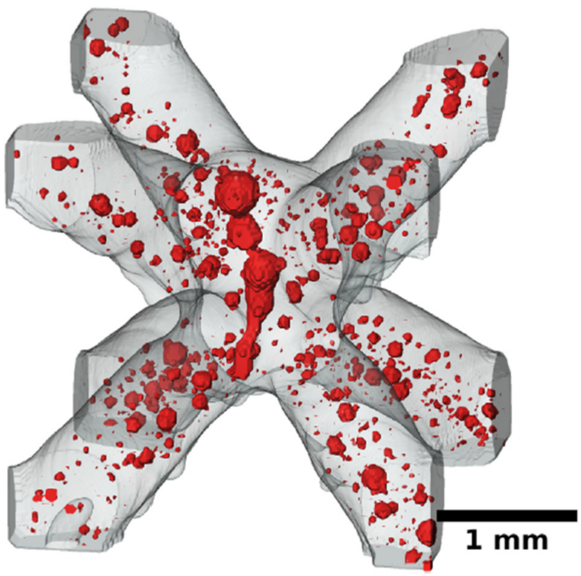


Figure 3.8: Porosity of thin-strut lattice produced by SLM [11]

geometrical inaccuracy appears as surface roughness. Even this problem is related to the melt pool instability and it has a negative effect on the mechanical properties of the structure. So far, it has been seen how different variations a designed lattice structure undergoes during its fabrication process. Actually, this brief overview is supposed to shed a spotlight on the causes of mechanical discrepancies that there are between fabricated lattice structures and designed ones.
Since the AM costs are not so small, it is not convenient to detect the actual mechanical properties of a fabricated lattice experimentally. For this reason, nowadays it is diffused to develop simulation models that can include most of the fabrication variations to correctly predict the properties of the final product.

Chapter 4

Structure modeling

4.1 Introduction

Nowadays, mechanical simulations are becoming increasingly important for engineering design processes. Thanks to the availability of powerful computers, simulation software allows the user to run a multiphysics analysis of a product saving a load of time and costs for industries. In this way it is possible to give shape to new products, to verify the feasibility of the prototype and where it is possible, to detect how it is improvable. In particular, it allows the engineering designer to understand the consequences of a possible modification to the initial model in terms of performance.
In conclusion, an accurate detail design phase permits to gather a lot of information about the product, from structural to manufacturing point of view. In particular, when a design team needs to face with lattice structures, where geometrical features strongly affect mechanical properties, simulation analysis is essential. For this reason, a simulacrum to process a Finite Element (FE) analysis is needed. In particular, when an engineer designs a lattice structure needs to deal with two issues. The first one is related to the complexity of lattice geometry because usually it is very detailed and running structural analyses on the CAD model could be too computationally expensive. The second one is about manufacturing variations. When lattice structures have very small-sized struts, accuracy limitations of manufacturing processes cause dimensional imperfections which substantially penalize the mechanical properties of the component. For this reason, an efficient simulacrum is needed.

4.2 Finite Element approach

The most simple way to approach a mechanical problem is that of adopting the continuum mechanics theory. It is well known that at the atomic level all bodies are made of an assemblage of particles in energetic equilibrium with each other. Considering that usually mechanical problems focus on studying macroscopic behavior of materials, the just explained microscopic scenario is simplified by continuum mechanics. In fact, as suggested by the name, the main assumption of this theory is to suppose the nature of a solid as a continuum. In this way, by applying physical relations like conservation of mass, momentum, or by doing an energy balance it is possible to describe the behavior of a solid or liquid body through partial differential equations (PDE). However, this approach is not suitable for complex problems because they can not be solved by means of analytical models, thus by means of calculators.
In order to make a PDE based problem analytically solvable, many discretization methods have been invented. The one that has found the most success is definitely the Finite Element Method (FEM).
The transition from a PDE based problem to an analytical one can be done in a few and simple mathematical steps. First of all, a functional integral in a finite domain is required and usually, the principle of virtual work is the most used. Later, the finite domain must be discretized by means of a shape function which in turn must be continuous and n times derivative depending on the element, able to represent rigid motion, constant strain, and to ensure compatibility between adjacent elements. The most used shape functions used in FEM are polynomial because with them it is very easy to perform mathematical operations. The last step is to substitute the shape function in the functional integral and a matrix problem is obtained.

4.2.1 Preliminary considerations

Several engineering applications require high-strength and lightweight structure, from aerospace components to medical implants. For this purposes lattice structures are particulartly appriciated. First of all because by doing minimum modifications of structure geometry it is possible to vary the overall mechanical properties. Since they are mainly produced by Additive Manufacturing, the increase of geometrical complexity is a costless improvement.
Generally speaking, an engineer designing a lattice structure is interested only on the equivalent mechanical properties, thus those of the whole structure. But the last ones are strongly affected by deformation mechanism of the unit cell. So micro-deformation mechanisms are strictly related to deformation of the macro-structure. In fact, if a lattice structure is homogeneously built, mechanical properties of the

whole structure are predictable by doing test on a single unit cell. However care must be taken because this assertion is only valid when lattice structure is large enough to neglect all the alterations due to boundary conditions.
Now that the relation between micro and macro is highlighted, it is easy to understand how accurate the model should be to detect the right outcomes.
The most accurate model would be a model made by solid elements because it is the closest to an ideal 3D model. However, this solution i very computationally expensive and time represents a high cost for a company. In order to optimize the available time it is possible to create slightly less accurate models but with a much shorter running time. In order to achieve this objective, a model made of beam elements can represent a good compromise.
In this chapter all the design steps are reported even to better understand all the micro deformation mechanisms must be taken into account. Several engineering applications require high-strength and lightweight structures, from aerospace components to medical implants. For these purposes, lattice structures are particularly appreciated. First of all because by doing minimum modifications of structure geometry it is possible to vary the overall mechanical properties. Since they are mainly produced by Additive Manufacturing, the increase of geometrical complexity is a costless improvement.
An engineer designing a lattice structure is interested only on the equivalent mechanical properties, thus those of the whole structure. But the last ones are strongly affected by the deformation mechanism of the unit cell. So micro-deformation mechanisms are strictly related to the deformation of the macro-structure.
In fact, if a lattice structure is homogeneously built, mechanical properties of the whole structure are predictable by doing tests on a single unit cell. However, care must be taken because this assertion is only valid when the lattice structure is large enough to neglect all the alterations due to boundary conditions.
Now that the relation between micro and macro is highlighted, it is easy to understand how accurate the model should be to detect the right outcomes.
The most detailed model would be a model made by solid elements because it is the closest to an ideal 3D model. However, this solution is very computationally expensive and time represents a high cost for companies. To optimize the available time it is possible to create slightly less accurate models but with much shorter running time. To achieve this objective, a model made of beam elements can represent a good compromise for the representation of lattice structures.
In this chapter all the design steps are reported even to better understand all the micro deformation mechanisms that must be taken into account.

4.2.2 Choice of a simulation software

Before starting to think about the model, simulation software should be chosen. This is a very important step because it represents the environment allowing the creation of the model and the execution of mechanical tests on it. So, depending on the desired design approach a particular software can be more or less suitable. Since the purpose of this work requires a high level of editability of the structure, Ansys APDL results to be the most appropriate. As stated on ANSYS website [12] *"Regardless of the type of simulation, each model is represented by a powerful scripting language ... the Ansys Parametric Design Language (APDL). APDL is the foundation for all sophisticated features, many of which are not exposed in the Workbench Mechanical user interface. It also offers many conveniences such as parameterization, macros, branching and looping, and complex math operations"*.

4.3 Choice of elements

As already stated, the most precise way to represent a lattice structure would be a model built with solid elements but a less computationally expensive model is needed. So to obtain the optimal model, a model of solid elements can be pointed out as a benchmark.
The idea is to create a model with beams but before identifying the best solution to be adopted it is better to do a brief overview of the elements available in the ANSYS library and the theory that is behind them.

4.3.1 Euler-Bernoulli beam

The Bernoulli-Euler beam is the simplest model to describe the mechanics of a beam. This approach introduces an approximation because it works under the assumption that deformation caused by shear is negligible compared to that caused by bending. As a consequence, even under load conditions, the beam cross-section remains always perpendicular to its neutral axis. However, when a beam has the main dimension, like the length, much larger than the other, like the thickness, the results of this model are quite reliable.
In ANSYS, the Bernoulli-Euler beam is represented by *beam3* element for two-dimensions analyses and by *beam4* element for three-dimensional environments.

Beam3

beam3 element is only suitable for two-dimensional analyses. Since this element can only undergo stretching and bending deformations, it is fully descripted by three degrees for freedom per each node. Making reference to Figure 4.1 it can produce two translation displacement, along x and y and a rotational one abou z axis. Furthermore, its geometry is set by the "real constants" command and the

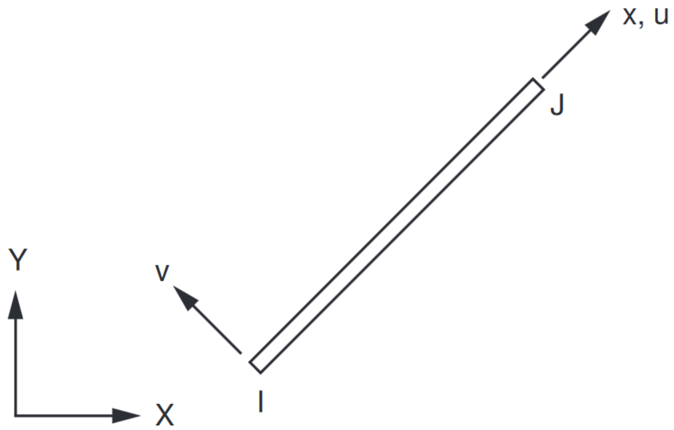


Figure 4.1: "beam3 element geometry [13]"

features that can be entered are the cross-section area and area moment of inertia, the height. Then, also the material of which the beam is made must be defined by the "material properties" option. Since it is a two-dimensional element and the shear deformation is neglected, for this purpose, only the elastic modulus E_x is mandatory to be set. Other key-options are available in the *ANSYS* library but this work aims to a simplified structural analysis they can be neglected.
Lastly, the solution that the user obtains from this element can be expressed as "nodal displacements solution".

Beam4

If the simulation model must be developed in a three-dimension environment a beam with a higher order of degrees of freedom is required. A 3-D elastic beam adhering to the Euler-Bernoulli theory is identified in the *Ansys* library as *beam4* element, Figure 4.2. This element is defined by at least 2 nodes, a third node can be optionally entered. Since this element has stretching, torsion, and bending capabilities, it is provided with six degrees of freedom per node: translations *UX, UY, UZ* and rotations *ROTX, ROTY, ROTZ* along and about *x, y, z* respectively. For this element, the necessary geometrical information that should be entered in the software to represent accurately all the possible deformations are more in number than those of *beam3*. So, for the sake of simplicity, they all are listed in Table 4.1. Material properties, instead, are set through Elastic and Shear moduli and the density. Other key options are available but they are not relevant for the work of this paper. Even here, the output can be easily read by nodal displacements.

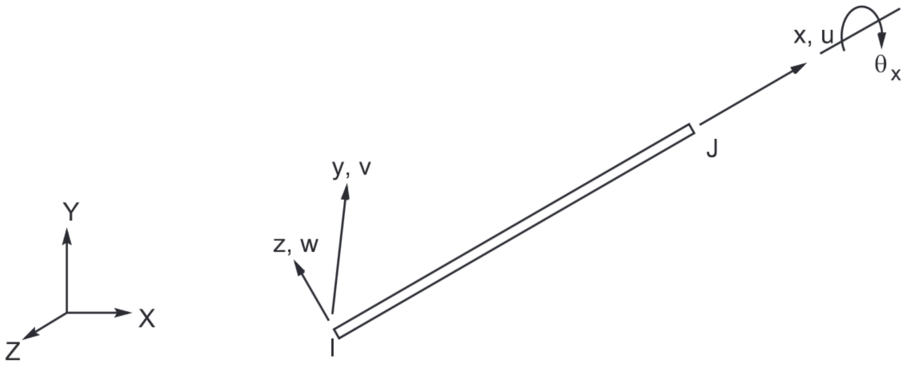


Figure 4.2: "beam4 element geometry [13]"

Table 4.1: *beam4* Real constants [14]

No.	Name	Description
1	AREA	Cross-sectional area
2	IZZ	Area moment of inertia
3	IYY	Area moment of inertia
4	TKZ	Thickness along Z axis
5	TKY	Thickness along Y axis
6	THETA	Orientation about X axis
7	ISTRN	Initial strain
8	IXX	Torsional moment of inertia
9	SHEARZ	Shear deflection constant Z
10	SHEARY	Shear deflection constant Y
11	SPIN	Rotational frequency (required if KEYOPT(7) = 1)
12	ADDMAS	Added mass/unit length

4.3.2 Timoshenko beam

Unlike the Euler-Bernoulli theory, Timoshenko has included the deformation contribution due to shear stress. So, this model overcomes many limitations found in the Euler-Bernoulli beam. For this reason, the Timoshenko beam is the most accurate model for the study of beam mechanics. However depending on the application, Euler-Bernoulli still provides a good representation of actual beams behavior, thus the choice of the element is not so obvious.
In the *ANSYS* library Timoshenko beam is identified by the *beam188* element.

Beam188

beam188 is a linear, quadratic, or cubic 3-D element and as *beam4*, it is defined by two nodes and provided with the same six degrees of freedom, Figure 4.3. This

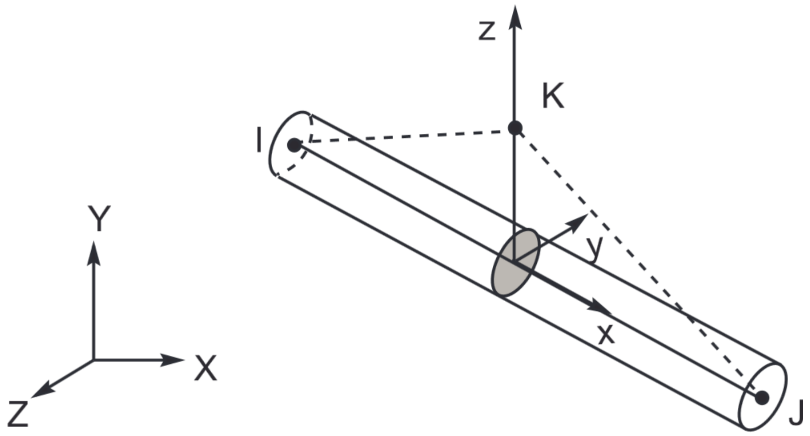


Figure 4.3: *beam188* Element geometry [13]

element becomes much accurate than *beam3* or *beam4* when the beam model to be analyzed is characterized by low slenderness. *beam188* is a 3D element and as *beam4*, it is defined by two nodes and provided with the same six degrees of freedom.
Furthermore, since this element has the limitation of considering a first-order shear deformation it is not suitable for a too stout beam. To make an idea of the difference in results between Timoshenko and Euler-Bernoulli beams, some values of deflections as functions of the slenderness are listed in Table 4.2. Where

Table 4.2: Ratio between Timoshenko and Euler-Bernoulli deflection against slenderness [14]

Slenderness Ratio ($GAL^2/(EI)$)	δ Timoshenko /δ Euler-Bernoulli
25	1.120
50	1.060
100	1.030
1000	1.003

slenderness ratio is expressed as a function of G shear modulus, A cross section area, L length of the member and EI flexural rigidity.
From the just observed beam deflections values, it can be noticed how for low values of slenderness there is a larger discrepancy of results between Timoshenko and

Euler beam while for high values of slenderness these two models are equivalent. This highlights that shear contribution becomes negligible with the increasing of the beam slenderness.

4.3.3 Solid element

As already described in the incipit of this chapter, a model made of solid elements is the most accurate. Ansys has several types of elements but for the representation of a beam, a *solid45* element is a good option, Figure 4.4.
Usually, *solid45* is used for the 3-D modeling of solid structures. The element is defined by eight nodes and provided with three degrees of freedom at each node: only translations along x, y, and z.

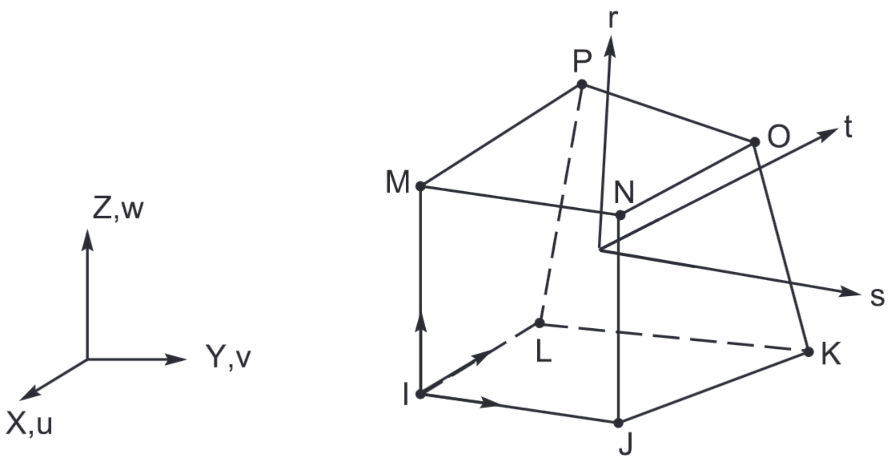


Figure 4.4: Schematic of *solid45* element geometry [13]

How can be understood from the element description, *solid45* represents the so called *brick element*. Even if it allows a very detailed representation of three-dimensional parts, the numerical model can become too computationally expensive because of the high number of nodes must be defined.

4.3.4 Comparison between element types

To understand which is the optimal element to model a lattice structure, the modeling of the simplest lattice unit is sufficient to be carried out.
As already explained lattice structures can be seen as a series of interconnected struts where interconnections can be modeled as rigid joints. Considering the constraints and loading conditions imposed by the continuity of lattice structure [1], for this analysis it is possible to isolate a half strut rigidly fixed to one end (left) and guided to the other (right), Figure 4.5.

Figure 4.5: Fixed-guided beam model

For this purpose, the strut made of *solid45*, *beam3* and *beam188* elements must have the same features of the solid beam. The main parameters defining a beam are:

- L is the length of the strut
- b is the width and t is the height or thickness of the strut, which multiplied by each other define A, the cross-section area
- I is the Area moment of inertia of the cross-section
- E is the Young's modulus

For the sake of clarity let express the dependent variables in formulae:

$$A = b \cdot t \tag{4.1}$$

and for this investigation the cross-section shape has been considered arbitrarily rectangular, thus the area moment of inertia turns out to be:

$$I = \frac{b \cdot t^3}{12} \tag{4.2}$$

As already shown in the first chapter, a lattice structure can be bending or stretching dominated and for this reason, a bending and a stretching test on a strut are required. Processing these two types of test it is possible to observe the strut deformation mechanism and to make a comparison between the above model built with different element types.

Stretching test

A stretching test is easy performed by applying an axial, thus horizontal, distribution of force on the right end of the strut.
Obviously, as can be seen in Figure 4.6, the stretching test produces an axial deformation and the produced displacement Δx is determined by the physical

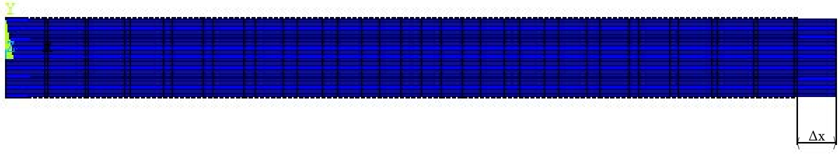


Figure 4.6: Stretching test of a strut

and geometrical properties of the strut. In particular it can be quantified by the following expression:

$$\Delta x = \frac{F}{k_x} \tag{4.3}$$

Where:

$$k_x = \frac{EA}{L} \tag{4.4}$$

k_x is the axial stiffness. Actually, this paragraph has been written just for illustrative purpose beacuse, since the error introduced by the different elements is related only to shear deformation, the results obtained by the strut built with the three element types are equivalent. So this investigation assumes meaning in the bending anlysis.

Bending test

To perform a bending test a vertical load is applied on the right end of the strut. As already said, the solid beam is the benchmark of this study. The parameters

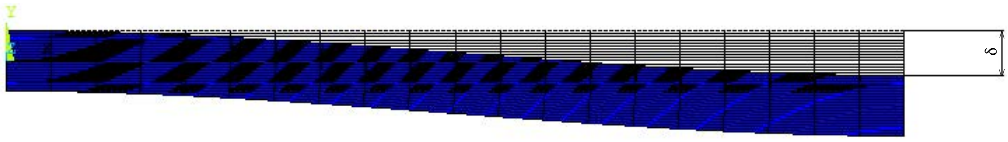


Figure 4.7: Strut under bending load

used to make a comparison between the element types during a bending test are the flexural rigidity k_f and the slenderness α which are defined as follows:

$$k_f = \frac{F}{\delta} \tag{4.5}$$

$$\alpha = L\sqrt{\frac{A}{I}} \propto \frac{L}{b} \qquad (for\ squared\ cross\ section) \tag{4.6}$$

where F is the vertical force applied on the right-end of the beam and δ is the vertical displacement produced by the beam under the described loading conditions.

