Technische Universität Graz

A multi-material topology optimization algorithm based on the topological derivative

P. Gangl

Berichte aus dem Institut für Angewandte Mathematik

Bericht 2019/9

Technische Universität Graz

A multi-material topology optimization algorithm based on the topological derivative

P. Gangl

Berichte aus dem
Institut für Angewandte Mathematik

Bericht 2019/9

Technische Universität Graz
Institut für Angewandte Mathematik
Steyrergasse 30
A 8010 Graz

WWW: `http://www.applied.math.tugraz.at`

A multi-material topology optimization algorithm based on the topological derivative

P. Gangl[*]

November 21, 2019

Abstract

We present a level-set based topology optimization algorithm for design optimization problems involving an arbitrary number of different materials, where the evolution of a design is solely guided by topological derivatives. Our method can be seen as an extension of the algorithm that was introduced in [7] for two materials to the case of an arbitrary number M of materials. We represent a design that consists of multiple materials by means of a vector-valued level set function which maps into $\mathbb{R}^{M-1}$. We divide the space $\mathbb{R}^{M-1}$ into M sectors, each corresponding to one material, and establish conditions for local optimality of a design based on certain generalized topological derivatives. The optimization algorithm consists in a fixed point iteration striving to reach this optimality condition. Like the two-material version of the algorithm, also our method possesses a nucleation mechanism such that it is not necessary to start with a perforated initial design. We show numerical results obtained by applying the algorithm to an academic example as well as to the compliance minimization in linearized elasticity.

1 Introduction

Over the past decades, numerical shape and topology optimization techniques have become an integral part in the design process not only of mechanical structures, but also in applications from electromagnetics, fluid mechanics and many more. While classical shape optimization approaches [13] can only alter boundaries or material interfaces of a given design, topology optimization approaches are more general and can yield optimal designs of any topology. There are several classes of methods for optimizing the topology of a design. A thorough overview over the different approaches to topology optimization is given in the review article [26].

The idea of density-based topology optimization [8] is to represent a design by means of a density variable ρ, which can attain any value between 0 and 1, where regions with $\rho(x) = 1$ are interpreted as one material and regions with $\rho(x) = 0$ as the other material (e.g. an isotropic material and void). In order to avoid large areas of intermediate density values $0 < \rho(x) < 1$, penalization is performed in the constitutive equation, which – in combination with a constraint on the allowed volume – makes "black" ($\rho(x) = 1$) or "white" ($\rho(x) = 0$) regions more favorable and therefore removes the "gray" ($0 < \rho(x) < 1$) areas. In order to avoid numerical instabilities, which occur as a consequence of the nonexistence of solutions to the topology optimization problem, regularization techniques such as density filtering, sensitivity filtering or bounding the perimeter of the structure are used [25]. Density-based topology optimization approaches are among the most widely used topology optimization approaches and have been successfully applied to a large number of practical applications, see e.g. [1].

[*] Institute of Applied Mathematics, Graz University of Technology, Steyrergasse 30, 8010 Graz, Austria. E-mail: gangl@math.tugraz.at

In contrast to density-based methods, level set approaches [24] do not introduce intermediate material properties. Here, the domain of interest Ω is represented by means of a continuous function ψ which attains negative values inside Ω and positive values outside Ω. Thus, the boundary of Ω is given as the zero level set of ψ, i.e. $\partial\Omega = \{x : \psi(x) = 0\}$. In the level set method for shape and topology optimization [4], the evolution of the level set function ψ is guided by shape gradients by means of a Hamilton-Jacobi equation. The implicit geometry description by means of a level set function brings a lot of flexibility in the treatment of topological changes. However, since no topological sensitivity information is included, the method lacks a nucleation mechanism. Holes or different components can merge, but no holes can be created in the interior of the design. This problem is typically circumvented by choosing a perforated initial design with many circular holes. Also this approach has proven very useful in many practical applications, see e.g. [15].

As an alternative to choosing a perforated initial design in the level set method, a coupling of the Hamilton-Jacobi equation, which uses shape sensitivities, with the *topological derivative* was proposed in [3, 10]. The topological derivative [23] at a point z indicates whether a change of material at that point will yield an increase or decrease of the objective function.

As opposed to the level set method proposed in [4], the algorithm introduced in [6, 7], which also uses a level set representation of the domain, exploits topological sensitivity information. Here, the evolution of the level set function is guided by the topological derivative. We will give a thorough introduction to this method in Section 2. A related topology optimization approach is the one introduced in [31], where instead of the topological derivative the sensitivity of the objective with respect to a variation of the level set function is used.

Most applications of and methods for topology optimization deal with finding the optimal distribution of two different phases (e.g. one isotropic material and void, or two different materials) within a given design area. However, most of the approaches mentioned above have been extended to the multi-material case where one is interested in distributing a certain number of different materials within the design region in an optimal way. Multi-material topology optimization in a density-based setting was investigated in [20]. The level set method based on shape sensitivities and a Hamilton-Jacobi equation was generalized to the multi-material case in [2, 29, 30] and similar techniques have also been used in image segmentation [28]. Multi-material topology optimization has also been used in a phase field setting [9] and from an optimal control point of view [11]. Moreover, the approach introduced in [31] has been extended to the multi-material case, e.g., in [21].

In this work, we propose an alternative way of multi-material topology optimization which is based solely on topological derivatives. Our algorithm can be seen as a generalization of the work presented in [6, 7] to the case of multiple materials. We use a description of the design by a vector-valued level set function. The goal of the method is to reach a local optimality condition which is expressed by means of topological derivatives. By construction, our algorithm is capable of altering shape and topology of an initial design. Holes can be nucleated and new components can be created. Therefore, there is no need to start with perforated initial designs.

The rest of this paper is organized as follows: In Section 2 we will revisit the algorithm introduced in [7] and point out its main ingredients. In Section 3 we will first introduce a level set framework for multiple materials and then generalize the algorithm introduced in [7] to the case of an arbitrary number of materials. Finally, we present numerical examples in Section 4.

2 Two-material topology optimization using topological derivatives

In this section, we revisit the algorithm introduced in [7] and recall its main ingredients. The algorithm uses a level set description of the design and the evolution of the design is guided solely by the topological derivative, allowing for nucleation of holes or creation of new components anywhere in the design domain.

2.1 Topological derivative

The topological derivative of a domain-dependent shape function represents its sensitivity with respect to a topological perturbation of the domain. The idea of the topological derivative was first used in [14] as the "bubble method" and was introduced in a mathematically rigorous way for the first time in [27], see also [22, 23] for an overview on the topic.

In the following, let $d \in \{1,2,3\}$ denote the space dimension and let an open hold-all domain $D \subset \mathbb{R}^d$ be given. Let $\mathscr{P}(D)$ denote the power set of D, i.e., the set of all subsets of D, and let $\mathscr{A} \subset \mathscr{P}(D)$ denote a set of admissible subsets of D, the definition of which may depend on the problem at hand. Let a shape function $\mathscr{J}$

$$\begin{aligned} \mathscr{J} : \mathscr{A} &\to \mathbb{R}, \\ \Omega &\mapsto \mathscr{J}(\Omega), \end{aligned} \tag{2.1}$$

be given. Let us further fix an open set $\Omega_1 \in \mathscr{A}$ and define $\Omega_2 := D \setminus \overline{\Omega}_1$. We define the topological derivative of the shape function $\mathscr{J}$ at a point $z \in \Omega_1$ with respect to a change of the material in a neighborhood ω_ε of z, see Fig. 1 (left).

Definition 2.1. Let $\mathscr{J} : \mathscr{A} \to \mathbb{R}$ a shape function and $\Omega_1 \in \mathscr{A}$ an open admissible set. Let further $\omega \subset \mathbb{R}^d$ open with $0 \in \omega$. Let $z \in \Omega_1$ and, for $\varepsilon > 0$ small, let $\omega_\varepsilon := z + \varepsilon\omega$. Then, the topological derivative of $\mathscr{J}$ at the point $z \in \Omega_1$ is defined as the limit

$$\mathscr{T}^{1\to 2}(z) := \lim_{\varepsilon \searrow 0} \frac{\mathscr{J}(\Omega_1 \setminus \overline{\omega}_\varepsilon) - \mathscr{J}(\Omega_1)}{\ell(\varepsilon)}$$

where $\ell : \mathbb{R}^+ \to \mathbb{R}^+$ is a continuous positive function satisfying $\ell(\varepsilon) \to 0$ as $\varepsilon \to 0$.

Here, the set ω determines the shape of the small inclusion ω_ε. The most common choice is $\omega = B(0,1)$ to deal with ball-shaped inclusions, but also other shapes such as ellipses are possible. The function $\ell(\varepsilon)$ has to be chosen depending on the concrete application at hand. For most applications one has to choose $\ell(\varepsilon) = |\omega_\varepsilon| = \varepsilon^d |\omega|$, however certain applications require different choices of $\ell(\varepsilon)$ [23].

Similarly to Definition 2.1, by interchanging the roles of Ω_1 and Ω_2, the topological derivative can also be defined for a point $\tilde{z} \in \Omega_2$ with the corresponding inclusion $\tilde{\omega}_\varepsilon := \tilde{z} + \varepsilon\omega$. Then, the topological derivative reads

$$\mathscr{T}^{2\to 1}(\tilde{z}) := \lim_{\varepsilon \searrow 0} \frac{\mathscr{J}(\Omega_1 \cup \tilde{\omega}_\varepsilon) - \mathscr{J}(\Omega_1)}{\ell(\varepsilon)}.$$

Remark 2.2. The mathematically rigoros derivation of topological derivatives for shape functions that depend on the shape via the solution to a partial differential equation is a research field on its own. There exist different approaches for the derivation of topological derivatives, see e.g. [5, 16, 22, 23]. Since the current work is concerned with an optimization algorithm based on topological derivatives, we assume that closed formulas for the topological derivatives are available for the sake of this paper.

Throughout this work, we focus on minimization problems and remark that maximization problems can be treated analogously after a simple modification. Based on these definitions, we obtain the following notion of local optimality with respect to topological changes.

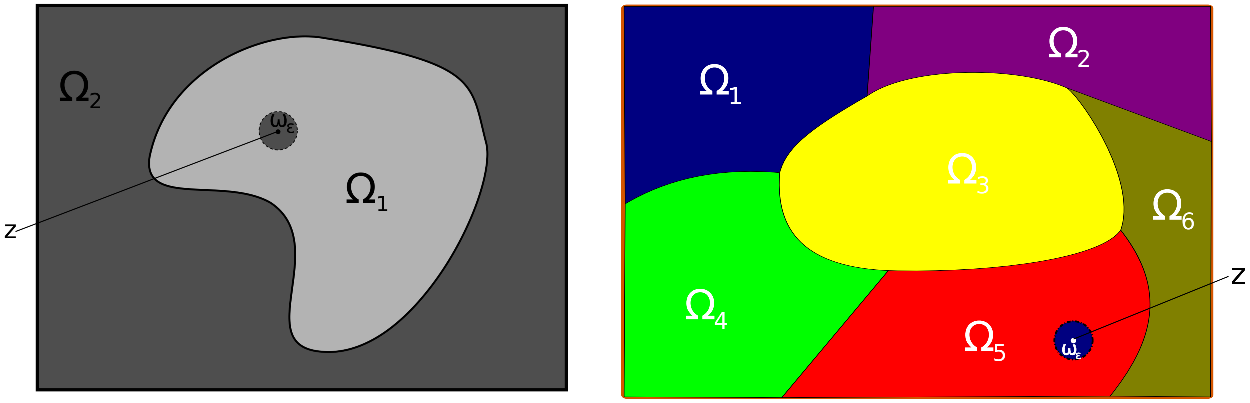


Figure 1: Illustration of topological derivative $\mathcal{T}^{1\to2}$ in case of two materials (left) and of topological derivative $\mathcal{T}^{5\to1}$ in case of multiple (here: six) materials (right).

