Support optimization in additive manufacturing for geometric and thermo-mechanical constraints

Grégoire Allaire, Martin Bihr, Beniamin Bogosel

Support optimization in additive manufacturing for geometric and thermo-mechanical constraints

Grégoire Allaire[1], Martin Bihr[2], Beniamin Bogosel[1]

February 6, 2020

[1] Centre de Mathématiques Appliquées, École Polytechnique, CNRS, Institut Polytechnique de Paris, 91128 Palaiseau, France.
[2] Safran Tech, Rue des jeunes Bois, 78117 Chateaufort, France.

Abstract

Supports are often required to safely complete the building of complicated structures by additive manufacturing technologies. In particular, supports are used as scaffoldings to reinforce overhanging regions of the structure and/or are necessary to mitigate the thermal deformations and residual stresses created by the intense heat flux produced by the source term (typically a laser beam). However, including supports increase the fabrication cost and their removal is not an easy matter. Therefore, it is crucial to minimize their volume while maintaining their efficiency. Based on earlier works, we propose here some new optimization criteria. First, simple geometric criteria are considered like the projected area and the volume of supports required for overhangs: they are minimized by varying the structure orientation with respect to the baseplate. In addition, an accessibility criterion is suggested for the removal of supports, which can be used to forbid some parts of the structure to be supported. Second, shape and topology optimization of supports for compliance minimization is performed. The novelty comes from the applied surface loads which are coming either from pseudo gravity loads on overhanging parts or from equivalent thermal loads arising from the layer by layer building process. Here, only the supports are optimized, with a given non-optimizable structure, but of course many generalizations are possible, including optimizing both the structure and its supports. Our optimization algorithm relies on the level set method and shape derivatives computed by the Hadamard method. Numerical examples are given in 2-d and 3-d.

1 Introduction

Additive manufacturing (AM) is a collection of processes for building structural parts using a layer by layer deposition system. There is a great deal of excitement around AM because these fabrication processes have the advantage of being able to build complex structures without the usual geometric limitations associated to classical fabrication techniques, like moulding or casting [8], [22]. Here, we focus on metallic AM and more precisely on selective laser melting (SLM) or laser powder bed fusion (LPBF), where successive layers of metallic powder are coated by a roller or a rake, then selectively melted by a laser (or electron) beam.

Although AM is very promising because of the liberty of shapes and topologies of parts that can be built by AM, it still suffers from some limitations, as underlined in many works [3, 12, 13, 19, 21, 25, 26, 27, 28, 29, 30, 35, 36, 40, 41]. Typically, these limitations appear because the final printed design is not fully conformed to the intended design. The reason is that the high and unevenly distributed temperatures, generated by the laser beam, induce thermal residual stresses or thermal dilations of the printed structure. Instances of this phenomenon

can be observed on structures which have large portions of surfaces which are close to being horizontal (assuming that the build direction is vertical, i.e. layers of powder are horizontal, as well as the baseplate). Such horizontal regions are called overhangs. In order to mitigate these deformation effects dut to the building process, so-called support parts can be added to the structure with the goal of improving the construction process, which will be removed after the fabrication is finished. Of course, a minimal amount of supports should be added because more supports increase the build time, the material consumption and their removal can be a tricky post-processing operation. Therefore, it is necessary to optimize these supports to maximize their beneficial effects and to minimize their additional cost. Shape and topology optimization is the right tool for optimizing supports. It is a classical technique to automatically design optimal structures [2], [10] and, more recently, it has been extended to the framework of additive manufacturing (see the already cited papers and references therein).

The first main new contribution of the present paper is the coupling of shape and topology optimization for supports with a pre-processing step of geometric optimization to find the optimal build orientation of a given structure for various criteria. Indeed, this orientation step is quite simple but can drastically change the design of supports, as will be demonstrated on several examples. In particular, one of our new geometric criterion takes into account non-accessible surfaces of the structure, where supports are forbidden to attach (greatly simplifying the post-processing step of support removal). In Section 2 several criteria for optimizing the build direction are proposed. They all share the property of being very simple, purely geometric and therefore computationally cheap. The orientation of the part related to various aspects of the AM process was already considered in several works in the literature. Orientation optimization for minimizing the volume and contact area between the shape and supports was considered by [1], [20], [23]. In [15] orientation is optimized for minimizing the stress in vertical supports under overhanging regions. Note that the effect of orientation on the stress distribution was experimentally investigated in [37]. In [14] a measure of tool accessibility was also considered but not coupled to topology optimization of supports. In [18] a weaker notion of accessibility for support removal was introduced (only the main axes are considered as accessible directions), in addition to minimizing the volume and contact area. The article [30] is the closest to the present one since it optimized supports and build orientation within a SIMP framework in 2-d. We depart from this previous work by considering other orientation criteria in 2-d and 3-d (see the beginning of Section 2.1 for more details about our new contributions.

The second main contribution of the present paper is to extend the analysis of our previous work [3]: two new mechanical models are introduced to assess the performance of the supports during their optimization, and new constraints on the contact zone between the part and its supports are taken into account. Indeed, in [3] we focused on the mitigation of overhanging effects: supports were optimized for minimal compliance in a model where gravity loads were applied to the union of the structure and its supports. This model produced satisfactory supports but had two drawbacks. First, because of a volume constraint, supports were not continuously supporting overhanging surfaces but were evenly spaced, which may be problematic in the context of the building process. Second, overhanging surfaces are undesirable, not only because of gravity loads, but mostly because of thermal deformations, which were not taken into account. In Section 3 two models are proposed to correct those drawbacks. The first model amounts to optimize the compliance of the supports submitted to gravity-type load applied only on the overhanging part of the boundary of the built structure (the structure itself is not taken into account in the mechanical analysis). It has the obvious effect that all overhanging surfaces are supported by the resulting optimized supports. Note that a similar model was used in [33] in the different context of the SIMP method (note that these pseudo gravity loads are called "similar to transmissible loads" in [33]). The second model is based on equivalent thermal loads applied to the union of the structure and its support, the compliance of which is minimized. The main motivation is to take into account the thermo-mechanical deformation

induced by the building process. To avoid time consuming optimization, in a first step we perform a precise layer-by-layer thermo-mechanical analysis of the structure alone (without any supports). From this we deduce so-called equivalent static loads which are used in a second step for a standard compliance minimization for a purely elasticity model. Note that using a two-stage process for optimization is a classical idea, already used in other works, see e.g. [9].

Section 4 gives some 3D numerical examples of the coupled approach of build direction optimization followed by shape and topology optimization of the supports. Some examples also illustrate the dramatic change in the support design when adding further constraints. For example, considering several copies of the same part leads to completely different supports. Similarly, adding the constraint that supports should not touch more than once the structure yields also very different topologies. Our numerical methods are identical to those in our previous work [3]. We rely on the level set method for shape and topology optimization. The optimization algorithm is based on an augmented Lagrangian and the mechanical analysis is performed with the finite element software FreeFEM [24].

In the present work, only the supports are optimized and the structure to be built is always kept unchanged. Let us mention that another possibility, instead of adding (optimized) supports, is to redesign the structure to be built, optimizing to make it self-supported. We shall not discuss this alternative approach here and refer to our previous works [4, 5, 6] and references therein.

Notations: in the whole paper the shape to be printed is denoted by $\omega \subset \mathbb{R}^d$ with $d \in \{2, 3\}$, while the supports are denoted by $S \subset \mathbb{R}^d$. The build direction is $\vec{d}$ and for all figures it will be the upward vertical direction.

2 Orientation optimization

Given a design to be printed, there are some preliminary tasks to be done before sending it to the AM machine. Before the support structures are added, the orientation of the shape needs to be chosen. It is obvious that different orientation may need different support strategies and therefore, there exist orientations which may behave better from this point of view. In this section we present some criteria which may be used in order to evaluate what is the best orientation of the structure to be printed, corresponding to particular applications.

Before presenting the criteria used, we recall some aspects related to the SLM printing process which should be taken into account.

- Overhanging regions usually need to be supported in order to guarantee the quality of the printed design. However, removing the supports and ensuring a good quality of the surface of the printed part, which was in contact with the supports, are time consuming post-processing operations. Therefore, the area of these supported surfaces should be minimized.
- The powder deposition system (roller or rake) can induce significant forces on the printed part. The impact of the roller is more important if the projection of the part on the roller plane is larger. Thus, it is interesting to minimize the projected area on this given roller plane. Computing the orientation which minimizes the projected area can also be useful when considering plateaus containing multiple instances of the same part.
- Various software proposing automatic support management produce vertical supports everywhere under the overhanging regions. Therefore, optimizing the orientation for diminishing the volume of these vertical supports is a meaningful problem.
- Large variations in surface areas of horizontal slices may induce big thermal gradients due to phase change. It is therefore desirable to find the orientation for which these horizontal slices have the smallest variation.
- As discussed earlier, often it is necessary to remove the supports and access the contact areas in order to polish them using some tools. As a consequence, no supports should

touch inaccessible zones of the part surface, and the part should be oriented such that inaccessible regions do not need supports. Thus, an important issue is to find which regions of the part surface are inaccessible.

2.1 Optimizing the orientation of the structure

As discussed earlier, different build orientation give different behaviors with respect to support structures or to the whole printing process. In the following, four criteria are considered when looking for an optimal orientation: the area of supported parts of the shape ($\mathcal{G}_1$), the projection of the shape on a plane ($\mathcal{G}_2$), the volume of supports under surfaces that need to be supported ($\mathcal{G}_3$) and the variation of the areas of parallel slices of the shape ($\mathcal{G}_4$). These criteria can be used in a single constrained optimization process or in a multi-objective optimization problem.

The supported area functional ($\mathcal{G}_1$) is a popular constraint and has already been used, but in this work two more aspects are investigated. The first aspect concerns the computation of an approximate gradient using a regularization approach. This allows to make a local refinement of the orientation, once the values of the functional are computed on a discrete grid. The second aspect is related to accessibility issues. Accessible regions can be detected using various methods as shown in Section 2.2. Once these regions are known, a weight function may be considered on each facet regarding its accessibility. Minimizing the supported surface area with this weight function gives orientations avoiding as much as possible the contact of supports with inaccessible regions.

The same aspects regarding the construction of an approximate gradient apply for the projection functional ($\mathcal{G}_2$). Moreover, this functional may be used to model the contact between the shape and the powder deposition system and can be used in order to perform a secondary optimization of the orientation in the horizontal plane, as explained in Remark 2.3.

The volume functional $\mathcal{G}_3$ is classical and is recalled for the sake of completeness, as it is used in some of the illustrative examples.

The fourth proposed functional is the variation of the areas of consecutive layers of the shape ($\mathcal{G}_4$). This is a relevant quantity, since two successive layers which have a large difference in area may generate important thermal gradients.

2.1.1 Minimize area of supported surfaces.

The contact surface between the structure and the support often needs some post-processing. Therefore, for some applications, it is important to minimize, as much as possible, the area of supported surfaces. Depending on the material and process used, there usually is an angle, denoted β in the following, such that surfaces making an angle less than β with the baseplate need supports. Equivalently, the normal vectors of such surfaces are close to being vertical, i.e. the angle between the normal and the build direction is larger than $\pi - \beta$. Varying the orientation of the part, is equivalent to varying the build direction. In the following this simpler point of view is adopted. The unit vector $\vec{d}$ denotes the build direction. This allows us to characterize regions which need supports by the inequality $\vec{n} \cdot (-\vec{d}) \geq \cos\beta$, where $\vec{n}$ is the outer unit normal to the part ω to be printed. This leads to the following formula for the area of regions needing support:

$$\mathcal{G}_1(\vec{d}) = \int_{\partial\omega} \chi_{\{\vec{n}\cdot(-\vec{d})-\cos\beta\geq 0\}} ds = \int_{\partial\omega} H(\vec{n}\cdot(-\vec{d}) - \cos\beta) ds,$$

where H is the Heaviside function and χ_Z denotes the characteristic function of the set Z, taking the value 1 inside Z and 0 outside. The functional $\mathcal{G}_1$ becomes an integral on a fixed domain of the Heaviside function of a regular expression containing $\vec{d}$. The functional can already be used in its current form to perform a parametric search for the direction $\vec{d}$ which gives the least surface area $\mathcal{S}(\vec{d})$.

It is possible to slightly regularize the above quantity, which allows the use of gradient algorithms for a more fine search for the minimizers. In order to make $\mathcal{G}_1$ differentiate it is possible to use a regularization of the Heaviside function. Given a parameter $\varepsilon > 0$, a C^1 regularization of $H(x)$ is given by $H_\varepsilon(x) = \frac{-0.25}{\varepsilon^3}x^3 + \frac{0.75}{\varepsilon}x + 0.5$ on $[-\varepsilon, \varepsilon]$ and $H_\varepsilon(x) = H(x)$ otherwise. Since the variable x is the cosine of an orientation angle, the regularization parameter ε is non-dimensional. In the sequel, the same notation ε will denote several different non-dimensional regularization or penalization parameters.

With the function H_ε it is possible to define the following regularized functional

$$\mathcal{G}_{1,\varepsilon}(\vec{d}) = \int_{\partial\omega} H_\varepsilon(\vec{n} \cdot (-\vec{d}) - \cos\beta) ds \tag{1}$$

and to compute its gradient using the formula

$$\frac{\partial \mathcal{G}_{1,\varepsilon}}{\partial z}(\vec{d}) = \int_{\partial\omega} H'_\varepsilon(\vec{n} \cdot (-\vec{d}) - \beta)\vec{n} \cdot \frac{\partial(-\vec{d})}{\partial z} ds, \tag{2}$$

where z is any of the parameters defining $\vec{d}$ (see (3)).

Practical implementation. The numerical minimization of (1) is based on the gradient given in (2). The direction $\vec{d}$ is parametrized by two angles $\theta \in [0, 2\pi]$ and $\varphi \in [0, \pi]$ in the following natural way

$$\vec{d}(\theta, \varphi) = (\sin\varphi\cos\theta, \sin\varphi\sin\theta, \cos\varphi). \tag{3}$$

In this way $\mathcal{G}_{1,\varepsilon}$ becomes a function of two variables, which is to be optimized on $[0, 2\pi] \times [0, \pi]$. In order to avoid local minima, a discrete grid of $[0, 2\pi] \times [0, \pi]$ is used in order to compute some initial values of $\mathcal{G}_{1,\varepsilon}$. The point giving the lowest value in the considered grid is identified and a quasi-Newton optimization method is used starting from there to optimize further the value of the functional, using the gradient computed with (2).

