Department of Precision and Microsystems Engineering

Gradient based optimization of part orientation for 3D printing

Jasper Boogers

Report no.	CDM 2024.036
Coach	D.P. Munro
Supervisors	C. Ayas & P. Fanzio
Specialisation	MSD
Type of report	Thesis
Date	20-06-2024

TUDelft
Delft
University of
Technology

Challenge the future

Abstract

In Additive Manufacturing (AM), typically a trade-off exists between part quality and build time. Part orientation with respect to the print direction may significantly influence both. In this thesis, the consequences of part orientation on support volume requirements are studied. Build time, material consumption, and post-processing efforts are influenced by the amount and configuration of required supports. Using triangular surface meshes, the support requirement for a given part orientation is calculated for each triangle facet individually and summed. Gradient descent methods are used to optimize part orientation for minimum support volume. To enable implementation of gradient descent optimization, focus is placed on obtaining derivative information of the support volume on a per-facet basis. The resulting support volume function contains discontinuities, for which smooth approximation strategies are implemented. This approach is first applied to convex shapes, with promising results. For non-convex shapes however, non-local information is required. A novel method for indicating the presence of on-part supports is presented. All possible candidates for support on part are computed for each facet before the start of the optimization process. The resulting connectivity set is an inherent property of the shape and only requires calculation once. The new method is tested using numerical experiments, which indicate that gradient-based optimization of the smooth volume function outperforms the population-based approaches commonly used in the literature. Moreover, the presented work provides a framework for optimizing total part cost in which other metrics are easily appended.

Contents

1

Introduction

Additive Manufacturing (AM), also known as 3D Printing, has gained global interest over the decades, and is considered a disruptive technology [1]. Since its origins in the 1980s [2], numerous studies have been performed to investigate all intricacies of AM processes [3]. More specifically, part quality and build time (and thus cost) are considered important requirements of a 3D printed part [4, 5]. Part orientation significantly influences part quality and build time, resulting in extensive studies on optimization of part orientation [6, 7]. Common metrics are support structure volume [8, 9, 10], build time [11, 12], surface quality [13], volumetric and tolerance errors [14, 15, 16], (thermo)mechanical properties [13, 17, 18, 19], post processing time [20] and total cost [21, 22].

Of all metrics used in the literature, the required support structure volume (SSV) is used most often as it influences both build time and surface quality. Naturally, SSV directly prolongs printing times, but it also implies increased post-processing time as more material has to be removed and a larger interface area between the support structure and part has to be cleaned and polished [23]. Utilizing supports can significantly increase part cost, as the support structure is often discarded as waste after printing is completed.

There are multiple reasons to employ support structures [8]. First of all, supports are required if a certain overhang angle threshold is exceeded, see Fig. 1.1. Secondly, some part features may float above the print bed, which require a structure to start printing from. Also, supports may be used to conduct heat from the part to the build environment. Finally support structures can prevent the part from warping during production.

To determine the required amount of support, multiple approaches have been proposed. Morgan et al. [9] use a surface mesh to represent part shapes. Each triangle facet in the mesh that requires support is projected onto the build plate and multiplied by the distance between the facet and the build plate to obtain the SSV for each mesh element of interest. This approach works well for convex shapes, where support structures will never intersect the part. However, for non-convex shapes, the part itself may
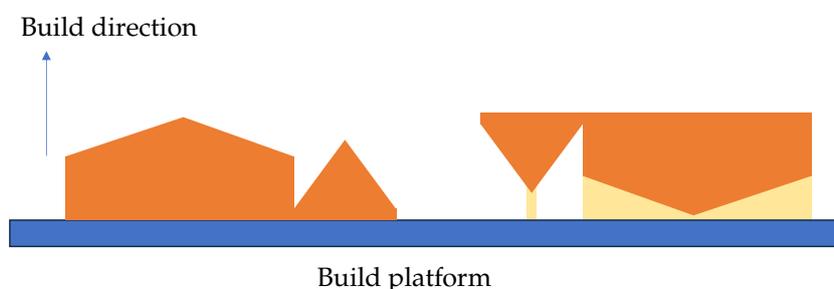
Figure 1.1: Influence of the build direction on the amount of support volume required. In the orientation on the left, no support structures are necessary, whereas the part on the right requires support (yellow) for both the overhanging faces as well as to enable printing the smaller triangle.
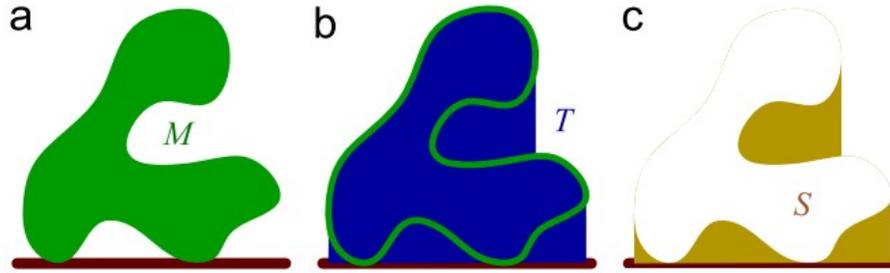
Figure 1.2: Graphical illustration of the method used by Ezair et al. [10]. The top surface T of model M is determined using grid sampling. Subtracting the volume of M from the volume below T gives the volume of required support structures S

also partake in supporting overhanging features. To identify regions where on-part support occurs, ray tracing methods have been implemented extensively throughout the literature [8, 9, 21, 24, 25]. Rays are projected from the overhanging faces to the build plate. If a ray intersects any surface of the part before the build plate, the required support volume is only to that surface. The support volume calculation is then adjusted.

Ezair et al. [10] propose an alternative method to calculate the SSV. Considering a part model M, the top cover T is calculated by sampling the highest z-coordinate along a grid on the build plane, as illustrated in Fig. 1.2. This is similar to placing a heavy cloth over the shape, where the points in contact with the cloth form the top cover. The support volume S is then calculated by computing the volume below the top cover T and subtracting the volume of the model M. While computationally efficient, it does not allow to set a user-defined overhang threshold. Interestingly, the authors observe that the obtained SSV as function of part orientation is continuous, but non-smooth. The required support structure of a vertical face forms the boundary between overhanging and self-supporting surfaces. This results in a piece-wise continuous SSV function.

The process of minimizing the SSV is extended by Strano et al. [23] through optimization of the support structure itself, after the minimum SSV is obtained. Considering the weight of the part above, cellular support structures are generated, which consume less material. In recent years, open-source slicers have implemented support structures with optimized material usage as well [26].

Optimization of support structures nears the field of topology optimization, in which numerous studies have been performed on optimizing shapes for AM as well [27, 28]. Langelaar [29] presents an overhang filter to penalize mesh elements containing support structures. Topology optimization is then used to obtain an optimal shape with minimum support requirements. In a later paper, Langelaar [30] includes a grid search to simultaneously optimize shape and orientation. This combination of shape and orientation is also considered by Hu et al. [31], although their focus is on refining existing shapes. Each node in the mesh is allowed to move. Small local deformations may reduce the total support requirement while maintaining the global shape.

While combining optimization of part shape and orientation seems a very promising field of study, it is not further considered here. First, emphasis should be placed on obtaining a metric to unify the work in the field. Furthermore, extended research is required on part orientation alone before shape refinement can be included. In many applications, part shape is (mostly) fixed. Topology optimization is therefore also regarded as outside the scope of this thesis, where the focus is on the quality and build time of existing parts.

Part quality metrics are not the only point of discussion in the literature: computing the time required to manufacture a 3D part is also studied in the literature. A common approximation is to use the number of layers to be printed as an indicator for build time [32, 33]. The number of layers closely relates to part height, which is also used as an indicator of build time [12, 34]. It should be noted that the focus is on planar layers only, non-planar 3D printing as presented in e.g. [35] is considered out of scope.

For their build time computation, Byun et al. [36] consider multiple machine operations, such as the time required for drawing the exterior contour, filling the internal area and generating supports. These

times are based on layer area and height, as well as machine speeds. Canellidis et al. [5] use similar time predictions. Frank and Fadel [37] use slicing software to provide a time estimate, whereas Di Angelo et al. [38] propose a custom machine code reader based on acceleration and travel times to gain an even more accurate estimate. The same authors also presented a parametric estimator based on a neural network in an earlier study [11].

Running a slicing algorithm for every possible orientation is computationally expensive. Also, it does not provide a closed-form expression to analyze. To circumvent this, Günaydin et al. [25] propose an estimation model based on mostly orientation dependent variables such as up- and downward facing surface area, SSV and printing speeds for shell structures. A clear division is thus observed in the literature: much research is performed on obtaining metrics as close to reality as possible. However, other studies, such as the work by Günaydin et al. [25], suggest that a metric can be considered valid as long as it remains proportional and behaves similarly to the actual value. The resultant minimum objective value has little physical meaning, but may still represent the optimal orientation. Using a reduced order model may increase computation speeds significantly while maintaining a valid solution.

When considering AM, post-processing time is a factor that deserves attention. After a part is printed, some finishing work is required. The type and amount of necessary finishing depends on the AM process, as well as the application. For example, parts might need reinforcement to the print bed using a so-called skirt or brim, which must be removed afterwards. Many AM processes utilize support structures, and their removal can be time-consuming. Especially when printing metallic materials, post-processing can be a very laborious task to meet surface quality specifications. This is achieved by using one or more abrasive techniques, ranging from (manual) sanding, tumbling, and machining to (electro)chemical treatments [20, 39]. In case of metallic parts, post-process heat treatments may be necessary to achieve required mechanical properties and relieve internal stresses [24].

Alexander et al. [21] introduce an extensive cost model as an objective function, in which the total cost is formulated as the sum of preparation, build, and post-processing costs. Within these categories, a number of operations such as setting up the machine and cleaning afterwards is included. The time required for each task is multiplied by the cost of the operator to perform these tasks. Long-term and indirect costs, such as the depreciation of the machine and power usage, are also included. Byun et al. [36] use a similar approach.

To ensure that each individual objective is considered equally, multi-objective optimization strategies may be implemented [25, 40]. These studies look for Pareto-optimal solutions. After a set of solutions is found, a set of decision making strategies is presented to obtain a single solution. Griffiths et al. [22] argue that every metric can be expressed as a function of cost, and therefore, total part cost should be considered. During the (economical) lifetime of a 3D printer, the number of printed parts should be maximized such that the write-off costs per part are minimal. However, part cost is heavily dependent on company specific variables, such as operator wages, machine setups and environment, post-processing facilities and requirements, and renewal plans. This information may not be readily available for research, rendering this method difficult to use. Instead, it may be more appropriate to optimize the required printing time rather than the cost. Multiple objectives f_i may be incorporated into a single optimization problem by using a weighted sum of the form

$$f = \sum_i w_i f_i \, ,$$

(1.1)

with user-defined weights w_i. This approach introduces the challenge of assigning proper weights to each individual objective function f_i. This is commonly solved by either seeking expert advice [24], or presenting multiple combinations of weights to highlight different outcomes of the optimization procedure [19, 32, 33, 34, 41]. Byun et al. [36] propose a slightly different solution method where first each objective is evaluated individually and then summed using user-specified weights. To avoid dominance of one objective over the others, individual objectives are commonly normalized [25, 42].

So far, differences in objective function and calculation methods have been considered. There are also multiple proposed approaches to describe the orientation of a 3D part during the optimization process. Firstly, considering rotations about the print direction vector does not influence the objective function

[25]. Therefore, it is often not considered. Most studies opt to use the other two rotation axes directly in their algorithms [12, 16, 41], while Günaydin et al. [25] use quaternion representation to define the orientation. Zhou et al. [28] implement an orientation tensor in their work, in which the authors argue it circumvents numerical instability issues associated with other representations. While it is noted, this study does not further consider differences in orientation description methods.

Once an objective and orientation description are selected, many different strategies are employed for the actual optimization of the objective. Frank and Fadel [37] implement a decision matrix, based on expert knowledge. Other early studies consider a finite number of intuitively selected candidate orientations [8, 21, 32]. Masood et al. [4] employ an extensive grid search with a fixed angular step size. This method is used in other studies as well [19, 31, 36, 43]. Naturally, the outcome is dependent on the step size. A decrease in angular step size may aid, but at increased computational cost. Zhang et al. [44] attempt to reduce the solution space by first identifying possible print orientations for each feature on the part. However, the proposed method comes with a significant computational cost, even for small models.

Population-based optimization is a commonly used family of strategies for obtaining an optimal orientation, partially because of its straightforward implementation. Many studies employ a Genetic Algorithm (GA) to find the optimal orientation [5, 24, 25, 33, 41, 45, 46]. Pereira et al. [47] implement another population based algorithm, Particle Swarm Optimization (PSO). Matos et al. [12] utilize a similar optimization strategy known as the electromagnetism-like algorithm. While solutions can be found relatively swiftly for low-dimensional optimization problems, a common disadvantage for population-based algorithms is the significant increase in computational cost for higher-dimensional solution spaces. For every iteration numerous particles have to be evaluated in every dimension.

Das et al. [16] propose a gradient based method to find the optimal orientation in their work. The initial starting point significantly affects which minimum is obtained [15]. To extend the search space , Singhal et al. [34] first orient the part about various axes to obtain a better initial guess than random selection would. Other studies also employ this method [14, 48]. Another approach is to run the optimization multiple times using different initial points [9, 48]. This enables the algorithm to explore more of the solution space and increase the likelihood of finding the global optimum. However, it consequently increases computation time. To reduce complexity and calculation time, Jaiswal et al. [49] propose the use of surrogate modeling.

It is observed that while gradient based optimization strategies are implemented in the literature, little effort is put into obtaining any gradient information. The line search method implemented by Morgan et al. [9] used finite differences to obtain the direction of descent. Other studies also do not supply analytic derivatives, which indicates a clear potential for new research. Furthermore, the treatment of discontinuities, as noted by Ezair et al. [10], lacks information and discussion. At every discontinuity in the objective function, the gradient does not exist, reducing optimization efficiency.

The aim of this thesis is to provide continuous gradient information during optimization to increase performance. Following the work of Griffiths et al. [22], the optimization problem can be formulated as minimizing a cost function, which may be a sum of cost objectives. As a first step, in this study the required support volume is selected as the cost function to be minimized, as it influences material cost, printing time and post-processing efforts. Based on Günaydin et al. [25], an approximation of the required support structures may be sufficient for obtaining an optimal orientation. This allows for the implementation of a continuous approximation of the objective in this thesis. The work of Morgan et al. [9] can provide a starting point for computing the support volume for convex shapes. The intricacies of treating non-convex parts are not provided in their work either. Combined with gradient information, computing the volume of on-part support will be the main scientific contribution of this work, leading to the following research question:

What is a computationally efficient way to optimize the orientation of non-convex parts for minimum support structures in a 3D printing environment?

The remainder of this thesis is arranged as follows. First, a brief overview of typical AM processes is presented in Chapter 2. Chapter 3 then builds on the work of Morgan et al. by incorporating derivative calculations to their methodology of computing support volume for each facet individually. Also, a smoothing strategy is presented to obtain continuous derivatives. For clarity, Chapter 3 focuses solely

on convex shapes. Chapter 4 extends the framework to include non-convex objects. The novelty of the presented approach is highlighted in Chapter 5. A discussion of the new approach is presented in Chapter 6 and the report is concluded in Chapter 7.

2

Overview of 3D printing processes

To obtain a 3D-printed part from a digital design, a number of steps are required. In general, a pipeline such as presented in Fig. 2.1 is used. From a Computer Aided Design (CAD) model, the shape is approximated using a triangular surface mesh, see Fig. 2.2. Typically, the STL (Standard Tessellation Language) file format is used. The mesh thus consists of triangular facets of varying size. Depending on the shape and resolution of the triangulation, this conversion results in a difference between the CAD model and the STL representation [32]. Next, the STL is imported into a Computer Aided Manufacturing (CAM) program, in which the part orientation is selected and support structures are generated. The STL model is then sliced along the printing direction into individual layers that the 3D printer will deposit. The slicing operation introduces a second source of approximation error in the processing pipeline, as the tessellated surface cannot be exactly represented by the layered model. This is commonly referred to as the staircase-effect, as illustrated in Fig. 2.3. Machine code is generated for each slice and the complete set of code is sent to the 3D printer. Lastly, a post-processing step may be needed to remove support material and enhance surface quality.

The choice of printing process has a direct influence on both part quality and cost. The manufacturing process may require the use of different materials, resulting in for example varying mechanical effects. Some processes may achieve smaller layer heights, thus reducing the staircase effect. One unifying feature among all categories is the fact that the product is built up in a layer-by-layer fashion. This chapter provides an overview of four common AM techniques considered in the literature. The discussed aspects are also summarized in Table 2.1. Note that this section only provides a concise description of each technique and its corresponding support volume requirements. More comprehensive reviews and comparisons of these processes and many more can be found in e.g. [3, 39, 51, 52].

