



Coupled topology and shape optimization using an embedding domain discretization method

Gabriel Stankiewicz¹ · Chaitanya Dev¹ · Paul Steinmann¹

Received: 1 April 2021 / Revised: 6 July 2021 / Accepted: 21 July 2021 / Published online: 19 August 2021
© The Author(s) 2021

Abstract

Density-based topology optimization and node-based shape optimization are often used sequentially to generate production-ready designs. In this work, we address the challenge to couple density-based topology optimization and node-based shape optimization into a single optimization problem by using an embedding domain discretization technique. In our approach, a variable shape is explicitly represented by the boundary of an embedded body. Furthermore, the embedding domain in form of a structured mesh allows us to introduce a variable, pseudo-density field. In this way, we attempt to bring the advantages of both topology and shape optimization methods together and to provide an efficient way to design fine-tuned structures without predefined topological features.

Keywords Coupled topology and shape optimization · Embedding domain discretization · Structural optimization

1 Introduction

Combining shape and topology optimization is favourable due to their dissimilar but complementary nature. The inherent limitations of the first method are overcome by the second method and vice versa. On the one hand, density-based topology optimization is capable of producing complex structures, but fails to represent the component boundary exactly. On the other hand, shape optimization represents the component boundary exactly, but it does not support topological changes and it is not well suited for large design changes. A coupling of topology and shape optimization is possible when an embedding domain discretization is employed. The embedding domain discretization method allows us to incorporate both the density and nodal design variables. The structured mesh that serves as a computational (embedding) domain is assigned with pseudo-densities, whereas the vertices of the discretized boundary of the embedded body serve as nodal design variables.

Shape optimization in its original form treats the finite element nodes directly as design variables (Zienkiewicz 1973). Another approach is to use a parametrized boundary description to improve the smoothness of the final design. Some of the approaches involve using cubic spline functions (Luchi et al. 1980) or B-spline polynomials (Braibant and Fleury 1984) which are used for CAD parametrization. To improve the smoothness of the optimized geometries and the convergence of the optimization algorithms, regularization techniques are employed. Some of these techniques include sensitivity filtering (Bletzinger 2014), sensitivity weighting (normalization) (Kiendl et al. 2014; Wang et al. 2017) or the traction method (Azegami and Wu 1996; Azegami and Takeuchi 2006). The latter treats the sensitivity vector as an external force in a fictitious boundary value problem (BVP). The solution of such BVP yields a smooth, mesh-independent update vector which also handles the update of internal nodes. The traction method is exceptionally attractive due to its flexibility, i.e. the formulation of the fictitious BVP can be modified to obtain any desired smoothing behaviour. Therefore, it is adapted for the shape optimization used in this work. Thanks to these developments, node-based shape optimization is capable of generating smooth, mesh-independent designs. Therefore, it lends itself as a good choice for optimization problems requiring an exact boundary representation. Despite all the improved methodology in shape optimization, its application is still usually limited

Responsible Editor: YoonYoung Kim

✉ Gabriel Stankiewicz
stankiewicz.ga@gmail.com

¹ Institute of Applied Mechanics, Friedrich-Alexander-Universität Erlangen-Nürnberg, Egerlandstrasse 5, 91058 Erlangen, Germany

to problems involving small design changes, the so-called fine tuning steps.

As opposed to node-based shape optimization, topology optimization offers almost unlimited flexibility in designing geometrically complex structures. Without doubt, the most extensive research was devoted to the density method introduced by Bendsoe (1989). For the past three decades, density-based topology optimization has been continuously improved and it successfully found its way into a number of commercial software applications. For a detailed description of the method, we refer to the book by Bendsoe and Sigmund (2013) and the review article by Sigmund and Maute (2013), which also studies the similarities to other approaches in topology optimization. The density-based topology optimization is offered as an easy to implement, powerful tool, see for instance the work by Sigmund (2001a) and by Andreassen et al. (2011), in which compact topology optimization codes are outlined. An inherent challenge of density-based topology optimization is the placement of a crisp boundary based on the grey transition regions. There are a number of methods that treat this matter, including various projection and filtering techniques, see for instance the work by Wang et al. (2011). Additionally, regarding the accuracy of stress computation, De Troya and Tortorelli (2018) incorporated an adaptive mesh refinement scheme in topology optimization to increase the accuracy of stress computation. da Silva et al. (2019) achieved very good accuracy in stress computation using the robust approach, in which smoothed heaviside projection is employed to obtain three black and white density fields to consider manufacturing uncertainties.

In the context of this article, another approach worth discussing is the level set method (LSM) (Allaire et al. 2002; Wang et al. 2003). Conceptually, LSM can be positioned between density-based topology optimization and node-based shape optimization as it offers a possibility to perform topological changes, namely it allows topological holes to be merged (however, it inherently does not support hole nucleation), and it provides an exact boundary representation. Nevertheless, as pointed out by Sigmund and Maute (2013), most approaches to the LSM, which use an ersatz material for the geometry mapping, do suffer from blurred (grey) boundary regions in the mechanical model, similar to the density-based topology optimization. In the approaches involving an ersatz material for the geometry mapping, the challenge of hole nucleation has been addressed e.g. in the works by Burger et al. (2004) and Allaire et al. (2005), in which topological derivatives (Sokolowski and Zochowski 1999) are incorporated and holes are systematically introduced. While these methods fully support topological changes, their boundary representation is not mechanically crisp due to the fact that mechanical models are based on a density distribution. An alternative formulation utilizing the LSM has been proposed by Yamada et al. (2010), which

incorporates a fictitious interface energy. In this method, full topological flexibility is allowed, at the same time guaranteeing a mechanically crisp mechanical model by using a Heaviside function in the equilibrium equation.

One way to ensure a mechanically crisp boundary representation in LSM is to employ a conforming discretization (Ha and Cho 2008). However, this approach shares the drawbacks of node-based shape optimization due to the necessary mesh deformation during design updates. Therefore, to overcome the difficulties that arise with mesh deformation, more preferable approach is to incorporate immersed boundary techniques (IBTs). One of such technique is the extended finite element method (XFEM). The core concept in XFEM is the introduction of additional shape functions to handle discontinuities of the solution fields and their spatial gradients within the elements. XFEM in combination with LSM (LSM-XFEM) has been successfully applied in the works by Belytschko et al. (2003) for simple 2D elasticity, by Kreisss and Maute (2012) for fluid flow, by Villanueva and Maute (2014) for 3D elasticity and by Jenkins and Maute (2015) for fluid-structure interaction. Villanueva and Maute (2017) incorporate CutFEM for fluid flow problems, which is a combination of LSM, XFEM and face-oriented ghost penalty method. Moreover, stress-based topology optimization using LSM-XFEM has been introduced by Sharma and Maute (2018), in which an improved stress computation is carried out using an XFEM informed stabilization scheme. The LSM-XFEM for structural optimization proves to be a successful approach thanks to the mechanically exact representation of the boundary and the ability of hole elimination. Nevertheless, in the context of topological flexibility, LSM for structural optimization involving IBTs still exhibits the abrupt nature of hole nucleation schemes.

Due to the fact that each of the previously introduced methods exhibits a certain set of advantages and drawbacks, it is generally not simple to indicate a single method that is definitely superior over the others. Hence, there is recently a growing number of publications that treat ways to combine different approaches in structural optimization. Eschenauer et al. (1994) introduced a bubble method, in which shape optimization is followed by an introduction of a topological change in form of a spherical hole (bubble). The sequence is then repeated until a certain number of holes were introduced. This allows one to select an optimal topology out of a series of solutions with varying number of holes. Hassani et al. (2013) proposed a method, in which a shell structure is designed at the same time by density variables and, in the out of plane direction, by shape optimization. Christiansen et al. (2014) and Lian et al. (2017) presented a topology optimization using the Deformable Simplicial Complex (DSC) method which utilizes an unstructured mesh able to represent the boundary exactly. The optimization problem is driven by

shape sensitivity information and the topological changes are performed using the topological derivatives. Riehl and Steinmann (2015) proposed a staggered approach in which optimization is driven concurrently by an element removal scheme and shape refinement steps. Andreasen et al. (2020) incorporated a cut element discretization used in LSM into a density-based optimization problem. The methodologies adopted in aforementioned contributions incorporate hole insertion schemes, which generally provide good results, and the boundary is represented exactly before and after the topological change. However, the hole insertion schemes often utilize simple spherical or mesh-dependent geometrical forms (single element removal or element patch removal) and exhibit a sudden solid-to-void transition. For further discussion about the hole insertion schemes and their comparison to the shape generation scheme used in this work, please consult Sect. 3.3.4. Alternatively, Nguyen et al. (2020) combined implicit and explicit discretizations to optimize the topology and generate a structure with crisp interfaces with adaptive mesh coarsening using the DSC method, respectively. In other words, this approach provides an automated transition step from topology to shape optimization.

In the following work, we exploit the shape optimization method using an embedding domain technique, introduced by Riehl and Steinmann (2017), and couple it with a density-based topology optimization. The optimization problem now operates on design variables adopted from both methods, namely the pseudo-densities of the inner elements and the degrees of freedom of the vertices of the embedded boundary. The key feature of the coupling is that both sets of design variables operate on the same mechanical model. This provides full flexibility to perform topological changes using the variable density in the embedding domain and at the same time, define a mechanically crisp interface by means of embedded boundary.

In this coupling method, there is no need to include a hole nucleation scheme, because the holes emerge naturally as a consequence of the pseudo-density updates. Instead, we provide a functionality that recognizes topological holes and introduces a variable shape. This approach is characterized by a significantly less discontinuous nature as compared to the standard hole nucleation schemes. Furthermore, by employing the embedding domain discretization, the shape updates only require the deformation of the embedded boundary as opposed to classical shape optimization using an explicit mesh, in which mesh movement strategies are necessary for the update of dependent nodes. Therefore, larger shape design changes are easy to handle and, consequently, the regularization is simpler. At the end of the optimization run, the design is in general free from grey regions and is defined by an exact boundary.

Thus, this article's contribution to the field of structural optimization can be understood twofold. (1) It extends the shape optimization approach of Riehl and Steinmann (2017) by introducing density design variables into the embedding domain. This allows to perform topological changes in a flexible manner, characteristic to density-based topology optimization. (2) It enhances the density-based topology optimization with a $d-1$ dimensional (where d corresponds to the number of dimensions of the computational space) boundary mesh which serves as a variable, mechanically crisp boundary that clearly defines the shape of the optimized structure. Our method brings a number of advantages as compared to other methods of combining topology and shape optimization and, naturally, at the same time introduces a set of challenges, which are thoroughly studied in the scope of this article.

At this point we would like to remark that the purpose of this article is not to strive for an approach that is ultimately superior over the other, well-established methods, but rather to show that (1) an actual coupling of density-based topology optimization and node-based shape optimization is indeed possible and (2) it offers a certain set of advantages that makes this approach a reasonable choice for a number of engineering problems. Nonetheless, the scope of this article is limited to a number of simple, benchmark problems of linear elasticity.

2 Embedding Domain method

We consider a body \mathcal{B} , embedded into a d -dimensional domain $\Omega \subset \mathbb{R}^d$ such that $\mathcal{B} \subset \Omega$ (see Fig. 1). The boundary of \mathcal{B} is defined by the Dirichlet portion Γ^D and the Neumann portion Γ^N , such that $\Gamma^D \cup \Gamma^N = \Gamma$ and $\Gamma^D \cap \Gamma^N = \emptyset$.

2.1 Physical problem

For the verification of the proposed coupling method by means of classical benchmark examples, we employ the BVP formulation based on linear isotropic elasticity, which is given by the following system of equations:

$$\begin{aligned} \operatorname{div} \boldsymbol{\sigma} + \mathbf{b} &= \mathbf{0} \\ \boldsymbol{\sigma} \cdot \mathbf{n} &= \bar{\mathbf{t}} \quad \text{on } \Gamma^N \\ \mathbf{u} &= \bar{\mathbf{u}} \quad \text{on } \Gamma^D \\ \boldsymbol{\sigma} &= \mathbb{C} : \boldsymbol{\epsilon} \\ \boldsymbol{\epsilon} &= \frac{1}{2} [\nabla \mathbf{u} + \nabla \mathbf{u}^T] \end{aligned}, \quad (1)$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, \mathbf{b} body forces, \mathbf{n} outward pointing normal to Γ^N , $\bar{\mathbf{t}}$ prescribed traction forces, \mathbf{u} the sought-after displacement vector, $\bar{\mathbf{u}}$ imposed displacement and \mathbb{C} the fourth-order elasticity tensor.

