

## Accessibility of support structures in topology optimization for additive manufacturing

van de Ven, Emiel; Ayas, Can; Langelaar, Matthijs; Maas, Robert; van Keulen, Fred

**DOI**

[10.1002/nme.6611](https://doi.org/10.1002/nme.6611)

**Publication date**

2021

**Document Version**

Final published version

**Published in**

International Journal for Numerical Methods in Engineering

**Citation (APA)**

van de Ven, E., Ayas, C., Langelaar, M., Maas, R., & van Keulen, F. (2021). Accessibility of support structures in topology optimization for additive manufacturing. *International Journal for Numerical Methods in Engineering*, 122(8), 2038-2056. <https://doi.org/10.1002/nme.6611>

**Important note**

To cite this publication, please use the final published version (if applicable). Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

# Accessibility of support structures in topology optimization for additive manufacturing

Emiel van de Ven<sup>1,2</sup>  | Can Ayas<sup>1</sup> | Matthijs Langelaar<sup>1</sup>  | Robert Maas<sup>2</sup> | Fred van Keulen<sup>1</sup> 

<sup>1</sup>Department of Precision and Microsystems Engineering, Faculty of 3mE, Delft University of Technology, Delft, The Netherlands

<sup>2</sup>Netherlands Aerospace Centre, Amsterdam, The Netherlands

## Correspondence

Emiel van de Ven, De Groene Haven 280, 2627CD, Delft, The Netherlands.  
Email: research@emielvanden.nl

## Abstract

Additive manufacturing (AM) and topology optimization (TO) have a synergistic relation, as AM can produce complex TO designs, and TO provides high-performance parts that utilize the form freedom provided by AM. Recently, TO has been tailored more toward AM with the inclusion of the minimum allowable overhang angle as a design constraint: resulting designs can be built without any support structures. This work is an extension thereof, by allowing support structures only if they are accessible, such that they can be removed after manufacturing. This is achieved by applying a conventional overhang filter twice, combined with basic operations such as geometry inversion, union, and intersection. The result is an accessibility-aware overhang filter that can be incorporated in TO. Compared with conventional overhang filtered designs, the accessibility filter results in increased part performance and better convergence behavior. Furthermore, a modular filter structure is presented to easily construct the accessibility filter, and its effectiveness is demonstrated on several numerical cases.

## KEYWORDS

accessibility of supports, additive manufacturing, overhang, topology design

## 1 | INTRODUCTION

The most attractive feature of additive manufacturing (AM) is the unprecedented degree of permitted form freedom. No longer bounded by the design limitations of conventional manufacturing techniques, such as casting or milling, the challenge becomes to find the geometrical layout of the component that truly gives the best performance.<sup>1,2</sup> This is exactly what is provided by topology optimization (TO). Compared with other structural optimization methods, such as size or shape optimization, it exploits maximum design freedom, and is widely recognized as a design approach perfectly suitable for AM.<sup>3,4</sup>

Although AM can produce geometrically complex parts created by TO, there are design restrictions that have to be taken into account to prevent an increase in production cost. For example, a design might require a large amount of supports if the angle between a down-facing surface and the base plate is below a material/process specific value, the so called critical overhang angle  $\alpha_{oh}$ .<sup>5-7</sup> Support structures not only increase the printing cost due to increased material and postprocessing cost, they can even become impossible to remove once the printing is completed.

-----  
This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. *International Journal for Numerical Methods in Engineering* published by John Wiley & Sons Ltd.

Several studies have been published that determine design rules for AM,<sup>8-11</sup> and implementation of these design rules into TO is an active research topic; see Reference 12 for an overview. Especially, minimizing the need for support structures received considerable attention. Three approaches have emerged: first by enforcing a minimum overhang angle on the surface of the geometry,<sup>13-16</sup> second by sweeping through the structure layer-by-layer to detect areas that violate the minimum overhang angle,<sup>17-21</sup> and, finally, by modeling a physical aspect of the printing process, accounting for the heating or the self-weight of each layer.<sup>13,22-24</sup>

Most of the methods mentioned above focus on completely eliminating all the supports. Consequently, the design becomes printable at the expense of its performance. However, in practice support structures might be acceptable if they are significantly beneficial for the part weight or performance. For example in high performance applications, the cost of support fabrication and removal might be irrelevant. Then, no performance shall be sacrificed to prevent support structures, unless these supports are inaccessible, that is, impossible to remove after the build. Therefore, only those regions which are inaccessible, such as internal channels and cavities, must be free of overhang. This can be enforced by the above mentioned support elimination schemes. Consequently, it is of paramount importance to be able to assess where the design is accessible.

The partial admittance of support structures in TO has been addressed in several studies. In Reference 16 support structures are assumed to be columnar, and the total support volume is constrained, while in Reference 25 the support layout is optimized simultaneously with the part, and minimized according to a cost function. The approach presented in Reference 25 is extended in Reference 26 to also optimize the support structure for machining forces required in post-printing operations. Although in these studies supports are reintroduced to improve performance, the ease of support removal is not taken into account. This could result in undesirable configurations where the admitted support cannot be removed after printing. Furthermore, in contrast to References 25 and 26, the aim of this study is not to generate an explicit support layout, but merely account for regions in which supports could be removed afterward. As such, no additional design variables are introduced for the support layout. In this study, we propose a TO methodology to enforce the minimum allowable overhang angle exclusively in regions where support structures are difficult to remove. This results in an accessibility filter for TO, which eliminates regions that both violate the minimum allowable overhang angle *and* are inaccessible. An integral part of the method is the use of an overhang filter. In this study the front-propagation-based overhang filter presented in References 21,27 is used, which can identify and suppress overhanging regions in an efficient manner on unstructured meshes.

Because the proposed accessibility filter is a combination of several individual filters, a framework is presented in which filters can be easily combined and reordered. This framework is then used to define the accessibility filter set, which is demonstrated on numerical examples.

This article is organized as follows. In Section 2, the concept of the accessibility filter is detailed, and the filter combination framework is explained. Then, numerical examples are given in Section 3, followed by the discussion (Section 4) and conclusions (Section 5).

## 2 | METHOD

After a brief introduction to TO, the accessibility filter is presented in this section. As the accessibility filter consists of a combination of other filters, we present a general framework to combine filters and calculate sensitivities, which allows for quick experimentation and adaptation. In this framework, the individual filter steps of the accessibility filter are described in detail.

### 2.1 | Topology optimization

There are several approaches to TO,<sup>28-30</sup> and in this article we use density-based TO,<sup>3,31</sup> where the layout of a part is specified throughout the design domain by a pseudo-density field  $\rho$ , which indicates for each location if it contains material ( $\rho = 1$ ), or is void ( $\rho = 0$ ). In order to avoid an integer programming problem, the optimization problem is relaxed by allowing intermediate densities ( $0 \leq \rho \leq 1$ ).

During the topology optimization, the density field is optimized to provide the best performance for an objective under certain constraints. This is an iterative procedure, where in every iteration the design is evaluated and then updated based on the sensitivities. The latter are the derivatives of the objective and constraints with respect to the design variables

(the density field  $\rho$ ), and indicate the effect of a change in density value on the objective and constraint values. The optimization is usually initiated with a homogeneous density field, for example,  $\rho = 0.5$ .

One of the most basic and well-studied optimization problems is compliance minimization subject to a volume constraint. The most common TO scheme for compliance minimization is given in Figure 1. The density field  $\rho$ , provided by the optimization algorithm, is first processed by a density filter, which smooths the density field to prevent artifacts such as checkerboarding<sup>32</sup> and introduces a length scale to prevent mesh-dependence. In this study, the filtered densities  $\rho^*$  are a weighted average of the densities  $\rho$  within a certain filter radius  $r$  around the point of interest  $\mathbf{x}_j$ <sup>33</sup>:

$$\rho_i^* = H_{ij}\rho_j, \quad (1)$$

$$H_{ij} = \omega_{ij} / \sum_k \omega_{ik}, \quad (2)$$

$$\omega_{ij} = \max(r - \|\mathbf{x}_i - \mathbf{x}_j\|, 0), \quad (3)$$

where  $\mathbf{H}$  is the filter matrix and  $r$  the length scale that is introduced. The filtered densities  $\rho^*$  are then used to evaluate the objective and the volume constraint. The process of calculating the objective  $f_0$  is depicted in Figure 2(A). First, the Young's modulus is evaluated in each element, which is scaled with the penalized filtered densities. The penalization is done using SIMP interpolation, in order to suppress intermediate densities in the final design.<sup>31,34</sup> The effective Young's moduli for material and void regions are  $E_0$  and  $E_{\min}$ , respectively. The lower bound  $E_{\min}$  is introduced to avoid singularity of the stiffness matrix  $\mathbf{K}$ . Then, the displacements are calculated with a finite element analysis (FEA), where  $\mathbf{f}$  and  $\mathbf{u}$  are the discretized force and displacement vectors resulting from the FEA, and the stiffness matrix  $\mathbf{K}$  is a function of the Young's modulus field  $\mathbf{E}$ . Finally the compliance is calculated. For the volume constraint  $f_1$ , the volume fraction is calculated by dividing the volume integral of the filtered densities  $\rho^*$  by the volume limit  $V_{\lim}$ , depicted in Figure 2(B). Altogether, the minimization problem can be formulated as

$$\begin{aligned} \min_{\rho} \quad & f_0 \\ \text{s.t.} \quad & \mathbf{K}\mathbf{u} = \mathbf{f}, \\ & f_1 \leq 0, \\ & \mathbf{0} \leq \rho \leq \mathbf{1}. \end{aligned} \quad (4)$$

## 2.2 | Accessibility of supports

Recall that the aim of this article is to obtain a geometry using topology optimization, of which the supports that are required to print the part are accessible for ease of removal. We define this requirement as only allowing supports that are

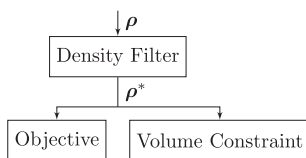
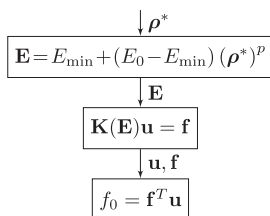
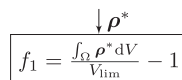


FIGURE 1 Basic TO scheme



(A) Objective calculation



(B) Volume constraint calculation.

