

Preface

This thesis is the closing part of a 5 years Master's degree program, in Mechanical Engineering at the Department of Mechanical and Industrial Engineering (MTP) at the Norwegian University of Science and Technology (NTNU). The work was conducted in the period of January-July 2019. The main supervisor was Associate Professor Jan Torgersen.

The thesis work counts for 30 ECT credits and is a continuation of a specialization project the previous semester of Fall 2018, counting for 15 ECT. This project, regarding titanium scaffolds for orthopedic implants, has had several master's theses over the course of three years and currently under research by PhD student Kristin Sirnes Ødegaard, who also co-supervised this thesis.

The reader would benefit from a technical background to follow all the technical arguments and explanations, although anyone able of reading a scientific paper should be able to get the main parts.

Trondheim, 2019-07-02

Markus Joakim Lid

Acknowledgment

I would not have ended up where I am today, without the support of others. First of all I would like to thank my supervisors, who have guided me through the last months of work, always available and interested in my project. Especially thanks to my main supervisor, Associate Professor Jan Torgersen, for the many interesting discussions, and ideating sessions we have had. You have inspired me to continue with research, and opened the doors for me to do so. I look forwards to working with you in my PhD.

I am also very thankful to Kristin Sirnes Ødegaard, with whom I have planned and conducted many experiments in the NanoLab. The effort you have laid down in guidance and editorial work these last days has been priceless.

Thank you Christer Westum Elverum for following up my project and constantly keeping me updated on the field of Additive Manufacturing. Thanks to Felippo Berto for counseling when something needs to be broken or published. You know everything about mechanical testing, fracture and fatigue, and I don't know anyone who gets more citations than you.

The Research Council of Norway is acknowledged for the support to the Norwegian Micro- and Nano-Fabrication Facility, NorFab, project number 245963/F50. This has allowed me to use the fantastic facilities at the NanoLab.

Thanks to my family, fiancé and friends for your loving kindness, cheering exclamations, flowers and biscuits. You have believed in me and supported me all the way. Lastly, to the source of my greatest inspirations; the one who gives me life and continually renews my soul; my greatest thanks and praise be to You, my God.

M.J.L

Abstract

Cellular materials are interconnected structures consisting of beams or shells to form a porous volume. By careful design, these complex structures can be made to obtain specific properties, which makes them interesting for bone implant applications, in such cases called scaffolds. Scaffolds can be tuned to have a stiffness which matches that of the patient's bone, which is preferable for bone regrowth. A special class of cellular materials, based on triply periodic minimal surfaces (TPMS) have zero mean curvature, which makes them interesting for bone implants. Additive Manufacturing techniques (AM), such as Electron Beam Melting (EBM), can be used to produce scaffolds in Ti-6Al-4V, which has shown to have great biocompatible properties. However, EBM manufactured Ti-6Al-4V components show many surface defects. As cellular structures have a large surface area, these surface defects could significantly reduce its structural integrity and fatigue properties.

Throughout this thesis work, TPMS scaffold modeling is investigated. Features such as volume fraction and pore size are analyzed, such that these features can be precisely described in a design. An app is made with MATLAB, which allows the user to design cellular TPMS structure with advanced volume fraction distributions, using the distance transform of an input geometry. Fatigue specimen with a TPMS structure were designed using the scripts, and produced by EBM.

An electrochemical setup for polishing of EBM manufactured Ti-6Al-4V parts was made. Specially designed plates were tested with the polishing setup for variable polishing parameters, and characterized with Scanning Electron Microscope. The plates will later be used for cell growth experiments. The results from this study could hopefully contribute to a way of modifying the surface of Ti-6Al-4V scaffolds to improve both biocompatibility and structural integrity.

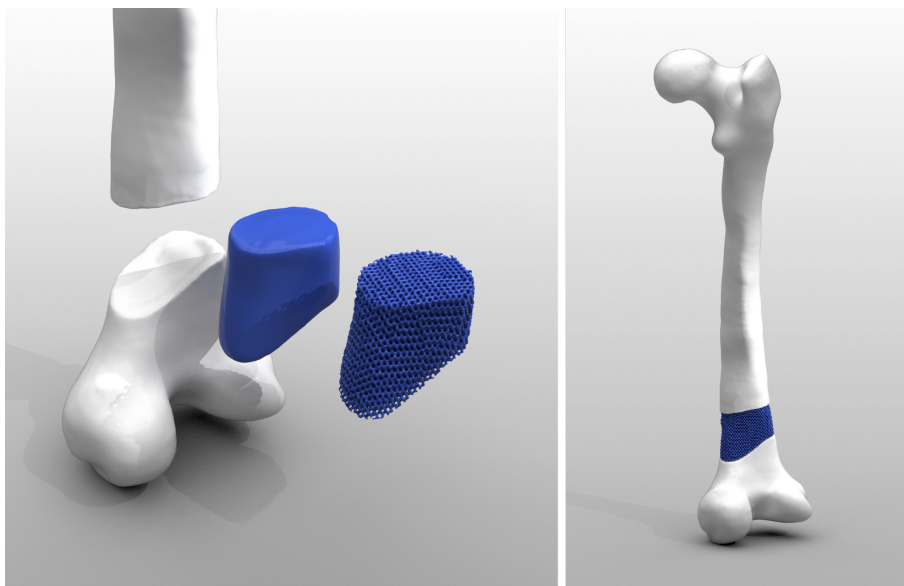


Figure 1: Demonstration of a scaffold made by the developed App.

Sammendrag

Cellebaserte materialer består av sammensetninger av små bjelker og plater som tilsammen utgjør et porøst volum. Ved nøye design kan disse strukturane produserast til å oppnå spesielle egenskaper, noe som gjør dem interessante for benimplantatsapplikasjoner. Slike cellebaserte materialer kan finjusteres til å oppnå samme stivhet som pasientens ben, noe som er foretrukket for benvekst. En spesiell klasse cellebaserte materialer, trippelt periodiske minimale overflater (TPMO), har en gjennomsnittlig krumning lik null, noe som gjør dem interessante for ben implantater. Additive tilvirkningsmetoder, som elektronstørlesmelting (ESS), kan brukes for å produsere cellebaserte materialer av Ti-6Al-4V, som har gode biocompatible egenskaper. Deler som er laget med ESS har en høy overflateruhet og mange overflatefeil. Siden overflaten til cellebaserte materialer er så høy, vil den strukturelle integriteten og utmattingslevetiden til implantater laget med denne tilvirkningsmetoden kunne bli betydelig redusert.

Gjennom dette tesearbeidet, har modellering av TPMO strukturer blitt utforsket. Egenskaper som volumfraksjon og porestørrelse er blitt analysert slik at disse nøyaktig kan bli tilskrevet et design. Et program er laget i MATLAB, som lar brukeren designe cellebaserte TPMO strukturer med varierende volumfraksjon utfra valgte designparametre og inndata geometri. Utmattingsprøver er designet ved hjelp av dette programmet.

Et oppsett for elektrokjemisk polering er laget for overflatebehandling av ESS produserte Ti-6Al-4V komponenter. Prøver formet som små sirkulære plater er bitt laget, og behandlet med poleringsutstyret med varierende prosessparametre, og karakterisert ved elektronmikroskop. Behandlingen har gitt en glattere overflate på prøvene. Prøvene vil senere bli brukt til celledyrkingsforsøk. Resultatene fra studien vil kunne bidra til en økt forståelse av effekten overflatebehandling har på strukturell integritet og cellevekst.

Contents

Preface	iii
Acknowledgment	v
Abstract	vii
Sammendrag	ix
1 Introduction	1
1.1 Triply Periodic Minimal Surfaces	2
1.2 Power Law Nature of Porous Structures	2
1.3 Homogenization	3
1.3.1 Representative Volume Element	3
1.4 Features that Affect Cell Growth	4
1.4.1 Geometrical design features	5
1.4.2 Materials	5
1.5 Algorithms	5
1.5.1 Marching Cubes Algorithm	5
1.6 Additive Manufacturing Techniques	7
1.7 Printing features	7
1.8 Surface Effects	9
1.9 Electrochemical Polishing	9
2 Aim of Thesis	11
2.1 Objectives	11
2.2 Approach	11
2.3 Research Background and Motivation	12
2.4 Outline	12
3 Experimental Methods	15
3.1 Scaffold Design	15
3.1.1 TPMS	15
3.1.2 Making the Scaffold	15
3.1.3 Volume Fraction Control	16
3.1.4 Pore Size	17
3.1.5 Scaffold App	18

3.1.6	Stiffness Matrix	19
3.1.7	Boundary Conditions	21
3.2	Surface Modification	22
3.2.1	Specimen Design	22
3.2.2	Sample Preparation	23
3.2.3	Chemical Etching	26
3.2.4	Electrochemical Polishing	26
3.2.5	Characterization	28
4	Results and Discussion	31
4.1	Scaffold Design	31
4.1.1	Volume Fraction Relations for Different TPMS	31
4.1.2	Example of Volume Fraction Gradients	31
4.1.3	Pore Size	32
4.1.4	Stiffness Matrix	32
4.2	Surface Modification	34
4.2.1	μ CT Imaging	34
4.3	Chemical Etching and Electrochemical Polishing	37
5	Conclusions	45
6	Reccomandations for Further Works	47
6.1	Removal of powders	48
	Bibliography	49
	Appendix	50
A	Code	51
A.1	Scaffold Coding	51
A.1.1	Function Definition	51
A.1.2	Measurement	53
A.1.3	Design	62
A.2	Electrochemical process control and measurement	67
A.2.1	More Details	72

Chapter 1

Introduction

In the recent years, the field of Additive Manufacturing (AM) has moved from being mostly a method used for prototyping and visualization, to become an important method for manufacturing finished products. The field of AM has grown massively, the variety of compatible materials and uses are constantly growing, and include a variety of different metals, polymers and ceramics. The two main motivations for using AM is the "batch-size-indifference" and "complexity-for-free". In the case of orthopedic implants, where the design will be patient specific, they will often be manufactured in a one-piece batch. Complexity for free indicates that there is a huge freedom in terms of making complex designs. Whereas traditional manufacturing techniques, such as machining or milling, will have an exponentially rising price in terms of complexity and even feasibility of design, the main costs related to AM is the time and material consumption.

A very useful class of complex structures are topologically ordered cellular materials. Typically there will be a small repeating unit geometry. Based on how much material is used, how the material is ordered within this repeating unit (topology), and the size of the unit, the mechanical characteristics, such as stiffness and strength, but also mass transport properties, will be affected.

Cellular materials produced by AM can have a wide range of applications, and are seen more frequently in uses such as aerospace, aviation industry, and high performance applications, where high specific strength and specific stiffness is desired. The implementation of lattice design tools in CAD-software is becoming more common (Siemens NX-12, Netfab, SolidWorks). Although many of the principles presented in this thesis are true for a wide range of applications, the focus for this thesis is on the application of tissue engineering structures, specifically orthopedic implants. In this context, an orthopedic implant is referred to a device which is meant to be a permanent or temporary replacement of a bone structure. By using cellular materials, a porous *scaffold* can be manufactured, for patient-specific implants.

There are two main reasons why topologically ordered cellular materials produced by additive manufacturing are ideal for orthopedic implants. Firstly, the stiffness of such a structure can be tuned to match that of trabecula and cortical bone, which has a stiffness 10-15

times lower than that of implant bulk material. Secondly, *osseointegration* may occur, meaning that the bone tissue may form a direct connection between the bone tissue and implant, and thereby integrating into the interconnected pores of the scaffold, making a robust interface between the bone and implant.

For a topologically ordered cellular material, there will be geometrical features that are repeated. The smallest repeating unit of a periodic structure is called a *unit cell*. There are two main categories of topologies that we find in literature of porous biomaterials; beam-based topologies, and sheet-based topologies. Beam-based structures will have structural elements consisting of a length which is generally greater than its diameter. A common way of creating beam-based structures, is by filling the space, either by one, or a combination of different polyhedron, and make the edges of the polyhedron a beam, typically with a circular cross section with constant diameter. By modifying the beam junctions, such that there is a radius has shown to increase the strength. Sheet based topologies, show some advantages over beam-based ones, as they show superior mechanical properties in terms of fatigue life. Most sheet-based porous structures have a closed pore structure, which is not desirable for biological purposes.

1.1 Triply Periodic Minimal Surfaces

Some structures can be made from surface representation. The class of surfaces that have gained a lot of attention are minimal surfaces. If such a surface is showing space group symmetry, and periodic in three independent directions, then it is called a triply periodic minimal surface (TPMS). TPMS are mathematically defined surfaces having a mean curvature equal to zero, and show some very interesting properties which are useful for scaffold design. It is shown that surface curvature has important effects when it comes to the regrowth of bones (Bade et al., 2017, 2018; Gouveia et al., 2017; Vetsch et al., 2016). It is shown that natural formation of trabecula bone is very close to zero (Nishikawa et al., 2001).

TPMS can be used both to make beam-based and sheet-based scaffolds. To achieve a sheet-based design, a thickness is simply added to the surface to obtain a volume. Since TPMS have two sides, they will split a unit cell into two distinct volumes. By repeating only one of the phases, a beam-based lattice appears. The volume ratios of the two different phases can be modified, to achieve variable beam diameters and volume fractions. While correct TPMS can only be described using differential geometry, and very complex functions, they can be approximated quite well using Fourier approximations.

1.2 Power Law Nature of Porous Structures

In the literature the term *relative density* is used to describe the same property. For the further writings of this thesis, the term volume fraction will mostly be used. The volume fraction ρ is defined as

$$\rho = \frac{V_{Solid}}{V_{Tot}} = \frac{V_{Solid}}{V_{Solid} + V_{Porous}}$$

Where V_{Solid} is the volume of the solid phase, V_{Porous} and V_{Tot} is the surrounding, and combined volume of the porous and solid phase.

Porosity, describes the same relationship, and is defined as $1 - \rho$. Gibson and Ashby provided many useful tools for characterizing different characteristics of cellular solids and physical properties. The most important amongst these are the equations relating the volume fraction of the solid to the elastic properties (Ashby and Gibson, 1997). They state that the Young's Modulus, E , of porous materials behave by a power law nature:

$$E = C\rho^n \quad (1.1)$$

Where C and n are power function coefficients, found either by analytical or experimental means. Similar expressions can also be made for the other Lamé parameters, shear modulus G and Poisson ratio ν . Depending on the degree of isotropy, there will be a different amount of Lamé parameters necessary to assemble the stiffness matrix.

1.3 Homogenization

Multiscale modeling is a way, within several fields in science and engineering of solving problems by analysing a system at different length and time scales and transferring insight about the behaviour at one scale to another scale. One single system can often be modelled at many different scales, depending on which features and behaviour are interesting to investigate. There are modeling techniques for sub-atomic scale, atomic scale, dislocation, dislocation interaction and many others.

One often uses the terms *micro-scale* and *macro-scale* as a reference to the relative size-scale, unrelated to actual dimensions (e.g. μm). For the context of this thesis, there will be a focus on how the mechanical behaviour observed at unit cell scale of the porous bio-material (micro-scale) can be transferred to describe the mechanics of a complete structure (macro-scale).

1.3.1 Representative Volume Element

Structures on the micro-scale often show either periodicity or randomness or a combination of the two. If we generate random microstructures (e.g. for porous structures), we might randomize the various beam cross sections, and each generation of a microstructure is a different realization. Several realizations make up an ensemble of realizations. By increasing the number of realizations, the ensemble will become statistically closer to the real heterogeneous structure. Another approach is to enlarge the sample, (e.g. by evaluating more beam elements). An ensemble showing statistically which is statistically able to describe the meso-

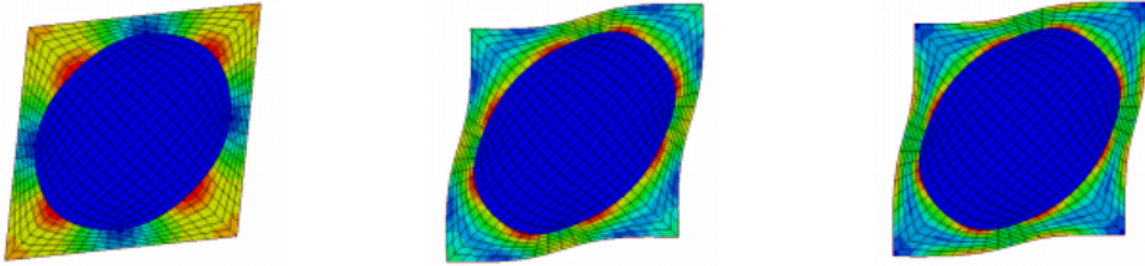


Figure 1.1: Illustration of the KUBC, SUBC and PBC for a two dimensional analysis. Image taken from Homtools webpage: <http://homtools.lma.cnrs-mrs.fr/spip/spip.php?article2>.

scale properties is called a representative volume element (RVE). For periodic structures, the RVE will often be the unit cell, however, sometimes there are deformation patterns that extend beyond the unit cell due to non-linearities such as buckling.

Homogenization is a way of estimating the average mechanical response of a heterogeneous material by making a Finite Element Analysis (FEA) on a RVE. There are three main methods to apply boundary conditions. Kinematic Uniform Boundary Condition (KUBC), Static Uniform Boundary Condition (SUBC) and Periodic Boundary Condition (PBC).

KUBC, also called Affine Deformation, is a way where deformation is prescribed to the boundaries of the RVE. After running a FEA, the stresses are averaged over the boundary. By multiplying the average stresses with the undeformed area, the reaction forces can be determined, and the effective stiffness of the RVE derived.

SUBC, also called Uniform Traction, is an evenly distributed surface traction that is applied to the boundaries. Upon running the FEA, the strains on the boundary are averaged, and once again the the stiffness properties of the structure can be obtained. However, a problem in this method is the low degrees of freedom, which can result in numerical instabilities. It is however possible to lock the degrees of freedom for one node, and use a set of weak springs to the avoid rotation and translation of the RVE.

Finally, with PBC, a uniform traction is applied to the boundary. Additional constraints are added to the boundary. Opposing boundaries of the RVE are constrained to undergo the identical deformation. For SUBC, the boundaries after deformation are not necessarily periodic after deformation. Whereas for PBC, both the the stresses and strains are averaged in the same way as for SUBC. Convergence of the averaged value will be quicker for PBC with respect to increasing the size of the RVE. A two dimensional dimensional example of the boundary conditions seen in Figure 1.1.

1.4 Features that Affect Cell Growth

There are various reasons for making cellular biomaterials for orthopedic implants. In the following subsections, a variety of different topics will be presented to give the reader a thor-

ough understanding of the basic theories and principles necessary for a good understanding of the field of additive manufactured porous metallic biomaterials.

1.4.1 Geometrical design features

As earlier described, different topological design will give different mechanical properties. This has a clear impact on the formation of bone. It is well accepted that bone regeneration is dependent on mechanical stimulus applied to the cells, explained by the Wolff's law (Chen et al., 2010; Christen et al., 2015). Not only is the mechanical characteristics affected by the topological design, but also mass transport properties, such as permeability, which will affect cell nutrition and oxygenization.

1.4.2 Materials

The body reacts in different ways when placing a material inside the body. Biotolerant materials, such as stainless steels, are not necessarily rejected when implanted into living tissue, but are surrounded by a fibrous layers. With bioinert materials, such as Ti-6Al-4V, the adjacent bone can under certain conditions form a direct contact with the implant surface. A third class are bioactive materials, typically different kinds of ceramic materials. These materials will also allow the formation of bone on the surface, but in addition, ion exchange can lead to chemical bonds between the implant and bone.

1.5 Algorithms

Throughout the work of this thesis different algorithms were used during scripting and designing of the TPMS structures. A few of them will be described in this section.

1.5.1 Marching Cubes Algorithm

Often one wishes to convert three dimensional, gridded scalar data (3D image), given in volumetric pixels (voxel) to a 3D surface mesh. A typical approach is to generate an iso-surface. One of the most popular algorithms for extracting an isosurface is the Marching Cubes algorithm. This algorithm splits the 3D image into a finite number of cubes, where each cube is containing one voxel for each corner of the cube. Next, the algorithm will "march" through each cube, evaluating the 8 voxels, and assigning the appropriate triangulated mesh to the cube volume. Fifteen possible combinations of triangulated surfaces can be made, illustrated in Figure 1.2. After the triangulated iso-surface is generated, additional surface mesh modification can be performed. An example of a Marching cubes generated surface is seen in Figure 1.3.

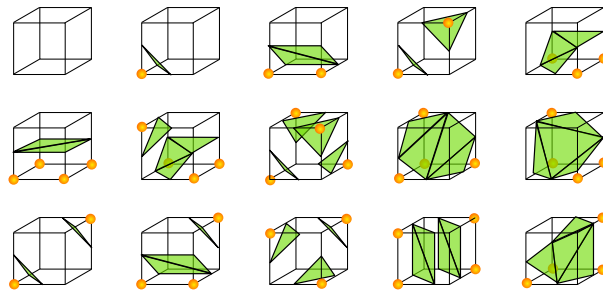


Figure 1.2: The original 15 different cube arrangements. The yellow label indicates a scalar value smaller or equal to the iso-value. With different kinds or symmetry, these basic cubes can build any surface.

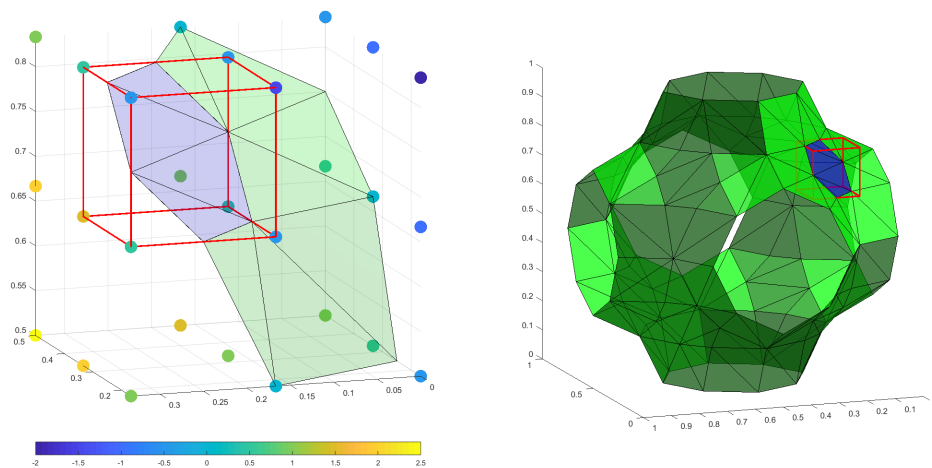


Figure 1.3: Left: A voxel grid consisting of 3^3 scalar values, graphically displayed as points with a color value corresponding to its scalar value. To produce an iso-surface with the iso-value, $T = 0$, the "marching cube", illustrated in red, shows how the algorithm evaluates all the eight scalar values associated with the cube, and selects the correct cube configuration. Right: Shows the P-unit cell with a resolution of 7^3 with the same marching cube.

1.6 Additive Manufacturing Techniques

Among the AM-manufacturing techniques used for metals, there are three main categories cited in the literature. Direct energy deposition techniques, material extrusion techniques, and powder bed fusion techniques. For direct energy deposition, a moving nozzle, often in combination with a rotating and tilting work bed, will deposit metallic powder which is directly sintered or melted to the part as it is built. The energy source will often be a powerful laser. The technique is frequently used for large parts.

The material extrusion technique is usually performed in a way where the metallic material is bound together with a binder material, using simple filament deposition molding. This can be achieved even with cheap machines. Subsequent to the printing, removal of the binder is necessary, and the metal is properly sintered or melted together. Finally, powder bed fusion (PBF), the technique which has gained the greatest interest with respect to manufacturing of metallic biomaterials. The technique works by distributing a fine layer of metallic powder on a build plate, followed by melting the powder using a highly energetic beam. Selective laser melting (SLM) and Electron Beam Melting (EBM) are the two methods used. There are some differences with respect to how these techniques effect the resulting product. SLM uses a set of optics and mirrors to direct the laser beam. Electron Beam Melting (EBM) is an AM technology that utilises a focused electron beam to melt metallic powder into a solid part. A prerequisite for a controllable electron beam is to have vacuum so that the electrons will not collide with gas molecules. As seen in Figure 1.4, the EBM system consists of a vacuum chamber and an electron beam column with a series of lenses.