In order to detect the accuracy of the element types at different levels of slenderness, this test is repeated many times by keeping constant the cross-section area and increasing the beam length.
As shown in Figure 4.8, the model built with *beam3* elements has a higher flexural rigidity at low values of slenderness. On the contrary, the strut made of *beam188* elements and that of *solid45* seem to behave the same way, independently from the slenderness level.
The different trend of *beam3* is mainly because the Euler beam does not consider any shear deformation which becomes a relevant mechanism when the strut is stout. Thus, the lower is the level of slenderness and the higher is the stiffening effect

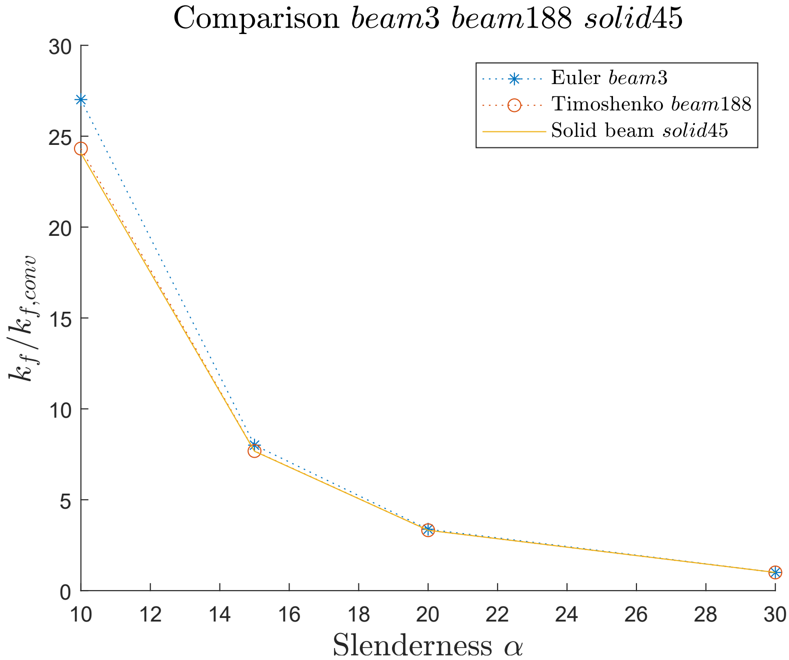


Figure 4.8: Comparison of flexural behavior between different elements

caused by the *beam3* element. So the gap existing between the flexural rigidity of the strut made of *beam3* and those of *beam188* and *solid45* can be seen as an error. So having a look at Figure 4.9, it can be seen that for α ranging from 10 to 15 the percentage of error takes values up to 11.
From all that has been said so far, *beam188* seems to be the most suitable element. The problem is that, to get the correct results a strut, made of *beam188* elements, needs to reach convergence in the solution. A mesh convergence study is therefore indispensable to evaluate the feasibility of using *beam188* element type. Taken a strut with defined geometrical features, in this case, α=*20*, Figure 4.10a shows how the strut is studied by discretizing it in many parts, where variable *ndiv*

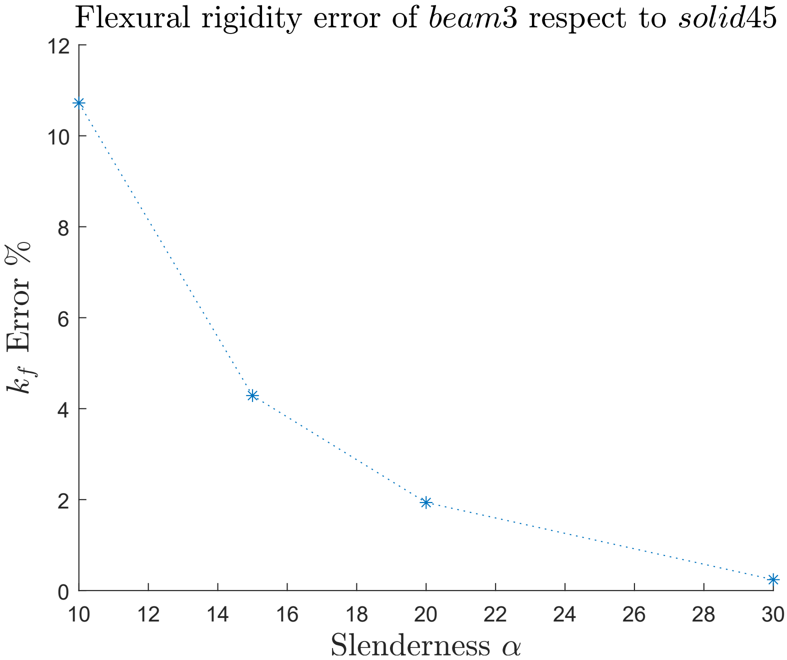


Figure 4.9: Error of *beam*3 at varying the slenderness

(a) Strut with $ndiv = 1$

(b) Strut with $ndiv = 20$

Figure 4.10: Discretization of the strut

indicates the number of divisions of the strut. The analysis has been carried out discretizing until *ndiv=20*, Figure 4.10b, because at this mesh level the solution of beam response already converges so it is meaningless to investigate further, Figure 4.11. As can be seen in Figure 4.12, for a number of divisions lower than three the strut made of *beam188* produces an important error: minimum error 10%. Considering such limitation, it is important to investigate whether it is convenient to use *beam188* rather than *beam3*. In order to do so, the number of *ndiv* above which *beam188* gives an error lower than that of *beam3* must be identified.
The first consideration is that a strut discretized by using *beam3* elements keeps the error constant regardless of the number of elements used. So, for the strut characterized by α=*20* the percentage error is about 2%, Figure 4.9.
Now, analyzing the percentage error trend produced by *beam188* elements illustrated in Figure 4.12, it is easy to conclude that the strut must be divided at least in six elements to get an error lower than that of *beam3*.

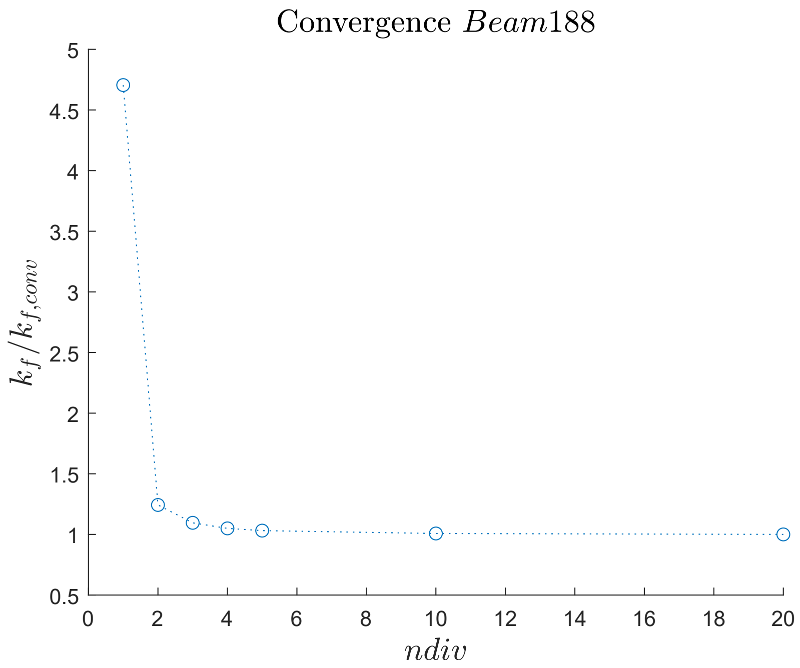


Figure 4.11: Convergence for $beam188$ element

It is worth highlighting that the choice of considering α=*20* is completely arbitrary. It is true that decreasing the slenderness the error produced by *beam3* increases while the error produced by *beam188* depends much more on mesh level than the strut slenderness. Usually, for lattice structures struts α is about 20, thus having an advantage in the accuracy of *beam188* over *beam3*, the already indicated discretization is required.
In conclusion, the choice of using *beam188* can be quite reliable in terms of accuracy of results but it is computationally more expensive compared to a model made of *beam3* elements since this latter does not need convergence.
Considering that during manufacturing processes lattice structures are subject to dimensional inaccuracies due to limitations of machines, the use of *beam3* element is advantageous within a certain margin of error.
Obviously, this choice strictly depends on design constraints and since for the prediction of lattice structures mechanical properties the computational error coming from the "element type" is negligible compared to all the issues related to the fabrication process, the use of *beam3* elements is preferred.

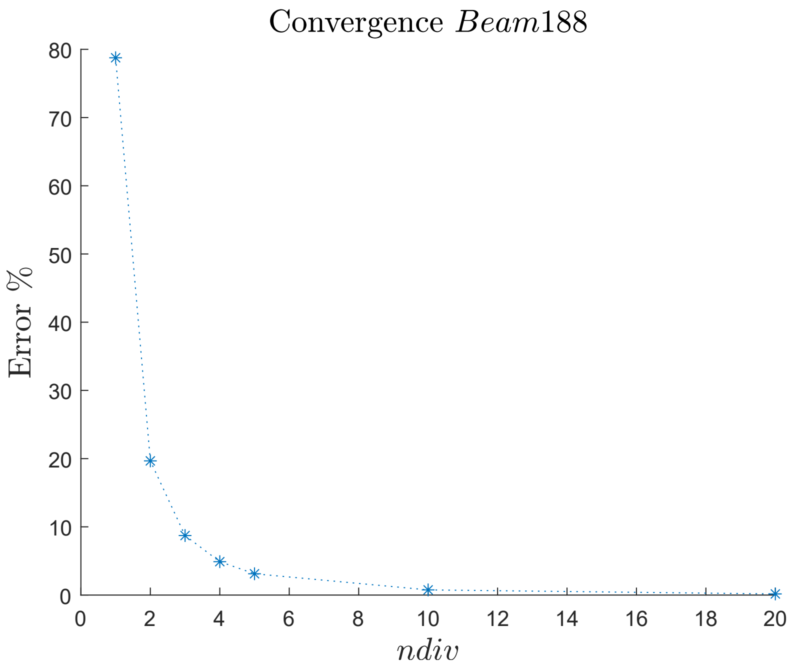


Figure 4.12: Convergence solution for $beam188$ element

4.4 Modelling approach

As seen in literature, the prediction of mechanical properties of lattice structures is very important during a design process because it represents important time and costs saving for an entire project. Thus, accurately modeling a structure becomes crucial for this phase. However, it has been seen also how different fabricated lattice structures are compared to designed ones because of the many building limitations of additive manufacturing processes. The layer by layer deposition generates a lot of dimensional inaccuracies especially for the thin inclined struts used to build lattice structures.

It is worth remembering the nature of many variations the structures undergo during fabrication processes in the way to understand which effects they can produce on the structure from a mechanical point of view. Lattice structure imperfections are mainly dimensional inaccuracies, porosity, and over-deposition of material in the vicinity of nodes.

Dimensional inaccuracies are strongly influenced by the machine level of accuracy and by many process parameters. In particular, AM technologies build the structure by following a specific orientation of the component. The oriented deposition of material becomes a significant factor for the fabrication of an accurate lattice structure, even for the anisotropy generated by the layering. Furthermore, lattice

structures struts are very small-sized and when they are oriented along with particular directions respect to the building one, they are very sensitive to inaccuracies. What really matters is that this effect strongly reduces the core strut dimension and consequently also its strength. Porosity is a marginal effect on fabricated structures but it is more or less important depending on the used technology. They appear in the form of voids and represent stress concentration areas. Then, in the case of a relatively high percentage of porosity, it may also alter the elastic mechanical properties of members. Lastly, another issue related to layer-by-layer building procedure is the excess of material deposited close to struts junctions. Even if the extra-deposition of material at the interconnections of members can imply a localized stiffening of the joint, the overall behavior of fabricated structures turns to be softer than the ideal designed one.
This thesis aims to create a structure model able to provide a certain tunability to the structure in order to reproduce all the possible geometric variations due to fabrication processes.
For this purpose, the *Gibson* approach illustrated in Figure 4.5 and widely used to model lattice structures has been revisited by adding compliant supports allowing to control the stiffness of the structure. Specifically, the just mentioned model has been improved by substituting the clamped end with an end supported by a set of springs.
The conceptual phase has been developed by considering a two-dimensional problem, thus the left end is equipped with two concentrated longitudinal springs, one along *x-axis* and the other one along *y-axis*, and one torsional spring allowing to control the flexural behavior of the beam about *z-axis*, Figure 4.13. The model has been approved by using a simple FE approach.

Figure 4.13: Springs-supported and guided beam model

Considering that the strut is treated as a single Euler-Bernoulli beam element, here it is possible to directly describe the mechanics of the member just introducing a polynomial expression. This mathematical relation is able to correctly describe the generalized displacements and setting the right boundary conditions it is possible to obtain the correct nodal displacement. Before describing the shape function it is worth highlighting the relationship linking the beam vertical displacement $u_y(x)$

to rotations $\phi_z(x)$, bending moment $M_z(x)$ and shear force $T_y(x)$.

$$\phi_z(x) = \frac{du_y}{dx} = u'_y(x) \tag{4.7}$$

$$M_z(x) = EI\frac{d^2u_y}{dx^2} = EIu''_y(x) \tag{4.8}$$

$$T_y(x) = EI\frac{d^3u_y}{dx^3} = EIu'''_y(x) \tag{4.9}$$

After defining many relations about the mechanics of beam, a shape function able to represent the deformation of the beam and to respect all the boundary conditions must be defined. From the just listed equations it can be noticed that in order to describe the shear stress, a polynomial of third order is required. Since rotations moments and shear forces are a function of displacement, the easiest approach is to use a Hermite polynomial. In this way it is also easy to compute its derivatives as shown below:

$$u_y(x) = ax^3 + bx^2 + cx + d \tag{4.10}$$

$$u'_y(x) = 3ax^2 + 2bx + c \tag{4.11}$$

$$u''_y(x) = 6ax + 2b \tag{4.12}$$

$$u'''_y(x) = 6a \tag{4.13}$$

where a, b, c, d are unknown constant multipliers that must be determined by imposing the boundary conditions characterizing the problem.
For the model under analysis it has been seen that the left end is supported by two longitudinal and a torsional spring. Thus, on this side, the shear force is balanced by the vertical longitudinal spring and the flexural moment by the torsional spring.

$$EIu'''_y(0) = -K_y u_y(0) \tag{4.14}$$

$$EIu''_y(0) = K_t u'_y(0) \tag{4.15}$$

On the right end, the force applied on the beam is balanced by the shear while the rotation can be considered null because of the vertical guide.

$$EIu'''_y(L) = F \tag{4.16}$$

$$u'_y(L) = 0 \tag{4.17}$$

By substituting the boundary conditions in the polynomial formulations and assuming

$$\beta = \frac{K_t L}{EI} \tag{4.18}$$

the unknown constants assume the following values:

$$a = \frac{F}{6EI} \tag{4.19}$$

$$b = -\frac{3aL}{2(1+\beta L)} \tag{4.20}$$

$$c = 2\beta L b \tag{4.21}$$

$$d = -\frac{F}{K_y} \tag{4.22}$$

This analytical solution has been implemented in *Matlab* and verified in *Ansys APDL*. So this model is validated and when it is characterized by infinitely stiff springs it produces a deformed shape as follows in Figure 4.14: It is interesting

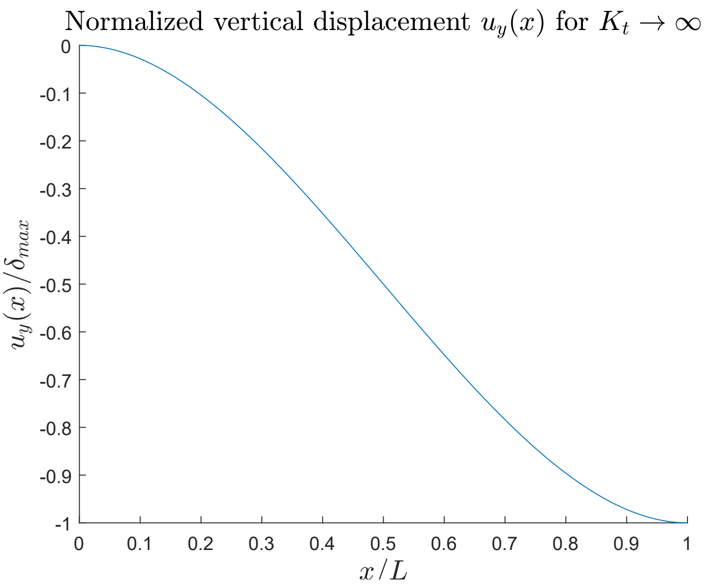


Figure 4.14

observing that the deformation values of the beam under load is dependent on the stiffnesses K_t and K_y of the springs. Obviously, in this case K_x does not appear because the load is vertically oriented and the axial deformation is neglected. This

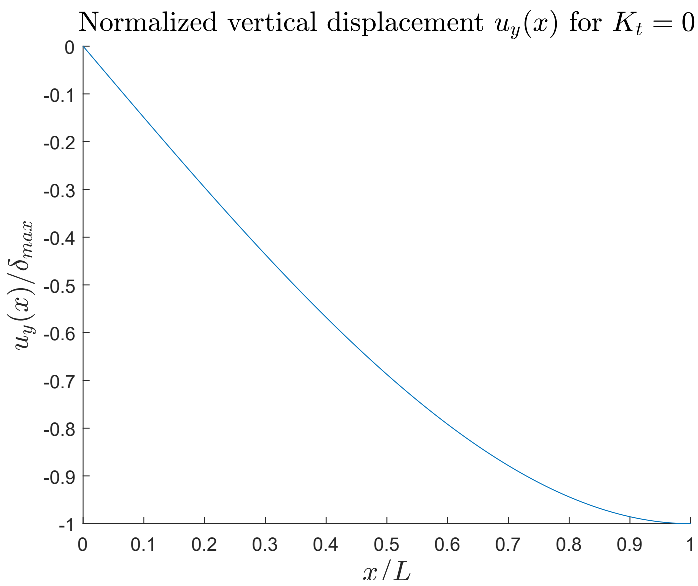


Figure 4.15

means that by tuning the stiffness of the springs the beam becomes more or less softer. What is worth observing is that when K_t assumes an infinite value the model is equivalent to a clamped beam and the derivative of the displacement, thus the rotation $u'_y(0) = \phi_z(0) = 0$ while when K_t is tuned to lower values $u'_y(0) = \phi_z(0) > 0$. In this way, it is possible to give a softening effect to the beam. More specifically, as illustrated in Figure 4.16, by keeping constant all the beam features and decreasing K_t, thus β, the rotation about the left-side node and the maximum vertical displacement δ_{max} of the right-end both increase. Anyway, even if the torsional spring takes a null value the vertical displacement is limited by the right guided end.

4.4.1 Spring model

How it is well explained in 4.1, simulation model equipped with torsional springs is aimed to allow the designer a certain control of the deformation mechanism in the vicinity of the nodes.
In the beginning, this model has been created to understand if it was possible to represent many complications due to anisotropy of the deposited layer or to the creation of porosity by tuning the stiffness of the available set of springs. However, since the porosity inside the node is randomly distributed, it has not been possible to establish a correlation between the level of present voids and the stiffness of the springs.

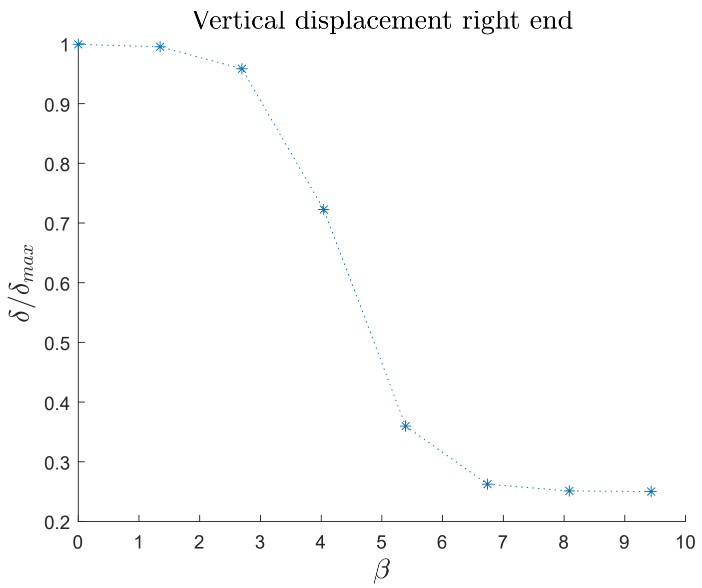


Figure 4.16: Maximum deflections against β

Furthermore, it has been noticed that the torsional spring, obviously allows the rotation of the strut about the node. If this effect is studied in a solid simulation model, the torsional spring can not control the deformation of the solid-represented node. This is mainly caused by the fact that the solid strut turns out to be fixed at the joint anyway. So, to reproduce a starting rotation of the strut in the vicinity of the node, the addition of symmetrical notches in the solid model came out as the best choice, as illustrated in Figure 4.17. With this configuration is clear that the strut can rotate in the vicinity of the node and a study has been carried out to figure out the correlation between the size of the notches and the stiffness of the torsional spring.
First of all, the notch shape is rectangular and the study has been performed by increasing only its main dimension: the depth. For a strut with $1mm$ of thickness, the notches depth has been increased from $0.1mm$ to $0.3mm$. The results are expressed in Table 4.3.