Expressions (1) and (2) can be computed efficiently when a triangulation of $\partial\omega$ is available. Geometries are often described in AM using STL files. The STL files contain a triangulation of the shape ω to be fabricated. Once a triangulation $\mathcal{T} = (T_i)_{i=1}^{N_T}$ of $\partial\omega$ is known, the quantities (1) and (2) can be naturally approximated using functions which are piecewise constant on triangles.

In the following the behavior of our optimization algorithm is investigated in the case of a torus. In order to avoid potential local minima, an initial search is performed on a 20×10 grid of $[0, 2\pi] \times [0, \pi]$. In a second stage, a quasi-Newton optimization algorithm is run using the approximate gradient starting from the best point found on the previous grid. The regularization parameter ε is set to 0.05. In Figure 1 the behavior of the algorithm is illustrated for two different initial orientations of the torus. The algorithm manages to capture approximately the same optimal cost: the optimal values obtained for the two chosen starting orientations are 2.3934 and 2.3936, respectively. Notice however that this does not mean that the same orientation is obtained. In fact, once an optimal orientation is given, one can turn the object horizontally, maintaining the value of the objective function constant. Therefore, the optimal orientation is not unique and an optimization algorithm will only capture one of these optimal orientations. For more realistic cases, the multimodal behavior of the functional at hand may require to consider a denser search grid on $[0, 2\pi] \times [0, \pi]$ in order to avoid local minima and to find a proper initialization for the gradient algorithm.

Remark 2.1. *The fact that the value of the functional $\mathcal{G}_1$ does not change when turning the object horizontally amounts to say that $\mathcal{G}_1$ is constant when $\phi = 0$ and $\theta \in [0, 2\pi]$. When $\phi > 0$ this is no longer the case, and this can clearly be seen in the 2D plots in Figure 1. In other words, the functional $\mathcal{G}_1$ is constant and minimal along a curve which is not necessarily a straight line. In particular, when changing the initial orientation, the optimization domain changes even though the minimal values do not.*

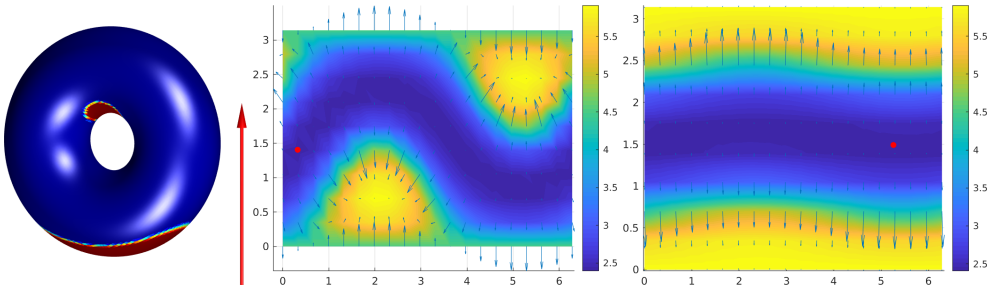

Figure 1: Optimization of the orientation starting from different initial orientations. The 2D plots show the values of the functional $\mathcal{S}_\varepsilon$. The red dots show two found optimal orientations which have the same optimal values.

Regions with unwanted support. It is possible to add information about regions on which no supports are desired. This can be made with the aid of a penalization: instead of minimizing $\mathcal{G}_{1,\varepsilon}(\vec{d})$ we can optimize a functional of the form

$$\mathcal{G}p_{1,\varepsilon}(\vec{d}) = \int_{\partial\omega} \eta(s) H_\varepsilon(\vec{n} \cdot (-\vec{d}) - \cos\beta) ds, \tag{4}$$

where $\eta : \partial\omega \to \mathbb{R}_+$ is a chosen weight function. The weight function is set to 1 on accessible regions, while on inaccessible regions, or, for example, regions which need to have a guaranteed quality, the weight function can be set to a large value $\eta(s) = M >> 1$. If the constant M is large enough, the optimization algorithm will try to minimize the contact of the support with the unwanted region. This idea is particularly useful when combined with aspects shown in in Section 2.2 which deals with the detection of unreachable regions. Indeed, an important issue is to optimize the build orientation such that supports avoid inaccessible regions (or touch them as little as possible).

2.1.2 Minimizing the projected area on a given plane

Another quantity of interest is the projected area of the shape ω on a plane. There are at least two motivations for considering the optimization of such a criterion (corresponding to different choices of the projection direction):

1. When preparing a plateau it may be interesting to maximize the number of structures that can be printed simultaneously. Of course, there are some optimal packing algorithms which can do the job. However, it may be useful to first orientate all the structures so that their projected area on the baseplate is minimal. Choosing the projection direction to be the build direction and minimizing the projected area will produce an educated guess to initialize the packing algorithm.
2. Another application is the indirect minimization of the forces induced by the powder deposition system (roller or rake) on the part to be printed. Taking the projection direction equal to the horizontal normal vector to the roller and minimizing the projected area amount to minimize the surface on which the roller may apply lateral forces. Note in passing that those lateral forces are critical when the high temperature gradients of the building process induce vertical displacements of the structure which can be hit by the roller. This phenomenon is a major concern in industrial applications of additive manufacturing.

The projected area (with repetition) of a shape ω on a plane with normal direction $\vec{r}$ is given by

$$\mathcal{G}_2(\vec{r}) = \int_{\partial\omega} (\vec{n} \cdot \vec{r})_+ ds,$$

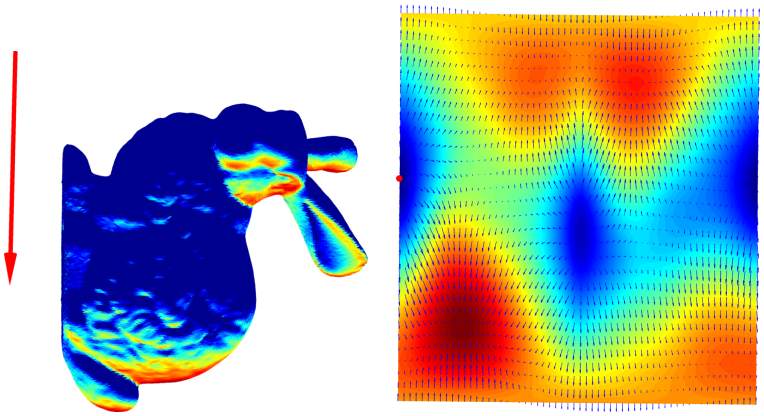

Figure 2: Optimization of orientation in order to minimize the projection (left). Two dimensional view of the angle space (θ, φ) with computed gradients. The red dot is the minimizer.

where $x_+ = \max(0, x)$ is the positive part of x. Notice that when this formula is used for complex (non convex) surfaces, the projection is counted with repetition (overlapping surfaces are counted multiple times). This is perfectly adapted to the second point raised above where the impact of the roller on the part is taken into account. The formula for $\mathcal{G}_2$ is the exact projection area when the shape is convex.

Similarly to the minimization of $\mathcal{G}_1$ defined in Section 2.1.1, it is possible to use this formula directly for doing parametric optimization. The positive part is non-smooth and does not allow the computation of the gradient of $\mathcal{G}_2$. In order to be able to differentiate this quantity a regularization of the positive part can be used. For a given parameter $\varepsilon > 0$, a C^1 regularization of x_+ is given by $f_\varepsilon(x) = \frac{(x+\varepsilon)^2}{4\varepsilon}$ on $[-\varepsilon, \varepsilon]$ and $f_\varepsilon(x) = x_+$ otherwise. The regularized functional becomes

$$\mathcal{G}_{2,\varepsilon}(\vec{r}) = \int_{\partial\omega} f_\varepsilon(\vec{n} \cdot \vec{r}) ds, \tag{5}$$

and its gradient can be computed with a formula similar to (2).

Practical Implementation. The minimization of (5) is performed as in the previous section. The direction $d(\theta, \varphi)$ defined in (3) is considered for $\theta \in [0, 2\pi]$ and $\varphi \in [0, \pi]$ and $\mathcal{G}_{2,\varepsilon}$ becomes a function of these two variables. In order to avoid local minima, the values of $\mathcal{G}_{2,\varepsilon}$ are evaluated on a discrete grid. Once the best point in this grid is identified, a quasi-Newton optimization method is used to optimize further the value of the functional. The functional is discretized in the same way as for the area of supported regions when $\partial\omega$ is a triangulated surface.

An example of computation made with a Matlab implementation can be seen in Figure 2. Together with the optimal orientation for the projection one can see a 2D representation of the space of angles, the colors representing the value of the projection and the arrows are the gradients. This computation is made with a regularization parameter $\varepsilon = 0.1$.

Remark 2.2. Exact projection (without repetition). *If the exact projection needs to be computed (neglecting repeated surfaces) then it is possible to proceed as follows.*

- *Given a direction $\vec{d}$, find all points of ω which are accessible from that direction.*
- *Compute the projected area using the same formula as $\mathcal{G}_2$, but integrating only on the accessible region.*

Remark 2.3. Combining two objective functions. *As seen in Section* 2.1.1 *it is possible to find the orientation giving the minimal area of regions which need to be supported. Once the build direction $\vec{d}$ is fixed, rotating the shape ω in the horizontal plane does not change the supported areas. It is therefore possible to look at a different relevant quantity, namely the projection on the plane of the roller. As remarked previously, the powder deposition system induces non-negligible impact and friction forces on*

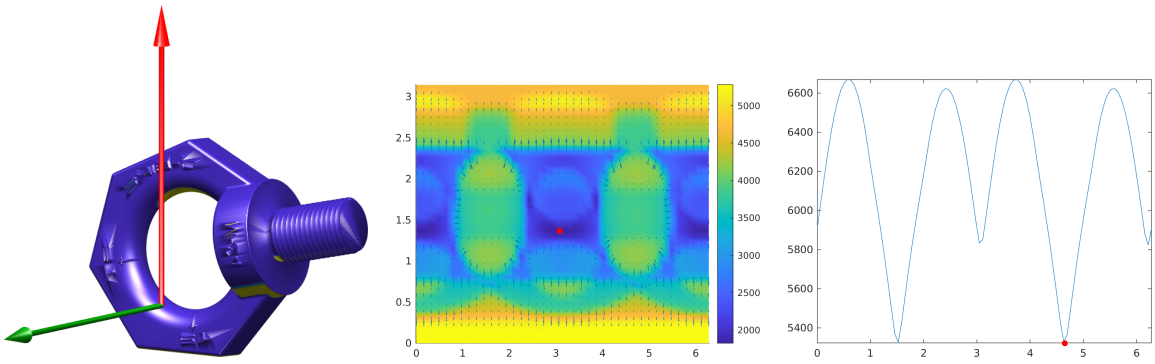

Figure 3: Two objective functions: minimize the area of supported surfaces (red arrow) and the projection on the plane of the roller (green arrow). The 2D plot (middle) shows the supported area $\mathcal{G}_1$ in terms of the two angles (θ, φ). The 1D plot (right) shows the projection on the plane of the roller, depending on a single angle, once the optimized orientation for $\mathcal{G}_1$ is fixed.

the shape to be printed. In order to reduce these forces the projection of the shape on the plane of the roller should be as small as possible. It is therefore possible to use the projection functional described in this section in order to minimize the projected area on this plane and to find the optimal normal direction $\vec{r}$ for the roller. One example of such computation is shown in Figure 3 where the optimized orientation of a bolt is considered. The red arrow shows the build direction $\vec{d}$ obtained when minimizing the area of supported surfaces for $\beta = 45°$. The green arrow shows the optimal orientation $\vec{r}$ which minimizes the projection on the plane of the roller. In the same figure the 2D representations of the area of the supported surfaces with respect to the two rotation angles, as well as the one dimensional graph of the projection on the roller plane with respect to the rotation angle in the horizontal plane are presented.

2.1.3 Minimizing the volume of vertical supports

Some commercial software propose automatic generation of supports based on geometric criteria. The most common criterion is the fact that surfaces making an angle smaller than a threshold β with the horizontal plane need to be supported. As underlined in the previous sections these surfaces can be characterized by the inequality $\vec{n} \cdot (-\vec{d}) \geq \cos\beta$, where $\vec{d}$ is the build direction. In the following, given a surface $\partial\omega$ these surfaces are denoted by $\Gamma_\beta(\partial\omega)$.

The most basic supports are vertical supports under $\Gamma_\beta(\partial\omega)$. Such supports can be built in the following manner. Pick a point $x \in \Gamma_\beta(\partial\omega)$ and construct a segment s_x going in the opposite build direction until it either meets another part of $\partial\omega$ or the baseplate denoted by Γ_D. Formally, one may define the vertical supports as $V_\beta(\partial\omega) = \cup_{x\in\Gamma_\beta(\partial\omega)} s_x$. Then define the volume of these supports as

$$\mathcal{G}_3(\vec{d}) = |V_\beta(\partial\omega)|,$$

where, as usual, $|V|$ denotes the volume of the set V. These kind of supports may be constructed in a straightforward manner using *ray-casting* algorithms. A ray casting algorithm takes as inputs a triangulated surface $\mathcal{T}$, a family of source points (x_i) and a direction $(\vec{d_i})$ for each source points. Then it returns for each source point x_i and direction $\vec{d_i}$ wether or not the ray starting from x_i in direction d_i intersect the triangulation $\mathcal{T}$ and, if yes, it gives the first intersection point. In the particular case of supports, the triangulation $\mathcal{T}$ is the union between the triangulation of $\partial\omega$ (obtained, for example from a STL file) and its projection on the baseplate Γ_D. The directions $\vec{d_i}$ are all equal to $-\vec{d}$ and the source points x_i are all vertices of triangles in $\Gamma_\beta(\partial\omega)$.

Examples of computation of such supports are shown in Figure 4. It is possible to construct the supports for a finite family of orientations and choose the orientation giving the

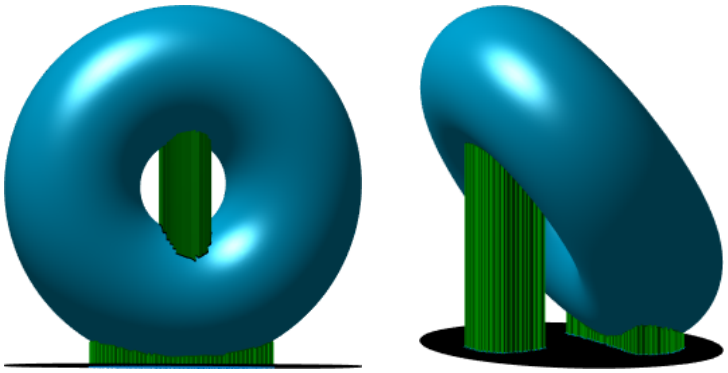

Figure 4: Example of vertical supports found using ray casting algorithms

least volume of vertical supports. Our computations are done in Matlab using the toolbox *Ray casting for deformable triangular 3D meshes* [42]. However, these techniques are well known in Computer Graphics and the computation speed can be greatly improved using more advanced techniques involving Graphical Processing Units (GPU). For the optimization of the volume of vertical supports we do not use any gradient information (which we did not study) and rather we simply consider a discrete grid in the angle space, compute the functional $\mathcal{G}_3(\vec{d}(\theta,\varphi))$ for each point of the grid and then pick the orientation with the lowest value.