During the manufacturing with EBM, a layer of metallic powder is evenly distributed on the build platform with a rake. Typical layer thickness is in the range of 50 to 100 μm . When the focused electron beam is passing across the powder on the build plate, it generates sufficient with heat that the metallic powder in the immediate surrounding of the focus point melts. Due to the high thermal conductivity of metals, this heat is soon dissipated to the surrounded material causing the pit of molten material to solidify. In certain cases it can cool so quick that the metal is quenched. After one layer is melted, the build plate is lowered and a new layer of powder is distributed. The fact that the melting is happening in vacuum eliminates the need for an inert gas atmosphere. The astigmatism lens and focus lens are intended to focus the in a single point, where as the deflection lens shifts the focused beam to the desired location of the building plane.

The EBM technology is patented by the Sweedish manufacturer, Arcam EBM.

1.7 Printing features

When printing straight in up, the material will support itself. However, most AM techniques will require some sort of supporting structure when the overhanging structure feature exceeds a critical level or range. This is because the deposited material is not able to support

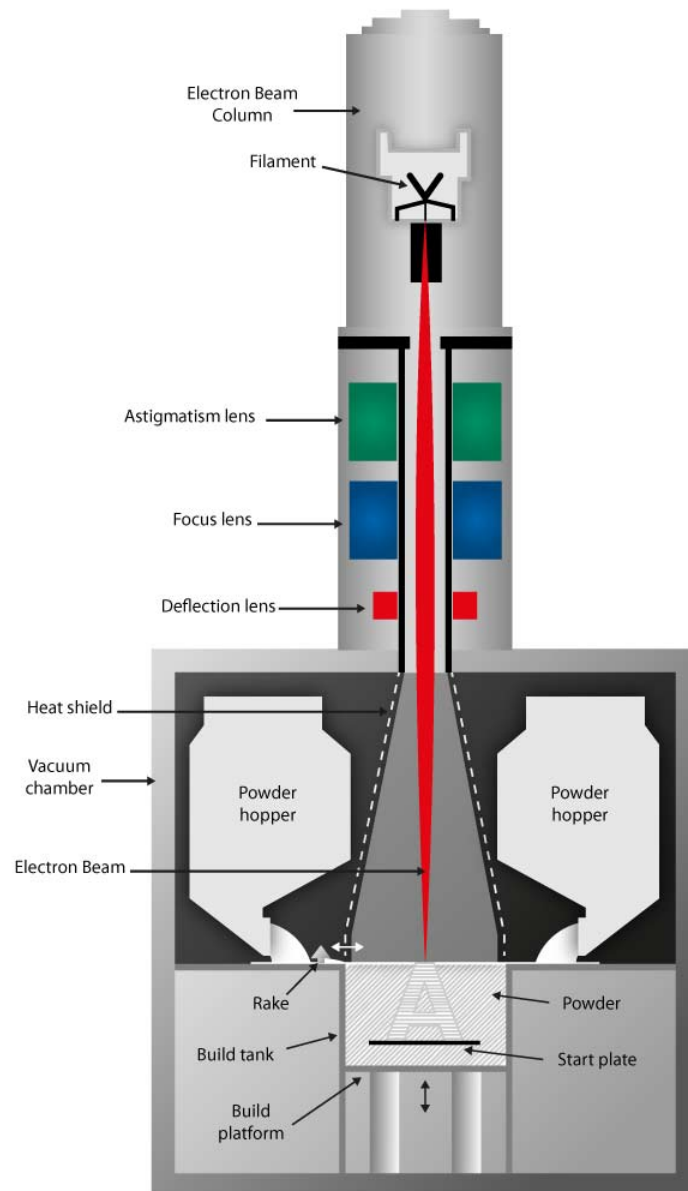


Figure 1.4: This is a schematic of an EBM system. Figure is borrowed from the manufacturer web page: <http://www.arcam.com/technology/electron-beam-melting/hardware/>

its own weight. However, for powder bed methods, the powder bed will provide some level of support, and be able to support the bridging parts of a cellular material.

After printing, unmelted powder must be removed. It is shown in some cases to be difficult to remove unmelted powder, for cellular materials, where the pore sizes are small.

1.8 Surface Effects

An artifact of the powder bed manufacturing technique, is that the surface roughness is very high, and will often have partially melted powders on the surface. This can cause cracks to initiate and propagate upon cyclic loading.

1.9 Electrochemical Polishing

Material can be removed by electrochemical polishing (ECP), using an electrolytic bath, with two electrodes, anode and cathode. The electrodes are connected to a direct current power supply, where voltage and current can be controlled. A commonly accepted theory is that a viscous layer, called the anolyte layer, containing byproducts from the polishing forms on top of the surface. With a variable height profile of the material surface, the anolyte layer thickness will vary. A thinner layer will give a lower electrolytic resistance, giving a higher current flow. This increased current flow makes the polishing go faster in the elevated areas, causing a leveling effect. Many ECP setups will have a relationship between the input voltage and current density (CD) 1.5.

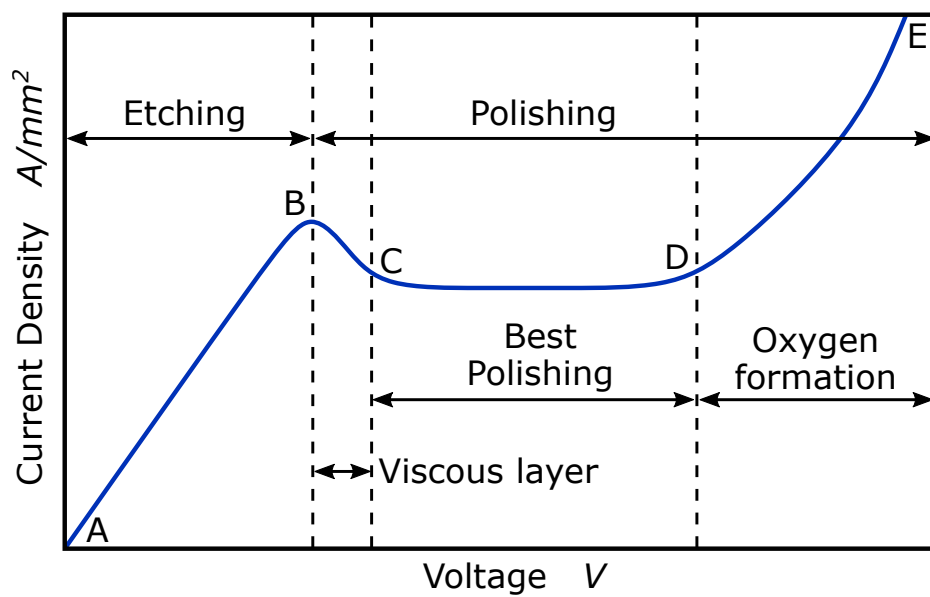


Figure 1.5: Characteristic plot of a current density vs voltage for electro chemical polishing. At low voltages, there will not be any polishing, just etching. Above some threshold V_B , the polishing reaction takes place. Between B-C, there is formed the anolyte layer, and between C-D the best polishing will take place. Above D, there will form oxygen gas on the surface, giving uneven polishing.

Chapter 2

Aim of Thesis

2.1 Objectives

The main objectives of this Master's thesis can be summarized as:

1. Develop a method to generate CAD-files of porous cellular structures, making them easy to design, specifically focused on making lattices that are meant for tissue engineering and orthopedic implants. The design workflow must be in such a way that it allows the designer to control relevant parameters for implants.
2. Develop a process for surface modification of scaffolds made by EBM manufactured Ti-6Al-4V, such that the scaffolds can enhance their performance with respect to its structural integrity as well as cell adhesion and proliferation to the material.

2.2 Approach

Get familiar with the design methodology for scaffold materials. Through investigation, a method for designing scaffolds with TPMS structures has been found, and adopted. From a lack of available software to produce these structures a determination was made to build one from scratch using Matlab. Through the course of the master work, the necessary skills have been acquired, and code been made. The basic design concept is simple, but controlling the design parameters is not simple. Therefore numerical experiments have been done, to relate different mathematical features, its more comprehensive and useful geometrical parameters as described in literature. To further control the behaviour of the scaffold, the stiffness properties could be tuned with varying the volume fraction. By designing a representative volume element, the stiffness matrix could be determined from finite element analysis. The representative volume elements were designed using the self written Matlab Script, and then a scripting procedure with Abaqus was attempted to obtain the constitutive variables. Lastly, the Matlab script was modified for usability and extended functionality, such that an arbitrary solid geometry can be imported and turned into a lattice, with easy methods for creating functional gradients in the lattice.

For the surface modification of Ti-6Al-4V, etching with strong etchants, in particular Hydrofluoric acid, necessitates cleanroom training to work in the NanoLab at NTNU, which is the only place at NTNU for safe handling of this chemical. The NanoLab also offers good facilities for characterization, with a scanning electron microscope and optical profilometer. An electrochemical polishing setup must be made and tested to understand the characteristics of the polishing. As the polishing is characterized, different surfaces can be produced for cell growth research. The developed scaffold script can be used to make tension specimens for a fatigue analysis, and as the etching procedure is developed, the specimen can be treated, to investigate effect of surface treatment on the fatigue performance of the material.

2.3 Research Background and Motivation

The science behind biomedical implants is diverse. The combination of mechanical engineering fields focusing on construction and design, merging with Additive Manufacturing techniques and analyses for their mechanical behaviour and structural integrity, is intricate itself. The fact that these objects go into ones bodies, involve completely different fields of science, requiring knowledge in cell functions and behaviour, and how they interact and interface with implants. The total makes a complex field, requiring a collaboration between different disciplines. The width of the field is something that intrigued me.

When I was introduced to this project as a semester thesis, one of the initial objectives was to improve the fatigue performance of additive manufactured scaffolds. However, when discussing the stiffness properties of some previously manufactured scaffold of our lab, my intuition was that there could be a better scaffold topology than the one used at that present. Upon a small search I discovered an exiting research field in the design of cellular structures for additive manufacturing. In an attempt in realizing some of the structures that I read about, I found that design tools where sparse, lacking the functionality that I desired. This intrigued me to develop my own, and eventually shaped large amounts of thesis work.

To perform some of the necessary processing, cleanroom facilities at NanoLab at NTNU was needed, requiring substantial training. Once getting a foot inside NanoLab, I got the chance to see many processing techniques and characterization methods that I was unfamiliar with. Several of which proved to be useful for this thesis work. This opened up new fields of professional interest and skills, and has contributed in shaping my future carrier plans.

2.4 Outline

Within the first chapter of this thesis, a presentation will be made of the underlying theory, forming a baseline for the research conducted in this thesis. This includes the reasoned use of materials and structures in orthopedic implants, and further how these structures

can be modelled and analysed. An introduction to the principles in some of the possible modification techniques are made.

The subsequent chapter contains a description of the methods, code developments, and experiments used throughout the thesis work. Both successful and unsuccessful methods that had to be discarded are described.

Because of the diversity of topics studied, the results and discussions are given within the same chapter to avoid jumping back and fourth within different parts. The next chapter will conclude with the biggest findings of the study, before the last chapter will deal with potential future works, based on improvements and new ideas based on findings from own work in combination with findings in literature. After these chapters, the cited literature is listed, and relevant appendices including much of the code, and additional figures is supplied. At the very end, a draft of a paper to be published in a journal is supplied. The paper regards a pore size analysis, that has been conducted as part of the master's thesis. The paper is a draft, and will still be edited, before submitting to review. It will be submitted to the journal *Materials Design and Processing Communications*.

Chapter 3

Experimental Methods

This chapter will go through the experimental methods used for this study. First of all, the scaffold design methodology is described. There are some results presented, as they are key for the further development and explanation of the methods. Further, the experiments and surface treatment conducted with EBM-manufactured Ti-6Al-4V specimen are described.

3.1 Scaffold Design

3.1.1 TPMS

Several of the most common TPMS functions with a cubic symmetry are used. The Schwarz P "Primitive" and Schwarz D "Diamond" surface is used to make the beam based structures. The TPMS Gyroid surface was discovered and described by P and D. Shoen, which is used to make the beam based G1. By making offset to both sides of the Gyroid surface, two separate, intertwined gyroid beam structures can be made, referred to as G2. The Primitive, Diamond and Gyroid surfaces are also modelled as sheet-based structures, given the names PS, DS, and GS. Shoens IWP surface is used for making the two beam based models, IWP-1 and IWP-2. The Shoen's OCTO TPMS is used to make the beam based OCTO. Examples of the different unit cells are given in Figure 3.1. The functions are defined as listed in table 3.1.

3.1.2 Making the Scaffold

MATLAB was chosen as a tool for creating the parametric designs. Apart from using MATLAB there are other tools that could have been used. Rhino, which is a scripting based program for parametric design could have been an alternative. Using Grasshopper, would assist for users with a non programming background. Due to the licence cost, the program was not used for this work.

Table 3.1: TPMS functions definitions as defined in this script. Notation in the table is as follows: $C_i = \cos(ik)$, $S_i = \sin(ik)$, $C_{2i} = \cos(2ik)$ where $i = x, y, z$, and $k = \frac{2\pi}{a}$, where a is the unit cell size.

TPMS lattice type	Surface model $f_S(x, y, z) =$
P (Schwarz P beam)	$C_x + C_y + C_z = T$
PS (Schwarz P sheet)	$(C_x + C_y + C_z)^2 = T^2$
D (Diamond)	$S_x S_y S_z + S_x C_y C_z + C_x S_y C_z + C_x C_y S_z = T$
DS (Diamond sheet)	$(S_x S_y S_z + S_x C_y C_z + C_x S_y C_z + C_x C_y S_z)^2 = T^2$
G1 (Gyroid)	$C_x S_y + C_y S_z + C_z S_x = T$
G2 (Gyroid 2)	$(C_x S_y + C_y S_z + C_z S_x)^2 = T^2$
GS (Gyroid Sheet)	$(C_x S_y + C_y S_z + C_z S_x)^2 = -T^2$
I-WP 2	$2[C_x C_y + C_y C_z + C_z C_x] - [C_{2x} C_{2y} C_{2z}] = T$
I-WP 1	$[C_{2x} C_{2y} + C_{2y} C_{2z} + C_{2z} C_{2x}] - [C_{4x} + C_{4y} + C_{4z}]/2 = T$
OCTO	$4[C_{2x} C_{2y} + C_{2y} C_{2z} + C_{2z} C_{2x}] - 3[C_{2x} + C_{2y} + C_{2z}] = T$

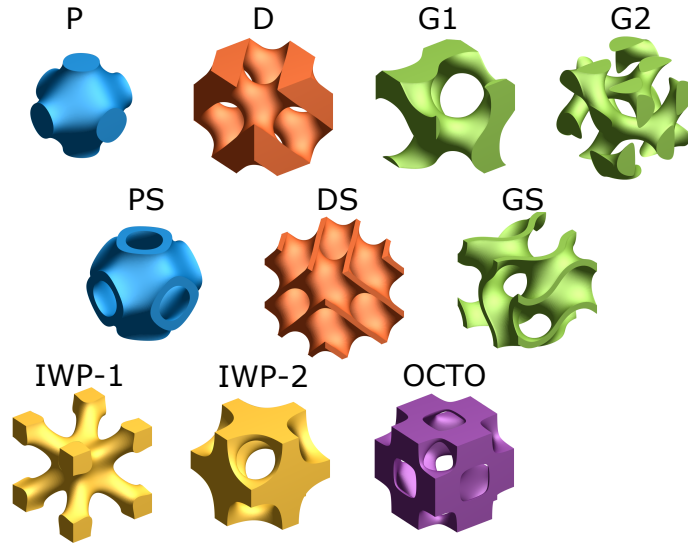


Figure 3.1: TPMS unit cells.

3.1.3 Volume Fraction Control

To be able to control the volume fraction of the different unit cells, a relationship between T , and ρ must be determined. The volume fraction of the lattice is measured by calculating the volume of the lattice and divided by the enclosing bulk volume. Given that the volume is closed, the volume of the triangulated surface is very easily calculated using the divergence theorem (Zhang and Chen). The volume fraction is computed for approximately 30 isovalues, T . A fifth degree polynomial is fitted to the data points corresponding to $0.05 < \rho < 0.95$, such that:

$$t(\rho) = c_1 \rho^n + c_2 \rho^{n-1} + \dots + c_n \rho + c_{n+1} \quad (3.1)$$

where $c_1 \dots c_{n+1}$ are the polynomial coefficients and $n = 5$. The polynomial coefficients are stored in a separate function, `volFrac2isoVal()`, where the isovalue T for a given lattice is

evaluated based on the input volume fraction ρ .

3.1.4 Pore Size

A script was developed in order to obtain pore size and pore throat size for a given TPMS unit cell, using MATLAB function given in the Appendix. There are four main steps which is performed by the function *poreSize()*:

1. A voxel representation of a TPMS surface with the desired volume fraction is made. Because of symmetric properties of the unit cells, one octile of a full unit cell is sufficient for analysis, and will enhance performance. The voxels corresponding to dense material is given boolean value true(white), and the void is given false(black).
2. Next, a distance transform is made of the void space (Maurer et al., 2003). This means that the euclidic distance is calculated for every black pixel to its closest white pixel, and stored as a scalar value to that voxel. The inbuilt MATLAB function *bwdist()* is used for the euclidic distance transform.
3. A skeletonization is performed on the void volume, using the inbuilt matlab *bwskel()* function, which reduces a 3D volume to a 3D line, using the medial axes transform (Lee et al., 1994). The result is a set of voxels that represent the core, or the "skeleton" of the input volume. Figure 3.2 shows the skeleton within different unit cells.
4. By using the index positions of the skeleton image on the distance transform, an array of possible void sizes (by radius) is given. Taking the minimum value, the pore throat size is given, and taking the maximum value, the maximum pore size is given. The obtained skeleton doesn't necessarily correspond to the exact maximum-ridge of the distance transform. Therefore, the maximum value from the local surrounding of the position of the previously obtained minimum and maximum pore size positions are used to improve the accuracy.

In addition to finding the pore size, the minimum and maximum thickness, or beam diameter is given. In practice this is the same calculation as the pore size, except that the solid phase, instead of the void phase is considered. A second function, *poreSizeSweep()*, sweeps through linearly increasing volume fractions, calling on *poreSize()*. The obtained dimensions are fitted to 3rd degree polynomials.

Functional Gradient

The volume fraction can be set as a field function, $\rho(x, y, z)$ and passed on and evaluated by *volFrac2isoVal()* to obtain $T(x, y, z)$, and set to create a functional gradient through the structure. The volume fraction can be varied in different ways, such as in a linear fashion. A more sophisticated way of controlling $\rho(x, y, z)$ is with a distance transform of a chosen geometry, and increase or decrease with a given distance from the boundary surface.

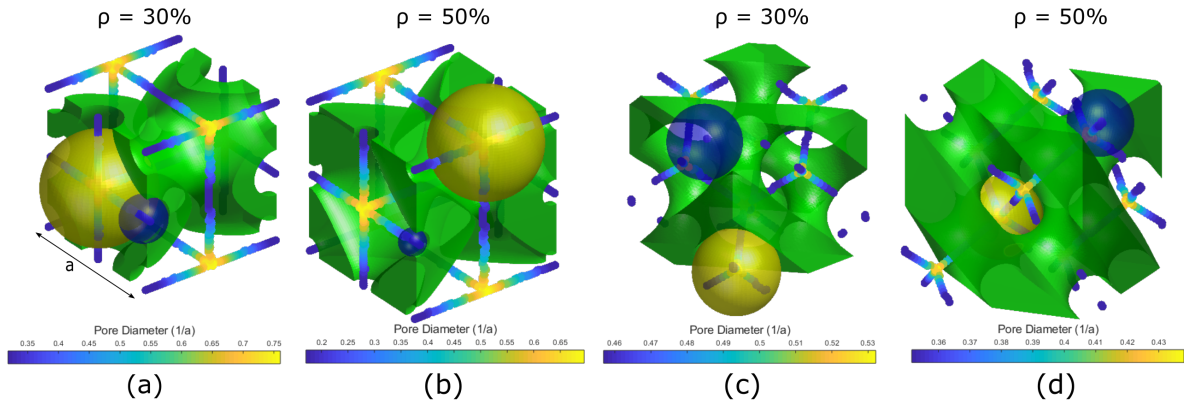


Figure 3.2: (a) and (b) show the unit cell of PS. (c) and (d) show the unit cell of D. The interconnected lines are the skeleton of the porous phase, and the color of the skeleton indicates the distance to the nearest surface.

3.1.5 Scaffold App

A Matlab App is developed, that allows for creating TPMS cellular structures inside a given design volume. The chosen input geometry is given as an STL file, and the desired TPMS is chosen, with and volume fraction. By choosing two different volume fractions ρ_1 and ρ_2 , an *offset length* L_O , and *ramp length* L_R , the app will generate the volume fraction function as:

$$\rho(D(x, y, z)) = \begin{cases} \rho_1 & D(x, y, z) < L_O \\ \rho_1 + (\rho_1 - \rho_2) \frac{L_O + L_R - D(x, y, z)}{D(x, y, z)} & L_O < D(x, y, z) < L_O + L_R \\ \rho_2 & D(x, y, z) > L_O + L_R \end{cases} \quad (3.2)$$

where $D(x, y, z)$ is the distance transform. The distance transform is also used to set the boundary of the scaffold, with a penalty function, meaning that the farther away from a set point, the density will decrease very quickly. The penalty is implemented by altering the TPMS function $f(x, y, z)$, and defining $f_{OUT}(x, y, z)$, which will be used for making the iso-surface. Figure 3.3 illustrates the use of the distance transform.

$$f_{OUT} = f - (n \times D)^m \quad (3.3)$$

Where n and m are factors used to set the aggressiveness of the penalty. Trial showed that $n = 1.5$ and $m = 2$ would work well.

A graphical user interface is made for the scaffold program, to ease the use (Figure 3.4). The user can browse for an STL file, which will be used as the boundary of the scaffold. ρ_1, ρ_2, L_O and L_R can be set, according to equation 3.2. Unit cell size is defined. The user is allowed to view a preview of the volume fraction distribution, from three orthogonal cross section views.

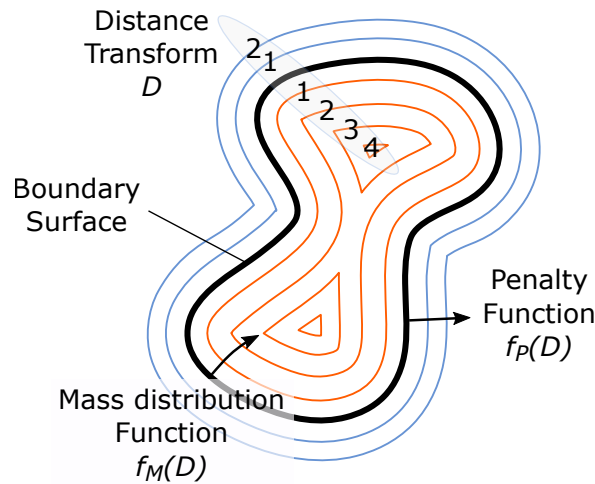


Figure 3.3: A distance transform D is made from the boundary surface of the input geometry. This distance field is used as an input to a mass distribution function, controlling the volume fraction at various points. In a similar fashion a penalty function can reduce the mass exponentially with the distance away from the surface.

3.1.6 Stiffness Matrix

The stiffness matrix of an orthogonal structure needs 9 independent variables. Given an instance where the properties are equal in the three orthogonal directions, these condense down to 3 independent variables. This could either be done analytically, numerically, or by experimental means. For very low volume fractions, some of the TPMS structures can be approximated as beams, and give a good representation of the stiffness of the structure. The experimental approach is used to get the Young's Modulus, but the shear modulus is quite difficult to obtain experimentally, although some approaches are suggested for measuring shear stiffness on lattice cores for sandwich structures (Liang and Wadley, 2016). A numerical approach is chosen, based on the need for various measurements of different volume fractions.

The finite element mechanical solver, Abaqus was used. In an attempt to automate the procedure, the Abaqus scripting functionality was used. It allows the user to record all actions done by the user, and later edit these in a Python Script. The following actions were recorded using the scripting function: Imported as STL structure designed using the Matlab Script, converted to a solid geometry (file format). Unlike importing a .STEP file, or other typical CAD-files, geometric features such as edges and faces are not recognized when importing a surface meshed structure, such as a triangulated surface model. Using the virtual topology functionality in Abaqus, these geometric properties were distinguished. Further the model was seeded along the edges, and meshed with linear tetragonal elements.