Fused Deposition Modeling

Fused Deposition Modeling (FDM) is arguably the most commonly known and used printing technique [51]. FDM involves heating source material in the form of a filament and depositing it in layers through a nozzle or orifice, as illustrated in Fig. 2.4. The material exits the nozzle in a molten state, after which it fuses with the previous layer and solidifies. Common materials include Polylactic Acid (PLA),
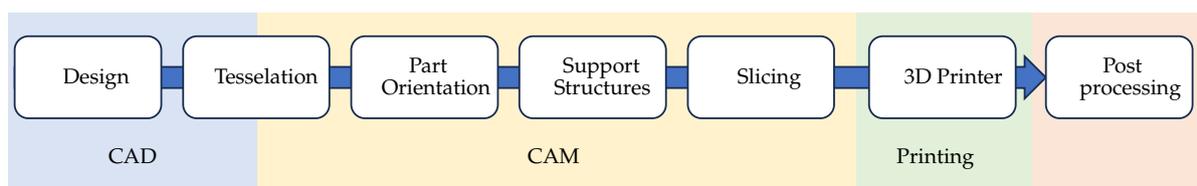
Figure 2.1: Overview of the processing pipeline for Additive Manufacturing, adapted from [50]. From a 3D model, the part is converted to a triangular mesh. The part orientation is selected and support structures are computed, after which machine code for each layer is generated. After printing, support structures are removed and the part is polished during post-processing.
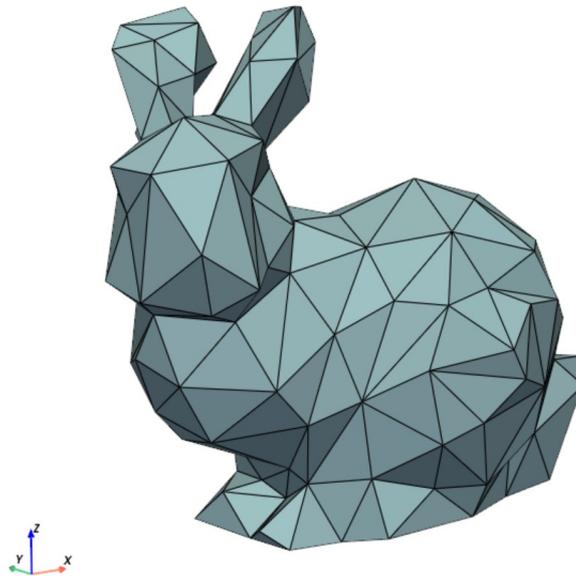
Figure 2.2: Graphical example of a triangulated surface mesh, consisting of 300 triangle facets.
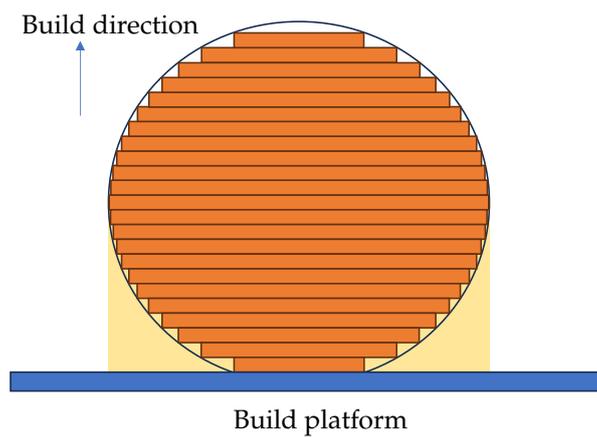
Figure 2.3: Schematic example of the staircase effect. The sphere is approximated by flat printed layers, resulting in a difference between the computer model and printed part.
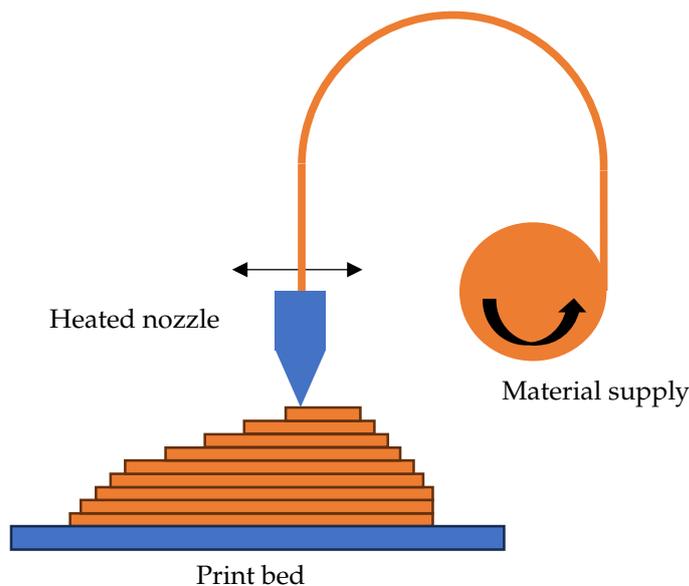
Figure 2.4: Schematic overview of the Fused Deposition Modeling (FDM) process. Filament material is heated and pushed through a moving nozzle, which places molten material at the required locations.

Acrylonitrile Butadiene Styrene (ABS) and Thermoplastic PolyUrethane (TPU). The low cost printers and materials, as well as its ease of use and very few requirements for the production environment make this 3D printing method popular for Rapid Prototyping and hobbyists. Also, FDM allows for combining different materials within the same part [3]. For instance, this can be used to print support structures with a soluble material, allowing for easy removal afterwards. Downsides include relatively low dimensional accuracy and resolution. Layer heights range from 50 to 400 µm [52]. Also, FDM generally attains lower (thermo)mechanical properties when compared to the other printing processes discussed in this chapter [19]. Surface artifacts such as the staircase effect may be reduced using sanding or even an acetone vapor bath as a post-processing step [53]. Lastly, controlling the flow of molten material through a nozzle can be challenging, leading to problems such as clogging, over-, and under-extrusion, resulting in failed prints.

Every subsequent layer can only be deposited on top of the previous layer. Slightly overhanging faces can be supported by the part itself, but steep overhangs require support structures. FDM can thus benefit from optimizing part orientation to reduce post-processing time and the cost of materials.

Photopolymerization
Early AM machines use a moving light source to locally cure polymer resin, which is held in a reservoir. The light's path and its speed dictate where the resin is cured. One of the first commercially available AM techniques is Stereolithography (SLA), developed in 1986 [2]. Modern machines are able to cure an entire layer at once using Digital Light Projection [2]. The first layer is adhered to the support bed, which moves up gradually such that a new layer of resin can flow between the scanner and the part, as shown in Fig. 2.5. Just as for FDM, overhanging features need extra support structures beyond a certain overhang threshold. Also, supports may be required to prevent warping.

Typical materials used are acrylic and epoxy resins [52]. To ensure a full cure of the resin and enhance mechanical properties, a post-cure using UV light is commonly applied. Compared to FDM, a higher resolution in the range between 10 to 100 µm can be obtained using SLA [2, 52], allowing for finer details. However, this comes at the cost of potential use of toxic resins that require removal from the finished part.

Selective Laser Sintering
Its high degree of accuracy and versatility makes Selective Laser Sintering (SLS) a popular process. A wide range of materials such as thermoplastic polymers and nylon composites can be used [3, 51]. Thin layers of fine powder are spread over the build surface and heated by a laser to just below melting point.
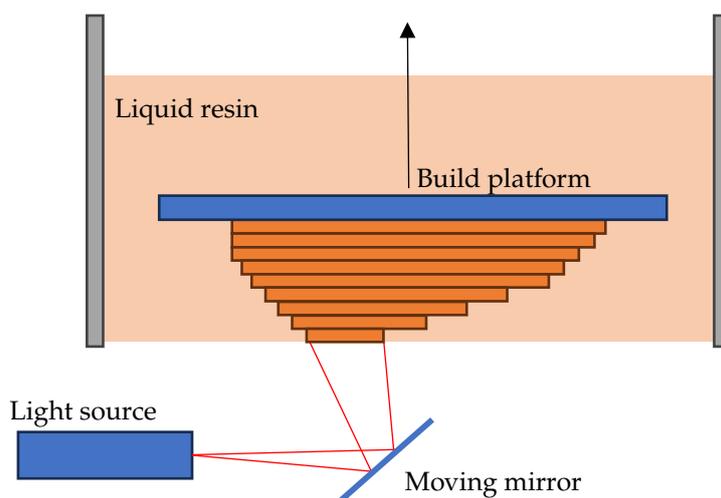
Figure 2.5: Schematic overview of the Stereolithography (SLA) process. Liquid resin is cured locally by a (UV) light source. After each layer is cured, the build platform is raised.

The powder particles are sintered, thus material fusion occurs while remaining in a solid phase. Once a layer is finished, the build platform is lowered and a new layer of unsintered powder is applied. See Fig. 2.6 for a schematic overview of the process. Excess powder material remains in place and provides support to subsequent layers [3]. Dedicated support structures are thus not needed for SLS, although extra escape holes may be required to remove unsintered powder from inner sections. After printing, the unsintered support powder can be removed and reused, no material is wasted through support structures.

Because of the local heating of material, large temperature gradients may occur during printing. To mitigate thermal stresses and warping, the unsintered powder is pre-heated [40]. This requires an enclosed environment, making the machine more expensive. With a large resolution range of 80 to 250 µm [52], SLS can be used for a range of applications such as biomedical implants, heat exchangers and interlocking parts [3].
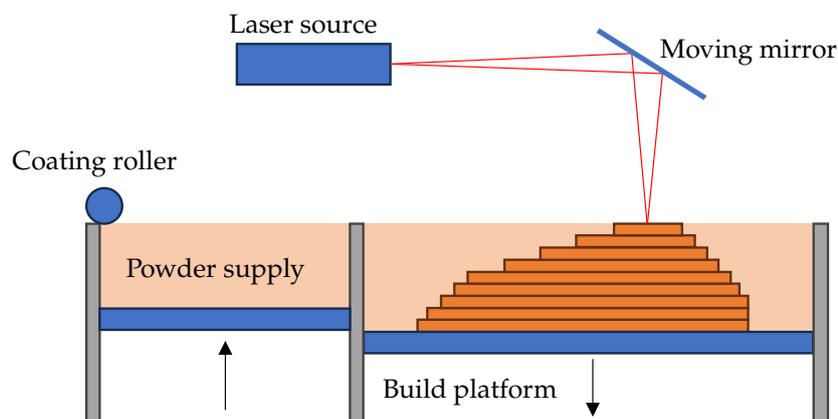
Figure 2.6: Schematic overview of the Selective Laser Sintering (SLS) process. Powder material is locally heated by a laser source and sintered to the previous layer. After a layer is complete, a thin layer of unsintered powder is applied on top and the process is repeated.

Laser Powder Bed Fusion

Like SLS, Laser Powder Bed Fusion (LPBF) uses a laser to form a layer of material from powder. However, unlike SLS, LPBF involves heating the material beyond its melting point to fuse the powder particles. Otherwise, the processes are closely related. Selective Laser Melting (SLM) is commercially used name for LBPF, which also highlights the difference with SLS more clearly. Layer height may range between 20 and 100 µm [39]. SLM is applied for manufacturing parts of specific metal alloys like magnesium

Process name	Materials used	Layer height (μm)	Advantages	Disadvantages
FDM	Thermoplastics	50 - 200	Affordable Accessible Combine multiple materials	Low resolution Low mechanical properties May require support material Use of toxic materials
SLA	Polymer resins	10 - 100	High resolution	Post-cure necessary May require support material
SLS	Nylon Ceramics Metal alloys	80 - 250	Wide range of materials High resolution High mechanical properties No support material required	Expensive Enclosure necessary Thermal stresses
SLM	Metal alloys	20 - 100	High mechanical properties Allows for specialist materials	Expensive process and material Enclosure necessary Thermal stresses Support material required

Table 2.1: Overview of the presented AM techniques and their characteristics.

and titanium. This makes it a valuable process to produce high end structural parts as well as implants and automotive parts [54]. Atzeni et al. [55] argue that SLM is more cost-efficient for small batches than 5-axis machining or die-casting. Downsides include the high cost of operation and the need for extensive, labor-intensive post-processing [25], as well as the introduction of residual stresses during production due to high thermal gradients [25, 56]. Support structures are often required to provide a heat sink to the build environment. This can mitigate stress build-up [17]. Furthermore, supports may be used to anchor a part in place and prevent warping. Especially in case of SLS and SLM, support structures may thus have many additional functions beside providing support to overhanging features. While noted, this thesis does not further consider these requirements. Therefore, the proposed method can be considered mainly applicable to FDM and SLA.

3

Support volume calculation for convex shapes

In an AM environment, the STL file format is the most common method for representing shapes. STL files consist of a collection of triangular facets, defined by point coordinates and normal vectors. Together they form a mesh, see Fig. 2.2. Morgan et al. [9] utilize this format by proposing to first calculate the support volume requirements for each facet individually. This chapter will build on their work by investigating gradients of support volume. To better illustrate the proposed approach, a 2D shape is used. Consider a simple unit square which is rotated by an angle θ in Fig. 3.1. The shape is represented by points $\mathbf{p}_1$ to $\mathbf{p}_4$ and lines with normal vectors $\mathbf{n}_i$. Of interest is the yellow area below the square, which can be considered the 2D support area [1].
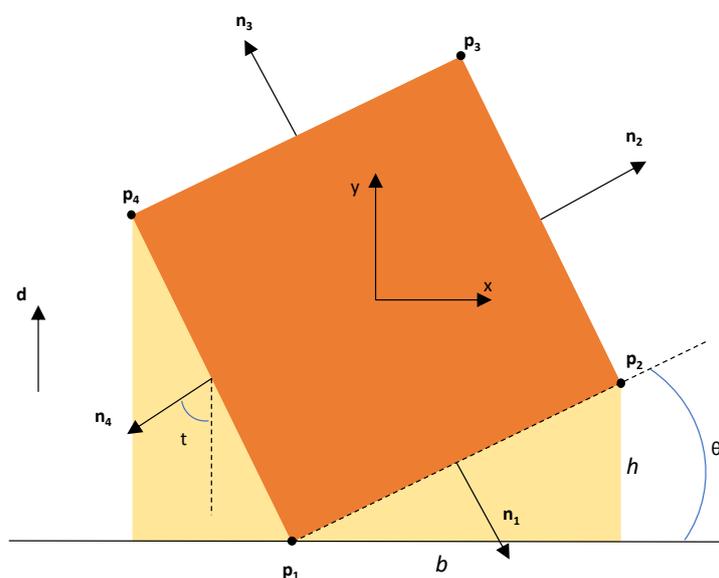
Figure 3.1: A 2D square with sides of length 1, rotated about its geometric center by angle θ. The center of rotation coincides with the origin. Each side is one facet, defined through point coordinates $\mathbf{p}$ and facet normal $\mathbf{n}$. Overhang threshold t determines whether a facet requires support. With build direction $\mathbf{d}$ pointing upwards, the support area between the part and the build plate is indicated in yellow.

---

[1]Corresponding to the 2D representation in Fig. 3.1, an area is calculated. Considering a 3D figure, a volume would be considered, which is discussed in Section 3.2.

3.1. 2D support calculation using mesh surfaces

The total support area A in Fig. 3.1 is expressed as function of rotation angle θ using simple trigonometric relations as

$$A = |\sin\theta\cos\theta| \, , \tag{3.1}$$

with b and h as indicated in Fig. 3.1. Absolute values are used to avoid negative area computations for θ outside the domain $\left[0, \frac{\pi}{2}\right]$, which is undesired. Eq. 3.1 is differentiated with respect to θ as

$$A_{,\theta} = \frac{\sin 4\theta}{2|\sin 2\theta|} \, , \tag{3.2}$$

where $A_{,\theta}$ denotes the derivative of A with respect to θ, i.e. $\frac{dA}{d\theta}$. In the context of surface meshes, the aim is to obtain the area below the part using mesh points $\mathbf{p} = [x, y]$, facet normal vectors $\mathbf{n} = [n_x, n_y]$, and rotation angle θ. A threshold angle t is used to determine whether an overhanging facet requires support, as indicated in Fig. 3.1. For simplicity, this section considers any overhanging facet to require support. The triangular area A_{12} below the line between points $\mathbf{p}_1$ and $\mathbf{p}_2$ of Fig. 3.1 is considered as an example. The area is the product of the base b and height h of that area as a function of the point coordinates in rotated configuration, such that

$$A_{12} = \frac{1}{2}bh \, . \tag{3.3}$$

Projecting the line between points 1 and 2 to the print bed using the vector norm results in base b

$$\begin{aligned} b &= |\mathbf{p}_2 - \mathbf{p}_1|\cos\theta \\ &= x_2 - x_1 \, . \end{aligned} \tag{3.4}$$

Next, the height can be computed using the difference between the y-coordinates of points 1 and 2,

$$h = y_2 - y_1 \, . \tag{3.5}$$

Using rotation matrix $\mathbf{R}$ and the point coordinates in the un-rotated initial configuration $\mathbf{p}^0$, the node positions in the rotated configuration $\mathbf{p}$ are obtained,

$$\begin{aligned} \mathbf{p} &= \mathbf{R}\mathbf{p}^0 \\ \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} , \end{aligned} \tag{3.6}$$

It is important to observe that the axis of rotation can significantly alter the transformation result, and therefore the obtained support volume and its derivative. Rotations are therefore always calculated about the geometric centroid of the shape considered. The explicit appearance of θ in Eq. 3.6 allows for easy differentiation of point $\mathbf{p}$ with respect to the rotation angle

$$\begin{aligned} \mathbf{p}_{,\theta} &= \mathbf{R}_{,\theta}\mathbf{p}^0 \\ \begin{bmatrix} x_{,\theta} \\ y_{,\theta} \end{bmatrix} &= \begin{bmatrix} -\sin\theta & -\cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix} \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} . \end{aligned} \tag{3.7}$$

The derivative of Eq. 3.3 can be calculated using a combination of the product rule and chain rule for differentiation with respect to θ, resulting in

(a) Resulting support area function.