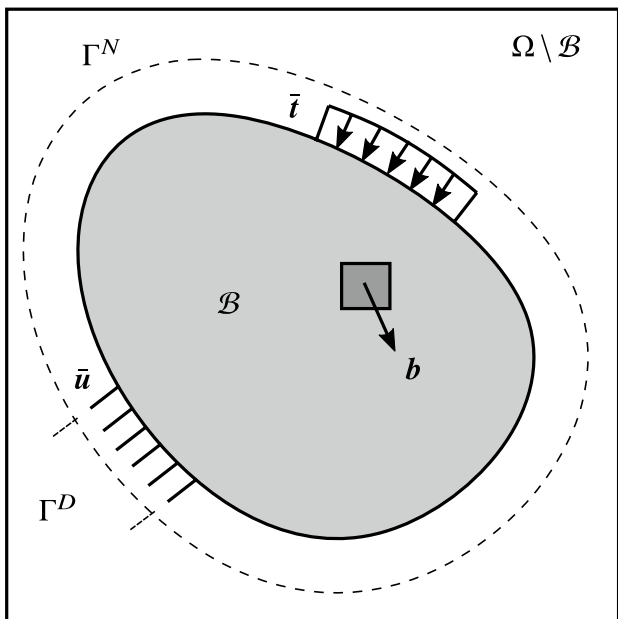


Fig. 1 Embedded body

The weak form of equilibrium, adapted for the embedding domain method, is given by

$$\int_{\Omega} \chi[\sigma : \delta\epsilon] dV - \int_{\Omega} \chi[b \cdot \delta u] dV - \int_{\Gamma^N} \bar{t} \cdot \delta u dA = 0 \tag{2}$$

where χ is a characteristic function defined as

$$\chi = \chi(x) = \begin{cases} 1, & \text{if } x \in B \\ 0, & \text{if } x \in \Omega \setminus B. \end{cases} \tag{3}$$

Equation (2) represents the principle of virtual work, given by

$$\delta W_{\chi}(u, \delta u) = a_{\chi}(u, \delta u) - l_{\chi}(\delta u) = 0, \tag{4}$$

where $a_{\chi}(u, \delta u)$ is the bilinear form representing the variation of stored energy

$$a_{\chi}(u, \delta u) = \int_{\Omega} \chi[\sigma : \delta\epsilon] dV, \tag{5}$$

and $l_{\chi}(\delta u)$ is the linear form representing the variation of external energy

$$l_{\chi}(\delta u) = \int_{\Omega} \chi[b \cdot \delta u] dV + \int_{\Gamma^N} \bar{t} \cdot \delta u dA. \tag{6}$$

The treatment of the Dirichlet boundary conditions cannot be realized directly due to the misalignment of the finite element nodes of Ω^h with the boundary Γ^D . Thus, we introduce

a penalty functional to Eq. (4), so that the total (penalized) variation of the virtual work is given by

$$\delta W_{\chi}^p(u, \delta u) = \delta W_{\chi}(u, \delta u) + \beta \int_{\Gamma^D} [u - \bar{u}] \cdot \delta u dA, \tag{7}$$

where the penalty factor is in the range $\beta \in [10^6, 10^9] \times \|C\|$.

2.2 Discretization

In order to solve the equilibrium equation given by Eq. (2), we introduce an approximation of the embedding domain Ω by d dimensional finite elements

$$\Omega \approx \Omega^h = \bigcup_{e=1}^{n_e} \Omega_e. \tag{8}$$

Consequently, we introduce a piecewise linear representation of the boundary $\Gamma \approx \Gamma^h$ consisting of $d - 1$ dimensional element-like segments Γ^s , i.e. (1) line segments for $d = 2$, (2) polygons for $d = 3$

$$\Gamma \approx \Gamma^h = \bigcup_{s=1}^{n_s} \Gamma_s, \tag{9}$$

where n_s corresponds to the number of segments that belong to Γ^h . As depicted in Fig. 2a, the embedded boundary Γ^h assigns the finite elements Ω^e of the embedding domain Ω^h into three sets: inner elements Ω_{in}^e that are completely enclosed within the embedded body, boundary elements Ω_{bnd}^e collecting all the elements intersected by the embedded boundary, and outer elements Ω_{out}^e that lie outside of the embedded body.

Following the isoparametric concept, the geometry and the solution field are both interpolated using the same (vector-valued) shape functions

$$X^e(\xi) = \sum_{n=1}^{n_{dof}} N^{(i)}(\xi) X^{(i)}, \tag{10}$$

$$u^e(\xi) = \sum_{n=1}^{n_{dof}} N^{(i)}(\xi) u^{(i)}. \tag{11}$$

The system of equations resulting from discretization of Eq. (2) is then given by

$$Ku^h = f, \tag{12}$$

where the IJ -th contribution (capital letters I, J indicate global numbering of the DoFs) of the stiffness matrix K reads

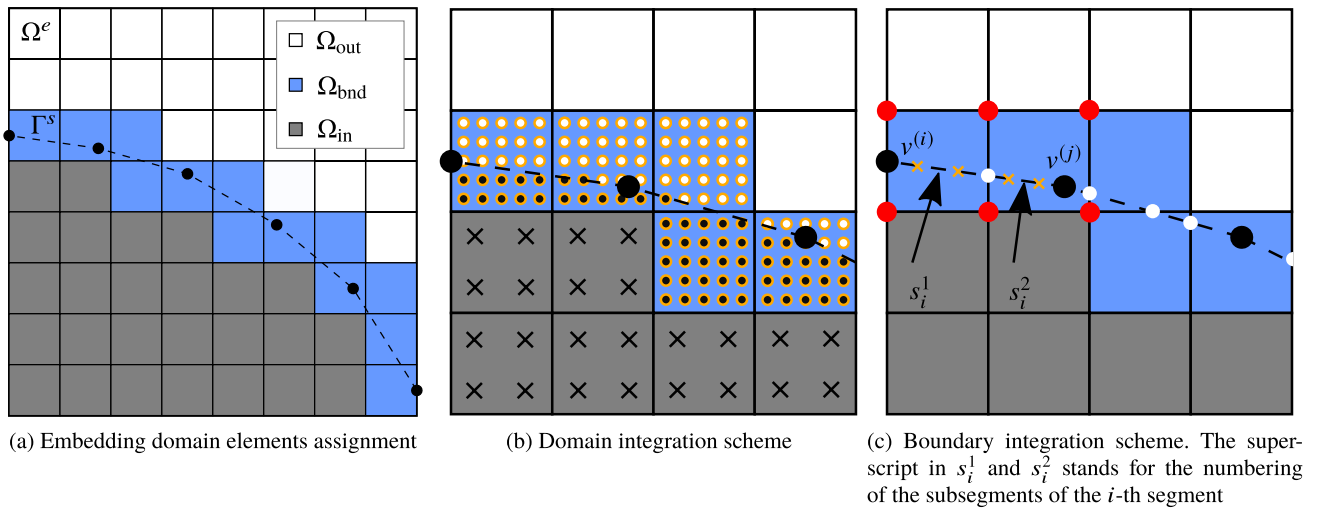


Fig. 2 Embedding domain discretization

$$K^{IJ} = \mathbf{A}_{e=1}^{n_e} \int_{\Omega^e} \chi \nabla N^{(i)} : \mathbb{E} : \nabla N^{(j)} \, dV \tag{13}$$

and the force vector is given by

$$f^I = \mathbf{A}_{e=1}^{n_e} \int_{\Omega^e} \chi N^{(i)} \cdot \mathbf{b} \, dV + \mathbf{A}_{e=1}^{n_e} \int_{\Gamma^N \cap \Omega^e} N^{(i)} \cdot \bar{\mathbf{t}} \, dA. \tag{14}$$

2.3 Integration schemes

Evaluation of the domain integrals in the inner elements Ω_{in} incorporates the usual Gaussian quadrature rule as in the standard finite element discretization (see Fig. 2b). The elements that lie entirely outside of the embedded domain Ω_{out} are excluded from the assembly routine and, thus, the size of the mechanical problem is greatly reduced. Boundary elements Ω_{bnd} are only partially enclosed within the embedded body and, hence, they require a special treatment during numerical integration. There are several different approaches for the integration of cut elements advocated in the literature, which, for instance, involve the area weighting method (Dunning et al. 2011; García-Ruíz and Steven 1999), subtriangulation of the boundary elements (Nadal et al. 2013) or introduction of additional, discontinuous shape functions (XFEM) (Chessa et al. 2002). The first approach does not capture the actual location of the boundary resulting in a low accuracy, whereas the second method loosens the concept of a structured mesh. On the other side, XFEM formulation does not suffer from these drawbacks. In our work, we employ a simple method of integration points oversampling (Riehl 2019) in which a larger amount of integration points, which are usually in the range $n_{ip} = (5..10)^d$, is considered. As depicted in Fig. 2b, for each integration point, we individually decide whether it

contributes to the integral, based on the value of the characteristic function χ

$$\int_{\Omega^e} \bar{\chi}(\mathbf{x}(\xi)) f(\xi) \, dV \approx \sum_{m=1}^{n_{ip}} \bar{\chi}(\xi_m) J(\xi_m) w_m. \tag{15}$$

To avoid numerical instabilities due to sharp material discontinuity at the boundary of the embedded body, we slightly regularize the definition of the characteristic function

$$\bar{\chi} = \bar{\chi}(\mathbf{x}^e(\xi_m)) = \begin{cases} 1, & \text{if } \mathbf{x} \in \mathcal{B} \\ 10^{-3}, & \text{if } \mathbf{x} \in \Omega^h \setminus \mathcal{B} \text{ and } e \in \Omega_{bnd} \\ 0, & \text{if } e \in \Omega_{out}. \end{cases} \tag{16}$$

in which $\mathbf{x} \in \mathcal{B}$ indicates that \mathbf{x} belongs to the inside elements Ω_{in} and inside portion of the boundary elements Ω_{bnd} . The treatment of the boundary integrals is more straightforward as compared to the domain integrals. As shown in Fig. 2c, one needs to further split each segment Γ^s at their intersections with the element boundaries $\partial\Omega^e$ (coloured white in Fig. 2c) and integrate each of the subsegments using the standard Gauss quadrature rule. Contributions from these subsegments (see for instance the subsegments s_i^1 and s_i^2 in Fig. 2c) are then assembled at the respective nodes of the boundary elements Ω^e (coloured red in Fig. 2c).

Contribution of the k -th node in the element e from the tractions is then given by

$$\int_{\Gamma^N \cap \Omega^e} \bar{\mathbf{t}} \cdot N^k \, dA = \sum_{i=1}^{n_s} \int_{s_i^k} \bar{\mathbf{t}} \cdot N^k \, dA. \tag{17}$$

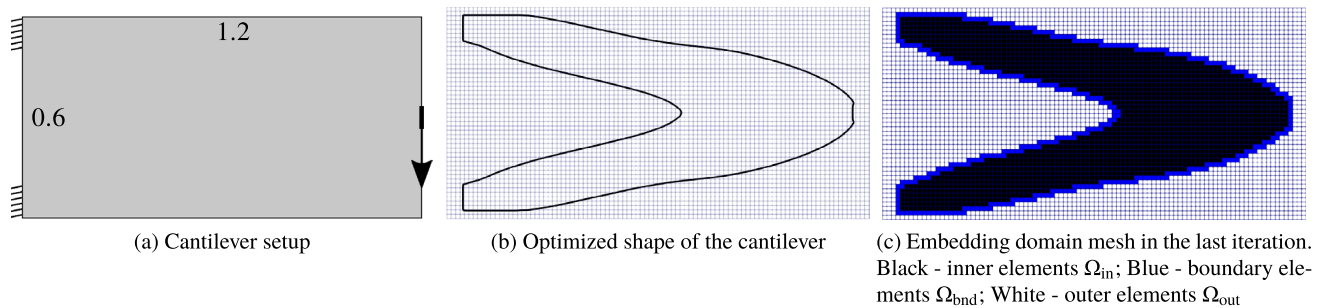


Fig. 3 Shape optimization of the cantilever using embedding domain discretization

3 Optimization method

In the following work, we employ the augmented Lagrangian formulation for a constrained optimization problem and solve it using the Multiplier Method (Hestenes 1969; Rockafellar 1973). Since the purpose of this article is to primarily demonstrate a method to unify topology and shape optimization, we restrict ourselves to a standard compliance minimization problem with a volume constraint.

3.1 Tackling shape

The boundary of the embedded body Γ explicitly represents the variable shape. Discretization of Γ into segments Γ_s introduces vertices, which are naturally adopted as a vector of discrete design variables \mathbf{v} for shape optimization. At this point we would like to remark that working directly with the vertices \mathbf{v} is essentially equivalent to working with the finite element nodes of an explicit mesh in a standard shape optimization procedure. The difference is that in case of an embedded body we deal with a $d - 1$ dimensional mesh, which can be imagined as a hollow body. This fact means that no morphing of the so-called dependent nodes of the interior of a shape is necessary to maintain mesh regularity. Thus, by using the embedding domain discretization, we overcome one of the main difficulties associated with the classical shape optimization method. Furthermore, easier manipulation of the shape means that much larger design changes can be handled without sacrificing the mesh quality.

The shape sensitivity information is obtained on the continuum level and transformed into boundary integral form (Choi and Kim 2004). For a detailed account on this matter, see the Appendix in Sect. 1.

No need for expensive remeshing/morphing schemes for the dependent, inner nodes does not mean that we can directly treat the sensitivity information as the design update. Jagged boundaries and mesh dependency is an integral part of any shape optimization routine and need to be addressed in any event. To handle this issue, we have adapted the traction

method by Azegami and Takeuchi (2006) for the embedded body. For details, please consult Sect. 1.

In Fig. 3, a benchmark compliance minimization problem was solved with a volume fraction constraint of 0.5. Fig. 3a depicts the problem setup and Fig. 3b, c show the optimized shape together with the embedding mesh. The material parameters used are $E = 70$ GPa; $\nu = 0.3$; the bending load equals $F = 0.1$ N for an assumed thickness of $t = 1$ mm. One can immediately notice the large shape change to reach the optimal state. In particular, the left hand side region was updated towards the inside of the cantilever by more than half of the cantilever length. Instead, if a standard approach with an unstructured mesh were used, the optimization would lead to severe mesh distortion and, thus, the usability of the solution would be questionable.