Plugin called *Homtools* which is a homogenization toolbox for Abaqus (Lejeunes and Bourgeois). It consists of a set of Python scripts which helps to automatize the necessary procedures to retrieve the homogeneous properties of an RVE.

Both a mesh convergence analysis and an RVE size convergence analysis was conducted.

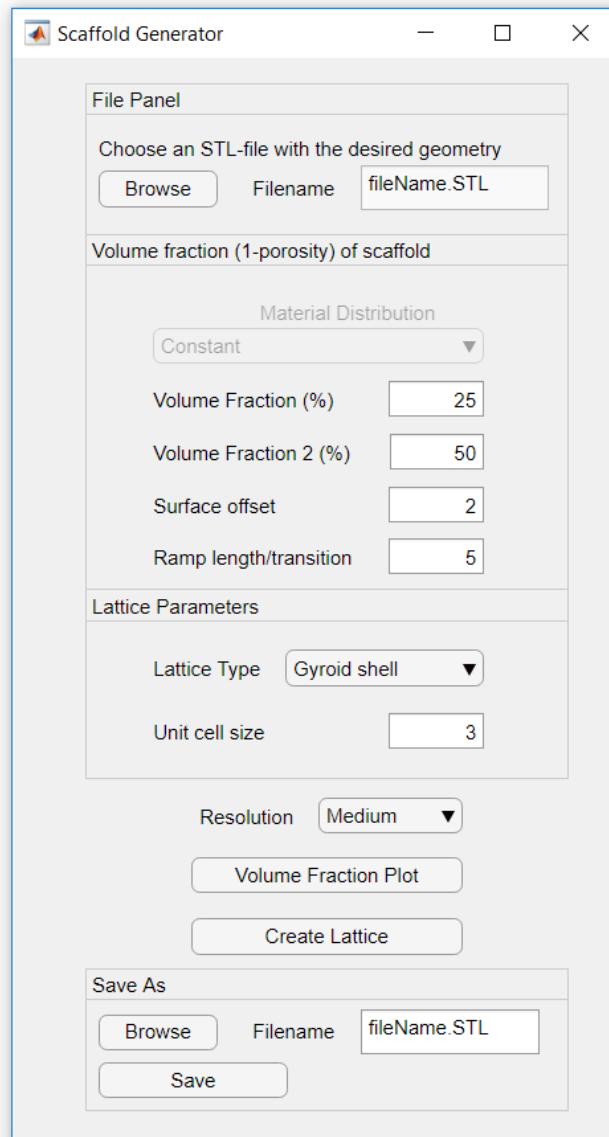


Figure 3.4: Scaffold App

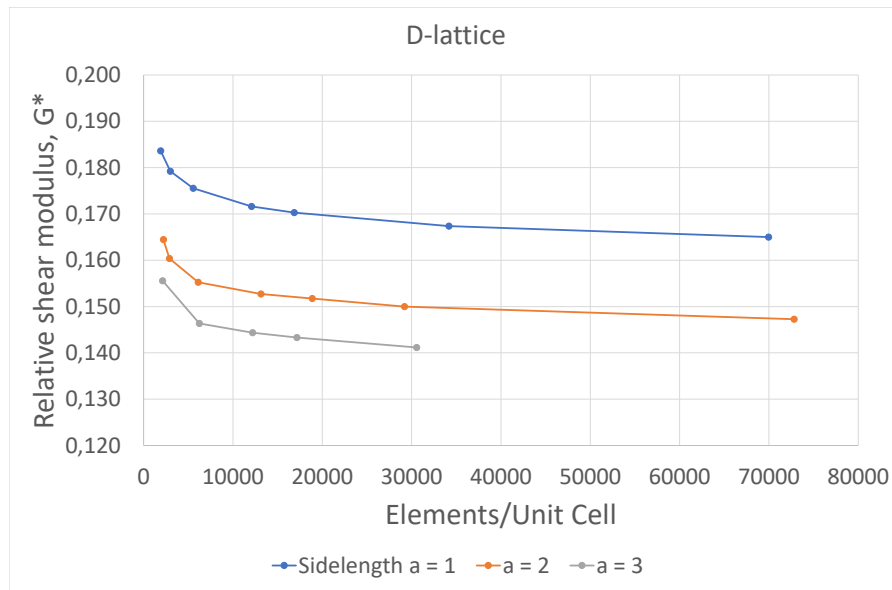


Figure 3.5: This is my caption

For the meshing, a $3 \times 3 \times 3$ unit cell cube was used. Further, an RVE size analysis was done. Unit cells ranging from 1^3 to 5^3 were made.

3.1.7 Boundary Conditions

Period boundary conditions lead to a quicker convergence in terms of unit RVE size. Given an integer number of unit cells wide cuboid will have, at least ideally, periodic boundaries, it makes sense to use these boundary conditions for homogenization. For the FEM solver to work, element nodes on the boundaries must be coupled with the corresponding node on the opposite side of the RVE. The edges of the virtual topology was seeded, and the volume mesh generated. To achieve identical surface meshes of opposite sides the following steps were performed. First, the surface meshes of the boundary mesh with negative normal vector sign were copied as a reference. Next, the volume mesh was deleted. The reference surface meshes are mapped to the boundary surfaces, both the positive and negative normal vector sign. Using the meshed boundary surface and a global element size control, the volume could be meshed top down.

Using kinetically uniform boundary conditions, the size of the RVE must be set at a level where the average property converges as the size of the RVE increases. The FEA analysis was conducted with a nonlinear solver, linear space functions.

There was an increasing number of unit cells and elements within each unit cell, until the computer would crash. The crash would come when the Python script prepared the FEM model, not the FEM calculation itself.

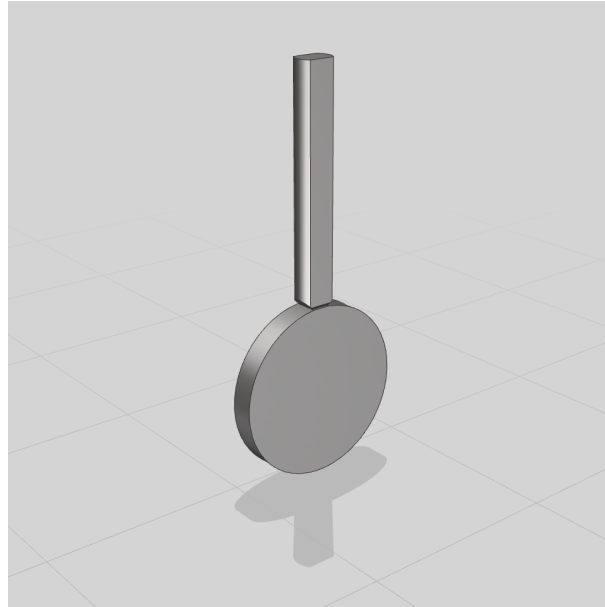


Figure 3.6: The small disk is 14.6 mm in diameter and 2 mm thick. A sharp notch is placed in the transition between the disk and the 20 mm long rod, for easier detachment after surface treatment.

3.2 Surface Modification

3.2.1 Specimen Design

For further studies, with Kristin Sirnes Ødegaard, smaller disks were used with diameter 14.6 mm and thickness of 2mm, as they fit into wells used for cell culturing. The small disks were designed with a 20 mm rod, such that the disk can be fully submerged in the electrolyte, yet keeping part of the specimen above the electrolyte surface to connect to the power supply, without causing damage to the wiring or risking the integrity of the connection during experiments. Figure 3.6 shows the design used. The specimen were manufactured using EBM by FIT AG, Germany.

During manufacturing, the parts were printed with the pin pointing along the z-axis, and had some support structure on the bottom downwards facing surfaces. The layer thickness was 50 μm . Powder size varies between 45- 100 μm . The manufacturer would not specify other parameters, such as beam intensity and beam velocity. For the following experiments, 16 samples were tested, ECP7 - ECP23. ECP1-ECP6 were used for calibration purposes.

Further, a novel specimen was designed to be used for fatigue testing. The main interest for orthopedic implants loading scenario is compression loading. Still the specimen was designed with gripping elements, and load introduction segments, in order to eliminate buckling effects. Scaffold was designed with a D lattice structure in the [100] lattice orientation. Unit cell size was 3mm. To reduce lattice-edge effects, the specimen were designed to have

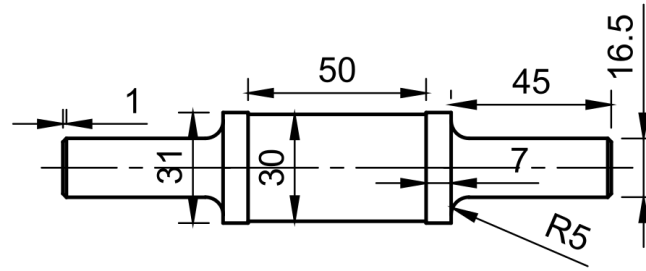


Figure 3.7: Dimensions of the fatigue specimen. The middle cylinder is made of a lattice design constructed with a MATLAB script.

10 unit cells in diameter, with total diameter of 30 mm. To reduce stress concentration at the load introduction, a material gradient was applied in form of a varying volume fraction along the main axis of the specimen. The dimensions, except of those regarding scaffold parameters, are given in Figure 3.7. Similar papers (Lietaert et al., 2018) used a linearly varying volume fraction, with a linear section. Because of deformation patterns that tend to align in a 45 degree angle, it was desirable to maintain a section in comparable length to the width of the specimen, with a constant volume fraction, with a lower volume fraction which is comparable to the width of the specimen. An improved version of this design was made using a quadratic function for the volume fraction. This was to achieve a smooth transition from the linear section, but at the same time not having to make a very long sample. The upper limit was set to 70%, which is believed to be sufficiently high to not break, despite the stress riser of being connected to the bulk material. A plot of the volume fraction target function is seen in 3.8. Prototypes of the scaffold, without load introduction sections, were made by stereolithography printing (SLA) using Way2Production SolFlex350 with FlexVat (Figure 3.9). The specimen were manufactured using EBM by FIT AG, Germany.

The volume fraction, ρ , was defined as a function of Z .

$$\rho(Z) = \begin{cases} \rho_{min} + (\rho_{max} - \rho_{min}) \left(\frac{Z-Z_0}{Z_1-Z_0} \right)^2 & Z_0 < Z \leq Z_1 \\ \rho_{min} & Z_1 \leq Z \leq Z_2 \\ \rho_{min} + (\rho_{max} - \rho_{min}) \left(\frac{Z-Z_2}{Z_1-Z_2} \right)^2 & Z_2 < Z \leq Z_3 \end{cases} \quad (3.4)$$

where Z_0, Z_1, Z_2, Z_3 were set to be 0, 20, 30, 50 mm, ρ_{min} and ρ_{max} to 0.30, and 0.70.

3.2.2 Sample Preparation

The samples were washed using an ultrasonic bath (Elmasonic P, Elma Schmidbauer GmbH) in deionized water at 50 °C, for 10 minutes, at a frequency of 37 kHz. The samples were later rinsed with ethanol (90%), and dried with Ni gas.

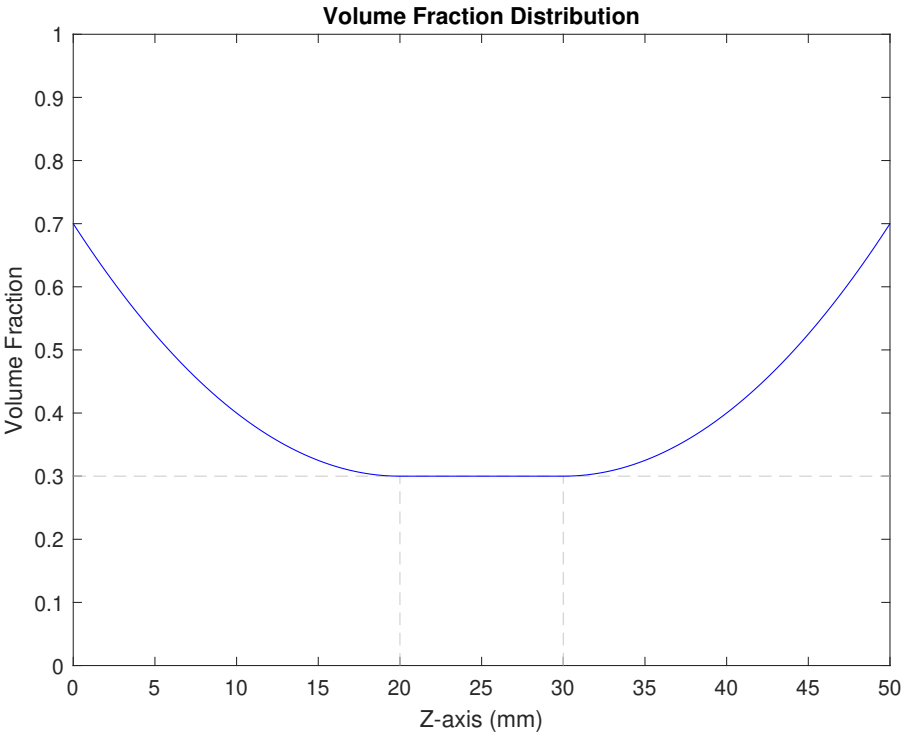


Figure 3.8: Plot showing the volume fraction function along the main axis of the specimen, which is passed on to the script used to generate the lattice structure.

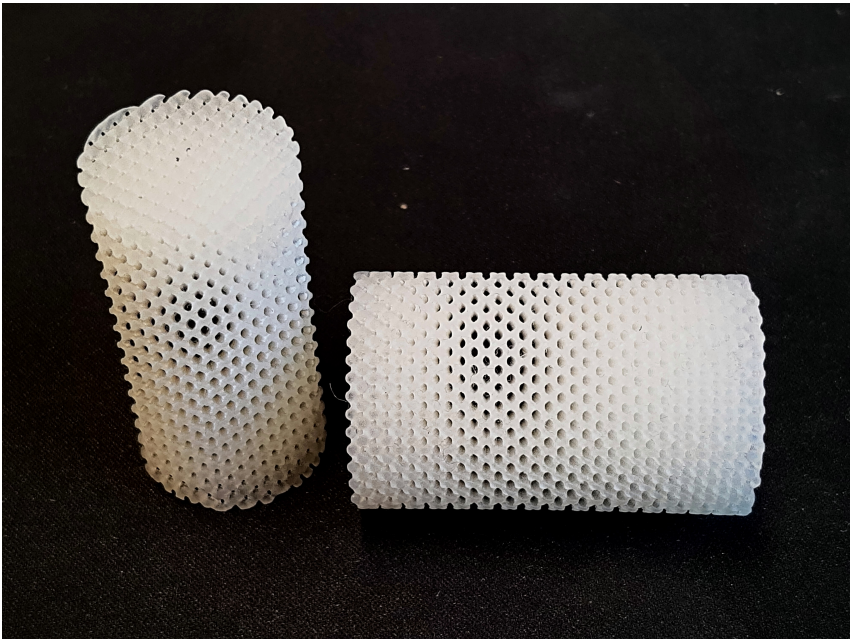


Figure 3.9: SLA produced prototype of fatigue specimen scaffolds. The circular cross section of each have a diameter of 25 mm and 30 mm.



Figure 3.10: A batch of EBM manufactured Ti-6Al-4V fatigue specimen with TPMS D topology and variable volume fraction. The image show two specimen displayed from different angles.

Table 3.2: The solution for chemical etching. The amount listed is enough to make 100 mL solution.

Chemical	Formula	Concentration (Label)	Amount
Hydrofluoric Acid	HF	48 %	1 ml
Deionized water	H_2O		99 mL

Table 3.3: The electrolyte used for electrochemical polishing. The amount listed is enough to make 100 mL electrolyte.

Chemical	Formula	Concentration (Label)	Parts
Acetic Acid (Glacial)	CH_3COOH	100%	55 ml
Sulfuric Acid	H_2SO_4	95-97%	30 ml
Hydrofluoric Acid	HF	48%	15 ml

3.2.3 Chemical Etching

The solution used for the chemical etching is given in table 3.2.3. The amount listed is enough for 100 mL solution.

Etching was done for 5 minutes in one batch, then 3 minutes in another bath with the same solution. To ensure safe handling of the specimen in and out of the chemical etchant, and to make sure that the surface of the specimen are properly submerged and aligned during the etching, a small stand was made, in which the specimen where placed. The holder was manufactured using the AM technique filament deposition molding (FDM), with polypropylene (PP) based filament.

3.2.4 Electrochemical Polishing

The solution used for the electrolyte is listed in table 3.2.4.

For electrochemical polishing, a wide beaker, measuring 180 mm was used to contain the elctrolyte and electrodes. The cathode used was a 40×40 mm platinum mesh. This was used due to its chemical inertness. The Ti-specimen worked as an anode. During the polishing, the beaker was placed on a heat plate, set to 30°C , with magnetic stirrer at 400 rpm. To ensure an even flow across the surface of the specimen, and to promote repeatability, a sample holder and fluid flow channel was designed and manufactured using FDM. It was made to perfectly fit into the wide teflon beaker, and was made with the same PP material as described earlier. Figure 3.12 shows the ECP setup in the fume hood. A Keysight 5771A was used as a power source, being interfaced with a PC. A computer script could be used to control the output current and measure the output voltage with the code `Measurement()`. To start with, a few samples were etched with a sweeping current density, starting at 0 mA mm^{-2} ,

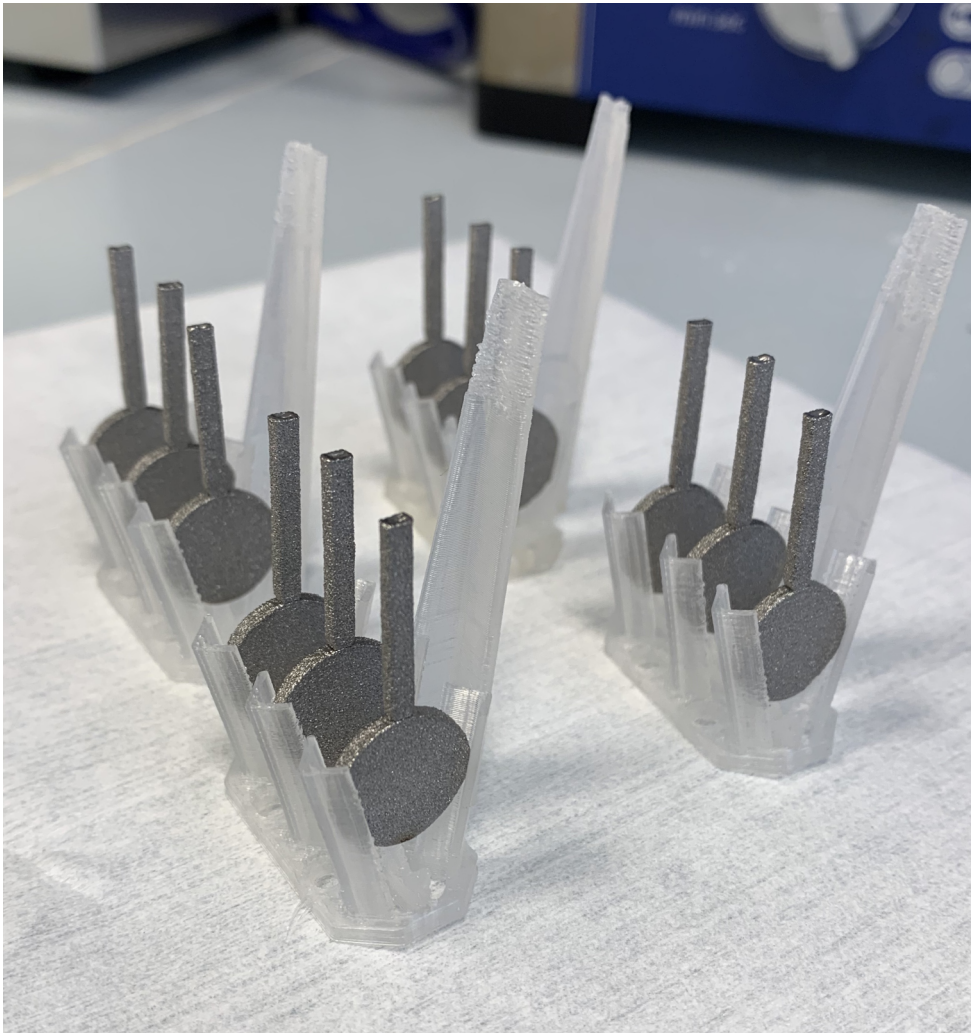


Figure 3.11: The specimen in the PP holder used during ultrasonic bath and CE.

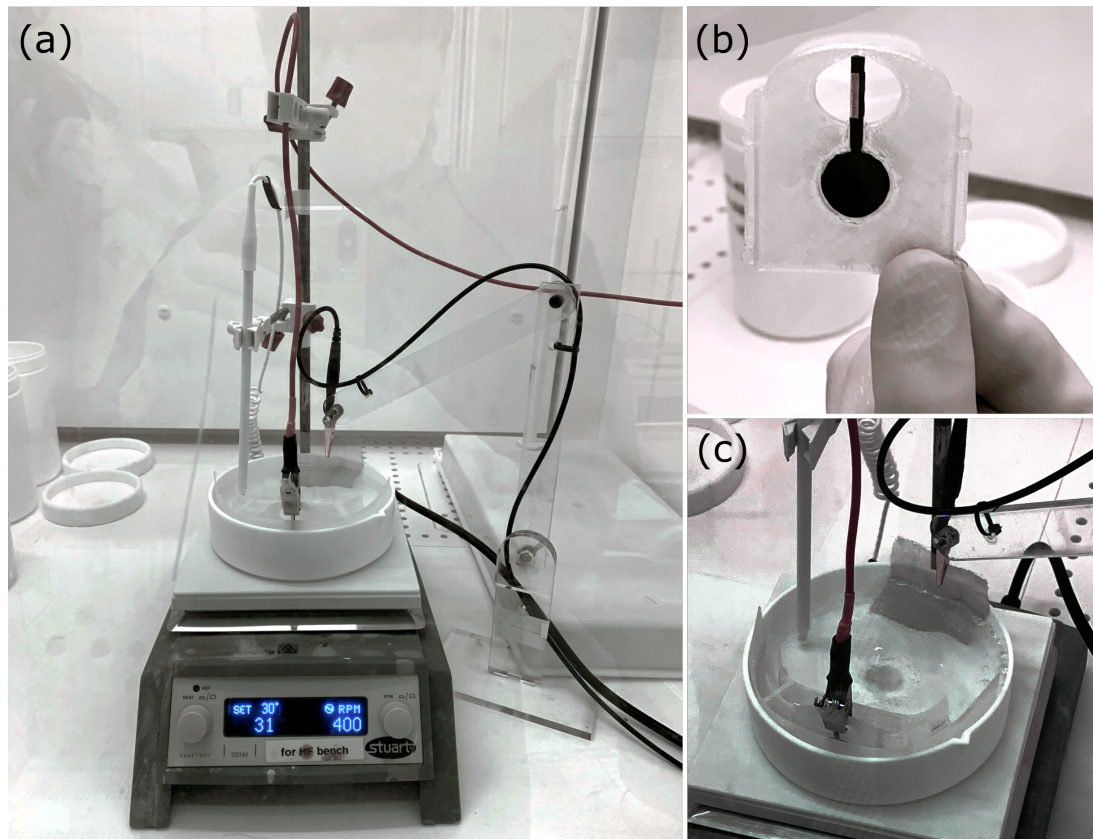


Figure 3.12: The electrochemical polishing setup in fume hood. Heat plate set to 30 °C and magnetic stirrer at 400 rpm.

increasing linealy stepwise up to 2 mA mm^{-2} in 30 steps, and then going linealy down again, with 5 at each step. For the proceedings, the specimen were polished for 5, 10 and 15, with a current density of 0.1, 0.3, 0.75, 1.25 and 1.8 mA mm^{-2} .

3.2.5 Characterization

μ CT

A SkyScan 1176 - Bruker Biospin was used to conduct μ CT imaging of the fatigue specimen, to analyse the as built geometry. A 1 mm aluminum filter, was used and an Voxel Size of $12.6 \mu\text{m}$. Voltage set to 90 kV, current $270 \mu\text{A}$. During the scan, the instrument had a break down, causing very limited imaging output. The instrument has remained out of service up to this date.