2.1.4 Minimizing the variation of section areas

When dealing with metal-based AM technology high temperatures are produced by the energy source, used to melt the metal powder. An indirect way to reduce the residual stresses present in the printed object at the end of the fabrication process is to have a low variation of the areas of the cross sections of the structure, parallel to the baseplate. This criterion obviously depends on the build orientation. The computation of section areas of a triangulated surface is a subject of interest in the CAD literature (see for example [34]).

Suppose the orientation of the structure ω is given, i.e. the build direction $\vec{d}$ is chosen. Consider the height function $h : \partial\omega \to \mathbb{R}$, $h(x,y,z) = z$. For the chosen orientation suppose that the image of h is the interval $[z_{\min}, z_{\max}]$. For every $h_z \in (z_{\min}, z_{\max})$ it is possible to compute the one dimensional contour which is the intersection of the plane $z = h_z$ with $\partial\omega$. When $\partial\omega$ is a triangulated surface the contour can be easily computed by detecting which triangles are cut by this contour, i.e. triangles for which the height function h takes values which are above and below h_z. In a second step the contours need to be oriented in the counter-clockwise direction so that the area of a polygon defined by the vertices $(x_1,y_1),...,(x_N,y_N)$ can be computed with the formula

$$\mathcal{A}(h_z) = \frac{1}{2}\left(\sum_{i=1}^{N}(x_{i+1}y_i - x_i y_{i+1})\right).$$

Note that this formula is valid even for a union of polygons with the proper anti-clockwise orientation. In practice, for a given build direction $\vec{d}$ and a family of heights $h_1,...,h_N \in [z_{\min}, z_{\max}]$,, all the contours are computed and are oriented counter clockwise. Then the previous formula is used to compute the areas $\mathcal{A}(h_1),...,\mathcal{A}(h_N)$ encompassed by each of these contours. Once the areas of the slices are known, there are various way to evaluate the variation of these quantities. Since the goal is to capture local variations, the following formula is used:

$$\mathcal{G}_4(\vec{d}) = \max_{1\le i\le N-1} |\mathcal{A}(h_i) - \mathcal{A}(h_{i+1})|,$$

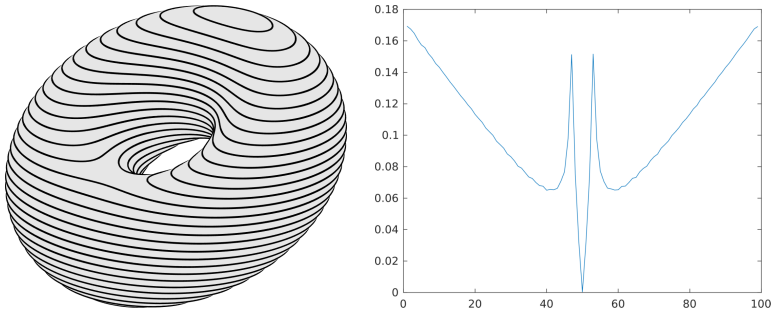

Figure 5: Contours of parallel slices of a torus minimizing the maximal difference between areas of consecutive layers (left). This difference of consecutive layers areas is plotted as a function of the layer index (right).

The process is repeated for each orientation which is of interest. The gradient of $\mathcal{G}_4(\vec{d})$ is not computed and its optimization is thus a simple exhaustive evaluation of its values on a discrete grid.

Figure 5 shows the optimal orientation found for a torus using 100 parallel slices, for $\vec{d}(\theta, \varphi)$ given by (3) computed on a grid of size 30×15 of $[0, 2\pi] \times [0, \pi]$ (for readability only 30 slices have been plotted). In order to illustrate the variation of the areas of the sections, a graph of the difference of consecutive areas is shown in Figure 5.

Considering other quantities related to the area of slices is possible and they may give different optimized orientations. Choosing the variance is not realistic, since the quantity does not depend on the order of the slices. Considering the total variation of areas (the sum of differences of areas of consecutive slices) is also an option, but it does not penalize maximal difference of consecutive areas as strongly as the functional proposed above.

Remark 2.4. Minimizing the height. *A trivial, but nonetheless important aspect is being able to diminish, as much as possible, the height of the shape ω to be printed. Modifying the orientation may greatly modify the height of the final design and the printing time is increased when using a bigger height, which necessitates more printing layers. The height is easily computable by looking at the extremal values of the z coordinate for all the points in the triangulation. In order to find the orientation which is of minimal height, it is enough to rotate the points using a rotation matrix and evaluate the width of the rotated shape in the z-direction.*

2.2 Detecting inaccessible regions

The support structures must be removed at the end of the fabrication process. Contact surfaces between the support and the structure ω often need to be accessed from the exterior with some tool for post-processing purposes. Therefore, all such contact surfaces need to be accessible from the exterior or from a given region Σ. We choose the following simple and reasonable criterion of accessibility: a point $x \in \partial\omega$ is accessible from Σ if there exists a point $y \in \Sigma$ such that the segment $[xy]$ does not intersect ω. The choice of such an accessibility criterion is dictated by the ease of devising an algorithm to detect all accessible points, given the shape ω and the region Σ.

A first approach is to use again ray-casting algorithms. Indeed, it is possible to choose a family of source points (y_i) (a discretization of Σ) and to send rays from each of these points towards all points (x_j) from the discretization of $\partial\omega$. If the first intersection point of the ray $[y_i x_j)$ with $\partial\omega$ is x_j then x_j is accessible. Our Matlab implementation is based on the toolbox [42]. It takes as an input the triangulation from an STL file and a family of source points (chosen arbitrarily for the moment) and it performs the algorithm described above. Two results for

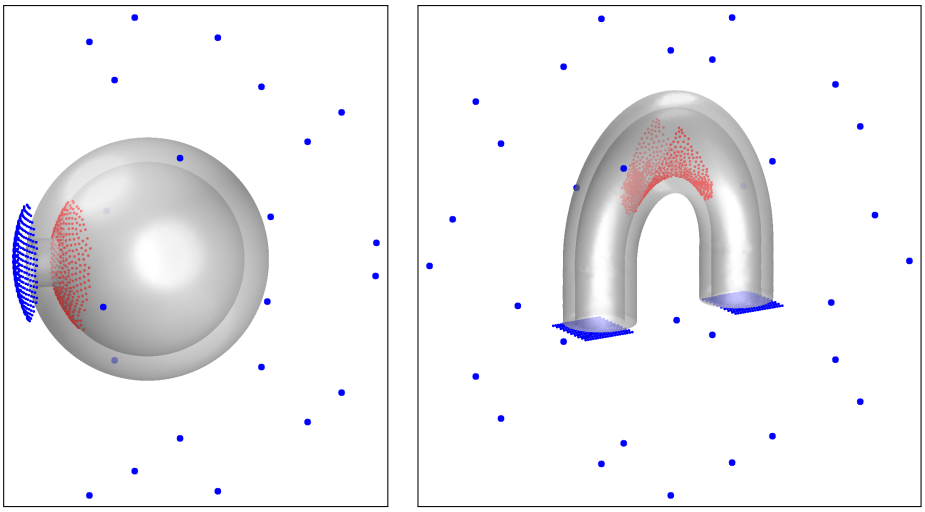

Figure 6: Detecting accessible regions using ray casting. The source points are represented with blue and the inaccessible regions with red.

two different geometries are given in Figure 6. The first geometry is a hollow sphere with an inaccessible region near the hole and the second one is a U-shaped tube. The results show that inaccessible results are successfully identified. This approach is based on Computer Science techniques and it can be further optimized using more advanced techniques using Graphical Processing Units (GPU).

A more mathematical approach is now described, which is a first step towards having a criterion that can not only detect inaccessible regions, but can also enforce an "accessibility constraint" when optimizing the shape ω itself. Let Σ be a surface or a family of points. The objective is to identify all points on $\partial\omega$ which are accessible from Σ. Given $y \in \Sigma$ and $x \in \partial\omega$ the following distance can be defined:

$$L_\omega(x,y) = \inf\{\ell(\gamma) \text{ with } \gamma(t) : [0,1] \to \mathbb{R}^d \setminus \omega, \gamma(0) = x,\ \gamma(1) = y\},$$

where $\ell(\gamma)$ is the length of the curve γ. This distance can be interpreted as a geodesic distance in $\mathbb{R}^d \setminus \omega$. This distance allows us to introduce a new definition of accessibility (which is actually equivalent to the previous one).

Definition 2.5. *A point $x \in \partial\omega$ is accessible from $y \in \Sigma$ if $L_\omega(x,y) = |x-y|$, i.e. the distance "avoiding" ω is equal to the Euclidean distance.*

A point $x \in \partial\omega$ is accessible from Σ if there exists $y \in \Sigma$ such that x is accessible from y.

Given x in $\partial\omega$, the accessibility from Σ may be evaluated using the following function:

$$f(x) = \min_{y\in\Sigma} \left(L_\omega(x,y) - |x-y|\right). \tag{6}$$

Obviously, $f(x) \geq 0$, and $f(x) = 0$ implies the existence of a point $y \in \Sigma$ from which x is accessible (at least if Σ is compact). This shows that knowing the function f defined in (6) is enough in order to determine accessible regions. It turns out to be easy to compute by using techniques already available when dealing with signed distance functions. Indeed, when $y \in \Sigma$ does not touch the structure ω in the sense that there exists $r > 0$ such that the ball $B_r = B(y,r)$ satisfies $B_r \cap \omega = \emptyset$, then $L_\omega(x,y)$ may be computed starting from the signed distance function to the sphere ∂B_r in $\mathbb{R}^d \setminus \omega$. Signed distance functions in $\mathbb{R}^d \setminus \omega$ can be computed efficiently on triangular meshes using Fast Marching Algorithms. The implementation used in our work is MshDist and is presented in [17]. It is freely available from the Github repository of the ISCD Toolbox and it can be interfaced with FreeFEM [24].

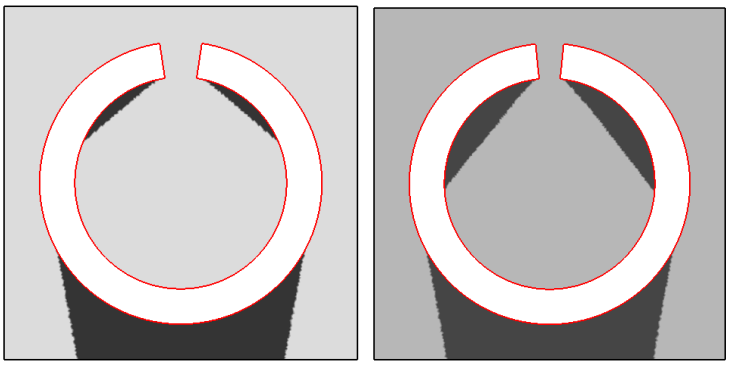

Figure 7: Detection of regions which are accessible/inaccessible from the top boundary of the square. The dark regions are those for which the accessibility function $f(x)$ is strictly positive.

When Σ is discrete the function f can readily be computed, yielding thus all information regarding the accessibility of points on $\partial\omega$. If Σ is a general surface, the function f can be approximated by computing the minimum on a fine enough discretization of Σ. A possible drawback of this method is the fact that a mesh of the exterior of ω (in a bounded box) is needed. In our case this is also a requirement for topology optimization algorithms and there are tools for building such meshes starting from STL files, like shown in the next section.

Two examples illustrate the computation of the accessibility function f given in (6). In Figure 7 a 2-d example is shown where the shape ω is a cut annulus and Σ is the top boundary of the square. The accessibility function (6) is computed and the region where this function is strictly positive are represented in black in the figures. A three dimensional example is shown in Figure 19 for a U-shaped tube geometry.

3 Shape and topology optimization of supports

In the previous section multiple criteria were proposed in order to find the optimal orientation of the shape to be manufactured. In this section the shape and the orientation of ω are fixed and the objective is the optimization of the shape and topology of the support structure. The support structures, denoted by S, lie in a design domain D containing ω, usually a rectangular box. Various works in the literature deal with the optimization of support structures. Among these we mention [13, 19, 21, 30], etc. The previous work [3] presented a framework for the optimization of the support structures using the level set method. In particular, to mitigate overhanging effects, supports were optimized for minimal compliance in a model where gravity loads were applied to the union of the structure and its supports. This model produces satisfactory supports but has two main drawbacks. First, because of a volume constraint, supports were not continuously supporting overhanging surfaces but were evenly spaced, which is meaningful from a mechanical point of view but may be problematic in the context of the building process. Second, overhanging surfaces are undesirable, not only because of gravity loads, but mostly because of thermal deformations, which were not taken into account in our previous work [3]. Therefore, the goal of the present section is to introduce two new models for support optimization, which further extend our framework in [3] with the goal of being more adequate for industrial requirements. The guiding ideas used in deriving these models are summarized below (more details are given in the beginning of Section 2).

- the overhanging surfaces need to be supported
- prevent deformations induced by thermal stresses
- avoid un-necessary contact between the part and the support
- produce supports that are resistant to horizontal loadings induced by the powder depo-

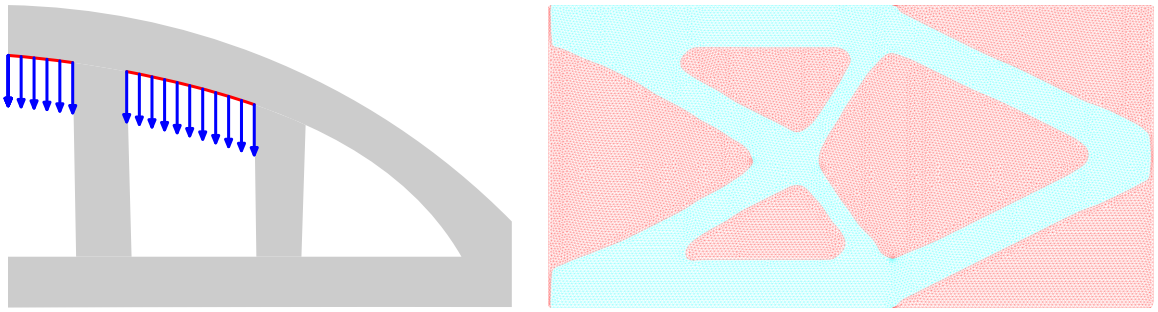

Figure 8: Vertical forces on the region Γ_β for $\beta = 45°$ (left). Exact meshing of the interior and exterior of ω using MMG [16] (right).

sition system.