Table 4.3: Comparison between a solid model with notches and a beam model with springs

Solid Model		E $[MPa]$	Beam model		E $[MPa]$
	0.1	43.11		5000	43.1
Notch depth $[mm]$	0.2	34.36	K_t $[Nmm/rad]$	2470	34.32
	0.3	20.27		900	20.2

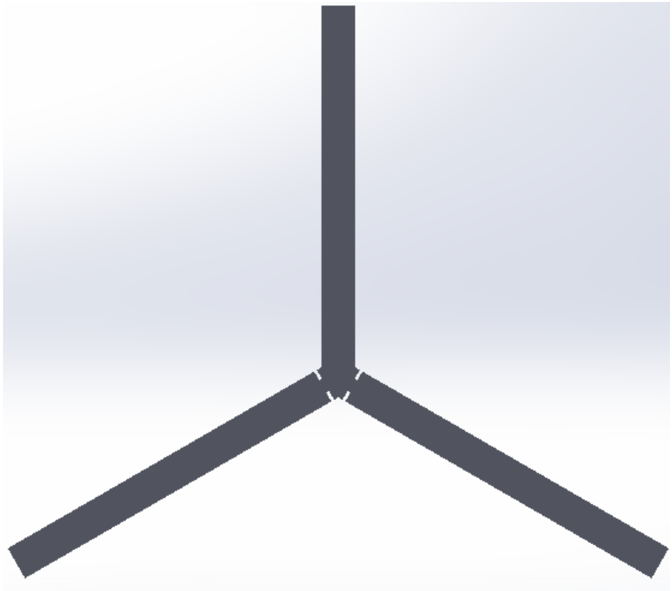


Figure 4.17: Representation of symmetrical notches

However, including micro notches in the structure, would mean introducing stress concentrations. Since there are many solutions to softening the structure, such as reducing the strut thickness, this design choice could be meaningless.

4.4.2 Modeling oversized nodes

Lattice structures could show an extra-deposition of material in the vicinity of struts junctions causing a localized stiffening of nodes. So, this paragraph aims to better understanding this microscopic effect and evaluating if it is exploitable in a design process.
So far the struts have been considered as beams with uniform cross-section but if the extra-material deposited at the node must be taken into account the model previously observed should be modified. In Figure 4.18 an interconnection of three struts of a honeycomb unit cell is represented.
From the modeling point of view, the lattice structure strut can be represented as a series of beams with different cross-section areas. Those beams located in the vicinity of interconnections (purple) with other struts are thought to be thicker to reproduce the extra-deposition of material while the rest of strut is characterized by the geometrical features of the designed strut (light-blue), Figure 4.19. To detect the differences between this configuration and that of a single beam at uniform cross-section a tensile and bending tests are performed and the results in terms of stretching and bending deformations are shown in Figure 4.20 and Figure 4.21

Figure 4.18: Representation of an oversized as-fabricated node

Figure 4.19: Modeling of a strut with oversized node

respectively.
Furthermore, it is possible to notice that since the cross-section is constant the axial deformation turns to have a linear trend along the largest dimension of the beam (*blue line*). This is the reason why the strut built with two different beams has a lower axial displacement slope in the first part (*red line*). Even the bending behavior is strongly affected by the beam close to the node position. In fact, the thickening of the node produces a stiffening in its surrounding area so the overall strut bends less causing a lower maximum vertical displacement.

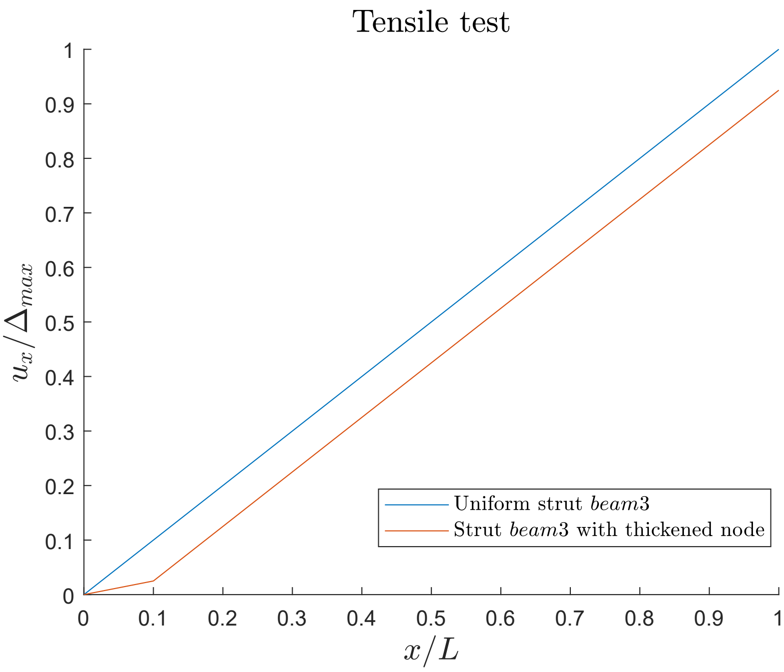


Figure 4.20: Comparison between uniform strut and one with thickened node

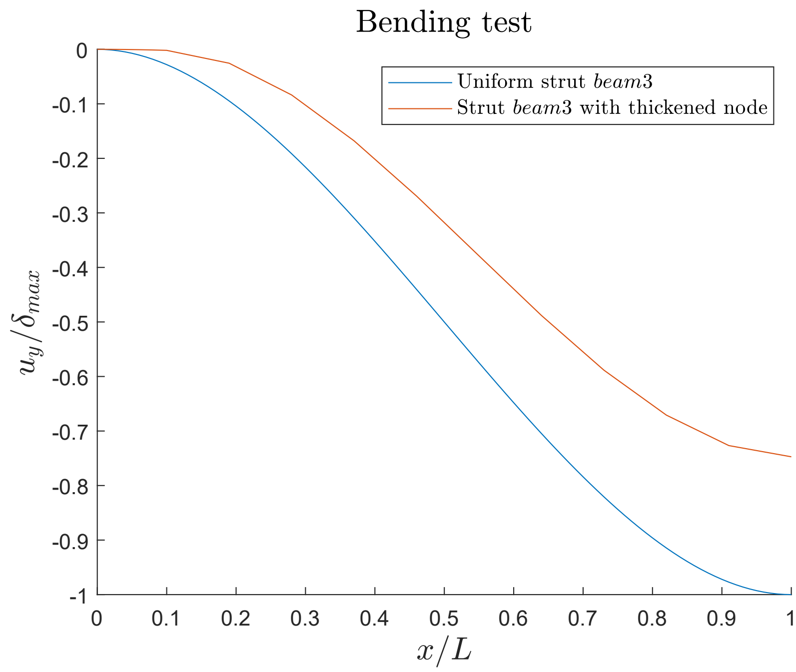


Figure 4.21: Comparison between uniform strut and one with thickened node

4.4.3 Modeling of dimensional inaccuracies of struts

The dominant variation that a designed lattice structure undergoes during the fabrication process is dimensional inaccuracy. It is clear that the magnitude of this imperfection depends on the accuracy level of the machine used to produce the lattice and also to the process parameters. So, it is difficult to create a model that works for each manufacturing process and for all combinations of parameter conditions.
This work aims just to give a method to include this kind of fabrication imperfection to a beam simulation model. From literature it has been already studied that for a predefined manufacturing process with its set of parameter processes, the magnitude of dimensional inaccuracy can be related to geometrical features of the struts: inclination angle θ and thickness b. In particular, it has been noticed by Suard [10] that for an EBM manufacturing process the geometrical inaccuracy of the struts was more important with decreasing the strut thickness and also with the inclination: the more is parallel to building direction ($\theta = 0$) (vertical struts) and the higher is the inaccuracy. Following this reasoning, a fictitious trend of dimensional inaccuracy is sketched in Figure 4.22. Since the geometry of the simulation model is built through a *Matlab* scripted that in turn generates a parametric code for *Ansys APDL*, it is possible to variate the dimensions of the

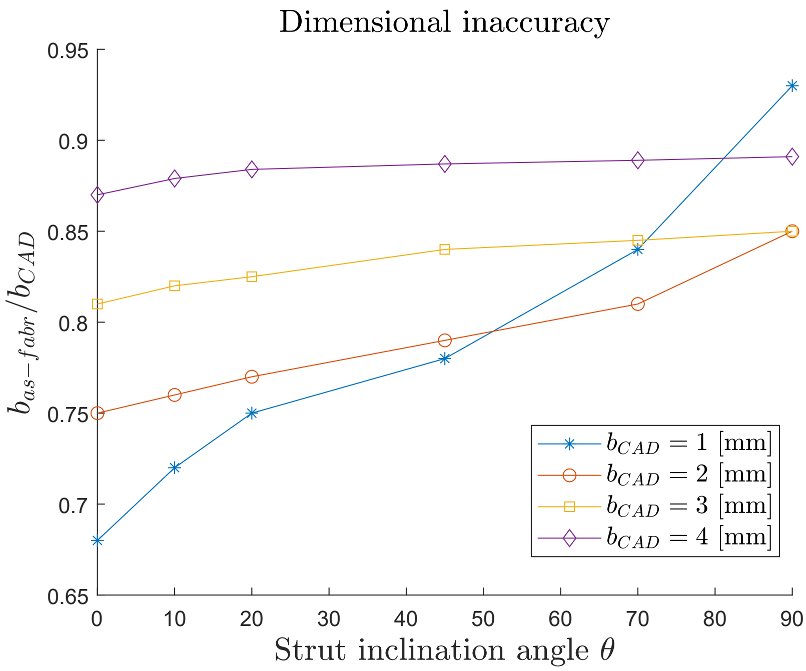


Figure 4.22: Dimensional inaccuracy trend as a function of b and θ

struts on the basis of the input thickness and inclination angle.

Chapter 5

Mechanical properties

5.1 Introduction

Usually, mechanical testing is performed to define the intrinsic materials' mechanical properties regardless of the geometry of the component. In the case of lattice structures, the study of bulk material properties is not enough. As already seen, cellular solids are becoming increasingly used in the engineering field because of their high strength to weight ratio and this appreciated mechanical feature is the result of a well-designed combination of geometric disposition of cells and the properties of the material chosen to build them.
From this reasoning it is possible to conclude that in lattice structures the microscopic deformation mechanism, for instance, that of a single cell strut, generates a completely different macroscopic effect on the whole structure. For example, a strut of generic metal lattice structure has a positive Poisson's ratio (it takes the value of the material which it is made of, like $\nu = 0.3$ in case of steel) but if the unit cell is characterized by an auxetic configuration the overall structure will have certainly $\nu < 0$. So the nature of the material adopted to build the structure is not the only information needed to evaluate the characteristics of the cellular solid. Even if in the ideal case the used material is considered isotropic the lattice structure can assume anisotropic behavior. For this reason, it is necessary to find out equivalent mechanical properties describing the behavior of the whole lattice.
Generally speaking, to evaluate Elastic and Shear Moduli of any component it is needed to apply a load on it and to quantify its response in terms of deformations or vice versa. So there is a strict connection between loads and deformations and in the specific case of elastic deformation field there is a constitutive equation relating them, the so-called Hook's law:

$$\sigma_{ij} = E_{ijhk}\varepsilon_{hk} \tag{5.1}$$

where σ_{ij} represents the stress vector, E_{ijhk} the tensor of elasticity and ε_{hk} the deformation vector. For cellular solids it is possible to assume three mutually perpendicular planes of symmetry, hence obtaining an orthotropic material. So in conclusion, the relationship between stress and deformation written respect to reference frame coincident with axes of symmetry can be expressed as follows: This section aims to provide a method of analysis intended to define equivalent

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{xz} \end{Bmatrix} = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{21}}{E_2} & -\frac{\nu_{31}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{12}}{E_1} & \frac{1}{E_2} & -\frac{\nu_{32}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{13}}{E_1} & -\frac{\nu_{23}}{E_2} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{12}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{23}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{13}} \end{bmatrix} \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{xz} \end{Bmatrix}$$

mechanical properties of the overall desired structure. As anticipated in previous chapters the main problem related to the numerical analysis of cellular solids is their geometry complexity. Since they are made of countless interconnected struts, lattices require a load of elements to reproduce accurately a simulation model. This means that running a numerical analysis of a lattice structure can be time expensive. For this reason, in literature, several approaches can be found and one of the fastest to process the analysis is the homogenization method. However, this approach works only for periodic lattice structures because it consists in characterizing only a block of the structure, creating a fully solid model having properties equivalent to those of the analyzed lattice block. Since the periodicity of the structure is valid, the lattice can be substituted by the equivalent solid which needs much fewer elements to be processed.
However, today's cutting-edge manufacturing technologies permit to create easily also non-periodic lattice structure and this requires a certain tunability of the simulation model. For this reason, the work of this thesis aims to create a beam numerical model able to adapt to all the possible details characterizing a lattice structure and a Finite Element Approach is used to analyze it.
Since this work studies only in plane linear elastic deformations of cellular solids, mechanical properties are expressed in terms of two Young's moduli E and a Shear modulus G. The formers are calculated by performing compression or tensile tests

(in the range of linear elastic deformation they are equivalent) while the latter by means of shear tests.

To minimize the computer processing time the mechanical tests are performed only on a quarter of the whole structure. Depending on the type of mechanical test to be run, particular boundary conditions must be set at the cut sections.
The nature of the problems imposes a set of symmetry boundary conditions for the compression test while anti-symmetry boundary conditions for the shear test. On

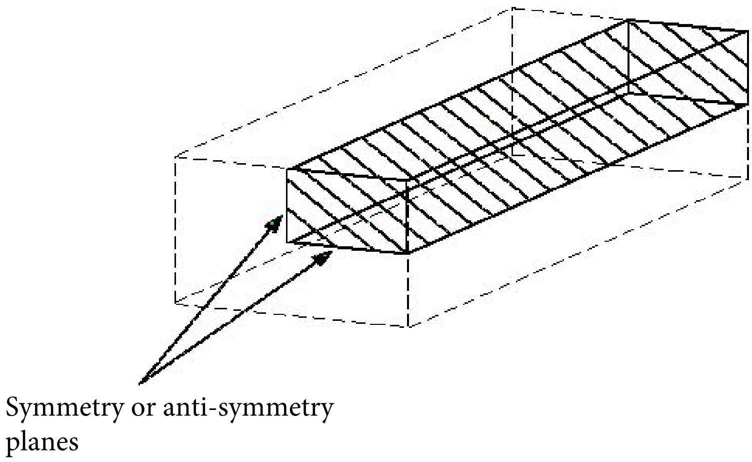


Figure 5.1: Symmetry or anti-symmetry planes

the contrary, anti-symmetry boundary conditions allow the displacement along the direction perpendicular to the plane and the rotations about the directions tangent to the plane.

5.2 Compression test

The strength of materials is detectable from tensile or compression tests. The response of a component under tensile or compressive load can be completely different under plastic deformation, especially for lattice structures because of the different failure mechanisms that occur when struts are stretched or compressed. However, under the linear elastic deformation assumption, the rigidity of the material is independent of the load direction but since the experimental tests of the actual sample of lattices are performed under compressive deformation, this latter is preferred.
Usually, mechanical tests are performed under displacement control because it

permits to study deeply how materials get deformed preventing sudden failures or collapses of the structure. However, this information is not relevant within the elastic deformation field but it is worth highlighting all the criteria adopted during the analysis. Since the lattice structure is considered to be orthotropic, it is clear that the response is different along with the various directions and one compression test must be performed for each of them.
As anticipated in the previous paragraph, to perform correctly a mechanical analysis on the quarter model it is necessary to set appropriate boundary conditions, in this case, *symmetry b.c.*. From theory it is known that a plane of symmetry allows the rotation about the direction perpendicular to it and the displacements along with the directions tangent to the plane. So as can be observed in Figure 5.2, along the *plane x-z* the displacement UY and rotations $ROTX$, $ROTZ$ are locked, along the plane *plane y-z* the displacement UX and rotations $ROTY$, $ROTZ$ are locked and all the remaining degrees of freedom are free. For the compression test along

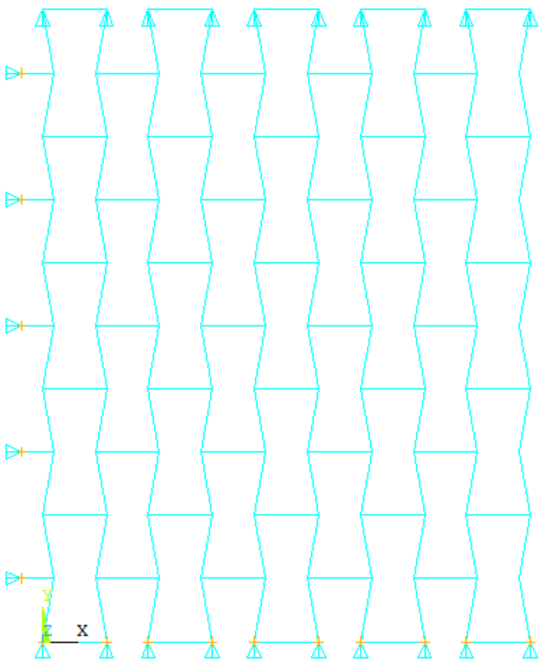


Figure 5.2: Constraints for compression test along with *y-direction*

the *y-direction*, a vertical downward displacement of about $2mm$ on all the upper surface is imposed. The deformation given to the structure generates reaction forces internal to the structure which are required to compute E_y. In order to compute the elastic modulus along this direction many and easy mathematical steps must be followed. It is possible to identify the physics of the problem by recalling the elastic constitutive Eq. 5.1 and isolating E_y it gets:

$$E_y = \frac{\sigma_y}{\varepsilon_y} \tag{5.2}$$

where σ_y it is easily computed by summing up the components along *y-direction* of all reaction forces R_y of the N nodes on *x-z plane* at the bottom divided by the surface A_{y0} lying on the same plane. This can be expressed in formulae as follows:

$$\sigma_y = \frac{F_y}{A_{0x}} \tag{5.3}$$

where:

$$F_y = \sum_{k=1}^{N} R_y \tag{5.4}$$

$$A_{0x} = x_0 z_0 \tag{5.5}$$

$$\varepsilon_y = \frac{\Delta y}{y_0} \tag{5.6}$$

with x_0 y_0 and z_0 representing respectively the total initial width height and depth of the structure while Δy is the vertical displacement imposed by the user to perform the compression test. In order to compute E_x a compression test along *x-direction* is performed so the simmetry contraints are kept the same as before while a horizontal leftward displacement Δx of about $2mm$ on all the right-side surface is imposed. Here:

$$E_x = \frac{\sigma_x}{\varepsilon_x} \tag{5.7}$$

where

$$\sigma_x = \frac{F_x}{A_{0y}} \tag{5.8}$$

$$F_x = \sum_{k=1}^{M} R_x \tag{5.9}$$

$$A_{0y} = y_0 z_0 \tag{5.10}$$

$$\varepsilon_x = \frac{\Delta x}{x_0} \tag{5.11}$$

with M the number of nodes lying on the vertical left-end surface.

When studying the intrinsic *in-plane* mechanical properties of a homogeneous infinite lattice structure the procedure indicated above needs a convergence study for one main reason. The boundary conditions generate distortions so the structure should be large enough to make this effect marginal, thus negligible.

