



Topology optimization for fail-safe designs using moving morphable components as a representation of damage

Hampus Hederberg¹ · Carl-Johan Thore¹

Received: 22 December 2020 / Revised: 9 June 2021 / Accepted: 10 June 2021
© The Author(s) 2021

Abstract

Designs obtained with topology optimization (TO) are usually not safe against damage. In this paper, density-based TO is combined with a moving morphable component (MMC) representation of structural damage in an optimization problem for fail-safe designs. Damage is inflicted on the structure by an MMC which removes material, and the goal of the design problem is to minimize the compliance for the worst possible damage. The worst damage is sought by optimizing the position of the MMC to maximize the compliance for a given design. This non-convex problem is treated using a gradient-based solver by initializing the MMC at multiple locations and taking the maximum of the compliances obtained. The use of MMCs to model damage gives a finite element-mesh-independent method, and by allowing the components to move rather than remain at fixed locations, more robust structures are obtained. Numerical examples show that the proposed method can produce fail-safe designs with reasonable computational cost.

Keywords Fail-safe · Topology optimization · Moving morphable components · Redundancy

1 Introduction

Topology optimization (TO) has gained increasing popularity since the first article by Bendsoe and Kikuchi (1988) and the research is constantly evolving (Deaton and Grandhi 2014). A commonly studied formulation when designing load-carrying structures using TO is to minimize the compliance of the structure subject to a volume restriction. This usually gives a structure which resembles a statically determinate structure with no redundant parts. This implies that if one of the structural members was to break, by for example a brittle failure in the material due to a sudden impact or by an explosion or a similar occurrence, there would not exist any alternative load-paths, and the structure would experience a catastrophic failure. In for example aerospace applications, this is highly unwanted, and a so-called fail-safe design with redundancy in terms of load-paths is required.

According to Niu (1998, 1997), fail-safe can be defined as:

- A fail-safe structure must support 80–100% limit loads without catastrophic failure.
- A fail-safe structure must support a single member failed in a redundant structure or partial failure in a monolithic structure.

The present article focuses on the second criteria (for work on the first criteria, see, e.g. Wang et al. (2020)). To satisfy this criteria implies there must exist additional load-paths when one structural member is completely removed or partially damaged. For an a priori discrete structure such as a truss or frame this results in a rather simple design problem—in principle one could simply remove one member at a time and minimize the maximum compliance (see Stolpe (2019) for a practical implementation of this idea). Formulation of the design problem for continuum structures is however not as straightforward. If one were to look at the structure, it would be possible to distinguish a structural member and then say where a damage patch would do a lot of damage, but of course this is not possible if it is going to be implemented into a completely automatic optimization algorithm.

Responsible Editor: Xu Guo

✉ Hampus Hederberg
hampus.hederberg@liu.se

¹ Division of Solid Mechanics, Linköping University, Linköping, Sweden

The first article published on fail-safe TO for continua is written by Jansen et al. (2014). In this article, a grid of fixed rectangular damage patches covers the design domain. The optimization problem is then written as a minimization of the compliance under the worst damage, i.e. the damage which gives the maximum compliance. To find the worst damage, all damage patches had to be tested, and since the number of patches is in the range of the number of finite elements the problem becomes very computationally demanding.

To reduce the number of damage patches, Jansen et al. (2014) tried to only include patches which contained material and another approach where only patches which had a non-monotonic density variation were considered. Zhou and Fleury (2016) tried to further decrease the number of patches by having two sets of damage populations in which they increased the number of damage zones until more patches would not do much difference. This was measured by looking at the so-called minimum hideout volume of the added patches.

Ambrozkiwicz and Kriegesmann (2018) examined how well the algorithm by Zhou and Fleury (2016) performed and found that it was sensitive to damage zones which weren't included in the optimization. They also proposed their own method where they used the stress-state in the structure to divide the structure into knots and beams and each knot and beam was then removed one at a time to simulate a damage to that particular place. This method diverges a bit from the rectangular damage zones seen in previous work.