3.2 Tackling topology

The mechanical model in density methods for topology optimization is based on a perfectly structured mesh, which also is the case with an embedding domain discretization. Upon further inspection, one can come to the conclusion that the characteristic function χ (see Eq. (2)) and the pseudo-density field ρ used in density methods are in a certain sense equivalent. Both of the functions indicate how much material is stored (or whether there is material or not) in each finite element. Hence, one can understand the characteristic function χ as a pseudo-density that is 1 only within the embedded body \mathcal{B} . The difference is that the original pseudo-density can take intermediate values within the range $\rho \in [0, 1]$ and it cannot assign two distinct values within a single element. The similarity between χ and ρ can be used to our advance. In the following, we introduce the pseudo-density field to the inner domain by defining a hybrid function ζ

$$\zeta = \zeta(\mathbf{x}, \rho(\mathbf{x})) = \begin{cases} \rho(\mathbf{x}), & \text{if } \mathbf{x} \in \mathcal{B} \\ 0, & \text{if } \mathbf{x} \in \Omega \setminus \mathcal{B} \end{cases} \quad (18)$$

which is a slight enhancement of the characteristic function χ in which all the points \mathbf{x} within the body \mathcal{B} are assigned

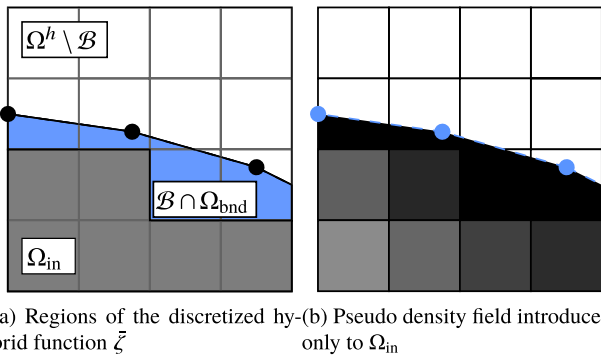


Fig. 4 Pseudo-density field in the discretized embedding domain

a pseudo-density ρ . The weak form of equilibrium is then given by

$$\int_{\Omega} \zeta^p [\boldsymbol{\sigma} : \delta \boldsymbol{\epsilon}] dV - \int_{\Omega} \zeta [\mathbf{b} \cdot \delta \mathbf{u}] dV - \int_{\Gamma^N} \bar{\mathbf{t}} \cdot \delta \mathbf{u} dA = 0, \tag{19}$$

where a penalization parameter is chosen as $p = 3$. Based on our experiments, the discretized version of the hybrid function ζ performs best when adapted in the following manner

$$\bar{\zeta}(\mathbf{x}^e, \rho(\mathbf{x}^e)) = \begin{cases} \rho(\mathbf{x}^e), & \text{if } e \in \Omega_{in} \\ 1, & \text{if } e \in \Omega_{bnd} \text{ and } \mathbf{x}^e \in \mathcal{B} \\ 10^{-3}, & \text{if } e \in \Omega_{bnd} \text{ and } \mathbf{x}^e \in \Omega^h \setminus \mathcal{B} \\ 0, & \text{if } e \in \Omega_{out}. \end{cases} \tag{20}$$

in which the only difference with respect to the characteristic function $\bar{\chi}$ (see Eq. 16) is that the inner elements Ω_{in} are assigned the pseudo-density field ρ (see Fig. 4a for a graphical interpretation of the regions defined in Eq. (20)). The elements that belong to the boundary set Ω_{bnd} are excluded from the pseudo-density field and, thus, treated in an unchanged manner. The motivation behind this exclusion comes from the fact that the shape sensitivities would otherwise show dependence on the value of the pseudo-density. This dependency occurs when coupling of shape and topology optimization is considered and can be avoided by assigning full density to the boundary elements. That is, as already mentioned in Sect. 3.1, due to the fact that the shape sensitivities are expressed exclusively in terms of boundary integrals.

The authors are aware that assigning full density to boundary elements is not conforming with the continuous formulation of the problem and it is, indeed, a discretization related simplification. However, we observe that the finer the embedding domain mesh, the narrower the band of full density along the boundary. The thickness of this band is on average half of the element size. Hence, we discuss that for a fine enough discretization, the "full density"

band is small enough to be considered negligible. See Fig. 4b for an exemplary distribution of pseudo-density.

3.3 Coupled optimization

In Sects. 3.1 and 3.2 we have introduced the design variables of shape and topology into a model defined using the embedding domain discretization technique. In the following, we setup an optimization problem that involves both sets of design variables simultaneously. In a continuum sense, we consider the shape of the embedded body and the pseudo-density of its interior as design variables. In the discretized model, the vector of design variables is given by concatenation of the vector of vertices of the embedded boundary \mathbf{v} and the vector of pseudo densities $\boldsymbol{\rho}$ of the inner elements Ω_{in} of the embedding domain

$$\bar{\mathbf{v}} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\rho} \end{bmatrix}. \tag{21}$$

3.3.1 Optimization workflow

The coupling of topology and shape optimization manifests itself in two ways: (1) there is a single mechanical model that contains both shape and density variables; (2) each iteration of the augmented Lagrange (AL) subproblem involves both shape and density sensitivity analysis, regularization and update. However, we cannot state that this is a fully coupled approach, since the update of shape and density is realized in separate line search steps (although in the same iteration). That means, the density and shape variables are treated in a different manner and they could not be randomly intermixed. In Algorithm 1, we show the workflow for the coupled optimization.

Algorithm 1: Coupled optimization workflow

```

initialization;
while true do
  while true do
    compute AL value;
    shape → compute AL sensitivity;
    density → compute AL sensitivity;
    density → filtering;
    if AL subproblem converged then
      break;
    shape → line search and regularization;
    track densities;
    density → line search;
    if open topological voids present then
      shape projection;
    if closed topological voids present then
      shape generation;
    if shape intersection present then
      shape merging;
    if coarse shape elements present then
      adaptive shape refinement;
  if optimization converged then
    break;
  update AL parameters;
  
```

The workflow is modified to account for the coupling between topology and shape optimization. Tracking of density variables is a consequence of changing the number of discrete density variables. Shape projection deals with open void regions, that is, zero or close to zero density regions adjacent to the shape. Shape generation introduces shape elements in place of closed void regions (topological holes). Furthermore, adaptive shape refinement is implemented to refine those shape elements, that elongate or create sharp corners as a consequence of design update.

In the following sections, we address these steps and the aspects related to coupling of shape and topology optimization and discuss both the profits and the limitations related to them.

3.3.2 Tracking of density variables

Update of the shape might result in truncated (shape contraction) or newly introduced (shape expansion) discrete density variables. Most of the time, due to the volume fraction constraint, the variable shape is contracted and, as a consequence, density design variables are eliminated. While this occurrence is trivial to handle, a special treatment has to

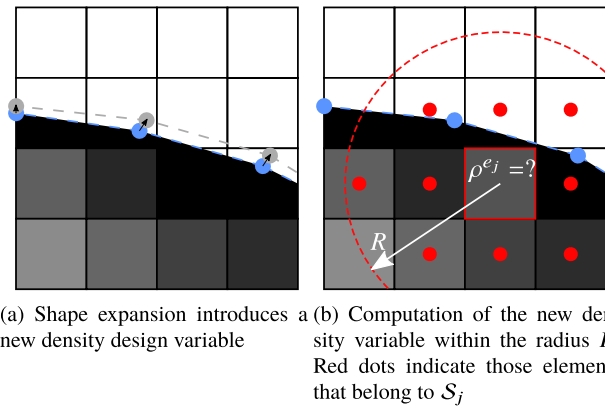


Fig. 5 Tracking of the density design variables. Initial density of the newly introduced density design variable is computed based on the density values of the neighbouring elements within radius R

be introduced when the shape expands and introduces new inner elements.

At this point, we would like to stress that the varying number of density design variables is related purely to the discretization. In a continuum formulation, understanding of what happens with the pseudo-density field as the shape expands (or contracts) is given more freedom. On the one hand, we might say that the pseudo-density field is being stretched since the amount of matter within the embedded body remains the same. On the other hand, the matter might be added or subtracted but no stretching of pseudo-density field takes place. To the specifics of the embedding domain discretization, in particular, the use of a structured mesh, only the second interpretation is suitable.

In order not to over-complicate the method, we introduce a simple approach based on the filtering schemes of Bruns and Tortorelli (2001). In our approach, an initial density of a newly introduced design region \bar{x} is computed by

$$\rho(\bar{x}) = \int_{\mathcal{N}_R(\bar{x}) \cap \mathcal{B}} f(\bar{x} - \mathbf{x}) \rho(\mathbf{x}) \, dV, \tag{22}$$

where $\mathcal{N}_R(\bar{x})$ is a neighbourhood region of \bar{x} within a radius R . In a discrete variant, for a new, j th finite element, its density is computed by

$$\tilde{\rho}^e = \frac{\sum_{e \in \mathcal{S}_j} \rho^e w(\mathbf{x}^e) v^e}{\sum_{e \in \mathcal{S}_j} w(\mathbf{x}^e) v^e}, \tag{23}$$

where \mathcal{S}_j is a set of support elements of the j th element, i.e. $\mathcal{S}_j = \{e \mid e \in \mathcal{N}(e_j) \text{ and } e \neq e_j \text{ and } e \notin \Omega_{\text{out}}\}$, see Fig. 5. The boundary elements are purposely included in \mathcal{S}_j and we use $\rho^e = 1$ for them. In this manner, we avoid a special case scenario when $\mathcal{S}_j \cap \Omega_{\text{in}} = \emptyset$ and the j th element is surrounded only by the boundary elements instead. By making sure that

$\mathcal{S}_j \cap \Omega_{\text{bnd}} \neq \emptyset$, the starting density of the j th will be non-zero and, moreover, it would ensure a smooth density distribution in the proximity of the boundary. To maintain simplicity, the initial sensitivities of the new elements are computed in a similar fashion, i.e. utilizing the filtering scheme without density weighting

$$\frac{\partial \tilde{\mathcal{F}}_j}{\partial \rho} = \frac{\sum_{e \in \mathcal{S}_j} \frac{\partial \mathcal{F}^{e_i}}{\partial \rho} w(\mathbf{x}^e)}{\sum_{e \in \mathcal{S}_j} w(\mathbf{x}^e)}, \tag{24}$$

as proposed by Sigmund (2001b).

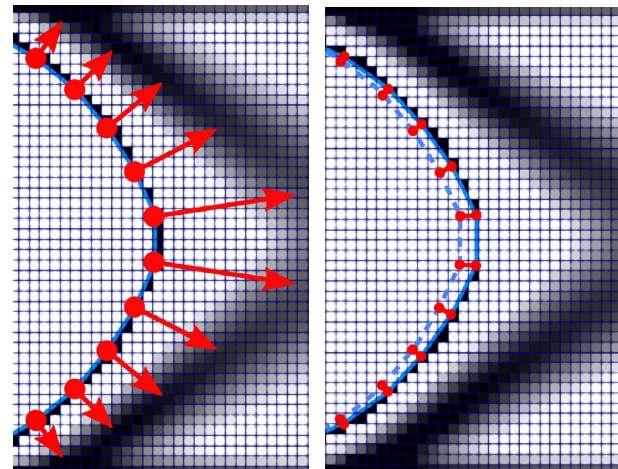
3.3.3 Shape projection

The shape update is preferably limited to small step sizes in order to avoid unwanted distortions. On the contrary, the topology undergoes large design changes, especially at the early stage, as it evolves from an evenly distributed pseudo-density to a recognizable skeleton of topological members. We observed that in a coupled optimization this phenomenon leads to creation of, what we call, open topological voids. That is, when the topology of a structure becomes clearly defined, regions of zero pseudo-density emerge, that are directly adjacent to the shape. Practice shows that these open topological voids eventually disappear as the shape approaches the closest topological feature. However, this might require a large number of iterations. Hence, in order to improve the robustness of our method, we introduce a shape projection step which aims to speed up the elimination of open topological voids.

In Fig. 6a, a raw projection vector is depicted that serves as a basis for the shape projection step. The raw projection vector defines the displacement of shape vertices along the normal to the shape direction, which is needed in order to eliminate the open topological void in one step. Our experience shows that the use of a raw projection vector is not robust enough to deal with certain, difficult scenarios. For example, it happens that the normal directions of neighbouring shape vertices differ to a level, that during projection the shape vertices interchange their positions. We resolved this issue by utilizing the regularization scheme that is primarily used for shape sensitivity (see the Appendix in Sect. 1). As a result, we obtain a smooth projection vector that fixes the issue of interchanging vertices. Furthermore, we introduce a relaxation factor θ to scale down the magnitude of the projection vector in order to limit its effect onto the property of the descent direction and to prevent the shape from an eventual penetration of the topological features.

The regularized and relaxed projection vector is given by

$$\tilde{\mathbf{p}} = \theta [\alpha [\mathbf{K}^{-1} \mathbf{p}] + [1 - \alpha] \mathbf{v}_N], \tag{25}$$



(a) Open topological void and a raw (b) Regularized and relaxed shape projection vector (iteration n) projection vector (iteration $n + 1$)

Fig. 6 Shape projection

where α is a weighting parameter, \mathbf{v}_N is a nodal averaging vector, \mathbf{p} is the raw projection vector, $\mathbf{K}^{-1} \mathbf{p}$ is a solution of the traction method BVP (for more details on these quantities, see Sect. 1). In Fig. 6b, a regularized and relaxed (with $\theta = 0.2$) projection vector is shown. Simulations show, that the relaxation vector works best within the range $\theta \in [0.2, 0.5]$.

We remark that this regularization procedure is not introduced specially for the shape projection, rather we directly make use of the already available regularization for the shape sensitivities. From the implementation point of view, the only additional work related to a shape projection step is the computation of the raw projection vector.