(b) Derivative of the area function with respect to rotation angle θ.
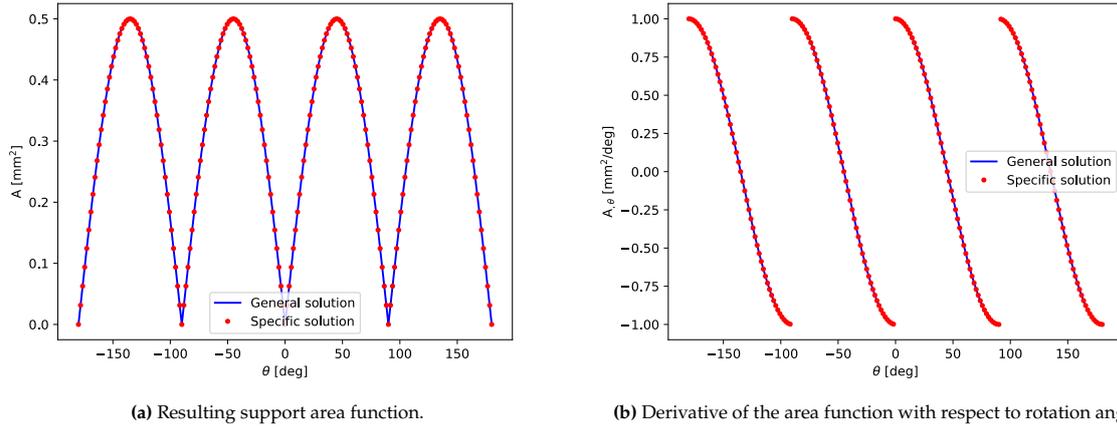
Figure 3.2: Comparison of specific and general solution method for calculating the support area below a unit square. The general solution makes use of facet coordinates and normal vectors, whereas the support area is calculated directly for the specific solution.

$$A_{12,\theta} = A_{,b}\left(b_{,x_1}x_{1,\theta} + b_{,x_2}x_{2,\theta}\right) + A_{,h}\left(h_{,y_1}y_{1,\theta} + h_{,y_2}y_{2,\theta}\right) \; . \tag{3.8}$$

The procedure of using Eqs. 3.4 to 3.8 is repeated to obtain the area A_{41} below the line between points $\mathbf{p}_4$ and $\mathbf{p}_1$ and the derivative with respect to θ. Summing areas A_{12} and A_{41} results in the total support area below the square in Fig. 3.1. Fig. 3.2 highlights the clear correspondence in results between the general approach discussed in this section and the specific solution for this unit square following Eq. 3.1. Stationary points are at equal points in the plot, and the derivatives have the same location of discontinuities.
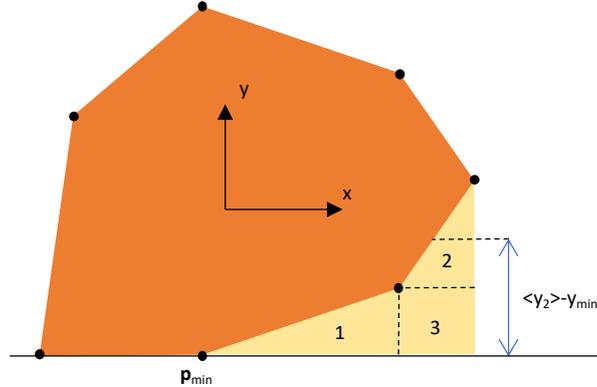
Figure 3.3: Example of a 2D shape where the supported facet is not directly in contact with the build plate. Areas 1 and 2 can be calculated, but the presented approach needs to be extended to implement calculation of area 3. For this, the distance between the facet's average y-coordinates $< y_2 >$ and the minimum y-coordinate of the shape $y_{\min}$ is used.

In most situations, not all facets are directly placed on the build surface, see Fig. 3.3. In the calculation, an additional area should be included. Eq. 3.5 can be adapted to apply to more general shapes. A point $\mathbf{p}_{\min}$, which would be the location of the print bed as indicated in Fig. 3.3, can be selected based on the minimum y-coordinate of all points of the shape,

$$y_{\min} = \min\left\{y_1, ..., y_n\right\} \; . \tag{3.9}$$

Taking the difference between the facet's average y-coordinate $< y >$ and $y_{\min}$ results in the height of the support area

$$h = <y> - y_{\min} ,\tag{3.10}$$

with derivative

$$h_{,\theta} = <y_{,\theta}> - y_{\min,\theta} .\tag{3.11}$$

For the unit square in Fig. 3.1, the calculated support remains the same. However, for more complicated shapes such as in the next section, Eq. 3.10 allows for a valid calculation of support structures.

3.2. Support calculation in 3D
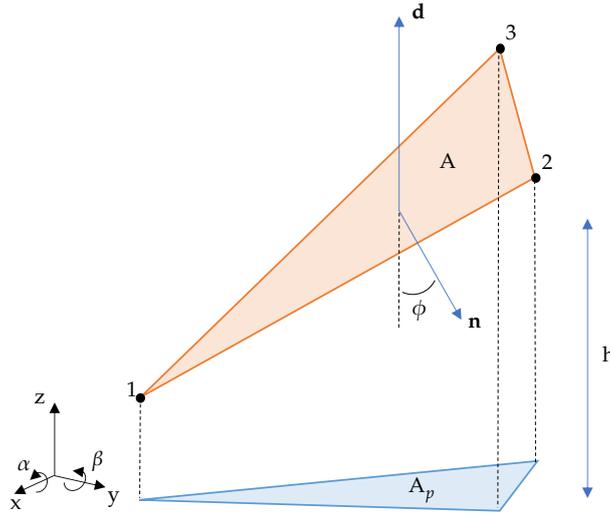
Figure 3.4: Illustration of projecting a triangle facet with area A to the build plate. Based on the acute angle ϕ between facet normal $\mathbf{n}$ and build direction $\mathbf{d}$, the projected facet area A_p on the build plate is calculated.

In the previous section, the foundation for calculating support volume below a shape was explained. This section extends the approach to three dimensions, where instead of an area, the volume is of interest. For convenient modeling and analysis of mesh objects, PyVista is used [57]. Firstly, Eq. 3.6 should be extended for a triangular facet illustrated in Fig. 3.4 to indicate angles α and β about the x- and y-axis, respectively

$$\mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$\mathbf{R}_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} .\tag{3.12}$$

In the context of this study, rotations around the build direction vector $\mathbf{d}$ - which is parallel to the z-axis - do not result in a change of support volume. Therefore, rotation about the z-axis is not considered in the remainder of this thesis. As a general convention, rotations are computed first about the global x-axis and subsequently about the global y-axis. The total rotation matrix is calculated as

$$\mathbf{R} = \mathbf{R}_y(\beta)\mathbf{R}_x(\alpha) ,\tag{3.13}$$

with derivatives with respect to α and β

$$\mathbf{R}_{,\alpha} = \mathbf{R}_y \mathbf{R}_{x,\alpha} , \tag{3.14}$$

and

$$\mathbf{R}_{,\beta} = \mathbf{R}_{y,\beta} \mathbf{R}_x . \tag{3.15}$$

The support volume required for an overhanging facet can be calculated similar to Eq. 3.3 as the product of the projected facet area and the facet height. Following Fig. 3.4, the projected facet area A^p on the xy-plane is computed using the facet area A, outward facet normal vector $\mathbf{n}$ and build direction vector $\mathbf{d}$ using

$$\begin{aligned} A^p &= -A\mathbf{n} \cdot \mathbf{d} \\ &= -A\mathbf{R}\mathbf{n}^0 \cdot \mathbf{d} , \end{aligned} \tag{3.16}$$

where $\mathbf{n}^0$ denotes the facet normal vector in the initial configuration. As the facet normal points opposite of build direction $\mathbf{d}$, it is overhanging, a minus sign is introduced to obtain the acute angle ϕ with $\mathbf{d}$, corresponding to Fig. 3.4. Eq. 3.16 has derivative with respect to α

$$A^p_{,\alpha} = -A\mathbf{R}_{,\alpha}\mathbf{n}^0 \cdot \mathbf{d} . \tag{3.17}$$

$A^p_{,\beta}$ is calculated in a similar manner. Similar to Eq. 3.10, the height h of the mesh facet is calculated using the average height of the 3 facet point coordinates and the minimum z-coordinate of the shape as

$$\begin{aligned} h &= \frac{1}{3} \sum_{i=1}^{3} z_i - z_{\min} \\ &= <z> - z_{\min} . \end{aligned} \tag{3.18}$$

It is important to observe the cross-coupling of rotation angles α and β when computing the derivative of a point. Given some coordinate $\mathbf{p} = [u, v, w]$, its derivatives with respect to the respective rotation angles α and β are

$$\mathbf{p}_{,\alpha} = \mathbf{R}_{,\alpha}\mathbf{p}^0 = \begin{bmatrix} v \sin\beta \\ -u \sin\beta - w \cos\beta \\ v \cos\beta \end{bmatrix} , \tag{3.19}$$

and

$$\mathbf{p}_{,\beta} = \mathbf{R}_{,\beta}\mathbf{p}^0 = \begin{bmatrix} w \\ 0 \\ -u \end{bmatrix} . \tag{3.20}$$

Although in the case of $\mathbf{p}_{\min}$ only the z-coordinate may seem relevant, the location of $\mathbf{p}$ on the xy-plane will have a significant impact on its derivative.

The volume should only be considered if the facet faces downward and hence requires support. This is based on whether the angle between the facet normal vector and the negative build direction is larger than the overhang threshold t, see Fig. 3.1. This definition of overhang threshold is somewhat unconventional, typically this angle is defined between the facet and the build plate [58, 59]. However, the facet orientation is provided through the facet normal vector when using surface meshes. Directly implementing an overhang threshold based on the normal vector itself is more convenient for upcoming computations.

Finally, combining Eqs. 3.16 and 3.18 results in the required support volume by summing over n facets, such that

$$V = \sum_{i=0}^{n} A_i^p h_i \qquad\qquad \text{if } -\mathbf{n}_i \cdot \mathbf{d} > \cos(t) \,. \qquad\qquad (3.21)$$

Product rule differentiation of the support volume with respect to a rotation angle θ results in

$$V_{,\theta} = \sum_{i=0}^{n} A_{i,\theta}^p h_i + A_i^p h_{i,\theta} \qquad\qquad \text{if } -\mathbf{n}_i \cdot \mathbf{d} > \cos(t) \,. \qquad\qquad (3.22)$$

Inserting Eqs. 3.16 and 3.18 in Eq. 3.21 results in the full equation for calculating the support volume in terms of the facet coordinates and normal vectors

$$V = -\sum_{i=0}^{n} A_i \mathbf{n}_i \cdot \mathbf{d} \,(<z_i> - z_{i,\min}) \qquad\qquad \text{if } -\mathbf{n}_i \cdot \mathbf{d} > \cos(t) \,. \qquad\qquad (3.23)$$

3.3. Realization of continuous support volume

The results presented in Fig. 3.2 contain non-smooth behavior of the support area functions. The gradient is thus not well-defined everywhere. This is in line with the observations of Ezair et al. [10]. Although some optimization algorithms are able to handle discontinuities, the performance of the optimization can be increased by through smoothing of the objective function [60]. This section revisits the aforementioned formulation, but with the aim of removing discontinuities.

In the extension of the formulation to 3D, it is evident that Eqs. 3.18 and 3.21 contain non-differentiable operations. Starting with the latter, instead of using a conditional statement to select which facets need support, the support volume can be calculated for all facets. Next, the volume under each facet can be multiplied with a scalar M to indicate whether the calculated volume actually contributes to the total volume. This essentially moves the conditional statement into Eq. 3.21, which then becomes

$$V = \sum_{i=0}^{n} M_i A_i^p h_i \,, \qquad\qquad (3.24)$$

where M functions as a mask used to determine the contribution of each facet to the total support volume. M being dependent on the rotation angles, the derivative of V in Eq. 3.22 should be extended accordingly to

$$V_{,\alpha} = \sum_{i=0}^{n} M_{i,\alpha} A_i^p h_i + M_i A_{i,\alpha}^p h_i + M_i A_i^p h_{i,\alpha} \,. \qquad\qquad (3.25)$$

In the current implementation, M is a discontinuous function of the facet normal and overhang threshold,

$$M = \begin{cases} 1, & \text{if } -\mathbf{n}_i \cdot \mathbf{d} > \cos(t) \\ 0, & \text{otherwise} \end{cases} \,. \qquad\qquad (3.26)$$

Fig. 3.5 presents an example of the mask applied to a mesh object . Each facet is assigned either a 0 or 1, indicating whether support is required or not. A sharp transition occurs between overhanging and self-supported facets. Parameter t is used to set the overhang threshold to 90 degrees, as defined in Fig. 3.1. Note that the bunny in Fig. 3.5 is actually a non-convex shape, and as such the actual support volume cannot be calculated, yet. Non-convex shapes are topic of Chapter 4, in which the bunny shape will be revisited. For comparison purposes, it is used here as well.
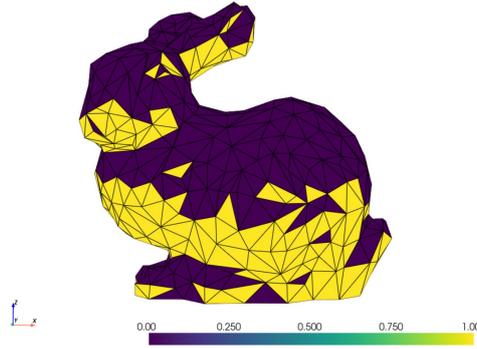
Figure 3.5: Binary mask values visualized on a decimated bunny. Each facet is either purple (self-supporting) or yellow (overhanging).

To generate the values for mask M in a differentiable manner, a smooth Heaviside approximation [61] is used in the form of

$$H(x, k, t) = \frac{1}{1 + e^{-2k(x-t)}} \ , \tag{3.27}$$

with k denoting a smoothing parameter. Applying Eq. 3.27 to the z-component of the facet normals results in a value between 0 and 1 in a smooth manner, as indicated by Fig. 3.6a. Depending on the value of k, the transition between 0 and 1 is more or less steep. Selecting different overhang thresholds shifts the transition, see Fig. 3.6b. In any case, for $n_z = \cos(t)$, the Heaviside approximation is 0.5, indicating a facet is on the boundary of exceeding the overhang threshold. Differentiation with respect to rotation angle α, realizing that $x = -n_z(\alpha)$, may be compactly written as

$$M_{,\alpha} = M(1 - M)(-2kn_{z,\alpha}) \ . \tag{3.28}$$
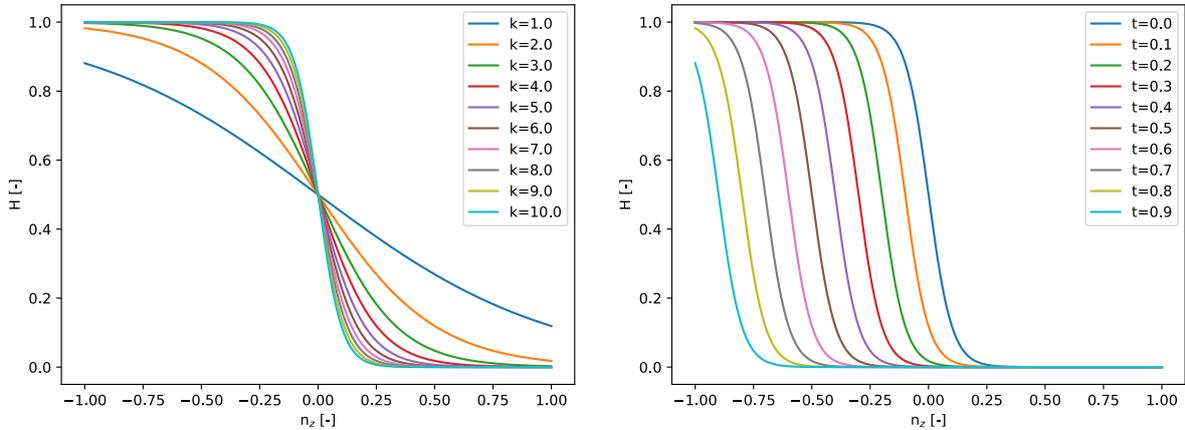
(a) Smooth approximation using varying values of smoothing parameter k.

(b) Smooth approximation using varying threshold parameter t with $k = 20$.

Figure 3.6: Smooth Heaviside approximation on facet normal z-component for varying parameter values. Instead of a discontinuous step function, a gradual transition from 1 to 0 is observed.

The effect of using Eq. 3.27 is presented in Fig. 3.7a. Eqs. 3.21 and 3.24 are applied to a cube with a side length of 1. Compared to Fig. 3.2, the shape of the curves correspond well, with corresponding locations of stationary points. Regarding the effects of smoothing, differences in the resulting volume function becomes small for $k > 10$. This is confirmed in Fig. 3.7b, where the error in volume function is plotted for different degrees of smoothing. At stationary points, the error approaches zero for larger values of k. Around the stationary points, an increase in error can be observed for all values of k. This is because the gradient gradually decreases instead of being undefined. Therefore, this difference is actually desired, because it allows for the gradient to exist.

(a) Resulting support volume approximation for different values of k.　　**(b)** Approximation error relative to non-smoothened support volume for different values of k.
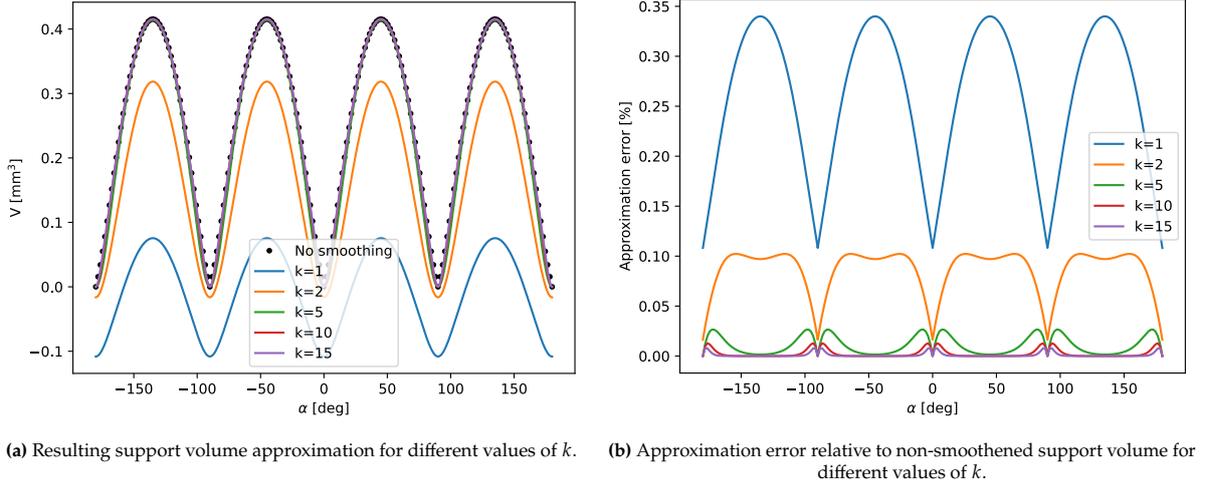
Figure 3.7: Effect of using different values of smoothing parameter k for obtaining a smooth approximation of the support volume using smooth Heaviside approximation.