There are also some research where the fail-safe criterion is formulated in a more implicit way (Wu et al. 2018; Dou 2019). These methods do not include a damage formulation but instead uses local volume constraints which makes the structure get more members and thus increase the redundancy and consequently the damage tolerance. But since fail-safe is not explicitly included the structure has to be tested after optimization.

The most recent article on fail-safe TO is Ambrozkiwicz and Kriegesmann (2020) where the authors first used a local volume method of the type mentioned in the preceding paragraph to get a first draft of a redundant structure and then they used their previous research to divide the structure into knots and beams. The optimization then comes to the second stage where a density-based shape optimization was performed to get the final design. To divide the structure into knots and beams in 2D, the stress-state was examined. This was not possible in 3D with multiple load-cases and thus a visualisation tool was used. The knot and beam formulation is close to that of an MMC but they used it in a different way.

In this article we propose a method to obtain fail-safe designs by using so-called moving morphable components (MMCs) (Guo et al. 2014) to model damage in the structure while using density-based TO to obtain an optimized material layout of the structure. Damage is inflicted on the structure by an MMC which removes material, and the goal of the design problem is to minimize the compliance for the worst possible damage. The worst damage for a given design is sought by optimizing the position of the MMC to maximize the compliance. The smooth but non-convex problem of finding the worst damage is treated using a gradient-based solver by initializing the MMC at multiple locations and taking the maximum of the compliances obtained. This heuristic method does not guarantee that the worst damage is actually found of course, but the numerical results indicate that the method works well in practise.

An advantage of proposed method compared to existing methods for fail-safe design cited above is finite element (FE) mesh independency in the sense that the only connection an MMC has to the mesh is when the MMC is projected onto the FE-mesh for density and gradient calculations. The position of an MMC is thus not bound to the FE-mesh but can move around freely using an optimization solver to find a position where it does the most damage. As will be seen in the numerical examples below, compared to a strategy with fixed damage patches such as in Jansen et al. (2014), the moving patches used herein give more robust structures.

Finally, one issue found in the reviewed literature was that there was very little verification that the structures obtained actually were fail-safe. In the present work, so-called damage maps are used just to show that the optimized structures actually are fail-safe.

2 Background theory

To get an understanding of how the proposed method works, a short explanation of both density-based TO and MMC TO is provided.

2.1 Density-based topology optimization

The basic problem considered in this article is to maximize stiffness under a volume constraint. The design is described by a (relative) density function, approximated as element-wise constant and taking values between 0 (no material) and 1 (material). The elemental values ρ_e are collected in a vector $\boldsymbol{\rho} \in \mathbb{R}^m$, where m is the number of FEs. The Solid Isotropic Material with P+enalization (SIMP) scheme is used to obtain close to binary-valued designs (Christensen and Klarbring 2008; Rietz 2001).

The mathematical formulation of the (FE-discretized) optimization problem is

$$\begin{cases} \min_{\rho} C(\rho) = \mathbf{F}^T \mathbf{U}(\rho) \\ \text{s.t.} \begin{cases} V(\rho) \leq \alpha V_0 \\ 0 \leq \rho_e \leq 1, \quad e = 1, \dots, m, \end{cases} \end{cases}$$

where $\mathbf{U}(\rho)$ solves the equilibrium equation $\mathbf{K}(\rho)\mathbf{U} = \mathbf{F}$, in which $\mathbf{K}(\rho)$ is the stiffness matrix, and \mathbf{F} is the load-vector. Furthermore, V_0 is the volume of the design domain Ω , $V(\rho)$ is the volume of the structure and α is the fraction of volume that the optimized structure is allowed to occupy.

2.1.1 Filtering

To avoid FE mesh dependency and checkerboards for low order FEs (Sigmund and Petersson 1998), a linear filter (Bourdin 2001) is applied to the design variables, giving a filtered design

$$\tilde{\rho}_e = \frac{1}{\sum_{i=1}^m Z_{ei}} \sum_{i=1}^m Z_{ei} \rho_i, \quad e = 1, \dots, m,$$

where $Z_{ei} = \max(0, R - \Delta(e, i))$ is a weight function which smooths the design. Here R is the filter radius and $\Delta(e, i)$ the Euclidean distance between the centroid of elements e and i .