FIGURE 2 Detailed objective and constraint evaluation for a compliance TO

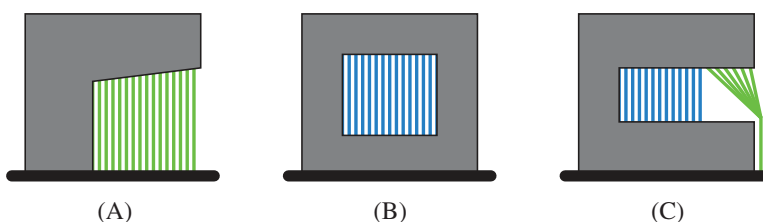
both (1) printable themselves, and (2) connected to the base plate. Examples of support that are accessible and inaccessible are given in Figure 3. Any location that cannot be supported by *accessible* supports, should be free of overhang. The first requirement is straightforward: supports must be realized through printing. The effect of requiring supports to connect to the base plate is to prevent supports from forming in internal voids, from which they cannot be removed (Figure 3(B)). Furthermore, it also prevents supports in internal channels, as the required supports cannot be built from the base plate (Figure 3(C)). Finally, when the supports connect to the base plate, only one side of the supports connects to the part. Supports that are connected to the part on both ends are thus not allowed. This can save postprocessing cost required to polish the surfaces affected by supports. Note that internal voids are inadmissible in certain printing technologies, but for this article we do not take that into consideration as other works have addressed this topic.<sup>35,36</sup>

In order to achieve the accessibility of supports, a filter is required that can take any geometry as input, and removes the regions that require supports that are inaccessible. Such a filter can then be inserted into the TO scheme. Before going into the details of the formulation, the concept of the accessibility filter will be explained on the 2D geometry displayed in Figure 4. If a conventional overhang filter would be applied to this structure, with  $\alpha_{oh} = 45^\circ$ , both overhanging areas (shaded red) are identified as not printable and thus removed by the filter. The overhanging area on the left should indeed be removed, as it is in an internal void and therefore the supports will be inaccessible for removal. However, the overhanging area on the right can be supported with a support connected to the base plate, as indicated by the green region in Figure 4. This overhanging region is thus accessible and should not be removed. Note that in order to allow supports to connect to the base plate, as shown by the accessible supports in Figure 4, the design domain should be slightly wider than the original design space. This is achieved by placing a void nondesign space around the geometry.

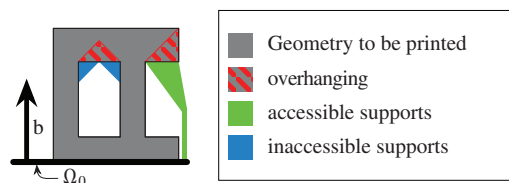
The basic principle of the accessibility filter is to apply a conventional overhang filter to the union of the original geometry and those regions where accessible supports can be printed. In order to achieve this, the following steps, visualized in Figure 5, are followed, starting with the input geometry shown in Figure 5(A):

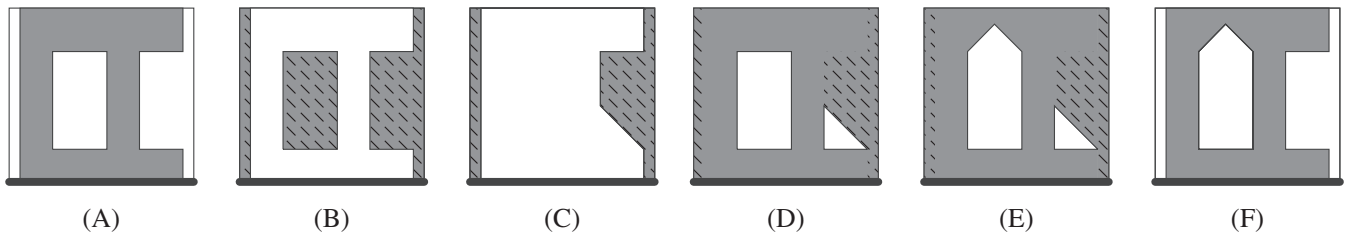
1. Obtain the region in which supports can be placed by inverting the domain, as shown in Figure 5(B). In order to differentiate between the original structure and the support region, the support region is shaded.
2. From the region in which supports can be placed, remove the regions that are not printable and/or accessible by applying a conventional overhang filter. In Figure 5(C) it can be seen that the enclosed void is indeed removed from the domain.
3. Combine the support region attained from the previous step and the original structure. The result, displayed in Figure 5(D), is the original structure, combined with all possible accessible supports.
4. Remove the regions from the original structure that cannot be supported by accessible supports by applying the conventional overhang filter once more. The result is given in Figure 5(E), showing that overhanging areas in internal voids are not printable since no accessible supports can be placed to print its overhanging top surface.
5. Finally, remove all the support material that was added in Step 3 by taking the intersection of the previous result with the original structure (Figure 5(F)). The inaccessible overhanging area is removed, while the accessible overhanging area is retained.

**FIGURE 3** Examples of accessible support that can be easily removed (green), and inaccessible supports that are difficult to remove (blue)

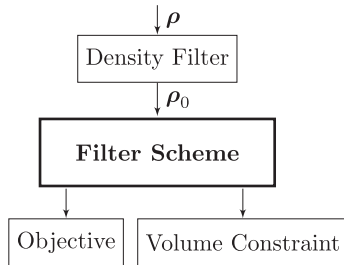


**FIGURE 4** Geometry that is to be printed on the base plate  $\Omega_0$  in the build direction indicated by **b**. Potential support structures are indicated. Supports of the left overhanging section in the cavity are inaccessible





**FIGURE 5** The progression of the geometry given in Figure 4 as the accessibility filter is applied step-by-step



**FIGURE 6** TO layout with generic filter scheme

Because of the many steps required for the accessibility filter, a general framework is set up to systematically organize and implement such a chain of filter operations. This is discussed in the next section.

### 2.3 | Filter structure

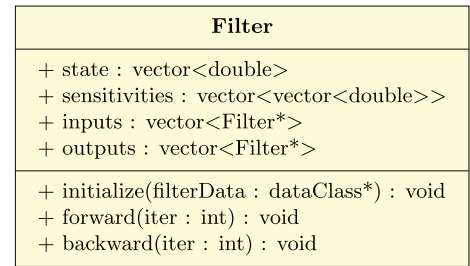
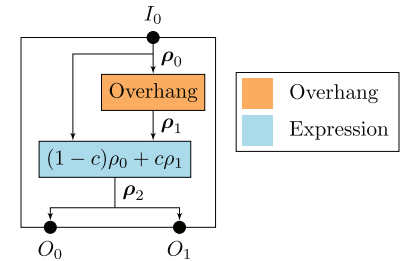
In order to construct the accessibility filter as outlined in the previous section, a scheme composed of several filters is to be inserted between the density filter and the objective and constraint evaluation, as can be seen in Figure 6. These filters might consist of simple expressions executed entrywise on a density field, similar to, for example, the SIMP penalization in Figure 2(A), or more complex operations such as a density or overhang filter.

Conventionally, the filter operations would be programmed in the order they occur, and the sensitivities have to be determined for the filter scheme. This can be a tedious task if the filter scheme consists of many operations. Furthermore, inserting an additional filter to the scheme requires reprogramming and compilation of the code. As the accessibility filter scheme contains a relatively large number of filters, a more flexible approach is taken, detailed in this section.

Although each filter performs a different operation, all the filters share some properties: they all produce a density field as output, and will have a function to process the filter and a function to calculate the sensitivities. To facilitate the systematic construction of complex filter chains, it is therefore convenient to approach the filter scheme in an object-oriented manner, where every filter is derived from the same base class: `Filter`, depicted in Figure 7. The `Filter` class has several attributes:

- `state`: a vector that contains the density field computed by the filter.
- `sensitivities`: a vector that for every response function contains a vector of the derivative of that response function to the output density field.
- `inputs`, `outputs`: vectors of pointers to the filters that are inputs/outputs to the current filter.

Furthermore, the `Filter` class has three functions: `initialize()`, which initializes the filter (e.g., set up the filtering matrix for a density filter), `forward()`, which applies the filter, and `backward()`, which processes the sensitivities. Filter-specific data, for example, filter radius or overhang angle, and general data such as mesh information or filter inputs and outputs, are passed to the `Filter` object when `initialize()` is called. To the other two functions, only the iteration number is passed.