Definition 2.3. Given a shape function $\mathcal{J}$, an open domain $\Omega_1 \in \mathcal{A}$ is locally optimal for the minimization of $\mathcal{J}$ with respect to topological changes if

$$\mathcal{T}(x) := \chi_{\Omega_1}(x)\mathcal{T}^{1\to2}(x) + \chi_{\Omega_2}(x)\mathcal{T}^{2\to1}(x) > 0$$

for all $x \in \Omega_1 \cup \Omega_2$.

Note that topological derivatives and therefore also $\mathcal{T}$ are only defined in the interior of subdomains and not on boundaries or material interfaces.

2.2 Level set algorithm solely based on topological derivative

In a level set framework, a domain Ω_1 is represented by means of a function $\psi : D \to \mathbb{R}$ such that

$$\begin{cases} \psi(x) < 0 \Longleftrightarrow x \in \Omega_1, \\ \psi(x) = 0 \Longleftrightarrow x \in \Gamma, \\ \psi(x) > 0 \Longleftrightarrow x \in D \setminus \overline{\Omega_1} =: \Omega_2. \end{cases} \tag{2.2}$$

Here, $\Gamma = \partial\Omega_1 \cap D$ includes the material interface between Ω_1 and Ω_2 and excludes the boundary of the hold-all domain ∂D. In the algorithm introduced in [7], the evolution of the level set function ψ is guided by the *generalized topological derivative*, which is defined as

$$G(x) := \begin{cases} -\mathcal{T}^{1\to2}(x), & x \in \Omega_1, \\ \mathcal{T}^{2\to1}(x), & x \in \Omega_2, \end{cases} \tag{2.3}$$

for a fixed domain Ω_1 and its complement $\Omega_2 = D \setminus \overline{\Omega_1},$. When the domains Ω_1 and Ω_2 are represented by the level set function ψ via (2.2), we will indicate the dependence of the generalized topological derivative G and of the topological derivatives $\mathcal{T}^{i\to j}$ on the design by the additional subscript ψ, i.e., we will write G_ψ and $\mathcal{T}_\psi^{i\to j}$.

In this setting, we can state a sufficient condition for a domain Ω_1 to satisfy the local optimality condition of Definition 2.3:

Lemma 2.4. Let $\mathscr{J} : \mathscr{A} \to \mathbb{R}$ a shape function and $\Omega_1 \in \mathscr{A}$ open. Let $\psi : D \to \mathbb{R}$ the level set function representing Ω_1 according to (2.2). Assume that Ω_1 is such that there exists $c > 0$ such that

$$\psi(x) = c\, G_\psi(x) \tag{2.4}$$

for all $x \in D \setminus \partial\Omega_1$. Then Ω_1 is locally optimal with respect to topological changes according to Definition 2.3.

Proof. Let x in Ω_1 arbitrary, but fixed. Due to (2.2), condition (2.4) and the definition in (2.3), we have

$$0 > \psi(x) = c\, G_\psi(x) = -c\, \mathscr{T}_\psi^{1\to 2}(x),$$

for some $c > 0$. Thus we have that $\mathscr{T}_\psi^{1\to 2}(x) > 0$. Similarly, we get for an arbitrary, but fixed point $\tilde{x} \in \Omega_2$ that

$$0 < \psi(\tilde{x}) = c\, G_\psi(\tilde{x}) = c\, \mathscr{T}_\psi^{2\to 1}(\tilde{x}),$$

and thus $\mathscr{T}_\psi^{2\to 1}(\tilde{x}) > 0$, which finishes the proof. □

The idea of the algorithm introduced in [7] is to start with an initial design $\Omega_1^{(0)}$ represented by a level set function ψ_0 and to reach a setting where condition (2.4) holds by means of a fixed point iteration. For numerical stability reasons, this fixed point iteration is performed on the unit sphere $\mathscr{S}$ of the Hilbert space $L^2(D)$, $\mathscr{S} = \{v \in L^2(D) : \|v\|_{L^2(D)} = 1\}$.

The algorithm reads as follows:

Algorithm 1. *Choose initial design $\psi_0 \in \mathscr{S}$ and $\varepsilon_\theta > 0$ (e.g. $\varepsilon_\theta = 0.5°$).*
For $k = 0, 1, 2, \dots$

1. *Compute G_{ψ_k} according to* (2.3).
2. *Compute $\theta_k := arccos\left[\left(\psi_k, \frac{G_{\psi_k}}{\|G_{\psi_k}\|_{L^2(D)}}\right)_{L^2(D)}\right]$.*
3. *If $\theta_k < \varepsilon_\theta$ then stop,*
   else set
$$\psi_{k+1} = \frac{1}{sin\, \theta_k}\left[sin((1-\kappa_k)\theta_k)\psi_k + sin(\kappa_k\theta_k)\frac{G_{\psi_k}}{\|G_{\psi_k}\|_{L^2(D)}}\right], \tag{2.5}$$
   where $\kappa_k = max\{1, \frac{1}{2}, \frac{1}{4}, \dots\}$ such that $\mathscr{J}\left(\Omega_1^{(k+1)}\right) < \mathscr{J}\left(\Omega_1^{(k)}\right)$.

In step 2 of Algorithm 1, the angle θ_k in an $L^2(D)$-sense between the current level set function ψ_k and a scaled version of the generalized topological derivative G_{ψ_k} is computed. Note that if $\theta_k = 0$ then condition (2.4) is satisfied and the corresponding domain $\Omega_1^{(k)}$ is locally optimal according to Definition 2.3. Also note that, by construction, we get that $\psi_k \in \mathscr{S}$ for all k. The sets $\Omega_1^{(k+1)}$ and $\Omega_1^{(k)}$ denote the shapes represented by the corresponding level set functions ψ_{k+1} and ψ_k, respectively. For more details on the algorithm, we refer the reader to [6, 7].

As it is emphasized in [7, Sec. 3.3], the algorithm is evolving along a descent direction, which is an important ingredient for it to be successful. This means that a local change of material around a point $\hat{x} \in \Omega_1 \cup \Omega_2$ from iteration k of the algorithm to the next iteration $k+1$ can only happen if the corresponding topological derivative at iteration k was negative. This observation can be summarized as follows:

Lemma 2.5. Let ψ_k and ψ_{k+1} be two subsequent iterates obtained by Algorithm 1. Then we have for $\hat{x} \in D$ that

(i) $\psi_k(\hat{x}) < 0 < \psi_{k+1}(\hat{x}) \quad \Longrightarrow \quad \mathcal{T}^{1\to 2}_{\psi_k}(\hat{x}) < 0,$

(ii) $\psi_k(\hat{x}) > 0 > \psi_{k+1}(\hat{x}) \quad \Longrightarrow \quad \mathcal{T}^{2\to 1}_{\psi_k}(\hat{x}) < 0.$

Proof. Since $\sin(\theta) > 0$ and $\sin(s\,\theta) > 0$ for all $\theta \in (0,\pi)$ and all $s \in (0,1)$, we conclude from $\psi_k(\hat{x}) < 0 < \psi_{k+1}(\hat{x})$ and (2.5) that $G_{\psi_k}(\hat{x}) > 0$ and thus, since $\hat{x} \in \Omega_1^{(k)}$, it follows that $\mathcal{T}^{1\to 2}_{\psi_k}(\hat{x}) < 0$. The second statement follows analogously. □

We will later establish a similar result in the case of multiple materials in Section 3.

We make the following important remark concerning the numerical realization of Algorithm 1 in the PDE-constrained case, i.e., in the case where the objective function $\mathcal{J}$ depends on the shape Ω via the solution u of a constraining boundary value problem. We restrict our discussion to the case of problems posed in an $H^1(D)$ setting such as linearized elasticity or Laplace-type problems. Problems where the solution u is not continuous across element boundaries like problems posed in $H(curl, D)$ deserve a separate investigation. The following aspects are discussed in [7] for the case of linearized elasticity:

Remark 2.6. We choose a finite-dimensional space V_h of $L^2(D)$ endowed with the $L^2(D)$ inner product to represent the design variable ψ. To guarantee the smoothness of the design, the level set function ψ should be continuous across the interface $\partial\Omega_1$. Therefore, we choose V_h to be the space of globally continuous and piecewise linear functions on a given triangulation of D. As proposed in [7], we will use the same mesh and the same finite elements for representing the design and for solving the PDE constraint. In elements that are cut by the material interface $\Gamma = \{x : \psi(x) = 0\}$, the material parameters are averaged by linear interpolation. Denoting the material parameters in Ω_i by α_i, $i = 1, 2$, the material parameter in element T is computed by

$$\alpha|_T = \frac{|T \cap \Omega_1|}{|T|}\alpha_1 + \frac{|T \cap \Omega_2|}{|T|}\alpha_2.$$

Using piecewise linear finite elements, the resulting topological derivatives are often piecewise constant functions. In order to get a representation of the topological derivative in V_h, which is necessary for the numerical realization of (2.5), we compute an average of the topological derivative around each node of the mesh. This gives us a function $\tilde{G} \in V_h$. Note that this latter procedure can also be seen as a sensitivity filtering technique. Such techniques are well-known in density-based topology optimization as regularization methods.

3 Multi-material topology optimization using topological derivatives

In this section, we introduce a generalization of Algorithm 1 to the case of an arbitrary number of materials. We will represent the design consisting of M materials by means of a vector-valued level-set function ψ mapping from the hold-all domain D into $\mathbb{R}^{M-1}$. The algorithm will be very similar to Algorithm 1 with an appropriately chosen generalized topological derivative G.

3.1 Topological derivatives in multi-material setting

We consider again an open and bounded hold-all domain $D \subset \mathbb{R}^d$ and a set of admissible subsets of D, $\mathscr{A} \subset \mathscr{P}(D)$. For $M > 2$ let

$$\mathscr{A}_M := \Big\{ (\Omega_1, \dots, \Omega_M) \in \mathscr{A} \times \cdots \times \mathscr{A} : \overline{D} = \bigcup_{l=1}^{M} \overline{\Omega}_l, \ \Omega_i \text{ open} \\ \text{and } \Omega_i \cap \Omega_j = \emptyset \text{ for } i, j \in \{1, \dots, M\}, i \neq j \Big\}.$$

A multi-material shape function is a mapping

$$\begin{aligned} \mathscr{J} : \mathscr{A}_M &\to \mathbb{R} \\ (\Omega_1, \dots, \Omega_M) &\mapsto \mathscr{J}(\Omega_1, \dots, \Omega_M). \end{aligned}$$

Similarly to Definition 2.1, we define the topological derivatives of a multi-material shape function $\mathscr{J}$ for $(\Omega_1, \dots, \Omega_M)$ at a point $z \in \Omega_i$, $i \in \{1, \dots, M\}$:

Definition 3.1. Let $\mathscr{J} : \mathscr{A}_M \to \mathbb{R}$ a multi-material shape function and $(\Omega_1, \dots, \Omega_M) \in \mathscr{A}_M$. Let $\omega \subset \mathbb{R}^d$ open with $0 \in \omega$ and $i, j \in \{1, \dots, M\}$ fixed with $i \neq j$. Let $z \in \Omega_i$ and, for $\varepsilon > 0$ small, let $\omega_\varepsilon := z + \varepsilon\omega$. Then, the topological derivative at the point $z \in \Omega_i$ with respect to Ω_j is defined as the limit

$$\mathscr{T}^{i \to j}(z) := \lim_{\varepsilon \searrow 0} \frac{\mathscr{J}(\Omega_1, \dots, \Omega_i \setminus \omega_\varepsilon, \dots, \Omega_j \cup \omega_\varepsilon, \dots, \Omega_M) - \mathscr{J}(\Omega_1, \dots, \Omega_M)}{\ell(\varepsilon)}$$

where $\ell : \mathbb{R}^+ \to \mathbb{R}^+$ is a continuous function satisfying $\ell(\varepsilon) \to 0$ as $\varepsilon \to 0$.