2.1.2 Projection for 0/1-designs

When applying a filter, the design becomes gray (intermediate densities) at the boundary of the structure. To instead get a 0/1 boundary without intermediate densities, a so-called projection-filter is often used. Ideally, the Heaviside-function (defined by $H(x) = 0$ for $x < 0$ and $H(x) = 1$ for $x \geq 0$) should be used. The discontinuity of this function may however cause problems for gradient-based solvers. To circumvent this, a regularized version of the Heaviside-function (Wang et al. 2011) is used to get the projected density

$$\bar{\rho}_e(\tilde{\rho}_e) = \frac{\tanh(\psi\eta) + \tanh(\psi(\tilde{\rho}_e - \eta))}{\tanh(\psi\eta) + \tanh(\psi(1 - \eta))}, \quad (1)$$

where ψ determines the sharpness of the step. A higher ψ will result in a sharper step but also a more unstable optimization process. Usually, a continuation scheme is used where ψ is gradually increased during the optimization.

A useful feature of the projection is that different values of η can be used for the objective function and the volume constraint. This is used in the present work due to convergence issues seen if the same value for η was used for both objective function and volume constraint. This procedure is known as a robust formulation (Wang et al.

2011) and is used by Jansen et al. (2014). The so-called nominal density $\bar{\rho}_e^{(n)}(\tilde{\rho}_e)$ is obtained by putting $\eta = 0.5$ and the eroded density $\bar{\rho}_e^{(e)}(\tilde{\rho}_e)$ is obtained by putting $\eta = 0.7$. By using the robust formulation, the design becomes more robust to slight changes in the material distribution and thin members vanishes. The densities to be plotted in the following are nominal densities.

2.1.3 The stiffness matrix

Finally, the eroded density is used with a material interpolation scheme to get the stiffness matrix for the structure. Young's modulus for each element is written as

$$E_e(\rho_e) = E_{min} + (E_0 - E_{min})(\bar{\rho}_e^{(e)})^{q_1}, \quad (2)$$

where $q_1 > 1$ is the SIMP penalization exponent, E_0 is the nominal stiffness of the solid phase, and $E_{min} > 0$ is the stiffness of the void phase needed to ensure a non-singular stiffness matrix. The global stiffness matrix is then

$$\mathbf{K}(\rho) = \sum_{e=1}^m E_e(\rho_e) \mathbf{k}_e^0,$$

where \mathbf{k}_e^0 is the stiffness of an element with unit Young's modulus.

2.2 Moving Morphable components

Topology optimization with MMCs is an alternative to the density-based approach which was introduced by Guo et al. (2014) and Zhang et al. (2016).

2.2.1 The mathematical description of a component

In this work, we consider components described by level-set functions of the form

$$\phi(\mathbf{x}) = 1 - \left(\frac{x'}{L} \right)^{p_\phi} - \left(\frac{y'}{f(x')} \right)^{p_\phi}, \quad (3)$$

where L is the length of the component, $f(x')$ is a function describing the thickness, and p_ϕ is a relatively large, even integer. A component with the associated design variables can be seen in Fig. 1. The relation between global xy - and local $x'y'$ -coordinates is expressed as