For the new models, a common pre-processing step is to detect the overhanging surfaces in the structure ω to be built. Given a limit angle β with the horizontal plane (the build direction is vertical), following the notations of Subsection 2.1.3, the overhanging surfaces are denoted by $\Gamma_\beta(\partial\omega)$.

3.1 Pseudo gravity loads on overhanging surfaces

An obvious way of forcing the supports to touch the surface $\Gamma_\beta(\partial\omega)$ is to consider the compliance minimization for the elasticity problem with surface loadings on $\Gamma_\beta(\partial\omega)$, while modeling void, as usual, by an ersatz material. Indeed, if the support does not touch $\Gamma_\beta(\partial\omega)$ where the loadings are applied, the compliance will be very large, and therefore will not be minimized. This type of models was already used in [33] using the SIMP method.

In this subsection, constant vertical forces applied on $\Gamma_\beta(\partial\omega)$ are considered, modeling the effects of gravity. In truth, when dealing with metal powder SLM technologies, it is not gravity that makes surfaces in $\Gamma_\beta(\partial\omega)$ difficult to manufacture without supports. The laser melts the powder and the liquefied metal, being unsupported, bends downward through the powder below creating irregular surfaces. The model of vertical forces is, however, well adapted from another point of view. As a consequence of thermal stresses in the overhanging regions, they tend to lift upwards, possibly causing collision with the roller and failure of the printing process. The supports should, therefore keep these regions at their desired height and they need to withstand these vertical loadings in the build direction. From a mathematical point of view, when considering the compliance for the elasticity system (7), changing the orientation of the forces simply changes the sign of the solution and the compliance stays the same.

Figure 8 displays how gravity (vertical) forces are applied on Γ_β and how the design domain for the support S is meshed in red (the structure ω is green). In order to have the desired accuracy the exterior of the shape ω, which is the design domain for S, is meshed exactly (here, by using the re-meshing library MMG [16]). Linearized elasticity equations are considered in the support S only

$$\begin{cases} -\operatorname{div} A_S e(u) &= 0 \quad \text{in } S \\ A_S e(u)n &= g \quad \text{on } \Gamma_\beta \\ A_S e(u)n &= 0 \quad \text{on } \partial\omega \setminus \Gamma_\beta \\ u &= 0 \quad \text{on } \Gamma_D \end{cases} \tag{7}$$

where, as usual, $e(u) = \frac{1}{2}(\nabla u + \nabla^T u)$ is the symetrized gradient and A_S is the Hooke's tensor (with Young modulus $E = 1$ and Poisson's ratio $\nu = 0.3$ for our numerical computations). The surface Γ_D, where Dirichlet boundary conditions are applied, may consist of different zones: the baseplate or some regions of the part boundary $\partial\omega$ where the support may fix. From an industrial point of view, in some cases it may be easier to construct a vertical wall on which the

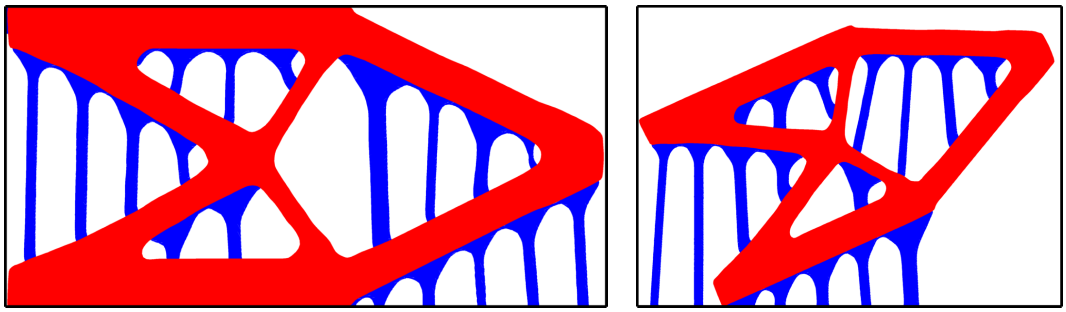

Figure 9: Optimized supports obtained by minimizing the compliance for vertical loadings, limit angle $\beta = 45°$ and two different build orientations.

supports may also fix: in our model this would simply add a vertical part to the boundary Γ_D. The goal is to minimize the compliance given by

$$\mathcal{C}(S) = \int_{\Gamma_\beta} g \cdot u \,, \tag{8}$$

with a volume constraint on S. In engineering practice, one would rather minimize the volume of supports with a constraint on the compliance. Here we choose the other way around because it is much simpler to fix a precise value for the volume constraint (note that there is no conceptual or numerical difficulty in the engineering practice but that it is only the choice of the constraint value which has to adequately chosen). In the sequel our choice of volume constraint is arbitrary and only motivated by our desire to find optimized supports which are neither too fine nor too big.

The shape and topology optimization procedure is based on the level set method described in [7]. The aforementioned reference also contains details about the shape derivative of the compliance functional used in the examples shown in this paper. The support S is parametrized using a function $\varphi_S : D \setminus \omega \to \mathbb{R}$ with the convention that $S = \{x \in D \setminus \Omega : \varphi_S(x) < 0\}$. The level set methodology is classical and it was also recalled in detail in [3]. The resolution of the partial differential equations (7) is done in FreeFEM [24], the reconstruction of a signed distance function uses the toolbox MshDist [17] and the advection of the level set in the direction which decreases the objective function is done with the toolbox Advect [11].

Figure 9 displays two results for two different build orientations of the same given cantilever structure ω (the size of the rectangular working domain D is different). The minimization of (8) is performed under a volume constraint, chosen as a fraction of the working domain. For the first computation (left figure) the volume constraint is $|S| = 0.22|D| = 0.25$ and the optimized compliance is equal to 0.24. For the second orientation (right figure) the volume constraint is $|S| = 0.2|D| = 0.38$ and the optimized compliance is equal to 1. As expected, the supports touch the whole region Γ_β and their optimization yields the expected vertical bars configuration. More complex three dimensional results are shown in Section 4.

3.2 Equivalent thermoelastic loads

It turns out that the typical forces applied to the part during its building process are often thermal loads rather than gravity loads. On the other hand, the deformations induced by the building process are not necessarily located on overhanging surfaces, especially for massive parts. Therefore, in this subsection a new model taking into account thermal loads is proposed. Simulating the building process requires to use a time-dependent thermo-mechanical model, taking into account the layer-by-layer process. Such an approach has been followed in [6] (for optimizing self-supported structures) but is has the inconvenient of being computationally very expensive. Therefore, we suggest another simpler two-stage approach. In a first stage, a

detailed layer-by-layer simulation of the building process is performed for the structure alone. From this computation, equivalent static loads are deduced. In a second stage, this equivalent static load is used in a "standard" compliance minimization problem for a simple linearized elasticity system. A similar two-stage process is used in other works, see e.g. [9].

3.2.1 Process simulation

A layer by layer process simulation has been developed in [6], in order to capture the displacements of the part during the fabrication. In this model the domain D, typically the build chamber, is composed of two subdomains, Ω_1 and Ω_2, separated by the interface $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$. These two subdomains are filled with two different thermoelastic materials, full metal and metallic powder in our case. Material parameters ξ are constant inside each subdomain, and are denoted as follows :

$$\xi(x) = \xi_1 \chi_{\Omega_1}(x) + \xi_2(1 - \chi_{\Omega_1}(x)),$$

where ξ_i are the material properties defined in the subdomain Ω_i and χ_{Ω_i} the characteristic function of Ω_i (equal to 1 if $x \in \Omega_1$ and 0 elsewhere).

The build chamber D is divided into M layers and $(t_i)_{0 \leq i \leq M}$ denotes a sequence of time steps with $t_0 = 0$ and $t_M = t_f$, the final time of the manufacturing process. Every t_i is associated to the building of the i^{th} layer, and the domain $D(t_i)$ corresponding to the first i built layers so that

$$D(t_1) \subset ... \subset D(t_i) \subset ... \subset D(t_f) \equiv D.$$

The state equations used to simulate the process are the heat equation and the linearized steady-state thermo-elasticity system. The heat equation reads

$$\begin{cases} \rho \frac{\partial T_i}{\partial t} - \operatorname{div}(\lambda \nabla T_i) = Q & \text{in } (t_{i-1}, t_i) \times D(t_i), \\ (\lambda \nabla T_i).n = -\beta(T_i - T_{ref}) & \text{on } (t_{i-1}, t_i) \times \partial D(t_i)_N, \\ T_i = T_{ref} & \text{on } (t_{i-1}, t_i) \times \partial D(t_i)_D, \\ T_i(t_{i-1}) = \tilde{T}_{i-1} & \text{in } D(t_i). \end{cases} \tag{9}$$

where T_i is the temperature field, $\rho > 0$ is the product of the material density by the specific heat, $\lambda > 0$ is the thermal conductivity coefficient and $\beta > 0$ the heat transfer coefficient. The thermal body source, $Q = Q_1 \chi_{\Omega_1} + Q_2(1 - \chi_{\Omega_1})$, is supported in the last layer and follows the laser path. Here, we slightly depart from [6] by taking a reference temperature, $T_{ref} = T_{ref1}\chi_{\Omega_1} + T_{ref2}(1 - \chi_{\Omega_1})$, which is not the same in the metal and in the powder. In the metal, the reference temperature is the fusion (or melting) temperature, while in the powder it is the ambient temperature. Finally, $\tilde{T}_{i-1}$ is the extension of the previous temperature $T_{i-1}(t_{i-1})$, such as

$$\tilde{T}_{i-1} = \begin{cases} T_{ref} & \text{in } D(t_i) \backslash D(t_{i-1}), \\ T_{i-1}(t_{i-1}) & \text{in } D(t_{i-1}). \end{cases}$$

The thermoelastic equilibrium system reads

$$\begin{cases} -\operatorname{div}(\sigma_i) = f_i & \text{in } (t_{i-1}, t_i) \times D(t_i), \\ \sigma_i = \sigma_i^{el} + \sigma_i^{th} & \text{in } (t_{i-1}, t_i) \times D(t_i), \\ \sigma_i^{el} = Ae(u_i) & \sigma_i^{th} = K(\min(T_i, T_{ref}) - T_{ref})\mathbb{I}_d, \\ \sigma_i.n = 0 & \text{on } (t_{i-1}, t_i) \times \partial D(t_i)_N, \\ u_i = 0 & \text{on } (t_{i-1}, t_i) \times \partial D(t_i)_D. \end{cases} \tag{10}$$

where σ is the Cauchy stress tensor, f denotes the body forces, σ_{el} is the elastic stress, σ_{th} the thermal stress, A the fourth-order elasticity tensor of an isotropic material with Young modulus $E > 0$, Poisson coefficient $-1 < \nu < 1/2$, $\mathbb{I}_d$ the identity matrix, α the thermal expansion coefficient and $K = -E\alpha/(1 - 2\nu)$. In (10) the formula for the thermal stress σ_{th} is slightly different from that in [6] since an upper bound is applied to T_i. Indeed, the temperature cannot

Parameter	Solid	Powder
Young modulus E (GPa)	110	1.6
Poisson coefficient ν	0.25	0.25
Specific heat C $(J.kg^{-1}.K^{-1})$	610	700
Heat transfer coefficient β $(W.m^{-2}.K^{-1})$	50	50
Thermal expansion coefficient α (K^{-1})	0.000009	0.0000001
Thermal conductivity coefficient λ $(W.m^{-1}.^{\circ}K^{-1})$	15	0.25
Density ρ $(kg.m^{-3})$	4300	2100
Reference temperature T_{ref} (K)	1923	290

Table 1: Material parameters for the thermo-elasticity model.

be higher than the fusion temperature T_{ref}. Since the heat equation does not take into account phase change, this upper bound has to be applied explicitly.

A cooling step has been added just after the heating phase (corresponding to the time required for coating a new layer of powder), allowing the fused metal to solidify and then shrink. During the cooling phase, the thermal body source Q is set to 0 in the heat equation.

For the numerical calculations, mechanical and thermal parameters have been taken from [31]. The titanium alloy TA6V, commonly used in additive manufacturing, is considered and its properties are set in Table 3.2.1.

The other numerical parameters are: $Q_1 = 8000\ W.cm^{-2}$, $Q_2 = 10^{-3}Q_1$. No elastic body forces are taken into account, i.e. $f_i = 0$ in (10). There are $M = 20$ layers, the final time is $t_f = 102\ s$, each layer is built in one single time step of $\Delta t_{heat} = 0.1\ s$ (heating phase), followed by another single time step of $\Delta t_{cool} = 5\ s$ for the cooling phase (with $Q = 0$).

3.2.2 Model validation

The results obtained with this process simulation have been compared with a commercial process simulation software *Simufact Additive* ® [39] for a simple 2-d test case, namely a plate of size 3×1. For each layer i, which is finished at time t_i, the displacements of all previous layers (including the i^{th} one) at time t_i are summed. Then, this "lumped" displacement is plotted on Figure 10 and 11: the results of both models show a qualitative good agreement. However, one can notice a slight difference at the bottom of the vertical wall in Figure 10, with no displacement in our simulation. This is due to the Dirichlet condition imposed on Γ_D, while *Simufact Additive* ® simulate the build platform behavior, also subjected to displacements. Our process simulation model could be improved by including the base plate in the computational domain and taking into account temperature-dependent material properties in order to get closer to the results of *Simufact Additive* ®.

Simufact Additive ® does not provide many details about its simulation model, however it uses non-linear finite element solver, anticipates plastic behaviors of the material, so it is surely more complex than the above simple model. Moreover, results obtained by this software have been experimentally approved by expert companies in additive manufacturing, so we can safely build on it to confirm our results.

The comparison between the model of Subsection 3.2.1 and *Simufact Additive* ® is made with the oversimplified vertical wall geometry (which nevertheless is a common example for AM-simulation) and not with a more realistic geometry (like the T-shape of Figure 12) because, in the absence of supports, the computation of *Simufact Additive* ® breaks down, due to too large mesh deformations. In particular, it is one of the reasons why equivalent loads are deduced from our model which never stops since it relies on linearized elasticity. Another reason is that, *Simufact Additive* ® having its own mesh generator, it is not so simple to export its simulation results in our optimization algorithm.