**FIGURE 7** Diagram of the `Filter` class**FIGURE 8** Example of a filter structure where an overhang filter is followed by an expression filter that interpolates between the original density field  $\rho_0$  and the overhang-filtered densities  $\rho_1$ 

### 2.3.1 | Filter types

From the base class `Filter`, the following filter classes are derived:

#### *Expression filter*

The expression filter evaluates a mathematical expression containing only componentwise operations. It is the only filter that can have multiple inputs. This allows for example interpolation between several density fields, or boolean operations on the geometry. Furthermore, the expression can have a continuation variable  $c$  that takes a value based on the iteration number. An example of the expression filter is the application of SIMP penalization in Figure 2(A).

#### *Overhang filter*

The overhang filter removes regions that violate the minimum overhang angle from a design, and leaves the printable part. In this work the front propagation based implementation as presented in Reference 21 has been used.

### 2.3.2 | Filter layout and initialization

The filter schemes used in this work can all be represented as a directed acyclic graph: the filter scheme has one input  $I_0$ , one output  $O_k$  per response function  $k$ , and the scheme does not contain any loops. A typical example is given in Figure 8, which takes the filtered densities  $\rho_0$ , applies the overhang filter, and then interpolates between  $\rho_0$  and the overhang filtered densities  $\rho_1$ , based on a continuation parameter  $c$ . The output is the density field  $\rho_2$ , which is passed to outputs  $O_0$  and  $O_1$ , which are connected to the objective and volume constraint (Figure 6).

The filter scheme can be stored in a data file (e.g., XML<sup>37</sup> or JSON<sup>38</sup>), such that it can be read at runtime. The data file will contain a list of the filters, their inputs and outputs, and parameters such as the expression or continuation scheme for the expression filter, or the minimum overhang angle and build direction for the overhang filter. At runtime, the filters are read from the file and added to a list, `filterList`, and initialized. During initialization, the state and sensitivities vectors are allocated, inputs and outputs vectors set, and filter specific initializations performed.

During the evaluation of the filter scheme, every filter can only be evaluated once all its inputs have been evaluated. For example, in Figure 8, the expression filter can only be evaluated after the overhang filter. It is therefore convenient to order the `filterList` in topological ordering, such that for every filter in `filterList`, its inputs have lower indexes in `filterList`. The topological ordering can be achieved manually, by specifying the input file in the correct order, or by using a topological ordering algorithm such as Kahn's algorithm.<sup>39</sup>



### 2.3.3 | Evaluating the filter scheme

Since the `filterList` is in topological ordering, one can simply loop over the `filterList` and call the `forward()` function of each filter to evaluate the filter scheme. Each filter can access the state vectors of its inputs through the `inputs` vector, and perform the necessary manipulations to calculate its own state vector.

### 2.3.4 | Evaluating sensitivities

In order to calculate the sensitivities of each response function  $k$  toward the input densities of the graph  $\rho_0$ , the chain rule of differentiation is used to systematically evaluate the sensitivities filter by filter. For a filter  $i$ , whose output density field  $\rho_i$  is input to the filters  $N$ , the sensitivity of its output density field with respect to a response function  $f_k$  is defined as

$$\frac{df_k}{d\rho_i} = \sum_{j \in N} \left[ \frac{df_k}{d\rho_j} \right]^T \frac{\partial \rho_j}{\partial \rho_i}, \quad (5)$$

where  $\partial f/\partial \mathbf{a}$  represents a column vector  $\mathbf{g}$  with entries  $g_i = \partial f/\partial a_i$ , and  $\partial \mathbf{a}/\partial \mathbf{b}$  represents a matrix  $\mathbf{A}$  with entries  $A_{lm} = \partial a_l/\partial b_m$ . The sensitivities are propagated through the graph by calculating the right-hand side of Equation (5) in the filters  $j \in N$ , and adding this contribution to  $df_k/d\rho_i$ . Since the `filterList` is in topological ordering, traversing it backward, from end to beginning, will ensure that each filter  $j$  will only evaluate its contribution to its inputs when the term  $df_k/d\rho_j$  is already calculated.

---

**Algorithm 1.** Sensitivity calculation for response  $k$

---

**Initialize:**

$N_f = \text{LENGTH}(\text{filterList})$

$\frac{df_k}{d\rho_i} = \mathbf{0}, \quad i = 1, \dots, N_f$

For filter  $i$  directly connected to  $O_k$  set:  $\frac{df_k}{d\rho_i} = \frac{df_k}{d\rho_{O_k}}$

**for**  $i = N_f$  to 1 **do**

$N_i \leftarrow$  filters that are input to filter  $i$

**for**  $j \in N_i$  **do**

$\frac{df_k}{d\rho_j} += \left[ \frac{df_k}{d\rho_i} \right]^T \frac{\partial \rho_i}{\partial \rho_j}$

**end for**

**end for**

---

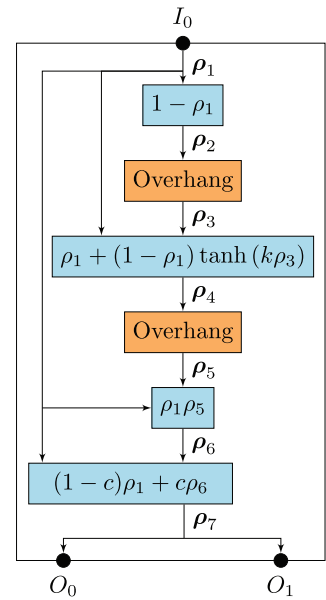
The sensitivity evaluation is initiated at the filters connected to the outputs  $O_k$ , and then propagated backward through the filter scheme, as described by Algorithm 1. For the scheme displayed in Figure 8, this would result in the following sensitivities

$$\frac{df_k}{d\rho_0} = \left[ \frac{df_k}{d\rho_{O_k}} \right]^T \left( \frac{\partial \rho_2}{\partial \rho_0} + \frac{\partial \rho_2}{\partial \rho_1} \frac{\partial \rho_1}{\partial \rho_0} \right), \quad k = 1, 2. \quad (6)$$

The sensitivities  $df_k/d\rho_{O_k}$  are obtained using adjoint sensitivity analysis for compliance minimization for the objective  $f_0$ , and standard differentiation of the volume constraint  $f_1$ .<sup>3</sup> Note that in Algorithm 1, the Jacobian matrix  $\partial \rho_i/\partial \rho_j$  can be calculated explicitly, as is usually done for, for example, the density filter. This matrix is usually sparse, for example, for componentwise operations it is diagonal. For more complex filters, for example, the overhang filter, an adjoint approach can be used to avoid explicit calculation of the Jacobian matrix, and instead directly calculate  $\partial f_k/\partial \rho_i$ .



**FIGURE 9** The accessibility filter scheme. The first six density fields correspond to the density fields shown in Figure 5



## 2.4 | Accessibility filter scheme

With the filter framework as defined in the previous section, the concept of the accessibility filter as presented in Section 2.2 can now be formulated in a similar manner. The full scheme is given in Figure 9. The first five filters exactly represent the five steps presented in Section 2.2, and the density fields  $\rho_1$ – $\rho_6$  resemble Figure 5(A–F) when applied to the geometry given in Figure 5(A). Numerical experience shows that convergence is improved when the accessibility filter is inactive for the first 10 iterations, and is then gradually activated over the next 10 iterations, until it is fully active. This is achieved in the last step, which is an interpolation between the original ( $c = 0$ ) and the accessibility filtered geometry ( $c = 1$ ), similar to Figure 8.

The inversion filter ( $\rho_1$  to  $\rho_2$ ) and the intersection filter ( $\rho_5$  to  $\rho_6$ ) are straightforward operations on the density fields. However, the superimposing filter ( $\rho_3$  to  $\rho_4$ ), is less trivial and will be discussed in more detail. The simplest union operation would be  $\rho_4 = \rho_3 + \rho_1$ . In theory, since  $\rho_2 = 1 - \rho_1$ , and  $\rho_3 \leq \rho_2$  because the overhang filter only removes material,  $\rho_3 + \rho_1 \leq 1$ . However, in practice, the overhang filter can slightly smooth the density fields,<sup>21</sup> and  $\rho_3 + \rho_1$  can be larger than 1. There are multiple possibilities to avoid this behavior, and after testing several options the union operation is chosen as

$$\rho_4 = \rho_1 + (1 - \rho_1) \tanh(k\rho_3). \quad (7)$$

This operation is an addition of the original density field  $\rho_1$ , and a term smaller or equal to  $1 - \rho_1$ , such that the combination cannot exceed a density of 1. The hyperbolic tangent function was used as it is bounded by 1. The parameter  $k$  scales how fast the second term of the right-hand side in Equation (7) approaches 1, and is chosen as  $k = 20$ .

## 3 | RESULTS

In this section, the accessibility filter is demonstrated both in 2D and 3D, for stiffness maximization as formulated in Equation (4). The Young's modulus is chosen as  $E_0 = 1$  Pa for material and  $E_{\min} = 1 \times 10^{-6}$  Pa for void, with a Poisson's ratio  $\nu = 0.3$ , and SIMP penalization factor  $p = 3$ . For the optimization, the method of moving asymptotes (MMA)<sup>40</sup> is used as optimization algorithm. The optimization is terminated when the maximum change of any of the design variables in subsequent iterations is smaller than 0.01, or when a maximum of 300 iterations has been reached. The continuation scheme and parameters for the accessibility filter are identical for all the presented cases, no case specific tuning has been applied.