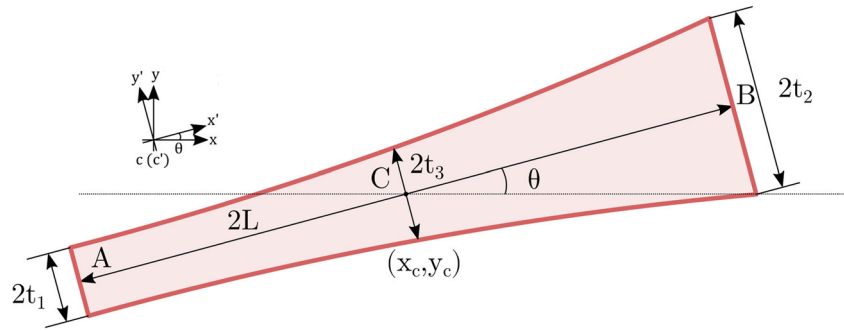
$$\begin{Bmatrix} x' \\ y' \end{Bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{Bmatrix} x - x_c \\ y - y_c \end{Bmatrix}, \quad (4)$$

in which x_c and y_c are the components of the centroid \mathbf{x}_c . The function $f(x)$ can be defined in different ways depending on what design variables one wants to use. For a quadratically changing width, the function can be written as

$$f(x') = \frac{t_1 + t_2 - 2t_3}{2L^2} (x')^2 + \frac{t_2 - t_1}{2L} x' + t_3,$$

where the parameters t_1 , t_2 , and t_3 are defined in Fig. 1.

Fig. 1 A moving morphable component together with its design variables



If the level-set function is greater than 0 at a point \mathbf{x} , then that indicates that the component exists at that position and thus that the elements covered by the component should contain material. If the function is equal to 0, then that defines the boundary of the component and if the function is less than 0, then that means the component is not present at that position and should not contribute with any material. In a more mathematical way, this is written as

$$\begin{cases} \phi(\mathbf{x}) > 0, & \text{if } \mathbf{x} \in \Omega_c \\ \phi(\mathbf{x}) = 0, & \text{if } \mathbf{x} \in \partial\Omega_c \\ \phi(\mathbf{x}) < 0, & \text{if } \mathbf{x} \in \Omega \setminus \Omega_c \end{cases}$$

where Ω_c and $\partial\Omega_c$ are the domain of the component and its boundary, respectively.

Multiple, potentially overlapping components can be combined using the max-function to a single function $\phi^s(\mathbf{x}) = \max\{\phi_1(\mathbf{x}), \dots, \phi_n(\mathbf{x})\}$ defining the resulting structure.

2.2.2 The Ersatz material model

To move from a level-set function ϕ to a corresponding (element-wise constant) density distribution ρ , a cheap method is to use the ersatz material model (Guo et al. 2014; Zhang et al. 2016) in which the density of an element e is given by

$$\rho_{e,MMC} = \frac{\sum_{j=1}^{n_n} (\tilde{H}(\phi^s(\mathbf{x}_j))^{q_2}}{n_n}, \quad (5)$$

where n_n is the number of sampling points \mathbf{x}_j and \tilde{H} is a smooth approximation of the Heaviside function specified in Section 3 below. The parameter q_2 is used to get a sharper boundary of the structure (Coniglio et al. 2019). Formula (5) is then used in the same manner as in (2), so that the effective Young's modulus becomes

$$E_{e,MMC} = E_{min} + (E_0 - E_{min})\rho_{e,MMC}.$$

2.2.3 Example of MMC TO

To get a better understanding how TO with MMCs actually works we consider an example of optimization of a

cantilever beam whose left end is fixed and whose right end is subject to a vertical load. In Fig. 2 (top) one can see the initial configuration of all the 24 MMCs that define the design. The black contour lines indicate $\phi = 0$, the green regions are where $\phi > 0$, and the white regions are where $\phi < 0$ as explained previously. In Fig. 2 (bottom), one can see the final design. In Fig. 3, one can then see how these components are projected onto an (180×60) FE-mesh through the Heaviside function.