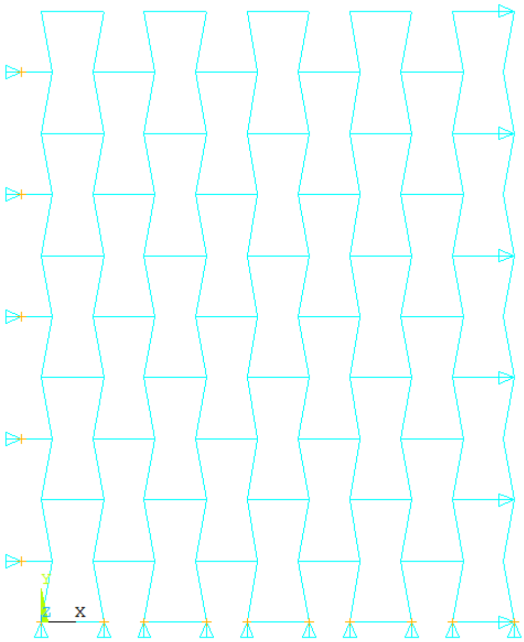


Figure 5.3: Constraints for compression test along with *x-direction*

5.3 Shear test

The shear test is performed to evaluate the ability of a certain body to withstand shear stresses. Keeping working within the limits of elastic deformations, the just mentioned mechanical property is quantified by the shear modulus G.
As in the compression test, even here it is possible to express a mathematical relation between stresses and deformations:

$$\tau = G\gamma \tag{5.12}$$

where τ is the shear stress and γ the angular deformation. Even here, to reduce the size of the problem the numerical analysis is performed on a quarter of the entire model but the constraints to be set at the cut sections are meant to be *anti-symmetry boundary conditions*. Thus, they allow longitudinal displacement normal to the interested plane and the rotations about the axes tangent to the same. For the purposed model, at the bottom *x-z plane* UX UZ and $ROTY$ while at the vertical left-end *y-z plane* UY UZ and $ROTX$ are locked.
Here the deformation of the body is produced by applying a sliding displacement to the external surfaces depending on the direction must be analyzed. In the plane investigation, it is enough to performe the shear test only along with one direction. In this case the test is done by applying a rightward displacement ΔX of about $0.1mm$ to the top horizontal surface. This deformation produces shear stress inside

the structure useful to compute the shear modulus G_{yx}, Figure 5.4. By recalling

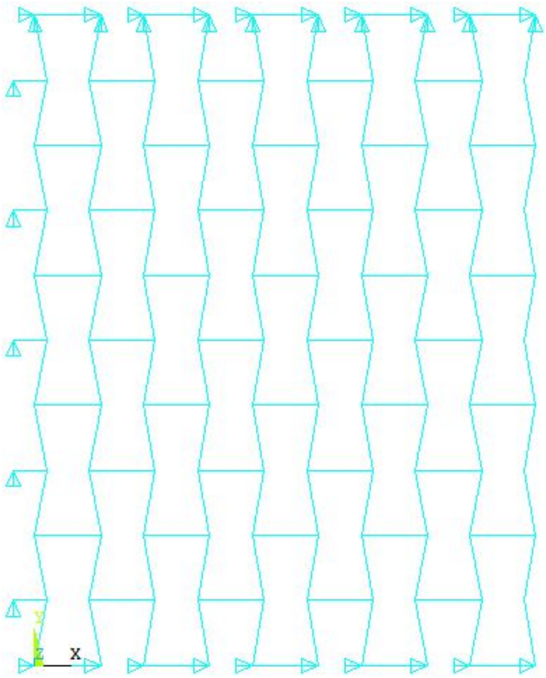


Figure 5.4: Constraints for shear test along with *x-direction*

and rearranging Eq. 5.12 it gets:

$$G_{yx} = \frac{\tau_{yx}}{\gamma_{yx}} \tag{5.13}$$

where τ_{yx} is the shear stress computed by summing up all the *x-components* of reaction forces at those nodes standing on the bottom surface parallel to *x-z plane* while γ_{yx} is the angular defomation. In formulae:

$$\tau_{yx} = \frac{F_x}{A_{0x}} \tag{5.14}$$

$$F_x = \sum_{k=1}^{N} R_x \tag{5.15}$$

$$\gamma_{yx} = \frac{\Delta x}{y_0} \tag{5.16}$$

5.4 Convergence study

A FE approach allows the discretization of a continuum model and the study of the deformation of all body by interpolating the displacements of known points, the

nodes. It is clear that the finer is the mesh, thus the number of known points, and the more accurate are the results of interest. This effect has been well observed during the choice of the elements to be used for building the structure and it has also been noticed that the chosen Eulerian beam doesn't need convergence for the nodal solution.
However, when a lattice structure is homogeneously developed the overall mechanical properties are dictated by the topology characteristics. On the other hand, if they must be studied by running the simulation above described, the results can be altered by the *b.c.*. So the structure to be analyzed should be sufficiently large in order to make the perturbation caused by *b.c.* negligible. As an example of a convergence study, a honeycomb structure with regular hexagons is chosen because this geometry behaves isotropically in the plane so it is easier to detect many effects due to topology and others to *b.c.*.

The convergence study is performed by increasing the number of cells lined up along the *x-axis* and *y-axis* and for the sake of simplicity, this number is increased proportionally. From Figure 5.5 it is possible to notice how E_x converges from the bottom upwards and E_y from the top downwards. This different trend is due to the different geometry along the two axes and when the number of cells is low the anisotropy is consistent. Furthermore, the large gap between the two Young's moduli at low *number of cells* is also attributable to *b.c.*. The constraints and loading conditions imposed on the various surfaces generate distortions around the boundary areas and the deformation of unit cells cannot deform uniformly as it happens at the core of the structure. On the contrary, when the number of cells is quite large the cell size becomes infinitesimal compared to that of the whole structure and the anisotropy disappears. As consequence, E_x and E_y tend to the same value.
Since the convergence of intrinsic mechanical properties of the cellular structure is strictly related to the size of the whole structure, thus to the number of cells involved to build it, also the solution of the shear modulus will convergence for the same number of cells at which elastic modulus converges.
The results of this discussion are reported in Figures 5.5 and 5.6, where k represents a constant multiplier and it is equal to 10^4.

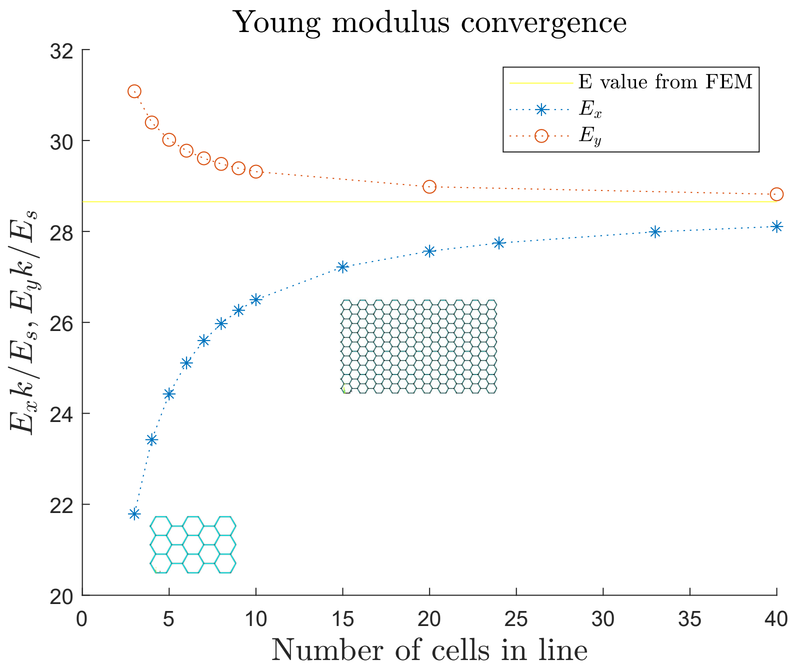


Figure 5.5: Elastic moduli convergence for a regular honeycomb

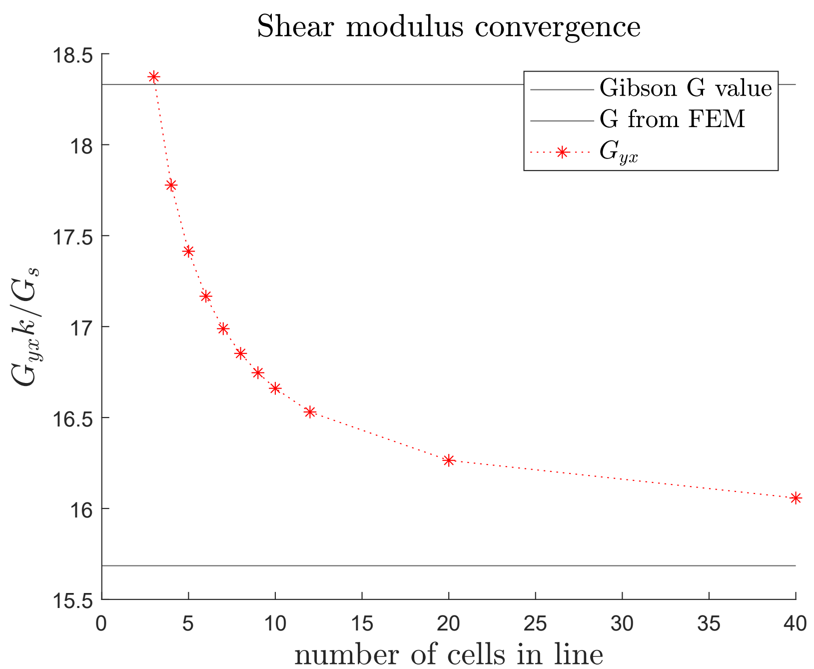


Figure 5.6: Shear modulus convergence for regular honeycomb

Chapter 6

Design and Optimization of the structure

6.1 Introduction

This work aims to provide a method to design and optimize a three-dimensional lattice structure through a simulation model able of including all the variation that may come from manufacturing processes. Specifically, the objective of the project is about the replacement of a honeycomb structure with a lattice one. Since this is a development of previous work, the architecture of the lattice to be used has already been conceived by Piccagli [15].
The architecture is a starting point for a designed phase, but since the mechanical properties of the lattice structure depend on the multitude of geometrical features, in this work a methodic approach is used to figure out the optimal design. In particular, the design process is developed as follows: Design of Experiments approach is used to find correlations between geometrical features of the lattice, choice of optimal lattice, analysis of the designed lattice through the as-fabricated lattice simulation model, study of mechanical discrepancies between designed and as-fabricated lattice, and final optimization of the structure.

6.1.1 Initial honeycomb structure

The initial honeycomb structure shown in Figure 6.1 is made of 316L stainless steel and its unit cells are characterized by the following relevant geometrical features:

- *length of inclined struts*, $L = 20$ $[mm]$
- *length of horizontal struts*, $h = 20$ $[mm]$
- *thickness of cell wall*, $t = 1$ $[mm]$

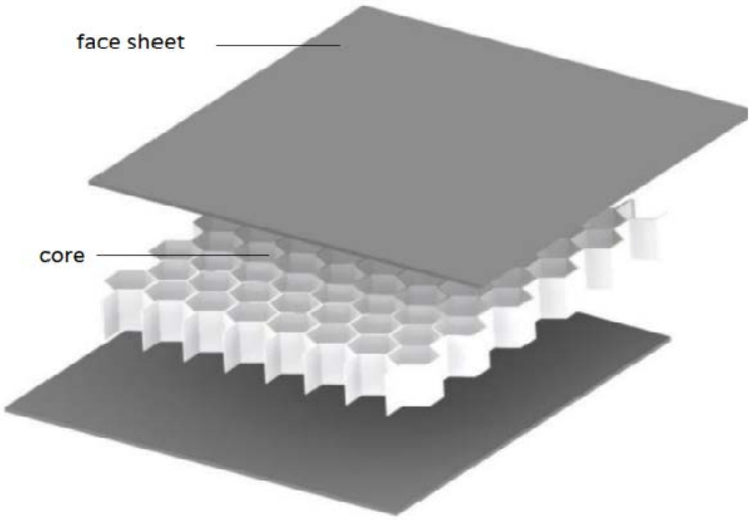


Figure 6.1: Honeycomb structure

The *in-plane* mechanical properties of this structure are characterized by the following moduli:

- $E_x \approx 59$ $[MPa]$
- $E_y \approx 59$ $[MPa]$
- $G_{yx} \approx 13$ $[MPa]$

and with this configuration it has the following relative density

- $RD = \rho/\rho_s = 0.0577$

where ρ is the density of the cellular solids while ρ_s is the density of bulk material. So, as anticipated, this honeycomb should be sobstituted by a lighter one and with much higher stiffness along *y-direction*, thus E_y must be increased.

6.1.2 From honeycomb to lattice structure

The optimization of mechanical properties and the weight minimization represent always an engineering challenge. So, even though the honeycomb structure is categorized as lightweight, the saving of material is a goal that keeps renewing itself. For this purpose, the first idea is the replacement of cell walls of the honeycomb with a particular set of interconnecting struts, in the way to obtain a lattice structure. The latter is characterized by a very detailed geometry that makes the design phase and fabrication process difficult. Specifically, nowadays manufacturing

technologies still have some resolution limitations so it is very unlikely to obtain a fabricated lattice the same as the designed one. So, considering the geometrical and structural discrepancies introduced by the fabrication process, the design of lattices becomes even more articulated. For this reason, this work aims to set a design procedure able of taking in account also the structural variations introduced during manufacturing.
The basic architecture of the lattice is illustrated in Figure 6.2 and it is provided by previous studies [15]. This configuration has been designed to give bending dominated *in plane* deformation mechanism to have high elastic energy absorption and a stretch dominated *out of plane* deformation mechanism so guaranteeing high rigidity. However, once this paper has put in evidence all the fabrication variations, this structure still deserves deeper analysis and for this reason here a methodic procedure thought to identify the optimal topology is reported: Design of Experiments.

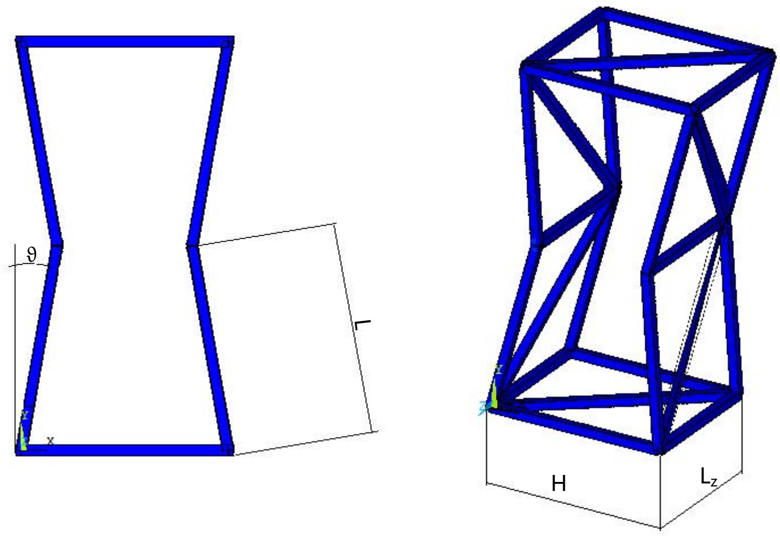


Figure 6.2: Unit cell of the lattice structure

6.2 Design of experiments

Design of Experiments (DOE) is an approach aimed to design and organize experiments to analyze systematically the nature and the outcomes of them. Thus, the main goal is to detect as much information as possible about a process minimizing the number of experiments. Through DOE, the costs of a project can drastically decrease.
Matrices are used to combine the various parameters involved in the process and to

get important evaluations by using a few experiments. This design phase consists mainly of understanding how input parameters affect the output. For this purpose, the engineering designer should choose a predetermined number of input parameters to be analyzed at different levels to detect many correlations between each other and how sensitive the output is to their variations.
Input parameters can be controllable or uncontrollable. In the case of prediction of lattice structures mechanical properties, the controllable parameters can be related to geometrical features such as dimension and inclination of struts while the uncontrollable parameters can be related to all variations determined by external factors.
The purpose of this work is the substitution of a honeycomb structure with an auxetic lattice one to increase the *in-plane* stiffness along with *y-direction* reducing its weight, by considering also the variations can be produced during the fabrication process. Lastly, a DOE turns out to be a very useful approach to understand how mechanical properties change by varying the geometrical features of the lattice.

The DOE is performed as follows:

- *setting of the problem and objectives*
- *identification of most interesting input variables affecting the output*
- *selection of the values that input variables can take*
- *study of the correlation existing between variables*
- *choice of the optimal solution*

It is worth saying that at the beginning this procedure is executed for an idealized lattice structure like that coming from a CAD model. Then the same study is performed by applying all the corrections due to the variations of the fabrication process.

6.2.1 Setting of the problem and objective

As already explained, the aim of this project is to replace the core of a honeycomb sandwich structure with a lighter one, thus a lattice structure, without neglecting the problems can come from the fabrication process. Since sandwich structures are generally compression-loaded *in plane* and considering this investigation is limited only to elastic deformations, the objectives of this design procedure are to increase the stiffness along the load direction, thus the equivalent Young modulus in this specific case along *y-direction* without penalizing too much the properties in other directions, and to reduce the weight as much as possible. Since the lattice is thought to be produced with the same material of the honeycomb, the weight saving is related only to the geometrical features of the lattice. Furthermore, it is important to highlight that one of the design constraints is the auxeticity of the lattice to obtain the peculiar macro-deformation mechanism characteristic of those materials with negative Poisson's ratio.

6.2.2 Identification of input and output factors

At this point of the paper, it is well known that the mechanical properties of the lattice structure are strongly affected by geometric features of the cells so after the overview of the problem given in the last paragraph, it is easy to deduce that the geometrical parameters defined in Figure 6.2 represent the input factors while the *in plane* elastic, shear moduli and relative density represent the output factors. Since the number of geometrical parameters is large, a preliminary study is done by varying them singularly to observe the effect of each parameter on the output of interest.
At this point it is worth remembering that mechanical properties of the structure are detected by performing every time the numerical tests illustrated in chapter 5. For the sake of clarity E_x and E_y are found by executing the compression tests along *x and y directions* respectively while G_{yx} through the shear test along *x-direction*. Since it is only an explorative phase, the Young and Shear moduli shown in the following graphs are preferred to be normalized. The input parameters to be analysed are θ, A, L, h, L_z.