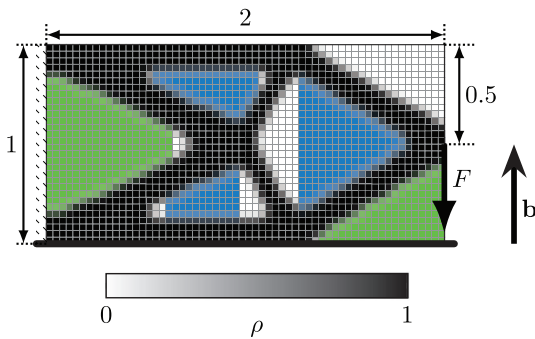
### 3.1 | Accessibility filter in 2D

#### 3.1.1 | 2D case: Cantilever

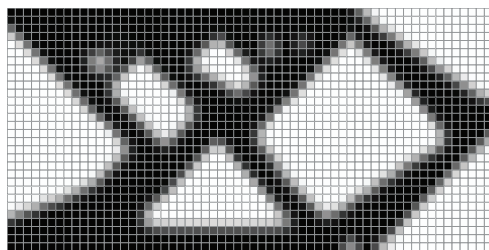
In 2D, the accessibility filter is demonstrated on a 2:1 aspect ratio cantilever beam case. For this purpose, a rectangular design domain is mechanically clamped on the left side, with a vertical point force acting on the right, as displayed in Figure 10. A 50% volume constraint is applied. The domain is discretized with  $60 \times 30$  square Q4 elements, and a density filter with a radius of  $2h$  is applied, where  $h$  is the element edge length. For demonstration purposes, it is assumed that the out-of-plane direction in the third dimension cannot be used for printing, or support removal. The geometry is assumed to be printed in the vertical direction as indicated with the vector  $\mathbf{b}$  in Figure 10. Unless specified otherwise, the displayed density field is the filtered density field that forms the input to the objective and volume constraint (see Figure 6).

The resulting density field generated by a conventional optimization without overhang or accessibility filter is displayed in Figure 10. The required support to print the part for a minimum allowable overhang angle  $\alpha_{\text{oh}} = 45^\circ$ , is indicated in blue and green. The support material required inside the three interior holes of the structure, indicated in blue, cannot be removed in a 2D setting, thus the design is not suitable for AM. The simplest method to enforce printability is to completely avoid the requirement of support material, with the addition of an overhang filter (see Figure 8 and Reference 21). The resulting design is displayed in Figure 11(A), which can be printed without supports.

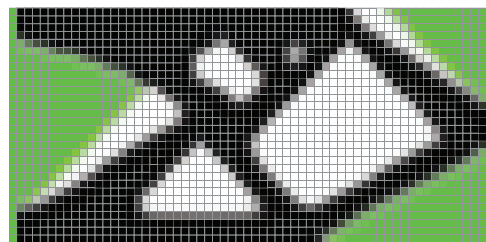
Depending on the type of component, one might allow supports, as long as they are removable, if, for example, performance can be gained. In this case, the accessibility filter is applied, as is displayed in Figure 9, which only suppresses overhanging regions that are inaccessible. The resulting design is displayed in Figure 11(B). Notice that the design domain is extended by one element in width on both sides to allow support material to connect to the base plate on both sides of the structure. The extra columns of elements are nondesign regions, and the boundary conditions are now one element from the actual boundary, such that the original aspect ratio is not influenced. As can be seen, the enclosed voids that are inaccessible can be printed without the need of support material, while support is required for the outside boundaries that are partially not printable without the use of support material. The field corresponding to  $\rho_3$  in Figure 9, is displayed in green in Figure 11(B). This field indicates the accessible region, thus everywhere in the domain where supports can



**FIGURE 10** Domain and boundary conditions for the cantilever case, with optimal geometry for a conventional stiffness maximization without accessibility filter. The green and blue regions indicate the accessible and inaccessible support, respectively, required for manufacturing with  $\alpha_{\text{oh}} = 45^\circ$ . Objective  $f = 74.2 \text{ N m}$



(A) Overhang filter. Objective  $f = 87.6 \text{ Nm}$ .



(B) Accessibility filter, allowing all supports that can be printed with  $\alpha_{\text{oh}} = 45^\circ$ . Objective  $f = 78.3 \text{ Nm}$ .

**FIGURE 11** Cantilever case result with overhang filter (A), and accessibility filter (B), with  $\alpha_{\text{oh}} = 45^\circ$ . The result in (A) can be printed without supports, while the green field in (B) indicates  $\rho_3$  from the accessibility filter (Figure 9), which is the printable support region. The design in (B) has a better performance because less material is sacrificed for manufacturability

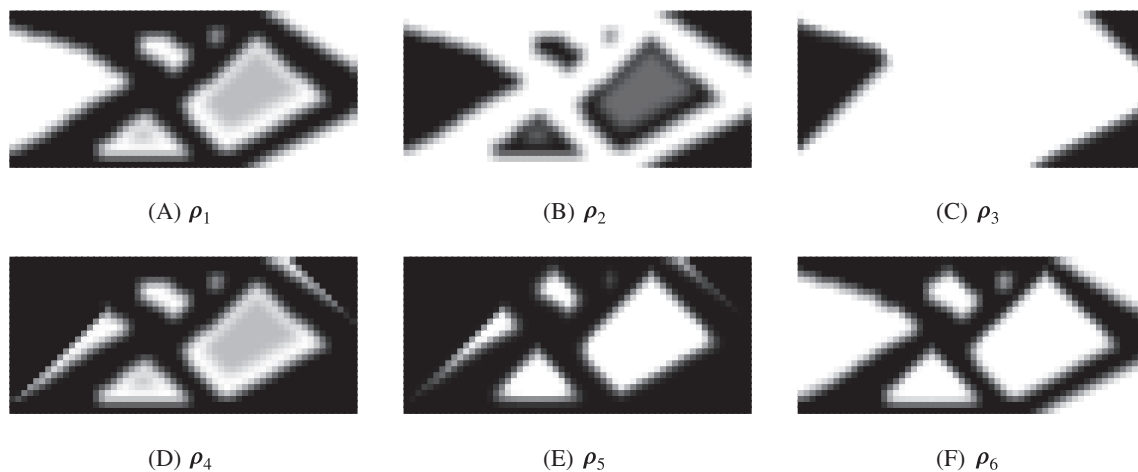
be printed that are directly connected to the base plate. The actual support layout will be only partially filling the green domain.

The consecutive filtering operations of the accessibility filter as illustrated in Figure 9, leading to density fields labeled as  $\rho_1$ – $\rho_6$ , are plotted in Figure 12. This figure is the numerical equivalent of the schematic picture in Figure 5. The filter is initiated with the filtered density field, and the regions of extension on both sides are nondesign regions where  $\rho_1 = 0$  (Figure 12(A)). This design is inverted (Figure 12(B)), and overhanging regions are removed (Figure 12(C)). The resulting field  $\rho_3$  (cf. the green regions in Figure 11(B)) indicates the printable support region. It is then combined with the original design  $\rho_1$  (Figure 12(D)), and the overhanging regions of  $\rho_4$  are removed (Figure 12(E)). Finally, by removing the printable support region  $\rho_3$ , the accessibility filtered design  $\rho_6$  is obtained (Figure 12(F)).

From the printable support field  $\rho_3$  in Figure 11(B), it can be seen that supports are required that are at the limit of the overhang angle. One can, for example, enforce columnar support by changing the minimum overhang angle used in the overhang filter in Step 2 of the accessibility filter (Section 2.2), referred to as  $\alpha_{\text{sup}}$ , to  $\alpha_{\text{sup}} = 90^\circ$ . The resulting design is displayed in Figure 13. Now, also part of the outer boundary is constrained by the overhang filter, and the design is a hybrid between Figure 11(A and B), which is also reflected in the performance, discussed next.