3 Topology optimization for fail-safe designs

We now combine density-based TO with the MMC formulation to obtain fail-safe designs. This is done in the form of a minmax formulation where instead of adding material, one or more MMCs now remove material to model damage on the structure. The goal of the inner maximization problem is thus to find the position, given by the centroid \mathbf{x}_c , of an MMC that causes the most damage to the structure. The shape of the MMC is taken as a square with rounded corner, more precisely a squircle (*sic*), but other shapes could of course also be considered (see sections 5.3 and 5.4

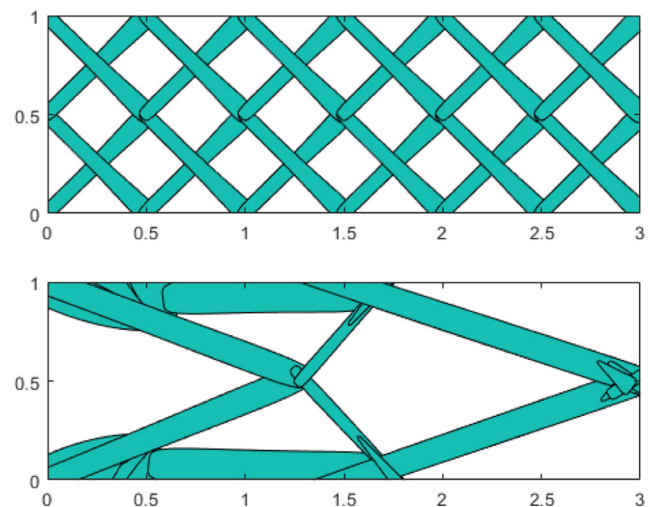


Fig. 2 Initial (top) and final configuration (bottom) of the MMCs

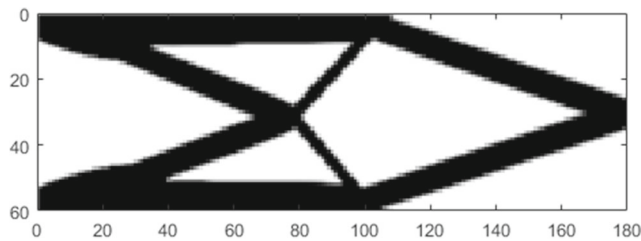


Fig. 3 Density distribution obtained by projection of the components in the final design in Fig. 2

below). An example of an optimized, damaged structure can be seen in Fig. 4 (top).

Mathematically, the problem of minimizing the worst compliance now reads:

$$\begin{cases} \min_{\rho} \max_{\mathbf{x}_c} \mathbf{F}^T \mathbf{U}(\rho, \mathbf{x}_c) \\ \text{s.t.} \begin{cases} V(\rho) \leq \alpha V_0 \\ 0 \leq \rho_e \leq 1, \quad e = 1, \dots, m \end{cases} \end{cases} \quad (6)$$

where $\mathbf{U}(\rho, \mathbf{x}_c)$ solves the equilibrium equation

$$\mathbf{K}(\rho, \mathbf{x}_c) \mathbf{U} = \mathbf{F}.$$

Note that no bounds are needed for \mathbf{x}_c since the worst damage is always obtained for a patch overlapping the design domain. Given a bounded design domain, the optimal centroid is thus confined to a compact set, meaning that the inner problem is well-posed since the objective is continuous in \mathbf{x}_c .

The non-convexity of the inner maximization problem in (6) is illustrated in Fig. 4 (bottom), where the dotted squares are the initial positions of two different damage patches and the solid squares are the final positions of the damage when optimizing the position. It is thus seen that even if the starting positions of two damage patches are close to

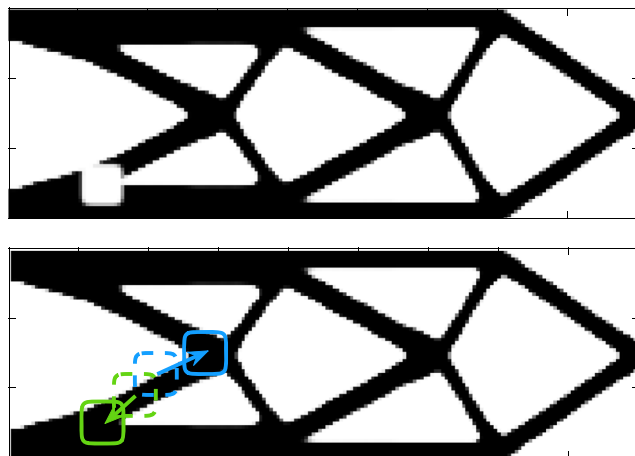


Fig. 4 Top: a damage patch located near the lower left corner of the design domain. Bottom: depending on the starting position (dashed lines) a damage patch may end up in different positions

each other they do not necessarily end up at the same end position.