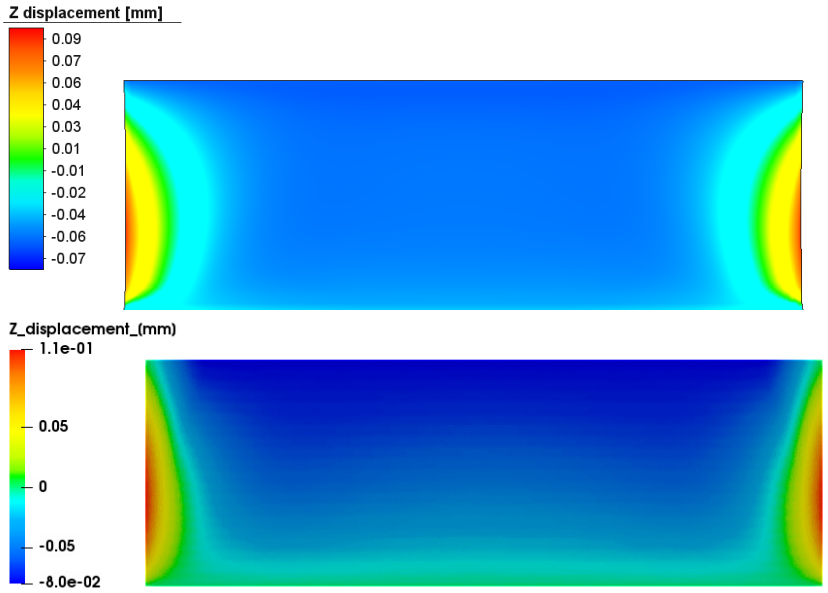

Figure 10: Lumped vertical displacements (along the building direction) of a plate, obtained with *Simufact.Additive* ® (top) and with our model, based on [6] (bottom)

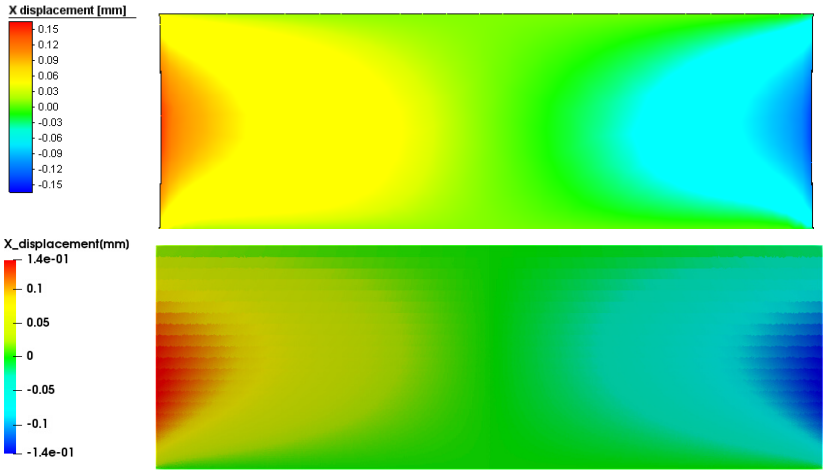

Figure 11: Lumped horizontal displacements of a plate, obtained with *Simufact.Additive* ® (top) and with our model, based on [6] (bottom)

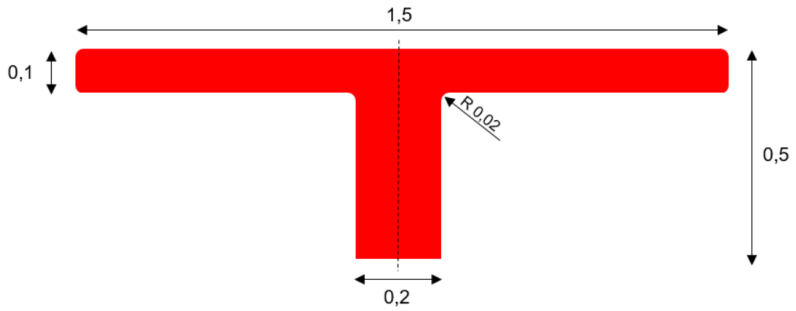

Figure 12: Geometry of T-shape (all corners, except the two bottom ones, are rounded with the same curvature radius).

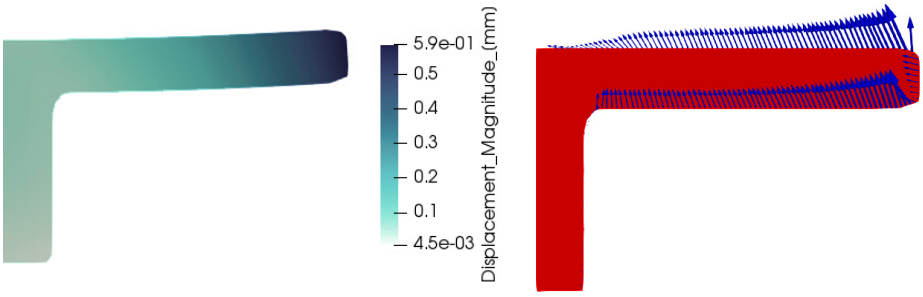

Figure 13: Maximum (in time) displacements of the half T-shape, simulated with our model based on [6] (left), and visualization of the equivalent forces obtained from (11) (right).

3.2.3 Support optimization

As a 2-d test case we consider a T-shape structure with dimensions given in Figure 12. By symmetry, only one half of the T-shape is displayed and used in the numerical simulations (the symmetry condition on the vertical axis is imposed by imposing a zero horizontal displacement). After solving (9) and (10), at each point x of the domain we compute the maximum displacement (in norm) over all times in $(0, t_f)$. This maximum displacement field u_{th} now depends only on the space variable x. For the half T-shape it is displayed on Figure 13. From this maximum displacement field, we now extract equivalent static loads as follows. The displacement field u_{th} is imposed as Dirichlet conditions on $\partial\omega$ and one solves the following linearized elasticity system

$$\begin{cases} -\operatorname{div} Ae(u_{reac}) = 0 & \text{in } \omega \\ u_{reac} = u_{th} & \text{on } \partial\omega \end{cases} \tag{11}$$

Notice that $u_{th} = 0$ on Γ_D in (10), so u_{reac} keeps this condition. From this, computing the normal stress, we deduce a definition of equivalent forces f_{th} on the boundary. More precisely, the equivalent forces f_{th} on $\partial\omega$ are defined by

$$f_{th} = Ae(u_{reac}) \cdot n \tag{12}$$

where n is the exterior unit normal vector. A visualization of these equivalent forces f_{th} for the half T-shape is shown on Figure 13. Definition (12) is somehow arbitrary and other recipes could be devised. For example, some steady-state stress field in ω could be deduced from the solution of (10) and imposed as a pre-stress in the following second stage, instead of the equivalent forces f_{th}.

In a second stage, the supports S are optimized. The mechanical performances of the supports are evaluated by solving a linearized elasticity system in the complete supported structure, denoted by Ω, which is defined by $\overline{\Omega} = \overline{S} \cup \overline{\omega}$, and is contained in the working domain D. The equivalent thermal loads are applied at the boundary of the structure ω:

$$\begin{cases} -\operatorname{div} A_S e(u_{spt}) = f_{th}\delta_{\partial\omega} & \text{in } \Omega \\ A_S e(u_{spt}).n = 0 & \text{on } \partial\Omega \backslash \partial\omega \\ u_{spt} = 0 & \text{on } \Gamma_D \cap \overline{\Omega} \end{cases} \tag{13}$$

where $\delta_{\partial\omega}$ is the Dirac measure for the boundary $\partial\omega$. As usual, the compliance is given by

$$J_1(S) = \int_{\partial\omega} f_{th} \cdot u_{spt}\, ds. \tag{14}$$

Figure 14 displays the supports obtained on the working domain D of size 1×0.5 by minimizing the compliance with a volume constraint on the supports of $|S| = 0.3|D| = 0.15$. The computational domain D is discretized using a 140×70 grid with 9800 nodes and P1 finite

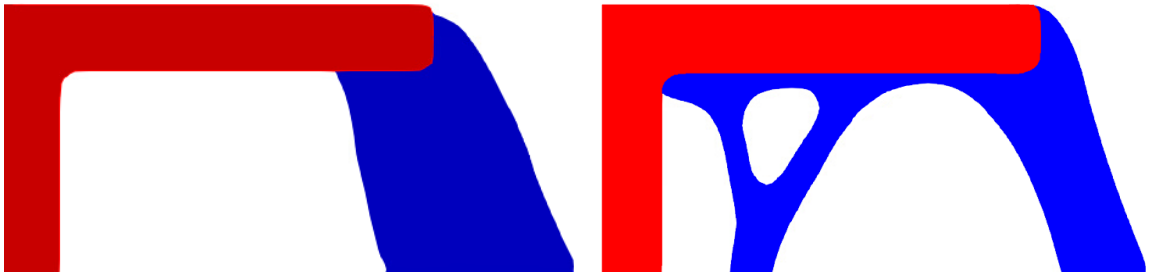

Figure 14: Optimal supports (in blue) obtained by compliance minimization (14) for the equivalent forces issued from the process simulation (left), and with the multiple load optimization (15) (right).

elements are used. Also in our case, the support and the fixed shape have the same mechanical parameters.

This result makes sense but is not satisfactory because it does not guarantee manufacturability since there remains some non-supported overhanging surfaces. To circumvent this drawback this model is combined with the previous one of Subsection 3.1 and a multiple load optimization is performed. More precisely, a new objective function is considered as a sum of two compliances

$$J_2(S) = \int_{\partial\omega} f_{th} \cdot u_{spt} ds + \int_{\Gamma_\beta} g \cdot u ds, \tag{15}$$

where u is the solution of (7), while u_{spt} is the solution of (13). As usual, the functional (15) is minimized with a volume constraint on S, giving the supports displayed in Figure 14 which are much more suitable for additive manufacturing. Remark that, taking u_{th} as the maximum displacement at every point during the fabrication preserves the history of the layer-by-layer process and highlights large displacements that could be hidden if only the displacements at the end of the fabrication were taken into account.

For the results of Figure 14 the compliance caused by the thermal loads is twice larger than the compliance caused by pseudo gravity loads (or transmissible loads in the vocabulary of [33]). We recall that these pseudo gravity loads are applied on overhang surfaces to model the fact that they cannot be manufactured properly, and thus the proportion between gravity loads and thermal loads in our model does not reflect reality. As the magnitude of the gravity loads are chosen in a way to assure the manufacturability, that could depends on the geometry of the part, we simply add the related compliance to the one caused by thermal loads, without any weight between them.

In a future work, the process simulation could be performed during the second stage of optimization, in order to get the impact of the support geometry on the thermoelastic behavior of the complete supported structure. However, at the moment our proposed approach is computationally cheap and provides a fair approximation of the residual displacements, indicating where the supports are needed.

3.2.4 Assessment of the optimized supports

In order to appreciate the effect of the support structure, a 2D simulation has been performed with *Simufact Additive* ® to compare the total displacements endured by the shape with the different supports obtained in Figure 14 : more precisely, the total displacements of the structure at the end of the manufacturing process, computed as the magnitude of the displacement vector, is plotted. According to *Simufact Additive* ®, the part with the supports obtained from the multiple load optimization (15) endures more displacements than the part with the supports

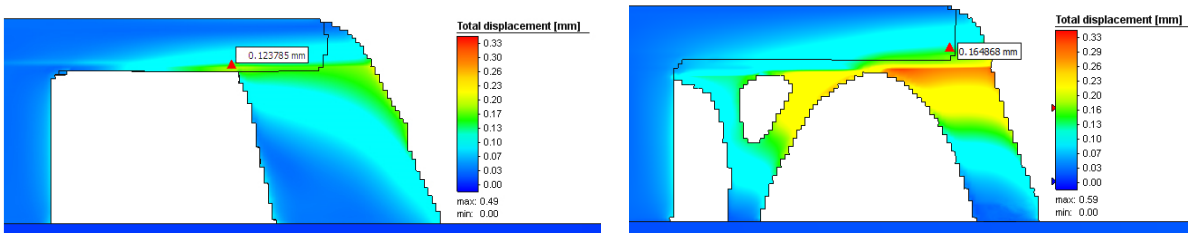

Figure 15: Manufacturing simuation by *Simufact Additive* ® of the T-shape supported with the thermal load optimized structure (left) (maximum displacement of the part : 0.12 mm), and with multiple load optimized structure (right) (maximum displacement of the part : 0.16 mm)

deduced from optimizing (14) (thermal loads only), for the same volume. The support structure of Figure 15 (left) is more efficient to offload the tip of the part, where the displacements are the biggest, and to maintain the overhang surface allowing the part to be manufactured correctly. Surprisingly, *Simufact Additive* ® does not detect any problem around the overhang region which is not supported, probably because of the layer deposition model. It highlights the fact that the physical phenomenon happening on horizontal surfaces is difficult to capture.

In practice supports are often made with lattice materials, which can be considered as homogenized material with lower stiffness than the pure material. Therefore, it makes sense to redo the same optimization and assessment with half reduced stiffness in the supports. Our numerical result indicates that, keeping the same volume and just changing the support rigidity, does not change much the optimized geometry of the supports (only the objective function increases). A more complex mechanical model could be considered to better take into account the lattice structure of supports, as well as other features related to the ease of removal of the supports used in the AM industry.

3.3 Penalization of the contact shape/support

As underlined previously, contact surfaces between the shape and the support often need special attention in the post-processing stage. It is therefore preferable to minimize the area of such contact surfaces. A simple way to do this is to add a penalization term computing the area of the parts of $\partial\omega$, which are not overhanging (namely outside $\Gamma_\beta(\partial\omega)$ and that are in contact with the support S. The shape derivative of a surface integral involves the mean curvature which is delicate to evaluate numerically. An alternative approach is preferred which consists in computing the following volumic approximation

$$\mathcal{P}_\varepsilon(S) = \frac{1}{\varepsilon L}\mathrm{Vol}\left(\{x \in \omega^c : d(x, \partial\omega \setminus \Gamma_\beta(\partial\omega)) < \varepsilon L\} \cap \{\varphi_S < \varepsilon L\}\right), \tag{16}$$

where ω^c is the complement of ω, d is the distance function, φ_S is the level set function of the support and L is a characteristic length of the shape. This function $\mathcal{P}_\varepsilon(S)$ computes the volume of an εL-layer around the non-overhanging boundary $\partial\omega \setminus \Gamma_\beta(\partial\omega)$ of the part ω, which is contained in the support S. Obviously, when $\varepsilon \to 0$ the function $\mathcal{P}_\varepsilon(S)$ converges to the area of the contact between $\partial\omega \setminus \Gamma_\beta(\partial\omega)$ and the support S. In numerical practice, the parameter ε is chosen as $\varepsilon = dx/L$, where dx is the size of a typical mesh cell.