The shape projection step is purely heuristic, therefore, the property of descent direction of the total shape update is not guaranteed anymore. Practice shows, however, that the addition of the regularized and relaxed projection step negligibly affects the final design.

3.3.4 Shape generation

As the topology optimization converges, recognizable hole-like features of topological members evolve. We call these regions closed topological voids, as opposed to open topological voids defined in Sect. 3.3.3. The key difference between these two types of topological voids is that closed ones are not adjacent to the shape boundary. To obtain crisp boundaries in place of the closed topological voids, we add a framework to introduce an additional, variable shape enclosing these regions. These newly introduced shape design variables will further undergo shape optimization to generate a smooth geometry.

The first step of shape generation is to identify the clusters of void cells. To determine these clusters, we do a neighbourhood search for each cell that is classified as a member of a closed topological void. For this classification, we choose all cells which pseudo-density is less than half solid, that is $\rho < 0.5$. This way, the newly introduced shape can ideally be positioned in the middle of the grey transition zone from solid feature to void hole. All the connected cells with $\rho < 0.5$ represent a cluster. In Fig. 7a these clusters are represented in different colours.

The second step is to establish a convex hull that encloses these clusters. To achieve this, we use a generic convex hull generation algorithm to develop a polygon that encloses a cluster. Our implementation is based on the algorithm presented in Cormen et al. (2009). Once the enclosing polygons are determined, we append them to the existing boundary of the embedded body. The updated shape is shown in Fig. 7b.

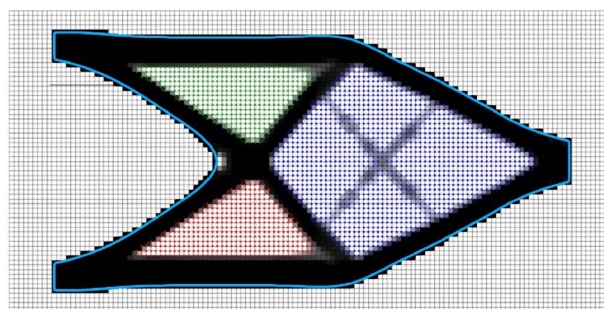
To prevent the generation of shape in place of closed topological voids too early, we introduce a control parameter that delays the shape generation until a desired "crispness" of the topological design is obtained. We measure the "crispness" by the percentage of grey cells remaining in the structure. We define a cell to be grey if its pseudo-density is in the range $\rho \in [0.01, 0.99]$. When the participation of the grey cells in all of the inside cells Ω_{in} of the current iteration falls below a user-defined threshold

$$\frac{n_{\text{grey}}}{|\Omega_{in}|} < \theta_{\text{grey}}, \quad (26)$$

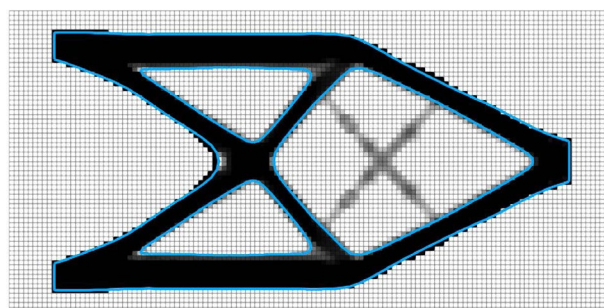
which we choose from the range $\theta_{\text{grey}} \in [0.01, 0.25]$, the shape generation function is allowed to be invoked in any iteration.

Intersecting holes The selection of the grey cell threshold for the shape generation step is based on experience. The general rule is the lower the value for the grey cell threshold the better, because the topological features are then easier to recognize. User might, however, prefer to invoke the shape generation step as early as possible and, therefore, a greater value of the grey cell threshold will be selected. In such case, the shapes might be generated in place of the topological voids that are not fully formed. This sometimes leads to a scenario in which two newly introduced shapes attempt to combine into one, leading to an intersection of shapes, as depicted in Fig. 8a. In this scenario, the convex hull algorithm is reinvoked to regenerate a single shape, see Fig. 8b.

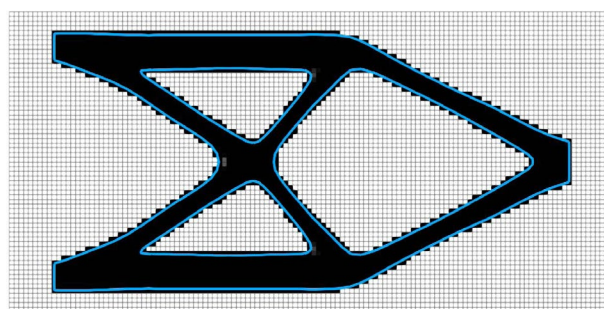
Continuity of the optimization routine Any design change not based on a response gradient information with respect to the design variables introduces a certain amount of discontinuity to the optimization routine. That is particularly pronounced in the hole insertion schemes, in which



(a) This figure shows a pseudo density distribution together with a variable shape (blue line) in n th iteration of the optimization. It represents a design state, for which the percentage of the grey cells fell below $\theta_{\text{grey}} = 0.2$. At this point, the clusters of closed topological voids were found and here depicted with colors: green, red and blue.



(b) Here, still the design state of n th iteration is shown. In place of the clusters shown in Fig. 7a, additional, variable shapes are introduced with help of the convex hull algorithm.



(c) In $n + 1$ th iteration, it can be seen that these cells that were fully contained with the closed topological voids are truncated and assigned to Ω_{out} . Those cells, that are cut but the new, variable shape, now belong to Ω_{bnd} and, thus, integration point oversampling rule applies to them (see section 2.3).

Fig. 7 Three steps in shape generation scheme. Note that the boundary elements Ω_{bnd} are depicted with full black colour due to limitations of the visualization tool used

a solid, spherical region is replaced with a void region or a patch of finite elements is removed with no allowance for intermediate states. On the contrary, the aim of the shape generation step is *not* to introduce a new hole but rather to modify the discretization in a region that is already void, in order to allow for shape sensitivity to control the design. The main requirement for the shape generation step is to alter the design state in a minimal

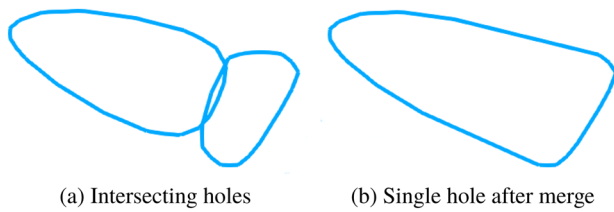


Fig. 8 Merging of two intersecting holes using the convex hull algorithm. An example taken from the cantilever optimization

amount. In other words, it is a quasi-void-to-void transition rather than solid-to-void transition. Moreover, shape generation does not suffer from the heuristics behind the choice of initial geometry in the hole nucleation schemes, which often assume a spherical form with a certain radius. Hence, we assess that the shape generation scheme exhibits a less abrupt nature as compared to hole insertion schemes. Although we believe that the discontinuity introduced by the shape generation is negligible, a mathematical treatment on this aspect is necessary for a further discussion.

3.3.5 Adaptive shape refinement

Use of a $d - 1$ dimensional mesh makes large shape updates possible. Although we do not have to incorporate morphing algorithms to handle mesh deformation (at least in 2D optimization), we still need to put some effort to maintain regularity of the shape discretization. On the one hand, we utilize the traction method and dual descent smoothing to ensure a smooth, mesh-independent design update. On the other hand, we still need to make sure that the discretization is able to represent the shape accurately. In particular, we tackle two aspects of discretization: elongation of the segments and creation of sharp corners. We introduce two simple criteria, based on which a single shape segment can be refined. The length criterion is given by

$$l(\Gamma_{s_i}) < l_{\text{avg}} c_l, \tag{27}$$

where l_{avg} is the average shape segment size from iteration 0 and c_l is a user-defined criterion for the allowable elongation. Most often, we choose $c_l = 1.5$. The curvature criterion is

$$N_{s_i} \cdot N_{s_j} > c_a, \tag{28}$$

where N_{s_i} is a unit normal vector of the i th shape segment and c_a is the angular criterion for the allowable angle between the normals of two adjacent shape segments. By default, we choose $c_a = 0.9$. See Fig. 9 for a comparison

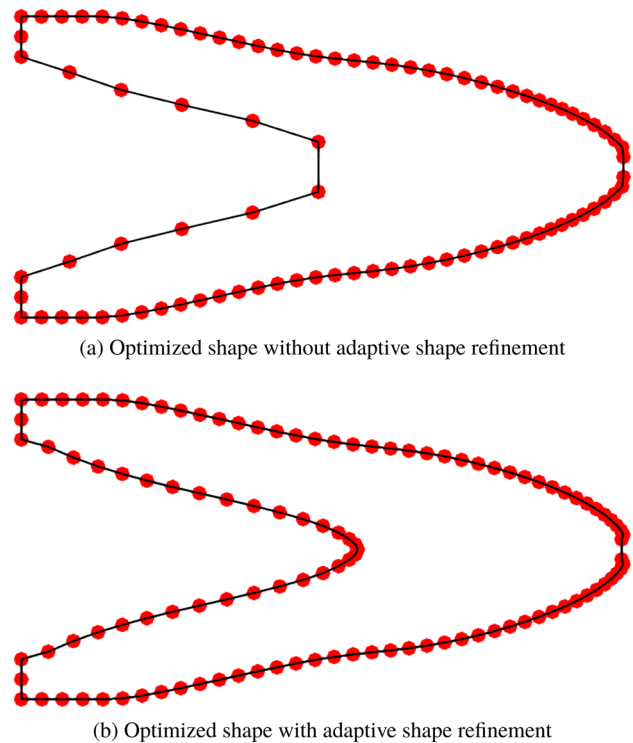


Fig. 9 Adaptive shape refinement in a shape optimization example of a cantilever. Large shape update of the left hand side wall caused the shape segments to elongate significantly and form a U-shaped feature with just three shape segments. When the shape adaptive refinement is utilized, the issues with elongated elements and sharp corners are overcome

between shape optimization of the cantilever from Fig. 3a without and with shape adaptive refinement.

4 Results

We verify our method with two most commonly used benchmark examples: the cantilever and the Messerschmitt-Bolkow-Blohm (MBB) beam. Both examples are optimized with varying values of the filter radius for density sensitivities (please refer to the filtering scheme of Bruns and Tortorelli (2001)). We also investigate the influence of the choice of the grey cell threshold for the delay of the shape generation. In particular, we are interested in how does the premature shape generation affects the final design. For the regularization of the shape sensitivities, we established a set of parameters that remains the same for all of the presented examples. We employ a standard setting for the optimization problem, that is minimization of the total compliance with an upper limit constraint on the volume fraction.

Cantilever In Fig. 10 we show a course of cantilever optimization using our method. For this, we choose an

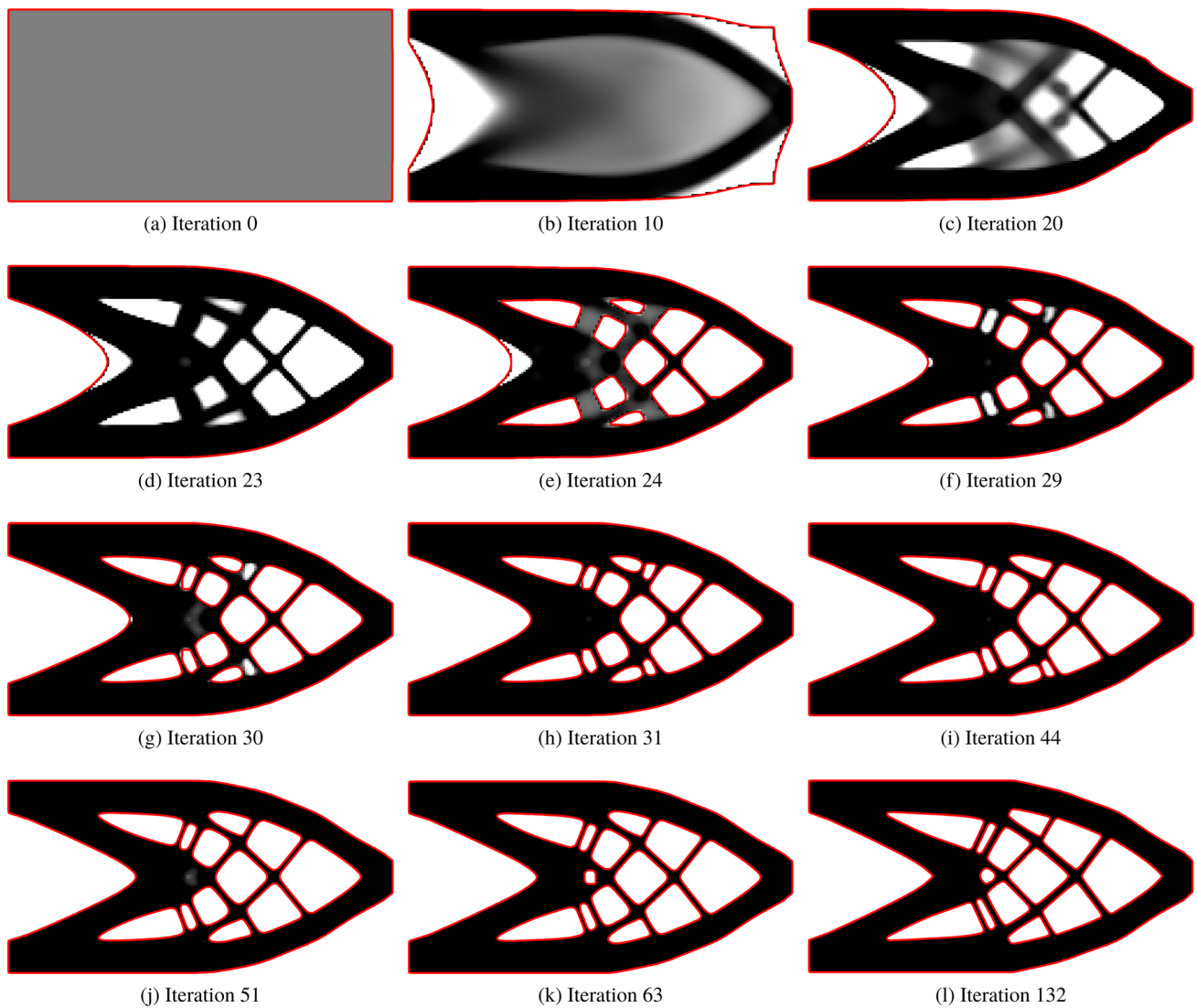


Fig. 10 Stages of cantilever optimization. The variable shape is depicted as red line. (Color figure online)

embedding domain discretization with element size of $h = 0.007$ and a filter radius for density-based sensitivities of $r = 0.02$. The cantilever is constrained with a volume fraction of $V_{\text{frac}} = 0.5$. The BVP setup is the same as in 3a.