Moreover, for all $x \in \Omega_i$, we define the vector of topological derivatives

$$\mathscr{T}^{(i)}(x) := \begin{pmatrix} \mathscr{T}^{i \to 1}(x) \\ \dots \\ \mathscr{T}^{i \to i-1}(x) \\ \mathscr{T}^{i \to i+1}(x) \\ \dots \\ \mathscr{T}^{i \to M}(x) \end{pmatrix} \in \mathbb{R}^{M-1}, \tag{3.1}$$

and for all $x \in D \setminus \left(\bigcup_{j=1}^{M} \partial\Omega_j \right)$ we define

$$\mathscr{T}(x) := \sum_{i=1}^{M} \chi_{\Omega_i}(x) \mathscr{T}^{(i)}(x).$$

Again, we obtain a notion of local optimality:

Definition 3.2. Let a multi-material shape function $\mathscr{J} : \mathscr{A}_M \to \mathbb{R}$ be given. A tuple of domains $(\Omega_1, \dots, \Omega_M) \in \mathscr{A}_M$ is locally optimal for the minimization of $\mathscr{J}$ with respect to topological changes if

$$\mathscr{T}(x) > \mathbf{0} \in \mathbb{R}^{M-1} \tag{3.2}$$

for all $x \in D \setminus \left(\cup_{j=1}^{M} \partial\Omega_j \right)$.

Here, $\mathbf{0}$ represents the zero vector in $\mathbb{R}^{M-1}$ and the inequality in (3.2) is meant componentwise.

3.2 Domain representation

We represent a design $(\Omega_1, \dots, \Omega_M) \in \mathscr{A}_M$ by means of a vector-valued level set function $\psi : D \to \mathbb{R}^{M-1}$. Similarly to the two-material case, we will divide the image space of ψ, $\mathbb{R}^{M-1}$, into M different convex, open sectors $S_1, \dots, S_M$ such that $\mathbb{R}^{M-1} = \bigcup_{l=1}^{M} \overline{S}_l$. Each sector S_l, $l = 1, \dots, M$, is uniquely determined by $M-1$ hyperplanes $H_{l,1}, \dots, H_{l,l-1}, H_{l,l+1}, H_{l,M}$. Each hyperplane $H_{i,j}$ is uniquely determined by its normal vector $n^{i\to j} \in \mathbb{R}^{M-1}$ which is oriented such that $n^{i\to j}|_{\overline{S}_i \cap \overline{S}_j}$ is pointing out of sector S_i and into sector S_j, see Figure 2 for an illustration in the case $M = 3$.

Definition 3.3. For $l \in \{1, \dots, M\}$, we define the matrix of normal vectors pointing into sector S_l by

$$N^{(l)} := \begin{pmatrix} (n^{1\to l})^\top \\ \dots \\ (n^{l-1\to l})^\top \\ (n^{l+1\to l})^\top \\ \dots \\ (n^{M\to l})^\top \end{pmatrix} \in \mathbb{R}^{M-1,M-1}.$$

Assumption A. We assume that the sectors S_l, $l = 1, \dots, M$,

(a) are convex and

(b) are chosen in such a way that $N^{(l)}$ is invertible for all $l \in \{1, \dots, M\}$.

Remark 3.4. In [30] and several other publications, the sectors are defined just by the signs of otherwise independent level set functions. As an example, when $M = 3$, in that context the three sectors would be given as $S_1 = \{(\psi_1, \psi_2) : \psi_1 < 0\}$, $S_2 = \{(\psi_1, \psi_2) : \psi_1 > 0 \wedge \psi_2 < 0\}$, $S_3 = \{(\psi_1, \psi_2) : \psi_1 > 0 \wedge \psi_2 > 0\}$. Such a configuration is excluded by virtue of Assumption A(b) since the normal vectors $n^{2\to 1}$ and $n^{3\to 1}$ would be linearly dependent and therefore the matrix $N^{(1)}$ would not be invertible in this setting. In our approach, it is important that the angles between the hyperplanes are smaller than 180°.

For the rest of this paper, we assume that Assumption A is satisfied. In this setting, it holds that every sector S_l, $l = 1 \dots, M$, can be written as

$$S_l = \{y \in \mathbb{R}^{M-1} : y \cdot n^{j\to l} > 0 \ \forall j \neq l\}. \tag{3.3}$$

The subdomains Ω_l, $l \in \{1, \dots, M\}$ are represented by means of a vector-valued level set function $\psi : D \to \mathbb{R}^{M-1}$ as

$$x \in \Omega_l \Longleftrightarrow \psi(x) \in S_l \tag{3.4}$$

for all $l \in \{1, \dots, M\}$.

3.3 Multi-material level set algorithm

The main challenge in generalizing Algorithm 1 to the multi-material case is to define a generalized topological derivative such that, on the one hand, an optimality condition similar to the one in Lemma 2.4 holds, and on the other hand the resulting algorithm is evolving along a descent direction, similarly to the observation made in Lemma 2.5.

We define the generalized topological derivative in the following way:

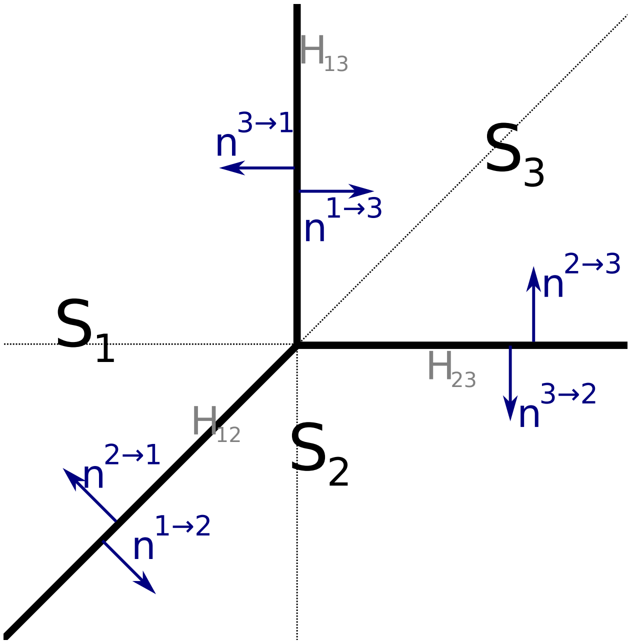


Figure 2: Illustration of sectors in case $M = 3$.

Definition 3.5. For $l = 1, \dots, M$ and $x \in \Omega_l$ we define $G_l : \Omega_l \to \mathbb{R}^{M-1}$,

$$G_l(x) := \left(N^{(l)}\right)^{-1} \mathscr{T}^{(l)}(x), \tag{3.5}$$

with $N^{(l)}$ given by Definition 3.3 and $\mathscr{T}^{(l)}$ defined in (3.1). Moreover, we define the generalized topological derivative $G : D \setminus \left(\bigcup_{j=1}^{M} \partial\Omega_j\right) \to \mathbb{R}^{M-1}$,

$$G(x) := \sum_{l=1}^{M} \chi_{\Omega_l}(x) G_l(x). \tag{3.6}$$

Again note that G is only defined on the union of the interior of the subdomains. We will again indicate the generalized topological derivative for a design represented by a vector-valued level set function ψ by adding a subindex, i.e. by writing G_ψ.

Lemma 3.6. Let the generalized topological derivative $G : D \setminus \left(\bigcup_{j=1}^{M} \partial\Omega_j\right) \to \mathbb{R}^{M-1}$ be defined as in Definition 3.5. Then it holds for all $l, i \in \{1, \dots, M\}$, $i \neq l$, and all $x \in \Omega_l$

$$G|_{\Omega_l}(x) \cdot n^{i \to l} = \mathscr{T}^{l \to i}(x). \tag{3.7}$$

Proof. By definition, we have that

$$\begin{aligned} G|_{\Omega_l}(x) \cdot n^{i \to l} = G_l(x) \cdot n^{i \to l} &= \left[\left(N^{(l)}\right)^{-1} \mathscr{T}^{(l)}(x)\right] \cdot n^{i \to l} \\ &= \left(\mathscr{T}^{(l)}(x)\right)^\top \left(\left(N^{(l)}\right)^\top\right)^{-1} n^{i \to l}. \end{aligned}$$

Since the vector $n^{i \to l}$ appears as a column in the matrix $\left(N^{(l)}\right)^\top$, we get that

$$G|_{\Omega_l}(x) \cdot n^{i \to l} = \left(\mathscr{T}^{(l)}(x)\right)^\top e_k$$

where e_k is the k-th unit vector in $\mathbb{R}^{M-1}$ and k is the column index of $n^{i\to l}$ in $\left(N^{(l)}\right)^\top$, i.e., $k=i$ if $i<l$ and $k=i-1$ if $i>l$. On the other hand, the k-th element of the vector $\mathcal{T}^{(l)}(x)$ is just $\mathcal{T}^{l\to i}(x)$ which finishes the proof. □

Now we are able to show the following optimality condition:

Theorem 3.7. Let $(\Omega_1,\dots,\Omega_M)\in\mathscr{A}_M$. Let $\psi : D\to\mathbb{R}^{M-1}$ be a vector-valued level set function such that (3.4) holds and let G_ψ the generalized topological derivative according to Definition 3.5 for the configuration given by ψ. Suppose that there exists a positive constant $c>0$ such that

$$\psi(x)=c\,G_\psi(x) \tag{3.8}$$

for all $x\in\bigcup_{l=1}^{M}\Omega_l$. Then, $(\Omega_1,\dots,\Omega_M)$ is locally optimal according to Definition 3.2.

Proof. Suppose that $\psi=G_\psi$ and take $\hat{x}\in\Omega_l$ for an arbitrary, but fixed $l\in\{1,\dots,M\}$. Then $\psi(\hat{x})\in S_l$, i.e.,

$$\psi(\hat{x})\cdot n^{i\to l}>0 \quad \text{for all } i\neq l,$$

due to (3.3). Thus, since $\hat{x}\in\Omega_l$, we have by assumption (3.8) and due to Lemma 3.6

$$0<\psi(\hat{x})\cdot n^{i\to l}=c\,G_\psi(\hat{x})\cdot n^{i\to l}=c\,G_\psi|_{\Omega_l}(\hat{x})\cdot n^{i\to l}=c\,\mathcal{T}^{l\to i}(\hat{x}) \quad \text{for all } i\neq l.$$

Since c is positive and l was arbitrary, the statement follows. □

Theorem 3.7 states that $\psi(x)=c\,G_\psi(x)$ with $c>0$ is a local sufficient optimality condition meaning that switching the material in only a small neighborhood of one point to any other material will yield an increase of the objective function. Analogously to the case of two materials, the idea of the algorithm is again to reach a design which satisfies (3.8). Again, this is achieved by a fixed point iteration on the unit sphere of a Hilbert space X where the angle θ between the level set function ψ and the corresponding generalized topological derivative G_ψ is driven to zero. Here, we choose $X=L^2(D,\mathbb{R}^{M-1})$.