SEM

Field Emission scanning electron microscopy (SEM) was done with FEI Apreo at NanoLab, as it allows to take high resolution images of the samples. All samples had been done CE and ECP were rinsed with deionized water and thoroughly dried before assembled to the SEM sample holder. To ensure proper electrical contact between sample and sample holder, strips

of copper tape was used. The Everhart-Thornley detector (ETD) was used to capture images in both secondary electrons (SE) and back-scatter electrons (BSE) mode, to capture different features of the specimen surface.

Chapter 4

Results and Discussion

4.1 Scaffold Design

4.1.1 Volume Fraction Relations for Different TPMS

The relationship between volume fraction and isovalue from the functions given in Table 3.1 is seen in Figure 4.1. The polynomial fitting coefficients fulfilling 3.1 are given in Figure 4.1. This relation is integrated into a function, *volFrac2isoVal()*, which evaluates the isovalue for a given volume fraction. This lays the foundation for several other analyses, and is essential for designing structures with gradual features.

Table 4.1: Comparison of percentages.

Lattice	Polynomial coefficients, p_n					
	c_1	c_2	c_3	c_4	c_5	c_6
P	51.3842	-128.4607	123.0141	-56.0604	15.5747	-2.7260
PS	1.0276	2.2172	-3.4121	1.3505	1.5550	0.0101
D	1.1523	-2.8808	2.3529	-0.6486	2.4123	-1.1941
DS	5.7753	-12.6840	9.9903	-3.4396	1.6977	-0.0171
G1	-0.6243	1.5609	-2.1752	1.7019	2.4495	-1.4564
G2	0.2445	-0.6398	0.7697	-0.6379	-1.1828	1.4535
GS	-0.2150	0.5041	-0.5772	0.2416	1.4936	0.0067
IWP1	7.1494	-16.0623	11.8583	-1.6612	2.3440	-1.5719
IWP2	22.1399	-31.4971	15.1787	-1.3372	2.5652	-2.0358
OCTO	184.2183	-384.2048	325.6822	-135.7217	33.3160	-6.7704

4.1.2 Example of Volume Fraction Gradients

Using the GUI a number of scaffolds have been made. The workflow is illustrated in figure 4.2. The computation time for various parts of the code can be seen in figure 4.3. It is noticeable that various post processing, such as the mesh cleaning with removal of unattached

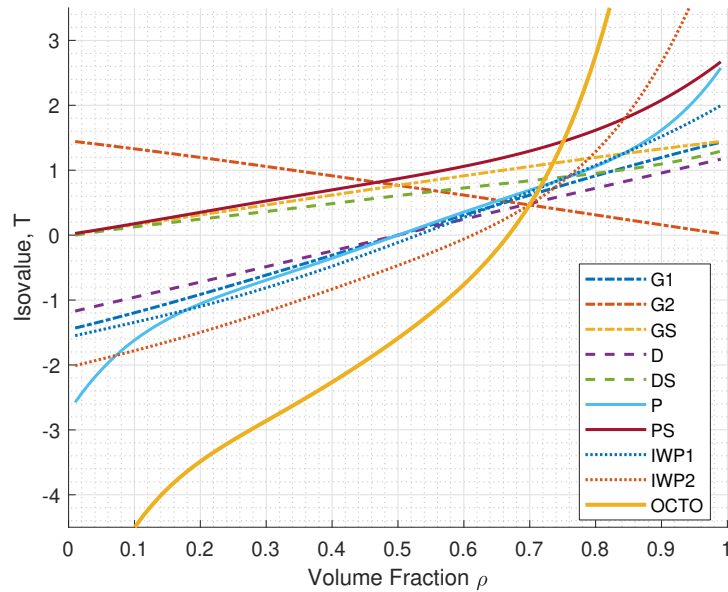


Figure 4.1: The volume fraction, ρ for different TPMS unit cells are evaluated for varying isovalues, T . The data is fitted with a 5th degree polynomial for the data points satisfying $0.05 < \rho < 0.95$.

patches and surface smoothing, takes up for big amounts of time spent. The method shows to work well, and be robust for a variety of designs, unlike unlike issues with boolean operations on complex surfaces (Yoo, 2013).

4.1.3 Pore Size

The pore analysing script was run for the different structures, and the values fitted to polynomial functions, which are shown in Figure 4.4 and Figure 4.5. The values from the function fitting is found in Table 4.1.3. This value can be used to scale the unit cell to the right dimension. The different topologies vary with the minimum and maximum values of the pore size. A relation like this is needed in order to be able to scale up the unit cell for a given topology to achieve the right pore size. To scale a unit cell to achieve a given pore size, the inverse of the relative pore size is made. How this relates to the volume fraction is found in 4.6. The gyroid based topologies, G1, G2, GS have nearly identical pore size throughout the structure for a given volume fraction, where the variation for the primitive based structures, P and PS, are much bigger. By defining the pore size aspect ratio, one can relate the smallest and biggest pore size within a structure. This is also shown in 4.6. The pore size aspect ratio, gives a measure for how much the pore size is varying, which could possibly have significant consequences for the way cells will grow in a scaffold.

4.1.4 Stiffness Matrix

Deriving the Lamé constants, which is necessary to assemble the stiffness matrix, proved to be more difficult than first expected. Figure 3.5 shows that even with the biggest RVE, the

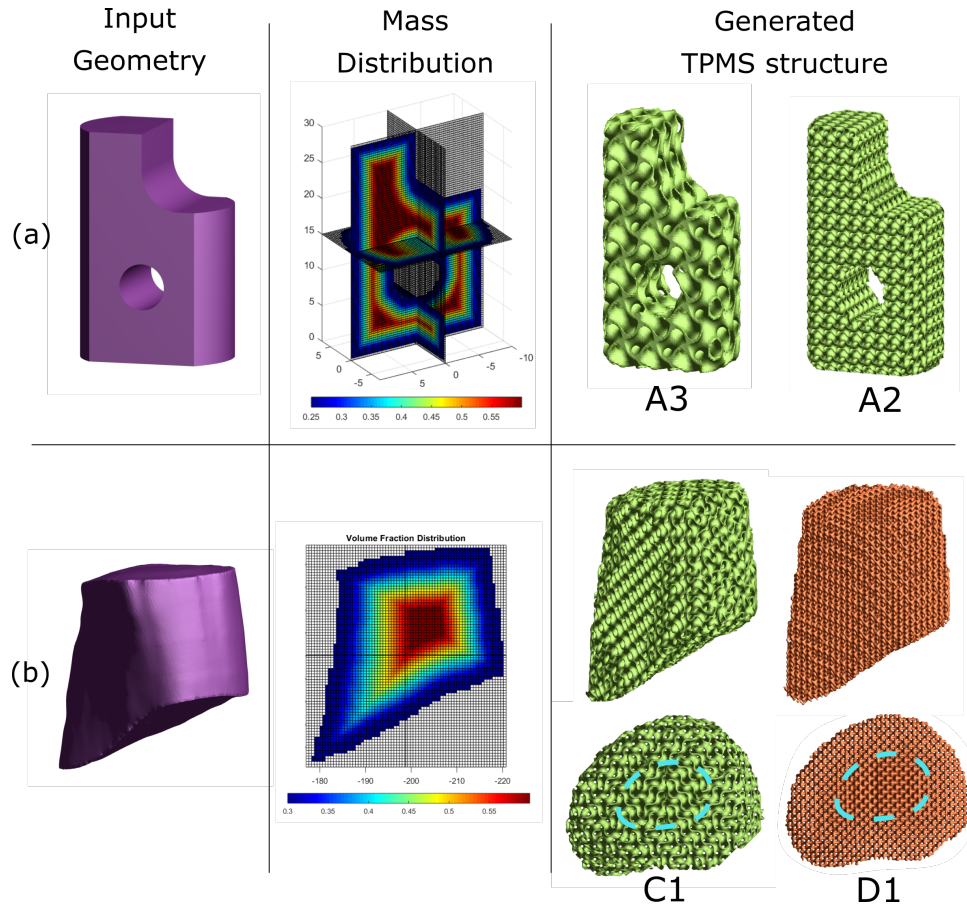


Figure 4.2: Two different input geometries are used (a) and (b) for demonstration. With the given settings, they will give a volume fraction distribution according to equation 3.2. Using different settings, has changes the generated TPMS structure. Top view images of C1 and D1 have highlighted the central part, to show that the denser structure in the core can be seen when viewed from the right angle.

relative shear modulus G^* , was not fully converged. Although it shows tendencies to converging, it is very probable that it would keep getting lower as the size of the RVE increases. Keeping in mind the Voight boundary, which is typically used for composite materials where the fibers are oriented in the preferred direction, indicates upper bound stiffness. The upper bound stiffness would for a two phase material, where the second phase has no stiffness (such as air), can be written as:

$$G_{Voight} = \rho * G_B = \rho \frac{E_B}{2(1 + \nu_B)} \quad (4.1)$$

Where G_B, E_B, ν_B is the shear modulus, Youngs Modulus and Poisson Ratio of the bulk material. Inserting for the values of Ti-6Al-4V, we have

$$G_{Voight30\%} = 0.30 \frac{113.8}{2(1 + 0.342)} = 12.72 GPa \quad (4.2)$$

The Voight boundary for for the bulk material of a relative stiffness of 1, is 0.3. Which is

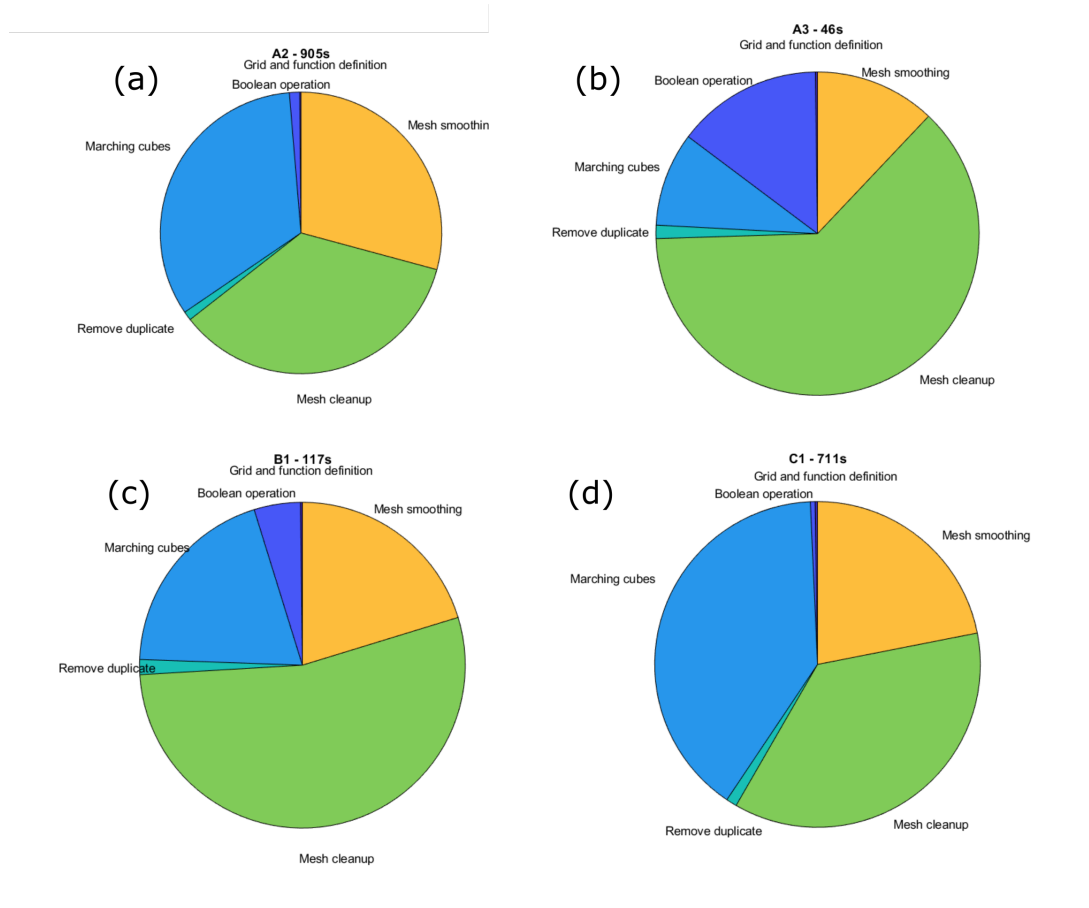


Figure 4.3: The time spent, for generating the different output structures is given beside the output sample name in seconds.

in the expected range. Unfortunately the methodology did not work well.

4.2 Surface Modification

4.2.1 μ CT Imaging

Based on the obtained μ CT image, there seems to be large amounts of residual powders in the scaffold. As seen in Figure 4.8. The cross section image was taken at $z \approx 11$ mm, which according to equation 3.2, gives a volume fraction of $\rho(11) = 0.421$. By the results of minimum pore size for D (4.1.3), the minimum relative pore size $d_{Dmin}(\rho = 0.421) = 0.3911$, giving a minimum pore size of 1.17 mm. With this pore size we would have expected the powder to be removed. Such residual powders would have to be removed before they could be utilized for tissue engineering applications. Possibly, a procedure consisting of etching, proceeded by thorough rinsing with ultrasonic bath, could remove significant amounts of the powder, but this has not yet been tested.

From the same image, darker regions are visible within the middle of the beam sections, indicating void sections within the solid material. Such large amounts of not fully melted cross

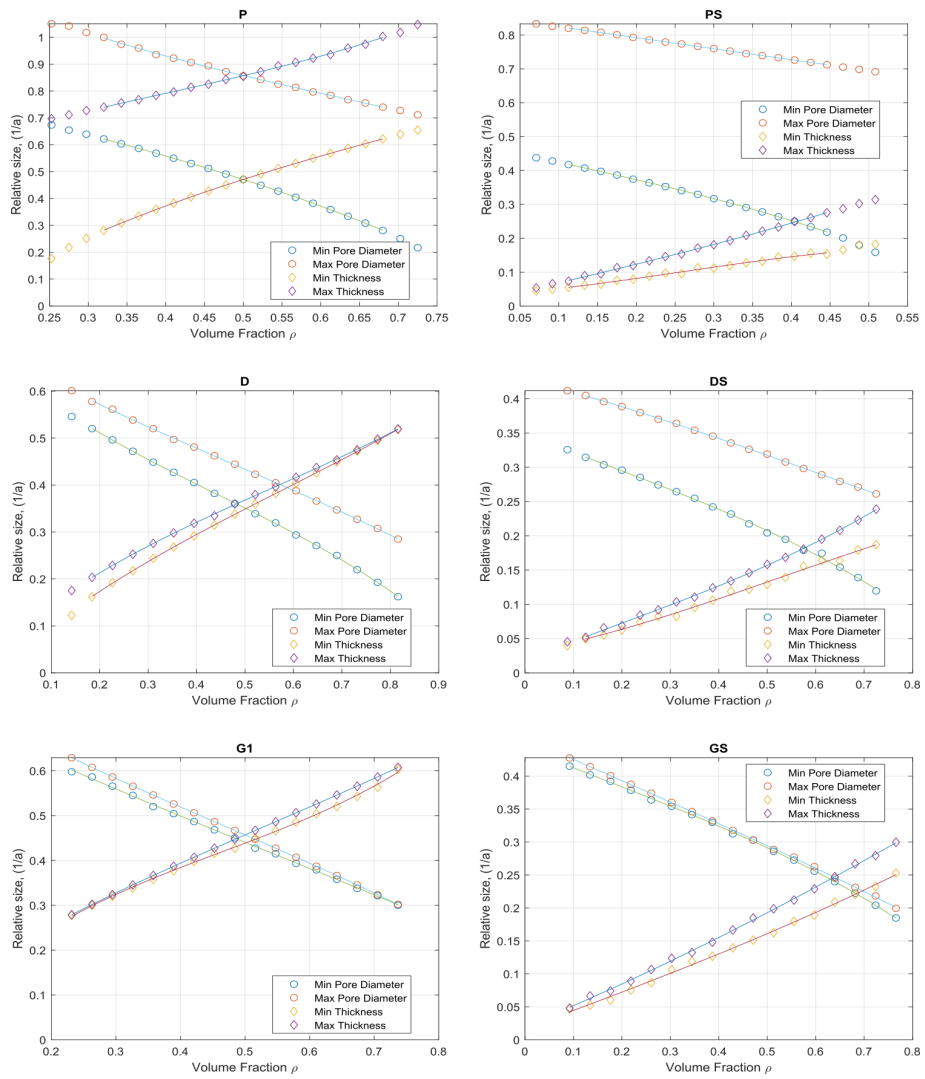


Figure 4.4: Relative pore size and relative section thickness for P, PS, D, DS, G1, GS.

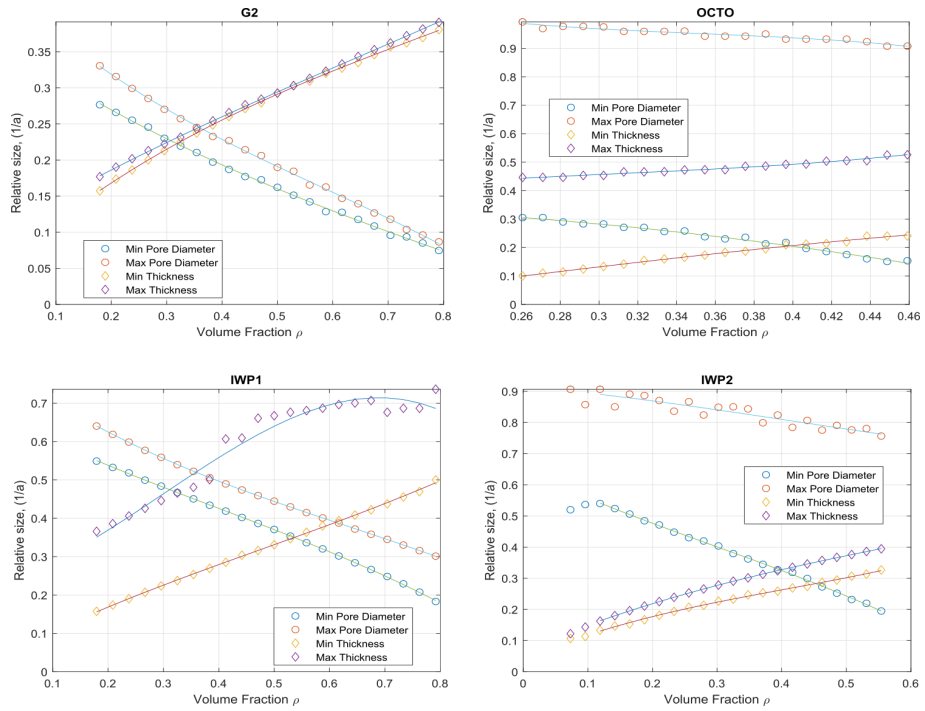


Figure 4.5: Relative pore size and relative section thickness for G2, OCTO, IWP1 and IWP2.

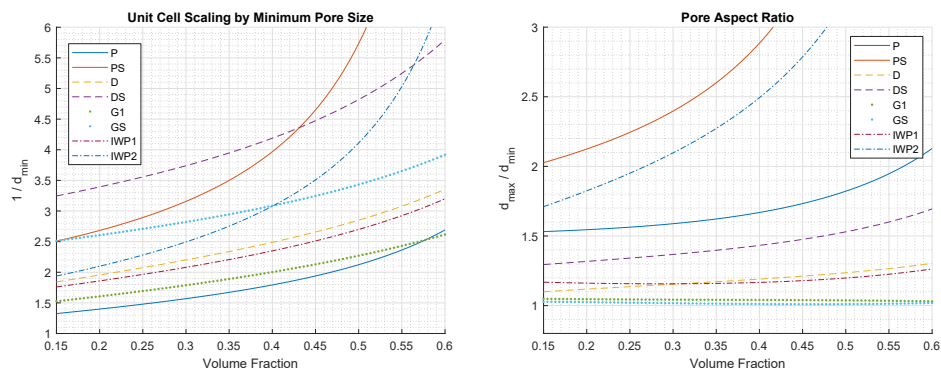


Figure 4.6: Left: The inverse of the minimal pore size w.r.t volume fraction. The inverse of the relative pore size, is the necessary side length of a unit cell to achieve a "unit sphere" pore. Right: The aspect ratio of the biggest and smallest pores size for a selection of different TPMS structures w.r.t. volume fraction.

TPMS	min	c_1	c_2	c_3	c_4	max	c_1	c_2	c_3	c_4
P	p_{min}	-0.859	0.667	-0.945	0.884	t_{min}	1.158	-2.378	2.423	-0.291
	p_{max}	-0.388	0.888	-1.292	1.330	t_{max}	1.213	-1.395	1.166	0.471
PS	p_{min}	-0.690	0.185	-0.522	0.475	t_{min}	-1.066	0.833	0.119	0.033
	p_{max}	0.075	-0.049	-0.317	0.857	t_{max}	0.279	-0.015	0.537	0.014
D	p_{min}	-0.537	0.716	-0.827	0.652	t_{min}	0.338	-0.589	0.865	0.021
	p_{max}	-0.188	0.270	-0.576	0.678	t_{max}	0.392	-0.654	0.821	0.072
DS	p_{min}	-0.184	0.095	-0.285	0.349	t_{min}	-0.144	0.227	0.127	0.030
	p_{max}	0.058	-0.110	-0.181	0.428	t_{max}	0.179	-0.101	0.282	0.019
G1	p_{min}	-0.195	0.314	-0.750	0.761	t_{min}	1.476	-2.051	1.489	0.022
	p_{max}	-0.399	0.559	-0.882	0.808	t_{max}	0.392	-0.632	0.961	0.086
G2	p_{min}	0.101	-0.039	-0.364	0.337	t_{min}	0.237	-0.522	0.680	0.052
	p_{max}	-0.171	0.331	-0.580	0.421	t_{max}	0.145	-0.268	0.491	0.098
GS	p_{min}	-0.185	0.106	-0.309	0.443	t_{min}	0.043	0.010	0.272	0.017
	p_{max}	-0.045	0.023	-0.327	0.458	t_{max}	-0.011	0.086	0.304	0.020
IWP1	p_{min}	-0.404	0.474	-0.733	0.669	t_{min}	0.342	-0.491	0.752	0.035
	p_{max}	-0.481	0.908	-1.051	0.803	t_{max}	-2.094	1.866	0.413	0.229
IWP2	p_{min}	-0.878	0.775	-0.978	0.648	t_{min}	0.990	-1.222	0.885	0.041
	p_{max}	0.264	-0.332	-0.172	0.915	t_{max}	0.409	-0.771	0.893	0.068
OCTO	p_{min}	0.542	-2.022	0.432	0.320	t_{min}	-1.382	0.836	0.675	-0.109
	p_{max}	-11.967	12.701	-4.778	1.582	t_{max}	6.102	-5.695	2.085	0.179

sections, would be expected to give significantly poorer structural integrity of the specimen. This issue is caused by manufacturing, not by design. It turned out that for very small cross sections, only the perimeter would be melted, and that for cross sections larger than 0.6 mm, there will remain unmelted powder. The maximum geometrical thickness at this cross section should, according to the above-mentioned calculations, would be 0.99 mm. Both of these findings are undesirable for the sample. Based on the undesirable state of the samples, further testing was not proceeded on the samples.

4.3 Chemical Etching and Electrochemical Polishing

For the experiments with sweeping current density, the measured voltage is plotted against the outgoing current. When interpreting the measured data it is seen that on the downwards sweep, the current is lower than in the upgoing sweep for the voltage for large amounts of the range. If the electrochemical system used has similar characteristics as what is illustrated in 1.5, means that the voltage is not unicyclically defined all for all current densities.

The weight loss, coming from the chemical etching, can be seen in table 4.10. The average weight loss from CE is 0.0184 g with a standard deviation of 0.003 g. The weight reduction is larger with higher current density, and with longer polishing time.