Eq. 3.18 is the next non-differentiable operation, which uses the shape's lowest point coordinate. To circumvent this, a variant to the P-norm called the mellowmin operation [62] is applied to all point coordinates to obtain an approximation of the lowest point using element-wise operations, such that

$$\mathbf{p}_{min} = -\frac{1}{w} \log \left(\frac{1}{n} \sum_{i=0}^{n} e^{-w\mathbf{p}_i} \right) , \tag{3.29}$$

where w is typically a large value and n is the total number of facets. The derivative of Eq. 3.29 with respect to rotation angle α is

$$\begin{aligned} \mathbf{p}_{\min,\alpha} &= \sum_{i=0}^{n} \mathbf{p}_{\min,i} \odot \mathbf{p}_{i,\alpha} \\ &= \sum_{i=0}^{n} \frac{e^{-w\mathbf{p}_i}}{\sum_{j=0}^{n} e^{-w\mathbf{p}_j}} \odot \mathbf{p}_{i,\alpha} , \end{aligned} \tag{3.30}$$

with $\odot$ denoting the Hadamard product for element-wise multiplication. Similar to the Heaviside approximation, the effects of using different weightings in Eq. 3.29 are studied. Fig. 3.8a presents the smoothened support volume functions compared the discontinuous formulation of the support volume. For higher values of w, the error relative to not using Eq. 3.29 decreases, as indicated in Fig. 3.8b. In practice however, a limit on w exists as Pyvista uses single precision coordinates. Increasing the weight will not result in more accurate approximations. Note also that discontinuities still occur in Fig. 3.8. Heaviside approximation was deliberately omitted to not interfere with comparing weights in the mellowmin operation.

Fig. 3.9a presents the implementation of both the Heaviside and mellowmin approximations compared to the non-smooth support volume function. As indicated by Fig. 3.9b, the discontinuities are replaced by a smooth curve, and its derivative now exists everywhere. To better highlight the differences, a low smoothing parameter k is used. However, this does not influence the location of stationary points. These approximation techniques will play a vital role in obtaining the support volume of non-convex shapes, which is discussed in the next chapter.
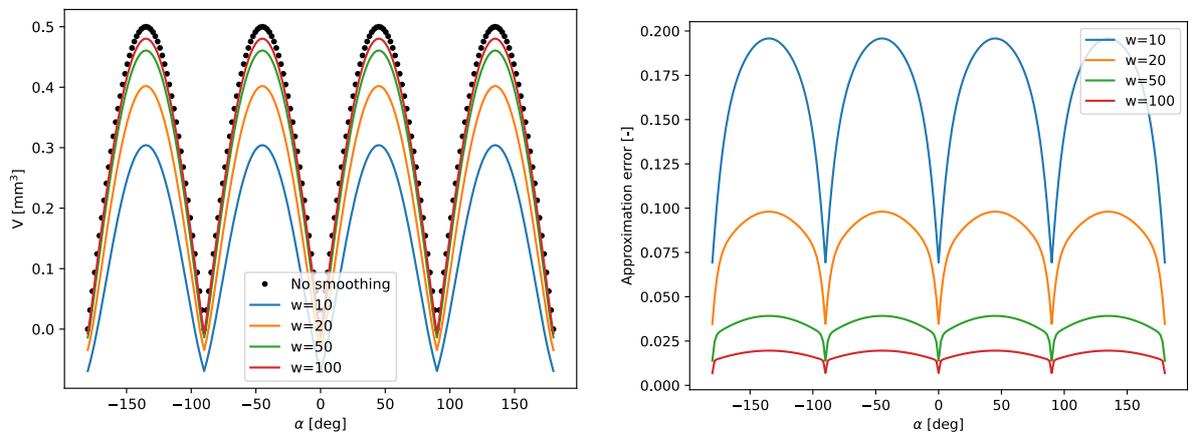
(a) Resulting support volume approximation for different values of w.

(b) Approximation error relative to non-smoothened support volume for different values of w.

Figure 3.8: Effect of using different values of weighting parameter w for obtaining a smooth minimum approximation using mellowmin approximation.
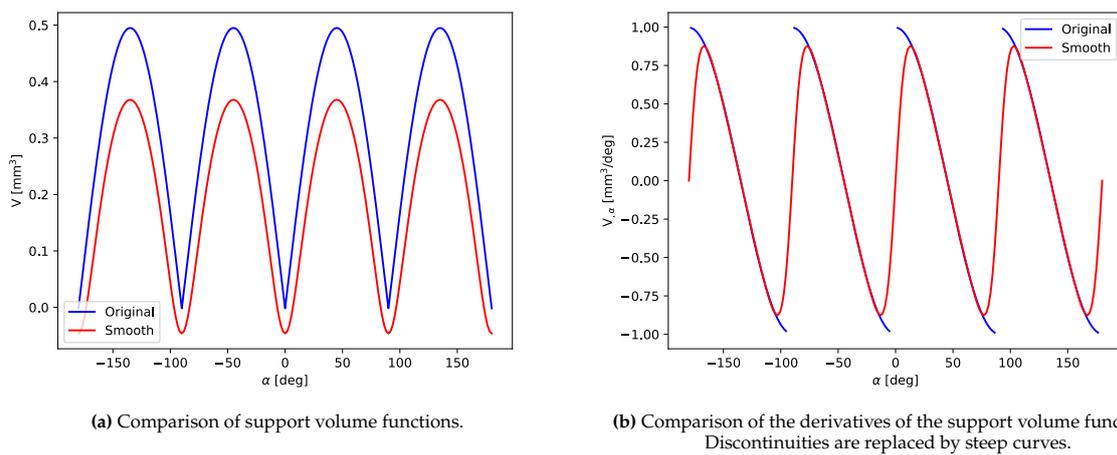
(a) Comparison of support volume functions.

(b) Comparison of the derivatives of the support volume functions. Discontinuities are replaced by steep curves.

Figure 3.9: Comparison of support volume, with and without smoothing strategies implemented. For illustrative purposes, low smoothing parameters are used, resulting in a big difference in calculated support volume. However, the locations of stationary points remain equal.

4

Calculating support volume for non-convex shapes

The methodology presented in the previous chapter revolved exclusively around convex shapes. The principles and concepts discussed form the foundation for this chapter, which considers non-convex shapes. A shape such as in Fig. 4.1 presents a new challenge: rather than the build plate, other facets may now also provide a basis for support. This implies the need to determine whether the build plate, an self-supporting facet of the part, or a combination of both provide the support. This chapter presents a novel method to make support volume calculations suitable for non-convex shapes. Similar to Section 3.3, the implementation of smoothing strategies is discussed as well. Furthermore, an extension of the cost function is demonstrated and the derivative calculations are verified. For sake of simplicity, consider all facets whose normal vector points opposite of build direction $\mathbf{d}$, i.e. $\mathbf{n}_i \cdot \mathbf{d} < 0$, to require support.

The general approach of calculating the required support volume using Eq. 3.21 remains unchanged. To incorporate support on part, inspiration is drawn from the work by Ezair et al. [10] and Fig. 1.2. The volume below some facets can be subtracted rather than added to the total support volume. Ezair et al. [10] additionally proved that subtracting volumes may provide valid support volume estimations. Fig. 4.1 presents a graphical example of the proposed concept. An overhanging facet i is projected onto the build plate, and its support volume is added. However, this support volume is overestimated, since the part feature containing facet j is also considered in the computation. The volume below facet i intersects the overall shape. To compensate for this, the volume below facet j should be subtracted from the initially computed support volume to arrive at the actual desired support volume between facets i and j. Observe that additional support may be required further below facet j. However, this should not be calculated when considering the support of facet i, but for any other overhanging facet below. Therefore, support volume is only computed between either an overhanging facet and the first intersected self-supporting facet, or between an overhanging facet and the build plate.
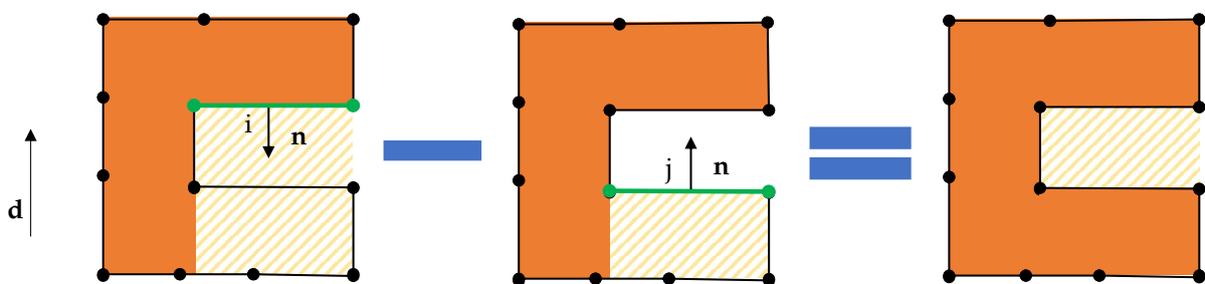
Figure 4.1: Graphical illustration of subtracting the volume below a self-supporting facet j to obtain the actual support volume below facet i. The calculated support volume below facet j intersects the part containing facet j. Subtraction of the support volume below facet j compensates for the overestimation in required support volume.

(a) The green facet is pointing downwards, and thus requires support.

(b) The green facet is self-supporting, and thus requires no support.

(c) The green facet i is self-supporting and forms the basis for support on part for overhanging facet j (blue).
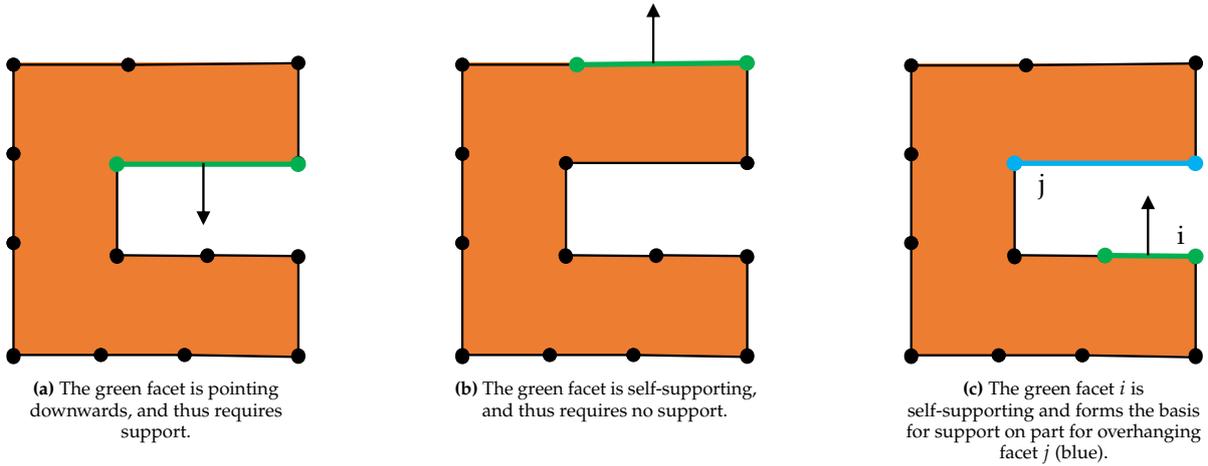
Figure 4.2: Schematic overview of the three possible states that a mesh facet can attain. The direction of the facet normal vector and the presence of overhanging facets above determine the state.

To implement this strategy, two aspects need further elaboration: first of all, a set of conditions is required to determine whether the volume below a facet should be added to the total support volume or subtracted from it. Secondly, a method to determine the actual contribution of each facet to the total support volume is necessary. The following sections discuss these aspects.

4.1. Conditions for support on part

To determine the conditions for support on part, Fig. 4.2 presents the three states of a mesh facet. These states are exclusive: a facet cannot attain two states in the same orientation. The conditions for each state are:

1. The facet is overhanging, its normal vector is pointing opposite of build direction **d**. The facet requires support.

2. The facet is self-supporting, its normal vector is pointing in the direction of build direction **d**. The facet does not require support.

3. The facet is self-supporting, its normal vector is pointing in direction of build direction **d**. However, there are other facets in direction of **d** which require support. Therefore, the facet forms a basis for support on part.

The first two states only require information about the facet of interest: a facet is classified as either requiring support or self-supporting based on the dot product between the facet normal **n** and build direction **d**. If the facet requires support, the associated support volume can be calculated using Eq. 3.24. If it does not require support, no further calculation is necessary. For the third state however, non-local information regarding other facets is required: for facet i to provide a base of support on part, facet j needs to be above facet i without any part features between them, and facet j must require support, i.e. overhanging, see Fig. 4.2c. This implies that the normal vectors $\mathbf{n}_i$ and $\mathbf{n}_j$ must be pointing toward each other.

This last observation can be used conveniently to identify where support on part may take place, and between which facets: while looping over all facets, check whether the normal vector of the current facet i is pointing to any of the other facets j. This is done using

$$w_{ij} = (\mathbf{c}_j - \mathbf{c_i}) \cdot \mathbf{n}_i , \tag{4.1}$$

where $\mathbf{c}_j$ and $\mathbf{c}_i$ denote the center coordinates of facets j and i respectively. A ray tracing method can then be used to verify that there are no other facets between facets i and j. If w_{ij} is positive, the facet normal vectors are pointed towards each other. Thus, facet j is a candidate to support facet i in some orientation. Performing this for all facets results in a connectivity set of facets that can provide support
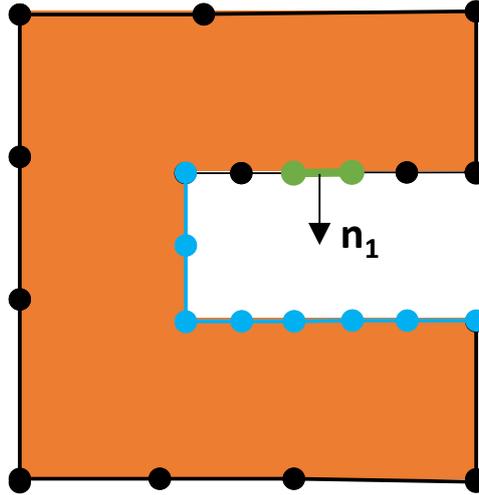
Figure 4.3: Example of a connectivity set of a particular facet (green). All blue facets are candidate to provide on-part support for the green facet in some orientation. The other facets (black) will not provide a base for support on part to the green facet in any orientation.

for facet i. Fig. 4.3 presents a connectivity set w_{1j} for an individual facet. Any of the blue facets can provide support to the green facet in some orientation.

The connectivity table is an inherent property of the parts' shape, independent of its orientation. This implies that the connectivity of the entire part can be computed once, before the optimization. The object's orientation determines which of the facets in the connectivity set will indeed provide support. Lastly, some facets will never provide support to other facets, regardless of orientation. Their connectivity set is therefore empty.

4.2. Computing connectivity weights

The previous sections outlined the necessary conditions for support on part to occur. This section discusses the implementation of these conditions into the support volume calculation presented in Section 3.3. Based on the conditions presented in Section 4.1, the volume below an individual facet is either added or subtracted. Fig. 4.1 indicates that the volume below facets needs to be subtracted only if the facet is self-supporting. For those facets, Eq. 3.16 returns a negative area value, thereby already implementing a mechanism to subtract volumes.

The state of a facet can be determined based on the facet normal, as in Fig. 3.5. Such discrete fields are less common compared to continuous, but find applications in representing geographical data for example [63]. Not considering smooth approximations for now, let D denote the field of overhanging facets, with $\mathbf{n} \cdot \mathbf{d} < 0$. Conversely, U is the field of self-supporting facets with $\mathbf{n} \cdot \mathbf{d} > 0$.

In Eq. 3.24, a mask M is implemented to only consider overhanging facets for the calculation of support volume. M can be extended to implement the conditions for on-part support. If the facet i requires support, $D_i = 1$ and the calculated area is positive. Its mask value M should be 1 to increase the total support volume. If the facet is self-supporting and not a base of support on part, it should not affect the total support volume. Its mask value M should then be 0.

For facet i to be a base of support on part, $D_i = 0$ and $U_i = 1$. Additionally, its connectivity set is not empty, and D_j should be 1 for some facet(s) j in the connectivity set. This implies that support is required for facet j. Being self-supporting, the projected area A_i^p is negative according to Eq. 3.16. A positive mask value for facet i will then decrease the total support volume. In general, the values in mask M need to be larger than zero for a correct computation of the support volume. Table 4.1 summarizes the possible states of a facet and the implications on its contribution to the total support volume.

Table 4.1: Overview of all possible states of a facet and evaluations of different fields and the resulting support volume contribution for each state.

State	$\mathbf{n} \cdot \mathbf{d}$	D_i	U_i	Connectivity set	M_i	$M_i A_i$	Volume contribution
Supported	<0	1	0	N.A	1	>0	>0
Support on part	>0	0	1	not empty	$0 < M_i < 1$	<0	<0
No support	>0	0	1	empty	0	0	0

Based on this analysis, the mask M of facet i can be expressed as a function of the up- and downward fields of facet i and the downward field of facets j in the connectivity set, such that

$$M_i = D_i + \frac{U_i}{k} \sum_{j=0}^{k} v_j D_j \; . \tag{4.2}$$

The subtraction should be proportional to the number and relative location of the facets in the connectivity set. To achieve this, all facets j in the connectivity set are multiplied with a weighting

$$v_j = -\mathbf{d} \cdot \frac{\mathbf{c}_i - \mathbf{c}_j}{|\mathbf{c}_i - \mathbf{c}_j|} \; , \tag{4.3}$$

which indicates the alignment of the center-to-center vector between facets i and j, and the build direction vector $\mathbf{d}$. If facet j is directly above facet i, facet i will provide much support to facet j. Therefore, the contribution weight v_j is large. On the other hand, not much contribution should be computed if the vector between facets i and j is nearly orthogonal to d, and thus v_j is small.

As this angle is only relevant for self supporting facets, a minus sign is included to obtain the acute angle between the two vectors. The vector $\mathbf{c}_i - \mathbf{c}_j$ always points away from build direction d. The derivative of Eq. 4.3 with respect to α is computed similarly to Eq. 3.7 as

$$v_{j,\alpha} = -\mathbf{d} \cdot \left(\mathbf{R}_{,\alpha} \frac{\mathbf{c}_i^0 - \mathbf{c}_j^0}{|\mathbf{c}_i^0 - \mathbf{c}_j^0|} \right) \; , \tag{4.4}$$

with $\mathbf{c}^0$ denoting a facet center coordinate in un-rotated initial orientation.

The weights v_j are multiplied with switch D_j in Eq. 4.2, which is 1 if facet j is pointing downward. This ensures only downward-facing facets are included in the summation. Division by the number of facets in the connectivity set ensures that M_i will attain a value between zero and one.

The result of the sum is only relevant if facet i is a base of support: if a facet is overhanging, $U_i = 0$ and the evaluation of the sum is discarded. Alternatively, a facet can be self-supporting, but its connectivity set is empty. The sum then evaluates to 0. For such a facet, $D_i = 0$ and thus $M_i = 0$, resulting in no contribution of that facet to the total support volume, corresponding to Table 4.1.