While implementing the algorithm it was found that due to the non-convexity of the inner problem, more than one damage patch was needed to make sure that the worst damage was found. (Since we only consider one damage patch at the time however, an alternative point-of-view is that we have one damage patch which is initialized at many different locations. Implementation-wise we get the same problem formulation however.) The objective will therefore be to mitigate the damage from the worst positioned damage obtained for any one of n_d damage patches. Introducing the sets $\mathcal{X}_i = \{\mathbf{x}_{ci} \mid \underline{\mathbf{x}}_{ci} \leq \mathbf{x}_{ci} \leq \bar{\mathbf{x}}_{ci}\}$, $i = 1, \dots, n_d$, the optimization problem is written as:

$$\begin{cases} \min_{\rho} \max_{i=1, \dots, n_d} \max_{\mathbf{x}_{ci} \in \mathcal{X}_i} \mathbf{F}^T \mathbf{U}(\rho, \mathbf{x}_{ci}) \\ \text{s.t.} \begin{cases} V(\rho) \leq \alpha V_0 \\ 0 \leq \rho_e \leq 1, \quad \forall e = 1, \dots, m \end{cases} \end{cases} \quad (7)$$

The box constraints in the sets \mathcal{X}_i are introduced to make sure that the entire design domain is searched without the patches getting stuck in positions where they do not cause much damage.

3.1 Inner problem

The innermost problem in (7) is to maximize the compliance by moving the i th MMC over a given design ρ . The MMC could be quite complicated as described in (3), but to be able to compare the current algorithm to previous algorithms from the literature a simpler function is used, namely

$$\phi_i(\mathbf{x}; \mathbf{x}_{ci}) = 1 - \left(\frac{x - x_{ci}}{L} \right)^{p_\phi} - \left(\frac{y - y_{ci}}{L} \right)^{p_\phi}, \quad (8)$$

whose zero level set is a (super-ellipse based) squircle. If p_ϕ is set to 2 a circle is obtained and the higher p_ϕ used, the sharper the edges on the MMC. For this study, a value of 6 was chosen which yields a square component with rounded corners.

The function ϕ_i in (8) is then put through a regularized Heaviside function and sampled to get a corresponding density distribution as described in Section 2.2.2. The regularized Heaviside function used in this article is defined as

$$\tilde{H}(x) = \frac{1}{2} (1 + \tanh(\beta x)), \quad (9)$$

where $\beta > 0$ controls the steepness of the (smooth) step. The stiffness of the structure will now depend on the MMC damage as well as the (eroded) density-distribution $\bar{\rho}_e = \bar{\rho}_e(\bar{\rho}_e(\rho))$ from (1), where the density variable adds material

and the MMC removes material. This gives the effective Young's modulus

$$\tilde{E}_e(\rho, \mathbf{x}_{ci}) = E_{min} + (E_0 - E_{min}) \tilde{\rho}_e^{q_1} \left(1 - \frac{\sum_{s=1}^{n_n} (\tilde{H}(\phi(\mathbf{x}_s; \mathbf{x}_{ci})))^{q_2}}{n_n} \right), \quad (10)$$

and thus the stiffness matrix

$$\mathbf{K}(\rho, \mathbf{x}_{ci}) = \sum_{e=1}^m \tilde{E}_e(\rho, \mathbf{x}_{ci}) \mathbf{k}_e^0.$$