In the initial configuration (Fig. 10a) the variable shape is only present on the outside of a filled body (no initial holes). The initial pseudo-density of the inner cells is chosen to be 0.5 so that the volume fraction constraint is only slightly violated¹. In the initial iterations the topology undergoes large changes and open topological voids occur, as depicted in Fig. 10b. At this stage, the shape update is supported by the shape projection scheme. Within the next ten iterations

the open topological voids are almost eliminated (Fig. 10c) and the topological features become recognizable. At this stage, the grey cell participation decreases rapidly until it reaches the threshold value of $\theta_{\text{grey}} = 0.1$ in the 23rd iteration (Fig. 10d). From this point, the shape generation function is allowed to introduce variable shapes in the place of the topological voids. Naturally, with a participation of grey cells below 10% the closed topological voids are recognized, therefore, variable shapes are introduced before the 24th iteration, see Fig. 10e. After invoking the shape generation step for the first time, the shape updates become dominating as compared to the density updates. Nevertheless, the density updates of the coupled optimization are still active and, as shown in Fig. 10f), the topology is further fine-tuned and two more pairs of topological holes are formed. Consequently, the shape generation step is activated in the 30th and

¹ Note that the actual volume fraction is slightly greater than 0.5 because the pseudo-density is not assigned to the boundary cells.

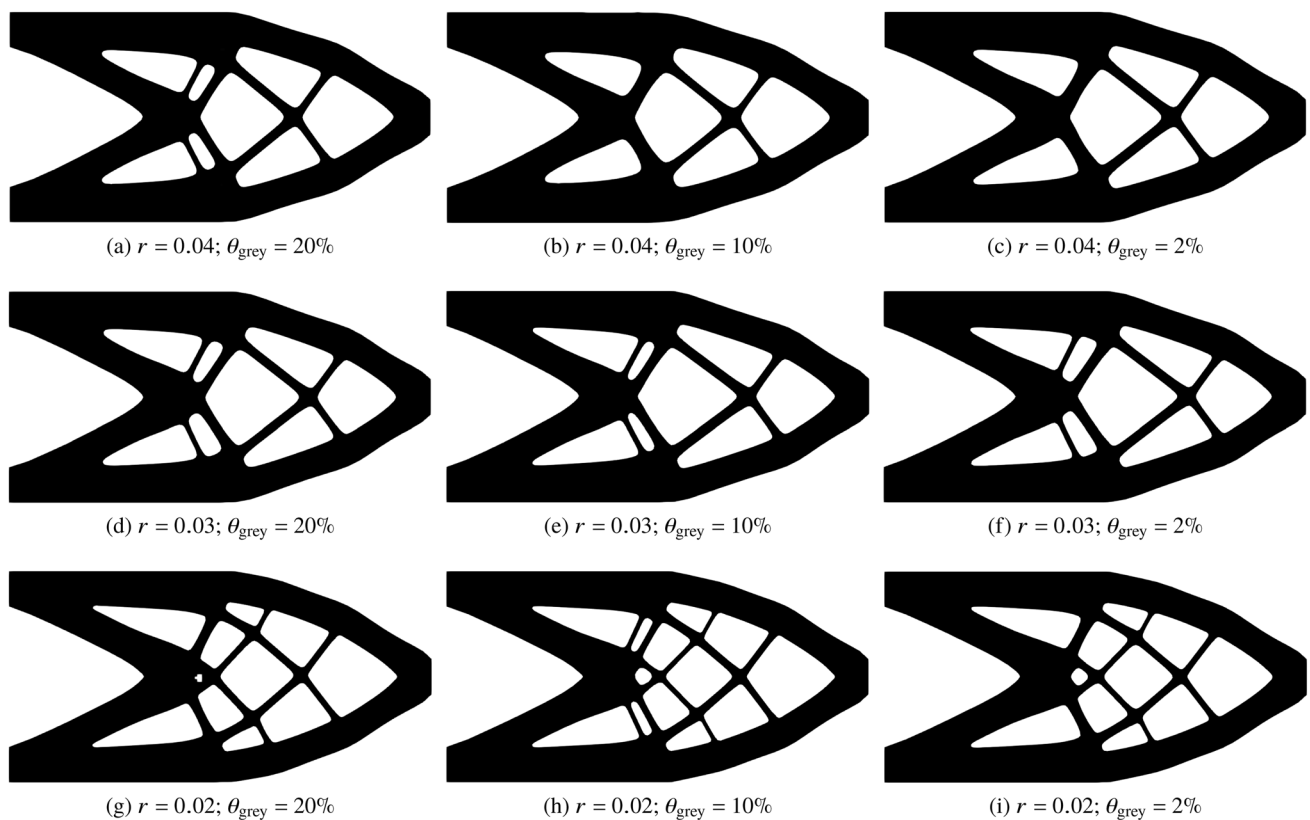


Fig. 11 Optimized designs of the cantilever for varying filter radii and grey cell thresholds

the 31st iterations to complement these holes with variable shapes (Fig. 10g, h). It is worth a notice, that at this point the open topological voids are fully eliminated. Hence, the shape updates are driven exclusively by the optimizer. In the 44th and the 51st iterations, a shape intersection occurred as the optimizer tried to eliminate two topological members (see Fig. 10i, j). To account for this, the shapes were merged to form a single hole. Although the design appears fully black and white, the density variables are still active and are free to react to the shape changes. As a consequence another topological void formed and a new shape was generated in the 63rd iteration, see Fig. 10k. The remainder of the optimization run fine-tunes the shape and converges in the 132nd iteration (Fig. 10l).

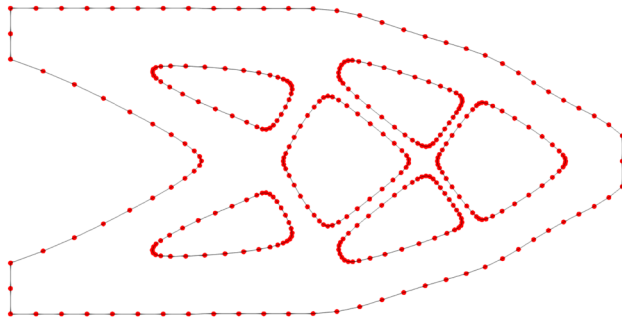
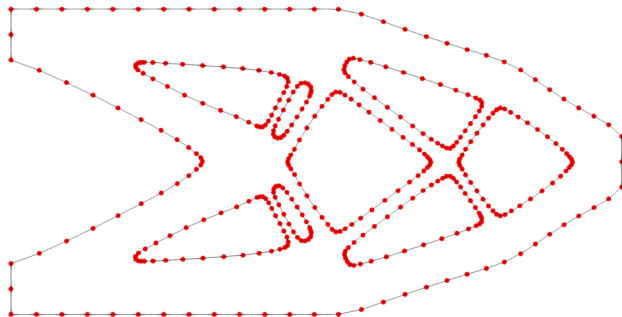
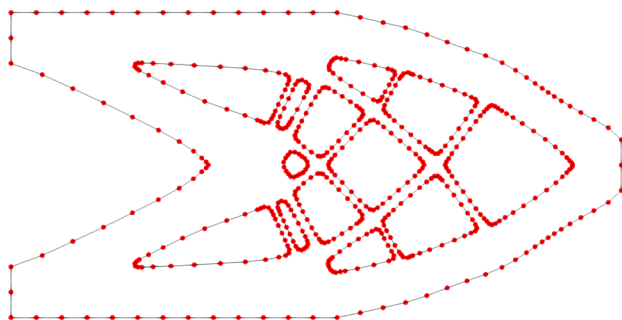
As the next step we examine the complexity of the design by choosing varying filter radii for the density sensitivities and the robustness of the method against a choice of varying grey cells thresholds. In Fig. 11a–i filter radii of $r = \{0.04, 0.03, 0.02\}$ (sorted row-wise) and grey cells thresholds of $\theta_{\text{grey}} = \{20\%, 10\%, 2\%\}$ (sorted column-wise) were chosen. The element size for the embedding domain discretization is $h = 0.007$.

The optimized structures show a mechanically crisp, smooth boundary thanks to the variable shape. The method produces similar outputs for different settings of the grey

cell threshold ranging from $\theta_{\text{grey}} = 20\%$ to $\theta_{\text{grey}} = 2\%$ for all the filter choices. This proves that the user has enough flexibility in the choice of the grey cell threshold. As the filter radius for density-based sensitivities decreases, the complexity of the topological design increases. This complexity can be further increased by choosing smaller filter radii and, consequently, with finer embedding domain mesh. However, we believe that the advantage of our method is that we can achieve finely optimized geometries with relatively coarse discretization of the embedding domain, since the actual geometry is represented only by the boundary of the embedded body. In other words, by using a relatively coarse embedding domain discretization, thin geometrical features can still be represented smoothly as opposed to pure topology optimization, in which a visible staircase-like pattern appears. Moreover, truncation of the outside elements Ω_{out} means a smaller mechanical domain. On the top of that, the size of the problem decreases with the decreasing volume of the design. This is particularly pronounced after the shape generation step, during which the elements within the topological voids are excluded from the BVP. For instance, the initial configuration in the examples from Fig. 11 contains 13944 inner $e \in \Omega_{\text{in}}^h$ and boundary elements $e \in \Omega_{\text{bnd}}^h$ together. The BVPs of the final designs for all the setups

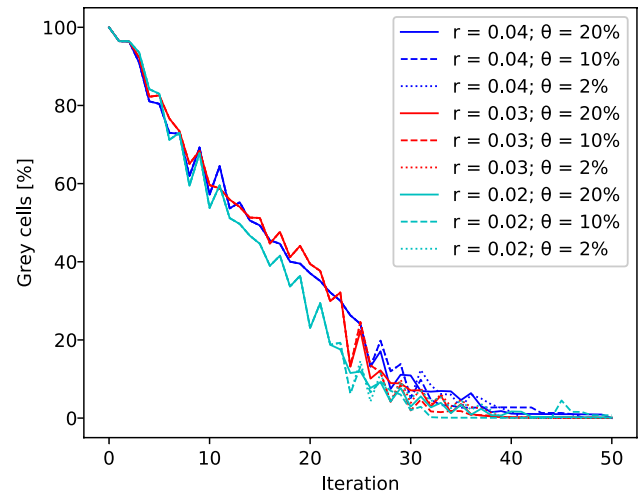
Table 1 Final compliance and grey cell participation for the cantilever

Com. $\times 10^{-6}$	$\theta_{\text{grey}} = 20\%$	$\theta_{\text{grey}} = 10\%$	$\theta_{\text{grey}} = 2\%$
$r = 0.04$	5.278	5.302	5.280
$r = 0.03$	5.274	5.265	5.262
$r = 0.02$	5.265	5.263	5.268

(a) $r = 0.04; \theta_{\text{grey}} = 10\%$ (b) $r = 0.03; \theta_{\text{grey}} = 10\%$ (c) $r = 0.02; \theta_{\text{grey}} = 10\%$ **Fig. 12** Optimized designs of the cantilever for varying filter radii. Distribution of the shape vertices

with $\theta_{\text{grey}} = 10\%$ (see Fig. 11b, e, h) involve 7494, 7568 and 7657 inner and boundary elements together, respectively.

In Fig. 12, the distribution of the shape vertices for the final designs from Fig. 11b, e, h are shown. The usage of the adaptive shape refinement scheme (see Sect. 3.3.5) for the length and curvature control allowed for a precise

**Fig. 13** Participation of grey cells as a function of iteration count for the first 50 iterations

representation of the shape at the intersections of the topological features.

In Fig. 13, the participation of the grey cells in the first 50 iterations is shown for the examples from Fig. 11. One can immediately see that for all the cases the grey cells (almost) fully disappear between the 30th and 40th iteration. This indicates that the topology forms and the problem transforms into a shape-driven optimization. Moreover, since in the final designs only two phases remain in the model: solid material and void, we deduce that the obtained structures are physically meaningful.

To gain an insight into the performance difference between the designs from Fig. 11, the final values of the compliance are shown in Table 1.

The differences in compliance values are exiguous, what suggests that the performance of the design is insensitive to the choice of grey cells threshold.