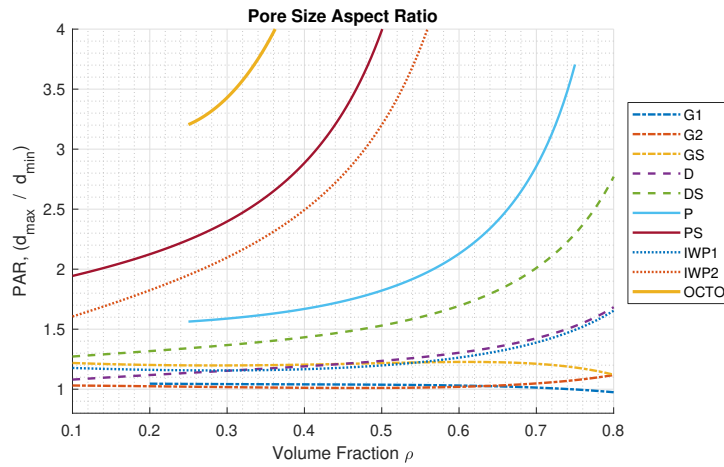


Figure 4.7: The pore size aspect ratio, as the ratio between the biggest and smallest pore for different TPMS structures.

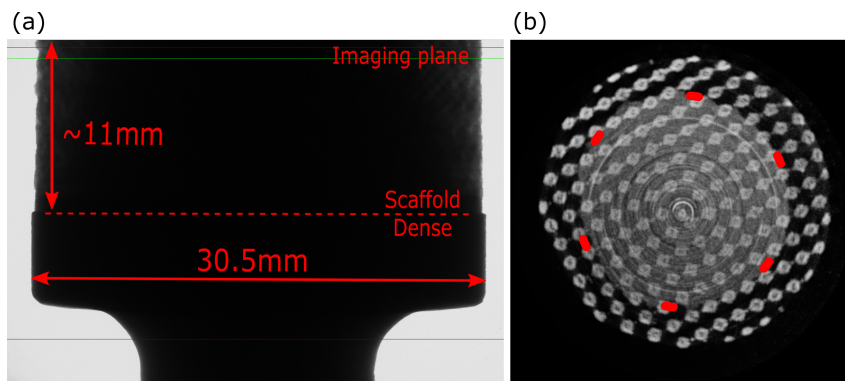


Figure 4.8: μ CT images taken of the Ti-6Al-4V fatigue specimen. (a): Shows progression in μ CT scan. (b): Cross sectional μ CT image taken from the imaging plane given in (a). Lighting is edited to highlight the features of the image. Red dashed line is added in (b) to indicate region of scaffold with residual unmelted powders.

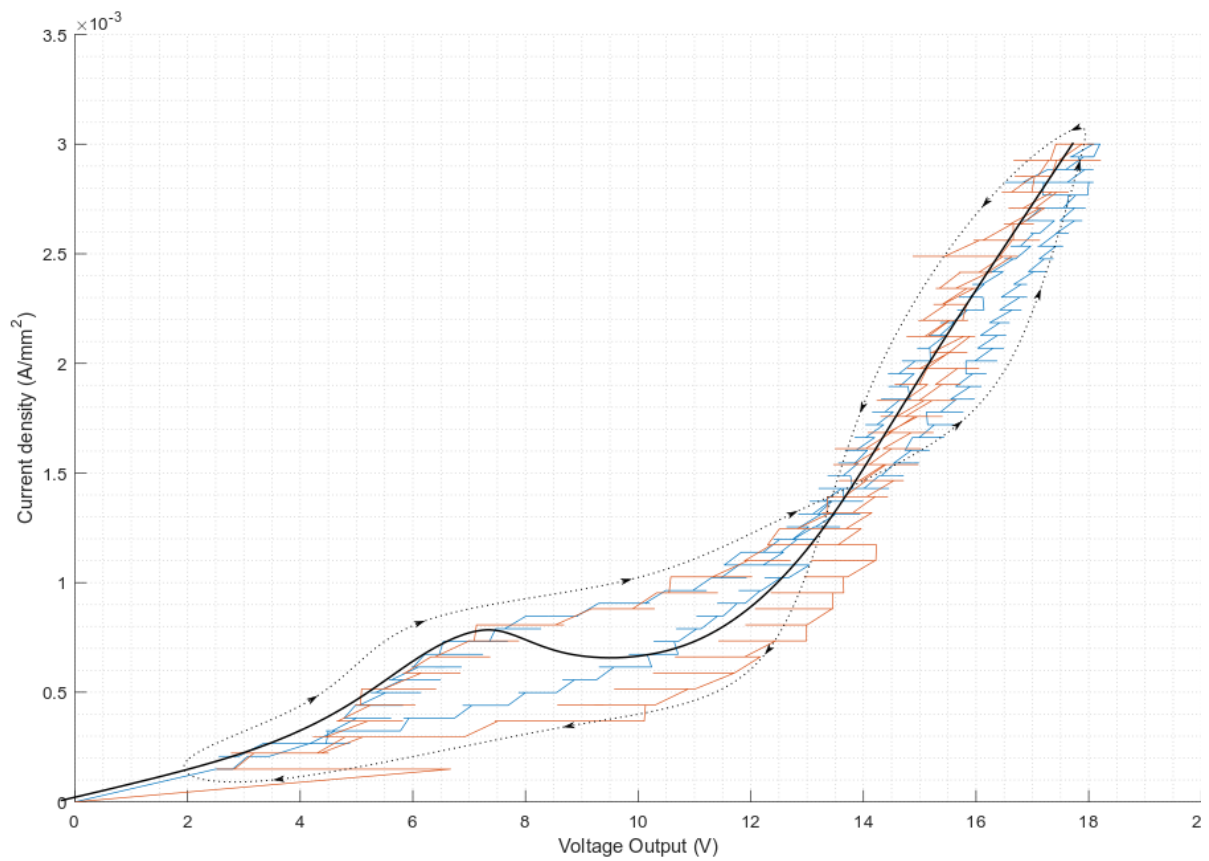


Figure 4.9: Plotted current density versus voltage. The orange and blue lines are subsequent sweeps. The dotted line with arrows is to indicate which path the curve is taking with time. The black curve is a free hand drawing, and interpolating of the possible current density value that would be obtained with a potentiostat.

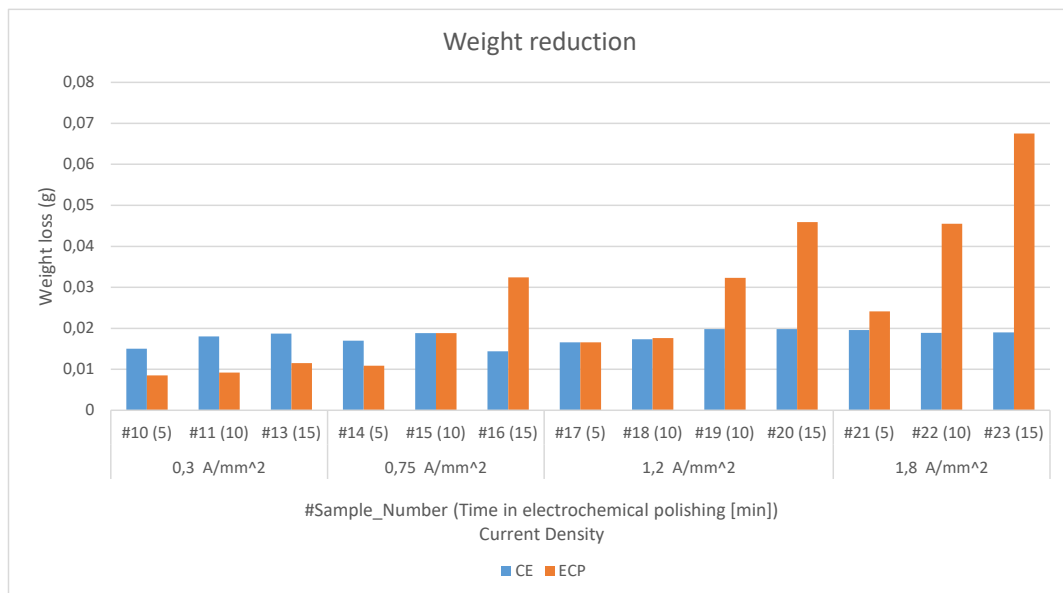


Figure 4.10: The weight loss from the chemical etching and electrochemical polishing.

SEM images of the ECP samples show various features, as seen in 4.11 and 4.12. Most distinguishable is the surface topography. Samples ECP9, ECP13, and ECP16 show very clearly partially melted metal powders on the sample surface. When comparing image (d) and (f) in 4.11, which are taken using the same stage, beam and detector settings, one can see a clear difference. While the powders particles in (d) shows apparent spherical appearance, the powders of (f) seems to be flattened at the top, indicating that the etching rate is higher with a greater height of the surface, which could indicate the presence of an anolyte layer during the polishing. Sample ECP20 has nearly all powders removed, as can be seen from Figure 4.12. The surface is yet not completely smooth. Sample ECP23, which was polished with the highest current density, could be seen to have a spotted surface seen with the naked eye. Upon inspecting the different regions of the surface, there are major differences, as can be seen in image (d) and (e) in 4.12. This could be a result of the high current density caused gas formation and bubbles sticking to the specimen surface, which again caused uneven polishing.

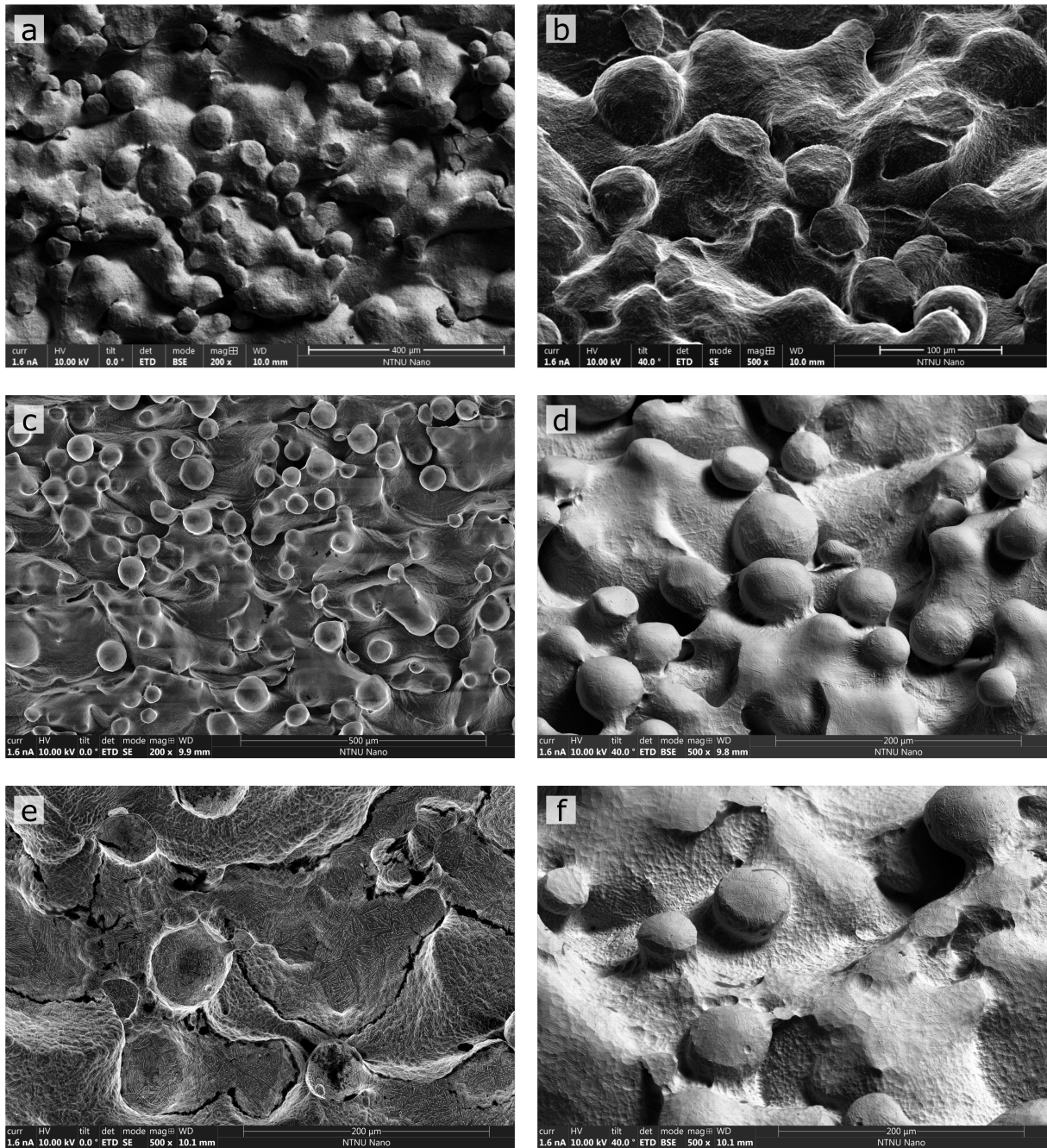


Figure 4.11: SEM images of samples that were polished for 15 minutes. (a),(b): ECP9, polished at 0.1 mAmm^{-2} . (c),(d): ECP13, polished at 0.3 mAmm^{-2} . (e),(f): ECP16, polished at 0.75 mAmm^{-2} .

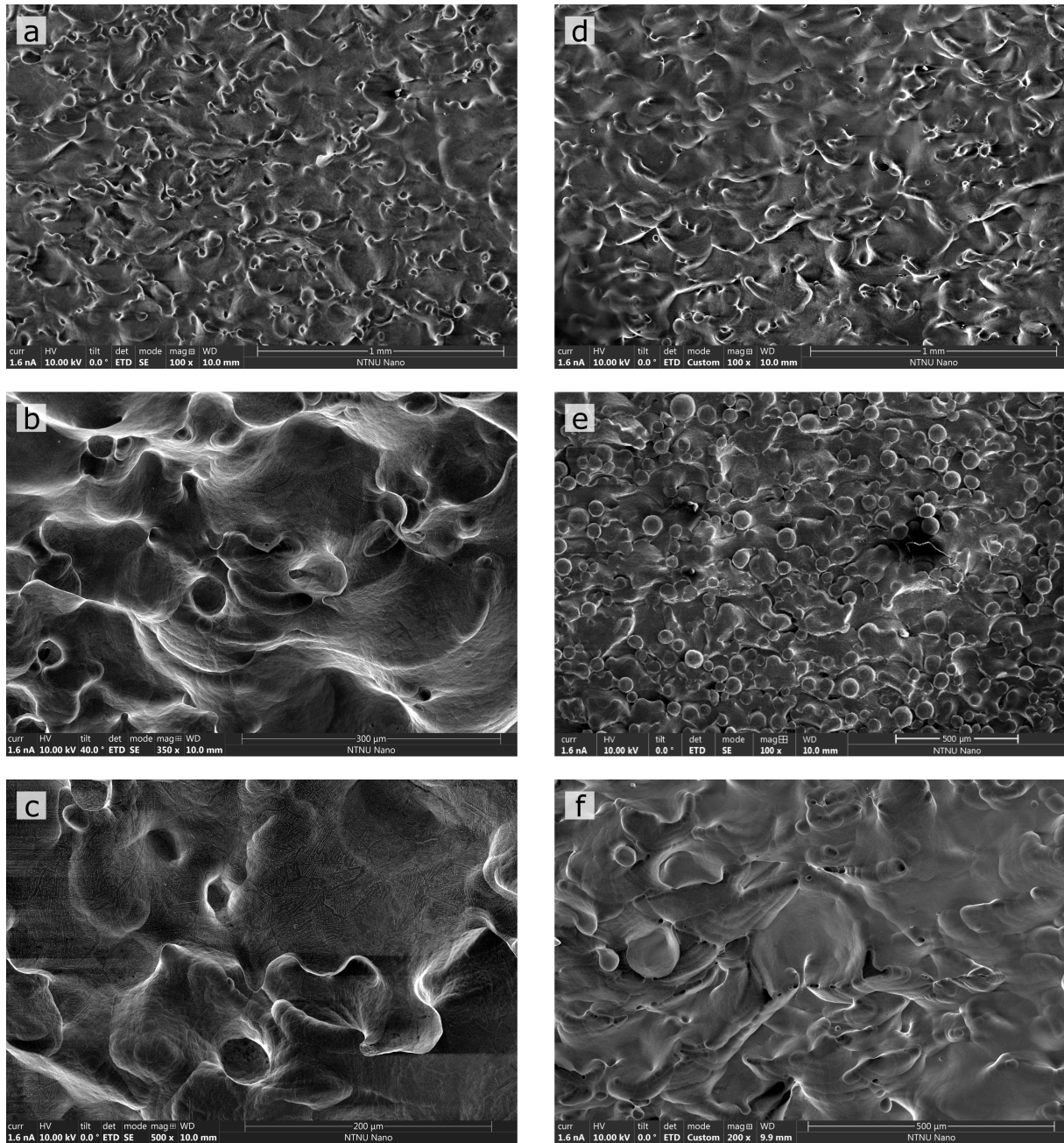


Figure 4.12: SEM images of samples that were polished for 15 minutes. (a),(b),(c): ECP20, polished at 1.25 mAmm^{-2} . (d),(e),(f): ECP23, polished at 1.8 mAmm^{-2} .

Chapter 5

Conclusions

In this final chapter you should sum up what you have done and which results you have got. You should also discuss your findings, and give recommendations for further work.

The biggest effort during the thesis work has been in gaining insight in scaffold design using TPMS structures and how different features can be controlled in the design. A valuable result is the relationship between the volume fraction of the cellular material and the isovalue of the TPMS function, which is the first step in gaining control of the design. This is quite straight forward, and has been done by others before, however, the data explaining the fit, or other formula is not given. The presented fitting coefficients can be used by others for a quicker implementation and extension of the presented functionalities.

Developing a material-datasheet for different TPMS structures was one of the biggest goals for this thesis. The chosen strategy involved designing a complex lattice structure with a surface mesh, and to import this into a commercial FEM solver. Using a FEM solver that could be integrated with Matlab, where the design of the structures were made, was desired, but not found. Going through a number of steps, in order to recognize specific features of the structure, a third party plugin was used to construct the desired boundary conditions. This had to be done for a variety of versions of the same lattice. Despite efforts to automatize the calculations with Python scripts, this proved to be time-consuming, and computationally non-efficient. Not only was this inefficient, the results were not accurate either. They lacked both a clear convergence, and showed stiffness results that were improbable. Even if the quality of the results had been better, the whole method would have been so slow and difficult to use. Despite putting big efforts into optimizing the routine, this objective was abandoned due to poor results.

A few months later, after this objective had been put aside, new insight was gained and ideas regarding ways to solve the aforementioned issues. The GIBBOn Code, which is an open-source MATLAB toolbox, includes tools for image and geometry processing and visualization, and allows tetrahedral meshing and FEM simulations through the interface with the free and open-source software TetGen Si (2015) and FEBio Maas et al. (2012) FEBio has the inbuilt functionality for periodic boundary conditions. This way, all steps, from design and RVE assembly, to FEM analysis can be done in one workflow within MATLAB. At the time

this tool was found by the candidate, it was too late to invest efforts into solving the stiffness mapping objective using this tool. View *Further Works* for further extensions.

A TPMS scaffold design App using MATLAB. The main goal was to find a way of designing scaffolds with TPMS structure, that could fill the boundaries of a given design volume, and exploit the possibilities of variable volume fraction throughout the structure. This was achieved using imaging processing techniques. It was demonstrated that the program is working, but with some limitations. The imaging processing is time-consuming. If the image had been set to the same resolution as the gridded scalar field of the TPMS function, the computation time would be impractically high. However, by reducing the resolution of the image of the input geometry, performing image analysis, and then later increase the resolution, using linear and cubic interpolation, showed to be very efficient. The increased time performance has a cost in the accuracy of and smoothness of the output. Especially is that visible along the boundaries of the scaffold.

Apart from using MATLAB, there are other tools that could have been used. Rhino, which is a scripting based program for parametric design could have been an alternative. Grasshopper is an algorithmic modeling plugin for Rhino, making it easier to design even for people that are not used to scripting. However, this was not tried due to license costs.

Fatigue specimen with TPMS D lattice was designed using variable volume fraction. These specimen were produced in Ti-6Al-4V by EBM. Due to large amounts of residual powders in the lattice, and not fully melted beam cross sections, the specimen were not further used.

The electrochemical polishing made smoother surfaces on the EBM manufactured Ti-6Al-4V specimen. The current density of 1.25 mA mm^{-2} showed the best results according to the SEM images.

Chapter 6

Reccomandations for Further Works

Throughout this thesis work I have greatly expanded my knowledge and skills within programming, geometrical modeling, additive manufacturing and cleanroom work, which exceeded my expectations. Throughout the work I have discovered several topics that would be interesting to further investigate, something I am lucky to have the possibility to do. Because the field of study was so interesting, several subjects were investigated throughout the course of the thesis work, making a very educating and exciting year. However, a consequence of diving into several topics is that all objectives might not be completed as planned. Sometimes, unexpected results would push the work in a new direction, and random discoveries would unveil new methods of solving a problem. This has led me to think of many useful and interesting topics to further investigate.

A big challenge when designing a part that is built up with local anisotropy, is to determine how the material should be oriented to optimize performance. As demonstrated in this thesis, a volume has been filled with a periodicity where the lattice coordinates are mapped one-to-one to spatial coordinates. Simple lattice coordinate transformations are possible, such as stretching the axis in one direction, or making a cylindrical coordinate system for the lattice. However, there are limitations and drawbacks for treating singularities, such as in the latter example, where there will be a problem in defining the lattice in the origin. A brilliant way to deal with this is inspired from the finite element method, where isoparametric functions, typically linear or quadratic, can be used to map the lattice coordinates of a cubic unit cell to the spacial coordinates of the arbitrarily shaped hexahedron element. Hexahedral meshing is a field of research in itself, and it would be interesting to look into using newly developed meshing algorithms, such as the singularity constrained optimization algorithm, or using other meshing techniques to achieve special lattice designs.

Another topic which would be interesting to pursue, is to combine lattice design with topology optimization. Shown by (reference) the "lims" method can be modified, to reduce the panalization such that the optimal solution does not end up with a binary value (bulk material or no material), but will allow continuously varying volume fraction, which can easily be passed to the lattice function to determine the local porosity of the final structure.

Further improvement can be made to this work, especially in improving the use of the stiffness matrix, where as in their work one single stiffness matrix is used throughout the whole computation. A weakness in the method lies in the assumptions made on the stiffness matrix. The authors have not taken into account that the TPMS lattice will have an orthotropic behaviour and that the stiffness properties will vary greatly with the volume fraction. I would suggest obtaining an orthotropic stiffness matrix, being a function of the volume fraction, as described in the thesis, and using this to make a better optimization. Possibly by iteratively updating the stiffness matrix throughout the optimization based on the density distribution. To be able to perform this, it is necessary to assemble the stiffness matrix for different lattice topologies with different volume fractions. Unlike the method tried in this thesis, a more automatic setup should be used, which is easier to integrate with the Scaffold script. Towards the end of the thesis work, two months after the attempt on making the stiffness matrix was put on the shelf, the GIBBON Code project, which stands for Geometry and Image-Based Bioengineering add-ON. It is a MATLAB toolbox which has built in features to enable surface meshing, volumetric meshing and finite element analysis. When some, or all of these objectives are met, many exciting results might come about. Maybe a combination of these features could be assembled into a plug-in that could be used together with more well known CAD tools, such as SolidWorks, AutoCad or Siemens NX, to make a diverse and functional design tool.

6.1 Removal of powders

Effors should be made in order to remove the residual powders from the manufactured fatigue specimen. This could possibly be done by etching and ultrasonic batch treatments.