Adding this kind of penalization in the objective function may force the support to avoid touching ω and rather to take roots directly on the baseplate. An illustration of the desired effect of the penalization term is given in Figure 16. Section 4 contains three dimensional examples where this penalization approach is effective.

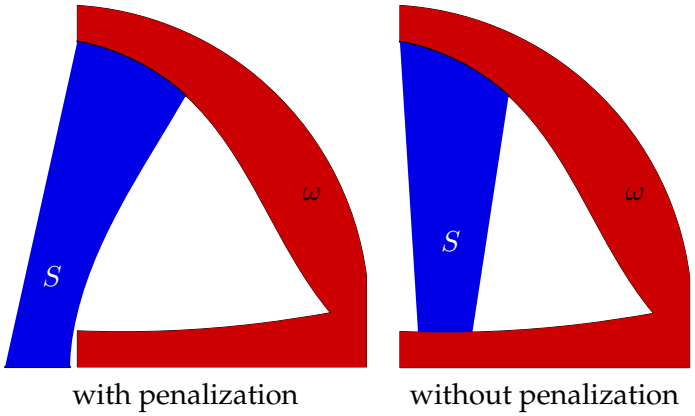

Figure 16: Desired effect of the penalization term (16) on the optimized support S: it avoids contact with the shape ω.

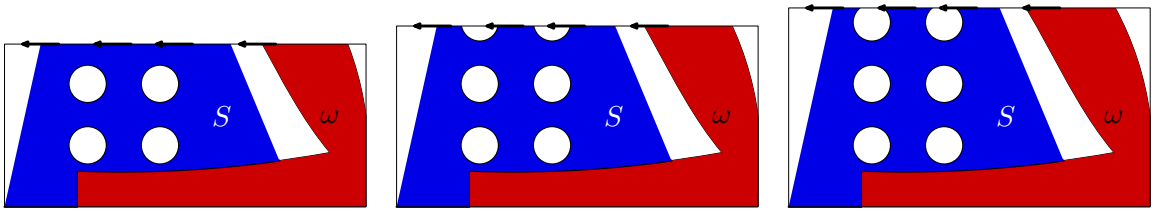

Figure 17: Three successive stages of the layer-by-layer fabrication process and their corresponding horizontal loads.

3.4 Resistance to lateral forces

The passage of the roller at each layer of the fabrication process induces high friction forces on the shape ω and on the support S. These forces are caused by multiple factors like: the layers are very fine (tens of micrometers), the thermal gradients may cause the structure to lift upwards. Therefore the supported structure should be as resistant as possible at any one of the layers with respect to these lateral efforts.

This observation inspired the following model. Consider a design domain D containing ω (typically a rectangular bounding box). The domain D is cut into N equal layers up to heights $h_1 < ... < h_N$. In the numerical computations these N layers are meshed exactly. For the layer i of height h_i the points on the upper boundary $x_d = h_i$ which are inside the shape ω or the support S are identified. The associated elasticity problem with horizontal surface loadings on $\{x_d = h_i\} \cap (\omega \cup S)$ and Dirichlet boundary conditions on the baseplate $\{x_d = 0\}$ is considered in the supported structure (the union of structure and supports) below height h_i. Figure 17 illustrates three stages of the layer-by-layer simulation, which is computationally expensive. The objective function is the sum of compliances for all the intermediate problems corresponding to heights $h_1, ..., h_N$. Figure 18 gives the optimized supports obtained using this model for a simple two-bar shape built with 20 layers. These supports match our intuition behind the proposed model: in order to better sustain lateral forces, supports are made of oblique trusses.

Note that, as was done in (15), this criterion could be used in a multiple load optimization

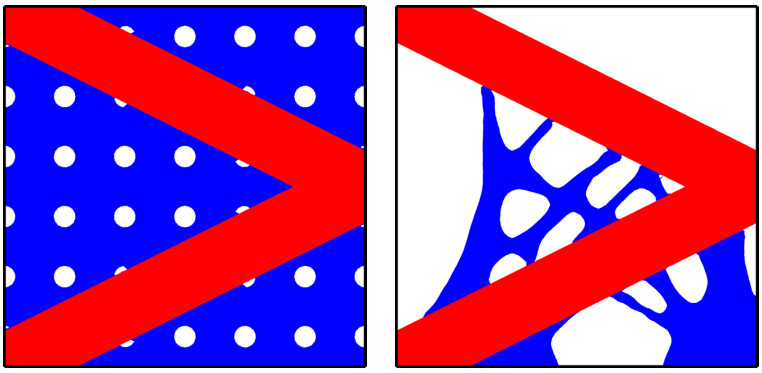

Figure 18: Optimized supports (blue) for better resistance to lateral loads of two-bar shape (red).

number of triangles	46964	606916	1779644
$\mathcal{G}_1$	0.004s	0.03s	0.09s
$\mathcal{G}_3$	0.1s	1.26s	5s
$\mathcal{G}_4$	0.15s	1.82s	7.8s

Table 2: Computation time (in seconds) to compute one criterion for shapes ω of variable complexity. The top row shows the number of triangles.

problem with other criteria as those previously discussed in this paper. This model is clearly dependent on simulations made at each one of the stages of the manufacturing process, therefore the simulation time is proportional to the number of layers considered in the simulation. This makes three dimensional applications quite costly although, each computation in an intermediate stage being independent of the others, it is easily amenable to parallel computing.

4 Numerical results in 3D

In this section all of the previously discussed models are applied to realistic test cases in 3D. A notable exception is the model for resistance to lateral forces discussed in Subsection 3.4, which, as a layer-by-layer model, necessitates important computational costs in 3D. All other models described are simple enough so that the simulations take at most a few hours of CPU time. The computational time is an important constraint from an industrial point of view: typically, the duration of the simulation should be smaller than the duration of the AM fabrication process.

4.1 Optimizing the orientation.

The models for optimizing the orientation shown in Section 2 are run for various STL files of different sizes. In order to illustrate the computational cost of these algorithms, some benchmarks are represented in Table 4.1 showing the time necessary for one evaluation of the functionals to be optimized. As underlined before, these computations based on the triangulation obtained from the STL files are quite fast even in our non-optimized Matlab implementations. These computations are completely parallelizable, so the computational cost may be greatly reduced on multi-core machines. The computational complexity for the area of supported regions $\mathcal{G}_1$ and the computation of the projection on a plane $\mathcal{G}_2$ is the same. Recall that $\mathcal{G}_3$ is the volume under the supported regions and $\mathcal{G}_4$ is the variation of the areas of parallel slices (100 equidistant slices for the computations shown in Table 4.1).

In the following, we illustrate our criteria regarding the optimization of the orientation, introduced in Section 2, with an example of structure ω, which is a U-shaped tube, as shown in

picture 1 of Figure 19. This example is interesting because it features some non-accessible parts (i.e. regions that cannot be reached by at least one straight line issued from the boundary of an enclosing box) and, according to the various proposed criteria, the optimal orientation will change dramatically when imposing that these non-accessible regions should not be supported.

As a first step, the inaccessible regions of the boundary of ω using methods of Subsection 2.2. These regions are shown in the picture 2 in Figure 19. Next, the approach described in Remark 2.3 is applied: the orientation which minimizes $\mathcal{G}_1$, the area of the supported regions (see Subsection 2.1.1), with the limit angle $\beta = 45°$. The build direction found for this optimal orientation is represented by a red arrow in picture 3 of Figure 19. Once this build direction is fixed, as underlined in Remark 2.3, rotating the shape in the horizontal plane does not change the supported area. The projection of ω on the plane of the roller is then minimized using the algorithm of Subsection 2.1.2. The optimal direction of the roller is represented by a green arrow in picture 3 on Figure 19. Minimizing $\mathcal{G}_3$, the volume under surfaces needing supports, as proposed in Subsection 2.1.3, gives the orientation shown in picture 4. Minimizing the functional $\mathcal{G}_4$ (see Subsection 2.1.4), the variation of the area of 100 parallel slices, gives the orientation in picture 5. Note that a vertical flip of the shape does not change the variation of the areas of parallel slices $\mathcal{G}_4$. Interestingly, for this U-shaped tube the minimization of $\mathcal{G}_1, \mathcal{G}_3$ and $\mathcal{G}_4$ give the same orientation. However, this orientation requires supporting non-accessible regions and is thus not useful from a practical point of view.

Up until now, the optimal orientation obtained did not take into account the non-accessible regions. Looking at the orientation obtained in the third picture in Figure 19 one can check that non-accessible regions are supported, and the same happens in the minimization of the volume. To avoid this drawback, another orientation optimization is performed with the function $\mathcal{G}p_{1,\varepsilon}$, defined in (4), which avoids supporting non-accessible regions. The regularization parameter is $\varepsilon = 0.05$ and the weight η is set as follows: triangles that are accessible have weight 1, while triangles which cannot be supported have weight 10^6. This change in the objective function gives the result shown in picture 6 of Figure 19.

This case study shows how the criteria proposed in Section 2 can be applied in order to choose an optimized orientation. Even though the optimal orientations will not be the same when changing the optimization criterion, they may give useful information and guide the user to consider other, maybe less-intuitive orientations for the shape ω. Further generalizations of this work may include the multi-objective optimization of these functionals or some variant of minimization under constraints.

4.2 Shape and topology optimization - pseudo gravity loads

The U-shaped tube ω from the previous subsection is considered with the orientation given by the sixth picture of Figure 19. Supports are obtained by the topology optimization algorithms of Subsection 3.1. More precisely, we consider the following two criteria:

- minimization of compliance (8) with respect to pseudo gravity surface loads,
- penalization (16) of unnecessary contact surfaces between the shape and its support.

In the same spirit as the 2-d example of Figure 9, a limit angle $\beta = 45°$ is imposed and vertical gravity loads are applied on the overhang surfaces $\Gamma_\beta(\partial\omega)$. The setup of the numerical algorithm is discussed below.

- **Construction of the mesh**: starting from an STL file, a mesh is constructed around the triangulated U-shaped tube ω using FreeFEM [24] and tetgen [38]. This mesh will serve as a design space for the supports. The re-meshing software MMG3D [16] is also used in order to increase the quality of the mesh and make it more suitable for shape optimization problems.
- **Identification of overhangs**: FreeFEM [24] allows us to easily identify the parts of the boundary of the mesh which correspond to overhang regions. The function `change` allows the attribution of particular labels for different zones of the boundary $\partial\omega$: the free

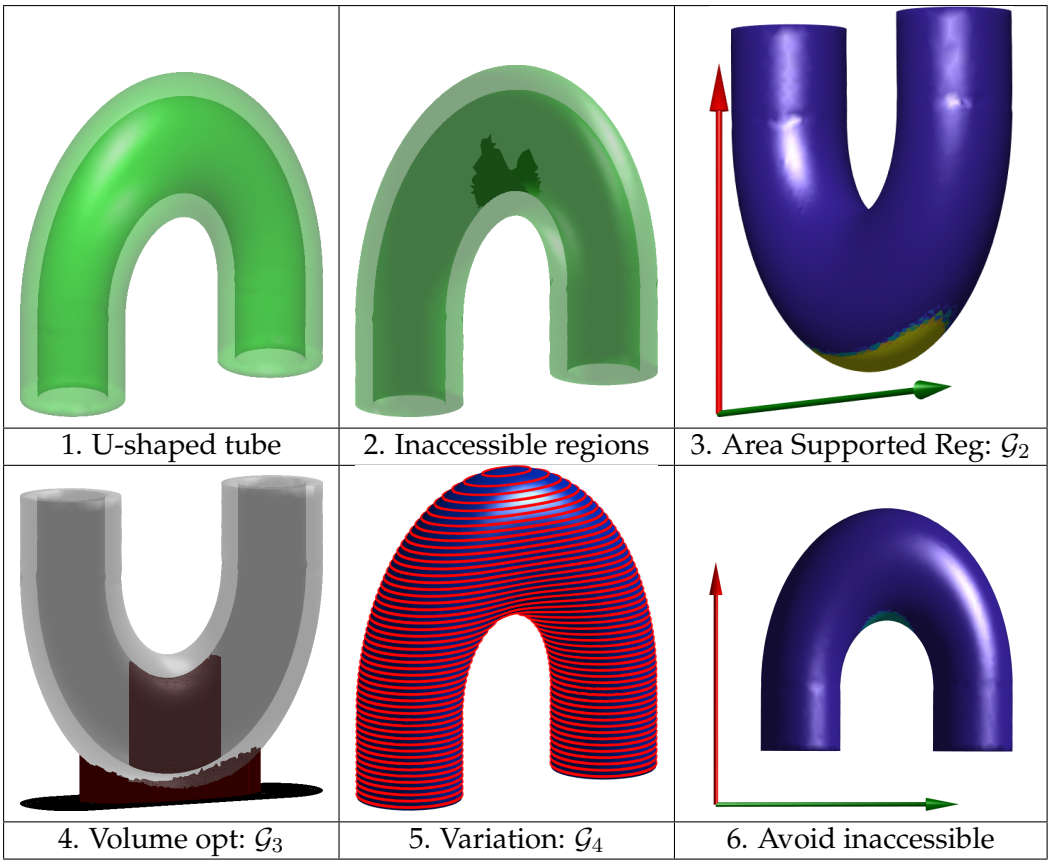

Figure 19: Case study: U-shaped tube. The criteria of Section 2 are tested, including the detection of the inaccessible regions (see Section 4.1).

boundary (i.e., none of the followings), the baseplate, the overhang regions Γ_β, the parts of $\partial\omega$ where the support may fix. These labels are then used to either add surface loadings or other boundary conditions. The mesh of the U-shaped tube with different labels is shown in Figure 20.

- **Compliance minimization:** once the mesh is constructed and boundary conditions are set-up, the linearized elasticity system (7) can be solved and the compliance is minimized with a volume constraint on the supports.

The working domain D, containing ω, is of size $2 \times 1 \times 2$. The mesh of $D \setminus \omega$ consists of 103237 nodes and 532242 tetrahedral elements. The cost of one iteration of the optimization loop in FreeFEM [24] is of about two minutes, including the resolution of the linearized elasticity system, the re-distancing algorithm and the advection step. The mechanical parameters are 1 for the Young modulus and 0.3 for the Poisson's ratio. The pseudo gravity load used in the equation (7) is $g = (0, -0.5)$. A Lagrangian algorithm, as in [3], is used for taking into account the volume constraint in the shape optimization.