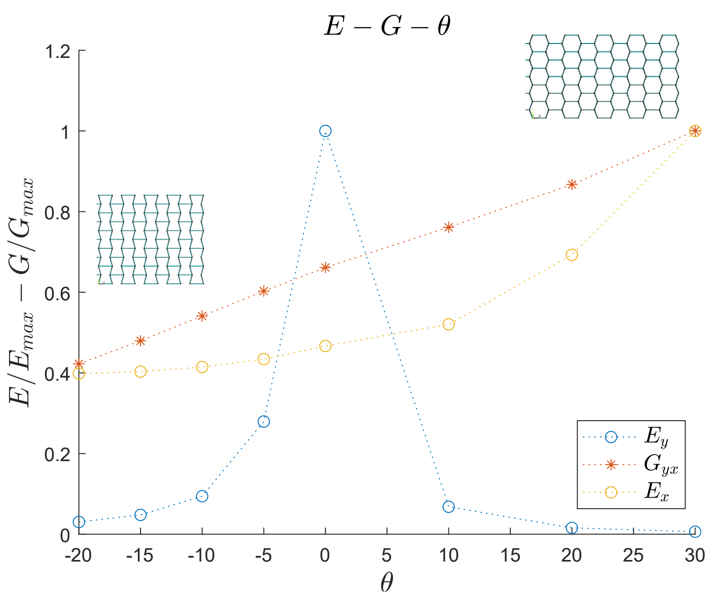


Figure 6.3: Moduli varying inclination angle

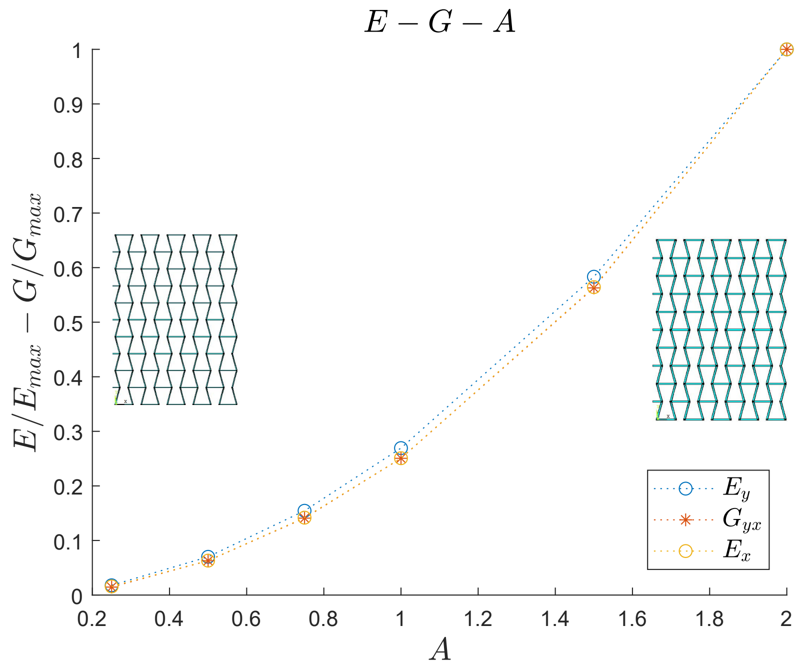


Figure 6.4: Moduli varying cross-section area

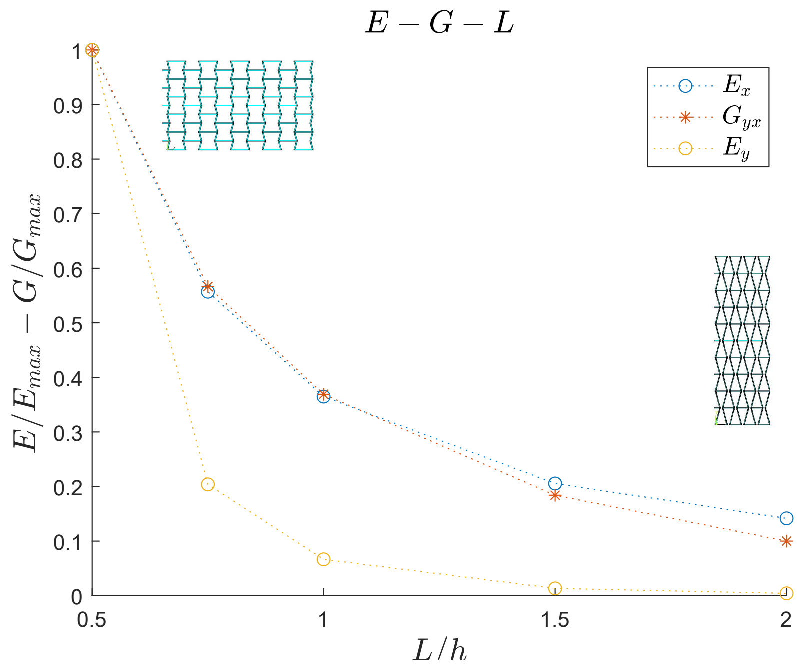


Figure 6.5: Moduli varying inclined struts length

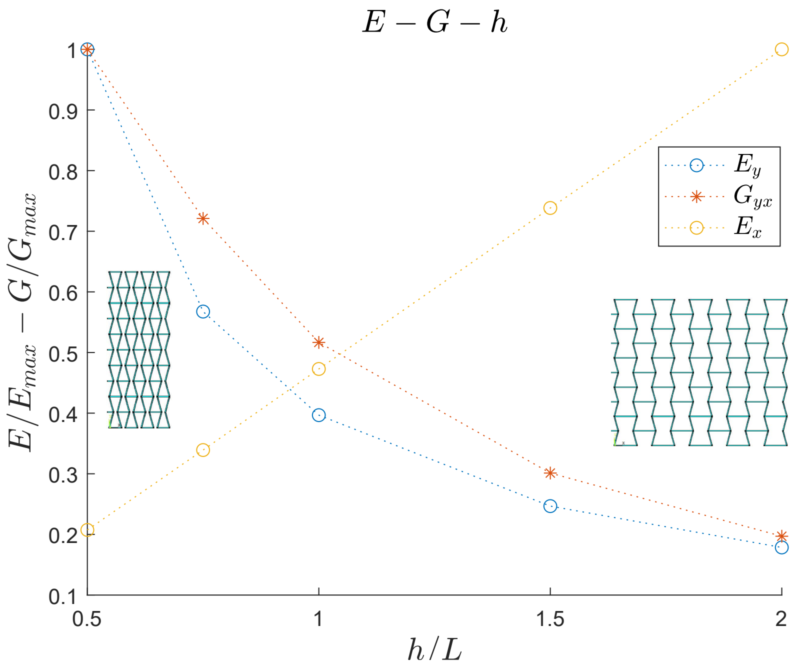


Figure 6.6: Moduli varying horizontal struts length

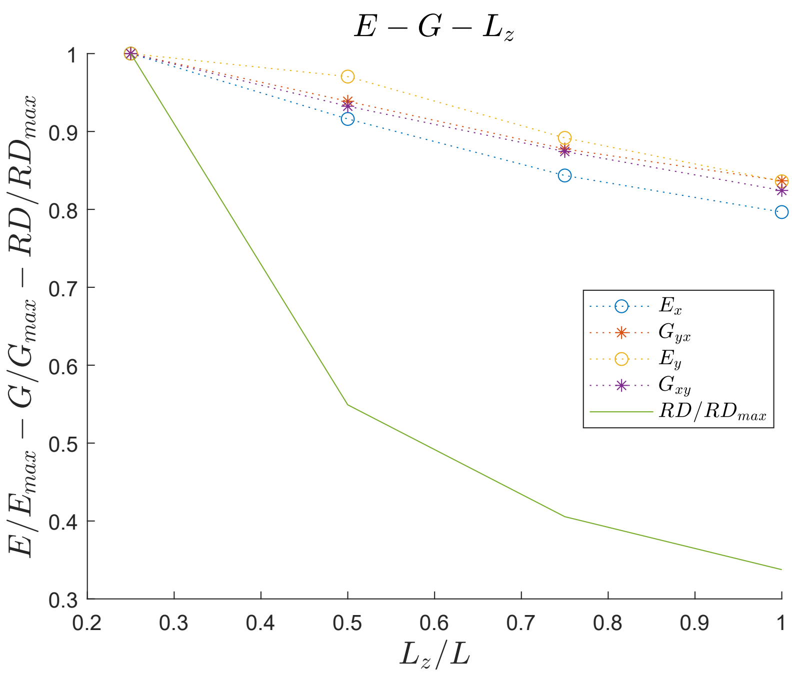


Figure 6.7: E and G moduli and relative density by varying L_z

From Figure 6.3 it is possible to observe how much E_y is sensitive to the variation of struts inclination angle. In particular, there is a range between $-5°$ and $10°$ in which Young's modulus along y increases drastically. This behavior is justified by the fact that, when the structure is compressed along this direction, the more the lateral struts are vertically orientated the less is the bending moment acting on these struts. So the struts are more subjected to stretching stresses and since the axial stiffness of the strut is larger than the bending one, the overall structure gains in E_y.
On the other hand, this reasoning has the opposite effect on E_x. More specifically, this modulus does not seem to be sensitive to angle variations when the topology has an auxetic configuration but according to what it has been said before when the struts start to orient along *x-axis* E_x shows an increasing trend. Lastly, the shear modulus shows a decrease in value as long as the structure becomes more auxetic.
From now on, since the auxeticity is a design constraint all the following tests are done on a structure with a constant strut inclination angle of about $-10°$. So when the cross-section is varied, lattice structure mechanical properties always increase. This trend is shown in Figure 6.4 is quite obvious because considering Eq. 4.4 the thickening of struts implies an increase of stiffness.
As Figure 6.5 shows, at increasing the size of L the structure undergoes a softening effect instead. This is explained by the fact that in any stress condition the struts are longer and since their stiffness is inversely proportional to the length, a softening effect occurs. From Figure 6.6 it is possible to observe that the lengthening of horizontal struts causes the same effect of the lengthening of L except for the trend of E_x that shows a strengthening trend. Lastly, Figure 6.7 shows how the contribution of L_z on mechanical properties is not so relevant as others. The Young and Shear moduli increase with decreasing L_z. However, the gain of mechanical properties at the expense of the increase in density, thus weight, is considered unsatisfactory from the beginning so this geometrical parameter is not involved in the following step of DOE.
Furthermore to reduce even more the size of DOE, variables h and L have been put together in only one dimensionless parameter $\alpha = h/L$. This first step of analysis has highlighted how the variations of single parameters affect the overall behavior of the structure. Even if this investigation provides the necessary information to begin the design process, it is not enough since it does not take into account the combination of multiple factors. For this purpose, a *two-way interactions* turns out to be a good approach to find correlations between factors.

6.2.3 Selection of levels

So just to recap, the input parameters are the inclination of the struts respect to the vertical θ, the cross-section of the struts A, and the dimensionless parameter $\alpha = h/L$ which is tuned by varying h. On the other side the output factors are the *in plane* elastic and shear moduli E_x, E_y, G_{yx}. Relative density ρ/ρ_s is also a factor of interest but it does not represent an output factor for DOE because it can be easily computed by every structure configuration set to perform the various analysis. So this does not produce the necessity of new experiments.
In order to study the correlation between multiple factors, their concomitant variation must be imposed. It is clear that by imposing a variation, each factor should take different values so the engineering designer must choose which are the most interesting and how many values be studied. Before taking this decision it is worth saying that the number of input parameters and levels are related to the number of analysis must be run to perform a complete DOE. In particular, they are linked by the following mathematical formulation:

$$n = (l)^f \tag{6.1}$$

$$N_{exp} = m \cdot n \tag{6.2}$$

where n is the number of combinations, N_{exp} is the total number of experiments, f is the number of input factors, l the number of levels and m the number of output parameters. And in the specific case, as abovementioned, $m = 3$ and $f = 3$. Only l must be chosen and since the trend of the outputs plotted by varying singularly the input factors is quite regular, two levels are considered sufficient.
The choice of values the parameters must take at different levels is dictated by the objectives of the project. Here the main goal is to increase Young's Modulus along *y-axis* without too much penalizing other output parameters. So considering the auxeticity requirement and observing the graphs studied in the previous paragraph, the following parameters have been considered the most interesting:

- $\theta(-) = -15°$, $\theta(+) = -5°$
- $\alpha(-) = 1$, $\alpha(+) = 1.5$
- $A(-) = 1\ mm^2$, $A(+) = 2\ mm^2$

It is worth highlighting that for the sake of simplicity the data indicated with a $(-)$ refer to *level 1* while $(+)$ to *level 2*. Once all the input parameters are set it is necessary to analyze all the possible combinations, listed in Table 6.1, in the way to study the multifactor correlations and how they affect the output. However considering that output parameters are four, this series of experiments must be

Table 6.1: Combinations of values used to perform experiments

	Input factors			*Output parameters*			
Comb	θ	α	A	E_y [MPa]	G_{yx} [MPa]	E_x [MPa]	ρ/ρ_s
1	$-$	$-$	$-$	54,74	0,76	2,40	0,0227
2	$-$	$-$	$+$	211,65	3,03	9,53	0,0305
3	$-$	$+$	$-$	32,83	0,47	3,85	0,0149
4	$-$	$+$	$+$	126,99	1,88	15,26	0,0201
5	$+$	$-$	$-$	319,13	0,96	2,54	0,0178
6	$+$	$-$	$+$	1017,55	3,81	10,12	0,024
7	$+$	$+$	$-$	205,15	0,53	3,86	0,0127
8	$+$	$+$	$+$	655,30	2,12	15,33	0,0171
	Average value			327,92	1,70	7,86	0,02

repeated for each of them. In fact, by solving Eq.(6.2) comes out that $N_{exp} = 24$ and the results of them are even reported in Table 6.1 under the voice of *"Output parameters"*.

6.2.4 Factorial analysis and two way interactions

As seen so far, the selection of input factors and levels let to gather a series of information useful to run the minimum number of experiments to study all the correlations existing between parameters. Now an interpretation of the results obtained from experiments is needed and also a correlation between input factors should be found out. Since all the needed experiments are already carried out, as suggested in [16] a full factorial analysis can be performed treating *"the data as a series of paired comparisons, one parameter at a time"*. Furthermore, the effect of the jth factor (E_j) can be computed through the following relation:

$$E_j = \frac{\sum_{i=1}^{n} l_{ij} Y_i}{\sum_{i=1}^{n} Y_i} \qquad (j = 1,2, ..., n) \tag{6.3}$$

where Y_i is the response variable for the ith combination and as reported in Table 6.1 the parameter l_{ij} is equal to -1 when the jth factor takes the lower level and to $+1$ at the higher level. While the interaction between jth and kth factors is obtained as follows:

$$I_{jk} = \frac{\sum_{i=1}^{n}[(l_{ij}\ l_{ji})Y_i]}{\sum_{i=1}^{n} Y_i} \qquad (j = 1,2,....,n) \tag{6.4}$$

Lastly, to give an idea of the interactions occurring between factors a graphical representation can be done plotting the variation of the normalized output by varying the two input parameters understudy at the two different levels. As can be noticed in Table 6.1, when a specific combination of levels between two factors appears twice then an average of the output can be done. The alternative procedure is to perform a *three-way interactions* which is more accurate but since the error produced by the *two-way interactions* study can be considered negligible, the latter is preferred because it is faster.

6.3 Results and discussion

To detect the effect of each geometrical factors on the mechanical properties of the structure, a full factorial analysis has been carried out. Specifically, the just mentioned effect has been studied by solving Eq. 6.3 and the results are listed as percentages in Table 6.2. These factorial effects are expressed as averaged

Table 6.2: Effects of geometrical parameters on Young's and Shear moduli

	θ[%]	α [%]	A [%]
E_y	67,51	-22,22	53,35
G_{yx}	9,39	-26,25	59,93
E_x	1,30	21,80	59,75

values considering computed over all the conducted experiments. For instance, the transition of inclination angle θ from the low level $-15°$ to the high level $-5°$ produces an *"average increase"* in E_y of about 67,51% respect to the *"Average value"* reported in the last row of Table 6.1.

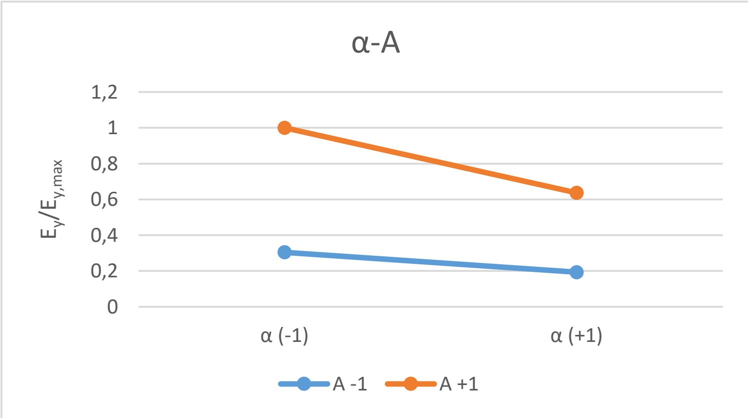
On the other hand, it is not said that each parameter, varied alone, produces the same effect on the desired output compared to when it is varied in combination with another parameter. To highlight this difference, a two-way interaction study has been conducted by solving Eq. 6.4 and results are shown in Table 6.3.
After the illustration of results, it is interesting observing how single inputs affect different output parameters disparately. In particular θ produces a strong effect

Table 6.3: Effect of two way interactions between geometrical parameters

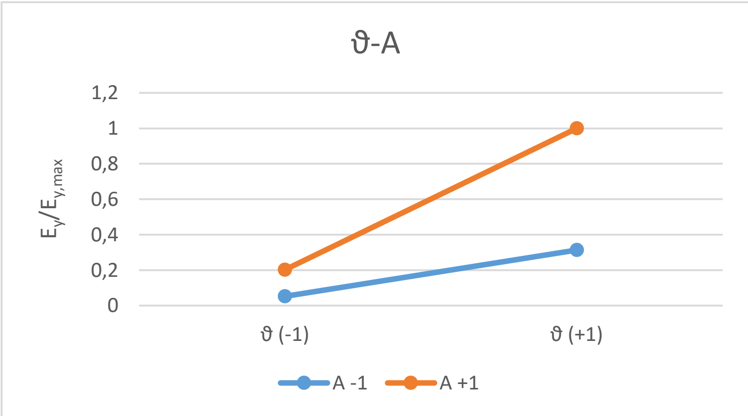
	θ-α[%]	θ-A[%]	α-A[%]
E_y	-14,09	34,21	-11,86
G_{yx}	-5,02	5,61	2,29
E_x	-1,04	0,80	13,00

on E_y but it loses its dominance on other output parameters. On the contrary, A seems to have a significant and similar influence on all mechanical features of the structure. Lastly, α shows a softening effect on all the output parameters except E_x which increases by 20% respect to its average value. Actually these results are completely in accordance to the trends shown in Figures 6.3, 6.4, 6.5, 6.6, confirming thus the validity of the approach used.

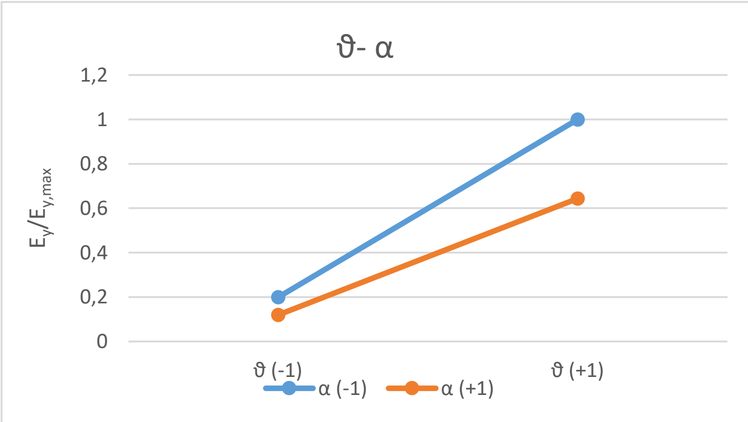
The two way interactions have been find out through the mathematical expression 6.4: the results are listed in Table 6.3 and can be also observed graphically in Figures 6.8, 6.9, and 6.10.
The strongest interconnection can be observed between θ-A related to E_y with about 34 percentage points, in fact in Figure 6.8b the divergence between the two line is quite marked. On the contrary, the lowest interaction effect is shown again by the θ-A but this time related to E_x and as predictable, Figure 6.10b shows two parallel segments. This point underlines how counter-intuitive the interactions between input factors can be and this demonstrates that a planned Design phase turns out to be useful, even to detect non-obvious information.



(a) Interaction between α and A

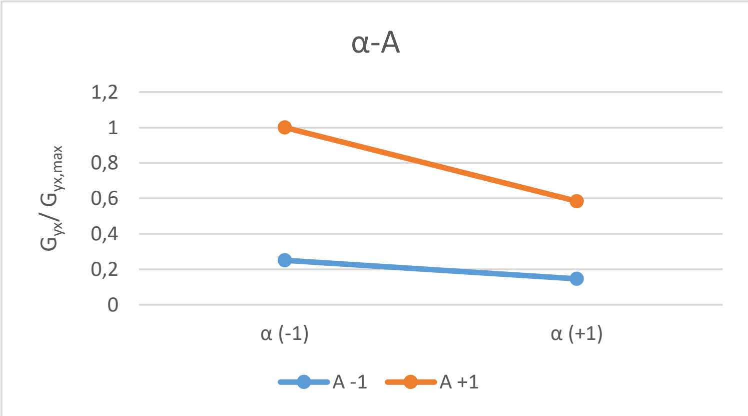


(b) Interaction between θ and A

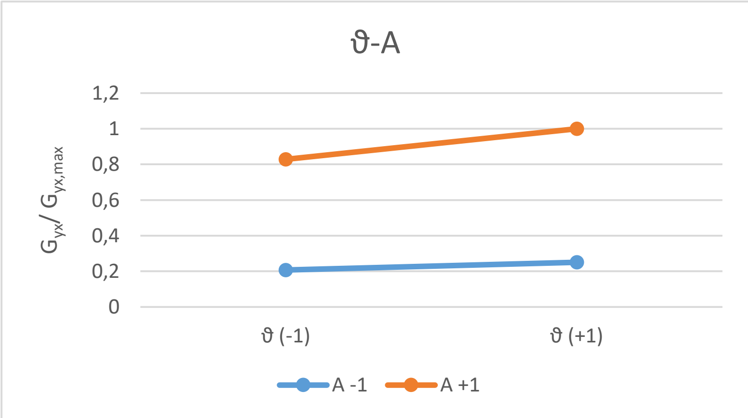


(c) Interaction between θ and α

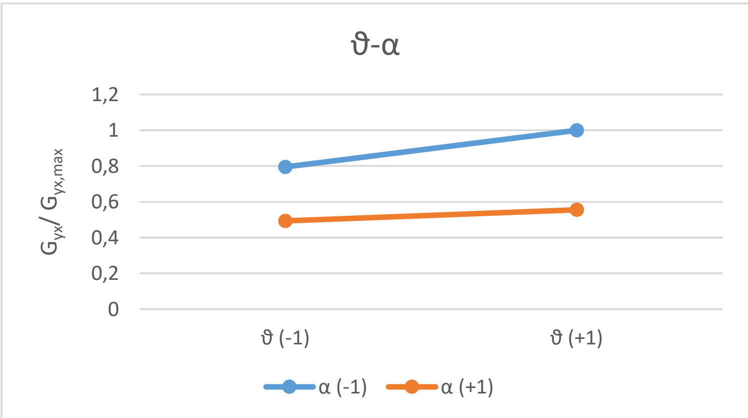
Figure 6.8: Two-way interactions for E_y



(a) Interaction between α and A

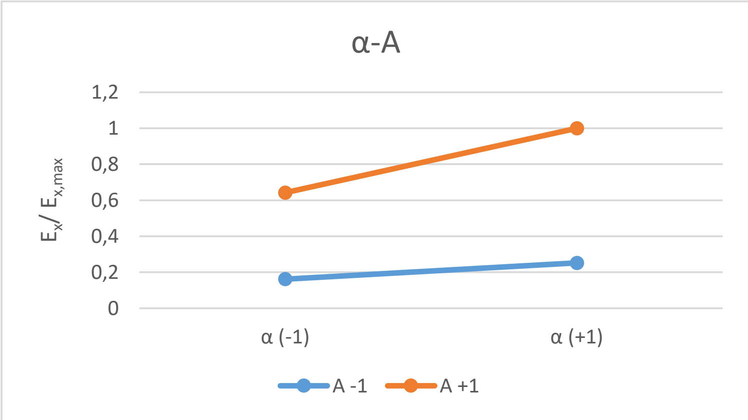


(b) Interaction between θ and A

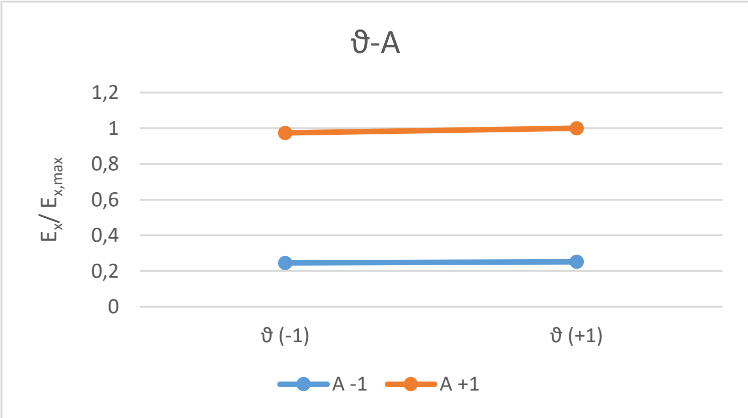


(c) Interaction between θ and α

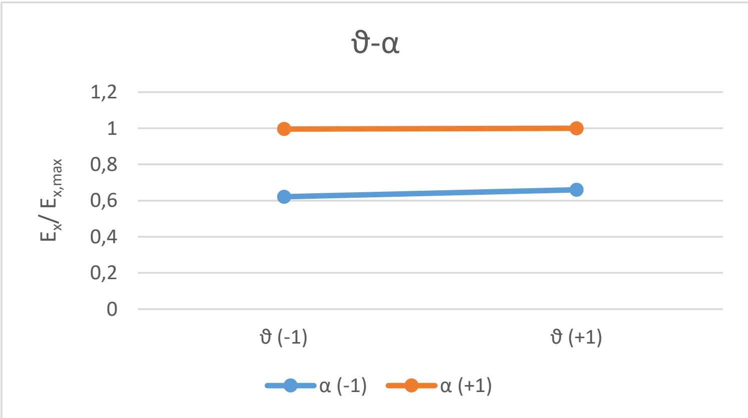
Figure 6.9: Two-way interactions for G_{yx}



(a) Interaction between α and A



(b) Interaction between θ and A



(c) Interaction between θ and α

Figure 6.10: Two-way interactions for E_x

6.3.1 Choice of the designed structure

Since the constraint of the design phase was limited to the increase of E_y trying to obtain the lightest structure as possible, *Combination 8* was considered the most suitable. As can be noticed from Table 6.4, the 8th combination results to

Table 6.4: Properties of the chosen designed structure respect to average values of DOE

	Input factors			*Output parameters*			
Comb	θ	α	A	E_y [MPa]	G_{yx} [MPa]	E_x [MPa]	ρ/ρ_s
8	+	+	+	655,30	2,12	15,33	0,0171
	Average value			327,92	1,70	7,86	0,02
	Difference[%]			+99,83	+24,7	+95,04	-14,5

be a good compromise even because it shows all the mechanical features over the average values and a relative density lower than the average. In particular E_y shows a value almost double than the average while the relative density is about 14,5% lower than the average.
However, the analysis conducted so far is not sufficient for an accurate design phase of a lattice structure because variations due to the fabrication process should be included and they can even compromise the effects of interaction between input parameters studied so far. For this purpose, many steps of a planned design of experiments must be done for the as-fabricated lattice structure.