Bibliography

- Ashby, M. F. and Gibson, L. J. (1997). Cellular solids: structure and properties. *Press Syndicate of the University of Cambridge, Cambridge, UK*, page 183e231.
- Bade, N. D., Kamien, R. D., Assoian, R. K., and Stebe, K. J. (2017). Curvature and Rho activation differentially control the alignment of cells and stress fibers. *Science advances*, 3(9):e1700150.
- Bade, N. D., Xu, T., Kamien, R. D., Assoian, R. K., and Stebe, K. J. (2018). Gaussian Curvature Directs Stress Fiber Orientation and Cell Migration. *Biophysical journal*, 114(6):1467–1476.
- Chen, J.-H., Liu, C., You, L., and Simmons, C. A. (2010). Boning up on Wolff’s Law: Mechanical regulation of the cells that make and maintain bone. *Journal of Biomechanics*, 43(1):108–118.
- Christen, P., Ito, K., Galis, F., and B. Rietbergen, v. (2015). Determination of hip-joint loading patterns of living and extinct mammals using an inverse Wolff’s law approach. *Biomechanics and Modeling in Mechanobiology*, 14(2):427–432.
- Gouveia, R. M., Koudouna, E., Jester, J., Figueiredo, F., and Connon, C. J. (2017). Template Curvature Influences Cell Alignment to Create Improved Human Corneal Tissue Equivalents. *Advanced Biosystems*, 1(12):1700135.
- Lee, T., Kashyap, R., and Chu, C. (1994). Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6):462–478.
- Lejeunes, S. and Bourgeois, S. Une Toolbox Abaqus pour le calcul de propriétés effectives de milieux hétérogènes. Technical report.
- Liang, D. and Wadley, H. (2016). Shear response of carbon fiber composite octet-truss lattice structures. *COMPOSITES PART A*, 81:182–192.
- Lietaert, K., Cutolo, A., Boey, D., and Van Hooreweder, B. (2018). Fatigue life of additively manufactured Ti6Al4V scaffolds under tension-tension, tension-compression and compression-compression fatigue load. *Scientific Reports*, 8(1):1–9.
- Maas, S. A., Ellis, B. J., Ateshian, G. A., and Weiss, J. A. (2012). FEBio: finite elements for biomechanics. *Journal of biomechanical engineering*, 134(1):011005.

- Maurer, C., Rensheng Qi, and Raghavan, V. (2003). A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):265–270.
- Nishikawa, Y., Koga, T., Hashimoto, T., and Jinnai, H. (2001). Measurements of Interfacial Curvatures of Bicontinuous Structure from Three-Dimensional Digital Images. 2. A Sectioning and Fitting Method.
- Si, H. (2015). TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator. *ACM Trans. Math. Softw.*, 41(11).
- Vetsch, J. R., Müller, R., and Hofmann, S. (2016). The influence of curvature on three-dimensional mineralized matrix formation under static and perfused conditions: an in vitro bioreactor model. *Journal of the Royal Society, Interface*, 13(123).
- Yoo, D. (2013). New paradigms in hierarchical porous scaffold design for tissue engineering. *Materials Science and Engineering: C*, 33(3):1759–1772.
- Zhang, C. and Chen, T. EFFICIENT FEATURE EXTRACTION FOR 2D/3D OBJECTS IN MESH REPRESENTATION. Technical report.

Appendix A

Code

For the master thesis work, much coding has been performed, most of it being performed in MATLAB. Some of the functions and scripts are concerning, the main operation of the scaffold generation, whilst some scripts are used for measuring purposes. The scripts used for controlling and monitoring of the power source for ECP has also been included.

A.1 Scaffold Coding

A.1.1 Function Definition

```
1 function fun = latticeFun(grid, size, phaseShift, LatticeType, T)
2 % LATTICEFUN Takes in a grid, size and phase shift and latticetype and
   iso
3 % value, and gives back the evaluated function, fun, being the same
   size as
4 % grid. Later use latticeSurf() to obtain an isosurface from fun.
5 %
6 X = grid.X + phaseShift(1);
7 Y = grid.Y + phaseShift(2);
8 Z = grid.Z + phaseShift(3);
9 % Parameters for grid size in x,y,z directions
10 a = size(1);
11 b = size(2);
12 c = size(3);
13 % Choosing lattice function geometry.
14 switch LatticeType
15     case 'G1'
16         fun = -(sin(2*pi*X/a).*cos(2*pi*Y/b) + ...
17             sin(2*pi*Y/b).*cos(2*pi*Z/c) + ...
18             sin(2*pi*Z/c).*cos(2*pi*X/a) - T);
```

```

19 case 'GS'
20     fun = -((sin(2*pi*X/a) .* cos(2*pi*Y/b) + ...
21             sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
22             sin(2*pi*Z/c) .* cos(2*pi*X/a)) ...
23             .* (sin(2*pi*X/a) .* cos(2*pi*Y/b) + ...
24                sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
25                sin(2*pi*Z/c) .* cos(2*pi*X/a))- T.^2 );
26 case 'G2'
27     fun = +((sin(2*pi*X/a) .* cos(2*pi*Y/b) + ...
28             sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
29             sin(2*pi*Z/c) .* cos(2*pi*X/a)) ...
30             .* (sin(2*pi*X/a) .* cos(2*pi*Y/b) + ...
31                sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
32                sin(2*pi*Z/c) .* cos(2*pi*X/a))- T.^2 );
33 case 'D'
34     fun = -(sin(2*pi*X/a) .* sin(2*pi*Y/b) .* sin(2*pi*Z/c) + ...
35            sin(2*pi*X/a) .* cos(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
36            cos(2*pi*X/a) .* sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
37            cos(2*pi*X/a) .* cos(2*pi*Y/b) .* sin(2*pi*Z/c) - T);
38 case 'DS'
39     fun = -((sin(2*pi*X/a) .* sin(2*pi*Y/b) .* sin(2*pi*Z/c) + ...
40             sin(2*pi*X/a) .* cos(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
41             cos(2*pi*X/a) .* sin(2*pi*Y/b) .* cos(2*pi*Z/c) + ...
42             cos(2*pi*X/a) .* cos(2*pi*Y/b) .* sin(2*pi*Z/c)).^2 - T.^2);
43 case 'IWPI'
44     fun = (( cos(2*X*pi/a) .* cos(2*Y*pi/b) + ...
45             cos(2*Y*pi/b) .* cos(2*Z*pi/c) + ...
46             cos(2*Z*pi/c) .* cos(2*X*pi/a) ) ...
47            - 0.5*( cos(4*X*pi/a) + cos(4*Y*pi/b) + cos(4*Z*pi/c) ) + T
48            );
49 case 'IWPS'
50     fun = -((( cos(2*X*pi/a) .* cos(2*Y*pi/b) + ...
51               cos(2*Y*pi/b) .* cos(2*Z*pi/c) + ...
52               cos(2*Z*pi/c) .* cos(2*X*pi/a) ) ...
53            - 0.5*( cos(4*X*pi/a) + cos(4*Y*pi/b) + cos(4*Z*pi/c) ))).^2
54            - T.^2);
55 case 'IWP2'
56     fun = -(2*( cos(2*X*pi/a) .* cos(2*Y*pi/b) + ...
57               cos(2*Y*pi/b) .* cos(2*Z*pi/c) + ...
58               cos(2*Z*pi/c) .* cos(2*X*pi/a)) - T);

```

```

57     case 'P'
58         fun = -(cos(2*X*pi/a) + cos(2*Y*pi/b) + cos(2*Z*pi/c) - T);
59     case 'PS'
60         fun = -( (cos(2*X*pi/a) + cos(2*Y*pi/b) + cos(2*Z*pi/c)).^2 - T
61                 .^2);
62     case 'OCTO'
63         fun = -(4*( cos(2*X*pi/a).*cos(2*Y*pi/b) + cos(2*Z*pi/c).*cos
64                   (2*Y*pi/b) + cos(2*X*pi/a).*cos(2*Z*pi/c) ) ...
65               -3*( cos(2*X*pi/a) + cos(2*Y*pi/b) + cos(2*Z*pi/c) ) - T)
66               ;
67     otherwise
68         disp('ERROR: Not a lattice structure')
69 end %Switch
70 end %Function

```

A.1.2 Measurement

Volume Fraction Calculation

```

1  function p = volFracCalc()
2  % volFracCalc() returns the coefficients for a polynomial fit of the
3  % given
4  % TPMS structure which is defined in the latticeFun() function.
5  %% Define problem
6  ps = [0 0 0]; % Can phase shift the unit cell. from 0 to 1 in the
7  % brackets
8  a = [1 1 1];
9  types = {'P', 'PS', 'D', 'DS', 'G1', 'G2', 'GS', 'IWP1', 'IWP2', 'OCTO'};
10 type = types{1}; % Change number to set to one of the above TPMS
11 % structures
12 res = [100]; % Resolution of the rectangular grid in each or three
13 % orthogonal directions.
14 tstart = -3; % Starting isovalue: Can be set outside of possible limit
15 tstop = 3; % Stopping isovalue: Can be set outside of possible limit
16 %% Run calculation
17 t = linspace(tstart, tstop, 30);
18 for i_res = 1:length(res) % The analysis can be performed at different
19     ...
20     %resolutions to check the sensitivity of the resolution of volfrac
21     .
22     x = linspace(0,1,res(i_res));
23     y = linspace(0,1,res(i_res));

```

```

18     z = linspace(0,1,res(i_res));
19     [grid.X,grid.Y,grid.Z] = ndgrid(x,y,z); % Defining the grid.
20     for i_t = 1:length(t)
21         j = j+1;
22         T = t(i_t)*ones(size(grid.X));
23         try % If the next function produce an error, it will be caught
24             and...
25             % continue to run. Therefore T-can be set outside possible
26             % limits.
27             fun = latticeFun(grid,a,ps,type,T); % Make the TPMS
28             function
29             [F,V] = latticeSurface(grid,fun); % creates a patch.
30             density(i_res,i_t) = stlVolume(V',F'); % This function
31             calculates
32             % the volume of the closed patch. Since the unit cell is
33             one a
34             % unit volume, the number value corresponds to the density.
35         catch
36             %warning('Not a valid t-value')
37             density(i_res,i_t) = NaN;
38         end
39     end % for
40 end % for
41
42 %% Plotting and interpretation
43 hold on
44 colInd = get(gca,'ColorOrderIndex');
45 plot(t,density,'d')
46 set(gca,'ColorOrderIndex',colInd)
47 t= t(~isnan(density));
48 density = density(~isnan(density));
49 mid = density<0.95 & density>0.05; % Only evaluating the data point
50     within the given limits.
51 t_mid=t(mid);
52 density_mid=density(mid);
53 [p,S] = polyfit(density_mid,t_mid,5); % POLYNOMIAL FIT
54 densVal = linspace(density(1),density(end),200);
55 [tVal,delta] = polyval(p,densVal,S);
56 hold on
57 plot(tVal,densVal,'-')

```



```

53 ylim ([0,1])
54 delMax = max(delta);
55 fprintf ('& %.4f & %.4f & %.4f & %.4f & %.4f & %.4f \\\ \n',p(1),p(2),p
(3),p(4),p(5),p(6));
56 end

```

Pore Size Calculation

```

1 function result = poreSize(vf, type, plotBol)
2 %%
3 n = [0.5 1 1];
4 a = [1 1 1];
5 % Phase shift
6 ps = [0 0 0]; % input a value from 0 to 1 in the brackets
7 res = 100;
8
9 x = linspace(-0.5*n(1),1.5*n(1)*a(1),2*n(1)*res);
10 y = x;
11 z = x;
12 [grid.X,grid.Y,grid.Z] = ndgrid(x,y,z);
13 [mX,mY,mZ] = meshgrid(x,y,z);
14 t = volFrac2isoVal(vf, type, 't');
15 T = t*ones(size(grid.X));
16 fun = latticeFun(grid,a,ps,type,T);
17
18 %% Finding skeleton and distance fields
19 BWsurface = fun==0;
20 BWlattice = fun<0;
21 %BWmatix = fun>0;
22
23 skel = bwskel(BWlattice, 'MinBranchLength',30);
24 voidSkel = bwskel(~BWlattice, 'MinBranchLength',30);
25 sz = size(skel);
26 lim1 = round(sz(1)/4)-1;
27 lim2 = round(sz(1)*3/4)+1;
28 skelCore = skel(lim1:lim2,lim1:lim2,lim1:lim2);
29 voidSkelCore = voidSkel(lim1:lim2,lim1:lim2,lim1:lim2);
30 D = 2*bwdist(~BWlattice)/res + 2/res;
31 D2 = 2*bwdist(BWlattice)/res + 2/res;
32 D = smooth3(D); D2 = smooth3(D2);
33 D_core = D(lim1:lim2,lim1:lim2,lim1:lim2);

```

```

34 D2_core = D2(lim1:lim2, lim1:lim2, lim1:lim2);
35 X = grid.X(lim1:lim2, lim1:lim2, lim1:lim2);
36 Y = grid.Y(lim1:lim2, lim1:lim2, lim1:lim2);
37 Z = grid.Z(lim1:lim2, lim1:lim2, lim1:lim2);
38
39 %% Plotting scatter and surface PLOTTING
40 if plotBol
41
42 hold on
43 S = scatter3(X(skelCore), Y(skelCore), Z(skelCore), 100, D_core(skelCore), '
      filled');
44 FV = isosurface(X, Y, Z, fun(lim1:lim2, lim1:lim2, lim1:lim2), 0);
45 p = patch(FV);
46 p.FaceColor = 'g';
47 p.EdgeColor = 'none';
48 alpha(p, 0.7)
49
50 FV2 = isocaps(X, Y, Z, fun(lim1:lim2, lim1:lim2, lim1:lim2), 0);
51 p2 = patch(FV2);
52 p2.FaceColor = 'g'; % [1 0.8 0];
53 p2.EdgeColor = 'none';
54 alpha(p2, 0.7)
55
56 end
57 %% Finding min/max pores
58 cX = X(skelCore);
59 cY = Y(skelCore);
60 cZ = Z(skelCore);
61 [sx, sy, sz] = sphere(60);
62 %% Minimum sphere
63 %D_core(isnan(D_core)) = 0;
64 [minVal, minId] = min(D_core(skelCore));
65 %Find max val in surrouding:*****
66 skelInd = find(skelCore);
67 D_core(skelInd(minId));
68
69 s = size(D_core);
70 [I, J, K] = ind2sub(s, skelInd(minId));
71 I = I + lim1;
72 n = 2;

```

```

73 I0 = I-n;
74 I1 = I+n;
75 J = J+lim1;
76 J0 = J-n;
77 J1 = J+n;
78 K = K+lim1;
79 K0 = K-n;
80 K1 = K+n;
81 [minVal, ind] = max(reshape(D(I0:I1, J0:J1, K0:K1), [], 1));
82 [I2, J2, K2]=ind2sub([I1-I0+1, J1-J0+1, K1-K0+1], ind);
83 ID = I0+I2-1;
84 JD = J0+J2-1;
85 KD = K0+K2-1;
86
87 spX1 = minVal*sx/2 +grid.X(ID, JD, KD);
88 spY1 = minVal*sy/2 +grid.Y(ID, JD, KD);
89 spZ1 = minVal*sz/2 +grid.Z(ID, JD, KD);
90
91 %% Maximum sphere
92 [maxVal, maxId]=max( D_core(skelCore) );
93
94 s = size(D_core);
95 [I, J, K] = ind2sub(s, skelInd(maxId));
96 I = I + lim1;
97 n = 5; % Search for the maximum value of D in a (+/- n)^3 cube around
          the
98 % Previously attained maximum value.
99 I0 = I-n;
100 I1 = I+n;
101 J = J+lim1;
102 J0 = J-n;
103 J1 = J+n;
104 K = K+lim1;
105 K0 = K-n;
106 K1 = K+n;
107 [maxVal, ind] = max(reshape(D(I0:I1, J0:J1, K0:K1), [], 1));
108 [I2, J2, K2]=ind2sub([I1-I0+1, J1-J0+1, K1-K0+1], ind);
109 ID = I0+I2-1;
110 JD = J0+J2-1;
111 KD = K0+K2-1;

```

```

112
113 spX2 = maxVal*sx/2 +grid.X(ID,JD,KD);
114 spY2 = maxVal*sy/2 +grid.Y(ID,JD,KD);
115 spZ2 = maxVal*sz/2 +grid.Z(ID,JD,KD);
116 %%Find max val in surrounding:
117
118 %% Minimum beam thickness:
119 [minVal2,minId2]=min( D2_core(voidSkelCore) );
120 skelInd = find(voidSkelCore);
121 D_core(skelInd(minId2));
122 s = size(D2_core);
123 [I,J,K] = ind2sub(s,skelInd(minId2));
124 I = I + lim1;
125 n = 2; % Search for the maximum value of D in a (+/- n)^3 cube
        around the
126 % Previously attained maximum value.
127 I0 = I-n;
128 I1 = I+n;
129 J = J+lim1;
130 J0 = J-n;
131 J1 = J+n;
132 K = K+lim1;
133 K0 = K-n;
134 K1 = K+n;
135 [minVal2,ind] = max(reshape(D2(I0:I1,J0:J1,K0:K1),[],1));
136
137 [maxVal2,maxId2]=max( D2_core(voidSkelCore) );
138 skelInd = find(voidSkelCore);
139 D_core(skelInd(maxId2));
140 s = size(D2_core);
141 [I,J,K] = ind2sub(s,skelInd(maxId2));
142 I = I + lim1;
143 n = 5;
144 I0 = I-n;
145 I1 = I+n;
146 J = J+lim1;
147 J0 = J-n;
148 J1 = J+n;
149 K = K+lim1;
150 K0 = K-n;

```

```

151 K1 = K+n;
152 [maxVal2,ind] = max(reshape(D2(I0:I1,J0:J1,K0:K1),[],1));
153 %% Plotting Spheres PLOTTING
154 if plotBol
155
156 sphere1 = surf(spX1,spY1,spZ1,'FaceColor',[0 0 1],'edgecolor','none','
        FaceAlpha',0.4);
157 sphere2 = surf(spX2,spY2,spZ2,'FaceColor',[1 1 0],'edgecolor','none','
        FaceAlpha',0.7);
158 daspect([1 1 1]);
159 caxis([minVal,maxVal])
160 c = colorbar;
161 c.Label.String = 'Pore Diameter (1/a)';
162 c.Label.FontSize = 14;
163 c.Location = 'South';
164 daspect([1 1 1])
165 view(3);
166 axis tight
167 camlight
168 lighting flat
169 set(gcf,'Units','Normalized','OuterPosition',[0,0.04,0.4,1]);
170 a = c.Position; %gets the position and size of the color bar
171 set(c,'Position',[a(1) a(2) a(3) a(4)]) % To change size
172
173 hold off
174 axis off
175
176 end
177 % subplot(1,2,2)
178 % histogram(D_core(skelCore),30,'Normalization','probability')
179 %% Evaluating Min/Max values
180
181 result = [vf,minVal,maxVal,minVal2,maxVal2];

1 function out = poreSizeSweep(type,vf1,vf2)
2 %poreSizeSweep will sweep through the volume fractions defined by vf1
   and
3 %vf2 for a TPMS of type "type". Return:
4 % out.b – min rel pore size
5 % out.c – max rel pore size
6 % out.d – min rel structure thick

```

```

7 % out.e – max rel structure thick
8 %% Defining grid, function
9 i = 0;
10 for vf= linspace(vf1 ,vf2 ,25)
11     try
12         i=i+1;
13         datas(i,1:5) = poreSize(vf ,type , false);
14     catch
15         i=i-1;
16     end
17 end
18 %%
19 figure
20 plot(datas(2:end-2,1),datas(2:end-2,2), 'o' ,...
21     datas(2:end-2,1),datas(2:end-2,3), 'o' ,...
22     datas(2:end-2,1),datas(2:end-2,4), 'd' ,...
23     datas(2:end-2,1),datas(2:end-2,5), 'd')
24 ylim([0, inf])
25 %xlim([0, 1])
26 x=datas(2:end-2,1);
27 y=datas(2:end-2,2);
28 ub = 0.80;
29 lb = 0.10;
30 ub2= ub;
31 y = y((lb<x)&(x<ub2));
32 x = x((lb<x)&(x<ub2));
33 X = [ones(length(x),1) x];
34 %b = X\y;
35 b = polyfit(x,y,3);
36 hold on
37 %yCalc2 = X*b;
38 yCalc1 = polyval(b,x);
39 plot(x,yCalc1)
40 %minfit = sprintf('d_{min} = %.2f + %.2f\\rho ',b(1),b(2),b(3),b(4));
41 minfit = 'D_{min} qubic fit';
42 x=datas(2:end-2,1);
43 y=datas(2:end-2,3);
44 y = y((lb<x)&(x<ub));
45 x = x((lb<x)&(x<ub));
46 X = [ones(length(x),1) x];

```

```

47 c = polyfit(x,y,3);
48 yCalc2 = polyval(c,x);
49 plot(x,yCalc2)
50 %maxfit = sprintf('d_{max} = %.2f + %.2f\\rho ',c(1),c(2),c(3),c(4));
51 maxfit = 'D_{max} qubic fit';
52 x=datas(2:end-2,1);
53 y=datas(2:end-2,4);
54 y = y((lb<x)&(x<ub));
55 x = x((lb<x)&(x<ub));
56 X = [ones(length(x),1) x];
57 d = polyfit(x,y,3);
58 yCalc3 = polyval(d,x);
59 plot(x,yCalc3)
60 %thinFit = sprintf('t_{min} = %.2f + %.2f\\rho ',d(1),d(2));
61 thinFit = 't_{min} qubic fit';
62 x=datas(2:end-2,1);
63 y=datas(2:end-2,5);
64 y = y((lb<x)&(x<ub));
65 x = x((lb<x)&(x<ub));
66 X = [ones(length(x),1) x];
67 e = polyfit(x,y,3);
68 yCalc4 = polyval(e,x);
69 plot(x,yCalc4)
70 %thickFit = sprintf('t_{max} = %.2f + %.2f\\rho ',e(1),e(2));
71 thickFit = 't_{max} qubic fit';
72 %plot(x,yCalc1./yCalc2)
73 title(type)
74 legend('Min Pore Diameter','Max Pore Diameter',...
75        'Min Thickness','Max Thickness',...
76        minfit,maxfit,thinFit,thickFit,'location','best')
77 grid('on')
78 xlabel('Volume Fraction \rho')
79 ylabel('Relative size')
80 hold off
81 out.b = b;
82 out.c = c;
83 out.d = d;
84 out.e = e;
85 end

```

A.1.3 Design

```

1 function FV = scaffold_app(app)
2 stic = tic;
3 ntic = tic;
4
5 grid.X = app.X;
6 grid.Y = app.Y;
7 grid.Z = app.Z;
8 type = app.type;
9 ps = [0 0 0]; % Phase shift. Adjust the phase in X,Y,Z direction, as a
    number from 0 - 1. Z-component to get the right top(or bottom)
    surface for microscopy
10 angles = [0 0 0]; % Possible to adjust the orientation of the lattice
    inside the cylinder. Enter in degrees.
11 % For different crystallographic orientations (Miller Indices):
12 % [ 0 0 1 ] --> angles = [ 0 0 0 ];
13 % [ 0 1 1 ] --> angles = [ 0 0 45 ];
14 % [ 1 1 1 ] --> angles = [ 45 0 atand(sqrt(2)) ];
15 % Rotgrid is the coordinate system of the lattice function. This can be
    rotated.
16 % Grid is the coordinate system of the plotted scaffold.
17 rotgrid = latticeCoordinateTransform2(grid, angles);
18 T = volFrac2isoVal(app.vf_dist, app.type, 't');
19 a = [app.aa, app.aa, app.aa];
20 fun = latticeFun(rotgrid, a, ps, type, T); % Make function
21 disp('Fun') % Displays progression
22 tfun = toc(ntic);
23 ntic = tic;
24 toc(stic)
25 bbd = norm(max(app.boxFV.vertices)-min(app.boxFV.vertices)); %The
    diagonal
26 voxelSize = bbd/80; % Scales the voxels according to diagonal according
    to
27 C = ones(size(app.boxFV.faces));
28 % Voxelates the image
29 [M,G] = patch2Im(app.boxFV.faces, app.boxFV.vertices, C, voxelSize);
30 disp('patch2im')
31 origin = G.origin; %+G.voxelSize*0.5;
32 szD = size(M);
33 Dx = linspace(0, szD(2)*G.voxelSize, szD(2)) + origin(1);