Algorithm 2. *Choose initial design $\psi_0\in\mathscr{S}$ and $\varepsilon_\theta>0$ (e.g. $\varepsilon_\theta=0.5°$).*
For $k=0,1,2,\dots$

1. *Compute G_{ψ_k} according to* (3.6)*.*
2. *Compute $\theta_k := arccos\left[\left(\psi_k,\frac{G_{\psi_k}}{\|G_{\psi_k}\|_X}\right)_X\right]$.*
3. *If $\theta_k<\varepsilon_\theta$ then stop,*
   else set
   $$\psi_{k+1}=\frac{1}{sin\,\theta_k}\left[sin((1-\kappa_k)\theta_k)\psi_k+sin(\kappa_k\theta_k)\frac{G_{\psi_k}}{\|G_{\psi_k}\|_X}\right], \tag{3.9}$$
   where $\kappa_k=max\{1,\frac{1}{2},\frac{1}{4},\dots\}$ such that $\mathscr{J}(\psi_{k+1})<\mathscr{J}(\psi_k)$.

Here we used the notation $\mathscr{J}(\psi_k)$ for the objective value for the design $(\Omega_1^{(k)},\dots,\Omega_M^{(k)})$ represented by the vector-valued level set function ψ_k.

Similarly as in the case of two materials, we can show that, when choosing the generalized topological derivative G according to Definition 3.5, Algorithm 2 is evolving along a descent direction:

Theorem 3.8. Let $\psi_k, \psi_{k+1} : D \to \mathbb{R}^{M-1}$ the vector-valued level set functions representing the designs at iterations k and $k+1$ of Algorithm 2, respectively. Then, for all $i, j \in \{1, \dots, M\}, i \neq j$, and all $\hat{x} \in D$, it holds:

$$\psi_k(\hat{x}) \in S_i \wedge \psi_{k+1}(\hat{x}) \in S_j \Longrightarrow \mathcal{T}_{\psi_k}^{i \to j}(\hat{x}) < 0.$$

Here, $\mathcal{T}_{\psi_k}^{i \to j}(\hat{x})$ represents the topological derivative for the design given by ψ_k for switching from material i to material j around point $\hat{x}$.

Proof. We have $\psi_k(\hat{x}) \in S_i$ and $\psi_{k+1}(\hat{x}) \in S_j$ where $i, j \in \{1, \dots, M\}, i \neq j$. From (3.9), we see that

$$\mu \frac{G_{\psi_k}(\hat{x})}{\|G_{\psi_k}(\hat{x})\|} = \psi_{k+1}(\hat{x}) - \lambda \psi_k(\hat{x}). \tag{3.10}$$

with $\lambda, \mu > 0$. Since $-n^{j \to i} = n^{i \to j}$ it holds that

$$(\psi_{k+1}(\hat{x}) - \lambda \psi_k(\hat{x})) \cdot (-n^{j \to i}) = \psi_{k+1}(\hat{x}) \cdot n^{i \to j} + \lambda \psi_k(\hat{x}) \cdot n^{j \to i} > 0$$

because of (3.3) since $\psi_{k+1}(\hat{x}) \in S_j$ and $\psi_k(\hat{x}) \in S_i$. Thus, due to (3.10) and $\hat{x} \in \Omega_i$, we have that also

$$0 < G_{\psi_k}(\hat{x}) \cdot (-n^{j \to i}) = -G_{\psi_k}|_{\Omega_i}(\hat{x}) \cdot n^{j \to i} = -\mathcal{T}_{\psi_k}^{i \to j}(\hat{x})$$

due to Lemma 3.6 and thus $\mathcal{T}_{\psi_k}^{i \to j}(\hat{x}) < 0$.

□

Theorem 3.8 says that, when the design is switched from one material to another at a certain point in the domain in the course of Algorithm 2, then the corresponding topological derivative at iteration k was negative. Thus, the algorithm is somehow evolving along a descent direction, which is an important ingredient for the algorithm to reach the optimality condition of Theorem 3.7.

4 Numerical Experiments

We will illustrate the flexibility and the potential of Algorithm 2 in two classes of examples. On the one hand, we consider classical applications of minimizing the compliance in the framework of linearized elasticity. On the other hand, we show an academic example where the exact solution, which consists of $M = 8$ phases, is known. Before showing numerical results, we will discuss how to obtain the necessary data structure for an arbitrary number of materials M.

4.1 Creating data structure in arbitrary dimensions

For the practical realization of Algorithm 2 with an arbitrary number of materials M, it is important to have access to the normal vector matrices $N^{(l)}$, $l = 1, \dots, M$, introduced in Definition 3.3, which define the sectors $S_1, \dots, S_M$ of $\mathbb{R}^{M-1}$. In particular, it is important to ensure that each of the matrices $N^{(l)}$ is invertible (cf. Assumption A). This is equivalent to making sure that, for each $l \in \{1, \dots, M\}$, the normal vectors $n^{i \to l}$, $i \neq l$, pointing into sector S_l are linearly independent. We emphasize that this data has to be created only once for any fixed M. For that reason, we provide a MATLAB implementation of this part of the code in A. Since the code has to be run only once for any fixed M and the computational time is negligible for a moderate number of materials M, we put an emphasis on readability of the code over computational efficiency.

M=3

i	j	$n^{i\to j}$
1	2	$(\sqrt{2}/2, -\sqrt{2}/2)^\top$
1	3	$(1,0)^\top$
2	3	$(0,1)^\top$

M=4

i	j	$n^{i\to j}$
1	2	$(\sqrt{2}/2, -\sqrt{2}/2, 0)^\top$
1	3	$(\sqrt{2}/2, 0, -\sqrt{2}/2)^\top$
1	4	$(1,0,0)^\top$
2	3	$(0, \sqrt{2}/2, -\sqrt{2}/2)^\top$
2	4	$(0,1,0)^\top$
3	4	$(0,0,1)^\top$

Table 1: Examples of normal vectors $n^{i\to j}$ for $M = 3$ and $M = 4$; results obtained by call of function `createDataStructure(M)` and subsequently `getNormal(...)`, see A.

Recall that each of the sectors S_l is defined by $M-1$ hyperplanes in $\mathbb{R}^{M-1}$. Each of the hyperplanes in $\mathbb{R}^{M-1}$ can be defined by $M-2$ linear independent vectors. We choose these vectors to be defined by two points: the origin $\mathbf{0} \in \mathbb{R}^{M-1}$ and one other point. The structure is fully determined by the choice of these points. The procedure reads as follows:

(1) Define $P_0 = \mathbf{0}$ the origin in $\mathbb{R}^{M-1}$ and choose $P_1, \dots, P_M \in \mathbb{R}^{M-1}$. These points have to be chosen in such a way that Assumption A is satisfied. This holds, e.g., for $P_i = e_i$ the i-th unit vector in $\mathbb{R}^{M-1}$ for $i = 1, \dots, M-1$ and $P_M = (-1, \dots, -1) \in \mathbb{R}^{M-1}$.

(2) Define $\binom{M}{M-2} = M(M-1)/2$ many hyperplanes as the span of all combinations of $M-2$ vectors out of M vectors of the form $(P_i - P_0)$, $i \in 1, \dots, M$.

(3) Define M sectors $S_1, \dots, S_M \subset \mathbb{R}^{M-1}$ where each sector is bounded by a certain combination of $M-1$ hyperplanes.

(4) For $i, j \in \{1, \dots, M\}$, define $n^{i\to j}$ as the unit normal vector of the hyperplane separating sector S_i from sector S_j, oriented in such a way that $n^{i\to j}$ points out of S_i and into S_j. It holds $n^{j\to i} = -n^{i\to j}$.

Examples of the involved normal vectors for $M = 3$ and $M = 4$ can be found in Table 1. Recall that the inverted normal vectors $n^{j\to i}$ can be obtained as $n^{j\to i} = -n^{j\to i}$. The function `createDataStructure(...)` computes the information needed for all normal vectors by following steps (1)–(4) from above. The normal can be extracted by a call of `getNormal(...)`, and the function `isInSector(...)` can be used to check whether a point $p \in \mathbb{R}^{M-1}$ is in a specific sector of $\mathbb{R}^{M-1}$. All of these methods are available in A.

4.2 Application to structural optimization

We illustrate the usage of Algorithm 2 for three classical problems from structural optimization where we minimize the compliance of a structure. We search for the optimal distribution of $M = 3$ different isotropic materials (a strong material, a weak material and void) within a given domain $D \subset \mathbb{R}^2$. For the three phases, we use Young's moduli $E_1 = 1$ (strong), $E_2 = 0.5$ (weak), $E_3 = 10^{-4}$ (void) and Poisson's ratios $\nu_1 = \nu_2 = 0.3333$ and $\nu_3 = 0.3333 * 10^{-4}$. In addition we impose a bound on the allowed volumes in an implicit way, by adding the volumes of the strong material $|\Omega_1|$ and the weak material $|\Omega_2|$, weighted by factors $\ell_1 = 2$ and $\ell_2 = 0.5$, respectively, to the objective function. We solve the minimal compliance problem constrained by the PDE system of linearized elasticity in the plane stress setting,

$$\min_{(\Omega_1,\Omega_2,\Omega_3)\in\mathcal{A}_3} \mathcal{J}(\Omega_1,\Omega_2,\Omega_3) = \mathcal{C}(u) + \ell_1|\Omega_1| + \ell_2|\Omega_2| \tag{4.1a}$$

$$\text{s.t.} \quad -\operatorname{div}(A_{\Omega_1,\Omega_2,\Omega_3}\epsilon(u)) = 0 \quad \text{in } D, \tag{4.1b}$$

$$A_{\Omega_1,\Omega_2,\Omega_3}\epsilon(u)n = g \quad \text{on } \Gamma_N, \tag{4.1c}$$

$$u = 0 \quad \text{on } \Gamma_D, \tag{4.1d}$$

$$A_{\Omega_1,\Omega_2,\Omega_3}\epsilon(u)n = 0 \quad \text{on } \partial D \setminus (\Gamma_N \cup \Gamma_D), \tag{4.1e}$$

on the domain $D \subset \mathbb{R}^2$. Here, $\epsilon(u) = \frac{1}{2}(\nabla u + \nabla u^\top)$ represents the symmetric gradient, $\mathcal{C}(u) = \int_D A_{\Omega_1,\Omega_2,\Omega_3}\epsilon(u) : \epsilon(u)\mathrm{d}x$ represents the compliance of the structure and $A_{\Omega_1,\Omega_2,\Omega_3}(x) = \chi_{\Omega_1}(x)A_1 + \chi_{\Omega_2}(x)A_2 + \chi_{\Omega_3}(x)A_3$ the elasticity tensor. The elasticity tensors A_i, $i = 1,2$, of strong and weak material are determined by the materials' Young's moduli E_i and Poisson's ratios ν_i as

$$A_i e = 2\mu_i e + \lambda_i \mathrm{tr}(e)I, \; e \in \mathbb{R}^{2\times 2},$$

with the Lamé parameters $\lambda_i = \frac{E_i}{2(1+\nu_i)}$ and $\mu_i = \frac{\nu_i E_i}{2(1+\nu_i)(1-\nu_i)}$. The elasticity tensor for the void region is chosen as $A_3 = 10^{-4}(A_1 + A_2)$.