3.1.1 Sensitivity analysis

To use a gradient-based algorithm to solve the inner-most problem in (7), the gradient of the compliance $C_i = \mathbf{F}^T \mathbf{U}(\rho, \mathbf{x}_{ci})$ with respect to the centroid is needed. The only constraints are the box constraints on the centroid and no derivative is needed of these. Differentiating the compliance with respect to the position of the damage-patch using the adjoint method (Christensen and Klarbring 2008), we get

$$\frac{\partial C_i}{\partial \mathbf{x}_c} = -\mathbf{U}(\rho, \mathbf{x}_{ci})^T \frac{\partial \mathbf{K}(\rho, \mathbf{x}_{ci})}{\partial \mathbf{x}_c} \mathbf{U}(\rho, \mathbf{x}_{ci}),$$

where

$$\frac{\partial \mathbf{K}(\rho, \mathbf{x}_{ci})}{\partial \mathbf{x}_c} = \sum_{e=1}^m \frac{\partial \tilde{E}_e(\rho, \mathbf{x}_{ci})}{\partial \mathbf{x}_c} \mathbf{k}_e^0.$$

The derivative of the regularized Heaviside function from (9) is

$$\frac{\partial \tilde{H}}{\partial \mathbf{x}_c} = \frac{1}{2} (1 - \tanh^2(\beta \phi(\mathbf{x}_s; \mathbf{x}_{ci}))) \beta \frac{\partial \phi}{\partial \mathbf{x}_c},$$

and finally, the derivative of the level-set function from (8) is

$$\frac{\partial \phi}{\partial \mathbf{x}_c} = \frac{p_\phi}{L} \left(\frac{x'}{L} \right)^{p_\phi-1}$$

The derivative with respect to y_c is obtained analogously.

3.1.2 Extra sampling points

Using too few sampling points in (10) can cause difficulties. For example, a common choice is to use only the FE nodes as sampling points. This can give a good representation of the component itself but not of the derivative with respect to the centroid which is non-zero only in a narrow region at the boundary of the component. To illustrate this, Fig. 5 shows how a component centred at $x = 0.5$ would look in 1D with a coarse 40-element grid (red), a coarse grid with 3 extra points per element (blue) and how the regularized Heaviside function would actually look (green). As one can see, there is not much difference between extra nodes and no extra nodes. In the bottom plot in Fig. 5; however, the derivative of the component is shown, and here some issues can be seen: the derivative on the coarse grid (red) is almost flat while the derivative with extra nodes (blue) is closer to the actual derivative (green) although averaged over the entire element. Worth noting is that the function is differentiated with respect to x_c and not x and thus the derivative is negative on the left side and positive on the right side since material is added on the right side and removed on the left side if the component moves to the right.

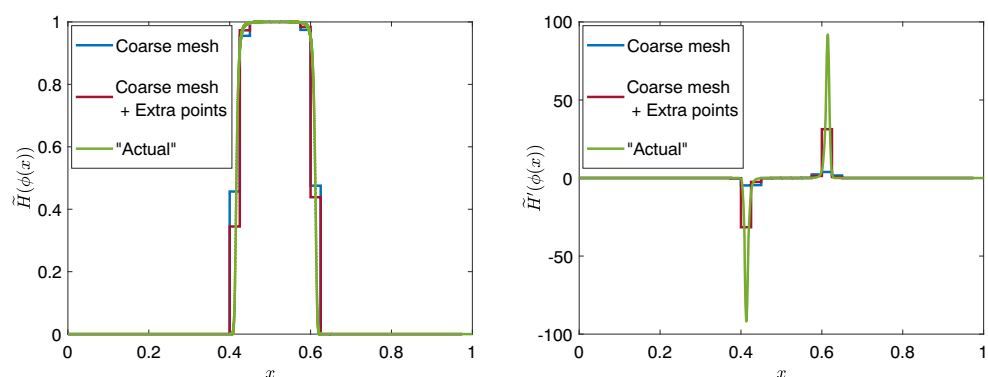
To avoid problems with vanishing derivatives, we use $n_n = 16$ sampling points in our numerical examples. Note that increasing the number of sampling points has negligible effect on the computational time which is dominated by solving the equilibrium equation.