```



```

34 Dy = linspace(0,szD(1)*G.voxelSize,szD(1)) + origin(2);
35 Dz = linspace(0,szD(3)*G.voxelSize,szD(3)) + origin(3);
36 [mx,my,mz] = ndgrid(Dx,Dy,Dz);
37 % Eroding parts of the boundary to get a smoother boundary.
38 M2 = M == 1;
39 se = strel('sphere',2);
40 M2 = imerode(M2,se);
41 se = strel('sphere',1);
42 M2 = ~imerode(~M2,se);
43 D = bwdist(M2)*G.voxelSize; % DISTANCE TRANSFORM OF BOUDARY.
44 szD = size(D);
45 Dx = linspace(0,szD(2)*G.voxelSize,szD(2)) + origin(1);
46 Dy = linspace(0,szD(1)*G.voxelSize,szD(1)) + origin(2);
47 Dz = linspace(0,szD(3)*G.voxelSize,szD(3)) + origin(3);
48 [mx2,my2,mz2] = ndgrid(Dx,Dy,Dz);
49 M = permute(M,[2 1 3]);
50 D = permute(D,[2 1 3]);
51 % Using interpolation of distance transform. This is a massive saver of
52 % computational power compared to making the distance tranform of the
53 % complete image – though it will reduce the output quality.
54 M = interpn(mx,my,mz,M,grid.X,grid.Y,grid.Z,'linear');
55 D = interpn(mx2,my2,mz2,D,grid.X,grid.Y,grid.Z,'cubic');
56 D = smooth3(D,'gaussian',3,1);
57 OUT = isnan(M) | M==0;
58 % Eroding again to smooth the rugged edges coming from the
    interpolation.
59 se = strel('sphere',1);
60 OUT = ~imerode(~OUT,se);
61 se = strel('sphere',0);
62 OUT = imerode(OUT,se);
63 voxelSize2 = (Dz(end)-Dz(1))/size(D,3);
64 d = voxelSize*1 + voxelSize2*1;
65 DOUT = OUT&(D>(d));
66 % Penalty function. At a set distance from the eroded boundary, d, (
    which will
67 % correspond to the original boundary the function is set to a low
    value,
68 % which upon isosurfacing will make no triangulation.
69 fun(DOUT) = fun(DOUT) - (1.5*(D(DOUT)-d*0.80)).^2;
70 disp('Boolean')

```

```

71 toc(stic)
72 tbool = toc(ntic);
73 ntic = tic;
74 %Slice: Uncomment to make a slice of a midsection.
75 % slicePlane = mean([max(app.X),min(app.X)]);
76 %cnd = grid.X<0;% | grid.Y>0;
77 %fun(cnd) = -10;
78
79 [F,V] = latticeSurface(grid,fun); % Generates Faces and Vertiecies
80 disp('Created Lattice')
81 toc(stic)
82 tlatt = toc(ntic);
83 ntic = tic;
84 FV.faces = F;
85 FV.vertices = V;
86 %% Meshfix
87 % Removes dublicate nodes and vertiecies.
88 [FV.vertices ,FV.faces]=meshcheckrepair(FV.vertices ,FV.faces , 'dup');
89 toc(stic)
90 tdup = toc(ntic);
91 ntic = tic;
92 %% Mesh reduction % Uncomment to use functionality. Increases time.
93 %[FV.faces ,FV.vertices]=reducepatch(FV.faces ,FV.vertices ,0.65); %
    Reduces the number of faces by 50% (effectively removing unnesassary
    small faces).
94 %[FV.vertices ,FV.faces]=meshcheckrepair(FV.vertices ,FV.faces , 'meshfix')
    ;
95 % toc(stic)
96 % tfix = toc(ntic);
97 % ntic = tic;
98 %% SMOOTHING
99 disp('Smoothing')
100 conn = meshconn(FV.faces ,size(FV.vertices ,1));
101 mask = []; % Can set certain vertices fixed.
102 FV.vertices = smoothsurf(FV.vertices ,mask,conn,5,0.35, 'lowpass');
103 disp('smoothened patch')
104 toc(stic)
105 tsmth = toc(ntic);
106 ntic = tic;
107 %% Mesh cleanup

```

```

108 % Identifies separate bodies, and identifies the biggest one. Cleans
    away
109 % all small bodies (usually very small pieces along the scaffold
    boundary)
110 % and saves only the biggest body.
111 splitPatch = splitFV(FV);
112 for i = 1:size(splitPatch,1)
113     numFaces(i) = size(splitPatch(i).faces,1);
114 end
115 [maxFace,maxInd] = max(numFaces);
116 FV = splitPatch(maxInd);
117 disp('Cleaned patch')
118 toc(stic)
119 tcln = toc(ntic);
120 ntic = tic;
121 %%
122 fprintf('Fun %d\n',tfun)
123 fprintf('Boolean %d\n',tbool)
124 fprintf('lattice %d\n',tlatt)
125 fprintf('Cleaned %d\n',tcln)
126 fprintf('Smoothing %d\n',tsmth)
127 % Plots the time distribution
128 figure
129 pie([tfun,tbool,tlatt,tdup,tsmth,tcln,tfix],{'Grid and function
    definition',...
130     'Boolean operation','Marching cubes','Remove duplicate','Mesh
    cleanup',...
131     'Mesh smoothing','Mesh reduction'});
132 end

1 function [nF,nV,D,Dgrid,D2] = inner_Surface(F,V,offset,app)
2 % Gets the distance transform, D, of the boundary of the input surface
    given
3 % with faces, F, and vertices, V. It will pass back a patch nF,nV,
    given as an
4 % offset of the input surface.
5 bbd = norm(max(V)-min(V));
6 voxelSize = bbd/100;
7 C = ones(size(F));
8 %voxelSize = 6/app.res;
9 disp('patch2im')

```

```

10 [M,G,bwLabels]=patch2Im(F,V,C,voxelSize); %C,voxelSize,imOrigin,imSiz)
    ;
11 disp('patch2im done')
12 BW = bwLabels==0;
13 disp('bwdist')
14 D = bwdist(BW)*voxelSize;
15 disp('bwdist done')
16 D2 = D;
17 D(isnan(M)) = NaN;
18 [nF,nV] = isosurface(D,offset);
19 origin = G.origin-G.voxelSize*0.5;
20 nV = (nV)*G.voxelSize +origin;
21 szD = size(D);
22 Dx = linspace(0,szD(2)*G.voxelSize,szD(2)) + origin(1);
23 Dy = linspace(0,szD(1)*G.voxelSize,szD(1)) + origin(2);
24 Dz = linspace(0,szD(3)*G.voxelSize,szD(3)) + origin(3);
25 [Dgrid.Dx,Dgrid.Dy,Dgrid.Dz] = ndgrid(Dx,Dy,Dz);
26 %figure
27 %qpatch(nF,nV);
28 end

1 function vf_dist = vf_distribution(app,mode,ramplength,D,dgrid)
2 % This function will take two different volume fraction values, and
3 % distribute them according to the assigned mode as a function of the
4 % distance, D, from the surface.
5 vf1 = app.vf1;
6 vf2 = app.vf2;
7 offset = app.offset;
8 D = permute(D,[2,1,3]); % Permute: Changing the organisation of the
    image D from
9 % "meshgrid"-type to "ndgrid"-type.
10 D = interpn(dgrid.Dx,dgrid.Dy,dgrid.Dz,D,app.X,app.Y,app.Z,'cubic');
11 vf_dist = vf1*ones(size(D));
12 switch mode
13     case 'linear'
14         vf_dist(D>offset) = vf2 - (vf2-vf1).*((offset+ramplength - D(D>
                offset)))/ramplength;
15         vf_dist(D>=offset+ramplength) = vf2;
16     case 'heavystep'
17         idx = D>offset;
18         vf_dist(idx) = vf1 + (vf2-vf1).*(1./(1+exp(-(D(idx)+offset)/

```

```

        ramplength));
19 end
20 vf_dist = smooth3(vf_dist);
21 %% Plotting of the field
22 xslice = mean([max(app.x),min(app.x)]);
23 yslice = mean([max(app.y),min(app.y)]);
24 zslice = mean([max(app.z),min(app.z)]);
25 vf_dist_plot = permute(vf_dist,[2,1,3]);
26 D = permute(D,[2,1,3]);
27 vf_dist_plot((isnan(D))) = NaN; n = 3;
28 [a,b,c] = meshgrid(app.x(1:n:end),app.y(1:n:end),app.z(1:n:end));
29 figure
30 colormap(jet)
31 k = slice(a,b,c,vf_dist_plot(1:n:end,1:n:end,1:n:end),xslice,yslice,
        zslice);
32 colorbar('horiz');
33 axis('equal')
34 end

```

A.2 Electrochemical process control and measurement

```

1 clear
2
3 id = '14_sweep'; % Set a unique id for each test
4 timeLimit = 5*60; % seconds
5
6 ovp = 50; % Over voltage protection
7 VOLT = 45; % Voltage output
8 CURR = 0.5;
9
10 currDensMax = 2/1000; % mA/mm^2
11 diameter = 14.6; %mm
12 area = diameter^2*pi/2; % Area on both sides of disk;
13 I_max = currDensMax*area;
14 t_step_min = 5; % Time at sweep step level
15 t_step_min2 = 4;
16 n_sweep = 30; % Number of sweep step levels (one way)
17 vol_tol = 0.02;
18 %%
19
20 % Variable declarations

```

```

21
22 % Connect steps
23 N5771A = visa('agilent', 'USB0::0x0957::0xA807::US16H7405R::0::INSTR');
24 N5771A.InputBufferSize = 8388608;
25 N5771A.ByteOrder = 'littleEndian';
26 fopen(N5771A);
27 %%
28
29 % Over Voltage Protection
30 fprintf(N5771A, sprintf(':OVP %g', ovp));
31 fprintf(N5771A, sprintf(':VOLT %g', VOLT));
32 fprintf(N5771A, sprintf(':CURR %g', CURR));
33 %fprintf(N5771A, sprintf(':VOLT %g', VOLT));
34
35 fprintf(N5771A, sprintf(':OUT %d', 1));
36
37 pause(1)
38 %%
39 %Defining the ramping values:
40 %Make steps up, and steps back down again.
41 I_sweep = linspace(I_max*0.05, I_max, n_sweep);
42 I_sweep = [I_sweep, flip(I_sweep(2:end-1))];
43 %%
44 % Defining ramping up and down:
45 % It goes up 6 values, 3 down, 6 up, 3 down, and so on...
46 % even_dist = linspace(I_max*0.05, I_max, n_sweep);
47 % m = 0; % the pointer for even_dist
48 % j = 0; % the indexing for I_sweep
49 % while m < (n_sweep-6)
50 %     for k = 1:6
51 %         j = j+1;
52 %         m = m+1;
53 %         I_sweep(j) = even_dist(m);
54 %     end
55 %     for k = 1:2
56 %         j = j+1;
57 %         m = m-1;
58 %         I_sweep(j) = even_dist(m);
59 %     end
60 % end

```

```

61 % I_sweep = [I_sweep, flip(I_sweep(2:end-1))];
62
63
64 %%
65
66 % Making the plotting figure
67 figure(1);
68 ax1 = subplot(3,1,1);
69 C_line = animatedline;
70 xlabel('Time')
71 ylabel('I [A]')
72 ax2 = subplot(3,1,2);
73 V_line = animatedline;
74 xlabel('Time')
75 ylabel('Voltage [V]')
76 ax3 = subplot(3,1,3);
77 R_line = animatedline;
78 xlabel('Time')
79 ylabel('Resistance [Ohm]')
80
81
82 figure(2)
83 xlabel('I')
84 ylabel('V')
85
86 sweep_tic = tic;
87 i = 1;
88 time = 0;
89 index_stabil = [];
90 V_stab = [];
91 I_stab = [];
92 R_stab = [];
93 ii = 0;
94 %while toc(sweep_tic) < timeLimit
95 for p = 1:4
96     i0 = i;
97     for CURR = I_sweep
98         fprintf(N5771A, sprintf(':CURR %g', CURR));
99
100         % Let firs stay at some current for some time, t_min,

```

```

101     step_timer = tic;
102     time_step = toc(step_timer);
103     while time_step < t_step_min
104         i = i+1;
105         V(i) = str2double(query(N5771A, ':MEASure:SCALar:VOLTage:DC
           ?'));
106         I(i) = str2double(query(N5771A, ':MEASure:SCALar:CURRent:DC
           ?'));
107         R(i) = V(i)/I(i);
108         time(i) = toc(sweep_tic);
109         % pause(0.025)
110         disp(i)
111         time_step = toc(step_timer);
112     end
113     t_step_min = t_step_min2;
114     % Stay until Voltage is stabilized
115     while abs(mean(V(i-1)) - mean(V(i))) > (25/n_sweep)*vol_tol
116         % MEASUREMENT NUMBERS
117         i = i+1;
118         V(i) = str2double(query(N5771A, ':MEASure:SCALar:VOLTage:DC
           ?'));
119         I(i) = str2double(query(N5771A, ':MEASure:SCALar:CURRent:DC
           ?'));
120         R(i) = V(i)/I(i);
121         time(i)=toc(sweep_tic);
122         % pause(0.025)
123         if toc(step_timer) >3
124             disp('Spends more than 3 sec for a step')
125             disp(toc(step_timer))
126         end
127         disp(i)
128     end
129
130
131     % Plot the
132     if ismembertol(CURR, I_max/2, I_max/n_sweep)
133         figure(1)
134         addpoints(C_line, time(i0:i), I(i0:i))
135         addpoints(V_line, time(i0:i), V(i0:i))
136         addpoints(R_line, time(i0:i), R(i0:i))

```



```
137     set(ax1, 'Ylim', [0, 1.2*max(I)]);
138     set(ax2, 'Ylim', [0, 1.2*max(V)]);
139     set(ax3, 'Ylim', [0, 1.2*max(R)]);
140
141     drawnow
142     end
143
144     ii = ii+1;
145     index_stabil(ii) = i;
146     V_stab(p, ii) = V(i);
147     I_stab(p, ii) = I(i);
148     R_stab(p, ii) = R(i);
149     t_stab(p, ii) = time(i);
150
151
152     end % for
153
154     figure(2)
155     hold on
156     p1 = plot(I(i0:i), V(i0:i));
157
158     figure(3)
159     hold on
160     p2 = plot(V_stab', I_stab');
161
162     %pause(0.05);
163     end
164     %%
165
166     % Turn off output
167     fprintf(N5771A, sprintf(':OUT %d', 0));
168
169
170     % Plot results
171     figure
172     subplot(1,2,1)
173     plot(time, V)
174     xlabel('Time [s]')
175     ylabel('Voltage [V]')
176     subplot(1,2,2)
```

```

177 plot(time, I)
178 xlabel('Time [s]')
179 ylabel('I [A]')
180
181
182 filename = ['DataLogging_', id, '_', date];
183 save(filename);
184
185 % Cleanup
186 fclose(N5771A);
187 delete(N5771A);
188 clear N5771A;
189
190 %%%% Unused code:
191 %     RANDOMLY GENERATING NUMBERS
192 %     R(i) = 18 + 0.8*rand(1)*(0.5-rand(1)) + 0.05*time(i)^1.5;
193 %     I(i) = CURR*(1 + 0.01*rand(1) + 0.005*rand(1) );
194 %     V(i) = I(i)*R(i);
195 %     P(i) = V(i)*I(i);
196 %     pause(0.2)
197 %     % Quiry Error from instrument, and turn off output if there is an
198 %     % error.
199 %     err = query(N5771A, ':ERR?');
200 %     if err
201 %         fprintf(N5771A, sprintf(':OUT %d', 0));
202 %         error('Power source gives an Error')
203 %     end

```

A.2.1 More Details

TPMS pore size variation with varied volume fraction

Markus Joakim Lid^{1*}

¹Department of Mechanical and Industrial Engineering, Norwegian University of Science and Technology, Trondheim, Postal Code, Norway, 7034

Correspondence

Department of Mechanical and Industrial Engineering, Norwegian University of Science and Technology, Trondheim, Postal Code, Norway, 7034
Email: markusjl@stud.ntnu.no

Funding information

Funder One, Funder One Department, Grant/Award Number: 123456, 123457 and 123458; Funder Two, Funder Two Department, Grant/Award Number: 123459

Triply Periodic Minimal Surfaces (TPMS) have shown to be a useful tool in designing scaffolds for tissue engineering and bone regrowth purposes. The characterization of pore size is used as a parameter for cellular biomaterials. In this article, a method involving image processing transforms, such as skeletonization and the distance transform, is presented in which the poresize can be determined for a periodic cellular material. The pore size characteristics are determined for a variety of porosities and different TPMS topologies, and fitted with polynomial coefficients. The article demonstrates how the pore size can be used as an input parameter in TPMS scaffolds design.

KEYWORDS

TPMS, Pore Size, Scaffold, Design, Functional Gradient

1 | INTRODUCTION

Cellular materials are repeating structures that display unique mechanical and biological properties. The structures consist of open cell arrangements referred to as *lattice structures* (ref: <https://www.sciencedirect.com/science/article/pii/B97801281245>). Lattices are used in order to achieve porosity by using three dimensional structures. Examples of lattices in nature are honeycombs or human bones. By implementing additive manufacturing (AM), complex porous cellular structures can be fabricated with materials such as titanium, stainless steel, or other metals. This is attractive in advanced mechanical applications such as in the transport sector, where lightweight components with high strength are important, or in the biomedical sector, as orthopaedic implants. By using AM to fabricate customized orthopaedic implants, the porosity and stiffness can be controlled, acting as scaffolds for bone and tissue integration. This will also help to reduce the problem of stress shielding associated with solid implants (ref: <https://www.hindawi.com/journals/ijbm/2012/245727/abs/> + <https://www.sciencedirect.com/science/article/pii/S1875389211001271>)

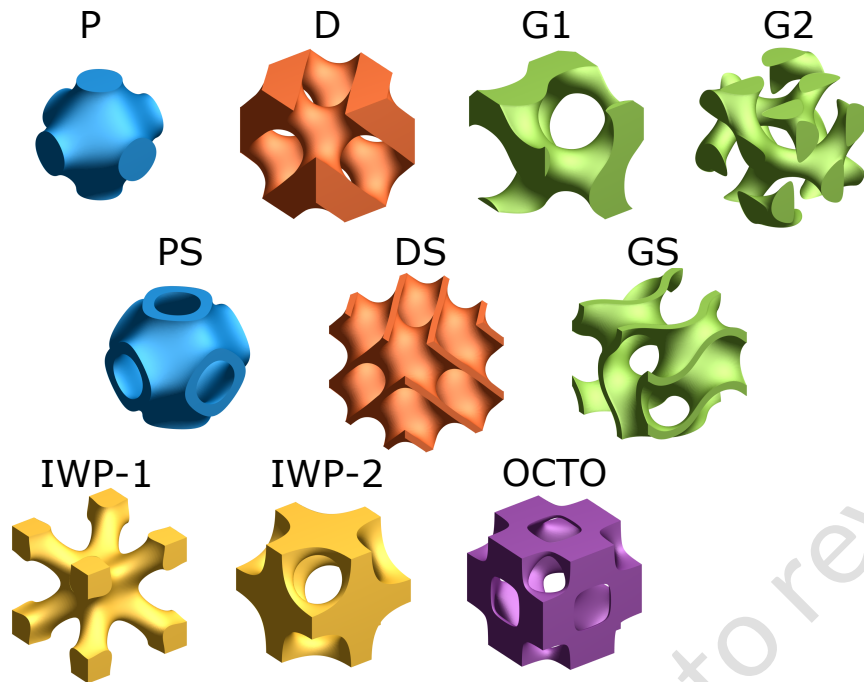


FIGURE 1 Different TPMS structures can be made, some sheet based, other beam bases. The different colors indicate that the structures originate from the same, or a similar minimal surface.

The structural stiffness and strength can be determined by modifying the lattice material, cell type and volume fraction (ref: <https://www.sciencedirect.com/science/article/pii/S0032386117311175>). It has been proved that the lattice types based on triply periodic minimal surfaces (TPMS), such as diamond and gyroid structures, have shown to have a more versatile biomorphic scaffold design for tissue engineering than previous existing porous scaffolds, such as porous ceramic and biomineralized scaffolds (<https://www.sciencedirect.com/science/article/pii/S0142961211006776>).

TPMS are mathematically defined surface, and show some very interesting properties such as high surface-to-volume ratio (<https://link.springer.com/article/10.1007/s12541-014-0516-5>), axisymmetric stiffness and pore connectivity (ref: <http://sffsymposium.engr.utexas.edu/sites/default/files/2014-097-Aremu.pdf>). The mean curvature for these surfaces are zero, which is important in orthopaedic implants, since it has been shown that the mean curvature for trabecula bone is close to zero [1]. The surface curvatures also have important effects when it comes to the regrowth of bones [2, 3, 4, 5].

TPMS can be used to make both beam-based and sheet-based scaffolds. To achieve a sheet-based design, a thickness is simply added to the surface to obtain a volume. Since TPMS have two sides, they will split a unit cell into two distinct volumes. By repeating only one of the phases, a beam-based lattice appears. The volume ratios of the two different phases can be modified, to achieve variable beam diameters and volume fractions. Examples of sheet-based TPMS structures are primitive-sheet (PS), diamond-sheet (DS) and gyroid-sheet (GS), as shown in figure 1, while primitive (P), diamond(D), gyroid (G1 and G2) are examples of beam-based structures.

When fabricating TPMS structures for orthopaedic implants, there are many things to consider, among others are stiffness, porosity, and materials. The choice of material is important due to fatigue strength, stiffness and biocompatibility. Titanium and its alloys, specifically Ti-6Al-4V, have shown to result in good biocompatibility, easy to manufacture using AM processes such as electron beam melting or selective laser melting, in addition to having excellent corrosion resistance and fatigue behavior (ref:<https://www.sciencedirect.com/science/article/pii/S1751616115002295>).

Natural bone is porous with interconnected voids consisting of outer cortical bone and inner trabecular bone. The cortical bone is dense with a porosity ranging from 5-10%, while trabecular bone has a higher porosity of 50-90% (ref: <https://www.sciencedirect.com/science/article/pii/S0142961201000023>). Hence the orthopaedic implant should possess similar interconnected pores and porosity as human bone. The porosity of the TPMS structures is not only important in achieving a similar porosity as human bone, but also important for reducing the stiffness of the orthopaedic metallic implant. For instance, dense Ti-6Al-4V has a Young's modulus of around 110 GPa, while human bone have an elastic modulus ranging from 0.5 GPa to 20 GPa (ref: <https://www.sciencedirect.com/science/article/pii/S1751616109001210>). The large variation of elastic modulus of human bone is due to important factors such as age, sex, weight and physical activity of the patient. Because of this, the elastic modulus of the orthopaedic implant should therefore be reduced to match the stiffness of human bone. This can be achieved by modifying the pore size and porosity of the implant.

However, since metal scaffolds will usually be produced using powder bed fusion, there is an increasing risk of trapped powder with smaller pores. Because of this, the pore size should be large enough to make sure all loose powder particles are removable after fabrication.

2 | THEORETICAL BACKGROUND

Since TPMS cellular structures and other cellular structures used for cell culturing are typically open-celled and fully interconnected, they form a network of curves, which meet in nodal positions. If the distance between the curve and the closest surface for every point along the curve is maximized, the curve will define all the possible center positions of possible spheres that can fit therein.

2.1 | Image processing

A topological skeleton is a thinned version of a geometry, which is equidistant to its boundaries. Within image processing there are several algorithms to obtain the medial axis transform, some are based on burning grass meeting fronts, others on local maxima of the distance transform. A skeletonization, of a binary image results in a single pixel-wide curve [6].

The euclidean distance transform is a typical method used in image processing, typically used for binary images, resulting in a gray-scale image where the gray-scale represents the euclidean distance to the closest white pixel of the input image.

3 | METHODOLOGY

3.1 | Triply Periodic Minimal Surfaces

The TPMS used are the Schwarz "Primitive" and surface, which is used to make both a beam based P and sheet based PS structure, and the Schwarz "Diamond" make up the beam based D and sheetbased D. The Schoen "Gyroid" surface is used to make a single beam structure G1, while G2 also includes the a reciprocal beam structure filling, such that the surface of each beam is equidistant from the Gyroid surface. The other beambased TPMS structures IWP-1 and IWP-2 and OCTO are also included.

They are mathematically defined functions, and can be written like this. In the table (reference) is a list of different functions. The original TPMS surface corresponds to an isovalue of $T = 0$, which splits the unit cube into two equally large volumes. If we choose to make a the volume corresponding to $< T$ a solid, a lattice structure will appear. By making

$T < 0$ the lattice will become more porous, eventually leading to no lattice, while $T > 0$ will make the lattice denser, until fully dense.

TABLE 1 TPMS functions definitions as defined in this script. Notation in the table is as follows: $C_i = \cos(ik)$, $S_i = \sin(ik)$, $C_{2i} = \cos(2ik)$ where $i = x, y, z$, and $k = \frac{2\pi}{a}$, where a is the unit cell size.