6.3.2 DOE for the as-fabricated lattice structure

As observed in subsection 3.4.1 a fabrication process is quite compromising for the structural integrity of the lattice: porosity, roughness, and dimensional inaccuracies may penalize mechanical properties of the structure. Specifically, it has been noticed that dimensional inaccuracies are the main geometric variation of struts representing thus the most critical structural influencing factor. For this purpose, as explained in subsection 4.4.3 a numerical model able of differentiating the features of each strut has been created. So, it is possible to observe all the discrepancies existing between the designed lattice and the as-fabricated one.
In figure 6.11, 6.12, 6.13 a qualitative representation of how geometrical parameters affect the varied structure and a comparison with the designed one can be made. From these graphs can be suddenly noticed that the discrepancy between the two models is negligible so the interactions between geometrical factors can be considered the same as those observed in the designed structure.

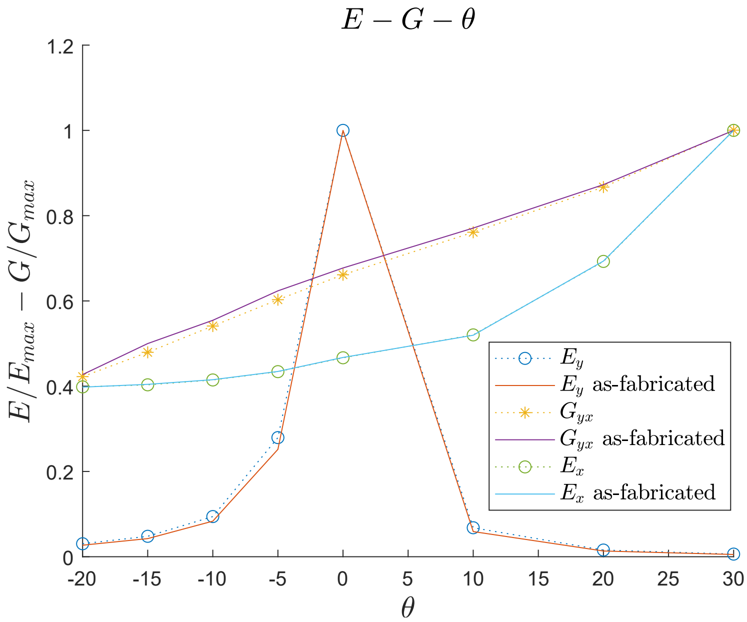


Figure 6.11: Qualitative comparison between designed and fabricated structure at varying θ

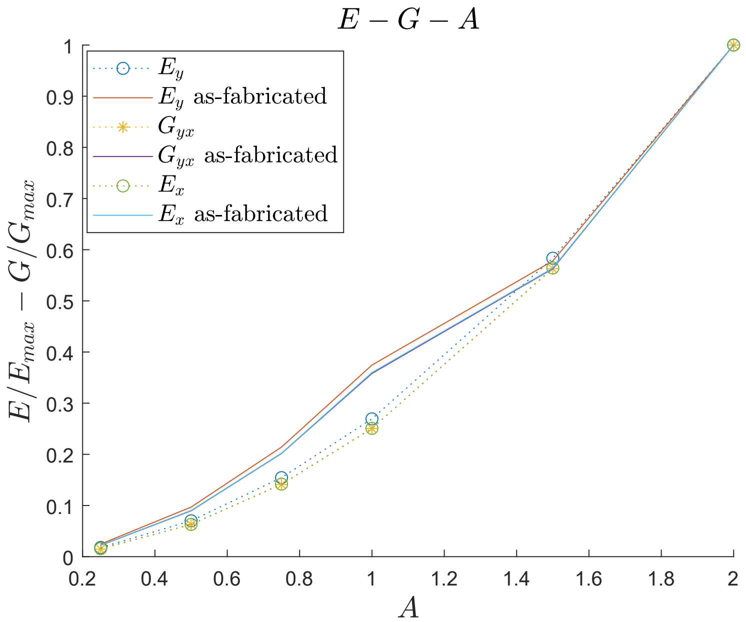


Figure 6.12: Qualitative comparison between designed and fabricated structure at varying A

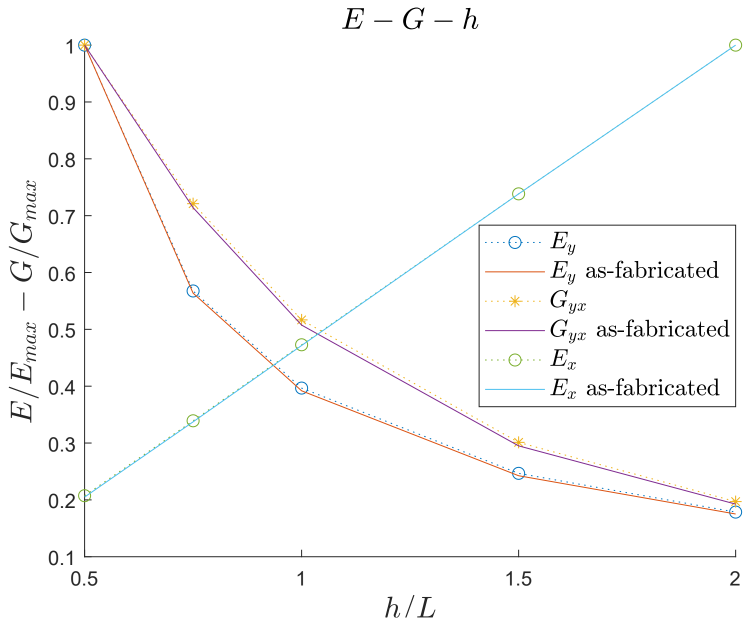


Figure 6.13: Qualitative comparison between designed and fabricated structure at varying h

6.3.3 Comparison between designed and as-fabricated structure models

The conclusion deduced in the last paragraph is just a qualitative observation and not a quantitative one. So, to understand the difference in mechanical properties the following steps are performed:

- building the structure identified in previous DOE (*combination 8*) by including the fabrication variations defined in the model explained in subsection 4.4.3
- running all the mechanical analyses to quantify the Young's and Shear moduli of the as-fabricated structure
- comparison between results of the as-fabricated and designed structure

The first step is performed by generating the structure using the *Matlab script* and the mechanical analyses are conducted by using the same values reported in Figure 4.22. In the end, the results are listed in Table 6.5
As predictable, the difference in mechanical properties between a designed structure and a fabricated one is huge and processing the model created in this work there is a loss at least of 44% in every *in-plane* modulus. So, it is clear that the fabricated structure could not withstand the loading conditions it is destined for.

Table 6.5: Mechanical properties differences between designed and fabricated lattice structure with the same starting geometric features

Comb 8	*Input factors*			*Output parameters*		
	θ	α	A	E_y [MPa]	G_{yx} [MPa]	E_x [MPa]
Designed structure	+	+	+	655,30	2,12	15,33
As-fabricated structure	+	+	+	366,21	1,05	8,45
	Difference[%]			-44,11	-50,25	-44,86

6.3.4 Optimization of the structure

This work proposes also an optimization of the lattice in order to get the mechanical features obtained in the first design. This optimization phase is conducted by working with the simulation model that includes the fabrication imprecisions. To accomplish this task a solution able of increasing the mechanical properties of the structure uniformly. The only two geometrical parameters allowing this modification are L_z and A. As seen in Figure 6.7 the gain of mechanical properties is too low compared to the increase in weight of the structure so the tuning of A is preferred.

It has been noticed that increasing A by 50%, the mechanical properties are pratically restored. The detailed elastic and shear moduli are listed in absolute terms in Table 6.6 at the voice *"Optimized structure"*.

Table 6.6: Mechanical properties differences between designed and fabricated lattice structure with the same starting geometric features

Comb 8	*Output parameters*			
	E_y [MPa]	G_{yx} [MPa]	E_x [MPa]	ρ/ρ_s
Designed structure	655,30	2,12	15,33	0,0171
As-fabricated structure	366,21	1,05	8,45	0,0115
Optimized structure	667,65	2,19	17,57	0,0173

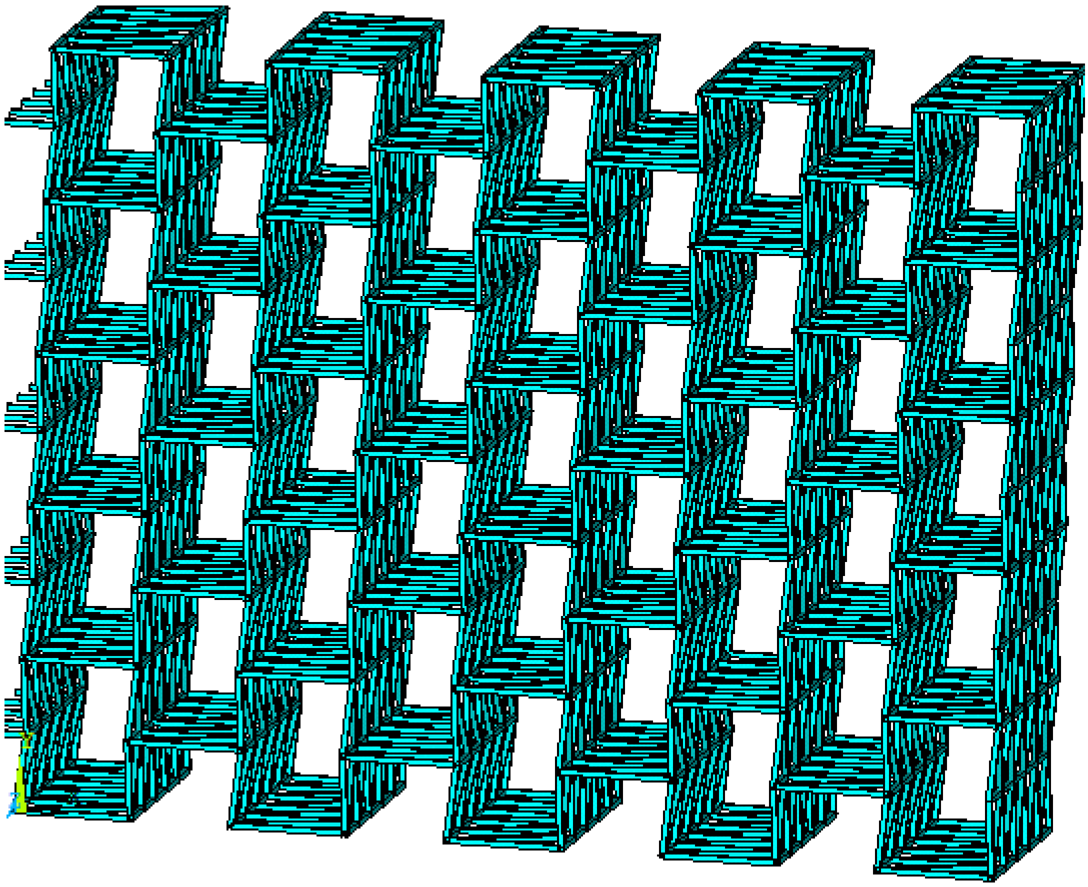


Figure 6.14: Representation of the optimized lattice structure

Chapter 7

Conclusions

This paper aimed to create a simulation model for an accurate prediction of an auxetic lattice structure's mechanical properties to be fabricated through an EBM process.
For this purpose a methodic procedure, such as DOE, has been provided to find out which are the geometrical features affecting most the *in plane* mechanical properties of the overall structure, in terms of elastic and shear moduli. It has been noticed that each geometrical parameter has a different influence on the mechanical properties depending on the analyzed modulus and direction. In particular θ turned out to be the most significant parameter affecting E_y for values ranging from $-10°$ to $10°$, A increased all the *in plane* moduli independently from the direction, while an increase of h has shown a softening effect for all moduli except for E_x.
However, a DOE is performed by tuning also many parameters simultaneously so also the interconnection existing between geometrical parameters has been studied and θ and A have shown the strongest interaction in the influence of E_y.
Furthermore, it has been noticed that, despite the high scan speed that the EBM process provides, it still has many limitations in terms of surface finish when manufacturing thin-struts lattices. This issue produces non-negligible structural variations to the designed structure. For this reason, an overview of the main inaccuracies generated by AM processes, and in particular by EBM, has been done. It has been concluded that dimensional inaccuracies represent the most compromising imperfection from a structural point of view: their magnitude depends on the strut inclination angle respect to the building direction, and on their cross-section size. Thus the lower is the dimensional accuracy of the manufactured strut and the lower is its strength. This effect can more or less reduce the mechanical properties of the overall structure depending on many process parameters. For this purpose, a simulation model of an EBM as-fabricated structure that includes just the qualitative trend of dimensional inaccuracies as a function of the designed strut diameter and the inclination angle.

Running the mechanical tests of the designed structure through the as-fabricated simulation model, it has been possible to highlight the structural discrepancies between a designed lattice and an as-manufactured one. In the specific case of the proposed lattice, it has been detected that the as-manufactured lattice has shown a decrease of the *in plane* moduli of about 45% respect to those of the designed one. So to close this gap, an optimization procedure has been identified: since A presented the most advantageous gain in mechanical properties compared to the increase of relative density, the tuning of this parameter has been preferred to recover the properties of the starting designed structure.
Lastly, the "as-fabricated simulation model" created in this paper is fictitious, thus these values can not be considered reliable if compared to the actual mechanical discrepancies existing between CAD models and actual fabricated structures. Even because this mechanical gap strongly depends on the EBM process parameters. However, if the input parameters of this model are pre-set accordingly to the mechanical characterization of single struts, this method could represent a time and material saving procedure for a correct prediction of mechanical properties of lattices.
Then, the DOE approach has proved to be a success because it allows the designer to clearly understand which mechanical effects can be produced by varying the multitude of geometrical parameters, as well as achieving the starting objectives, the increase of E_y and the decrease of ρ/ρ_s.