4.3. Continuous formulation for support on part

The previous section presented a strategy to determine whether volume below a facet should be added, subtracted, or ignored for the summation of the total support volume. The classification is based on a set of switches, a combination of up- and down field values of the facets. For sake of simplicity, these switches were assumed to be discrete functions in the previous section. However, the aim of this thesis is to obtain a smooth volume function, and thus these switches need to be replaced by differentiable equivalents. This can again be achieved using a Heaviside approximation. The down field D is then calculated as

$$D = H(-n_z, k, t_d),$$
(4.5)

and the up field U becomes

$$U = H(n_z, k, t_u),$$
(4.6)

using the z-component of the facet normal vectors, smoothing parameter k, and individual thresholds for up- and downward fields t_u and t_d. Derivatives are computed similarly to Eq. 3.28. These field derivatives are required for the derivative of the mask presented in Eq. 4.2, resulting in

$$M_{i,\alpha} = D_{i,\alpha} + \frac{U_{i,\alpha}}{k} \sum_{j=0}^{k} v_j D_j + \frac{U_i}{k} \sum_{j=0}^{k} v_{j,\alpha} D_j + \frac{U_i}{k} \sum_{j=0}^{k} v_j D_{j,\alpha},$$
(4.7)

which is then used in Eq. 3.25 to compute the total derivative of the support volume. Inserting Eqs. 4.5 and 4.6 into Eq. 4.2, and summing over all n facets using Eq. 3.24 results in a smooth evaluation of the support volume for non-convex shapes.

Applying Eqs. 4.5 and 4.6 to a mesh object is presented in Fig. 4.4. Facets can now also attain other colors on the scale instead of just yellow and purple, as shown in Fig. 3.5. This occurs when the facet is nearly vertical. These green facets form a transition layer rather than a sharp boundary between up- and downward facets.

(a) Upward field.   (b) Downward field.
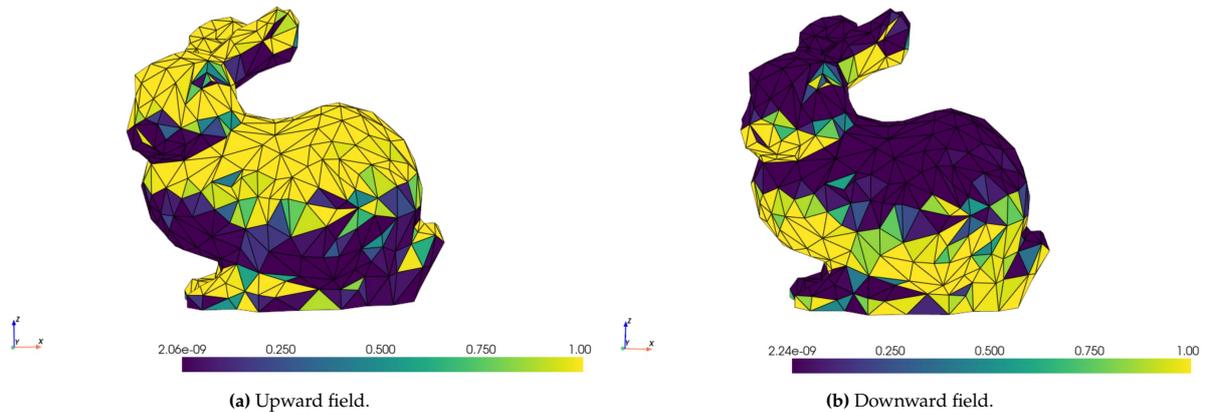
Figure 4.4: Up- and downward fields applied to a mesh bunny using Heaviside approximation on the facet normal vector. The (near) vertical green facets form a gradual boundary between up- and downward facing facets.

A comparison with the volume calculation for convex shapes used in Section 3.3 is presented in Fig. 4.5. A cube with a cutout in Fig. 4.5a is used for the comparison. The "naive" implementation of Eq. 3.21 on the non-convex shape results in an overestimation of support volume in Fig. 4.5b. Most noticeable is the difference around a 0-degree rotation angle, where now the on-part support volume indicated in Fig. 4.1 is present. The stationary points remain equal for both methods, indicating that no additional artifacts are introduced. Around a rotation of ±90 degrees about the x-axis, no on-part support is required, and both implementations yield the same result.

(a) A cube with a cutout.

(b) Comparison of support volume using convex and non-convex calculation methods.
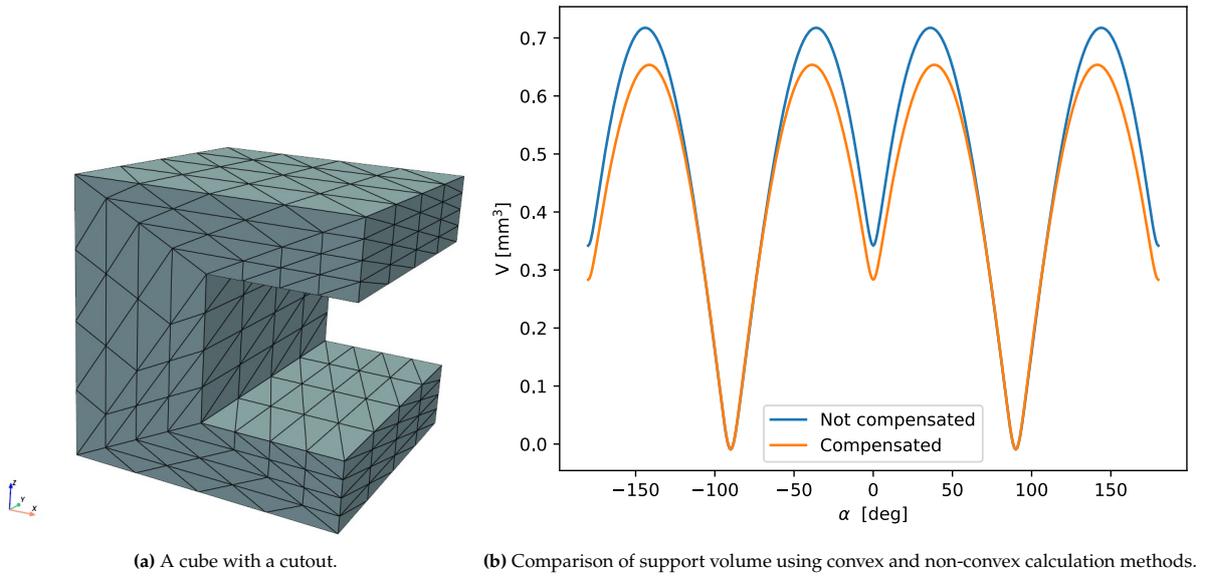
Figure 4.5: Comparison between support volume for a cube with a cut-out, using the method for convex shapes vs. applying connectivity sets to determine the required support on part.

4.4. Cost function extension

In essence, the up- and downward fields in Fig. 4.4 indicate not only support requirement, but also the amount of interface area between the supports and the part. Depending on the AM process, post-processing efforts to meet surface quality standards are very labor-intensive. The interface area between the support structure and the part can be penalized. The area of facets that is in contact with support structures A_s is

$$A_s = \sum_{i=0}^{n} M_i A_i \, ,$$ (4.8)

with A_i denoting the facet area and M_i being the mask defined in Eq. 4.2. Both the mask M and facet area A are already calculated and used for the support volume calculation, so little extra computational effort is required to calculate A_s.

Corresponding to the work of Griffiths et al. [22], part cost is selected as objective to minimize in this thesis. Until now, only support volume is considered in the computation of part cost. However, it is easily extended with the support interface area A_s using a weighted sum of the form

$$\text{Cost} = aV_s + bA_s \, ,$$ (4.9)

where V_s is the support volume using Eq. 3.24. Parameters a and b are user-selected weights. For example, post-processing is commonly less labor-intensive in FDM compared to SLM. b can then be set to a lower value. If material cost is a dominant factor, a can be increased relative to b. This can reduce material consumption, although possibly at the expense of increased post-processing requirements. Essentially, deciding on weights a and b affects the amount of on-part support used. Increased use of support on part reduces V_s but increases interface area A_s. Conversely, heavier penalization of the interface area will result in an orientation with little on-part support.

Because the support volume requirements are calculated for each individual facet, the cost of printing can be presented for each facet as well. This can provide the designer with a clear indication of which features of the part are expensive to manufacture. If feasible, design changes can be implemented and a new optimum may be obtained for the next iteration. An example is presented in Fig. 4.6. For two sets of cost weights a and b, the part cost is plotted per facet. With $a = 1$ and $b = 0$, only the cost of

(a) $a = 1$ and $b = 0$.   **(b)** $a = b = 0.5$.   **(c)** $a = 0$ and $b = 1$.
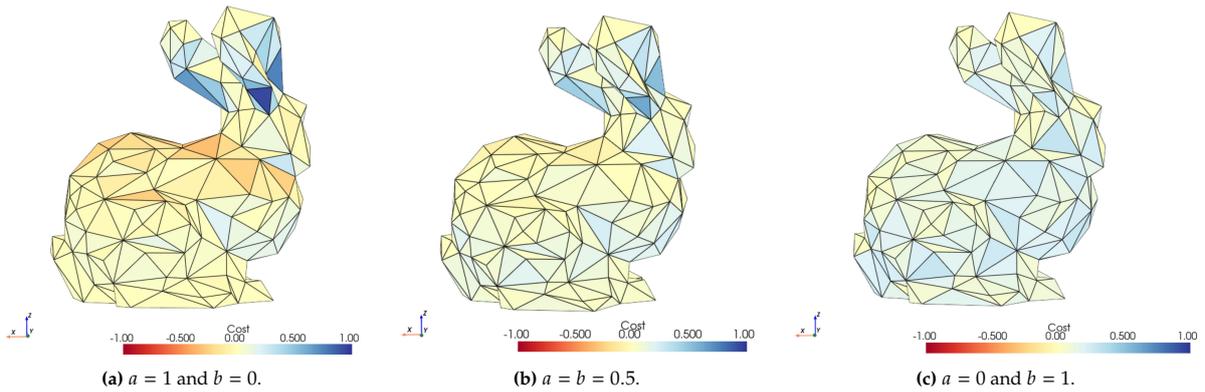
Figure 4.6: Part cost visualized for a mesh bunny using different weights a and b. Blue facets indicate an increase in cost, whereas red facets decrease the total part cost.

support volume is considered, resulting in Fig. 4.6a. Blue facets indicate support requirement, whereas (slightly) red facets act as a base of support. For $a = 0$ and $b = 1$, the cost is presented in Fig. 4.6c. Facets that were red before are now blue as well, because all facets interfacing with the support structure now increase the cost function. Using $a = b = 0.5$, the cost of support volume and interface area is penalized equally in Fig. 4.6b. Some of the facets on the back of the bunny are still selected for on-part support, indicated by the light red shading. Compared to Fig. 4.6a, more facets are colored blue to penalize the interface between support structures and the part.

Using a weighted sum in Eq. 4.9 provides a framework which can be easily extended with other cost considerations. Modules for optimizing the surface roughness and mechanical requirements are easily appended. A clear benefit of using a weighted sum is that the derivative of the cost function does not increase in complexity with more modules added: the total derivative remains the sum of the derivatives of each of the individual modules.

It should be noted that expanding the objective with more quality requirements is a research field on its own. Weighting of each objective remains a topic of discussion in the literature [7]. This thesis only focuses on a single objective; support volume. However, the mathematical approach may be used for other metrics as well. This section mainly serves as an indication of how this work fits in the context of other research and how separate works in the literature may be combined.

4.5. Verification of gradients

A significant portion of this thesis revolves around obtaining accurate sensitivity information. This section concludes the chapter by discussing the validity of these gradients. Considering the cube with cutout in Fig. 4.5a as a test case, the relative error between the analytic gradient $V_{,\theta}$ and finite forward difference $\tilde{V}_{,\theta}$ is calculated as

$$\text{error} = \frac{|\tilde{V}_{,\theta} - V_{,\theta}|}{|V_{,\theta}|} \ . \tag{4.10}$$

The error is calculated for a range of step sizes around the point $[30, -30]$ degrees, which is not near any of the stationary points and requires calculation of on-part support structures. The results are presented in Fig. 4.7, where it is clear that up to a step size of 10^{-4}, the finite differences match the calculated gradients. For smaller step sizes however, the error rapidly increases due to increased round-off errors in the single precision calculations used by the PyVista package.
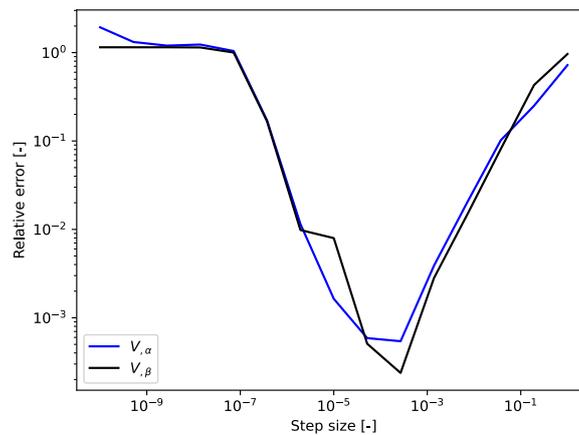
Figure 4.7: Relative error between analytic gradients and finite forward differences around $[30, -30]$ degrees for a cube with cutout. Derivatives are calculated for both rotation angles α and β. With the use of single precision coordinates, the increase in round-off error for step sizes smaller than $1e-4$ is as expected.

<div style="text-align: right; font-size: 4em;">5</div>

Numerical experiments

The previous chapters presented a novel approach for obtaining a smooth approximation of support volume. This chapter aims to demonstrate this approach by applying it to a real world shape. First, the functionality of some of the parameters presented in the previous chapters is discussed. Next, several gradient descent strategies are compared on performance. The best strategy is then compared with a genetic algorithm and the results are discussed. The source code used is publicly available on GitHub [1].

The part considered is presented in Fig. 5.1. This armadillo figure is a common shape for testing numerical and optimization methods in the literature. It is decimated to 400 facets, which aids in reducing computational cost as well as making the individual facets more visible. With its suspended arms and tail, a significant amount of support structures is expected if the shape is not in an optimal orientation. In the un-rotated configuration, 77 mm^3 of support volume is required.
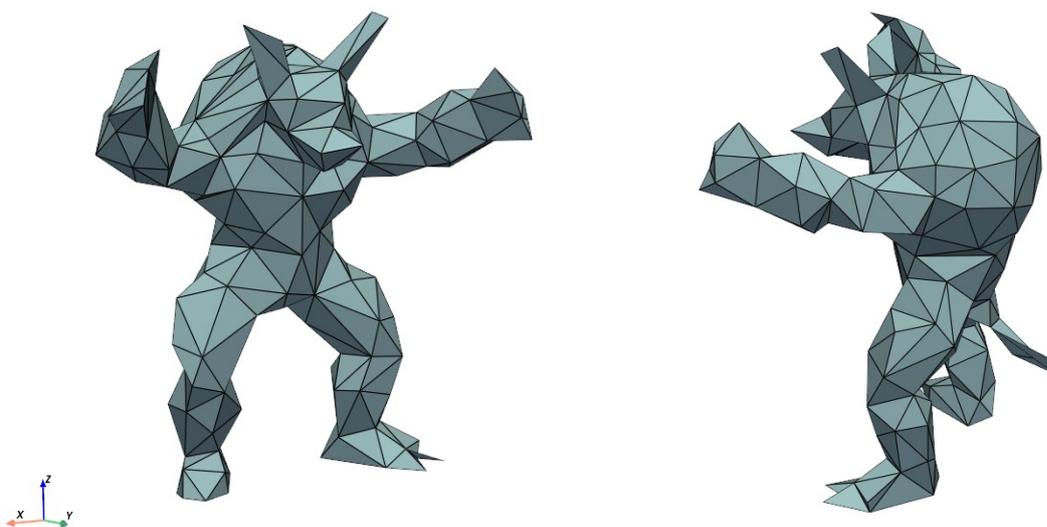
Figure 5.1: Two views of the decimated armadillo figure used to highlight the smooth support volume calculations. The model consists of 400 facets.

5.1. Effects of model parameters

To gain more insight into the presented model, a number of parametric studies are presented. First, the threshold angle in Eq. 4.5 is considered. As shown in Fig. 4.4, facets are considered overhanging or self-supporting in a smooth manner based on this threshold value. Selecting the appropriate threshold

---

[1]https://github.com/JasperBoogers/Thesis

thus significantly alters the support volume function, as shown in Fig. 5.2a. The optimum orientation may also change for an increased threshold angle. Overall, the resulting volume function decreases for higher threshold angles, which is as expected.

In addition to overhang threshold angles, the effects of various self-supported thresholds are investigated. Intuitively, a (nearly) vertical face cannot provide much base of support. Selecting a higher threshold angle may then provide more realistic results. Fig. 5.2b shows that this threshold value does not significantly affect the resulting volume function. This is an indication that op-part support only occurs in specific orientations. For example, around -70 degrees, the required support volume is approximately equal for all threshold angles.
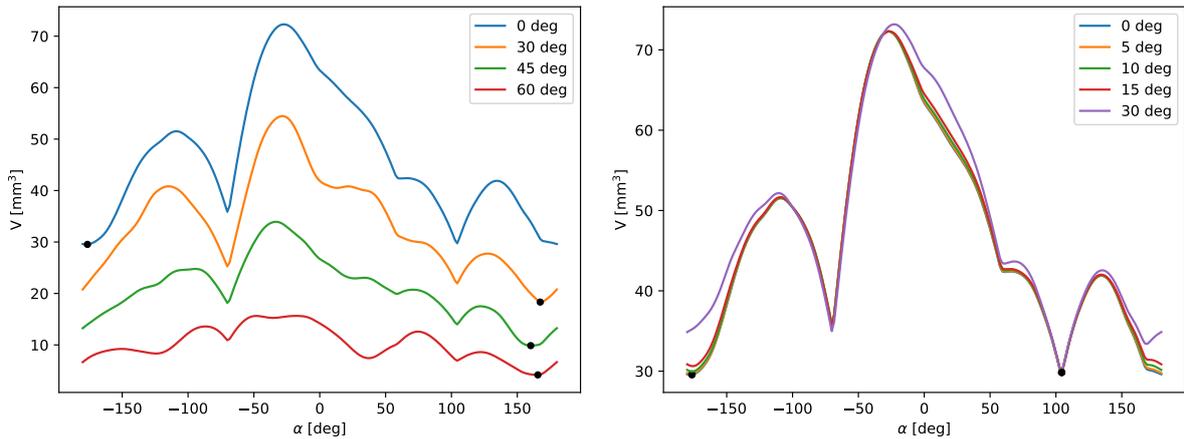
(a) Support volume approximation for varying overhang thresholds. **(b)** Support volume approximation for varying self-supported thresholds.