Two different optimized supports are shown on Figure 21. The first one (on the left of Figure 21) is obtained by allowing the support to attach, not only on the baseplate, but also on almost horizontal parts of the boundary $\partial\omega$. In our computations, the part of $\partial\omega$ where the support is allowed to fix is identified by the condition $\vec{n} \cdot \vec{d} > 0.2$, where $\vec{d}$ is the unit build direction and $\vec{n}$ is the unit normal to $\partial\omega$. The volume constraint in this case is 0.1. The resulting support, although having a smaller volume than the vertical supports, touches the shape $\partial\omega$ exactly on the inaccessible region and this should be avoided. Moreover, for post-processing issues, it is always preferred to avoid the unnecessary contact between the support S and the shape ω. For comparison purposes, the same computation was also made with volume constraint equal to 0.16 and the corresponding compliance is shown in Table 3. The resulting supports have

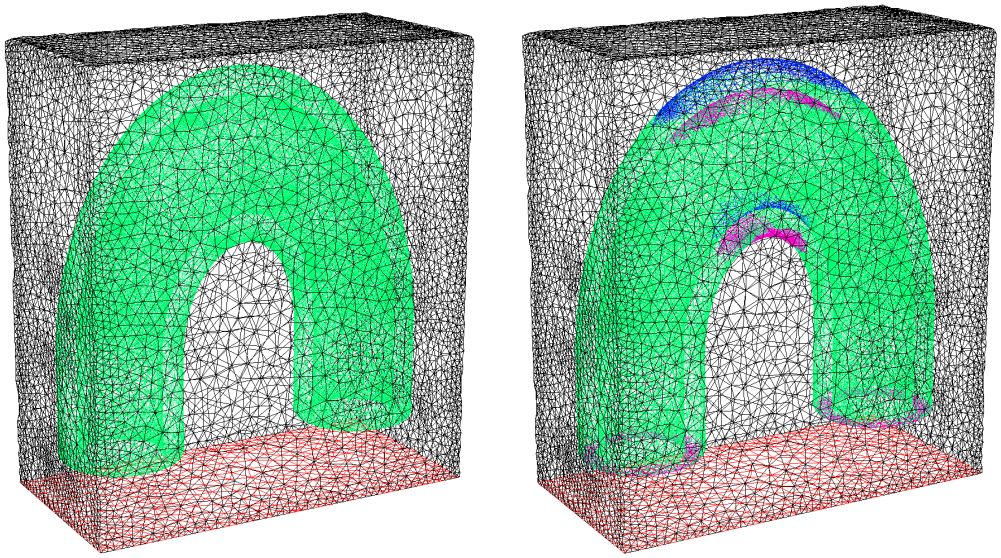

Figure 20: Mesh around the U-shaped tube ω obtained with MMG (left). Mesh with different colors for the different parts of the boundary of ω (right): green - free boundary, red - baseplate, purple - overhang regions Γ_β for $\beta = 45°$, blue - regions of $\partial\omega$ where the support may attach.

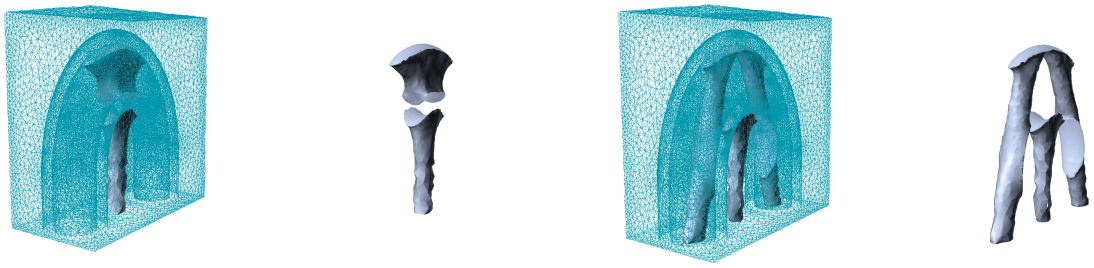

Figure 21: Optimized supports for minimal compliance under pseudo gravity loads. On the left, $u = 0$ is imposed on a part of $\partial\omega$ and on the baseplate. On the right, $u = 0$ is imposed only on the baseplate.

the same geometry as those shown in Figure 21, but they are more massive due to the larger volume constraint.

The second one (on the right of Figure 21) is obtained by forbidding the support to be attached to the structure ω. In other words, the Dirichlet boundary condition $u = 0$, for the elasticity equations (7), apply only on the baseplate Γ_D. The volume constraint in this case is 0.16 and the optimization algorithm has 150 iterations. It can be seen that although the new support, of course, is attached only on the baseplate, this does not prevent the contact between S and ω (there is no mechanical contact model here ; it is only a free boundary with homogeneous Neumann boundary condition).

In order to avoid the contact between the shape and the support a penalization term of the form (16) is used in a new simulation where the following objective function is minimized

$$\mathcal{C}(S) + P_\varepsilon(S),$$

where the compliance $\mathcal{C}(S)$ is defined by (8) and the penalization $P_\varepsilon(S)$ by (16). The parameter ε is chosen using the formula dx/L, where dx is the mesh size and L the diameter of D (as underlined in Section 3.3). The volume constraint is equal to 0.16 and the optimization algorithm has 150 iterations. The optimized supports for this new penalized function are displayed on Figure 22. It can be seen that this new support does not touch $\partial\omega$ anymore (except where

	Fix on $\partial\omega$	Fix only on the baseplate	Penalization
Compliance	0.12	0.55	0.57

Table 3: Comparison for the compliance of the three models using pseudo gravity loads for the same volume constraint $|S| = 0.16$: Dirichlet conditions on $\partial\omega$ and the baseplate, Dirichlet conditions only on the baseplate, Penalization of the contact.

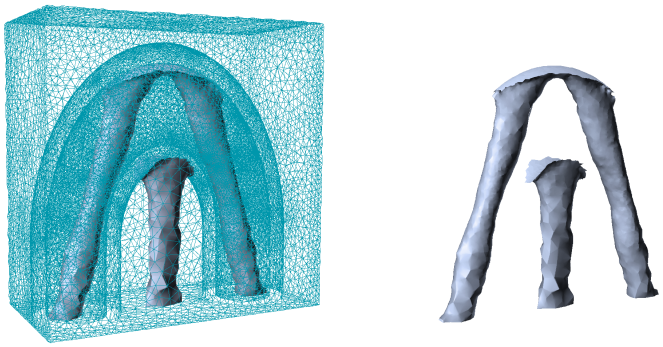

Figure 22: Optimized supports for minimal compliance under pseudo gravity loads with a penalization of the contact between the support S and the shape ω: the final support attaches only on the baseplate and does not touch ω.

it supports overhanging surfaces). The results obtained using this simple method give new insights for choosing original support structures, which are different of those obtained using purely geometric algorithms. A comparison of the compliances for the three models shown above is considered in Table 3. The case where supports may fix on the shape, naturally gives the lowest compliance. Furthermore, it can be seen that penalizing the contact between the shape and support slightly raises the value of the compliance.

Another possible application of the proposed method is the design of mutualised supports when several copies of the same shape ω are to be fabricated on the same plateau. On Figure 23 optimized supports are displayed for three half-tubes, oriented with a horizontal $120°$ rotation between them. Interestingly, a single support is preferred to three individual supports. The computational box D is $[-2, 2.2] \times [-2.4, 2.4] \times [0, 4]$. The volume constraint is 1.1 for the case where supports are allowed to touch the shape (left in Figure 23) and 1.3 for the second case where the contact between the supports and the shape is forbidden (right in Figure 23). The optimization algorithm takes 150 iterations.

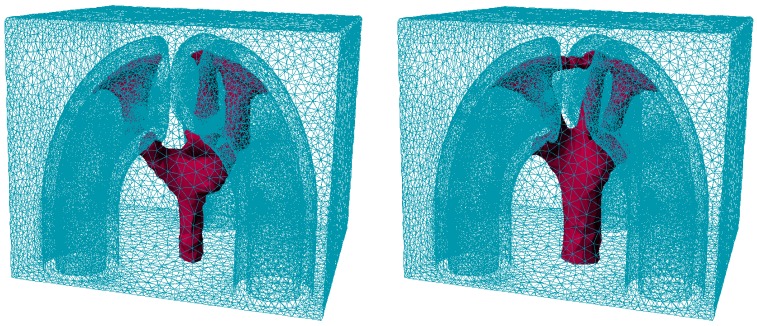

Figure 23: Optimized supports for minimal compliance under pseudo gravity loads for three copies of half-tubes: supports allowed to attach on the shapes (left), supports not allowed to attach on the shapes (right)

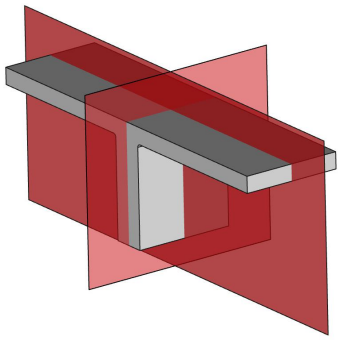

Figure 24: T-shaped twin cantilever: complete part with its symmetry planes. The computation is performed only in one quarter.

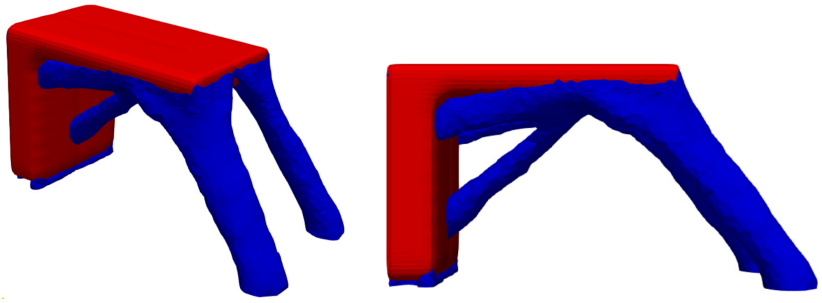

Figure 25: Optimized supports for one half of the twin cantilever obtained for compliance minimization with equivalent thermo-mechanical forces, without pseudo gravity loads (not suitable for AM).

Remark 4.1. *One possible critique that can be raised against our method is the lack of control of overhanging regions in the support S itself. There are works dealing with this aspect in the SIMP approach (e.g. [33]). In the level set framework, the papers [4], [43] shows how to perform support-free optimization. An obvious extension of our work is to include in the optimization process a constraint limiting the overhang regions in the support structures.*

4.3 Shape and topology optimization - equivalent thermoelastic loads

The approach described in Subsection 3.2 is now applied to another three dimensional example: a T-shaped twin cantilever shown in Figure 24. This shape is a well known example where the fabrication process induces large deformations at the extremities of the upper part. In order to reduce the complexity of the computations, the symmetry of the shape is exploited and only one quarter of the domain is used for all simulations (including optimization), as shown in Figure 24. The resulting (one fourth) computational domain D is $[0, 5] \times [0, 2] \times [0, 1.5]$ meshed using 50675 nodes and the optimization procedure takes 150 iterations. Contrary to the method used in the previous subsection, where only the support was simulated, here both the shape ω and its support S come into play in the equations.

As already explained in Subsection 3.2, using only the model (14) involving the compliance related to the thermal displacements of the shape may lead to non-manufacturable results as shown in Figure 25. Only the half of the twin-cantilever is displayed here. Considering the multiple load model, defined by (15), better supports are obtained in Figure 26. Here, the support not only prevents the thermal deformations, but also supports all overhanging regions. The volume constraints for the supports are $|S| = 0.2\ |D| = 3$ for the thermal load model and $|S| = 0.25\ |D| = 3.75$ for the multiple loads model.

To verify the efficiency of these supports obtained through a topology optimization process,

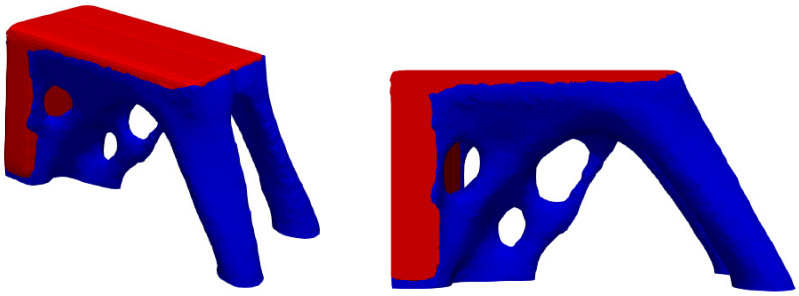

Figure 26: Optimized supports for the twin cantilever obtained for multiple loads compliance optimization with both pseudo gravity loads and equivalent thermo-mechanical forces.

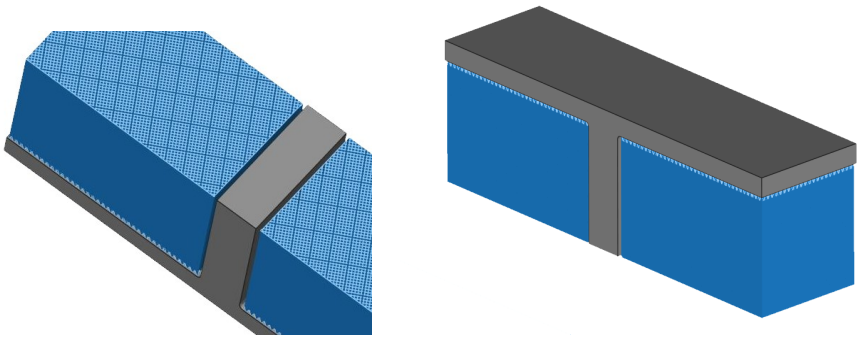

Figure 27: Block supports used by the software (left). The twin cantilever and its supports generated by *Materialise Magics* ® (right).

a comparison is made with the commercial software *Materialise Magics* ® [32], commonly used by additive manufacturing companies. It proposes a module able to design supports structures under overhang surfaces. This software was used to automatically generate support structures (see Figure 27) for the twin cantilever: they are made of thin walls blocks, so the support has a quite small volume fraction. Then, the software *Simufact Additive* ® is used to perform a simulation of the manufacturing process and thus a comparison of our supports with those of *Materialise Magics* ®. The total volume of our supports and of those of *Materialise Magics* ® is the same. The results of both simulations are displayed on Figure 28 where the total displacements at the end of the manufacturing process are plotted.