TPMS lattice type	Surface model $f_S(x, y, z) =$
P	$C_x + C_y + C_z = T$
PS	$(C_x + C_y + C_z)^2 = T^2$
D	$S_x S_y S_z + S_x C_y C_z + C_x S_y C_z + C_x C_y S_z = T$
DS (Diamond sheet)	$(S_x S_y S_z + S_x C_y C_z + C_x S_y C_z + C_x C_y S_z)^2 = T^2$
G1 (Gyroid)	$C_x S_y + C_y S_z + C_z S_x = T$
G2 (Gyroid 2)	$(C_x S_y + C_y S_z + C_z S_x)^2 = T^2$
GS (Gyroid Sheet)	$(C_x S_y + C_y S_z + C_z S_x)^2 = -T^2$
I-WP 1	$[C_{2x} C_{2y} + C_{2y} C_{2z} + C_{2z} C_{2x}] - [C_{4x} + C_{4y} + C_{4z}]/2 = T$
I-WP 2	$2[C_x C_y + C_y C_z + C_z C_x] - [C_{2x} C_{2y} C_{2z}] = T$
OCTO	$4[C_{2x} C_{2y} + C_{2y} C_{2z} + C_{2z} C_{2x}] - 3[C_{2x} + C_{2y} + C_{2z}] = T$

3.2 | Volume Fraction

The volume fraction of a given triangularised TPMS unit cell can efficiently be determined using the divergence theorem. The surface mesh is obtained using the MATLAB's inbuilt *isosurface()*, using the Marching Cubes algorithm. The calculated volume fraction can be fitted to a polynomial in order to relate the volume fraction to the TPMS isovalue. Unit cells with a voxel resolution of $100 \times 100 \times 100$ is used. The isovalues in the range of $T \in [-3, 3]$ varying from the different topologies. Roughly 30 isovalues, T are used for each topology, and fitted such that all the datapoints are within the range of $\rho = (0.05, 0.95)$. They are fitted with to a fifth degree polynomial.

3.3 | Pore size estimation

The TPMS surface is voxelized performing a logical operation corresponding to $F < T(\rho)$, where F is the TPMS function evaluated in a regular grid of points, and $T(\rho)$ is the isovalue as a function of the chosen volume fraction. For the following calculations a resolution of 100 voxels are used in each orthogonal direction per unit cell.

The MATLAB function *bwdist()* is used to obtain the distance transform of the porous space, using the voxelized TPMS as input. Skeletonization of the voxelized TPMS is performed using the inbuilt MATLAB function *bwskel()*, setting the minimum feature length to 1/10 of the unit cell size. By combining the two transforms, the all pore sizes are obtained. The minimum, d_{min} and maximum d_{max} values are obtained. The skeletonization algorithm will not always give the exact equidistant position, so the maximum values in a ± 3 voxels around position of d_{min} and d_{max} are used to improve the correctness of d_{min} and d_{max} . Performing this analysis on the inverted TPMS image, the beam thickness is

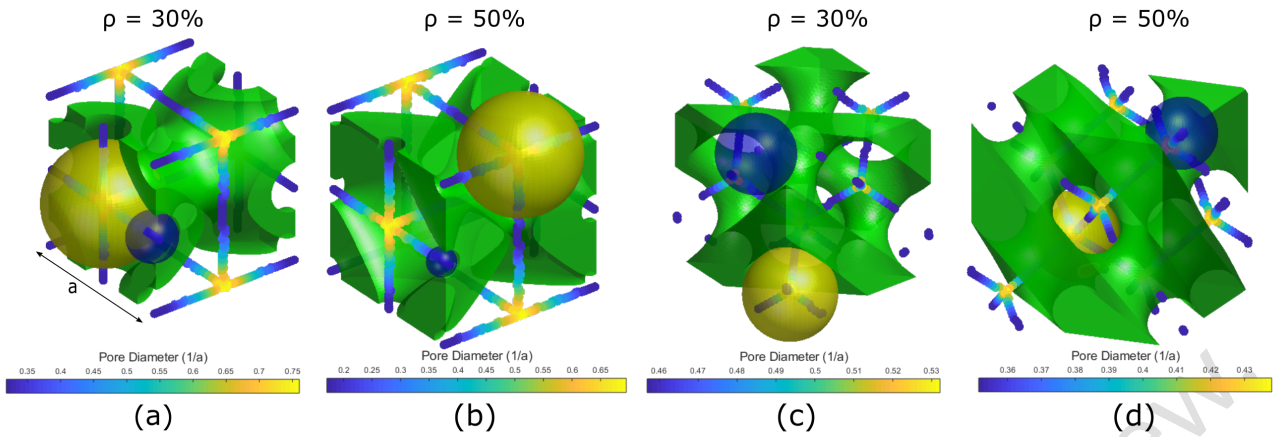


FIGURE 2 (a) and (b) shows a PS unit cell with volume fraction of 30% and 50%, while (c) and (d) shows a D unit cell with volume fraction of 30% and 50%. The curves are the medial axis transform of the porous volume, and the color indicates the distance transform value at that location. The biggest and smallest pores are demonstrated in yellow and blue.

obtained.

Because of symmetries in a TPMS, only one octant of a unit cell is sufficient for performing the analysis. The aforementioned search for *min* and *max* is performed on a variety of volume fractions, and only the values corresponding to having minimum pore size and minimum beam thickness greater than zero are used.

4 | RESULTS

4.1 | Volume Fraction

The volume fraction data is fitted to a fifth degree polynomial, using the MATLAB *polyfit()* function. The polynomial coefficients are given in Table 2.

4.2 | Pore Size

Whenever a pore would be closed because of increasing volume fraction, would cause the smallest detected relative pore size to jump. Such discontinuities are filtered out. The calculations of pore size and structure thickness is fitted to polynomials (Figure 4, 5 and Table 3). It is observed that the pore sizes decrease with increasing volume fraction as one would assume. By making the inverse of the relative pore size, the unit cell scaling factor is obtained. Fig 6 shows a relatively large difference between the minimum and maximum pore size. To better describe the difference, the pore size aspect ratio, R_p is defined as

$$R_p = \frac{d_{max}}{d_{min}} \quad (1)$$

Figure 6 shows the pore size aspect ratio w.r.t. the volume fraction, and one can see a clear difference between the

TABLE 2 The polynomial coefficients for the volume fraction for the different TPMS structures studied.

TPMS	c_1	c_2	c_3	c_4	c_5	c_6
P	51.384	-128.460	123.014	-56.060	15.574	-2.726
PS	1.027	2.217	-3.412	1.350	1.555	0.010
D	1.152	-2.880	2.352	-0.6486	2.412	-1.194
DS	5.775	-12.68	9.990	-3.439	1.697	-0.017
G1	-0.624	1.560	-2.175	1.701	2.449	-1.456
G2	0.244	-0.639	0.769	-0.637	-1.182	1.453
GS	-0.215	0.504	-0.577	0.241	1.493	0.0067
IWP1	7.149	-16.062	11.858	-1.661	2.344	-1.571
IWP2	22.139	-31.497	15.178	-1.337	2.565	-2.035
OCTO	184.218	-384.204	325.682	-135.721	33.316	-6.770

Such that $T(\rho) = c_1\rho^5 + c_2\rho^4 + \dots + c_5\rho + c_6$

different topologies.

5 | DISCUSSION

The results from this study can be used in several ways. The most evident is to use the relationship between volume fraction and relative pore size to determine the desired design parameters based on pore size specifications (e.g. the smallest allowable pore size in a scaffold to ease the removal of powders from the manufacturing process). An extension of this would be to use the pore size as a control parameter for functionally graded materials. As demonstrated in Figure 7, the pore size can be controlled to be constant with varying the unit cell size.

The pore size aspect ratio of the various structures vary massively. If there should exist some optimal value of a pore size that would be beneficially for cell growth, a bigger amount of the porous space would be in the ideal range given a low aspect ratio. The Gyroid based structures, G1, GS, have the lowest aspect ratio, being close to one for the complete range of volume fractions, while the PS has the greatest aspect ratio of the tested surfaces.

The developed method clearly defines for the maximum and minimum values, but lacks away of averaging. The average of the minimum and maximum value could be used as a measure, although it does not explain the volume average pore size in the porous phase. Averaging using the count of voxels would not be a correct measure, as the number of voxels going diagonally in an image is equal to the the number of voxels going vertically, although the lengths are different.

6 | CONCLUSION

It has been shown how the volume fraction is related to different isovalues of the implicit surface structures, and how the relative pore size is related to the volume fraction. A method based on the medial axis and distance transform is developed. These relations are useful in the design of TPMS based structures, as they assist in using the right parameters

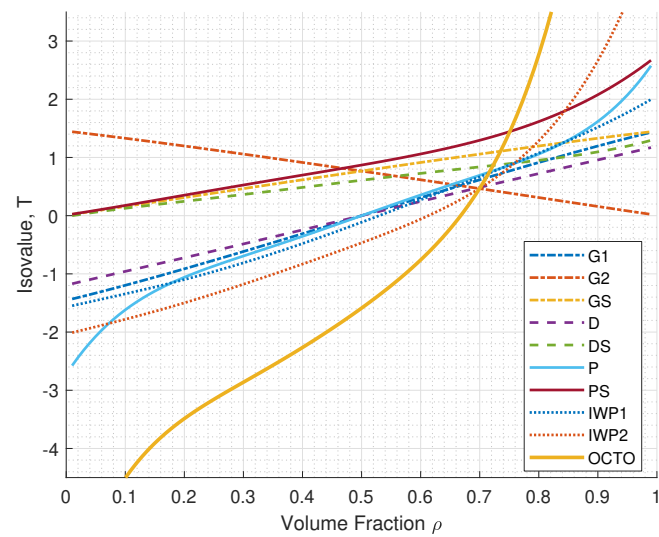


FIGURE 3 Relationship between the volume fraction and isovalue T for different TPMS structures as they are defined in Table 1. Plot is made from evaluation of the fitted 5th degree polynomials.

for a specific TPMS, as well as giving a way to evaluate one topology to another.

Developing functionally graded materials, and further utilize the possibilities in AM manufactured cellular biomaterials, one must be able to relate different features, and know which significance they have.

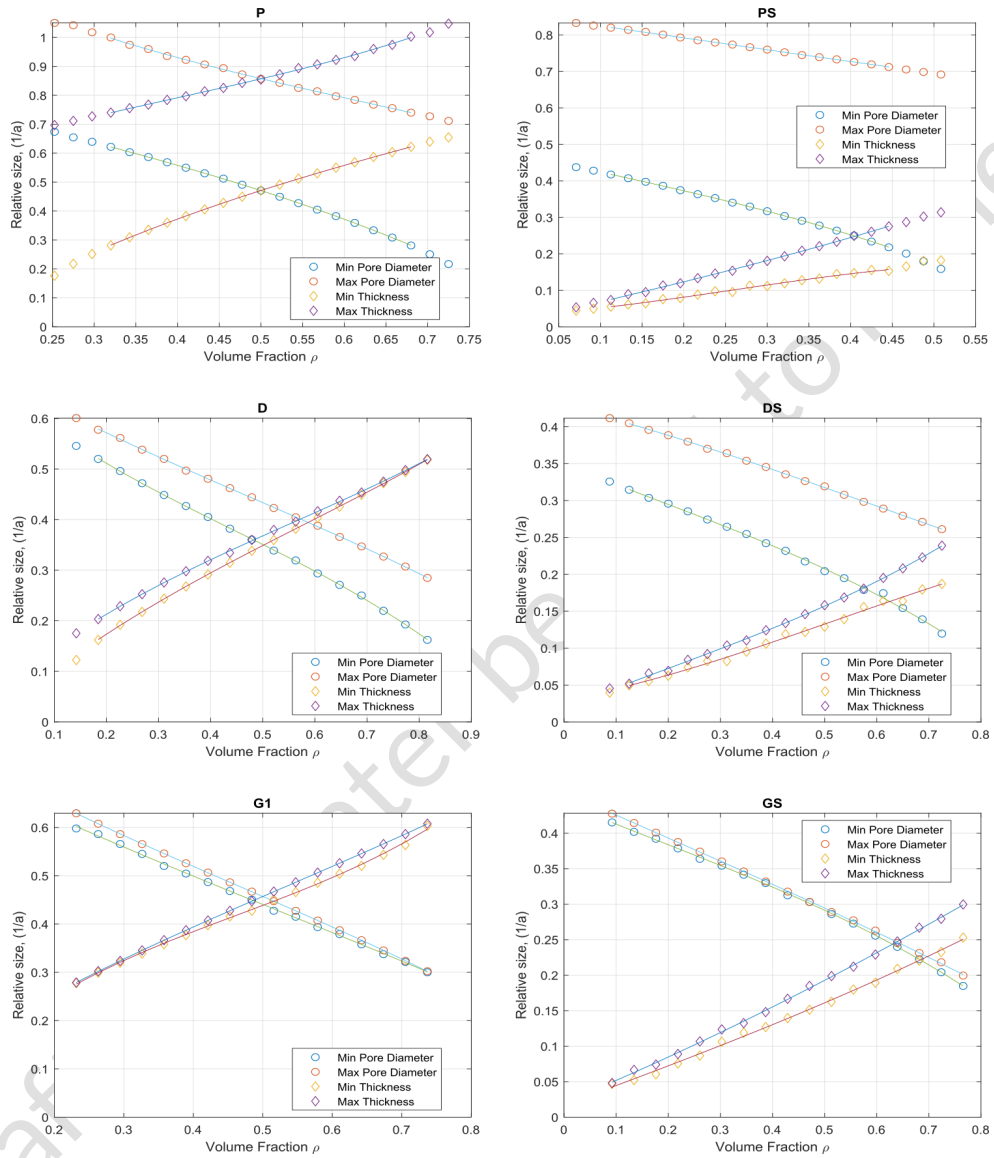


FIGURE 4 Relative pore size and relative section thickness for P, PS, D, DS, G1, GS.

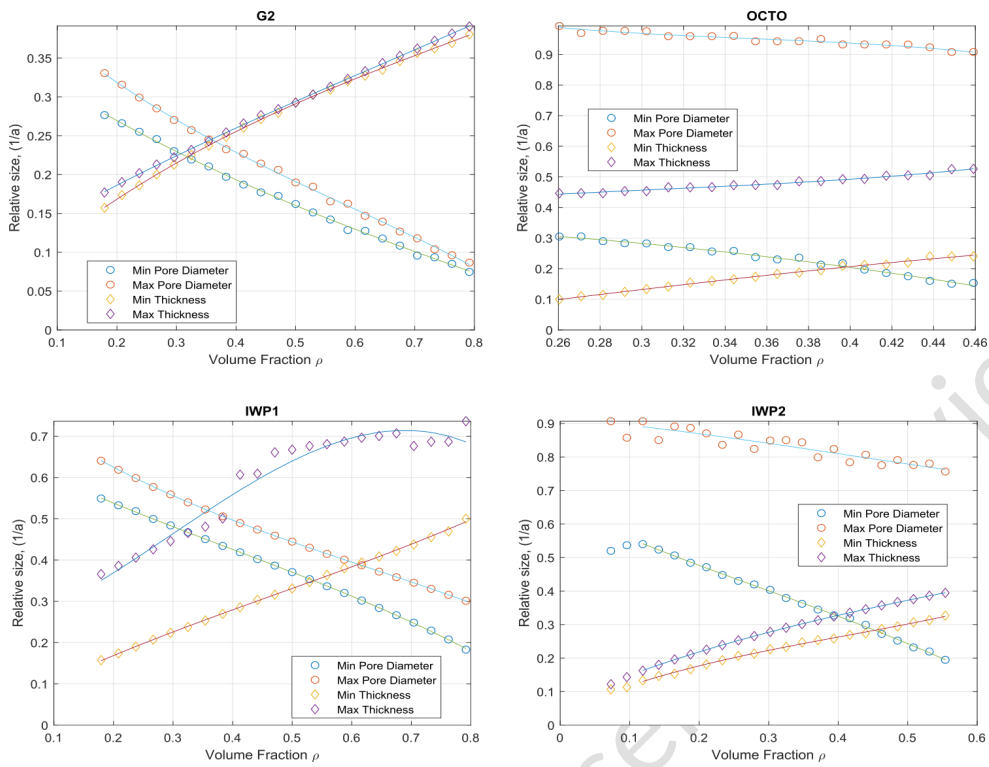


FIGURE 5 Relative pore size and relative section thickness for G2, OCTO, IWP1 and IWP2.

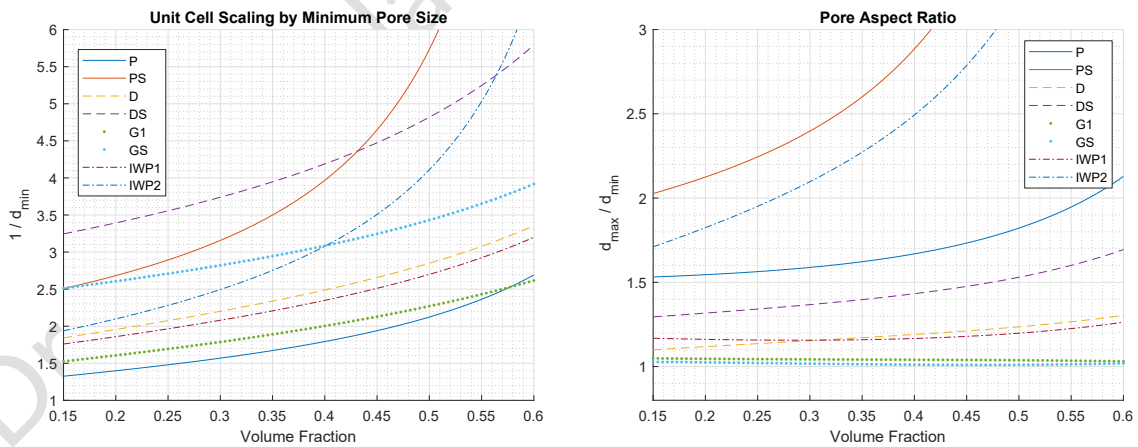
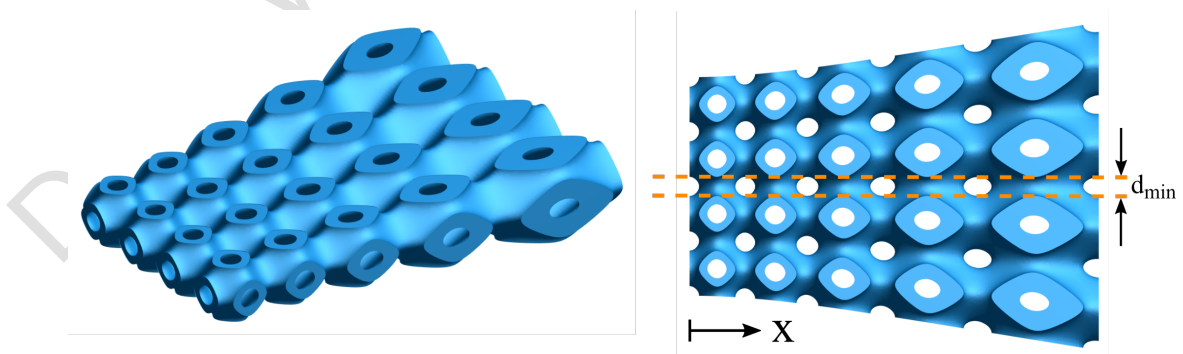


FIGURE 6 Left: The inverse of the minimal pore size w.r.t volume fraction. The inverse of the relative pore size, is the necessary side length of a unit cell to achieve a "unit sphere" pore. Right: The aspect ratio of the biggest and smallest pores size for a selection of different TPMS structures w.r.t. volume fraction.

TABLE 3 The polynomial coefficients for the minimum and maximum relative pore size and minimum and maximum structure thickness.

TPMS	min	c_1	c_2	c_3	c_4	max	c_1	c_2	c_3	c_4
P	d_{min}	-0.859	0.667	-0.945	0.884	d_{max}	-0.388	0.888	-1.292	1.330
P	t_{min}	1.158	-2.378	2.423	-0.291	t_{max}	1.213	-1.395	1.166	0.471
PS	d_{min}	-0.690	0.185	-0.522	0.475	d_{max}	0.075	-0.049	-0.317	0.857
PS	t_{min}	-1.066	0.833	0.119	0.033	t_{max}	0.279	-0.015	0.537	0.014
D	d_{min}	-0.537	0.716	-0.827	0.652	d_{max}	-0.188	0.270	-0.576	0.678
D	t_{min}	0.338	-0.589	0.865	0.021	t_{max}	0.392	-0.654	0.821	0.072
DS	d_{min}	-0.184	0.095	-0.285	0.349	d_{max}	0.058	-0.110	-0.181	0.428
DS	t_{min}	-0.144	0.227	0.127	0.030	t_{max}	0.179	-0.101	0.282	0.019
G1	d_{min}	-0.195	0.314	-0.750	0.761	d_{max}	-0.399	0.559	-0.882	0.808
G1	t_{min}	1.476	-2.051	1.489	0.022	t_{max}	0.392	-0.632	0.961	0.086
G2	d_{min}	0.101	-0.039	-0.364	0.337	d_{max}	-0.171	0.331	-0.580	0.421
G2	t_{min}	0.237	-0.522	0.680	0.052	t_{max}	0.145	-0.268	0.491	0.098
GS	d_{min}	-0.185	0.106	-0.309	0.443	d_{max}	-0.045	0.023	-0.327	0.458
GS	t_{min}	0.043	0.010	0.272	0.017	t_{max}	-0.011	0.086	0.304	0.020
IWP1	d_{min}	-0.404	0.474	-0.733	0.669	d_{max}	-0.481	0.908	-1.051	0.803
IWP1	t_{min}	0.342	-0.491	0.752	0.035	t_{max}	-2.094	1.866	0.413	0.229
IWP2	d_{min}	-0.878	0.775	-0.978	0.648	d_{max}	0.264	-0.332	-0.172	0.915
IWP2	t_{min}	0.990	-1.222	0.885	0.041	t_{max}	0.409	-0.771	0.893	0.068
OCTO	d_{min}	0.542	-2.022	0.432	0.320	d_{max}	-11.967	12.701	-4.778	1.582
OCTO	t_{min}	-1.382	0.836	0.675	-0.109	t_{max}	6.102	-5.695	2.085	0.179

Such that $f(\rho) = c_1\rho^3 + c_2\rho^2 + c_3\rho + c_5$ where f is one of the features.

**FIGURE 7** A grid of $5 \times 4 \times 1$ unit cells of PS. The elements are scaled as a function of x . The volume fraction at in one end is 25% and in the other end 43% yet, the minimum pore size is constant $d_{min} = 0.346$. Upon uneven scaling there will be shearing, possibly changing the pore size.

REFERENCES

- [1] Nishikawa Y, Koga T, Hashimoto T, Jinnai H. Measurements of Interfacial Curvatures of Bicontinuous Structure from Three-Dimensional Digital Images. 2. A Sectioning and Fitting Method 2001; <https://pubs.acs.org/doi/10.1021/1a001008p>.
- [2] Bade ND, Kamien RD, Assoian RK, Stebe KJ. Curvature and Rho activation differentially control the alignment of cells and stress fibers. *Science advances* 2017;3(9):e1700150. <http://www.ncbi.nlm.nih.gov/pubmed/28913421><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5587136>.
- [3] Bade ND, Xu T, Kamien RD, Assoian RK, Stebe KJ. Gaussian Curvature Directs Stress Fiber Orientation and Cell Migration. *Biophysical journal* 2018;114(6):1467–1476. <http://www.ncbi.nlm.nih.gov/pubmed/29590603><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5883952>.
- [4] Gouveia RM, Koudouna E, Jester J, Figueiredo F, Connon CJ. Template Curvature Influences Cell Alignment to Create Improved Human Corneal Tissue Equivalents. *Advanced Biosystems* 2017 12;1(12):1700135. <http://doi.wiley.com/10.1002/adbi.201700135>.
- [5] Vetsch JR, Müller R, Hofmann S. The influence of curvature on three-dimensional mineralized matrix formation under static and perfused conditions: an in vitro bioreactor model. *Journal of the Royal Society, Interface* 2016;13(123). <http://www.ncbi.nlm.nih.gov/pubmed/27733699><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5095209>.
- [6] Lee TC, Kashyap RL, Chu CN. Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. *CVGIP: Graphical Models and Image Processing* 1994 11;56(6):462–478. <https://linkinghub.elsevier.com/retrieve/pii/S104996528471042X>.