For any point $z \in \Omega_i$ and $i \in \{1,\ldots,M\}$, the topological derivative for introducing an inclusion of material $j \neq i$ around z reads (see e.g. [17])

$$\mathcal{T}^{i\to j}(z) = (\mathbb{P}^{i\to j}A_i\epsilon(u(z))) : \epsilon(u(z)) - \ell_i + \ell_j, \tag{4.2}$$

where $\ell_3 = 0$. Here, we used the Frobenius inner product defined by $A : B = \sum_{i,j=1}^n A_{ij}B_{ij}$ for $A,B \in \mathbb{R}^{n\times n}$. The fourth order polarization tensor $\mathbb{P}^{i\to j}$ is given by

$$\mathbb{P}^{i\to j} = \frac{1}{\beta\gamma + \tau_1}\left[(1+\beta)(\tau_1 - \gamma)\mathbb{I} + \frac{1}{2}(\alpha - \beta)\frac{\gamma(\gamma - 2\tau_3) + \tau_1\tau_2}{\alpha\gamma + \tau_2}(I \otimes I)\right]$$

with the (i,j)-dependent coefficients

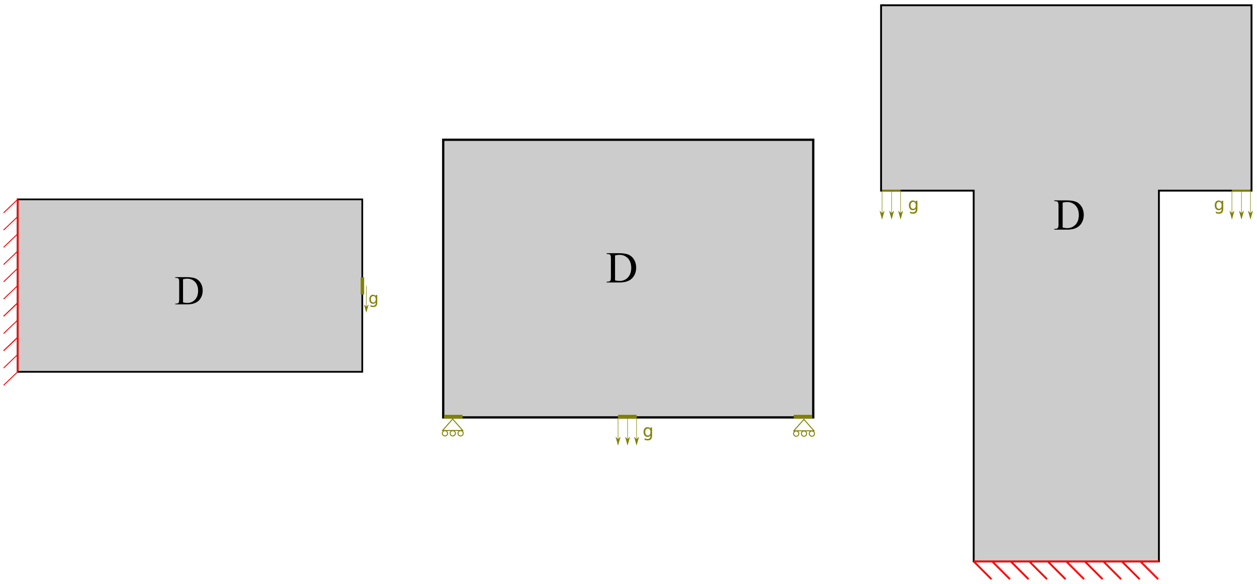
$$\alpha = \alpha^{(i)} = \frac{1+\nu_i}{1-\nu_i}, \qquad \beta = \beta^{(i)} = \frac{3-\nu_i}{1+\nu_i}, \qquad \gamma = \gamma^{(i\to j)} = \frac{E_j}{E_i},$$

$$\tau_1 = \tau_1^{(i\to j)} = \frac{1+\nu_j}{1+\nu_i}, \qquad \tau_2 = \tau_2^{(i\to j)} = \frac{1-\nu_j}{1-\nu_i}, \qquad \tau_3 = \tau_3^{(i\to j)} = \frac{\nu_j(3\nu_i - 4) + 1}{\nu_i(3\nu_i - 4) + 1}.$$

Here, $\mathbb{I}$ denotes the identity tensor of order 4 and I the usual identity matrix in $\mathbb{R}^2$. In the following, we will apply Algorithm 2 to three different settings, which are shown in Figure 3, where also the boundary regions Γ_D with zero displacement and the region Γ_N where a given force is acting, are indicated. For the solution of the constraining boundary value problem (4.1b)–(4.1e), we employed the finite element method using piecewise linear, globally continuous ansatz and test functions on a triangular mesh. The implementation is done in FreeFem++ [18] extending a code that was kindly provided by Dr. Charles Dapogny [12].

In the course of the algorithm, the material interfaces are evolving over a fixed grid. We adopt the procedure suggested in the two-material case in [7] and discussed here in Remark 2.6 to the case of three materials as follows: On the one hand, the elasiticity tensor in the elements T that are cut by a material interface is obtained by linear interpolation between all three materials,

$$A|_T = A_1\frac{|T \cap \Omega_1|}{|T|} + A_2\frac{|T \cap \Omega_2|}{|T|} + A_3\frac{|T \cap \Omega_3|}{|T|}. \tag{4.3}$$



(a) (b) (c)

Figure 3: Settings for three model problems from linear elasticity: (a) long cantilever, (b) bridge, (c) mast.

On the other hand, the topological sensitivities, which in the framework of piecewise linear finite elements are given as piecewise constant on the mesh, are interpolated the other way around, from the centers of the cells to the vertices. This is done by assigning the average of the sensitivities in the cells surrounding a mesh node x_j to the new, filtered sensitivity at this node x_j. It is well-known that this kind of sensitivity filtering has a regularizing effect and avoids the formation of checkerboard patterns [26].

In numerical experiments, we observed (both in the multi-material case as well as in the case of only two materials) that, when choosing an unstructured triangular mesh, the non-symmetry of the mesh can be critical and lead to non-symmetric optimal designs when the step size parameter κ in Algorithm 2 is chosen too large. For this reason, we set an upper bound $\overline{\kappa}$ on the maximum stepsize and choose κ_k in iteration k as $\kappa_k = \overline{\kappa} \max\{1, \frac{1}{2}, \frac{1}{4}, \dots\}$ such that a descent is achieved, cf. step 3 of Algorithm 2. In all of the presented examples, we ran 500 iterations of the algorithm.

4.2.1 Long Cantilever

As a first numerical example, we consider a long cantilever where $D = (-1,1)\times(0,1)$, $\Gamma_D = \{-1\}\times(0,1)$, $\Gamma_N = \{1\}\times(0.45, 0.55)$ and $g = (0,-1)^\top$. We used a mesh with 45824 triangles and 23218 vertices and we chose a maximum stepsize $\overline{\kappa} = 0.12$ in this example.

As an initial design we chose the whole computational domain D to be occupied with strong material. Figure 4 shows the evolution of the design in the course of Algorithm 2. The objective value $\mathcal{J}$ (4.1a) is reduced from 4.42779 to 1.7177 and the angle (in an $L^2(D, \mathbb{R}^{M-1})$ sense) between the level set function and the generalized topological derivative G defined in (3.6) is reduced from 123.5 degrees to approximately 0.4 degrees. Figure 5 shows the evolution of the objective value, of the compliance value, of the angle and the volumes of the strong and weak materials in the course of the optimization iterations.

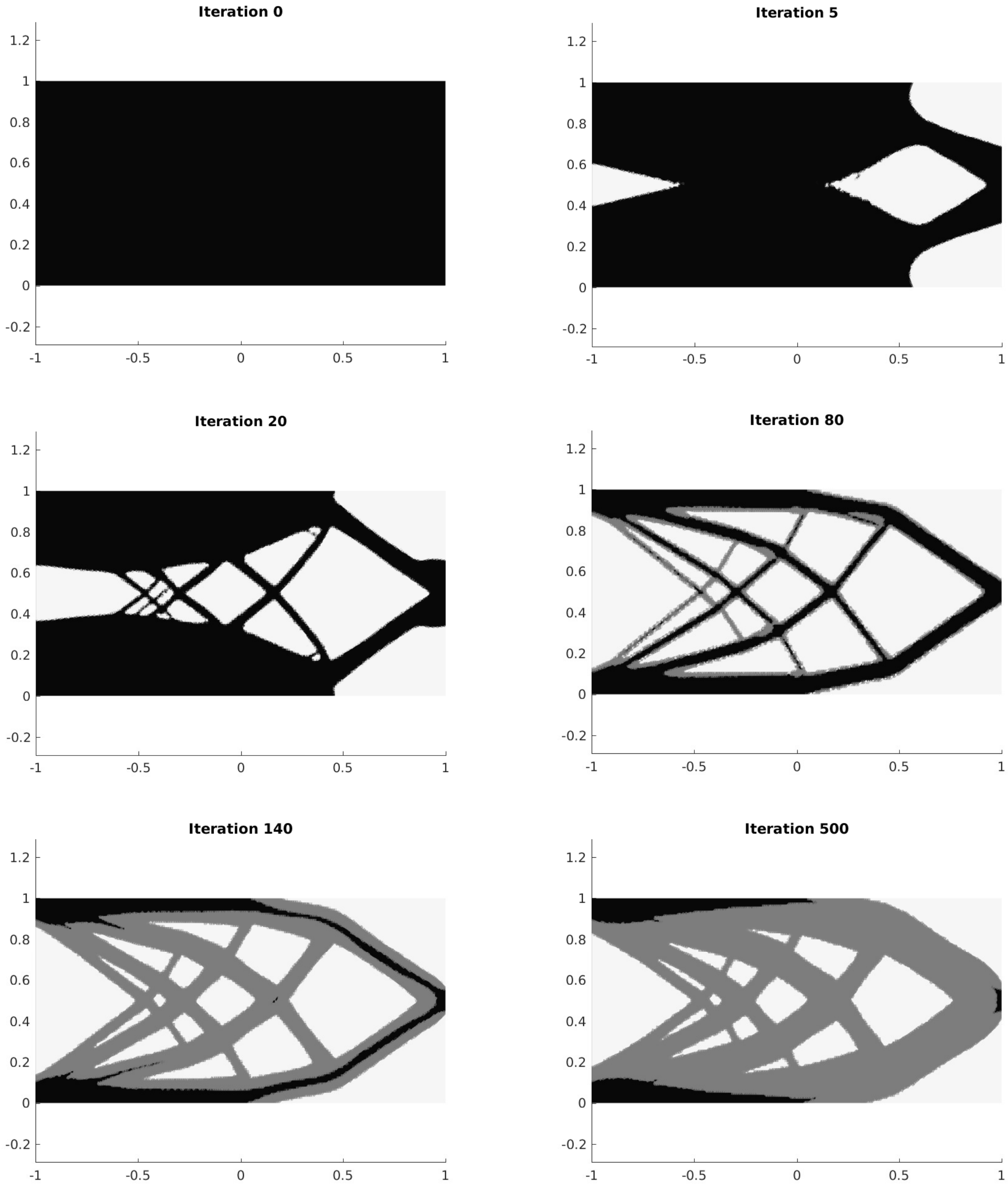


Figure 4: Evolution of design in the course of multi-material topology optimization by Algorithm 2 for long cantilever example. Black color corresponds to the strong material (Ω_1) and gray color corresponds to the weaker material (Ω_2).