Appendix A

2D honeycomb code

2D_cellular_solid.m

```
clc
clear all
close all
%
%Data for the material
nu=0.27;
E=205000;                         %MPa
rho=8.81e-9;                      %t/mm^3
%Geometrical features of the squared cross-section strut
b=1;                  %[mm]        side of the square
A=b^2;               %[mm^2]      cross-section area
J=b^4/12;            %[mm^4]      area moment of inertia
h=20;                %[mm]        length of horiz struts
L=20;                %[mm]        length of inclined struts
nod=20;              %(nod)^-1 indicates the length of
                     %thickened with respect to L
teta=(pi*(-5)/180); %rad     inclination angle respect
                     %the vertical line
%Geometrical features of the thickened node
v=2;                 %multiplier constant defining the
                     %thickness of node respect to the main
                     %strut
b5=v*b;
A5=b5^2;
J5=b^4/12;
%longitudinal spring constants
Kx=1000000000;      %N/mm
Ky=1000000000;      %N/mm
%torsional spring constant
Kt=1000000000;  %[N mm/rad]
```

```
%
%Structure geometry design
prompt = 'What is the desired number of cells to be lined up? ';
n_cell = input(prompt)
map_leng=n_cell*(2*h+2*L*sin(teta));    %[mm]
%Mapping all geometry structure
x1=[0:(2*h+2*L*sin(teta)):map_leng-h];
x2=[h:(2*h+2*L*sin(teta)):map_leng];
x_1((1:length(x1))*2-1)=x1;
x_1((1:length(x2))*2)=x2;
x_11=0;
k=1;
for i=1:length(x_1)
    if i<length(x_1) && round(abs(x_1(i)-x_1(i+1)))==h
        %
        x_11(k)=x_1(i);
        k=k+1;
        x_11(k)=x_1(i)+L/8;
        k=k+1;
        x_11(k)=x_1(i)+L*7/8;
        k=k+1;
    else
        x_11(k)=x_1(i);
        k=k+1;
    end
end
%
x1=[-(L/8*sin(teta)):(2*h+2*L*sin(teta)):map_leng-h];
x2=[h+(L/8*sin(teta)):(2*h+2*L*sin(teta)):map_leng];
x_2((1:length(x1))*2-1)=x1;
x_2((1:length(x2))*2)=x2;
%
x1=[-L*sin(teta)+(L/8*sin(teta)):(2*h+2*L*sin(teta)):map_leng-L*sin(teta)-h];
x2=[h+L*sin(teta)-(L/8*sin(teta)):(2*h+2*L*sin(teta)):map_leng];
x_3((1:length(x1))*2-1)=x1;
x_3((1:length(x2))*2)=x2;
%
x1=[-L*sin(teta):(2*h+2*L*sin(teta)):map_leng-L*sin(teta)-h];
x2=[h+L*sin(teta):(2*h+2*L*sin(teta)):map_leng];
x_4((1:length(x1))*2-1)=x1;
x_4((1:length(x2))*2)=x2;
%
x_44=0;
k=1;
for i=1:length(x_4)
    if i<length(x_4) && teta≠0 && round(abs(x_4(i)-x_4(i+1)))==h
        %
        x_44(k)=x_4(i);
        k=k+1;
```

```
            x_44(k)=x_4(i)+L/8;
            k=k+1;
            x_44(k)=x_4(i)+L*7/8;
            k=k+1;
        else
            x_44(k)=x_4(i);
            k=k+1;
        end
    end
    %
    x_44=[-L*sin(teta)-h/2,-L*sin(teta)-L/8,x_44];
    %
    N = length(x_1);
    x=[x_11,x_2,x_3,x_44,x_3,x_2];
    X=repmat(x,[1 N/2]);
    X=[X,x_11];
    %
    y1=[0:2*L*cos(teta):(n_cell*2*L*cos(teta))];
    y2=[L*cos(teta)/8:2*L*cos(teta):(n_cell*2*L*cos(teta))];
    y3=[L*cos(teta)*7/8:2*L*cos(teta):(n_cell*2*L*cos(teta))];
    y4=[L*cos(teta):2*L*cos(teta):(n_cell*2*L*cos(teta))];
    y5=[L*cos(teta)+L*cos(teta)/8:2*L*cos(teta):(n_cell*2*L*cos(teta))];
    y6=[L*cos(teta)+L*cos(teta)*7/8:2*L*cos(teta):(n_cell*2*L*cos(teta))];
    %
    y((1:length(y1))*6-5)=y1;
    y((1:length(y2))*6-4)=y2;
    y((1:length(y2))*6-3)=y3;
    y((1:length(y2))*6-2)=y4;
    y((1:length(y2))*6-1)=y5;
    y((1:length(y2))*6)=y6;
    % Generation of vector Y
    cont=1;
    for i=1:length(y)
        %
        if i==1
            %
            for j=1:2*N
                Y(j)=y(i);
            end
        end
        %
        %
        if i>1 && rem(y(i),L*cos(teta))==0
            %
             dummY=0;
             for j=1:(2*N)
                 dummY(j)=y(i);
             end
             Y=[Y,dummY];
```

```

    elseif i>1 && rem(y(i),L*cos(teta))≠ 0
        dummY=0;
        for j=1:N
            dummY(j)=y(i);
        end
        Y=[Y,dummY];
    end
end
%% Printing Ansys APDL code
fid = fopen('Structure.txt','w');
%
fprintf(fid,'/CLEAR \n');
fprintf(fid,'/PREP7 \n');
%Elements type definition
fprintf(fid,'ET,1,BEAM3\n');            %main struts
fprintf(fid,'ET,5,BEAM3\n');            %thickened nodes
fprintf(fid,'ET,2,COMBIN14,,6\n');      %Kt torsional spring
fprintf(fid,'ET,3,COMBIN14,,1\n');      %Kx longitudinal spring
fprintf(fid,'ET,4,COMBIN14,,2\n');      %Ky longitudinal spring
%
%Real constant definition
fprintf(fid,'R,1,%f,%f,%f\n',A,J,b);
fprintf(fid,'R,2,%f\n',Kt);
fprintf(fid,'R,3,%f\n',Kx);
fprintf(fid,'R,4,%f\n',Ky);
fprintf(fid,'R,5,%f,%f,%f\n',A5,J5,b5);
%
%Material Properties definition
fprintf(fid,'mp,ex,1,%d\n',E);
fprintf(fid,'mp,nuxy,1,%f\n',nu);
fprintf(fid,'mp,dens,1,%f\n',rho);
%
%NODES DEFINITION
%LAYER 1
for i=1:length(X)
   %
   string=['N,' num2str(i) ',' num2str(X(i)), ',' ...
    num2str(Y(i)) ',0 \n'];
   %
   fprintf(fid,string);
   %
end
%LAYER 2
for i=1:length(X)
   %
   string=['N,' num2str(i+length(X)) ',' num2str(X(i)), ',' ...
    num2str(Y(i)) ',0 \n'];
   %
```

```
    fprintf(fid,string);
    %
end
%LAYER 3
for i=1:length(X)
    %
    string=['N,' num2str(i+2*length(X)) ',' num2str(X(i)), ',' ...
     num2str(Y(i)) ',0 \n'];
    %
    fprintf(fid,string);
    %
end
%%
%ELEMENTS DEFINITION
%Struts
%LAYER 1
for i=2:length(X)
     %
     d=round(X(i)-X(i-1),3);
     if d == round(h-2/8*L,3)
         fprintf(fid,'TYPE,1 \n');
         fprintf(fid,'REAL,1 \n');
         string=['E,' num2str(i-1) ',' num2str(i) '\n'];
         fprintf(fid,string);
     end
     %
     if d == round(h/2-L/8,3)
         fprintf(fid,'TYPE,1 \n');
         fprintf(fid,'REAL,1 \n');
         string=['E,' num2str(i-1) ',' num2str(i) '\n'];
         fprintf(fid,string);
     end
     if d==round(L/8,3)
         fprintf(fid,'TYPE,5 \n');
         fprintf(fid,'REAL,5 \n');
         string=['E,' num2str(i-1) ',' num2str(i) '\n'];
         fprintf(fid,string);
     end
end
%%
%LAYER 2
for i=1:length(X)-2*N
     %
       for k=i:(i+2*N)
        %
        if round(X(k)-X(i),3) == round(L/8*sin(teta),3)
            %
            fprintf(fid,'TYPE,5 \n');
            fprintf(fid,'REAL,5 \n');
```

```
            string=['E,' num2str(i+length(X)) ',' ...
    num2str(k+length(X)) '\n'];
            fprintf(fid,string);
        elseif round(X(k)-X(i),3) == round((L-2*L/8)*sin(teta),3)
                %
                fprintf(fid,'TYPE,1 \n');
                fprintf(fid,'REAL,1 \n');
                string=['E,' num2str(i+length(X)) ',' ...
    num2str(k+length(X)) '\n'];
                fprintf(fid,string);
        end
       end
end
%
%LAYER 3
for i=1:length(X)-2*N
    %
      for k=i:(i+2*N)
       %
        if round(X(k)-X(i),3) == -round(L/8*sin(teta),3)
           %
            fprintf(fid,'TYPE,5 \n');
            fprintf(fid,'REAL,5 \n');
            string=['E,' num2str(i+2*length(X)) ',' ...
    num2str(k+2*length(X)) '\n'];
            fprintf(fid,string);
        elseif round(X(k)-X(i),3) == -round((L-2*L/8)*sin(teta),3)
                %
                fprintf(fid,'TYPE,1 \n');
                fprintf(fid,'REAL,1 \n');
                string=['E,' num2str(i+2*length(X)) ',' ...
    num2str(k+2*length(X)) '\n'];
                fprintf(fid,string);
        end
       end
end
%Spring elements
fprintf(fid,'TYPE,2 \n');
fprintf(fid,'REAL,2 \n');
for i=4:length(X)
    %
    if X(i)≠(-L*sin(teta)-h/2) && ...
    (round(X(i)-X(i-3),3)==round(h,3) || ...
    round(X(i+1)-X(i),3)==round(2*L*sin(teta)+h,3))
        %
        string=['E,' num2str(i) ',' num2str(i+length(X)) '\n'];
        fprintf(fid,string);
        string=['E,' num2str(i+2*length(X)) ',' ...
    num2str(i+length(X)) '\n'];
```

```
            fprintf(fid,string);
            string=['E,' num2str(i) ',' num2str(i+2*length(X)) '\n'];
            fprintf(fid,string);
            %
        end
        %
end
%
for i=1:length(X)-3
        %
        if X(i)≠(-L*sin(teta)-h/2) && ...
       (round(X(i+3)-X(i),3)==round(h,3) || ...
       round(X(i)-X(i-1),3)==round(2*L*sin(teta)+h,3))
            %
            string=['E,' num2str(i) ',' num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i+2*length(X)) ',' ...
       num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i) ',' num2str(i+2*length(X)) '\n'];
            fprintf(fid,string);
            %
        end
        %
end
%
fprintf(fid,'TYPE,3 \n');
fprintf(fid,'REAL,3 \n');
for i=1:length(X)
        %
        if X(i)≠ (-L*sin(teta)-h/2)
            %
            string=['E,' num2str(i) ',' num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i+2*length(X)) ',' ...
       num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i) ',' num2str(i+2*length(X)) '\n'];
            fprintf(fid,string);
            %
        end
        %
end
fprintf(fid,'TYPE,4 \n');
fprintf(fid,'REAL,4 \n');
for i=1:length(X)
        %
        if X(i)≠ (-L*sin(teta)-h/2)
            %
```

```
            string=['E,' num2str(i) ',' num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i+2*length(X)) ',' ...
        num2str(i+length(X)) '\n'];
            fprintf(fid,string);
            string=['E,' num2str(i) ',' num2str(i+2*length(X)) '\n'];
            fprintf(fid,string);
            %
        end
        %
    end
```

Appendix B

3D designed lattice code

designed_lattice.m

```
clear all
clc
%Data for the material
nu=0.27;
E=205000;%MPa
G=E/(2*(1+nu));
rho=8.81e-9;                       %t/mm^3
%Geometrical features of the squared cross-section strut
b0=1.4142;              %[mm] side of the square
H0=1.4142;              %[mm] side of the square
A0=b0*H0;               %[mm^2] cross-section area
Jz0=b0*(H0^3)/12;       %[mm^4] area moment of inertia
Jy0=(H0)*(b0^3)/12;     %[mm^4] area moment of inertia
b=b0;
H=H0;
A=A0;
Jz=Jz0;
Jy=Jy0;
h=30;                   %[mm] length of horiz struts
L0=20;                  %[mm] length of inclined struts
L=L0;
Lz=20;
teta=(pi*(-5)/180);     %[rad] inclination angle respect
                        %the vertical line
%Definition of the aspect ratio
asp=h/L;
gradasp=0;              %aspect ratio gradient variable
slend0=L0/b0;
slend=slend0;
gradA=0;
```

```
%struts angle gradient
gradalpha=0;
%Structure geometry design
prompt = 'How many cells would you like to have in X ...
    direction? ';
n_cell_x = input(prompt);
prompt = 'How many cells would you like to have in Y ...
    direction?  ';
n_cell_y = input(prompt);
prompt = 'How many cells would you like to have in Z ...
    direction?  ';
n_cell_z = input(prompt);
%
Z=(0:n_cell_z);
Z=-Z*Lz;
%Relative densities
%Volume of the lattice
Vl=A/4*(4*h+6*Lz+8*L)+(4*((L^2)+(Lz^2))^0.5+2*(Lz^2+h^2)^0.5)*A/2;
%Volume of the honeycomb
Vh=(4*L+2*h)*(H0/2*Lz);
%Volume of the solid
Vs=((2*L*cos(teta)*h)+2*(cos(teta)*sin(teta))*L^2)*Lz;
%Honeycomb relative density
rdh=Vh/Vs;
%Lattice relative density
rdl=Vl/Vs;
%
h_step=(2*h+2*L*sin(teta));
v_step=(2*L*cos(teta));
if (teta)>0
map_leng_x=n_cell_x*h_step;                    %mm
else
map_leng_x=n_cell_x*h_step+L*sin(teta);
end
map_leng_y=n_cell_y*v_step;                      %mm
%
%Mapping all geometry structure
X=0;
for k=1:n_cell_y
    x1=[0:h_step:map_leng_x-(h)];
    x2=[h:h_step:map_leng_x];
    x_1((1:length(x1))*2-1)=x1;
    x_1((1:length(x2))*2)=x2;
    if k==1
        deteta=teta;
    else
        deteta=deteta+(pi*(gradalpha)/180);
    end
    %
```

```
    for i=1:length(x1)
    x1(i)=x1(i)-L*sin(deteta);
    end
    %
    for i=1:length(x2)
        x2(i)=x2(i)+L*sin(deteta);
    end
    %
    x_2=0;
    x_2((1:length(x1))*2-1)=x1;
    x_2((1:length(x2))*2)=x2;
    x_2=[-L*sin(teta)-h/2,x_2];
    x=[x_1,x_2];
    X=[X,x];
    %
end
X=X(2:end);
X=[X,x_1];
y(1)=0;
for i=2:(n_cell_y*2+1)

    y(i)=y(i-1)+L*cos(teta);
    %
    if rem(i,2)≠0 && i≠1
        asp=asp+gradasp*asp/100;
        L=h/asp;
    end

end
%
N = length(x_1);
% Generation of Y vector
for i=1:length(y)
    %
    if i==1
        %
        for j=1:N
            Y(j)=y(i);
        end
    end
    %
    if rem(i,2)==1 && i>1
        dummY=0;
        for j=1:N
            dummY(j)=y(i);
        end
        Y=[Y,dummY];
    end
    %
```

```
    if rem(i,2)==0
            dummY=0;
        for j=1:N+1
            dummY(j)=y(i);
        end
        Y=[Y,dummY];
    end
end
%%
%Printing the Ansys APDL code
fid = fopen('Structure.txt','w');
%
fprintf(fid,'/CLEAR \n');
fprintf(fid,'/PREP7 \n');
%Elements type definition
fprintf(fid,'ET,1,BEAM4\n');
%%Real constant definition
fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
%Material Properties definition
fprintf(fid,'mp,ex,1,%d\n',E);
fprintf(fid,'mp,nuxy,1,%f\n',nu);
%
%NODES DEFINITION
for k=1:length(Z)
for i=1:length(X)
   %
   string=['N,' num2str(i+(length(X)*(k-1))) ',' num2str(X(i)) ...
    ',' num2str(Y(i)) ',' num2str(Z(k)) ', \n'];
   fprintf(fid,string);
   %
end
end
%%
%ELEMENTS DEFINITION - beams
%Beams
%LAYER 1 horizontal struts
fprintf(fid,'TYPE,1 \n');
fprintf(fid,'REAL,1 \n');
cont=0;
for i=1:length(X)
    %
    cont=cont+1;
    if cont<length(x) && rem(cont,2)≠0
        for k=1:length(Z)
            string=['E,' num2str(i+(k-1)*length(X)) ',' ...
    num2str((i+(k-1)*length(X)+1)) '\n'];
            fprintf(fid,string);
        end
    end
```

```
    %
    if cont==length(x)
        A=A+gradA/100;
        b=sqrt(A);
        H=sqrt(A);
        Jz=b*(H^3)/12;
        Jy=(H)*(b^3)/12;
        fprintf(fid,'R,1,%f,%f,%f %f %f\n',A,Jz,Jy,b,b);
        fprintf(fid,'REAL,1 \n');
        cont=0;
    end
end
%LAYER 2 left-sided inclined struts
for r=1:length(Z)
    L=L0;
    b=b0;
    H=H0;
    A=b*H;
    Jz=b*(H^3)/12;
    Jy=(H)*(b^3)/12;
    fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    q=0;
    for i=1:length(y)-1
        if rem(i,2)==1
            for k=1:2:length(x_1)-1
                string=['E,' ...
      num2str((i-1)/2*length(x)+k+((r-1)*length(X))) ',' ...
      num2str((i-1)/2*length(x)+k+length(x_2)+((r-1)*length(X))) ...
      '\n'];
                fprintf(fid,string);
            end
        end
    %
        if rem(i,2)==0
            if i==2
                for k=3:2:length(x_2)
                    string=['E,' ...
      num2str(length(x_1)+k+((r-1)*length(X))) ',' ...
      num2str(length(x_1)+k+length(x_1)+((r-1)*length(X))) '\n'];
                    fprintf(fid,string);
                end
            else
                for k=3:2:length(x_2)
                    string=['E,' ...
      num2str(((i)/2*length(x)-length(x_2))+k+((r-1)*length(X))) ...
      ',' ...
      num2str(((i)/2*length(x)-length(x_2))+k+length(x_1)+((r-1)*length(X))) ...
      '\n'];
```

```
                        fprintf(fid,string);
                    end
                end
            end
        %
        q=q+1;
        if q==2 && i≠length(y)-1
            %
            A=A+gradA/100;
            L=(y(i+2)-y(i+1))/cos(teta);
            b=sqrt(A);
            H=sqrt(A);
            A=b*H;
            Jz=b*(H^3)/12;
            Jy=(H)*(b^3)/12;
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            q=0;
        end
    end
    end
    %LAYER 3 right-sided inclined struts
    for r=1:length(Z)
    L=L0;
    b=b0;                                  %mm
    H=H0;                                  %mm
    slend=slend0;
    A=b*H;                                 %mm^2
    Jz=b*(H^3)/12;                         %mm^4
    Jy=(H)*(b^3)/12;                         %mm^4
    fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    q=0;
    %
    for i=1:length(y)-1
        if rem(i,2)==1
            for k=2:2:length(x_1)
                string=['E,' ...
       num2str((i-1)/2*length(x)+k+((r-1)*length(X))) ',' ...
       num2str((i-1)/2*length(x)+k+length(x_2)+((r-1)*length(X))) ...
       '\n'];
                fprintf(fid,string);
            end
        end
        %
        if rem(i,2)==0
            if i==2
                for k=2:2:length(x_2)
```

```
            string=['E,' ...
    num2str(length(x_1)+k+((r-1)*length(X))) ',' ...
    num2str(length(x_1)+k+length(x_1)+((r-1)*length(X))) '\n'];
            fprintf(fid,string);
            end
        else
        for k=2:2:length(x_2)
            string=['E,' ...
    num2str(((i)/2*length(x)-length(x_2))+k+((r-1)*length(X))) ...
    ',' ...
    num2str(((i)/2*length(x)-length(x_2))+k+length(x_1)+((r-1)*length(X))) ...
    '\n'];
            fprintf(fid,string);
        end
        end
    end
    q=q+1;
            if q==2 && i≠length(y)-1
               %
               A=A+gradA/100;
               L=(y(i+2)-y(i+1))/cos(teta);
               b=sqrt(A);
               H=sqrt(A);
               A=b*H;
               Jz=b*(H^3)/12;
               Jy=(H)*(b^3)/12;
               fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
               fprintf(fid,'REAL,1 \n');
               %
            q=0;
            end
end
end
%horizontal struts along with z-direction
L=L0;
b=b0;
H=H0;
A=b*H;
Jz=b*(H^3)/12;
Jy=(H)*(b^3)/12;
fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
fprintf(fid,'REAL,1 \n');
fprintf(fid,'TYPE,1 \n');
cont=0;
for i=1:length(X)
    %
    cont=cont+1;
    if cont≤length(x) && i≤length(X) && X(i)≠x_2(1)
        for k=1:length(Z)-1
```

```
                string=['E,' num2str(i+(k-1)*length(X)) ',' ...
        num2str((i+k*length(X))) '\n'];
                fprintf(fid,string);
            end
        end
        %
        if cont==length(x)
            A=A+gradA/100;
            b=sqrt(A);
            H=sqrt(A);
            A=b*H;
            Jz=b*(H^3)/12;                  %mm^4
            Jy=(H)*(b^3)/12;                  %mm^4
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            cont=0;
        end
    end
    %Inclined struts along with z direction
    b=b0;
    H=H0;
    A=b*H;
    Jz=b*(H^3)/12;
    Jy=(H)*(b^3)/12;
    fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    fprintf(fid,'TYPE,1 \n');
    cont=0;
    for i=1:length(X)
        %
        cont=cont+1;
        if cont≤length(x_1) && i≤length(X)-N
            for k=1:length(Z)-1
                string=['E,' num2str(i+(k-1)*length(X)) ',' ...
        num2str((i+k*length(X))+N+1) '\n'];
                fprintf(fid,string);
            end
        end
        %
        if cont>length(x_1) && i≤length(X)-N && X(i)≠x_2(1)
            for k=1:length(Z)-1
            string=['E,' num2str(i+(k)*length(X)) ',' ...
        num2str((i+(k-1)*length(X))+(N)) '\n'];
            fprintf(fid,string);
            end
        end
        if cont==length(x)
            A=A+gradA/100;
```

```
            b=sqrt(A);
            H=sqrt(A);
            Jz=b*(H^3)/12;
            Jy=(H)*(b^3)/12;
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            cont=0;
        end
    end
    %Diagonal horizontal struts
    b=b0;                                 %mm
    H=H0;                                 %mm
    A=b*H;                               %mm^2
    Jz=b*(H^3)/12;                       %mm^4
    Jy=(H)*(b^3)/12;                       %mm^4
    fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    fprintf(fid,'TYPE,1 \n');
    cont=0;
    for i=1:length(X)
        %
        cont=cont+1;
        if cont≤length(x_1) && i≤length(X) && rem(cont,2)≠0
            for k=1:length(Z)-1
                string=['E,' num2str(i+(k-1)*length(X)) ',' ...
       num2str((i+k*length(X))+1) '\n'];
                fprintf(fid,string);
            end
        end
        %
        if cont>length(x_1) && cont<length(x) && X(i)≠x_2(1) && ...
       rem(cont,2)≠0
            for k=1:length(Z)-1
                string=['E,' num2str(i+(k-1)*length(X)) ',' ...
       num2str((i+k*length(X))+1) '\n'];
                fprintf(fid,string);
            end
        end
        %
            if cont==length(x)
            A=A+gradA/100;
            b=sqrt(A);
            H=sqrt(A);
            A=b*H;
            Jz=b*(H^3)/12;                     %mm^4
            Jy=(H)*(b^3)/12;                     %mm^4
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
```

```
            %
            cont=0;
            end
end
```