In the next step, the volume fraction constraint is set to $V_{\text{frac}} = 0.3$. Additionally, we use a relatively coarse mesh with element size of $h = 0.015$ and we test two cases of the grey cell threshold $\theta_{\text{grey}} = 30\%$ and $\theta_{\text{grey}} = 5\%$. The filter radius is set to $r = 0.04$ for both cases. The shape generation steps and the final results are shown in Fig. 14. Moreover, the final results are depicted with shape vertices (Fig. 14c, f). As shown in Fig. 14a, b, d, e, although significantly varying grey cell thresholds were used, the generated shapes in 14b, e are very similar. The final shapes in Fig. 14c, f are almost identical, what confirms that the method deals well with varying setting of the grey cell threshold. On the top of that, the usage of the adaptive shape refinement resulted in a finely rendered geometry.

MBB Beam We continue benchmarking of our method with the MBB beam. In Fig. 15 we show the BVP setup for half of the beam with symmetry constraint on the left hand

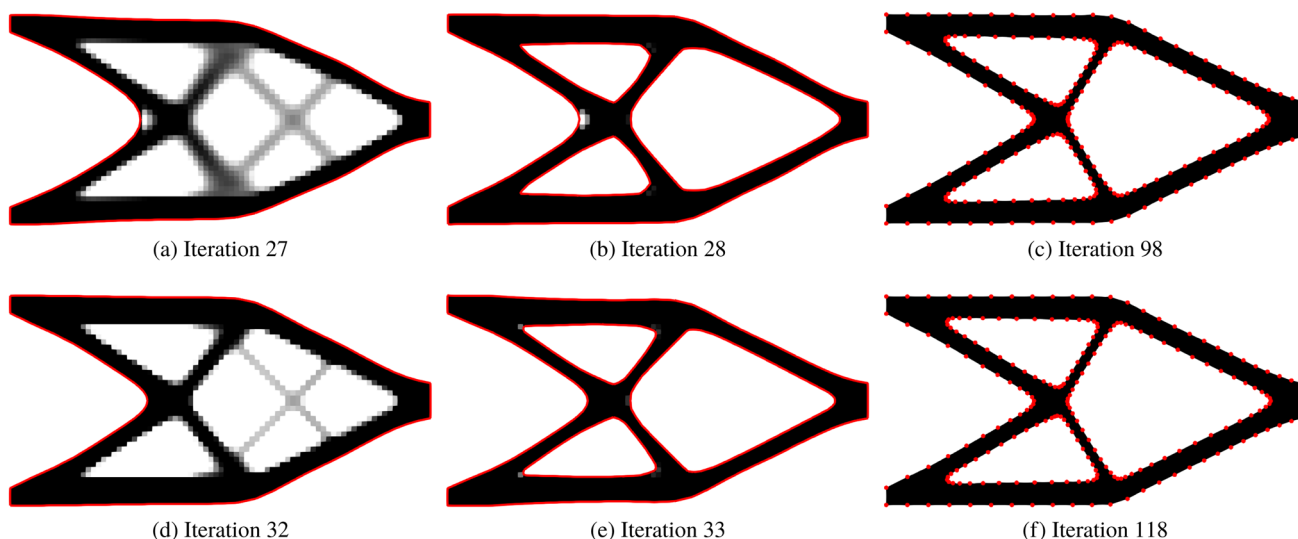


Fig. 14 Optimized cantilever with volume constraint of $V_{frac} = 0.3$, element size of $h = 0.015$ and filter radius of $r = 0.04$

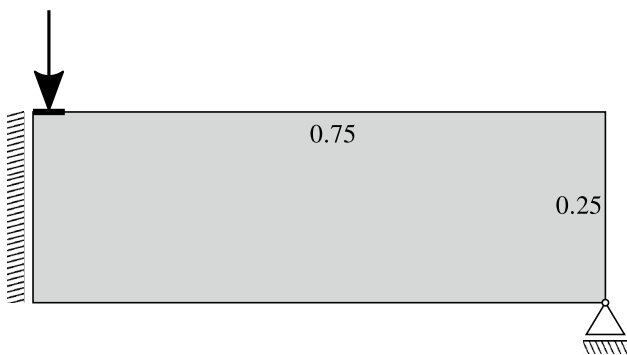


Fig. 15 BVP setup of the MBB beam

edge. The material parameters used are $E = 70 \text{ GPa}$; $\nu = 0.3$; the bending load equals $F = 0.1 \text{ N}$ for an assumed thickness of $t = 1 \text{ mm}$.

The selected element size is $h = 0.005$ for all the presented designs. As demonstrated with the cantilever example, the coupled optimization method is robust enough to produce similar results for different choices of grey cell levels. Hence, this time we restrict the setup of the optimization problems to $\theta_{grey} = 10\%$ only.

In Fig. 16, the design evolution of the MBB beam with filter radius of $r = 0.01$ is shown. In a similar manner to the cantilever design, in the early iterations (Fig. 16b) the topology undergoes large design changes, whereas the shape update is supported by the projection scheme. With the partial elimination of open topological voids the inner skeleton of topological features appear, see Fig. 16c. In 23rd iteration (Fig. 16d), the grey cell participation drops below the threshold value of $\theta_{grey} = 10\%$ and the shape generation step is invoked (Fig. 16e). Further development

of the topology design yields a particular scenario in which an open topological void appears adjacent to a shape hole, see Fig. 16f. Subsequent iterations successfully deal with the elimination of the open topological voids and two additional shapes are introduced (see Fig. 16g, h). The remaining iterations are naturally dominated by the shape update until the convergence is reached (Fig. 16i).

In Fig. 17, final designs of the MBB beam for varying filter radii are depicted. All three configurations shared the same initial step size for the shape update $\tau_x^{init} = 0.015$ and density update $\tau_\rho^{init} = 0.2$. The only varying quantity is the filter radius for density sensitivities. The obtained topologies naturally become more complex with decreasing filter radius. In the cases from Fig. 17a,b the end results are not fully black and white. This is caused by the Lagrange updates which caused a sudden volume drop in the design. The Lagrange updates affected those regions, which for smaller values of the volume constraint would most probably become void.

Figure 18 depicts the distribution of the vertices for the designs from Fig. 17. As in the case of the cantilever, highly curved regions are rendered with finely discretized shape to avoid sharp corners.

The final value of compliance and grey cells participation are listed in Table 2.

One can see that with the decreasing filter radius the compliance of the final design also slightly decreases. The grey cells participation values are non-zero for the cases with $r = 0.03$ and $r = 0.02$. Nevertheless, we consider the amount of the remaining grey cells as negligible. Therefore, we can safely state that the optimized structures are physically realistic.

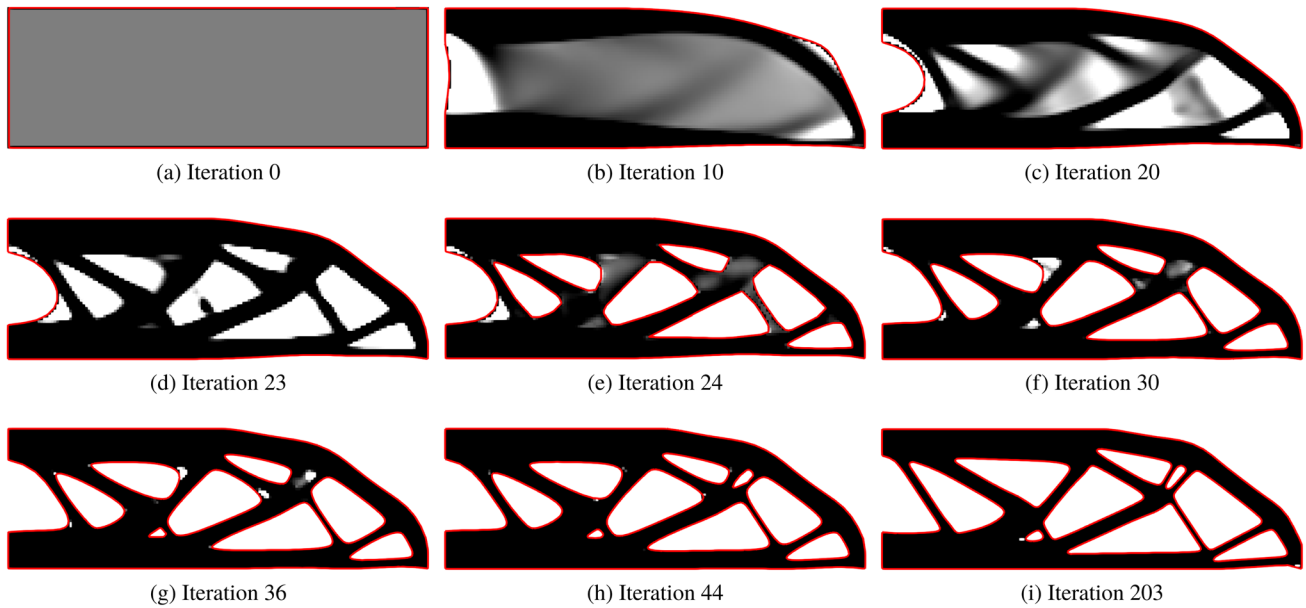


Fig. 16 Stages of MBB optimization. The variable shape is depicted as red line. (Color figure online)

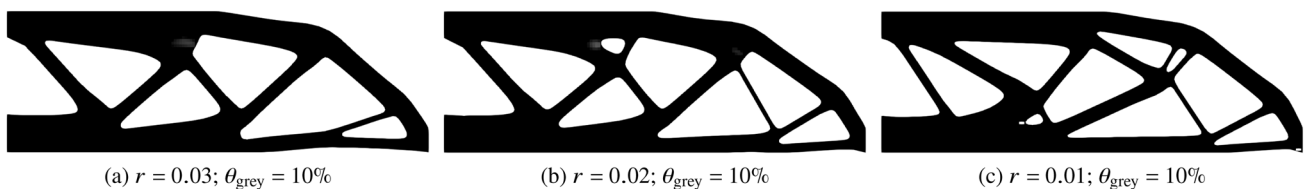


Fig. 17 Optimized designs of the MBB beam for varying filter radii

The evolution of the participation of grey cells is shown in Fig. 19. In all of cases the percentage drops to values close to zero between the 30th and 35th iterations. At this stage, the coupled optimization becomes dominated by the shape updates.

5 Conclusions

In this contribution, we propose a method that is an alternative to the traditional, sequential approach to topology and shape optimization. By exploiting the specifics of the embedding domain discretization, we couple density-based topology optimization and node-based shape optimization. An embedding domain, which is discretized with a structured finite element mesh, is assigned a pseudo-density field, whereas the boundary of the embedded body serves as a variable shape. In this manner, we setup a single optimization problem that operates on a hybrid design space that is comprised of two sets of design variables.

Our goal was to demonstrate that coupled topology and shape optimization might offer practical advantages. Considering the current state of our work, we discuss the characteristics of the method as compared to sequential and other combinations of topology and shape optimization.

First and foremost, the usage of the same computational domain for shape and topology optimization is the key feature. We skip any transition steps that are necessary in transforming the topological results into a shape mesh, which is general practice in sequential topology and shape optimization.

Moreover, shape variation is given a larger flexibility as compared to the standard, node-based shape optimization due to the usage of embedded boundary. As a consequence, the shape sensitivity analysis is numerically less expensive and the regularization is much easier. Moreover, adaptive shape refinement becomes a relatively simple task as, for 2D problems, it just requires vertex insertion in the segment centre. Besides that, the mechanical model relies upon the structured mesh that is made of quadrilateral elements (in

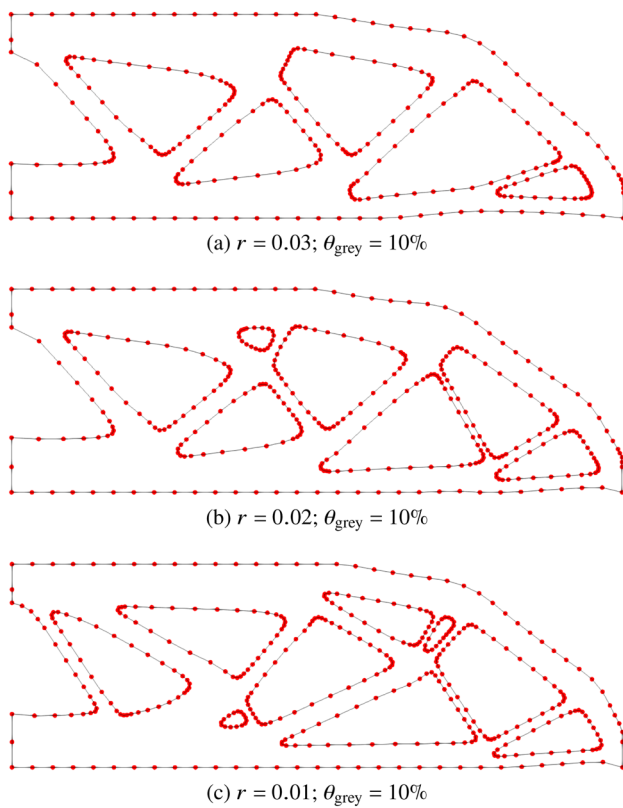


Fig. 18 Optimized designs of the MBB beam for varying filter radii. Distribution of the shape vertices

Table 2 Final compliance and grey cell participation for the MBB beam

Setting (r/θ_{grey})	Compliance	Grey cells (%)
0.03/10%	1.543×10^{-5}	0.563
0.02/10%	1.532×10^{-5}	0.536
0.01/10%	1.515×10^{-5}	0.000

case of 2D), whereas in shape optimization usually triangular elements are used, which offer a lower accuracy.