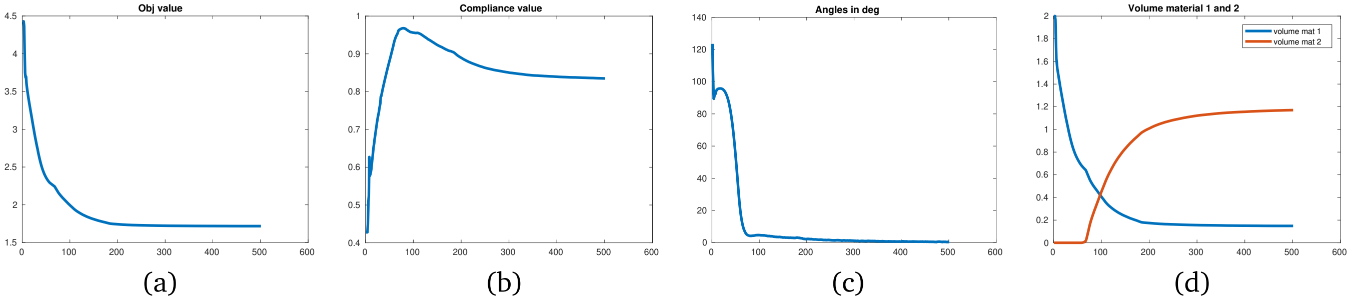


Figure 5: Evolution of different quantities in the course of Algorithm 2 for optimization of long cantilever. (a) Objective value $\mathscr{J}$ (4.1a). (b) Compliance value $\mathscr{C}$. (c) Angle θ between level set function ψ and generalized topological derivative G in $L^2(D, \mathbb{R}^{M-1})$ sense. (d) Volumes of strong and weak materials.

4.2.2 Bridge

In this example, we search for the optimal distribution of three materials (strong, weak and void) within the design region $D = (-1, 1) \times (0, 1.5)$ as depicted in Figure 3(b). The structure is subject to a load $g = (0, -1)^\top$ acting on the bottom in the center, $\Gamma_N = (0.95, 1.05) \times \{0\}$. Moreover, the second component of the displacement vector u is bound to vanish on the left and right bottom regions $(-1, -0.9) \times \{0\}$ and $(0.9, 1) \times \{0\}$. We used a grid consisting of 55144 triangles and 27878 vertices and chose the maximum allowed step size as $\overline{\kappa} = 0.2$. The evolution of the design is shown in Figure 6. The objective value $\mathscr{J}$ (4.1a) is reduced from 6.04865 to 0.646332 and the angle from 126.8 degrees to $4.467 \cdot 10^{-5}$ degrees, see Figure 7.

4.2.3 Mast

In this example, we consider the computational domain shown in Figure 3(c), which we decomposed into a mesh of 50523 triangles and 25638 vertices. The structure is fixed on the bottom, $\Gamma_D = (-0.5, 0.5) \times \{0\}$, and is subject to a vertical load on the left and right parts, $\Gamma_N = (-1, -0.9) \times \{2\} \cup (0.9, 1) \times \{2\}$. We used a maximum step size $\overline{\kappa} = 0.1$. The evolution of the design in the course of Algorithm 2 is depicted in Figure 8. The objective value is reduced from 8.22603 to 1.80394 and the angle θ from approximately 127 degrees to approximately 0.46 degrees, see also Figure 9.

The algorithm introduced in this paper is general and can handle an arbitrary number M of different materials. As mentioned in Remark 2.6, when dealing with PDE-constrained topology optimization problems, the treatment of the interface is important. Also for higher values of M, a linear interpolation of the material parameters in elements that are cut by an interface based on the volume fractions similar to (4.3) is definitely possible, but can be cumbersome as it involves many case distinctions. For this reason, we will illustrate the case of many (here: $M = 8$) materials using an academic example without a constraining PDE. We will see that, in this case, we obtain the expected results even when disregarding the material interfaces.

4.3 Academic Example with M=8

We consider an academic multi-material topology optimization problem which is not constrained by a partial differential equation. We search for the optimal distribution of a given number M of materials inside the fixed domain $D = (0, 1)^2$. For $i \in \{1, \dots, M\}$ let $f_i : D \to \mathbb{R}$ be continuous functions. We aim

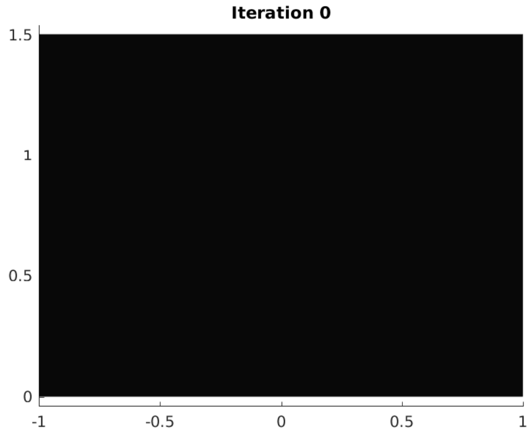


Figure 6: Evolution of design in the course of Algorithm 2 for bridge example. Black color corresponds to the strong material (Ω_1) and gray color corresponds to the weaker material (Ω_2).

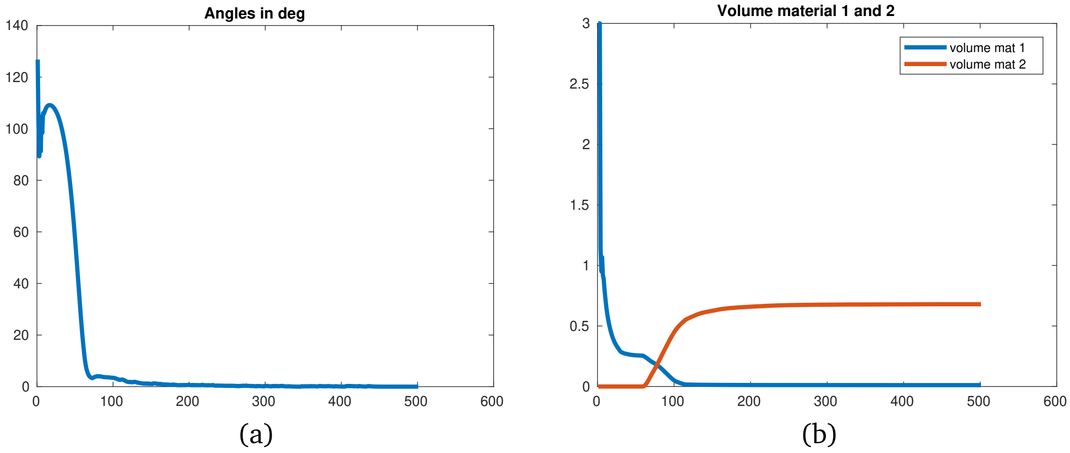


Figure 7: Evolution of angle and volumes in the course of Algorithm 2 applied to optimization of a bridge. (a) Angle between level set function and generalized topological derivative. (b) Volumes of strong material (material 1) and weak material (material 2).

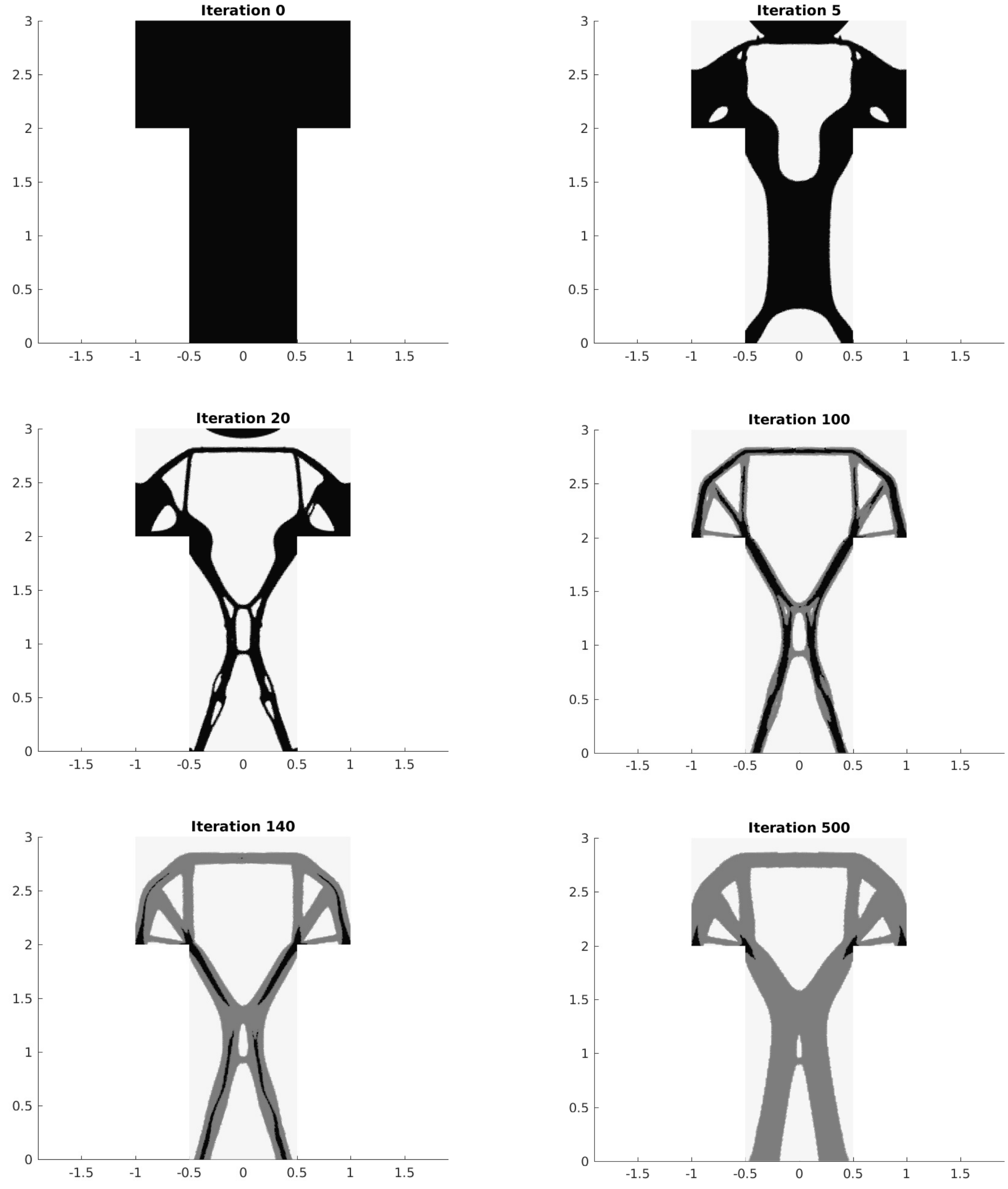
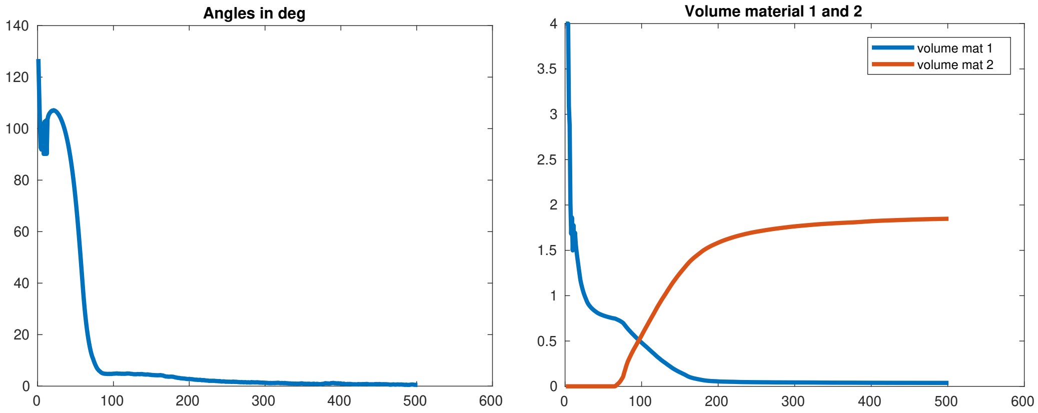


Figure 8: Evolution of design in the course of Algorithm 2 for mast example. Black color corresponds to the strong material (Ω_1) and gray color corresponds to the weaker material (Ω_2).