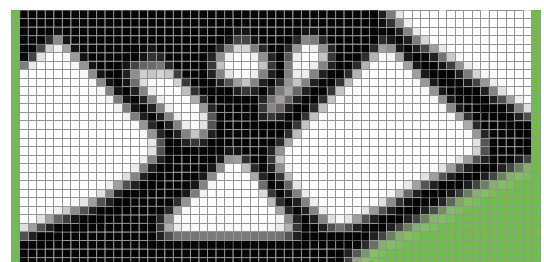
### 3.1.2 | Convergence

The convergence behavior of the presented 2D cantilever beam optimizations are shown in Figure 14. The overhang and accessibility filters are gradually introduced by a continuation scheme, which linearly ramps up the continuation parameter  $c$  (Figure 9) from 0 to 1 between iteration 10 and 20. Therefore, the convergence in the first 10 iterations is identical for all cases. As expected, the conventional optimization without overhang or accessibility filter has the lowest objective (maximum stiffness), at  $f = 74.2$  N m. The overhang filtered optimization, which is the most constrained, has the highest objective at  $f = 87.6$  N m. Since the accessibility filter allows for more overhang, the objective is between the conventional and overhang filtered designs at  $f = 78.3$  N m for  $\alpha_{\text{sup}} = 45^\circ$ , and  $f = 81.8$  N m for  $\alpha_{\text{sup}} = 90^\circ$ . Furthermore,



**FIGURE 12** Numerical results of the progression of the geometry shown in Figure 11(B) through the accessibility filter

**FIGURE 13** Cantilever case result with accessibility filter enforcing vertical supports. The green field indicates  $\rho_3$  from the accessibility filter (Figure 9), which is the printable support region. Objective  $f = 81.8$  N m



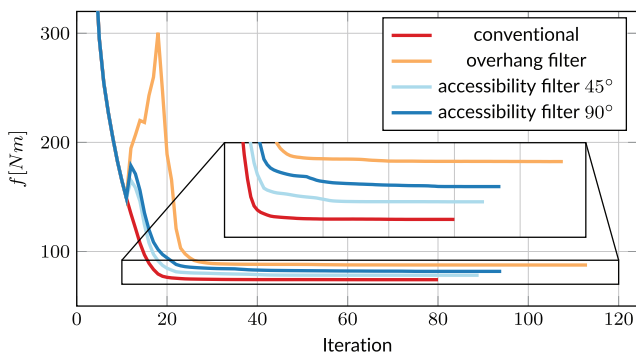
it can be seen that the less constrained optimizations, next to obtaining a lower objective, also converge slightly faster for this 2D example studied.

### 3.1.3 | 2D case: Crane hook

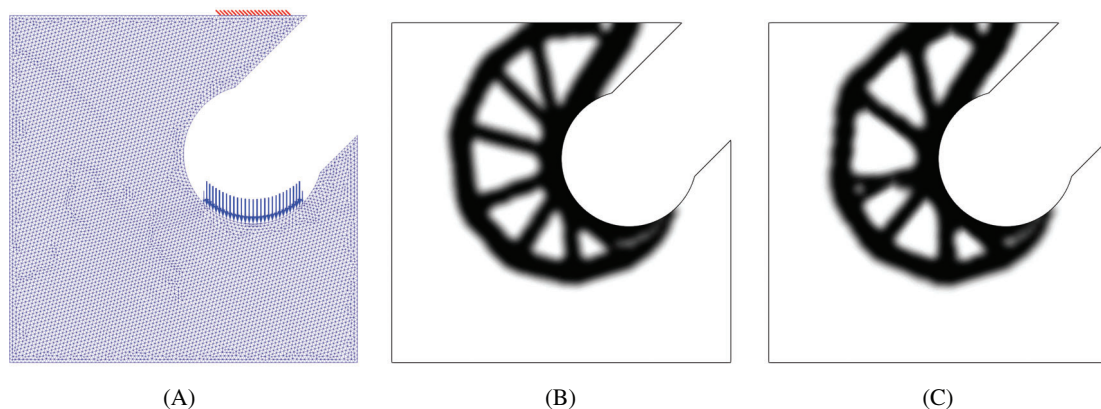
Finally, the accessibility filter is demonstrated on a more practical case shown in Figure 15. For this case, a crane hook design is optimized for stiffness and subject to a 25% volume constraint. To show the flexibility of the method the critical overhang angle is chosen as  $\alpha_{oh} = 30^\circ$  instead of  $45^\circ$ . The domain is mechanically fixed at the top, and a vertical load is applied where the hook is normally loaded (Figure 15(A)). Furthermore, the domain is discretized with unstructured triangular elements. The optimized design without overhang constraint or accessibility filter is shown in Figure 15(B). It can be seen that there is a considerable distance between the hook and the bottom plane. With a conventional overhang filter, the crane structure will connect to the bottom plane to enforce an overhang-free design. The location of the bottom plane is however arbitrary, and should have no influence on the final design. As shown in Figure 15(C), the optimized design with accessibility filter does not need to connect to the bottom plane while still enforcing the overhang constraint inside the cavities to ensure printability. Consequently, the accessibility filtered design has only a 3.0% higher objective compared with the conventional optimized design, while being printable.

## 3.2 | Accessibility filter in 3D

As there is no direct dimension dependency in the accessibility filter, the same filter scheme used in 2D can be directly applied to 3D, provided that a 3D overhang filter is used. Fortunately, many of the published overhang filters have



**FIGURE 14** Convergence plots for the cantilever case. The accessibility and overhang filters are gradually introduced by continuation from iteration 10 to 20, which causes the increase in objective in that interval



**FIGURE 15** The crane hook case. The unstructured mesh and boundary conditions are shown in (A). The shaded red boundary is fixed, while a vertical load is applied at the blue arrows. The conventional optimized design is shown in (B), while the design with accessibility filter is displayed in (C). In (C), the overhang filter is only active in the internal cavities, while the bottom of the crane hook is completely overhanging as it can be supported by easily removable supports

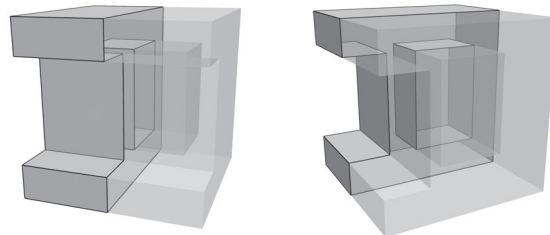
provided a 3D implementation.<sup>18,19,27</sup> For the 3D results, the front propagation based filter presented in Reference 27 is utilized.

The 3D results presented in this section are obtained with an optimization code based on the portable, extensible toolkit for scientific computing (PETSc).<sup>41-43</sup> The domains are discretized using Gmsh,<sup>44</sup> and visualized with ParaView.<sup>45</sup> The displayed geometries are isosurfaces of the density field at  $\rho = 0.5$ . Finally, the PETSc-based MMA implementation presented in References 46,47 has been used as optimization algorithm.

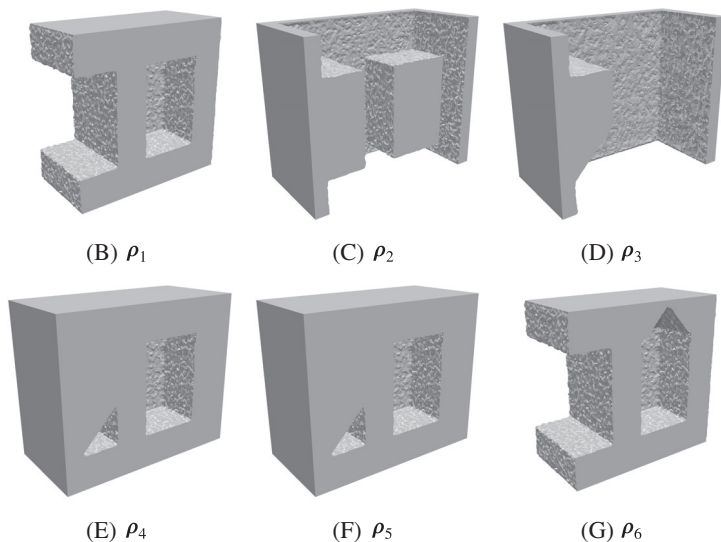
Before topology optimized results are presented, the filter operations of the accessibility filter are verified in 3D. The 3D layout illustrated in Figure 16(A) is reminiscent of the layout in Figure 4. The geometry contains two overhanging regions: one in the internal cavity, and one on the outside. After the accessibility filter is applied, the overhang filter removes the overhang in the internal cavity, as can be seen Figure 16(G). Therefore, the remaining structure is manufacturable, as supports can be removed on the outside, but not in the internal cavity.

### 3.2.1 | 3D case: Torsion box

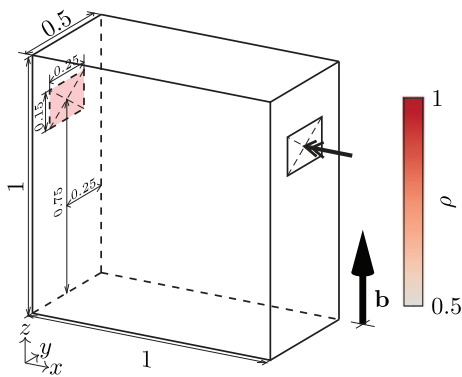
In order to demonstrate the full potential of the accessibility filter, it is applied to a case where the optimal design contains overhanging regions on its outer surface, and in an internal cavity. Closed shell structures, resulting in internal cavities, are often optimal under torsion.<sup>48</sup> Consequently, the test case is defined as shown in Figure 17. A torsional load is applied on a rectangular patch as displayed, and the domain is mechanically fixed on a similar patch at the opposite side. The torsional load is applied by applying a distributed load of 1 N/m on the outer perimeter of the rectangular patch, oriented such that a moment normal to the patch is generated. In order to force overhanging regions on the outer surface, the loading and clamping are placed well above the base plate (i.e., the bottom xy-plane). Furthermore, the volume is constrained at 30% of the design domain. The domain is discretized with an unstructured mesh, containing roughly  $7 \cdot 10^6$  tetrahedral elements and  $1 \cdot 10^6$  nodes. Finally, the density filter that is applied before the accessibility



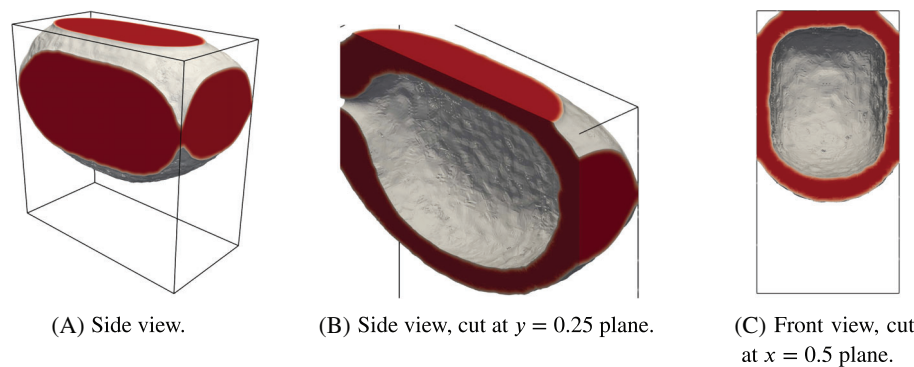
(A) Two views of the geometry on which the accessibility filter is applied. Half of the structure is displayed transparent.