Figure 5.2: Effect of overhang and self-supported threshold parameters on the resulting support volume function. The optimal orientation for each parameter value is indicated by a black dot.

Conversely, a larger difference is observed around +10 degrees rotation about the x-axis. Fig. 5.3 illustrates this difference. Blue facets indicate requirement of support structures. Red facets, for example on the upper legs of the armadillo, are used to provide on-part support to blue facets above in Fig. 5.3a. Dark red facets provide more on-part support than light red facets. Similarly, dark blue facets require more support than light colored facets. As the self-supporting threshold angle increases, fewer self-supporting facets can serve as a basis of support, leading to a reduction in the number of red-colored facets in Fig. 5.3b. In combination with a negative area calculation for self-supporting facets, the total support volume increases. In other words, the support volume calculation is less compensated for the fact that the support intersects the part, as indicated by Fig. 4.1.

5.2. Effects of mesh size

This section studies the effects of mesh size. The cube with cutout presented in Fig. 4.5a and the armadillo figure in Fig. 5.1 are used as test cases. Two metrics are of interest. First, the time scaling of computing the connectivity sets presented in Section 4.1 is investigated. When calculating the connectivity sets, each cell is looped over twice. Theoretically, the time complexity is then of $O(n^2)$. This is confirmed by the slope of approximately 2.2 in Fig. 5.4a, where the computation time for generating the connectivity is presented. Calculating connectivity thus becomes expensive for larger models. Fortunately, it only needs to be calculated once before the start of the optimization.

Next, the time complexity of calculating the required support volume for a single orientation is of interest. Fig. 5.4b presents the computation time of support volume for an increasing number of facets. Again, the time complexity is of exponential form.

With decreasing mesh size, the calculated support volume is expected to converge. Fig. 5.5a presents the obtained support volume for an increasing number of triangle facets. Although the convergence limit is not reached, a decrease in slope is observed for an increasingly finer mesh size. As mentioned before, an exact calculation of the support volume may not be required to obtain an optimum orientation [25].

(a) 0 degree threshold angle.

(b) 45 degrees threshold angle.
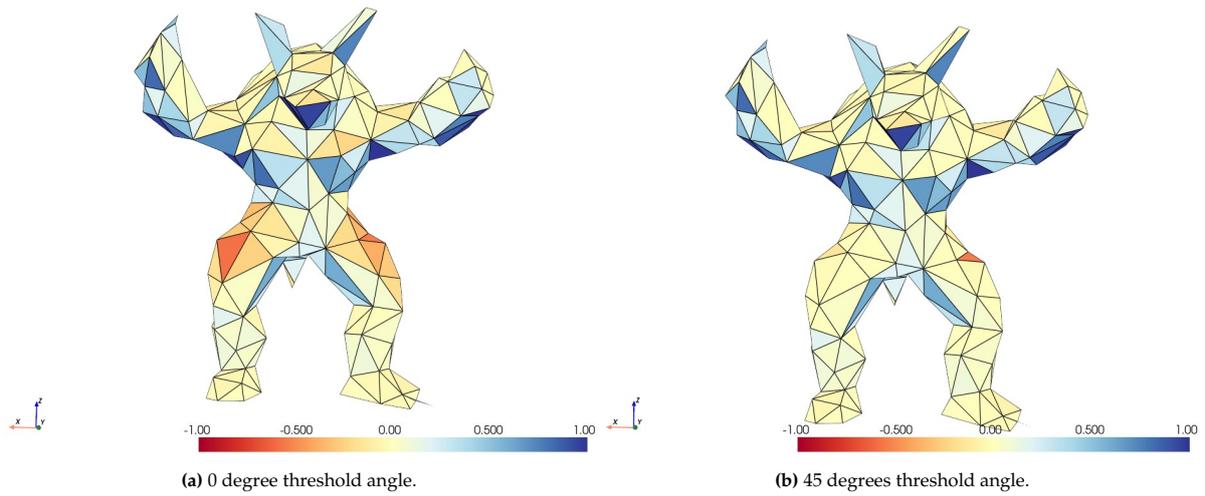
Figure 5.3: Comparison of support volume calculation for two different self-supported threshold angles. The orientation is equal for both models. Less self-supported facets are selected as basis of support on the legs of the armadillo (in red), resulting in increased support volume. Overhanging facets (in blue) are not affected.

(a) Computation time of generating connectivity sets.

(b) Computation time of calculating support volume in one orientation.
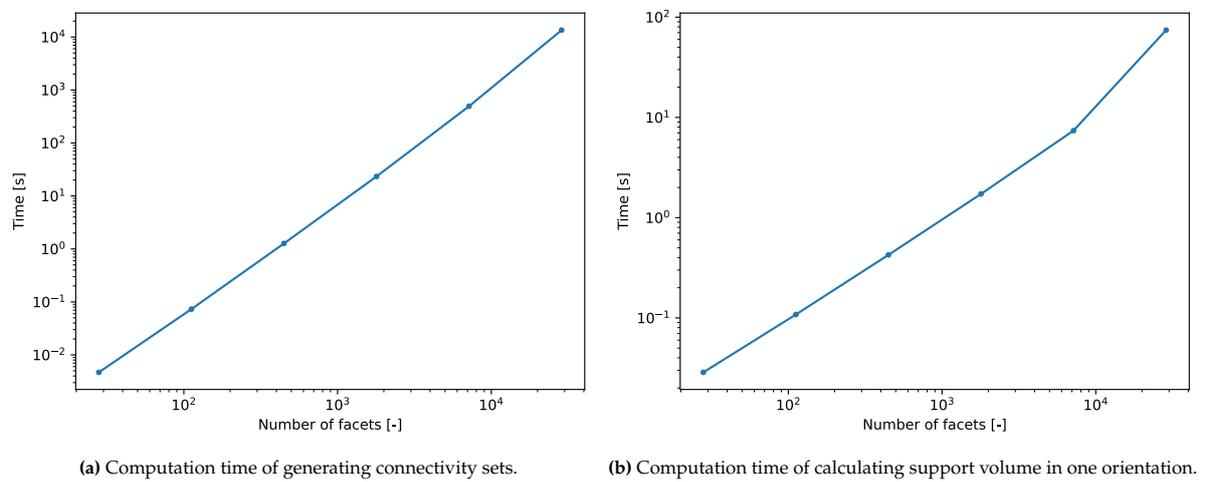
Figure 5.4: Time scaling for generating connectivity sets and calculating support volume for increasing number of mesh elements, using the cube with cut-out in Fig. 4.5a.

However, a smaller mesh size may be necessary based on required feature size to be captured.

(a) Using the cube with cut-out in Fig. 4.5a.

(b) Using the armadillo in Fig. 5.1.
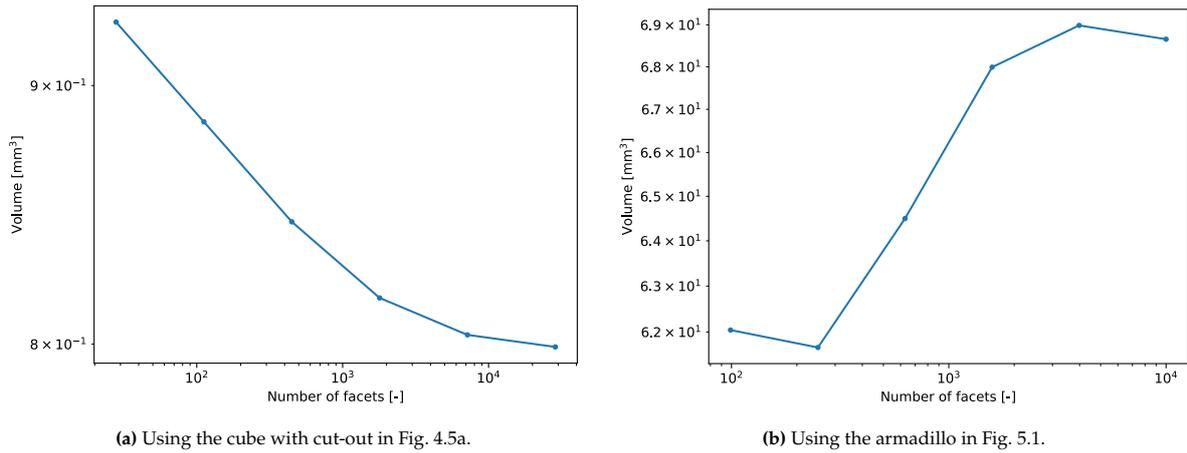
Figure 5.5: Calculated required support volume in a specific orientation, for increasing number of mesh elements.

The effect of mesh size on support volume calculation repeated using the armadillo shape in Fig. 5.1. Initially, Fig. 5.5b presents an increase in volume before a decrease in slope. Compared to the cube in Fig. 4.5a, the armadillo's shape is much more complex. The minimum number of facets to somewhat accurately describe the shape is therefore much higher. Again, the calculated volume does not converge for the applied number of mesh elements. The cost of computation limited the maximum number of mesh elements used.

5.3. Comparison of optimization strategies

The aim of this thesis is to obtain a smooth approximation of the support volume. This section presents a comparison on how different gradient-based optimization strategies treat the obtained smooth volume function. An appropriate starting point for is selected based on the iso-contour plot in Fig. 5.6. The overhang threshold is set to 90 degrees, such that all overhanging facets require support. The point $[-90, -70]$ degrees is selected. Additionally, the optimization is performed both using analytic gradients and first order finite differences. An overview of the optimization results is provided in Table 5.1. For the BFGS and Conjugate Gradient (CG) methods, providing the optimizer with analytic gradients clearly aided in finding an optimum. When gradients are supplied, the number of evaluations required before termination is significantly. The same optimum was obtained using the Newton-CG method, although the gradient is still somewhat large at that point. Overall, only a few evaluations were required to obtain a minimum. When no analytic derivatives are supplied, the required number of evaluations significantly increases. This is partly attributed to the fact that the optimization did not terminate successfully.
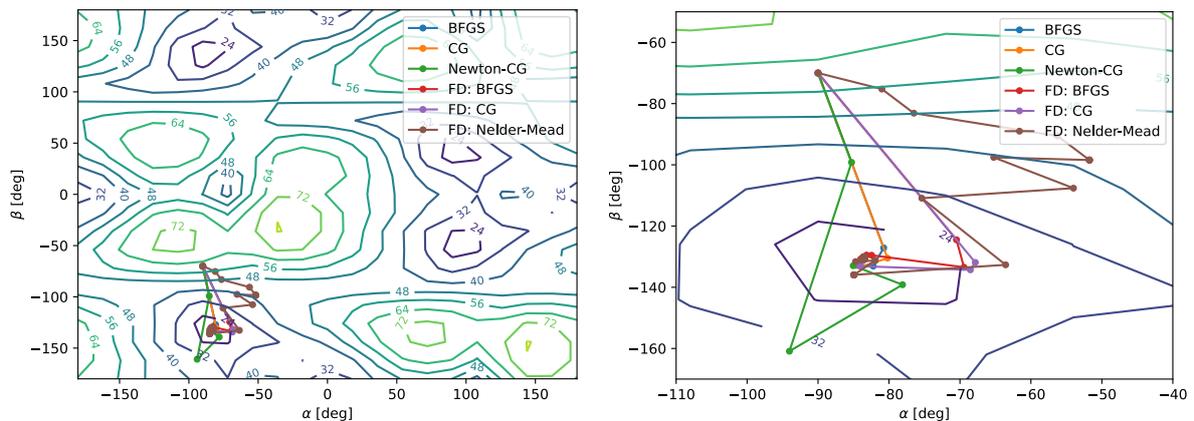
Figure 5.6: Iso-contour plot of the support volume of an armadillo figure. For a range of optimization methods, the descent path is plotted. FD indicates that first order finite differences were used instead of analytic gradients. On the right is a close-up of the relevant basin.

Table 5.1: Comparison of gradient based optimization methods. For every method, the same initial point $\mathbf{x} = [-90, -70]$ deg is used. The number of evaluations denotes the total amount of function calls during optimization.

Method	Gradients provided	Terminated Successfully	f_{min} [mm^3]	$\mathbf{x}^*$ [deg]	∇f [mm^3/deg]	# of evaluations
BFGS	Yes	Yes	15.050	$\begin{bmatrix} -83.582 \\ -130.82 \end{bmatrix}$	$\begin{bmatrix} -3.5e-5 \\ -6.9e-5 \end{bmatrix}$	13
BFGS	No	No	15.052	$\begin{bmatrix} -83.584 \\ -130.96 \end{bmatrix}$	$\begin{bmatrix} -1.3e+0 \\ -5.2e+0 \end{bmatrix}$	224
Conjugate Gradient	Yes	Yes	15.050	$\begin{bmatrix} -83.582 \\ -130.82 \end{bmatrix}$	$\begin{bmatrix} -4.3e-5 \\ -3.8e-4 \end{bmatrix}$	25
Conjugate Gradient	No	No	15.055	$\begin{bmatrix} -83.688 \\ -131.08 \end{bmatrix}$	$\begin{bmatrix} 2.2e-2 \\ -1.3e+0 \end{bmatrix}$	247
Newton-CG	Yes	Yes	15.050	$\begin{bmatrix} -83.592 \\ -130.84 \end{bmatrix}$	$\begin{bmatrix} 2.5e-1 \\ -3.1e-1 \end{bmatrix}$	12
Nelder-Mead	No	Yes	15.050	$\begin{bmatrix} -83.557 \\ -130.81 \end{bmatrix}$	-	67

With finite precision arithmetic, care is required to determine tolerance levels. To account for the single precision coordinates used for this comparison, the tolerance on the gradient is set to $1e-3$. Single precision floats have a maximum of seven significant digits [64]. In Table 5.1, differences in optimum value and coordinate arise from the fifth significant digit. This indicates that the obtained values can be considered valid results.

An iso-contour plot like Fig. 5.6 requires many evaluations to generate. It is inefficient to first perform such a grid search to select an appropriate starting point. It is included here for illustrative purposes. Rather, the gradient-based optimizer can be initialized multiple times with varying starting points. While a global optimum is not guaranteed, the chance of obtaining the optimal orientation is increased with a multi-start approach. This more realistic strategy is compared to a Differential Evolution (DE) algorithm, which is a population based strategy [65]. An initial population size of 30 is used. Based on Table 5.1, BFGS is selected as the optimization method for gradient descent. Four starting points are generated using Latin Hypercube Sampling and the best result is selected as the optimum. See Fig. 5.7 for the resulting descent paths. By chance, each starting point was sampled around a different basin. Only a few evaluations were required to obtain the optimum orientation.

Table 5.2 presents a comparison of the results. Both optimization strategies reached the same basin containing the global minimum. However, the DE approach required over 3 times the number of evaluations to complete. Additionally, there is a slight difference in the final optimum between the two strategies. The Jacobian at the optimum is multiple orders of magnitude lower for the gradient descent optimization. This indicates that the gradient descent strategy is superior both in computational efficiency and obtained minimum.

Table 5.2: Comparison of optimization results using gradient descent versus a population based algorithm. Note that for the gradient descent case, the number of evaluations is the sum of four sequential runs.

Strategy	Method	f_{min} [mm^3]	$\mathbf{x}^*$ [deg]	∇f [mm^3/deg]	# of evaluations
Gradient Descent	BFGS	15.050	$\begin{bmatrix} -83.581 \\ -130.82 \end{bmatrix}$	$\begin{bmatrix} 1.3e-3 \\ 1.4e-3 \end{bmatrix}$	181
Differential Evolution	best1bin	15.068	$\begin{bmatrix} -83.824 \\ -131.32 \end{bmatrix}$	$\begin{bmatrix} -8.6e-1 \\ -3.2e+0 \end{bmatrix}$	708

It should remain clear that obtaining a global optimum is not guaranteed for any of the methods considered. The gradient descent method is heavily dependent on the starting point. A multi-start approach may help, but at the cost of increased computation cost. However, Table 5.2 indicates a big margin in the number of evaluations for several extra optimization starts. Differential Evolution on the other hand is stochastic in nature. This increases the likelihood of finding a global minimum, but is not guaranteed [65].

The optimization process is repeated using a finer mesh size. The number of facets is increased from 400 to 10000. As the optimum location is already known, the initial point is set in the same basin. The resulting volume function is 15.670 mm^3 at $[-83.547, -130.85]$ degrees, which is a 77% reduction compared to the un-rotated configuration in Fig. 5.1. The result is similar to the obtained support volume using 400 facets. The slight increase in support volume is in line with the results obtained for increased number of mesh elements in Fig. 5.5b. Fig. 5.8 presents the armadillo in the optimal orientation. Overhanging facets are indicated in blue, whereas red facets are selected as a basis for support on part. The armadillo is placed on its back to reduce the height of the required support volume. The right arm is suspended above the shoulder, such that on-part support is used to reduce the volume required to support the right arm. The support structures are extruded to the print bed in Figs. 5.8c and 5.8d.
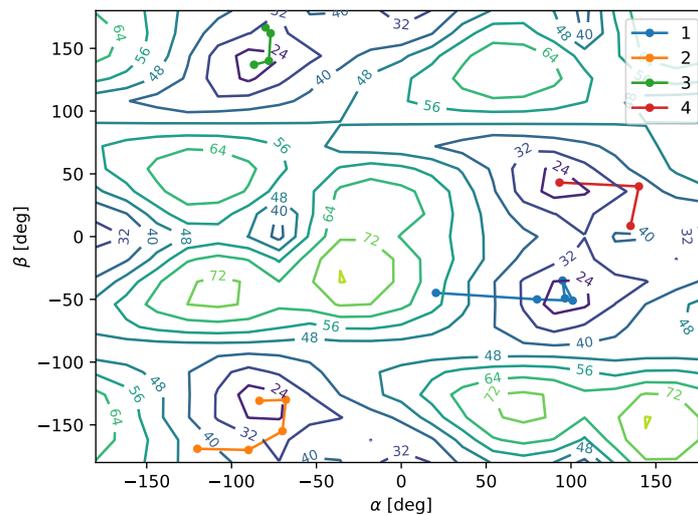
Figure 5.7: Visualization of descent paths using gradient-based optimization. Four distinct initial starting points are used, denoted by the separate colors. By chance, each starting point is sampled in a different basing using Latin Hypercube Sampling.