(a) (b)

Figure 9: Evolution of angle and volumes in the course of Algorithm 2 for mast example. (a) Angle θ between level set function and generalized topological derivative. (b) Volumes of strong material (material 1) and weak material (material 2).

to minimize the multi-material shape function $\mathscr{J} : \mathscr{A}_M \to \mathbb{R}$ defined by

$$\mathscr{J}(\Omega_1, \dots, \Omega_M) := \sum_{\ell=1}^{M} \int_{\Omega_\ell} f_\ell(x)\,\mathrm{d}x. \tag{4.4}$$

The exact solution for this problem is given by

$$\overline{\Omega_\ell^*} = \{x \in D : f_\ell(x) = \min_{j=1,\dots,M} f_j(x)\}$$

for $\ell = 1, \dots, M$. We conducted our experiments for functions f_ℓ of the form

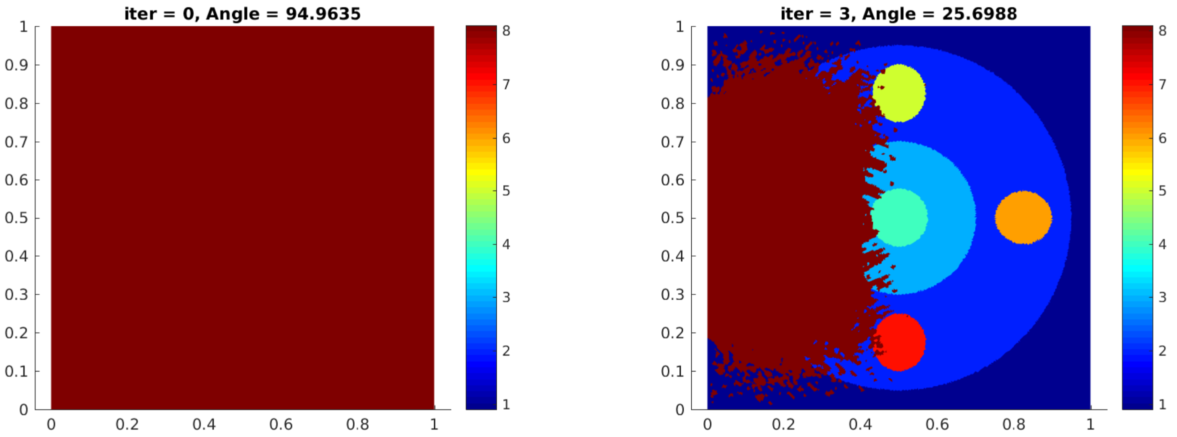
$$f_\ell(x) = w_\ell \|x - x_m^{(\ell)}\| - d_\ell$$

where $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^2$, $w_\ell, d_\ell \in \mathbb{R}$ and $x_m^{(\ell)} \in D$. We chose $M = 8$ materials and the data for w_ℓ, d_ℓ and $x_m^{(\ell)}$ as given in Table 2. Recall the notation $B(x, \delta)$ for the ball around point x of radius δ. The optimal subdomains are then given by

$$\begin{aligned}
\Omega_1^* &= (0,1)^2 \setminus \overline{B((1/2,1/2)^\top, 0.45)},\\
\Omega_3^* &= B((1/2,1/2)^\top, 0.2) \setminus \overline{B((1/2,1/2)^\top, 0.075)},\\
\Omega_4^* &= B((1/2,1/2)^\top, 0.075),\\
\overline{\Omega_i^*} &= \{x \in D : f_i(x) = \min_{j=1,\dots,8} f_j(x)\} = \{x \in D : f_i(x) \leq f_2(x)\},\ i = 5,6,7,8,\\
\Omega_2^* &= B((1/2,1/2)^\top, 0.45) \setminus \overline{B((1/2,1/2)^\top, 0.2) \cup \Omega_5^* \cup \Omega_6^* \cup \Omega_7^* \cup \Omega_8^*}.
\end{aligned}$$

We start with a homogeneous material distribution $\Omega_1 = \dots = \Omega_7 = \emptyset$, $\Omega_8 = D$, see Figure 10 (a). It can be easily verified that the topological derivative of the objective function $\mathscr{J}$ introduced in (4.4) for switching material around a point $z \in \Omega_i$ to material j reads

$$\mathscr{T}^{i \to j}(z) = f_j(z) - f_i(z)$$



(a) (b)

Figure 10: (a) Initial design for academic topology optimization problem (4.4). (b) Intermediate design after three iterations of Algorithm 2.

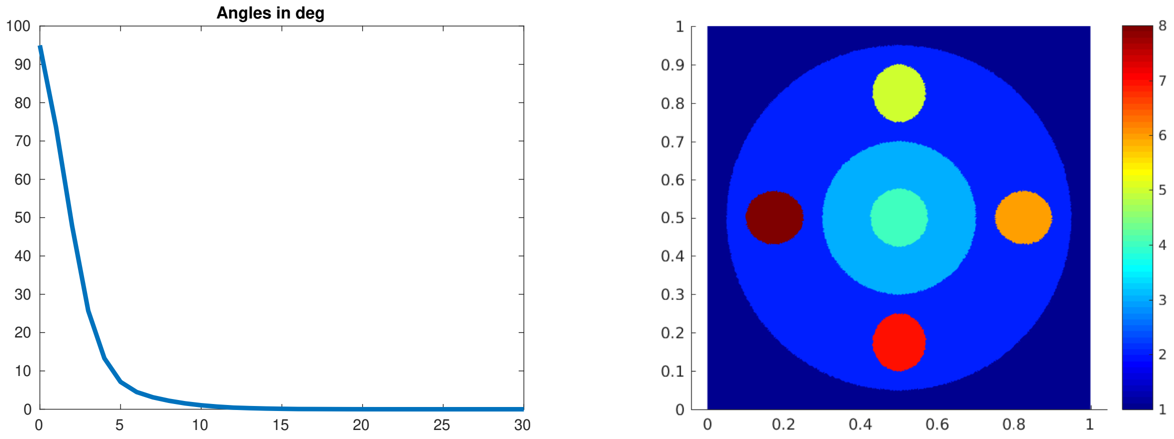
ℓ	w_ℓ	$x_m^{(\ell)}$	d_ℓ
1	0	–	0
2	1	$(0.5, 0.5)^\top$	0.45
3	$\frac{5}{4}$	$(0.5, 0.5)^\top$	0.5
4	$\frac{95}{12}$	$(0.5, 0.5)^\top$	1
5	2	$(0.5, 0.7875)^\top$	0.275
6	2	$(0.7875, 0.5)^\top$	0.275
7	2	$(0.5, 0.2125)^\top$	0.275
8	2	$(0.2125, 0.5)^\top$	0.275

Table 2: Data used for problem (4.4).

when choosing $\ell(\varepsilon) = |\omega_\varepsilon|$. In this case, the topological derivative is independent of the particular choice of the shape of the inclusion ω.

We applied Algorithm 2 with a constant step size $\kappa = 0.5$ and no line search; see Figure 10 (b) for an intermediate design after three iterations. We observe that, for this simple academic example, the angle θ between the vector-valued level set function ψ and the generalized topological derivative G decreases monotonically to zero, see Figure 11 (a), and we reach the optimal material distribution, see Figure 11 (b).

This example serves as a proof of concept and shows that an arbitrary number of materials can be treated. For this example, the treatment of the interfaces between different materials was not critical and we assigned each element of the underlying triangular grid completely to one of the materials based on the value of the vector-valued level set function at the centroid of the triangle. The implementation of this example was done in MATLAB.



(a)

(b)

Figure 11: (a) Evolution of angle between vector-valued level set function ψ and generalized topological derivative G_ψ in optimization. (b) Final design for academic topology optimization problem (4.4); angle $= 5.9764 \cdot 10^{-6}$ degrees.

Conclusion and Outlook

We introduced a multi-material topology optimization method using a level-set framework, which is based on the concepts of topological derivatives. The idea was to represent a design consisting of M different phases by means of a vector-valued level set function mapping into $\mathbb{R}^{M-1}$. Dividing the Euclidean space $\mathbb{R}^{M-1}$ into M sectors, each sector corresponds to one material. Based on this splitting we introduced a local optimality condition for a design consisting of M materials using topological derivatives. The proposed algorithm strives to reach this optimality condition using a fixed point iteration. We illustrated the potential of the algorithm for classical compliance minimization problems with $M = 3$ materials as well as an academic example consisting of $M = 8$ materials. We believe that, besides applications of topology optimization, the method could also be useful in applications from mathematical image segmentation, e.g. for treating a Mumford-Shah type functional as it was done in [19].

Acknowledgement

The author has been supported by Graz University of Technology and would like to gratefully acknowledge the funding. Moreover, the author is grateful for the discussions on the topic of the paper with Samuel Amstutz and to Charles Dapogny for making his FreeFem++ code of various topology optimization approaches publicly available at [12].

A MATLAB implementation for data structure

```
function [normalIdces, normals] = createDataStructure(M)
%Creates data structure necessary for extracting normal
%vectors of hyperplanes defining M sectors of space R^{M-1}
% Input:
%   M ... integer, M>1, number of different materials
% Output:
```

```
%   normalIdces ... M x (M-1) (absolute values between 1 and
%           nMat*(nMat-1)/2; sign of normalIdces(l,k) indicates if
%           corresponding normal is pointing into sector l (+) or
%           out of sector l (-))
%   normals ... M*(M-1)/2 x (M-1); each row corresponds
%           to normal vector of a hyperplane

    eps = 1e-10;    %numerical tolerance
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Step 1: set up M points p1,...,pM in R^{M-1} (p0 = (0,...,0) not stored)
    p=zeros(M,M-1);
    p(1:M-1,1:M-1) = eye(M-1);
    p(M,:) = -1*ones(1,M-1);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Step 2: Define M*(M-1)/2 hyperplanes

    %Step 2a: For each of the M sectors save M-1 points
    pIdxSec = zeros(M,M-1);
    for l=1:M
        count = 0;
        for i=1:M
            if l ~= i
                count = count+1;
                pIdxSec(l,count)=i;
            end
        end
    end

    %Step 2b: Set up hyperplanes; each hyperplane is given by any (M-2) points out of p1,...,pM (plus origin p0)
    nHyPl = M*(M-1)/2;
    nothyperplanes = zeros(nHyPl,2);
    count = 1;
    for i=1:M
        for j=i+1:M
            nothyperplanes(count,1)=i;
            nothyperplanes(count,2)=j;
            count = count+1;
        end
    end
    hyperplanes = zeros(nHyPl,M-2);
    for k=1:nHyPl
        count = 1;
        for j=1:M
            if ~ismember(j, nothyperplanes(k,:))
                hyperplanes(k, count) = j;
                count = count+1;
            end
        end
    end

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Step 3: For each sector, collect indices of bounding hyperplanes
    normalIdces = -(nHyPl+1)*ones(M,M-1); %initialize with a value that will not be attained
    for k=1:nHyPl
        for l=1:M
```

```
            if prod(ismember(hyperplanes(k,:),pIdxSec(l,:))) %if hyperplane k is part of closure of sector l
                j=1;
                while normalIdces(l,j)>=0
                    j = j+1;
                end
                normalIdces(l,j) = k;
            end
        end
    end
    assert( min(min(normalIdces)) > 0);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Step 4: Compute normal vectors of hyperplanes
    normals = zeros(nHyPl,M-1);
    for k=1:nHyPl
       mat = zeros(M-2, M-1);
       for j=1:M-2
           mat(j,:) = p(hyperplanes(k,j),:);
       end
       normals(k,:) = null(mat)'; %normal vector is null space of matrix mat
    end

    %add signs to normalIdces such that sign(..)normals points into sector
    for l=1:M
        for j=1:M-1
            idxNormal = normalIdces(l,j);
            nVec = normals(idxNormal,:);
            for i=1:M-1
                if nVec*p(pIdxSec(l,i),:)' < -eps
                    normalIdces(l,j) = -1*normalIdces(l,j);
                    break;
                end
            end
        end
    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

end
```