**FIGURE 16** Numerical results of the progression of a geometry (A) through the accessibility filter (B–G). Only half of the geometry is displayed in (B–G)



**FIGURE 17** Domain and boundary conditions for the torsion box case. A moment is applied on the rectangular patch in the front, while a similar patch on the back, shaded red, is fixed. The color scheme as shown is used in all consecutive figures to represent the density field



**FIGURE 18** Optimal design for a conventional stiffness maximization on the torsion box case

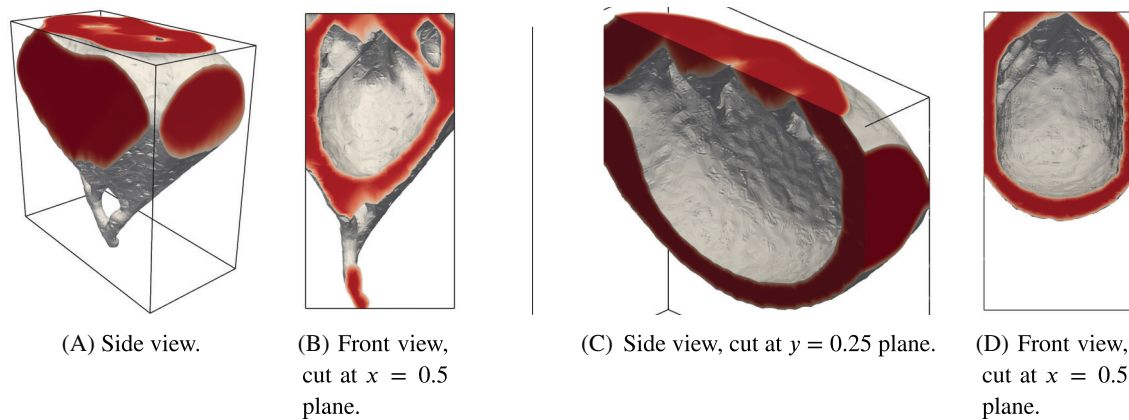
filter has a filter radius of  $2h$ , where  $h = 0.01$  m is the average element edge length. The minimum allowable overhang angle for structure and support is chosen as  $\alpha_{oh} = 45^\circ$ , although since the mesh is unstructured this is an arbitrary choice.

The resulting design for a conventional stiffness optimization, that is, without accessibility or overhang filtering, is displayed in Figure 18. The interior is displayed in Figure 18(B), where the domain is cut along the  $y = 0.25$  plane, and the  $x = 0.5$  plane (Figure 18(C)). As expected, as there is enough material available, a closed shell structured is formed to carry the torsional load. Material is placed as far from the centerline of the load as possible, creating an internal cavity. Supports inside the cavity are thus required in order to additively manufacture the component. However, these supports will be impossible to remove after the build.

Instead, when an overhang filter is included to suppress overhang inside the cavity, the result displayed in Figure 19(A and B) is obtained. As can be seen in the cross-section in Figure 19(B), there is an internal structure in the cavity with permitted degree of overhang, therefore the cavity can be printed without the need for supports. However, the overhang filter is also active on the outside of the structure, requiring the bottom of the component to be printable without supports. This is not realistic, as the bounds of the optimization domain do not need to correspond with the boundaries of the component. Also, certainly with metal AM, the part would not be printed on the base plate directly, but on a layer of supports. This is not well represented by applying the overhang filter on the bottom surface. Furthermore, it can be seen that there are some semidense areas left in Figure 19(B), and holes are present in the top of the structure (Figure 19(A)). Because of the overhang filter, the complete geometry pivots on the bottom supports. If the density in one of the bottom nodes is slightly altered, the complete structure can become overhanging and removed by the overhang filter. This large nonlinearity makes it difficult for the optimization to converge, and causes the semidense regions.

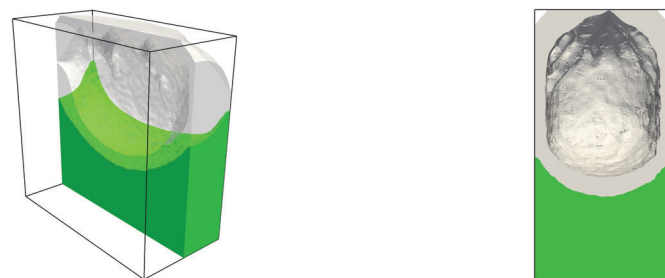
With the accessibility filter applied, a result closer to that of the conventional stiffness optimization is obtained, as can be seen in Figure 19(C and D). The supports that are required to manufacture the bottom of the structure are accessible, and therefore the overhang filter is suppressed. Inside the cavity, the overhang filter is active and a self-supporting internal structure is generated to support the top. Similar to the 2D case, the printable support region is displayed in green in Figure 20. It represents possible locations for support material, in order to support the bottom of the structure.





**FIGURE 19** Optimized design for the torsion box case with overhang filter (A, B), and accessibility filter (C, D), with  $\alpha_{\text{oh}} = 45^\circ$ . With overhang filter, the structure has to connect to the base plate at the bottom to satisfy the overhang constraint, sacrificing performance (A, B). For the accessibility filter (C, D), the structure can be printed on supports, shown in Figure 20, increasing the performance

**FIGURE 20** The printable support field ( $\rho_3$  from the accessibility filter in Figure 9) indicated in green, with the final geometry in gray



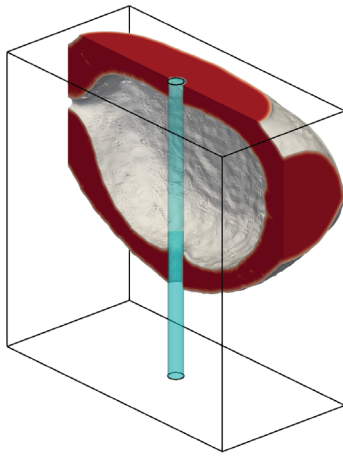
(A) Side view, cut at  $y = 0.25$  plane, with transparent geometry. (B) Front view, cut at  $x = 0.5$  plane.

### 3.2.2 | Torsion box with forced hole

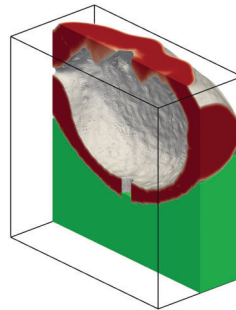
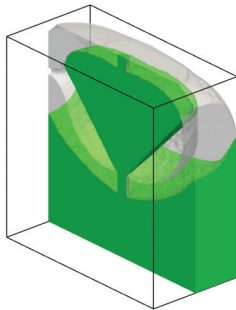
It remains to demonstrate the flexibility of the filter structure presented in Section 2.3 to adjust the accessibility filter for more challenging cases. For this purpose, the torsion case discussed above is slightly altered. The same domain and boundary conditions are used, but for demonstration purposes a cylindrical nondesign region parallel to the  $z$ -axis with a diameter of 0.05 m is introduced, as displayed in Figure 21. Consequently, a small hole in the bottom of the design emerges, through which supports stemming from the base-plate can reach the cavity. Therefore, the design obtained with the accessibility filter as displayed in Figure 21 is similar to a conventional optimization as shown in Figure 18. The support region required to manufacture the design is displayed in green in Figure 22(A). Although the supports are accessible in the sense that they are connected to the base plate, they still are difficult to remove.

To prevent this, the amount of supports inside the cavity can be reduced by increasing the overhang angle for the supports,  $\alpha_{\text{sup}}$ , as was done in the 2D case in Figure 13. However, this would still leave some supports inside the cylindrical nondesign region that are difficult to remove. Instead, we alter the filter to prevent supports from going through small holes. This is achieved by introducing an additional density filter before Step 2 in the accessibility filter, as shown in Figure 23. With the filter structure as presented in Section 2.3 in place, only the input file has to be changed, and the sensitivity calculation is automatically adjusted to the new scheme. The additional density filter smooths the original structure, and closes small holes, up to the filter radius. Note that also a dilate or close filter<sup>49</sup> could have been used, but as the linear density filter already proves effective it is preferred here. A filter radius of  $r = 0.07$  m was used, and the result is displayed in Figure 22(B). This results resembles the structure of the original accessibility filter in Figure 19(C and D), except with a hole in the bottom. A structure inside the cavity supports the top side, and no supports are required that are difficult to remove.