This comparison shows that the displacements at the end of the manufacturing process are smaller inside the twin cantilever (alone, not including supports) when it is maintained with our supports (maximum displacement : 0.20 mm) than with the supports generated by *Materialise Magics* ® (maximum displacement: 0.45 mm). However, our optimized supports endure more displacements than the ones created by the software, but present no risks of contact with the roller according to *Simufact Additive*, that has a dedicated module to detect this type of collision. So the support structures manufacturing should not be a problem. To further limit these displacements, support structures should also be taken into account in the process simulation, and this is the subject of a future work. On the one hand, since the comparison is made for the same volume of supports, it is a fair comparison from the point of view of material consumption and building time. On the other hand, the design of our support structures does not take into account the removal process, and obviously they cannot be removed by hand, on the contrary of the supports generated by *Magics Materialise* ®. From the post-processing point of view, the comparison is thus less fair. Note however that, to obtain a good surface quality, machining is required on surfaces maintained by supports, even for the ones that can be easily removed. So our supports require no additional steps in the manufacturing process, but surely

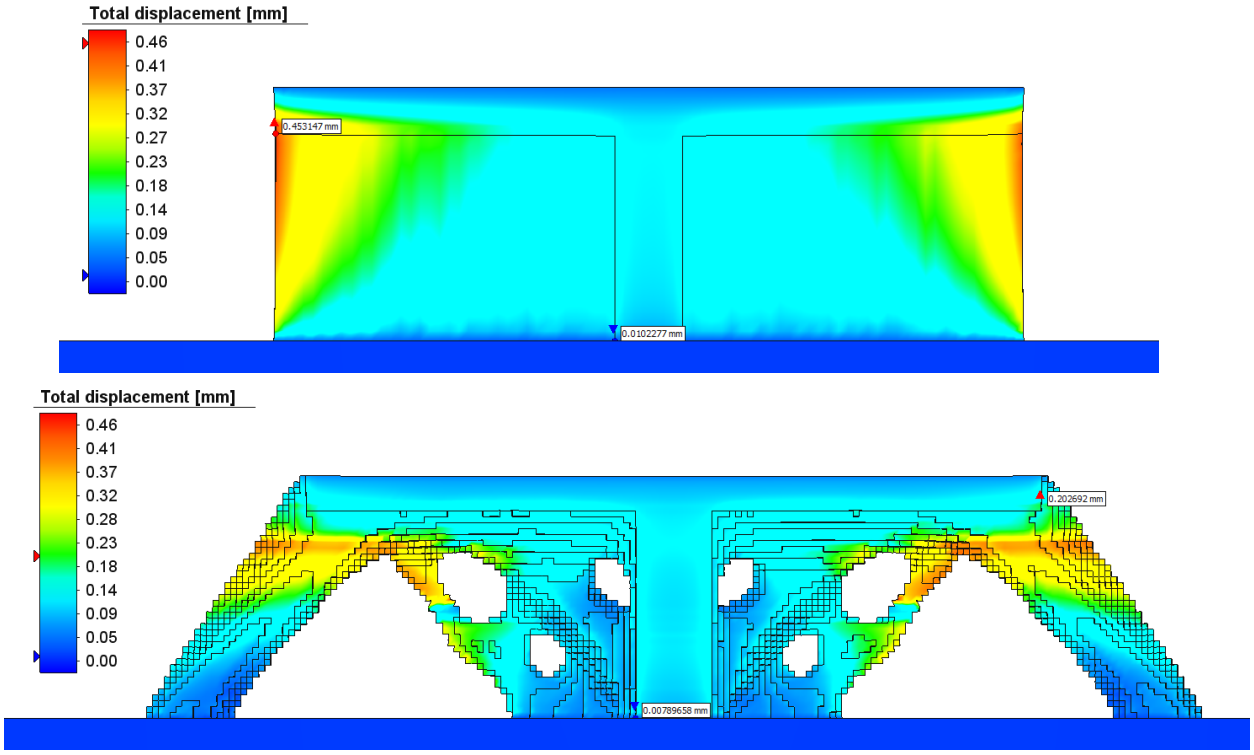

Figure 28: Manufacturing simulation by *Simufact Additive* ® of the twin cantilever maintained with *Materialise Magics* ® supports (maximum displacement of the part : 0.45) (top), and with optimal supports (maximum displacement of the part : 0.20 mm) (bottom) .

a heavier machining to remove them. For this reason, the mechanical model of our support structures will be improved in a future work.

5 Conclusions and perspectives

The first contribution of this paper is to propose some criteria for optimizing the build orientation of the object to be manufactured and to show their strong influence on the latter stage of shape and topology optimization of the supports (as can be checked on Figures 9 and 19). The suggested criteria are: the area of supported surfaces, the projection on a plane (typically that of the roller), the volume of vertical supports and the area variation of parallel slices. One must keep in mind that the relevance, and thus the choice, of these criteria depend significantly on the studied application. An important issue is to detect non-accessible regions which are inaccessible and finding the orientations which avoid supporting such regions.

Once the orientation of the part is fixed, supports can be added to facilitate its building and, of course, optimizing their shape and topology is a crucial issue. Our second contribution is to propose new models and criteria for this optimization, which turn out to be more relevant for industrial applications. In particular, pseudo gravity loads can be applied on overhanging surfaces, ensuring that the are evenly supported ; equivalent thermal loads, coming from a detailed thermo-mechanical simulation of the layer-by-layer fabrication process, can be used in a simple optimization loop ; penalization of the un-necessary contact between the shape and its support can be used to facilitate the support removal. For most of the criteria presented, realistic three dimensional computations are performed which show that the models are simple enough in order to be used for industrial applications.

There are many perspectives of our work. The accessibility constraint formulated in Section 2.2 is adapted for optimization problems where the shape is fixed. However, it is also of interest to optimize the shape under the same constraint, which would require to compute its gradient, a not so obvious task with our present formulation. Therefore, we are looking for alternative formulation of the inaccessibility constraint which should be differentiable with respect to the shape. Similarly, further constraints and models for the ease of removing supports should be taken into account. Another issue to be considered is that supports should not have overhanging regions themselves. Multiple works are available in the SIMP framework. In the level set framework we are working on an extension of the methods presented in [4] and [43]. The equivalent thermal load model can also be enriched by considering not only the deformations in the final shape ω, but also the deformations in the support structure S during the layer-by-layer manufacturing process. Of course, all these numerical simulations and optimizations should be compared with experiments, trying different type of supports.

6 Replication of Results

For the interested readers who want to reproduce our results the geometries and meshes used in our computations are made available at the following link

`http://www.cmap.polytechnique.fr/~beniamin.bogosel/supportsv2.html`.

In particular, the following are available:

- The geometries used in Section 2 given as Matlab structures containing the list of points and the corresponding triangulation.
- The meshes, initializations, final results and parameters for the results shown in Figure 9
- The Matlab file containing the triangulation of the U-shaped tube used in Figure 19 as well as the meshes, initializations, final results and parameters for the results shown in Figure 21.

Furthermore, the finite element software FreeFEM [24] is a free software which is available at the following link: http://www.freefem.org. The toolboxes MshDist and Advect are freely available on the GitHub page of the ISCD Toolbox: `https://github.com/ISCDtoolbox`.

7 Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgements

The PhD thesis of M. B. is supported by the Association Nationale de la Recherche et de la Technologie (ANRT), grant number CIFRE 2018/1405. This work was partially supported by the SOFIA project, funded by BPI (Banque Publique d'Investissement). G.A. is a member of the DEFI project at INRIA Saclay Ile-de-France. The authors thanks the anonymous referees for their remarks which, in particular, lead to a better formulation of the geometrical criterion of Subsection 2.1.4.

References

[1] P. Alexander, S. Allen, and D. Dutta. Part orientation and build cost determination in layered manufacturing. *Computer-Aided Design*, 30(5):343–356, Apr. 1998.

[2] G. Allaire. *Conception optimale de structures*, volume 58 of *Mathématiques & Applications (Berlin) [Mathematics & Applications]*. Springer-Verlag, Berlin, 2007.

[3] G. Allaire and B. Bogosel. Optimizing supports for additive manufacturing. *Struct. Multidiscip. Optim.*, 58(6):2493–2515, 2018.

[4] G. Allaire, C. Dapogny, R. Estevez, A. Faure, and G. Michailidis. Structural optimization under overhang constraints imposed by additive manufacturing technologies. *J. Comput. Phys.*, 351:295–328, 2017.

[5] G. Allaire, C. Dapogny, A. Faure, and G. Michailidis. Shape optimization of a layer by layer mechanical constraint for additive manufacturing. *C. R. Math. Acad. Sci. Paris*, 355(6):699–717, 2017.

[6] G. Allaire and L. Jakabčin. Taking into account thermal residual stresses in topology optimization of structures built by additive manufacturing. *Mathematical Models and Methods in Applied Sciences*, 28(12):2313–2366, 2018.

[7] G. Allaire, F. Jouve, and A.-M. Toader. Structural optimization using sensitivity analysis and a level-set method. *J. Comput. Phys.*, 194(1):363–393, 2004.

[8] C. Barlier and A. Bernard. *Fabrication additive - Du Prototypage Rapide à l'impression 3D*. Dunod, Paris, 2016.

[9] K. Bartsch, F. Lange, M. Gralow, and C. Emmelmann. Novel approach to optimized support structures in laser beam melting by combining process simulation with topology optimization. *J. of Laser Applications*, 31:022302, 2019.

[10] M. P. Bendsøe and O. Sigmund. *Topology Optimization*. Springer Berlin Heidelberg, 2004.

[11] C. Bui, C. Dapogny, and P. Frey. An accurate anisotropic adaptation method for solving the level set advection equation. *Internat. J. Numer. Methods Fluids*, 70(7):899–922, 2012.

[12] S. Cacace, E. Cristiani, and L. Rocchi. A level set based method for fixing overhangs in 3D printing. *Appl. Math. Model.*, 44:446–455, 2017.

[13] F. Calignano. Design optimization of supports for overhanging structures in aluminium and titanium alloys by selective laser melting. *Materials & Design*, 64:203–213, 2014.

[14] N. Chen and M. C. Frank. A method for metal am support structure design to facilitate removal. volume Solid Freeform Fabrication 2017 of *Proceedings of the 28th Annual International Solid Freeform Fabrication Symposium - An Additive Manufacturing Conference*. 2017.

[15] 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*, 113:1–23, Aug. 2019.

[16] C. Dapogny, C. Dobrzynski, and P. Frey. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *J. Comput. Phys.*, 262:358–378, 2014.

[17] C. Dapogny and P. Frey. Computation of the signed distance function to a discrete contour on adapted triangulation. *Calcolo*, 49(3):193–219, 2012.

[18] 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*, 14(sup1):1–13, May 2017.

[19] J. Dumas, J. Hergel, and S. Lefebvre. Bridging the gap: Automated steady scaffoldings for 3d printing. *ACM Trans. Graph.*, 33(4):98:1–98:10, July 2014.

[20] B. Ezair, F. Massarwi, and G. Elber. Orientation analysis of 3d objects toward minimal support volume in 3d-printing. *Computers & Graphics*, 51:117–124, Oct. 2015.

[21] M. Gan and C. Wong. Practical support structures for selective laser melting. *Journal of Materials Processing Technology*, 238:474 – 484, 2016.

[22] I. Gibson, D. Rosen, and B. Stucker. *Additive Manufacturing Technologies*. Springer New York, 2015.

[23] V. Griffiths, J. P. Scanlan, M. H. 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*, 273(1):334–352, Feb. 2019.

[24] F. Hecht. New development in FreeFem++. *J. Numer. Math.*, 20(3-4):251–265, 2012.

[25] K. Hu, S. Jin, and C. C. Wang. Support slimming for single material based additive manufacturing. *Computer-Aided Design*, 65:1 – 10, 2015.

[26] X. Huang, C. Ye, S. Wu, K. Guo, and J. Mo. Sloping wall structure support generation for fused deposition modeling. *The International Journal of Advanced Manufacturing Technology*, 42(11):1074, Aug 2008.

[27] A. Hussein, L. Hao, C. Yan, R. Everson, and P. Young. Advanced lattice support structures for metal additive manufacturing. *Journal of Materials Processing Technology*, 213(7):1019 – 1026, 2013.

[28] Y.-H. Kuo, C.-C. Cheng, Y.-S. Lin, and C.-H. San. Support structure design in additive manufacturing based on topology optimization. *Struct. Multidiscip. Optim.*, 57(1):183–195, 2018.

[29] M. Langelaar. Topology optimization of 3d self-supporting structures for additive manufacturing. *Additive Manufacturing*, 12(Part A):60 – 70, 2016.

[30] M. Langelaar. Combined optimization of part topology, support structure layout and build orientation for additive manufacturing. *Structural and Multidisciplinary Optimization*, 57(5):1985–2004, 2018.

[31] S. Lemkeddem, F. Khelfaoui, and O. Babahani. Calculation of energy lost by radiation and convection during laser welding of ta6v titanium alloy. *Journal of Theoretical and Applied Physics*, 12(2):113–120, Jun 2018.

[32] Materialise Magics. `https://www.materialise.com/en/software/magics`.

[33] F. Mezzadri, V. Bouriakov, and X. Qian. Topology optimization of self-supporting support structures for additive manufacturing. *Additive Manufacturing*, 21:666–682, may 2018.

[34] R. Minetto, N. Volpato, J. Stolfi, R. M. M. H. Gregori, and M. G. da Silva. An optimal algorithm for 3d triangle mesh slicing. *Computer-Aided Design*, 92:1–10, 2017.

[35] A. M. Mirzendehdel and K. Suresh. Support structure constrained topology optimization for additive manufacturing. *Computer-Aided Design*, 81:1 – 13, 2016.

[36] K. Mumtaz, P. Vora, and N. Hopkinson. A method to eliminate anchors/supports from directly laser melted metal powder bed processes. *Proc. Solid Freeform Fabrication Symposium, Sheffield*, pages 54–64, 2011.

[37] A. Salmi, G. Piscopo, E. Atzeni, P. Minetola, and L. Iuliano. On the effect of part orientation on stress distribution in AlSi10mg specimens fabricated by laser powder bed fusion (l-PBF). *Procedia CIRP*, 67:191–196, 2018.

[38] H. Si. TetGen, a Delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Software*, 41(2):Art. 11, 36, 2015.

[39] Simufact. `https://www.simufact.com/additive-manufacturing.html`.

[40] G. Strano, L. Hao, R. M. Everson, and K. E. Evans. A new approach to the design and optimisation of support structures in additive manufacturing. *The International Journal of Advanced Manufacturing Technology*, 66(9):1247–1254, Jun 2013.

[41] J. Vanek, J. A. G. Galicia, and B. Benes. Clever support: Efficient support structure generation for digital fabrication. *Computer Graphics Forum*, 33(5):117–125, 2014.

[42] V. Vijayan. Ray casting for deformable triangular 3d meshes. `https://fr.mathworks.com/matlabcentral/fileexchange/41504-ray-casting-for-deformable-triangular-3d-meshes`, 2014.

[43] Y. Wang, J. Gao, and Z. Kang. Level set-based topology optimization with overhang constraint: towards support-free additive manufacturing. *Comput. Methods Appl. Mech. Engrg.*, 339:591–614, 2018.