createDataStructure.m

```
function [ nij ] = getNormal( i, j, normalIdces, normals )
%Returns normal vector pointing out of sector i and into sector j
%   function uses data structure obtained by
%   'createDataSturcture(...)' to return the correct normal
%   vector between sectors i and j
% Input:
%   i ... index of sector (between 1 and number of materials nMat)
%   j ... index of sector (between 1 and number of materials nMat)
%   normalIdces ... nMat x (nMat-1) (absolute values between 1 and
%           nMat*(nMat-1)/2; sign of normalIdces(l,k) indicates if
%           corresponding normal is pointing into sector l (+) or
%           out of sector l (-))
%   normals ... nMat*(nMat-1)/2 x (nMat-1); each row corresponds
%           to normal vector of a hyperplane
% Output:
%   nij ... normal vector pointing from sector j into sector i
```

```
    assert(i ~= j && 0 < i && i <= size(normalIdces,1) && 0 < j && j <= size(normalIdces
    ,1) );
    for l=1:size(normalIdces,2)
        if ismember(abs(normalIdces(j, l)), abs(normalIdces(i,:)))
            k = l;
            break;
        end
    end
    nij = sign(normalIdces(j, k))*normals( abs(normalIdces(j, k)),:);
end
```

getNormal.m

```
function [banswer] = isInSector(l, p, normals, normalIdces)
% Determines if point p is in sector l given data structure
% obtained by 'createDataSturcture(...)'
% Input:
%   l ... index of sector (between 1 and number of materials nMat)
%   p ... point in R^(nMat-1); must be column vector
%   normalIdces ... nMat x (nMat-1) (absolute values between 1 and
%                   nMat*(nMat-1)/2; sign of normalIdces(l,k) indicates if
%                   corresponding normal is pointing into sector l (+) or
%                   out of sector l (-))
%   normals ... nMat*(nMat-1)/2 x (nMat-1); each row corresponds
%               to normal vector of a hyperplane
    assert(0 < l && l <= size(normalIdces,1) );
    nMat = size(normalIdces,1);
    for j=1:nMat-1  %nMat-1 is the number of hyperplanes bounding a sector
        idx = normalIdces(l,j); %idx has a sign
        if sign(idx)*normals(abs(idx),:)*p < -1e-10
            banswer = false;
            return;
        end
    end
    banswer = true;
end
```

isInSector.m

References

[1] Niels Aage, Erik Andreassen, Boyan Stefanov Lazarov, and Ole Sigmund. Giga-voxel computational morphogenesis for structural design. *Nature*, 550(7674):84–86, 2017.

[2] G. Allaire, C. Dapogny, G. Delgado, and G. Michailidis. Multi-phase structural optimization via a level set method. *ESAIM: COCV*, 20(2):576–611, 2014.

[3] G. Allaire and F. Jouve. Coupling the level set method and the topological gradient in structural optimization. In Springer, editor, *IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials*, pages 3–12, 2006.

[4] Grégoire Allaire, François Jouve, and Anca-Maria Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363 – 393, 2004.

[5] S. Amstutz. Sensitivity analysis with respect to a local perturbation of the material property. *Asymptotic analysis*, 49(1), 2006.

[6] S. Amstutz. Analysis of a level set method for topology optimization. *Optimization Methods and Software - Advances in Shape an Topology Optimization: Theory, Numerics and New Application Areas*, 26(4-5):555–573, 2011.

[7] S. Amstutz and H. Andrä. A new algorithm for topology optimization using a level-set method. *Journal of Computational Physics*, 216(2):573–588, 2006.

[8] M. P. Bendsøe and O. Sigmund. *Topology Optimization: Theory, Methods and Applications*. Springer, Berlin, 2003.

[9] Luise Blank, M Hassan Farshbaf-Shaker, Harald Garcke, Christoph Rupprecht, and Vanessa Styles. Multi-material phase field approach to structural topology optimization. In *Trends in PDE constrained optimization*, pages 231–246. Springer, 2014.

[10] M. Burger, B. Hackl, and W. Ring. Incorporating topological derivatives into level set methods. *Journal of Computational Physics*, 194(1):344–362, 2004.

[11] Clason, Christian and Kunisch, Karl. A convex analysis approach to multi-material topology optimization. *ESAIM: M2AN*, 50(6):1917–1936, 2016.

[12] C. Dapogny. Subject files and solutions to the mini-course “an introduction to shape and topology optimization”, 2018.

[13] M. C. Delfour and J.-P. Zolésio. *Shapes and geometries*, volume 22 of *Advances in Design and Control*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2011. Metrics, analysis, differential calculus, and optimization.

[14] H. A. Eschenauer, V. V. Kobelev, and A. Schumacher. Bubble method for topology and shape optimization of structures. *Structural optimization*, 8(1):42–51, 1994.

[15] F. Feppon, G. Allaire, F. Bordeu, J. Cortial, and C. Dapogny. Shape optimization of a coupled thermal fluid–structure problem in a level set mesh evolution framework. *SeMA Journal*, 76(3):413–458, Sep 2019.

[16] Peter Gangl and Kevin Sturm. A simplified derivation technique of topological derivatives for quasi-linear transmission problems, 2019.

[17] S.M. Giusti, A. Ferrer, and J. Oliver. Topological sensitivity analysis in heterogeneous anisotropic elasticity problem. theoretical and computational aspects. *Computer Methods in Applied Mechanics and Engineering*, 311:134 – 150, 2016.

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

[19] M. Hintermüller and A. Laurain. Multiphase image segmentation and modulation recovery based on shape and topological sensitivity. *Journal of Mathematical Imaging and Vision*, 35(1):1–22, Sep 2009.

[20] Christian Frier Hvejsel and Erik Lund. Material interpolation schemes for unified topology and multi-material optimization. *Structural and Multidisciplinary Optimization*, 43(6):811–825, Jun 2011.

[21] S. Lim and S. Min. Design optimization of permanent magnet actuator using multi-phase level-set model. *IEEE Transactions on Magnetics*, 48(4):1641–1644, April 2012.

[22] A. A. Novotny, J. Sokołowski, and A. Żochowski. *Applications of the Topological Derivative Method*, volume 188 of *Studies in Systems, Decision and Control*. Springer International Publishing, 2019.

[23] Antonio André Novotny and Jan Sokołowski. *Topological derivatives in shape optimization*. Interaction of Mechanics and Mathematics. Springer, Heidelberg, 2013.

[24] S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.

[25] O. Sigmund and J. Petersson. Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Structural optimization*, 16(1):68–75, Aug 1998.

[26] Ole Sigmund and Kurt Maute. Topology optimization approaches. *Structural and Multidisciplinary Optimization*, 48(6):1031–1055, Dec 2013.

[27] J. Sokołowski and A. Zochowski. On the topological derivative in shape optimization. *SIAM Journal on Control and Optimization*, 37(4):1251–1272, 1999.

[28] Luminita A. Vese and Tony F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International Journal of Computer Vision*, 50(3):271–293, Dec 2002.

[29] Michael Yu Wang and Xiaoming Wang. “Color” level sets: a multi-phase method for structural topology optimization with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 193(6):469 – 496, 2004.

[30] Yiqiang Wang, Zhen Luo, Zhan Kang, and Nong Zhang. A multi-material level set-based topology and shape optimization method. *Computer Methods in Applied Mechanics and Engineering*, 283:1570 – 1586, 2015.

[31] Takayuki Yamada, Kazuhiro Izui, Shinji Nishiwaki, and Akihiro Takezawa. A topology optimization method based on the level set method incorporating a fictitious interface energy. *Computer Methods in Applied Mechanics and Engineering*, 199(45):2876–2891, 2010.

Erschienene Preprints ab Nummer 2017/1

2017/1 O. Steinbach, H. Yang: An algebraic multigrid method for an adaptive space-time finite element discretization

2017/2 G. Unger: Convergence analysis of a Galerkin boundary element method for electromagnetic eigenvalue problems

2017/3 J. Zapletal, G. Of, M. Merta: Parallel and vectorized implementation of analytic evaluation of boundary integral operators

2017/4 S. Dohr, O. Steinbach: Preconditioned space-time boundary element methods for the one-dimensional heat equation

2017/5 O. Steinbach, H. Yang: Comparison of algebraic multigrid methods for an adaptive space–time finite element discretization of the heat equation in 3D and 4D

2017/6 S. Dohr, K. Niino, O. Steinbach: Preconditioned space-time boundary element methods for the heat equation

2017/7 O. Steinbach, M. Zank: Coercive space-time finite element methods for initial boundary value problems

2017/8 U. Langer, M. Schanz, O. Steinbach, W.L. Wendland (eds.): 15th Workshop on Fast Boundary Element Methods in Industrial Applications, Book of Abstracts

2018/1 U. Langer, M. Schanz, O. Steinbach, W.L. Wendland (eds.): 16th Workshop on Fast Boundary Element Methods in Industrial Applications, Book of Abstracts

2018/2 S. Dohr, J. Zapletal, G. Of, M. Merta, M. Kravcenko: A parallel space-time boundary element method for the heat equation

2018/3 S. Dohr, M. Merta, G. Of, O. Steinbach, J. Zapletal: A parallel solver for a preconditioned space-time boundary element method for the heat equation

2018/4 S. Amstutz, P. Gangl: Toplogical derivative for nonlinear magnetostatic problem

2018/5 O. Steinbach, M. Zank: A Stabilized Space–Time Finite Element Method for the Wave Equation

2018/6 O. Steinbach, H. Yang: A Space-Time Finite Element Method for the Linear Bidomain Equations

2018/7 O. Steinbach, M. Zank: Coercive space-time finite element methods for initial boundary value problems

2018/8 S. Dohr, K. Niino, O. Steinbach: Space-time boundary element methods for the heat equation

2018/8 O. Steinbach, H. Yang: Space-time finite element methods for parabolic evolution equations: Discretization, a posteriori error estimation, adaptivity and solution

2019/1 O. Steinbach (eds.): 15th Austrian Numerical Analysis Day, Book of Abstracts

2019/2 P. Gangl, K. Sturm: A simplified derivation technique of topological derivatives for quasi-linear transmission problems

2019/3 M. Merkel, P. Gangl, S. Schöps: Shape Optimization of Rotating Electric Machines using Isogeometric Analysis and Harmonic Stator-Rotor Coupling

2019/4 P. Gangl and K. Sturm: Asymptotic analysis and topological derivative for 3D quasi-linear magnetostatics

2019/5 M. Holzmann and G. Unger: Boundary integral formulations of eigenvalue problems for elliptic differential operators with singular interactions and their numerical approximation by boundary element methods

2019/6 M. Neumüller, O. Steinbach: Regularization error estimates for distributed control problems in energy spaces

2019/7 U. Langer, M. Schanz, O. Steinbach, W. L. Wendland (eds.): 17th Workshop on Fast Boundary Element Methods in Industrial Applications , Book of Abstracts

2019/8 O. Steinbach, M. Zank: A note on the efficient evaluation of a modified Hilbert transformation