For a designer, this method of visualizing support requirements per facet can be useful to identify possible design changes, which may reduce the minimum required support volume even further. For the armadillo in Fig. 5.8 for example, right arm can be tilted upward such that it is vertical in this orientation. It would then require much less support structures.

For sake of comparison, the support volume can also be maximized. Fig. 5.9 presents the armadillo figure in the least optimal orientation. When rotated by $[-22.504, 25.152]$ degrees, the required support volume is 77 mm^3. The arms and upper body are suspended far above the bed and tilted forward, resulting in a high required support volume. Little on-part support occurs, as indicated by the lighter shades of red. The body and limbs are mostly supported by the print bed, as indicated by Figs. 5.9c and 5.9d.

(a) View 1.

(b) View 2.

(c) View 1 with extruded support structures.

(d) View 2 with extruded support structures.
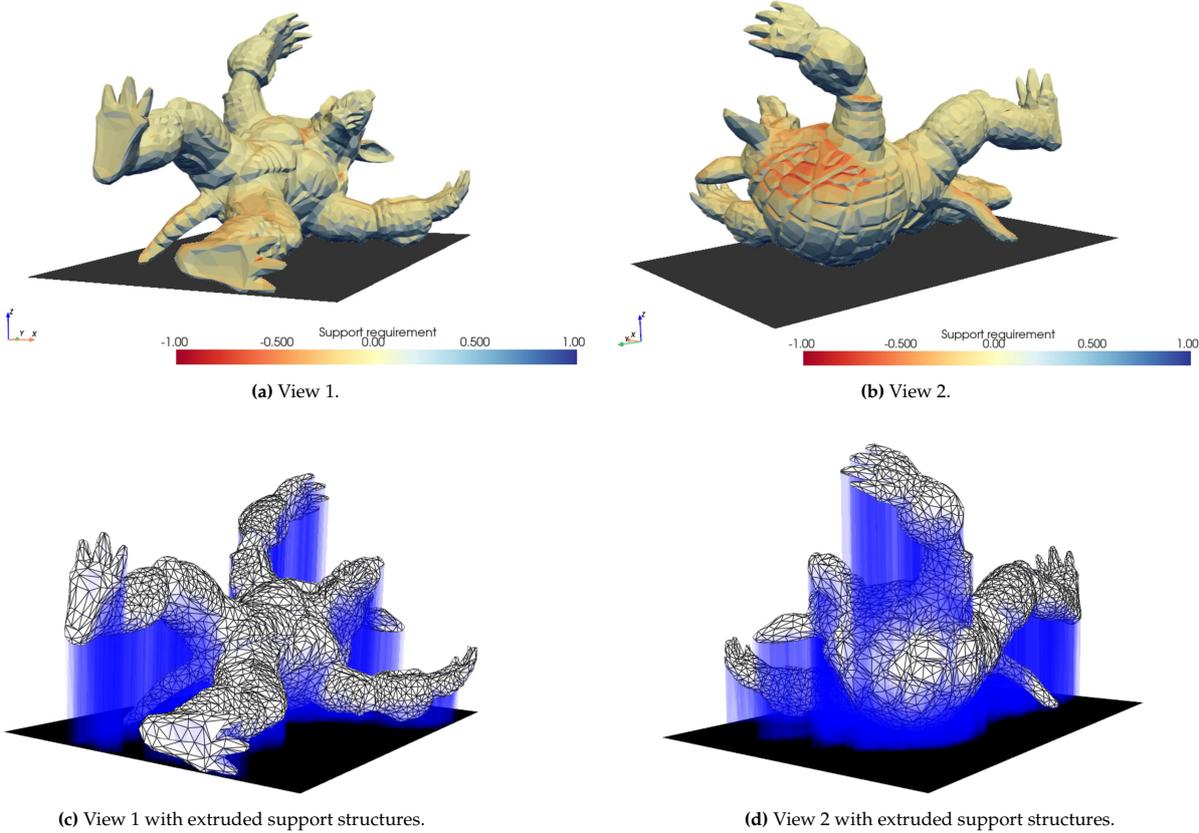
Figure 5.8: The armadillo figure in the optimal orientation of [-84, -131] degrees, with minimum support requirement of 15.1 [mm^3]. The back and the left arm and leg are placed on the bed. The right arm is suspended above the shoulder, which provides on-part support as indicated by the red facets on the shoulder.

(a) View 1.

(b) View 2.

(c) View 1 with extruded support structures.

(d) View 2 with extruded support structures.
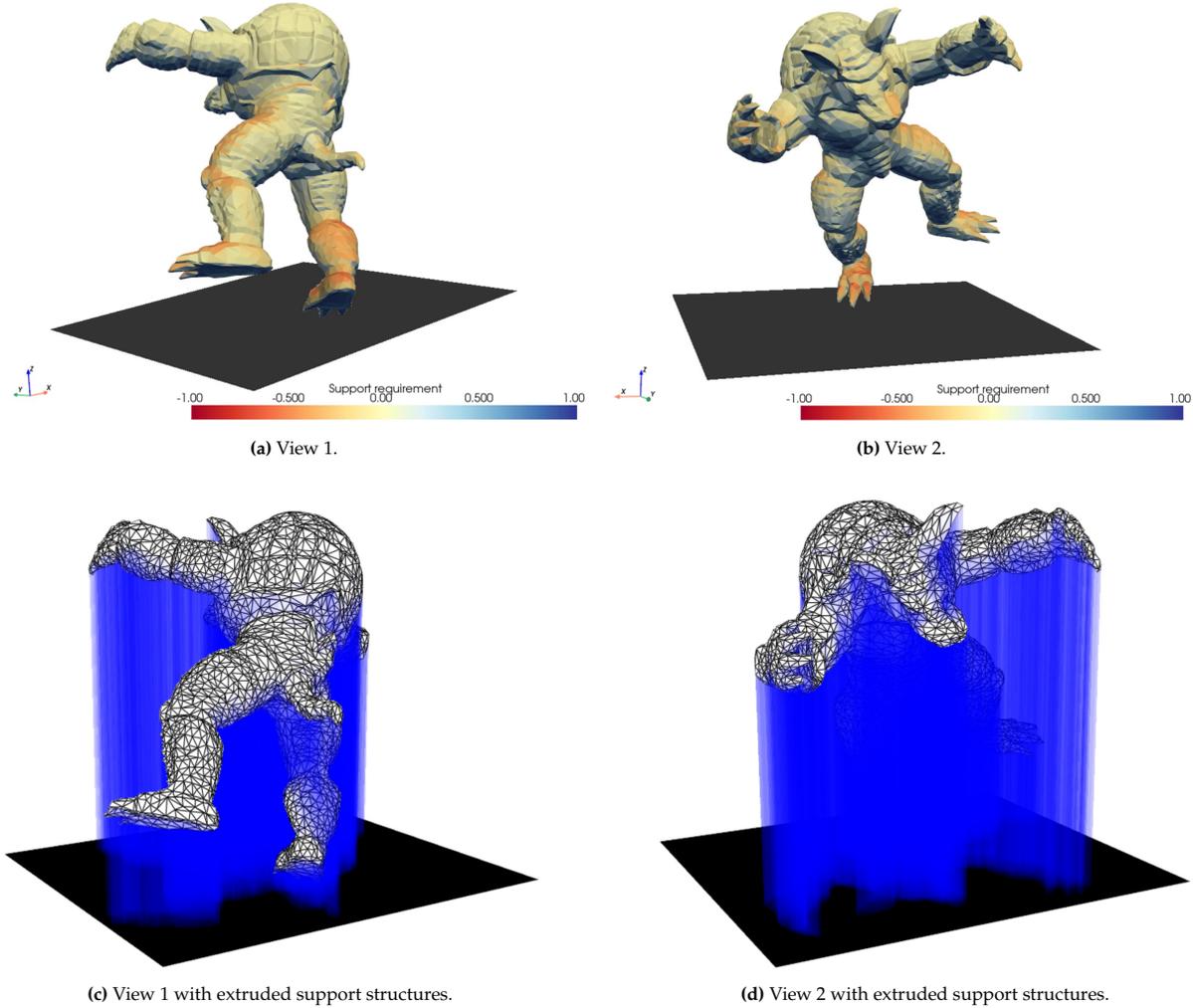
Figure 5.9: The armadillo figure in the least optimal orientation of [-22, 25] degrees, with maximum support requirement of 77 [mm^3]. Three of the four limbs are suspended above the print bed, and the body is tilted forward such that the entire chest and head require support. Less on-part support occurs, indicated by the lighter shades of red when compared to Fig. 5.8b.

6

Discussion

The presented work provides a strategy for calculating the support volume for a non-convex shape. Non-local information is required to determine whether support is provided by the build plate, the part itself, or a combination of both. Using connectivity sets allows to collect this information before optimization process starts. The previous chapter demonstrated that the presented approach is capable of calculating support volume requirements for highly non-convex shapes. This chapter presents a number of topics that require further research.

First of all, the current work does not consider any characteristics or requirements specific to a particular AM process. The presented work is a general framework, not specified to one process. Although optimization of support volume can be applied to a variety of AM processes, detailed parameter studies are required. For example, in SLS the unsintered powder may also act as support material. On the other hand, in SLM support structures serve an auxiliary function by dissipating heat and mitigating warping. In the current form, all of these requirements to support structures are not considered. They may be implemented by extending the mask used in Eq. 4.2.

Another metric that is not considered in this work is the contact area with the build platform, see Fig. 6.1. With an overhang threshold of 45 degrees, very few facets on the bunny require support structures. The contact area between the part and the build plate will be small, increasing the risk of the part falling over. Implementing a minimum required contact area can mitigate this issue. However, this may require an increase in support volume to achieve the minimum contact area.
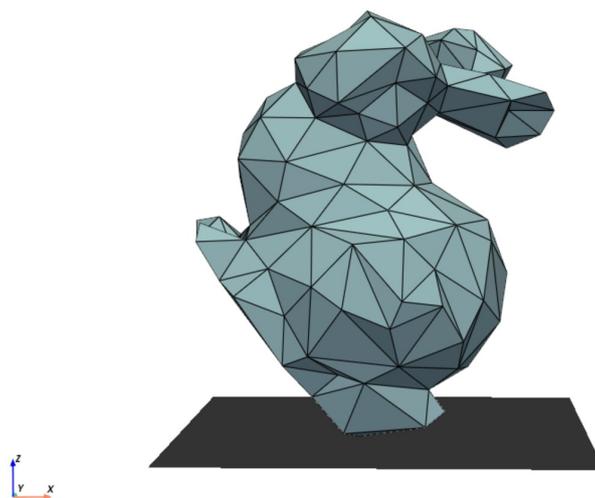
Figure 6.1: Example mesh in (locally) optimal orientation for minimum support volume. However, the small contact area with the bed increases the likelihood of toppling during manufacturing.

Designers and engineers are faced with a trade-off between part quality and cost. The presented concept of aggregating metrics in a cost function fits well in this setting. It provides the designer freedom to tune the model to match process and quality specifications. This study provides only one example of extending the cost function, and the weighting of objectives is not thoroughly studied. Many more modules can be included in the total cost model, which will again be dependent on process-specific requirements. Moreover, estimating labor costs and machine degradation is very company-specific. To maintain applicability to a wide range of users, minimizing the total production time would be more suitable. Implementing time estimator for each of the discussed modules may therefore be an appropriate direction for further research.

The use of connectivity sets allows for a smooth approximation of the support volume function. However, there are two key challenges that need further investigation. For large meshes, generating connectivity sets is a significant factor in the total computation time, even when it only needs to be generated once. There is potential for the time complexity to be reduced. For example, if facet i can be supported by facet j, the inverse is also true and does not have to be evaluated again. Improving the time complexity can aid significantly towards implementation of this approach.

Besides time complexity, using connectivity sets may have undesired sensitivity to both feature and mesh size. In essence, the mask weighting used in Eq. 4.3 not only calculates supports strictly downwards, but also slightly to the side, see Fig. 6.2a. This results in a cone-shaped support structure. For large overhanging features, this effect is averaged across many facets. It then has little influence on the resulting support volume requirement. However, for smaller features as in Fig. 6.2a, too many facets may be labeled as a base of support on part, resulting in an underestimation of the actual support volume requirement. These effects are not only dependent on shape-specific feature sizes, but also on distance between features as well as mesh size and uniformity. A thorough investigation of the effects of the weighting used in Eq. 4.2 is therefore highly recommended. One possible approach may be to implement a threshold angle in Eq. 4.3. Facets would be only considered a candidate for support on part if their facet normal vectors point more toward each other. Fig. 6.2b presents the same connectivity set but with a 45 degree threshold angle implemented. Much less facets are then included in the connectivity set of the blue facet.

An alternative method would be to square or cube the weight v_j used in Eq. 4.2. Raising v_j to some power would push weights that are smaller than one faster to zero. As a result, facets will provide less support if they are not directly below the supported facet.

Next to further investigations into connectivity sets, the impact of the presented smoothing parameters on optimization performance is not yet fully understood. These may also be dependent on mesh size, uniformity, and volume. Increasingly higher values of k in Eq. 3.27 will result in a more realistic support calculation, at the cost of steeper gradient changes near discontinuities. Its effects on the resulting optimization performance therefore requires further research.

This thesis considers a two-variable optimization problem with many local minima. This is typically the

(a) Connectivity set without threshold. **(b)** Connectivity set with a threshold of 45 degrees.
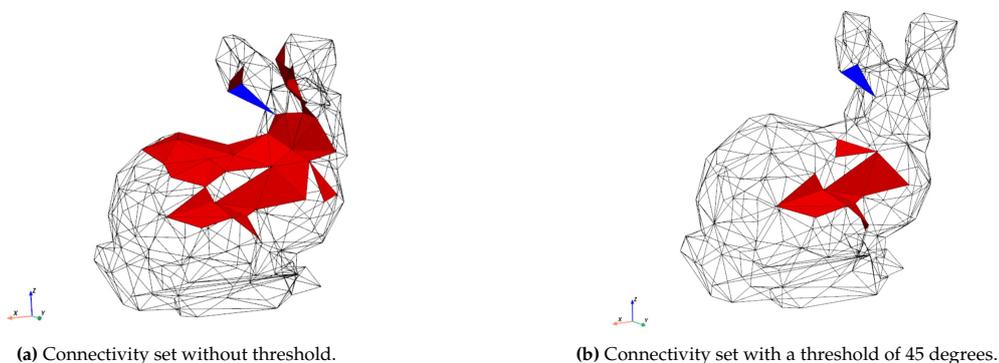
Figure 6.2: Graphical example of how using connectivity sets can result in artificial support volume requirements. Each of red colored facets contributes to supporting the blue facet on the bunny's ears. Implementation of a threshold angle when computing the connectivity sets may result in a more accurate support structures calculation.

domain of population-based algorithms, which have a higher chance of finding the global minimum. This thesis focuses on gradient-based optimization in anticipation of future research: combining the optimal orientation with refinement of the shape itself, as demonstrated by Hu et al. [31]. In their work, the optimum orientation is obtained using a grid search. This presents an opportunity for improvement and the integration of two fields of study to simultaneously optimize both shape and orientation for 3D printing. The position of each node in the mesh is represented by 3 design variables. It will then be computationally infeasible to search the entire solution space using population-based methods. Gradient-based optimization may be a feasible option, provided that sufficient gradient information is available. This may not be straightforward, as the connectivity sets used in this thesis change when the shape is modified. This implies that the connectivity sets require re-evaluation at every iteration, highlighting the importance of reducing the computational cost even more.

7

Conclusion

Additive Manufacturing allows for the production of more complex components directly from a 3D design. However, its layered nature presents a trade-off between part quality and build cost. Optimization of the required support volume can significantly improve both through reduction of material consumption, print time and post-processing efforts. Therefore, the objective of this thesis is to optimize part orientation for minimum support volume requirement.

Using triangular surface meshes, summing the support requirements for individual triangle facets results in the total support volume. For efficient computation, gradient based optimization is employed. The mathematical concepts for calculating support volume and its derivatives using smooth approximation techniques are first presented for convex shapes. Based on this approach, a novel method for calculating the support volume for non-convex shapes is presented.

The calculation of support volume is performed by adding or subtracting the volume below individual facets. Conditions to determine addition or subtraction are based on the direction of the facet normal vector with respect to the build direction. To ensure a smooth volume function, connectivity sets are generated. Combined with the conditional fields of a particular orientation, a smooth support volume function and analytic derivatives are obtained. This resultant method is not observed anywhere else in the literature. The presented concepts are implemented on a test part. The effects of smoothing and selecting user-defined overhang thresholds are discussed. It is shown that gradient based optimization outperforms population based algorithms on the smooth support volume approximation.

Optimization of part orientation for minimum support requirement is only one of many metrics to consider in AM. A framework to aggregate metrics in total part cost is proposed. Calculating objectives per facet and summing allows for process- and user-customization. Furthermore, for a given orientation, the manufacturing cost can be presented to the user for each individual facet, who can then decide on a design change. This allows for fast and easy iterations between the program and user.

References

[1]   J. Jiang, X. Xu, and J. Stringer, "Support structures for additive manufacturing: A review," *Journal of Manufacturing and Materials Processing*, vol. 2, 4 Dec. 2018, ISSN: 25044494. DOI: 10.3390/jmmp2040064.

[2]   F. Melchels, J. Feijen, and D. Grijpma, "A review on stereolithography and its applications in biomedical engineering," *Biomaterials*, vol. 31, pp. 6121–6130, 24 Aug. 2010, ISSN: 01429612. DOI: 10.1016/j.biomaterials.2010.04.050.

[3]   T. Ngo, A. Kashani, G. Imbalzano, K. Nguyen, and D. Hui, "Additive manufacturing (3d printing): A review of materials, methods, applications and challenges," *Composites Part B: Engineering*, vol. 143, pp. 172–196, Jun. 2018, ISSN: 13598368. DOI: 10.1016/j.compositesb.2018.02.012.