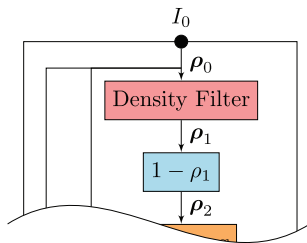


**FIGURE 21** Optimized design for the torsion box case with accessibility filter, with a nondesign region indicated in cyan



**FIGURE 22** With the conventional accessibility filter, supports can access cavities through small holes (A). By including an additional density filter, this can be prevented (B)

(A) Support structure for normal accessibility filter. (B) Layout and supports for modified accessibility filter.

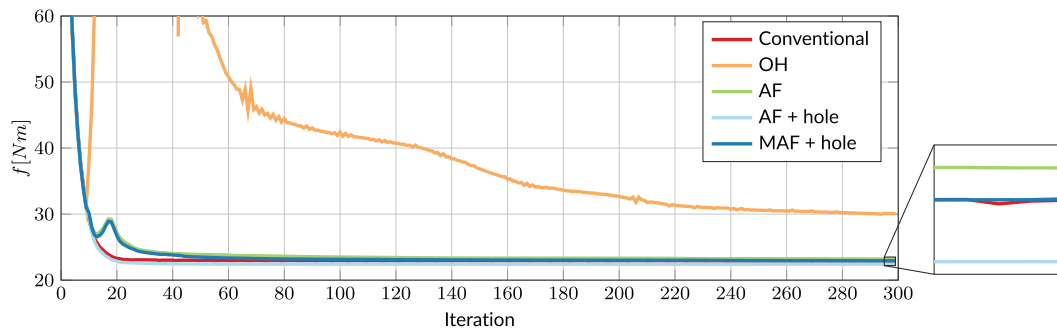


**FIGURE 23** The modified accessibility filter with additional density filter as indicated by the red block. The remainder of the filter is similar to Figure 9

### 3.2.3 | Convergence

The convergence behavior for the different optimizations of the torsion box case is plotted in Figure 24. Similar to the 2D case, the optimizations that include an overhang filter or accessibility filter show an increase in objective from iteration 10 to 20, as the filters are gradually activated with continuation. The exception is the accessibility filter with forced hole (Figure 21), which does not show an increase in objective. Because of the forced hole, the geometry is completely supported, and therefore activating the accessibility filter has no noticeable effect. Furthermore, it can be seen that the optimization with overhang filter has difficulties to converge due to the high degree of nonlinearity as discussed. This might be prevented by choosing a different optimization algorithm such as globally convergent MMA,<sup>50</sup> which adjusts the step size according to the degree of nonlinearity.

The final objective values of the conventional, overhang filtered, and accessibility filtered optimization are 22.9, 30.1, and 23.2 N m, respectively. This is in accordance to the amount of material in the final design that has to be used to make the structure printable, instead of improving the performance. That is none for the conventional case, the most for the overhang filtered case, and a small amount for the accessibility filtered case. The same holds for the optimizations with



**FIGURE 24** Convergence plots for the torsion box case for the conventional, overhang filter (OH), accessibility filter (AF), accessibility filter with force hole (AF + hole), and the modified accessibility filter with forced hole (MAF + hole)

forced hole, which reach a final objective of 22.4 and 22.9 N m for the accessibility filter and modified accessibility filter, respectively.

Surprisingly, the accessibility filter with hole has a somewhat lower objective than the conventional optimization. A contribution to the lower objective can be the fact that the two overhang filters present in the accessibility filter slightly modify the length scale, allowing sharper edges.<sup>21</sup> Consequently, there is less semidense material for the accessibility filtered case. This can lead to a lower objective as semidense material has an unfavorable stiffness–mass ratio due to the penalization. Besides, it is also possible that the conventional optimization is stuck in a local minimum with a higher objective value compared with the accessibility-filtered optimizations.

## 4 | DISCUSSION

Both the 2D and the 3D results demonstrate the effectiveness of the accessibility filter in relaxing a strict minimum overhang angle constraint to a more realistic accessibility of support requirement. An overhang filter requires the structure to connect to some predefined base-plate, which can have a large impact on performance. Although slightly exaggerated in the 3D case by placing the boundary conditions well above the base plate, a significant increase of performance is achieved when switching from overhang to accessibility filter.

Because the accessibility filter is composed of many individual filters, a filter structure was proposed that automates the filtering and sensitivity evaluation of any filter layout. This structure is much more widely applicable than just considering the accessibility filter case, and has already proved valuable for rapid prototyping of different filter configurations.

The numerical examples show that the relative impact on part performance of the accessibility filter is small, 6% for the 2D case and 1% for the 3D case, compared with 18% and 31% for the strict overhang filter. This is of course case specific, but the performance decrease is expected to be certainly lower than strict overhang constraints, as only inaccessible areas are affected. Furthermore, by inspecting different fields of the accessibility filter, one can get a good indication where supports are required. An optimization of the support structures inside the printable support domains identified by the accessibility filter could be complementary to the accessibility filter to generate an accurate support layout.

The accessibility of supports is currently assessed based on whether a support is connected to the base plate or not. With this definition of accessibility, it is possible to have a result with supports that are in practice difficult to remove, as shown by the last example in Figure 22(A). However, this case has specifically been fabricated to showcase this possibility. In practice, it is unlikely that a hole is created in the wall of an enclosed region, as having a closed wall is a local optimum. The optimization algorithm can only exploit this configuration after a hole is already present. Furthermore, it was shown that this behavior can be avoided with a slight modification of the accessibility filter. However, it would be interesting to improve on this by incorporating a simulation of an actual tool removing the support, for example, a type of milling constraint. This will result in a more accurate measure of accessibility.

On the other hand, there are printing process for which support removal is less of a problem, for example, by using soluble support material. For these processes, solutions such as the one shown in Figure 22(A) are acceptable. This requires a small initial hole in the shell of the enclosed void which might be triggered by including a constraint to prevent enclosed

voids,<sup>36</sup> or possibly the filter scheme can be adopted to detect internal cavities with front propagation. These constraints can also be used when cavities have to be opened to, for example, remove powder after printing.

Finally, in this study the accessibility filter is only demonstrated on compliance minimization problems. However, there is no fundamental limitation to the applicability of the accessibility filter to any problem that can be optimized using a density-based approach such as flow optimization or stress-minimization.

## 5 | CONCLUSION

In this work an accessibility filter for topology optimization is presented, and successfully demonstrated. The filter provides an optimized structure which is printable without support structures in regions where they are difficult to remove. In contrast to conventional overhang filters/constraints, which result in a structure that is printable without any support, supports are allowed if they are connected to the base plate. This resembles powder bed AM methods, where a part is never directly built onto the base plate. The accessibility filter is a composition of a number of regular filters, which is implemented in a novel modular filter structure. This arrangement makes it relatively easy to implement if an overhang filter is available. Future directions include to build on this idea and further refine the definition of support structures and their removability as required.

## ACKNOWLEDGMENT


This research was carried out under project number S12.7.14549a in the framework of the Metals for Additive Manufacturing Partnership Program of the Materials innovation institute M2i ([www.m2i.nl](http://www.m2i.nl)) and the Netherlands Organization for Scientific Research ([www.nwo.nl](http://www.nwo.nl)).

## DATA AVAILABLE STATEMENT

Data available on request from the authors

## ORCID

Emiel van de Ven  <https://orcid.org/0000-0001-6252-8263>

Matthijs Langelaar  <https://orcid.org/0000-0003-2106-2246>

Fred van Keulen  <https://orcid.org/0000-0003-2634-0110>

## REFERENCES

1. Frazier WE. Metal additive manufacturing: a review. *J Mater Eng Perform*. 2014;23(6):1917-1928. <https://doi.org/10.1007/s11665-014-0958-z>.
2. Hague R, Mansour S, Saleh N. Design Opportunities with Rapid Manufacturing. *Assem Autom*. 2003;23(4):346-356. <https://doi.org/10.1108/01445150310698643>.
3. Bendsoe MP, Sigmund O. *Topology Optimization*. Berlin/Heidelberg, Germany: Springer; 2004.
4. Christensen P, Klarbring A. *An Introduction to Structural Optimization Solid Mechanics and Its Applications*. Vol 153. Amsterdam, Netherlands: Springer; 2008.
5. Yasa E, Craeghs T, Badrossamay M, Kruth JP. Rapid Manufacturing Research at the Catholic University of Leuven. *RapidTech*; 2009.
6. Wang D, Yang Y, Yi Z, Su X. Research on the fabricating quality optimization of the overhanging surface in SLM process. *Int J Adv Manuf Technol*. 2013;65(9-12):1471-1484. <https://doi.org/10.1007/s00170-012-4271-4>.
7. Cloots M, Zumofen L, Spierings AB, Kirchheim A, Wegener K. Approaches to Minimize Overhang Angles of SLM Parts. *Rapid Prototyp J*. 2017;23(2):362-369. <https://doi.org/10.1108/RPJ-05-2015-0061>.
8. Thomas D. The Development of Design Rules for Selective Laser Melting [PhD thesis]. University of WalesCardiff; 2009.
9. Adam GAO, Zimmer D. Design for additive manufacturing&element transitions and aggregated structures. *CIRP J Manuf Sci Technol*. 2014;7(1):20-28. <https://doi.org/10.1016/j.cirpj.2013.10.001>.
10. Kranz J, Herzog D, Emmelmann C. Design guidelines for laser additive manufacturing of lightweight structures in TiAl6V4. *J Laser Appl*. 2015;27(S1):S14001. <https://doi.org/10.2351/1.4885235>.
11. Thompson MK, Moroni G, Vaneker T, et al. Design for Additive Manufacturing: Trends, opportunities, considerations, and constraints. *CIRP Ann*. 2016;65(2):737-760. <https://doi.org/10.1016/j.cirp.2016.05.004>.
12. Liu J, Gaynor AT, Chen S, et al. Current and future trends in topology optimization for additive manufacturing. *Struct Multidiscip Optim*. 2018;57(6):2457-2483. <https://doi.org/10.1007/s00158-018-1994-3>.