A common practice in a combined shape and topology optimization routines is to employ a hole insertion scheme. This approach shows a few inherent drawbacks, i.e. its abrupt and discontinuous nature and a limited flexibility in performing topological changes. In our work, we mitigate these issues by enabling a full topological flexibility by introducing a pseudo-density field and the shape generation scheme, which inserts a variable shape in place of already present topological holes.

Furthermore, the fact that the embedding domain must be larger than the design space means one can easily get rid of the bounding box constraints and allow for a geometrically unrestricted setup. Furthermore, since the design

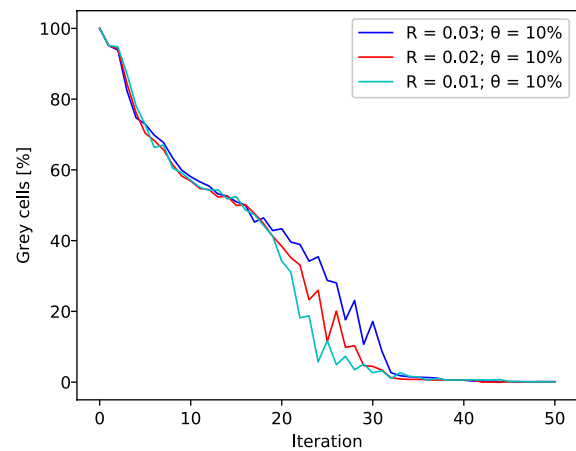


Fig. 19 Participation of grey cells as a function of iteration count for the first 50 iterations

is ultimately defined by the boundary of the embedded body, one can use a coarser discretization of the embedding domain and still achieve geometrically refined designs.

A typical phenomenon in our method is the constant interaction of the topology and evolving shape. Although in early iterations the topology changes are dominant and the shape changes are of low influence, in later stages the shape changes take over to position, form and smoothen the features. We are aware that this interaction might in fact be more of an issue than advantage. Practice shows, however, that the interaction is robust enough in a sense that the topological features manage to emerge in an early stage of optimization despite the shape changes. On the other hand, creation of the open topological voids takes place as a result of shape updates being more delicate than the topology changes. Although it is an unwanted phenomenon, the open topological voids are eventually eliminated within the course of optimization. Moreover, we enhance this elimination as described in Sect. 3.3.3.

The key drawback of our method is its inherent complexity and, consequently, less freedom in the selection of an optimization algorithm. Besides the augmented Lagrange formulation, which is utilized in this work, no other approach was tested.

A Continuum approach to shape sensitivity

We perform shape sensitivity analysis using the continuum approach, as described in Choi and Kim (2004). We utilize the material derivative concept, which we do not introduce here, but refer interested readers to Arora (1993); Choi and Kim (2004). In the following, we define the compliance functional as

$$\mathcal{F}^{\text{com}} = \int_B \mathbf{b} \cdot \mathbf{u} \, dV + \int_{\Gamma^N} \bar{\mathbf{t}} \cdot \mathbf{u} \, dA. \quad (\text{A.29})$$

We exploit the adjoint method and augment the compliance functional by a weak form of equilibrium equation, in which the variations of the state fields have been replaced by the adjoint variables:

$$\mathcal{L} = \mathcal{F}^{\text{com}} + \mathcal{W}^a = \int_B \bar{G} \, dV + \int_{\Gamma^D} \bar{g} \, dA + \int_{\Gamma^N} \bar{h} \, dA, \quad (\text{A.30})$$

$$\begin{aligned} \mathcal{W}^a = & - \int_B \boldsymbol{\sigma} : \boldsymbol{\epsilon}^a \, dV + \int_B \mathbf{b} \cdot \mathbf{u}^a \, dV \\ & + \int_{\Gamma^N} \bar{\mathbf{t}} \cdot \mathbf{u}^a \, dA + \int_{\Gamma^D} \mathbf{t} \cdot \bar{\mathbf{u}}^a = \mathbf{0}. \end{aligned} \quad (\text{A.31})$$

By means of variational calculus it was proven that $\delta \mathcal{W}^a = 0$ (Arora and Cardoso 1992). Thus, we obtain

$$\dot{\mathcal{F}}^{\text{com}} = \dot{\mathcal{L}} - \dot{\mathcal{W}}^a = \dot{\mathcal{L}}. \quad (\text{A.32})$$

Next, the boundary integral method (Arora 1993) is employed to express the material derivative of the augmented functional in the following form

$$\begin{aligned} \dot{\mathcal{F}}^{\text{com}} = & \int_B \bar{G}' \, dV + \int_{\Gamma} \bar{G} [\boldsymbol{\theta} \cdot \mathbf{n}] \, dV \\ & + \int_{\Gamma^D} \bar{g}' \, dA + \int_{\Gamma^D} [\nabla \bar{g} \cdot \mathbf{n} + \bar{g} H] [\boldsymbol{\theta} \cdot \mathbf{n}] \, dA \\ & + \int_{\Gamma^N} \bar{h}' \, dA + \int_{\Gamma^N} [\nabla \bar{h} \cdot \mathbf{n} + \bar{h} H] [\boldsymbol{\theta} \cdot \mathbf{n}] \, dA. \end{aligned} \quad (\text{A.33})$$

We calculate the partial derivatives of the integrands \bar{G} , \bar{g} and \bar{h} as follows:

$$\begin{aligned} \bar{G}' &= \partial_u G \cdot \mathbf{u}' - \boldsymbol{\sigma}' : \boldsymbol{\epsilon}^a - \boldsymbol{\sigma} : [\boldsymbol{\epsilon}^a]' + \mathbf{b}' \cdot \mathbf{u}^a + \mathbf{b} \cdot [\mathbf{u}^a]', \\ \bar{g}' &= \mathbf{t}' \cdot \bar{\mathbf{u}}^a + \mathbf{t} \cdot [\bar{\mathbf{u}}^a]', \\ \bar{h}' &= \partial_t h \cdot \bar{\mathbf{t}}' + \partial_u h \cdot \mathbf{u}' + \bar{\mathbf{t}}' \cdot \mathbf{u}^a + \bar{\mathbf{t}} \cdot [\mathbf{u}^a]'. \end{aligned} \quad (\text{A.34})$$

Finally, we introduce A.29 and A.34 into A.33 to arrive at a full expression for the material derivative of the augmented compliance functional.

$$\begin{aligned} \dot{\mathcal{F}}^{\text{com}} = & \int_B \mathbf{b}' \cdot \mathbf{u}^a \, dV \\ & + \int_{\Gamma} [\mathbf{b} \cdot \mathbf{u} - \boldsymbol{\sigma} : \boldsymbol{\epsilon}^a + \mathbf{b} \cdot \mathbf{u}^a] [\boldsymbol{\theta} \cdot \mathbf{n}] \, dA \\ & + \int_{\Gamma^D} \mathbf{t}' \cdot \bar{\mathbf{u}}^a \, dA \\ & + \int_{\Gamma^D} [\nabla [\mathbf{t} \cdot \bar{\mathbf{u}}^a] \cdot \mathbf{n} + [\mathbf{t} \cdot \bar{\mathbf{u}}^a] H] [\boldsymbol{\theta} \cdot \mathbf{n}] \, dA \\ & + \int_{\Gamma^N} \partial_t h \cdot \bar{\mathbf{t}}' + \bar{\mathbf{t}}' \cdot \mathbf{u}^a \, dA \\ & + \int_{\Gamma^N} [\nabla [\bar{\mathbf{t}} \cdot \mathbf{u} + \bar{\mathbf{t}} \cdot \mathbf{u}^a] \cdot \mathbf{n} \\ & + [\bar{\mathbf{t}} \cdot \mathbf{u} + \bar{\mathbf{t}} \cdot \mathbf{u}^a] H] [\boldsymbol{\theta} \cdot \mathbf{n}] \, dA \\ & + \left\{ - \int_B \boldsymbol{\sigma} : [\boldsymbol{\epsilon}^a]' \, dV + \int_B \mathbf{b} \cdot [\mathbf{u}^a]' \, dV \right. \\ & + \int_{\Gamma^N} \bar{\mathbf{t}} \cdot [\mathbf{u}^a]' \, dA + \int_{\Gamma^D} \mathbf{t} \cdot [\bar{\mathbf{u}}^a]' \, dA \left. \right\} \\ & + \left\{ \int_B \partial_u G \cdot \mathbf{u}' - \boldsymbol{\sigma}' : \boldsymbol{\epsilon}^a \, dA + \int_{\Gamma^N} \partial_u h \cdot \mathbf{u}' \, dA \right\} \end{aligned} \quad (\text{A.35})$$

By exploiting the variational principle for design sensitivity analysis, it is required that the variation of the augmented functional w.r.t. the primary and adjoint state fields vanishes. Hence, the sum of the two expressions within curly braces vanishes. Moreover, the first expression within curly braces represents the weak form of equilibrium for the primary structure, in which the explicit derivative of the adjoint state field $[\mathbf{u}^a]'$ serves as a kinematically admissible test function and $[\boldsymbol{\epsilon}^a]'$ is a compatible strain tensor. We may express the first expression in the curly braces as

$$\mathcal{W}(\mathbf{u}, [\mathbf{u}^a]') = 0, \quad (\text{A.36})$$

and, consequently, we obtain

$$\int_B \partial_u G \cdot \mathbf{u}' - \boldsymbol{\sigma}' : \boldsymbol{\epsilon}^a \, dA + \int_{\Gamma^N} \partial_u h \cdot \mathbf{u}' \, dA = 0. \quad (\text{A.37})$$

Now, we utilize the symmetry of the Cauchy stress tensor and major symmetry of the tangent tensor

$$\boldsymbol{\sigma}' : \boldsymbol{\epsilon}^a = \boldsymbol{\epsilon}' : \mathbb{E} : \boldsymbol{\epsilon}^a = \boldsymbol{\epsilon}' : \boldsymbol{\sigma}^a = \boldsymbol{\sigma}^a : \boldsymbol{\epsilon}', \quad (\text{A.38})$$

to obtain the weak form of equilibrium for the adjoint structure

$$\begin{aligned} \mathcal{W}(\mathbf{u}^a, \mathbf{u}') = & \int_B \partial_u G \cdot \mathbf{u}' - \boldsymbol{\sigma}^a : \boldsymbol{\epsilon}' \, dA \\ & + \int_{\Gamma^N} \partial_u h \cdot \mathbf{u}' \, dA = 0, \end{aligned} \quad (\text{A.39})$$

in which \mathbf{u}' is a kinematically admissible test function and $\boldsymbol{\epsilon}'$ a compatible strain tensor. By evaluating the boundary conditions:

$$\begin{aligned} \mathbf{b}^a &= \partial_u G = \mathbf{b} \text{ in } \mathcal{B}, \\ \bar{\mathbf{t}}^a &= \partial_u h = \mathbf{t} \text{ on } \Gamma^N, \\ \bar{\mathbf{u}}^a &= \partial_u g = \mathbf{0} \text{ on } \Gamma^D, \end{aligned} \tag{A.40}$$

we realize that the loading of the adjoint structure is the same as of the primary structure. Thus, the problem is self-adjoint and $\mathbf{u} = \mathbf{u}^a$. Under the assumption that the explicit derivatives of the external loads are zero, i.e. $\mathbf{b}' = \mathbf{0}$, $\bar{\mathbf{t}}' = \mathbf{0}$ and by using the fact that $\bar{\mathbf{u}}^a = \mathbf{0}$ we obtain the final expression for the continuum shape sensitivity of the compliance functional

$$\begin{aligned} \dot{\mathcal{J}}^{\text{com}} &= \int_{\Gamma} [2[\mathbf{b} \cdot \mathbf{u}] - \boldsymbol{\sigma} : \boldsymbol{\epsilon}][\boldsymbol{\theta} \cdot \mathbf{n}] \, dA \\ &+ \int_{\Gamma^N} [2\nabla[\bar{\mathbf{t}} \cdot \mathbf{u}] \cdot \mathbf{n} + 2[\bar{\mathbf{t}} \cdot \mathbf{u}]H][\boldsymbol{\theta} \cdot \mathbf{n}] \, dA. \end{aligned} \tag{A.41}$$

If the body force \mathbf{b} is zero and the region with traction force is not a part of the design space, then the sensitivity expression reduces to a very simple formula

$$\dot{\mathcal{J}}^{\text{com}} = - \int_{\Gamma} \boldsymbol{\sigma} : \boldsymbol{\epsilon}[\boldsymbol{\theta} \cdot \mathbf{n}] \, dA. \tag{A.42}$$

The sensitivity of the volume functional, given by

$$\mathcal{F}^{\text{vol}} = \int_{\mathcal{B}} 1 \, dV, \tag{A.43}$$

is obtained by direct differentiation. Thus, it reads

$$\dot{\mathcal{F}}^{\text{vol}} = \int_{\Gamma} 1[\boldsymbol{\theta} \cdot \mathbf{n}] \, dA. \tag{A.44}$$

The derived expressions for the continuum sensitivity are then discretized to obtain the nodal contributions in the form

$$\dot{\mathcal{J}}^I = \mathbf{g}_n^I \cdot \boldsymbol{\theta}^I, \tag{A.45}$$

where \mathbf{g}_n^I is the sought-after nodal shape sensitivity.