[4]   S. Masood, W. Rattanawong, and P. Iovenitti, "A generic algorithm for a best part orientation system for complex parts in rapid prototyping," *Journal of Materials Processing Technology*, vol. 139, pp. 110–116, 1-3 SPEC Aug. 2003, ISSN: 09240136. DOI: 10.1016/S0924-0136(03)00190-0.

[5]   V. Canellidis, J. Giannatsis, and V. Dedoussis, "Genetic-algorithm-based multi-objective optimization of the build orientation in stereolithography," *International Journal of Advanced Manufacturing Technology*, vol. 45, pp. 714–730, 7-8 2009, ISSN: 14333015. DOI: 10.1007/s00170-009-2006-y.

[6]   M. Taufik and P. Jain, "Role of build orientation in layered manufacturing: A review," *International Journal of Manufacturing Technology and Management*, vol. 27, p. 47, 1/2/3 2013, ISSN: 1368-2148. DOI: 10.1504/IJMTM.2013.058637.

[7]   L. D. Angelo, P. D. Stefano, and E. Guardiani, "Search for the optimal build direction in additive manufacturing technologies: A review," *Journal of Manufacturing and Materials Processing*, vol. 4, p. 71, 3 Jul. 2020, ISSN: 2504-4494. DOI: 10.3390/jmmp4030071.

[8]   S. Allen and D. Dutta, "On the computation of part orientation using support structures in layered manufacturing," in *International Solid Freeform Fabrication Symposium*, 1994.

[9]   H. D. Morgan, J. A. Cherry, S. Jonnalagadda, D. Ewing, and J. Sienz, "Part orientation optimisation for the additive layer manufacture of metal components," *International Journal of Advanced Manufacturing Technology*, vol. 86, pp. 1679–1687, 5-8 Sep. 2016, ISSN: 14333015. DOI: 10.1007/s00170-015-8151-6.

[10] B. Ezair, F. Massarwi, and G. Elber, "Orientation analysis of 3d objects toward minimal support volume in 3d-printing," *Computers and Graphics (Pergamon)*, vol. 51, pp. 117–124, Jul. 2015, ISSN: 00978493. DOI: 10.1016/j.cag.2015.05.009.

[11] L. D. Angelo and P. D. Stefano, "A neural network-based build time estimator for layer manufactured objects," *International Journal of Advanced Manufacturing Technology*, vol. 57, pp. 215–224, 1-4 Nov. 2011, ISSN: 02683768. DOI: 10.1007/s00170-011-3284-8.

[12] M. Matos, A. Rocha, and A. Pereira, "Improving additive manufacturing performance by build orientation optimization," *International Journal of Advanced Manufacturing Technology*, vol. 107, pp. 1993–2005, 5-6 Mar. 2020, ISSN: 14333015. DOI: 10.1007/s00170-020-04942-6.

[13] D. Wang, Y. Yang, Z. Yi, and X. Su, "Research on the fabricating quality optimization of the overhanging surface in slm process," *International Journal of Advanced Manufacturing Technology*, vol. 65, pp. 1471–1484, 9-12 Apr. 2013, ISSN: 14333015. DOI: 10.1007/s00170-012-4271-4.

[14] R. Paul and S. Anand, "Optimization of layered manufacturing process for reducing form errors with minimal support structures," *Journal of Manufacturing Systems*, vol. 36, pp. 231–243, 2015, ISSN: 02786125. DOI: 10.1016/j.jmsy.2014.06.014.

[15] P. Das, R. Chandran, R. Samant, and S. Anand, "Optimum part build orientation in additive manufacturing for minimizing part errors and support structures," *Procedia Manufacturing*, vol. 1, pp. 343–354, 2015, ISSN: 23519789. DOI: 10.1016/j.promfg.2015.09.041.

[16] P. Das, K. Mhapsekar, S. Chowdhury, R. Samant, and S. Anand, "Selection of build orientation for optimal support structures and minimum part errors in additive manufacturing," *Computer-Aided Design and Applications*, vol. 14, pp. 1–13, Nov. 2017, ISSN: 16864360. DOI: 10.1080/16864360.2017.1308074.

[17] L. Cheng and A. To, "Part-scale build orientation optimization for minimizing residual stress and support volume for metal additive manufacturing: Theory and experimental validation," *Computer-Aided Design*, vol. 113, pp. 1–23, Aug. 2019, ISSN: 00104485. DOI: 10.1016/j.cad.2019.03.004.

[18] U. Yaman, "Shrinkage compensation of holes via shrinkage of interior structure in fdm process," *The International Journal of Advanced Manufacturing Technology*, vol. 94, pp. 2187–2197, 5-8 Feb. 2018, ISSN: 0268-3768. DOI: 10.1007/s00170-017-1018-2.

[19] M. Birosz, F. Safranyik, and M. Andó, "Build orientation optimization of additive manufactured parts for better mechanical performance by utilizing the principal stress directions," *Journal of Manufacturing Processes*, vol. 84, pp. 1094–1102, Dec. 2022, ISSN: 15266125. DOI: 10.1016/j.jmapro.2022.10.038.

[20] W. Hung, "Postprocessing of additively manufactured metal parts," *Journal of Materials Engineering and Performance*, vol. 30, pp. 6439–6460, 9 Sep. 2021, ISSN: 1059-9495. DOI: 10.1007/s11665-021-06037-z.

[21] P. Alexander, S. Allen, and D. Dutta, "Part orientation and build cost determination in layered manufacturing," *Computer-Aided Design*, vol. 30, pp. 343–356, 5 Apr. 1998, ISSN: 00104485. DOI: 10.1016/S0010-4485(97)00083-3.

[22] V. Griffiths, J. Scanlan, M. Eres, A. Martinez-Sykora, and P. Chinchapatnam, "Cost-driven build orientation and bin packing of parts in selective laser melting (slm)," *European Journal of Operational Research*, vol. 273, pp. 334–352, 1 Feb. 2019, ISSN: 03772217. DOI: 10.1016/j.ejor.2018.07.053.

[23] G. Strano, L. Hao, R. Everson, and K. Evans, "A new approach to the design and optimisation of support structures in additive manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 66, pp. 1247–1254, 9-12 Jun. 2013, ISSN: 02683768. DOI: 10.1007/s00170-012-4403-x.

[24] S. Brika, Y. Zhao, M. Brochu, and J. Mezzetta, "Multi-objective build orientation optimization for powder bed fusion by laser," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 139, 11 Nov. 2017, ISSN: 15288935. DOI: 10.1115/1.4037570.

[25] A. Günaydln, A. Ylldlz, and N. Kaya, "Multi-objective optimization of build orientation considering support structure volume and build time in laser powder bed fusion," *Materialpruefung/Materials Testing*, vol. 64, pp. 323–338, 3 Mar. 2022, ISSN: 00255300. DOI: 10.1515/mt-2021-2075.

[26] Prusa Research, *Tree supports*, https://help.prusa3d.com/article/tree-supports_1515, 2021. (visited on 10/03/2023).

[27] J. Liu *et al.*, "Current and future trends in topology optimization for additive manufacturing," *Structural and Multidisciplinary Optimization*, vol. 57, pp. 2457–2483, 6 Jun. 2018, ISSN: 1615-147X. DOI: 10.1007/s00158-018-1994-3.

[28] Y. Zhou, Y. Nomura, and K. Saitou, "Anisotropic multicomponent topology optimization for additive manufacturing with build orientation design and stress-constrained interfaces," *Journal of Computing and Information Science in Engineering*, vol. 21, 1 Feb. 2021, ISSN: 15309827. DOI: 10.1115/1.4047487.

[29] M. Langelaar, "Topology optimization for additive manufacturing with controllable support structure costs," in *Proceedings of the 7th European Congress on Computational Methods in Applied Sciences and Engineering*, ECCOMAS Congress, 2016, pp. 3689–3699, ISBN: 978-618-82844-0-1.

[30] M. Langelaar, "Combined optimization of part topology, support structure layout and build orientation for additive manufacturing," *Structural and Multidisciplinary Optimization*, vol. 57, pp. 1985–2004, 5 May 2018, ISSN: 16151488. DOI: 10.1007/s00158-017-1877-z.

[31] K. Hu, S. Jin, and C. Wang, "Support slimming for single material based additive manufacturing," *CAD Computer Aided Design*, vol. 65, pp. 1–10, Aug. 2015, ISSN: 00104485. DOI: 10.1016/j.cad.2015.03.001.

[32] F. Xu, Y. Wong, H. Loh, J. Fuh, and T. Miyazawa, "Optimal orientation with variable slicing in stereolithography," *Rapid Prototyping Journal*, vol. 3, pp. 76–88, 3 Sep. 1997, ISSN: 1355-2546. DOI: 10.1108/13552549710185644.

[33] K. Thrimurthulu, P. Pandey, and N. Reddy, "Optimum part deposition orientation in fused deposition modeling," *International Journal of Machine Tools and Manufacture*, vol. 44, pp. 585–594, 6 May 2004, ISSN: 08906955. DOI: 10.1016/j.ijmachtools.2003.12.004.

[34] S. Singhal, P. Jain, P. Pandey, and A. Nagpal, "Optimum part deposition orientation for multiple objectives in sl and sls prototyping," *International Journal of Production Research*, vol. 47, pp. 6375–6396, 22 Jan. 2009, ISSN: 00207543. DOI: 10.1080/00207540802183661.

[35] P. Bhatt, R. Malhan, P. Rajendran, and S. Gupta, "Building free-form thin shell parts using supportless extrusion-based additive manufacturing," *Additive Manufacturing*, vol. 32, Mar. 2020, ISSN: 22148604. DOI: `10.1016/j.addma.2019.101003`.

[36] H. Byun and K. Lee, "Determination of the optimal build direction for different rapid prototyping processes using multi-criterion decision making," *Robotics and Computer-Integrated Manufacturing*, vol. 22, pp. 69–80, 1 Feb. 2006, ISSN: 07365845. DOI: `10.1016/j.rcim.2005.03.001`.

[37] D. Frank and G. Fadel, "Expert system-based selection of the preferred direction of build for rapid prototyping processes," *Journal of Intelligent Manufacturing*, vol. 6, pp. 339–345, 5 Oct. 1995, ISSN: 0956-5515. DOI: `10.1007/BF00124677`.

[38] L. D. Angelo, P. D. Stefano, and E. Guardiani, "A build time estimator for additive manufacturing," in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT)*, IEEE, Jun. 2019, pp. 344–349, ISBN: 978-1-7281-0429-4. DOI: `10.1109/METROI4.2019.8792907`.

[39] C. Y. Yap *et al.*, "Review of selective laser melting: Materials and applications," *Applied Physics Reviews*, vol. 2, 4 Dec. 2015, ISSN: 1931-9401. DOI: `10.1063/1.4935926`.

[40] N. Padhye and K. Deb, "Multi-objective optimisation and multi-criteria decision making in sls using evolutionary approaches," *Rapid Prototyping Journal*, vol. 17, pp. 458–478, 6 2011, ISSN: 13552546. DOI: `10.1108/13552541111184198`.

[41] W. Wang, H. Shao, X. Liu, and B. Yin, "Printing direction optimization through slice number and support minimization," *IEEE Access*, vol. 8, pp. 75 646–75 655, 2020, ISSN: 21693536. DOI: `10.1109/ACCESS.2020.2980282`.

[42] J. Zhao, "Determination of optimal build orientation based on satisfactory degree theory for rpt," in *Ninth International Conference on Computer Aided Design and Computer Graphics (CAD-CG'05)*, IEEE, 2005, pp. 225–230, ISBN: 0-7695-2473-7. DOI: `10.1109/CAD-CG.2005.32`.

[43] D. Bender and A. Barari, "Using 3d density-gradient vectors in evolutionary topology optimization to find the build direction for additive manufacturing," *Journal of Manufacturing and Materials Processing*, vol. 7, 1 Feb. 2023, ISSN: 25044494. DOI: `10.3390/jmmp7010046`.

[44] Y. Zhang, A. Bernard, R. Gupta, and R. Harik, "Feature based building orientation optimization for additive manufacturing," *Rapid Prototyping Journal*, vol. 22, pp. 358–376, 2 2016, ISSN: 13552546. DOI: `10.1108/RPJ-03-2014-0037`.

[45] Z. Luo, F. Yang, G. Dong, Y. Tang, and Y. Zhao, "Orientation optimization in layer-based additive manufacturing process," in *Proceedings of the ASME Design Engineering Technical Conference*, vol. 1A-2016, American Society of Mechanical Engineers (ASME), 2016, ISBN: 9780791850077. DOI: `10.1115/DETC2016-59969`.

[46] M. Matos, A. Rocha, and L. Costa, "Many-objective optimization of build part orientation in additive manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 112, pp. 747–762, 3-4 Jan. 2021, ISSN: 14333015. DOI: `10.1007/s00170-020-06369-5`.

[47] S. Pereira, A. Vaz, and L. Vicente, "On the optimal object orientation in additive manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 98, pp. 1685–1694, 5-8 Sep. 2018, ISSN: 14333015. DOI: `10.1007/s00170-018-2218-0`.

[48] S. Chowdhury, K. Mhapsekar, and S. Anand, "Part build orientation optimization and neural network-based geometry compensation for additive manufacturing process," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 140, 3 Mar. 2018, ISSN: 15288935. DOI: `10.1115/1.4038293`.

[49] P. Jaiswal, J. Patel, and R. Rai, "Build orientation optimization for additive manufacturing of functionally graded material objects," *International Journal of Advanced Manufacturing Technology*, vol. 96, pp. 223–235, 1-4 Apr. 2018, ISSN: 14333015. DOI: `10.1007/s00170-018-1586-9`.

[50] M. Livesu, S. Ellero, J. Martínez, S. Lefebvre, and M. Attene, "From 3d models to 3d prints: An overview of the processing pipeline," *Computer Graphics Forum*, vol. 36, pp. 537–564, 2 May 2017, ISSN: 14678659. DOI: `10.1111/cgf.13147`.

[51] H. Bikas, P. Stavropoulos, and G. Chryssolouris, "Additive manufacturing methods and modeling approaches: A critical review," *International Journal of Advanced Manufacturing Technology*, vol. 83, pp. 389–405, 1-4 Mar. 2016, ISSN: 14333015. DOI: `10.1007/s00170-015-7576-2`.

[52] X. Wang, M. Jiang, Z. Zhou, J. Gou, and D. Hui, "3d printing of polymer matrix composites: A review and prospective," *Composites Part B: Engineering*, vol. 110, pp. 442–458, Feb. 2017, ISSN: 13598368. DOI: `10.1016/j.compositesb.2016.11.034`.

[53] A. Lalehpour and A. Barari, "Post processing for fused deposition modeling parts with acetone vapour bath," *IFAC-PapersOnLine*, vol. 49, pp. 42–48, 31 2016, ISSN: 24058963. DOI: 10.1016/j.ifacol.2016.12.159.

[54] Y. Qin, Q. Qi, P. Shi, P. Scott, and X. Jiang, "Automatic generation of alternative build orientations for laser powder bed fusion based on facet clustering," *Virtual and Physical Prototyping*, vol. 15, pp. 307–324, 3 Jul. 2020, ISSN: 17452767. DOI: 10.1080/17452759.2020.1756086.

[55] E. Atzeni *et al.*, "Additive manufacturing as a cost-effective way to produce metal parts," in *High Value Manufacturing: Advanced Research in Virtual and Rapid Prototyping - Proceedings of the 6th International Conference on Advanced Research and Rapid Prototyping, VR@P 2013*, 2014, pp. 3–8, ISBN: 9781138001374. DOI: 10.1201/b15961-3.

[56] P. Mercelis and J. Kruth, "Residual stresses in selective laser sintering and selective laser melting," *Rapid Prototyping Journal*, vol. 12, pp. 254–265, 5 2006, ISSN: 13552546. DOI: 10.1108/13552540610707013.

[57] C. Sullivan and A. Kaszynski, "Pyvista: 3d plotting and mesh analysis through a streamlined interface for the visualization toolkit (vtk)," *Journal of Open Source Software*, vol. 4, p. 1450, 37 May 2019. DOI: 10.21105/joss.01450.

[58] J. Jiang, J. Stringer, X. Xu, and R. Zhong, "Investigation of printable threshold overhang angle in extrusion-based additive manufacturing for reducing support waste," *International Journal of Computer Integrated Manufacturing*, vol. 31, pp. 961–969, 10 Oct. 2018, ISSN: 0951-192X. DOI: 10.1080/0951192X.2018.1466398.

[59] E. van de Ven, R. Maas, C. Ayas, M. Langelaar, and F. van Keulen, "Overhang control based on front propagation in 3d topology optimization for additive manufacturing," *Computer Methods in Applied Mechanics and Engineering*, vol. 369, Sep. 2020, ISSN: 00457825. DOI: 10.1016/j.cma.2020.113169.

[60] D. Wilke, S. Kok, and A. Groenwold, "The application of gradient-only optimization methods for problems discretized using non-constant methods," *Structural and Multidisciplinary Optimization*, vol. 40, pp. 433–451, 1-6 Jan. 2010, ISSN: 1615147X. DOI: 10.1007/s00158-009-0389-x.

[61] K. Pfeiffer and Q. Pham, "Time-optimal control via heaviside step-function approximation," in *IEEE International Conference on Intelligent Robots and Systems*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 8223–8230, ISBN: 9781665491907. DOI: 10.1109/IROS55552.2023.10342255.

[62] K. Asadi and M. L. Littman, "An alternative softmax operator for reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning*, D. Precup and Y. W. Teh, Eds., ser. Proceedings of Machine Learning Research, vol. 70, PMLR, Aug. 2017, pp. 243–252.

[63] A. de By and O. Huisman, *Principles of geographic information systems : an introductory textbook*. The International Institute for Geo-Information Science and Earth Observation (ITC), 2009, ISBN: 9789061642695.

[64] C. Wang, C. Yin, H. Chen, and Y. K. Chen, "Arithmetic operations beyond floating point number precision," *International Journal of Computational Science and Engineering*, vol. 6, p. 206, 3 Sep. 2011, ISSN: 1742-7185. DOI: 10.1504/IJCSE.2011.042024.

[65] Z. Hu, Q. Su, X. Yang, and Z. Xiong, "Not guaranteeing convergence of differential evolution on a class of multimodal functions," *Applied Soft Computing Journal*, vol. 41, pp. 479–487, Apr. 2016, ISSN: 15684946. DOI: 10.1016/j.asoc.2016.01.001.