13. Allaire G, Dapogny C, Estevez R, Faure A, Michailidis G. Structural optimization under overhang constraints imposed by additive manufacturing technologies. *J Comput Phys*. 2017;351:295-328. <https://doi.org/10.1016/j.jcp.2017.09.041>.
14. Guo X, Zhou J, Zhang W, Du Z, Liu C, Liu Y. Self-supporting structure design in additive manufacturing through explicit topology optimization. *Comput Methods Appl Mech Eng*. 2017;323:27-63. <https://doi.org/10.1016/j.cma.2017.05.003>.
15. Qian X. Undercut and overhang angle control in topology optimization: a density gradient based integral approach. *Int J Numer Methods Eng*. 2017;111(3):247-272. <https://doi.org/10.1002/nme.5461>.
16. Mirzendehtdel AM, Suresh K. Support structure constrained topology optimization for additive manufacturing. *Comput Aided Des*. 2016;81:1-13. <https://doi.org/10.1016/j.cad.2016.08.006>.
17. Gaynor AT, Guest JK. Topology optimization considering overhang constraints: eliminating sacrificial support material in additive manufacturing through design. *Struct Multidiscip Optim*. 2016;54(5):1157-1172. <https://doi.org/10.1007/s00158-016-1551-x>.
18. Johnson TE, Gaynor AT. Three-dimensional projection-based topology optimization for prescribed-angle self-supporting additively manufactured structures. *Addit Manuf*. 2018;24:667-686. <https://doi.org/10.1016/j.addma.2018.06.011>.
19. Langelaar M. Topology optimization of 3D self-supporting structures for additive manufacturing. *Addit Manuf*. 2016;12:60-70. <https://doi.org/10.1016/j.addma.2016.06.010>.
20. Langelaar M. An additive manufacturing filter for topology optimization of print-ready designs. *Struct Multidiscip Optim*. 2017;55(3):871-883. <https://doi.org/10.1007/s00158-016-1522-2>.
21. van de Ven E, Maas R, Ayas C, Langelaar M, van Keulen F. Continuous front propagation-based overhang control for topology optimization with additive manufacturing. *Struct Multidiscip Optim*. 2018;57(5):2075-2091. <https://doi.org/10.1007/s00158-017-1880-4>.
22. Allaire G, Dapogny C, Faure A, Michailidis G. Shape optimization of a layer by layer mechanical constraint for additive manufacturing. *Comptes Rendus Mathematique*. 2017;355(6):699-717. <https://doi.org/10.1016/j.crma.2017.04.008>.
23. Amir O, Mass Y. Topology optimization for staged construction. *Struct Multidiscip Optim*. 2018;57(4):1679-1694. <https://doi.org/10.1007/s00158-017-1837-7>.
24. Ranjan R, Ayas C, Langelaar M, Keulen A. Towards design for precision additive manufacturing: a simplified approach for detecting heat accumulation. Paper presented at: Proceedings of the ASPE and EUSPEN Summer Topical Meeting, ASPE; 2018. Berkeley, CA: Lawrence Berkeley National Laboratory.
25. Langelaar M. Topology optimization for additive manufacturing with controllable support structure costs. Paper presented at: Proceedings of the 7th European Congress on Computational Methods in Applied Sciences and Engineering. National Technical University of Athens; 2016; NTUA, Greece.
26. Langelaar M. Integrated component-support topology optimization for additive manufacturing with post-machining. *Rapid Prototyp J*. 2019;25(2):255-265. <https://doi.org/10.1108/rpj-12-2017-0246>.
27. van de Ven E, Maas R, Ayas C, Langelaar M, van Keulen F. Overhang control based on front propagation in 3D topology optimization for additive manufacturing. *Comput Methods Appl Mech Eng*. 2020;369:113169. <https://doi.org/10.1016/j.cma.2020.113169>.
28. Eschenauer HA, Olhoff N. Topology optimization of continuum structures: a review. *Appl Mech Rev*. 2001;54(4):331. <https://doi.org/10.1115/1.1388075>.
29. Sigmund O, Maute K. Topology optimization approaches: a comparative review. *Struct Multidiscip Optim*. 2013;48(6):1031-1055. <https://doi.org/10.1007/s00158-013-0978-6>.
30. Deaton JD, Grandhi RV. A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Struct Multidiscip Optim*. 2014;49(1):1-38. <https://doi.org/10.1007/s00158-013-0956-z>.
31. Bendsoe MP. Optimal shape design as a material distribution problem. *Struct Optim*. 1989;1(4):193-202. <https://doi.org/10.1007/BF01650949>.
32. Sigmund O, Petersson J. Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Struct Optim*. 1998;16(1):68-75. <https://doi.org/10.1007/BF01214002>.
33. Bruns TE, Tortorelli DA. Topology optimization of non-linear elastic structures and compliant mechanisms. *Comput Methods Appl Mech Eng*. 2001;190(26-27):3443-3459. [https://doi.org/10.1016/S0045-7825\(00\)00278-4](https://doi.org/10.1016/S0045-7825(00)00278-4).
34. Rozvany GIN, Zhou M, Birker T. Generalized shape optimization without homogenization. *Struct Optim*. 1992;4(3-4):250-252. <https://doi.org/10.1007/BF01742754>.
35. Gaynor AT, Johnson TE. Eliminating occluded voids in additive manufacturing design via a projection-based topology optimization scheme. *Addit Manuf*. 2020;101149. <https://doi.org/10.1016/j.addma.2020.101149>.
36. Li Q, Chen W, Liu S, Tong L. Structural topology optimization considering connectivity constraint. *Struct Multidiscip Optim*. 2016;54(4):971-984. <https://doi.org/10.1007/s00158-016-1459-5>.
37. Bray T, Paoli J, Sperberg-McQueen CM, Mailer E, Yergeau F. Extensible Markup Language (XML) 1.0 5th Edition. W3C recommendation; 2008.
38. The JSON Data Interchange Syntax. Standard ECMA-404: Ecma International. Geneva, CH; 2017.
39. Kahn AB. Topological sorting of large networks. *Commun ACM*. 1962;5(11):558-562. <https://doi.org/10.1145/368996.369025>.
40. Svanberg K. the method of moving asymptotes—a new method for structural optimization. *Int J Numer Methods Eng*. 1987;24(2):359-373. <https://doi.org/10.1002/nme.1620240207>.
41. Balay S, Abhyankar S, Adams MF, et al. PETSc Users Manual. ANL-95/11 - Revision 3.8: Argonne National Laboratory; 2017.
42. Balay S, Gropp WD, McInnes LC, Smith BF. *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*. Boston, MA: Birkhäuser; 1997:163-202.

43. Karypis G, Kumar V. A parallel algorithm for multilevel graph partitioning and sparse matrix ordering. *J Parall Distrib Comput.* 1998;48:71-85.
44. Geuzaine C, Remacle J-F. Gmsh: a 3-D finite element mesh generator with built-in pre-and post-processing facilities. *Int J Numer Methods Eng.* 2009;79(11):1309-1331.
45. Ahrens J, Geveci B, Law C. ParaView: an end-user tool for large-data visualization. In: Hansen CD, Johnson CR, eds. *Visualization Handbook.* Burlington: Butterworth-Heinemann; 2005:717-731.
46. Aage N, Lazarov BS. Parallel framework for topology optimization using the method of moving asymptotes. *Struct Multidiscip Optim.* 2013;47(4):493-505.
47. Aage N, Andreassen E, Lazarov BS. Topology optimization using PETSc: an easy-to-use, fully parallel, open source topology optimization framework. *Struct Multidiscip Optim.* 2015;51(3):565-572.
48. Sigmund O, Aage N, Andreassen E. On the (Non-)optimality of michell structures. *Struct Multidiscip Optim.* 2016;54(2):361-373. <https://doi.org/10.1007/s00158-016-1420-7>.
49. Sigmund O. Morphology-based black and white filters for topology optimization. *Struct Multidiscip Optim.* 2007;33(4-5):401-424. <https://doi.org/10.1007/s00158-006-0087-x>.
50. Svanberg K. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM J Optim.* 2002;12(2):555-573. <https://doi.org/10.1137/S1052623499362822>.

**How to cite this article:** van de Ven E, Ayas C, Langelaar M, Maas R, van Keulen F. Accessibility of support structures in topology optimization for additive manufacturing. *Int J Numer Methods Eng.* 2021;122:2038–2056. <https://doi.org/10.1002/nme.6611>