B Shape design update

B.1 Modified traction method

For the specifics of the embedding domain discretization we adapt the traction method, which was originally introduced in Azegami and Wu (1996); Azegami and Takeuchi (2006). Briefly explained, the traction method is based upon solving an auxiliary BVP, in which the raw shape sensitivity is employed in form of external, nodal traction forces. The

solution of such BVP yields a smooth, mesh-independent design update vector. The original traction method (using Robin conditions) relies upon an elasticity energy in form of a body integral. Our adaptation concerns (1) the integration domain, that is, all the energies are defined by means of boundary integrals and (2) the substitution of the elastic energy with a simple spring and Laplacian smoothing energies. In this way, the auxiliary BVP can be directly defined over the embedded boundary in which we treat the shape segments as $d-1$ dimensional finite elements.

The energy form of our version of the traction method is given by

$$\Pi_t = \Pi_{\text{int}} + \beta [\Pi_{\text{ext}} + \mathcal{P}] \rightarrow \text{Min} \tag{B.46}$$

where Π_{ext} is the energy of the external forces in form of the raw sensitivities, \mathcal{P} is the penalty functional, which is responsible for the bounding box constraints and β is the scaling factor which ensures that the solution of the traction method BVP yields an exact design update vector.

The internal energy in our adapted formulation is defined as a sum of smeared spring and Laplacian smoothing energies as follows

$$\begin{aligned} \Pi_{\text{int}} &= \Pi_{\text{sp}} + \Pi_{\text{sm}} \\ &= \int_{\Gamma} \frac{c_{sp,N}}{2} [\mathbf{u}_N]^2 \, dA \\ &+ \int_{\Gamma} \frac{c_{sp,T}}{2} [\mathbf{u}_T]^2 \, dA \\ &+ \int_{\Gamma} \frac{c_{sm}}{2} \nabla \mathbf{u} : \nabla \mathbf{u} \, dA. \end{aligned} \tag{B.47}$$

At this point, we remark that the chosen smoothing procedure is based on the following boundary value problem:

$$\begin{aligned} \Delta \mathbf{u} &= \mathbf{u} \quad \text{on } \Gamma, \\ \mathbf{u} &= \mathbf{0} \quad \text{on } \partial\Gamma', \end{aligned} \tag{B.48}$$

where, as opposed to Poisson’s equation, the BVP solves for a vector field instead of a scalar field. The motivation behind using a vector field is that, besides the normal to shape smoothing, we additionally introduce the tangent to shape smoothing to control the node distribution of the variable shape. The vector Laplacian, in the particular case when a cartesian coordinate system is utilized, is given by

$$\Delta \mathbf{u} = \frac{\partial^2 \mathbf{u}}{\partial x^2} + \frac{\partial^2 \mathbf{u}}{\partial y^2} + \frac{\partial^2 \mathbf{u}}{\partial z^2} = \text{div} [\nabla \mathbf{u}]. \tag{B.49}$$

Moreover, note that the spring energy is split into the normal to shape and tangent to shape contributions. This division is specifically introduced for the discrete variant of the sensitivity. This is because with help of the normal spring we eliminate the mesh dependency of sensitivities, whereas with the tangent spring, in conjunction with the

Laplacian smoothing, we introduce and control the in-plane regularization.

B.2 Geometrical smoothing

While the traction method deals with sensitivity regularization, it does not address the irregularities of the geometry itself, for instance, the ear formation problem. To account for that, we additionally introduce a geometric smoothing strategy, in which the final design update is given as a weighted sum of the regularized design update and a nodal averaging vector in the following form

$$\bar{\mathbf{u}}_{\text{upd}} = \alpha \mathbf{u}_{\text{upd}} + [1 - \alpha] \mathbf{u}_{\text{avg}} \quad (\text{B.50})$$

where $\alpha \in [0, 1]$ is a weighting parameter, $\bar{\mathbf{u}}_{\text{upd}}$ is the final design update vector, \mathbf{u}_{upd} is the regularized design update vector and \mathbf{u}_{avg} is the nodal averaging vector, for the l th node defined as

$$\mathbf{u}_{\text{avg}}^l = \frac{1}{|\mathcal{N}(v^l)|} \sum_{v^j \in \mathcal{N}(v^l)} [v^j - v^l]. \quad (\text{B.51})$$

The weighting parameter $\alpha \in [0, 1]$ is automatically determined such that the following conditions are met

$$\cos(\bar{\mathbf{u}}_{\text{upd}}, -\nabla_x \mathcal{L}) \geq 0, \quad (\text{B.52})$$

$$\cos(\bar{\mathbf{u}}_{\text{upd}}, -\nabla_x \mathcal{L}) \geq \eta \cos(\mathbf{u}_{\text{upd}}, -\nabla_x \mathcal{L}), \quad (\text{B.53})$$

where $\nabla_x \mathcal{L}$ is the raw augmented Lagrange sensitivity (steepest descent direction) and η is the user-defined sufficient decrease parameter, usually in the range $\eta \in [0.85, 0.95]$.

Acknowledgements The authors gratefully acknowledge financial support for this work by the Deutsche Forschungsgemeinschaft: Project Numbers 399073171 (GRK2495/B) and 407523036.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflicts of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Replication of results All the presented methodology is implemented in C++ utilizing the finite element library deal.II, v. 9.2 (Arndt et al. 2020). The version of the code, the executable, the parameter settings and the result files are available from the corresponding author upon request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source,

provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Allaire G, Jouve F, Toader AM (2002) A level-set method for shape optimization. *Comptes Rendus Math* 334(12):1125–1130
- Allaire G, De Gournay F, Jouve F, Toader AM (2005) Structural optimization using topological and shape sensitivity via a level set method. *Control Cybernet* 34(1):59
- Andreassen CS, Elingaard MO, Aage N (2020) Level set topology and shape optimization by density methods using cut elements with length scale control. *Struct Multidiscip Optim* 62:1–23
- Andreassen E, Clausen A, Schevenels M, Lazarov BS, Sigmund O (2011) Efficient topology optimization in matlab using 88 lines of code. *Struct Multidiscip Optim* 43(1):1–16
- Arndt D, Bangerth W, Blais B, Clevenger TC, Fehling M, Grayver AV, Heister T, Heltai L, Kronbichler M, Maier M et al (2020) The deal. II library, version 9.2. *J Numer Math* 28:131–162
- Arora JS (1993) An exposition of the material derivative approach for structural shape sensitivity analysis. *Comput Methods Appl Mech Eng* 105(1):41–62
- Arora JS, Cardoso J (1992) Variational principle for shape design sensitivity analysis. *AIAA J* 30(2):538–547
- Azegami H, Takeuchi K (2006) A smoothing method for shape optimization: traction method using the robin condition. *Int J Comput Methods* 3(01):21–33
- Azegami H, Wu ZC (1996) Domain optimization analysis in linear elastic problems: approach using traction method. *JSME Int J Ser A* 39(2):272–278
- Belytschko T, Xiao S, Parimi C (2003) Topology optimization with implicit functions and regularization. *Int J Numer Methods Eng* 57(8):1177–1196
- Bendsøe MP (1989) Optimal shape design as a material distribution problem. *Struct Optim* 1(4):193–202
- Bendsoe MP, Sigmund O (2013) *Topology optimization: theory, methods, and applications*. Springer Science & Business Media/Springer Science & Business Media, New York
- Bletzinger KU (2014) A consistent frame for sensitivity filtering and the vertex assigned morphing of optimal shape. *Struct Multidiscip Optim* 49(6):873–895
- Braibant V, Fleury C (1984) Shape optimal design using b-splines. *Comput Methods Appl Mech Eng* 44(3):247–267
- Bruns TE, Tortorelli DA (2001) Topology optimization of non-linear elastic structures and compliant mechanisms. *Comput Methods Appl Mech Eng* 190(26–27):3443–3459
- Burger M, Hackl B, Ring W (2004) Incorporating topological derivatives into level set methods. *J Comput Phys* 194(1):344–362
- Chessa J, Smolinski P, Belytschko T (2002) The extended finite element method (xfem) for solidification problems. *Int J Numer Methods Eng* 53(8):1959–1977
- Choi KK, Kim NH (2004) *Structural sensitivity analysis and optimization I: linear systems*. Springer Science & Business Media, New York
- Christiansen AN, Nobel-Jørgensen M, Aage N, Sigmund O, Bærentzen JA (2014) Topology optimization using an

- explicit interface representation. *Struct Multidiscip Optim* 49(3):387–399
- Cormen TH, Leiserson CE, Rivest RL, Stein C (2009) Introduction to algorithms. MIT Press, Cambridge
- da Silva GA, Beck AT, Sigmund O (2019) Topology optimization of compliant mechanisms with stress constraints and manufacturing error robustness. *Comput Methods Appl Mech Eng* 354:397–421
- De Troya MAS, Tortorelli DA (2018) Adaptive mesh refinement in stress-constrained topology optimization. *Struct Multidiscip Optim* 58(6):2369–2386
- Dunning PD, Kim HA, Mullineux G (2011) Investigation and improvement of sensitivity computation using the area-fraction weighted fixed grid fem and structural optimization. *Finite Elements Anal Des* 47(8):933–941
- Eschenauer HA, Kobelev VV, Schumacher A (1994) Bubble method for topology and shape optimization of structures. *Struct Optim* 8(1):42–51
- García-Ruiz M, Steven GP (1999) Fixed grid finite elements in elasticity problems. *Eng Comput* 16:145–164
- Ha SH, Cho S (2008) Level set based topological shape optimization of geometrically nonlinear structures using unstructured mesh. *Comput Struct* 86(13–14):1447–1455
- Hassani B, Tavakkoli SM, Ghasemnejad H (2013) Simultaneous shape and topology optimization of shell structures. *Struct Multidiscip Optim* 48(1):221–233
- Hestenes MR (1969) Multiplier and gradient methods. *J Optim Theory Appl* 4(5):303–320
- Jenkins N, Maute K (2015) Level set topology optimization of stationary fluid-structure interaction problems. *Struct Multidiscip Optim* 52(1):179–195
- Kiendl J, Schmidt R, Wüchner R, Bletzinger KU (2014) Isogeometric shape optimization of shells using semi-analytical sensitivity analysis and sensitivity weighting. *Comput Methods Appl Mech Eng* 274:148–167
- Kreissl S, Maute K (2012) Levelset based fluid topology optimization using the extended finite element method. *Struct Multidiscip Optim* 46(3):311–326
- Lian H, Christiansen AN, Tortorelli DA, Sigmund O, Aage N (2017) Combined shape and topology optimization for minimization of maximal von mises stress. *Struct Multidiscip Optim* 55(5):1541–1557
- Luchi M, Poggialini A, Persiani F (1980) An interactive optimization procedure applied to the design of gas turbine discs. *Comput Struct* 11(6):629–637
- Nadal E, Ródenas J, Albelda J, Tur M, Tarancón J, Fuenmayor F (2013) Efficient finite element methodology based on cartesian grids: application to structural shape optimization. In: Abstract and applied analysis, Hindawi, vol 2013
- Nguyen TT, Bærentzen JA, Sigmund O, Aage N (2020) Efficient hybrid topology and shape optimization combining implicit and explicit design representations. *Struct Multidiscip Optim* 62(3):1061–1069
- Riehl S (2019) Structural optimization of shape and topology using an embedding domain discretization technique. PhD thesis, Institute of Applied Mechanics, Friedrich-Alexander-Universität Erlangen-Nürnberg
- Riehl S, Steinmann P (2015) A staggered approach to shape and topology optimization using the traction method and an evolutionary-type advancing front algorithm. *Comput Methods Appl Mech Eng* 287:1–30
- Riehl S, Steinmann P (2017) On structural shape optimization using an embedding domain discretization technique. *Int J Numer Methods Eng* 109(9):1315–1343
- Rockafellar RT (1973) The multiplier method of hestenes and powell applied to convex programming. *J Optim Theory Appl* 12(6):555–562
- Sharma A, Maute K (2018) Stress-based topology optimization using spatial gradient stabilized xfm. *Struct Multidiscip Optim* 57(1):17–38
- Sigmund O (2001a) A 99 line topology optimization code written in matlab. *Struct Multidiscip Optim* 21(2):120–127
- Sigmund O (2001b) Design of multiphysics actuators using topology optimization-part ii: two-material structures. *Comput Methods Appl Mech Eng* 190(49–50):6605–6627
- Sigmund O, Maute K (2013) Topology optimization approaches. *Struct Multidiscip Optim* 48(6):1031–1055
- Sokolowski J, Zochowski A (1999) On the topological derivative in shape optimization. *SIAM J Control Optim* 37(4):1251–1272
- Villanueva CH, Maute K (2014) Density and level set-xfem schemes for topology optimization of 3-d structures. *Comput Mech* 54(1):133–150
- Villanueva CH, Maute K (2017) Cutfem topology optimization of 3d laminar incompressible flow problems. *Comput Methods Appl Mech Eng* 320:444–473
- Wang F, Lazarov BS, Sigmund O (2011) On projection methods, convergence and robust formulations in topology optimization. *Struct Multidiscip Optim* 43(6):767–784
- Wang MY, Wang X, Guo D (2003) A level set method for structural topology optimization. *Comput Methods Appl Mech Eng* 192(1–2):227–246
- Wang ZP, Abdalla M, Turteltaub S (2017) Normalization approaches for the descent search direction in isogeometric shape optimization. *Comput-Aided Des* 82:68–78
- Yamada T, Izui K, Nishiwaki S, Takezawa A (2010) A topology optimization method based on the level set method incorporating a fictitious interface energy. *Comput Methods Appl Mech Eng* 199(45–48):2876–2891
- Zienkiewicz OC (1973) Shape optimization and sequential linear programming. *Optimum Structural Design Theory and Applications*

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.