Appendix C

3D as-fabricated lattice code

as_fabricated_lattice.m

```
clear all
clc
%Data for the material
nu=0.27;
E=205000;                         %[MPa]
G=E/(2*(1+nu));
rho=8.81e-9;                      %[t/mm^3]
%Geometrical features of the squared cross-section strut
b0=1.7321;                    %[mm] side of the square
H0=1.7321;                    %[mm] side of the square
A0=b0*H0;                     %[mm^2] cross-section area
Jz0=b0*(H0^3)/12;             %[mm^4] area moment of inertia
Jy0=(H0)*(b0^3)/12;           %[mm^4] area moment of inertia
b=b0;
H=H0;
A=A0;
Jz=Jz0;
Jy=Jy0;
h=30;                         %[mm] length of horiz struts
L0=20;                        %[mm] length of inclined struts
L=L0;
Lz=20;
angle=-5;                     %[  ] inclination angle respect
                              %the vertical line
teta=(pi*(angle)/180);        %[rad]
%Definition of the aspect ratio
asp=h/L;
gradasp=0;
slend0=L0/b0;
slend=slend0;
```

```
gradA=0;
gradalpha=0;
Vf=0;
%Structure geometry design
prompt = 'How many cells would you like to have in X ...
    direction? ';
n_cell_x = input(prompt);
prompt = 'How many cells would you like to have in Y ...
    direction?  ';
n_cell_y = input(prompt);
prompt = 'How many cells would you like to have in Z ...
    direction?  ';
n_cell_z = input(prompt);
%
Z=(0:n_cell_z);
Z=-Z*Lz;
%Relative densities
Vl=A/4*(4*h+6*Lz+8*L)+(4*((L^2)+(Lz^2))^0.5+2*(Lz^2+h^2)^0.5)*A/2;
Vh=(4*L+2*h)*(H0/2*Lz);
Vs=((2*L*cos(teta)*h)+2*(cos(teta)*sin(teta))*L^2)*Lz;
rdh=Vh/Vs;
rdl=Vl/Vs;
h_step=(2*h+2*L*sin(teta));
v_step=(2*L*cos(teta));
if (teta)>0
map_leng_x=n_cell_x*h_step;
else
map_leng_x=n_cell_x*h_step+L*sin(teta);
end
map_leng_y=n_cell_y*v_step;
%Mapping all geometry structure
X=0;
for k=1:n_cell_y
    %first row
    x1=[0:h_step:map_leng_x-(h)];
    x2=[h:h_step:map_leng_x];
    x_1((1:length(x1))*2-1)=x1;
    x_1((1:length(x2))*2)=x2;
    %second row
    if k==1
        deteta=teta;
    else
        deteta=deteta+(pi*(gradalpha)/180);
    end
    %
    for i=1:length(x1)
    x1(i)=x1(i)-L*sin(deteta);
    end
    %
```

```
    for i=1:length(x2)
        x2(i)=x2(i)+L*sin(deteta);
    end
    %
    x_2=0;
    x_2((1:length(x1))*2-1)=x1;
    x_2((1:length(x2))*2)=x2;
    x_2=[-L*sin(teta)-h/2,x_2];
    x=[x_1,x_2];
    X=[X,x];
    %
end
X=X(2:end);
X=[X,x_1];
y(1)=0;
for i=2:(n_cell_y*2+1)
    y(i)=y(i-1)+L*cos(teta);
    %
    if rem(i,2)≠0 && i≠1
        asp=asp+gradasp*asp/100;
        L=h/asp;
    end
end
%
N = length(x_1);
% Generate vector Y
for i=1:length(y)
    %
    if i==1
        %
        for j=1:N
            Y(j)=y(i);
        end
    end
    %
    if rem(i,2)==1 && i>1
        dummY=0;
        for j=1:N
            dummY(j)=y(i);
        end
        Y=[Y,dummY];
    end
    %
    if rem(i,2)==0
            dummY=0;
        for j=1:N+1
            dummY(j)=y(i);
        end
        Y=[Y,dummY];
```

```
    end
end
%%
fid = fopen('Structure.txt','w');
fprintf(fid,'/CLEAR \n');
fprintf(fid,'/PREP7 \n');
%Elements type definition
fprintf(fid,'ET,1,BEAM4\n');
%Real constant definition
fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
%Material Properties definition
fprintf(fid,'mp,ex,1,%d\n',E);
fprintf(fid,'mp,nuxy,1,%f\n',nu);
%NODES DEFINITION
for k=1:length(Z)
for i=1:length(X)
   %
   string=['N,' num2str(i+(length(X)*(k-1))) ',' num2str(X(i)) ...
   ',' num2str(Y(i)) ',' num2str(Z(k)) ', \n'];
   fprintf(fid,string);
   %
end
end
%%
%%Printing the Ansys APDL code
%LAYER 1-horizontal struts
fprintf(fid,'TYPE,1 \n');
fprintf(fid,'REAL,1 \n');
cont=0;
[A,Jz,Jy]=horiz(b0,H0);
for i=1:length(X)
    %
    cont=cont+1;
    if cont<length(x) && rem(cont,2)≠0
        for k=1:length(Z)
            string=['E,' num2str(i+(k-1)*length(X)) ',' ...
    num2str((i+(k-1)*length(X)+1)) '\n'];
            fprintf(fid,string);
        end
    end
    %
    if cont==length(x)
        A=A+gradA/100;
        b=sqrt(A);
        H=sqrt(A);
        Jz=b*(H^3)/12;
        Jy=(H)*(b^3)/12;
        fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
        fprintf(fid,'REAL,1 \n');
```

```
            cont=0;
        end
    end
    Vf=Vf+A*h;
    %LAYER 2 left-sided inclined struts
    for r=1:length(Z)
        L=L0;
        [A,Jz,Jy]=resetparameters(b0,H0);
        [A,Jz,Jy]=parametri(angle,b0,H0);
        b=sqrt(A);
        fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
        fprintf(fid,'REAL,1 \n');
        q=0;
        for i=1:length(y)-1
            if rem(i,2)==1
                for k=1:2:length(x_1)-1
                    string=['E,' ...
    num2str((i-1)/2*length(x)+k+((r-1)*length(X))) ',' ...
    num2str((i-1)/2*length(x)+k+length(x_2)+((r-1)*length(X))) ...
    '\n'];
                    fprintf(fid,string);
                end
            end
        %
            if rem(i,2)==0
                if i==2
                    for k=3:2:length(x_2)
                        string=['E,' ...
    num2str(length(x_1)+k+((r-1)*length(X))) ',' ...
    num2str(length(x_1)+k+length(x_1)+((r-1)*length(X))) '\n'];
                        fprintf(fid,string);
                    end
                else
                    for k=3:2:length(x_2)
                        string=['E,' ...
    num2str(((i)/2*length(x)-length(x_2))+k+((r-1)*length(X))) ...
    ',' ...
    num2str(((i)/2*length(x)-length(x_2))+k+length(x_1)+((r-1)*length(X))) ...
    '\n'];
                        fprintf(fid,string);
                    end
                end
            end
        %
        q=q+1;
        if q==2 && i≠length(y)-1
            %
            A=A+gradA/100;
            L=(y(i+2)-y(i+1))/cos(teta);
```

```
            b=sqrt(A);
            H=sqrt(A);
            A=b*H;
            Jz=b*(H^3)/12;                          %mm^4
            Jy=(H)*(b^3)/12;
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            q=0;
        end
    end
    end
    %
    Vf=Vf+A*L;
    %LAYER 3 right-sided inclined struts
    for r=1:length(Z)
        L=L0;
        [A,Jz,Jy]=resetparameters(b0,H0);
        [A,Jz,Jy]=parametri(angle,b0,H0);
        b=sqrt(A);
        fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    q=0;
    %
    for i=1:length(y)-1
        if rem(i,2)==1
            for k=2:2:length(x_1)
                string=['E,' ...
       num2str((i-1)/2*length(x)+k+((r-1)*length(X))) ',' ...
       num2str((i-1)/2*length(x)+k+length(x_2)+((r-1)*length(X))) ...
       '\n'];
                fprintf(fid,string);
            end
        end
        %
        if rem(i,2)==0
            if i==2
                for k=2:2:length(x_2)
                string=['E,' ...
       num2str(length(x_1)+k+((r-1)*length(X))) ',' ...
       num2str(length(x_1)+k+length(x_1)+((r-1)*length(X))) '\n'];
                fprintf(fid,string);
                end
            else
            for k=2:2:length(x_2)
```

```
                string=['E,' ...
      num2str(((i)/2*length(x)-length(x_2))+k+((r-1)*length(X))) ...
      ',' ...
      num2str(((i)/2*length(x)-length(x_2))+k+length(x_1)+((r-1)*length(X))) ...
      '\n'];
                fprintf(fid,string);
            end
        end
    end
    q=q+1;
                if q==2 && i≠length(y)-1
                    %
                   A=A+gradA/100;
                   L=(y(i+2)-y(i+1))/cos(teta);
                   b=sqrt(A);
                   H=sqrt(A);
                   A=b*H;
                   Jz=b*(H^3)/12;
                   Jy=(H)*(b^3)/12;
                   fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
                   fprintf(fid,'REAL,1 \n');
                   %
                q=0;
                end
end
end
Vf=Vf+A*L;
%horizontal struts along with z-direction
L=L0;
[A,Jz,Jy]=resetparameters(b0,H0);
[A,Jz,Jy]= horiz(b0,H0);
fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
fprintf(fid,'REAL,1 \n');
fprintf(fid,'TYPE,1 \n');
cont=0;
for i=1:length(X)
    %
    cont=cont+1;
    if cont≤length(x) && i≤length(X) && X(i)≠x_2(1)
        for k=1:length(Z)-1
            string=['E,' num2str(i+(k-1)*length(X)) ',' ...
      num2str((i+k*length(X))) '\n'];
            fprintf(fid,string);
        end
    end
    %
    if cont==length(x)
        A=A+gradA/100;
        b=sqrt(A);
```

```
            H=sqrt(A);
            A=b*H;
            Jz=b*(H^3)/12;
            Jy=(H)*(b^3)/12;
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            cont=0;
        end
    end
    Vf=Vf+6*Lz*A/4;
    %%Inclined struts along with z direction
    [A,Jz,Jy]=resetparameters(b0,H0);
    [A,Jz,Jy]=inclined(teta,L,Lz,b0,H0);
    b=sqrt(A);
    fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
    fprintf(fid,'REAL,1 \n');
    fprintf(fid,'TYPE,1 \n');
    cont=0;
    for i=1:length(X)
        %
        cont=cont+1;
        if cont≤length(x_1) && i≤length(X)-N
            for k=1:length(Z)-1
                string=['E,' num2str(i+(k-1)*length(X)) ',' ...
        num2str((i+k*length(X))+N+1) '\n'];
                fprintf(fid,string);
            end
        end
        %
        if cont>length(x_1) && i≤length(X)-N && X(i)≠x_2(1)
            for k=1:length(Z)-1
            string=['E,' num2str(i+(k)*length(X)) ',' ...
        num2str((i+(k-1)*length(X))+(N)) '\n'];
            fprintf(fid,string);
            end
        end
        if cont==length(x)
            A=A+gradA/100;
            b=sqrt(A);
            H=sqrt(A);
            Jz=b*(H^3)/12;
            Jy=(H)*(b^3)/12;
            fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
            fprintf(fid,'REAL,1 \n');
            %
            cont=0;
        end
    end
```

```
Vf=Vf+4*sqrt((L^2)+(Lz)^2)*A/2;
%Diagonal horizontal struts
[A,Jz,Jy]=resetparameters(b0,H0);
[A,Jz,Jy]= horiz(b0,H0);
fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
fprintf(fid,'REAL,1 \n');
fprintf(fid,'TYPE,1 \n');
cont=0;
for i=1:length(X)
    %
    cont=cont+1;
    if cont≤length(x_1) && i≤length(X) && rem(cont,2)≠0
        for k=1:length(Z)-1
            string=['E,' num2str(i+(k-1)*length(X)) ',' ...
    num2str((i+k*length(X))+1) '\n'];
            fprintf(fid,string);
        end
    end
    %
    if cont>length(x_1) && cont<length(x) && X(i)≠x_2(1) && ...
    rem(cont,2)≠0
        for k=1:length(Z)-1
            string=['E,' num2str(i+(k-1)*length(X)) ',' ...
    num2str((i+k*length(X))+1) '\n'];
            fprintf(fid,string);
        end
    end
    %
        if cont==length(x)
        A=A+gradA/100;
        b=sqrt(A);
        H=sqrt(A);
        A=b*H;
        Jz=b*(H^3)/12;
        Jy=(H)*(b^3)/12;
        fprintf(fid,'R,1,%f,%f,%f,%f,%f\n',A,Jz,Jy,b,b);
        fprintf(fid,'REAL,1 \n');
        %
        cont=0;
        end
end
Vf=Vf+2*sqrt((h^2)+(Lz)^2)*A/2;
rdl=Vf/Vs;
```

horiz.m

```
function [A,Jz,Jy]= horiz(b0,H0)
%function adjusting geometrical features
```

```
%of horizontal as-fabricated struts
if b0≤1
b=0.93*b0;
H=0.93*H0;
elseif b0>1
b=0.85*b0;
H=0.85*H0;
end

A=b*H;                               %mm^2
Jz=b*(H^3)/12;                        %mm^4
Jy=(H)*(b^3)/12;
end
```

parametri.m

```
function [A,Jz,Jy]=parametri(angle,b0,H0)
if b0≤1
if abs(angle)==0
    b=0.68*b0;
    H=0.68*H0;
elseif abs(angle)>0 && abs(angle)≤10
    b=0.72*b0;
    H=0.72*H0;
elseif abs(angle)>10 && abs(angle)≤20
    b=0.75*b0;
    H=0.75*H0;
elseif abs(angle)>20 && abs(angle)≤45
    b=0.78*b0;
    H=0.78*H0;
elseif abs(angle)>45 && abs(angle)≤70
    b=0.84*b0;
    H=0.84*H0;
elseif abs(angle)>70 && abs(angle)≤90
    b=0.93*b0;
    H=0.93*H0;
end
elseif b0>1
if abs(angle)==0
    b=0.75*b0;
    H=0.75*H0;
elseif abs(angle)>0 && abs(angle)≤10
    b=0.76*b0;
    H=0.76*H0;
elseif abs(angle)>10 && abs(angle)≤20
    b=0.77*b0;
    H=0.77*H0;
elseif abs(angle)>20 && abs(angle)≤45
```

```
    b=0.79*b0;
    H=0.79*H0;
elseif abs(angle)>45 && abs(angle)≤70
    b=0.81*b0;
    H=0.81*H0;
elseif abs(angle)>70 && abs(angle)≤90
    b=0.85*b0;
    H=0.85*H0;
end
end
A=b*H;
Jz=b*(H^3)/12;
Jy=(H)*(b^3)/12;
end
```

resetparameters.m

```
function [A,Jz,Jy]=resetparameters(b0,H0)
    b=b0;
    H=H0;
    A=b*H;
    Jz=b*(H^3)/12;
    Jy=(H)*(b^3)/12;
end
```

Appendix D

Mechanical tests

mechanical_tests.m

```
%Mechanical tests to be chosen by the user
%Code working for all the generated structures
prompt = 'What is the test to be performed? Shear (S) or ...
    tensile (T) test? ';
test = input(prompt,'s');
if isempty(test)
    test = 'T';
end
switch test
    case 'T'
        %Constraining the y direction
        fprintf(fid,'NSEL,S,LOC,Y,0 \n');
        fprintf(fid,'D,ALL,UY,0 \n');
        fprintf(fid,'D,ALL,ROTZ,0 \n');
        fprintf(fid,'D,ALL,ROTX,0 \n');
        fprintf(fid,'NSEL,ALL \n');
        %
        %Constraining the x direction
        fprintf(fid,'NSEL,S,LOC,X,%f \n',x(N+1));
        fprintf(fid,'D,ALL,UX,0 \n');
        fprintf(fid,'D,ALL,ROTZ,0 \n');
        fprintf(fid,'D,ALL,ROTY,0 \n');
        fprintf(fid,'NSEL,ALL \n');
        %
        %
        prompt = 'What is the load direction? X/Y \n';
        dir = input(prompt,'s');
        if isempty(test)
            test = 'X';
        end
```

```
        %
        if dir == 'X'
            %Imposing displacement
            if teta≥0
            fprintf(fid,'NSEL,S,LOC,X,%f \n',X(2*N+1));
            fprintf(fid,'D,ALL,UX,-0.1 \n');
            fprintf(fid,'NSEL,ALL \n');
            end
            %
            if teta<0
            fprintf(fid,'NSEL,S,LOC,X,%f \n',X(N));
            fprintf(fid,'D,ALL,UX,-0.1 \n');
            fprintf(fid,'NSEL,ALL \n');
            end
            %
        elseif dir == 'Y'
            %
            fprintf(fid,'NSEL,S,LOC,Y,%f \n',Y(end));
            fprintf(fid,'D,ALL,UY,-0.1 \n');
            fprintf(fid,'NSEL,ALL \n');
            %
        end
    %
    case 'S'
        %
        %
            %Constraining the x plane
            fprintf(fid,'NSEL,S,LOC,Y,0 \n');
            fprintf(fid,'D,ALL,UX,0 \n');
            fprintf(fid,'NSEL,ALL \n');
            %
            %Constraining the y plane
            fprintf(fid,'NSEL,S,LOC,X,%f \n',x(N+1));
            fprintf(fid,'D,ALL,UY,0 \n');
            fprintf(fid,'NSEL,ALL \n');
            %
            %
            %
        prompt = 'What is the load direction? X/Y \n';
        dir = input(prompt,'s');
        if isempty(dir)
            dir = 'X';
        end
        %
        if dir == 'X'
            %
            %Displacement definition
            fprintf(fid,'NSEL,S,LOC,Y,%f \n',Y(end));
            fprintf(fid,'D,ALL,UX,0.1 \n');
```

```
            fprintf(fid,'D,ALL,UY,0 \n');
            fprintf(fid,'NSEL,ALL \n');
%
        elseif dir == 'Y'
            %
            %Displacement definition
            if teta≥0
            fprintf(fid,'NSEL,S,LOC,X,%f \n',X(2*N+1));
            fprintf(fid,'D,ALL,UY,0.1 \n');
            fprintf(fid,'D,ALL,UX,0 \n');
            fprintf(fid,'NSEL,ALL \n');
            end
            %
            if teta<0
            fprintf(fid,'NSEL,S,LOC,X,%f \n',X(N));
            fprintf(fid,'D,ALL,UY,0.1 \n');
            fprintf(fid,'D,ALL,UX,0 \n');
            fprintf(fid,'NSEL,ALL \n');
            end
        end
        %
end
fprintf(fid,'/SOLU \n');
%Solution and post processing
fprintf(fid,'antype, static \n');
fprintf(fid,'SOLVE \n');
fprintf(fid,'/POST1 \n');
fprintf(fid,'PLDISP,1 \n');
```

Bibliography

[1] Lorna J. Gibson and Michael F. Ashby. *Cellular Solids: Structure and Properties*. 2nd ed. Cambridge Solid State Science Series. Cambridge University Press, 1997. DOI: 10.1017/CBO9781139878326 (cit. on pp. 4–6, 32).

[2] V.S. Deshpande, M.F. Ashby, and N.A. Fleck. «Foam topology: bending versus stretching dominated architectures». In: *Acta Materialia* 49.6 (2001), pp. 1035–1040. ISSN: 1359-6454. DOI: https://doi.org/10.1016/S1359-6454(00)00379-7 (cit. on p. 7).

[3] M.F. Ashby. «The properties of foams and lattices». In: *Philosophical Transactions of the Royal Society* 364:15-30 (Nov. 2005). DOI: http://doi.org/10.1098/rsta.2005.1678 (cit. on pp. 8–11).

[4] K.Evans. «Tailoring the negative Poisson ratio». In: *Chemistry and Industry* 20 (2001), pp. 654–657 (cit. on pp. 11, 12).

[5] Mohammad Sanami, Naveen Ravirala, Kim Alderson, and Andrew Alderson. «Auxetic Materials for Sports Applications». In: *Procedia Engineering* 72 (2014). The Engineering of Sport 10, pp. 453–458. ISSN: 1877-7058. DOI: https://doi.org/10.1016/j.proeng.2014.06.079 (cit. on p. 14).

[6] Ian Gibson, David W. Rosen, and Brent Stucker. «Design for Additive Manufacturing». In: *Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing*. Boston, MA: Springer US, 2010, pp. 299–332. ISBN: 978-1-4419-1120-9. DOI: 10.1007/978-1-4419-1120-9_11 (cit. on pp. 15–17, 19–21).

[7] Ifeanyichukwu Echeta, Xiaobing Feng, B. Dutton, Samanta Piano, and Richard Leach. «Review of defects in lattice structures manufactured by powder bed fusion». In: *International Journal of Advanced Manufacturing Technology* 106 (Jan. 2020). DOI: 10.1007/s00170-019-04753-4 (cit. on p. 22).

[8] O. Cansizoglu, O. Harrysson, D. Cormier, H. West, and T. Mahale. «Properties of Ti–6Al–4V non-stochastic lattice structures fabricated via electron beam melting». In: *Materials Science and Engineering: A* 492.1 (2008), pp. 468–474. ISSN: 0921-5093. DOI: https://doi.org/10.1016/j.msea.2008.04.002 (cit. on pp. 22, 23).

[9] Zahra S. Bagheri, David Melancon, Lu Liu, R. Burnett Johnston, and Damiano Pasini. «Compensation strategy to reduce geometry and mechanics mismatches in porous biomaterials built with Selective Laser Melting». In: *Journal of the Mechanical Behavior of Biomedical Materials* 70 (2017), pp. 17–27. ISSN: 1751-6161. DOI: https://doi.org/10.1016/j.jmbbm.2016.04.041 (cit. on p. 23).

[10] Mathieu Suard. «Characterization and optimization of lattice structures made by Electron Beam Melting». PhD thesis. Nov. 2015 (cit. on pp. 22, 23, 47).

[11] Yasin Amani, Sylvain Dancette, Pauline Delroisse, Aude Simar, and Eric Maire. «Compression behavior of lattice structures produced by selective laser melting: X-ray tomography based experimental and finite element approaches». In: *Acta Materialia* 159 (2018), pp. 395–407. ISSN: 1359-6454. DOI: https://doi.org/10.1016/j.actamat.2018.08.030 (cit. on pp. 23, 24).

[12] *[Online] Introduction to Ansys Mechanical APDL*. URL: https://www.ansys.com/services/training-center/structures/introduction-to-ansys-mechanical-apdl (cit. on p. 28).

[13] *Theory Reference for the Mechanical APDL and Mechanical Applications*. Canonsburg, PA, 2009 (cit. on pp. 29–32).

[14] *[Online] Ansys help*. URL: https://ansyshelp.ansys.com (cit. on pp. 30, 31).

[15] Piccagli. «Numerical investigation of gradient 3D auxetic metalattice realized via additive manufacturing». PhD thesis. 2018 (cit. on pp. 59, 61).

[16] Xuehong Lu and Lau Soo Khim. «A statistical experimental study of the injection molding of optical lenses». In: *Journal of Materials Processing Technology* 113.1 (2001), pp. 189–195. ISSN: 0924-0136. DOI: https://doi.org/10.1016/S0924-0136(01)00606-9 (cit. on p. 68).