3.2 Outer problem

The outer problem in (7) consists of finding a design which minimizes the worst compliance obtained for any of the damage patches, i.e.

$$C(\rho) = \max_{i=1, \dots, n_d} C_i(\rho) = \mathbf{F}^T \mathbf{U}(\rho, \mathbf{x}_{ci}^*),$$

Fig. 5 Left: the Heaviside function and the sampled Heaviside function in a coarse mesh and a coarse mesh with extra points. Right: the derivative of the component



where $\mathbf{x}_{ci}^* = \mathbf{x}_{ci}^*(\boldsymbol{\rho})$ solves $\max_{\mathbf{x}_{ci} \in \mathcal{X}_i} \mathbf{F}^T \mathbf{U}(\boldsymbol{\rho}, \mathbf{x}_{ci})$. The optimization problem can thus be written as

$$\begin{cases} \min_{\boldsymbol{\rho}} C(\boldsymbol{\rho}) \\ s.t. \begin{cases} V(\boldsymbol{\rho}) \leq \alpha V_0 \\ 0 \leq \rho_e \leq 1, \quad \forall e = 1, \dots, m \end{cases} \end{cases}$$

The max-function is non-smooth and therefore the Kreisselmeier-Steinhauser (KS) approximation (Kreisselmeier and Steinhauser 1979) is used to get

$$C(\boldsymbol{\rho}) \approx C_{KS}(\boldsymbol{\rho}) = \frac{1}{\gamma} \ln \left(\sum_{i=1}^{n_d} \exp(\gamma C_i(\boldsymbol{\rho})) \right), \quad (11)$$

where a higher value of γ makes the approximation closer to the max-function.

3.2.1 Sensitivity analysis

The derivative of the KS-approximation with respect to ρ_e is given by

$$\frac{\partial C_{KS}}{\partial \rho_e} = \frac{\sum_{i=1}^{n_d} \exp(\gamma C_i) \frac{\partial C_i}{\partial \rho_e}}{\sum_{i=1}^{n_d} \exp(\gamma C_i)}, \quad (12)$$

where

$$\frac{\partial C_i}{\partial \rho_e} = -\mathbf{U}(\boldsymbol{\rho}, \mathbf{x}_{ci}^*)^T \left(\sum_{j=1}^m \frac{\partial \mathbf{K}}{\partial \tilde{\rho}_j} \frac{\partial \tilde{\rho}_j}{\partial \rho_e} \right) \mathbf{U}(\boldsymbol{\rho}, \mathbf{x}_{ci}^*), \quad (13)$$

in which

$$\frac{\partial \mathbf{K}}{\partial \tilde{\rho}_j} = q_1(E_0 - E_{min}) \cdot$$

$$\tilde{\rho}_j^{q_1-1} \left(1 - \frac{\sum_{s=1}^{n_n} (\tilde{H}(\phi(\mathbf{x}_s; \mathbf{x}_{ci}^*)))^{q_2}}{n_n} \right) \mathbf{k}_j^0,$$

$$\frac{\partial \tilde{\rho}_j}{\partial \tilde{\rho}_j} = \frac{\psi(1 - \tanh^2(\psi(\tilde{\rho}_j - \eta)))}{\tanh(\psi\eta) + \tanh(\psi(1 - \eta))},$$

and $\partial \tilde{\rho}_j / \partial \rho_e = Z_{je} / \sum_{i=1}^m Z_{ji}$.

The expression (13) is valid provided that the i th inner problem $\max_{\mathbf{x}_{ci} \in \mathcal{X}_i} \mathbf{F}^T \mathbf{U}(\boldsymbol{\rho}, \mathbf{x}_{ci})$ has a unique solution (c.f. Theorem 1 in Klarbring and Strömberg (2012)). That is not necessarily the case in our numerical examples but we nevertheless observe good convergence as shown in Fig. 13 for example.

3.3 The optimization algorithm

Problem (7) is solved using an algorithm which can be seen in the flowchart in Fig. 6.

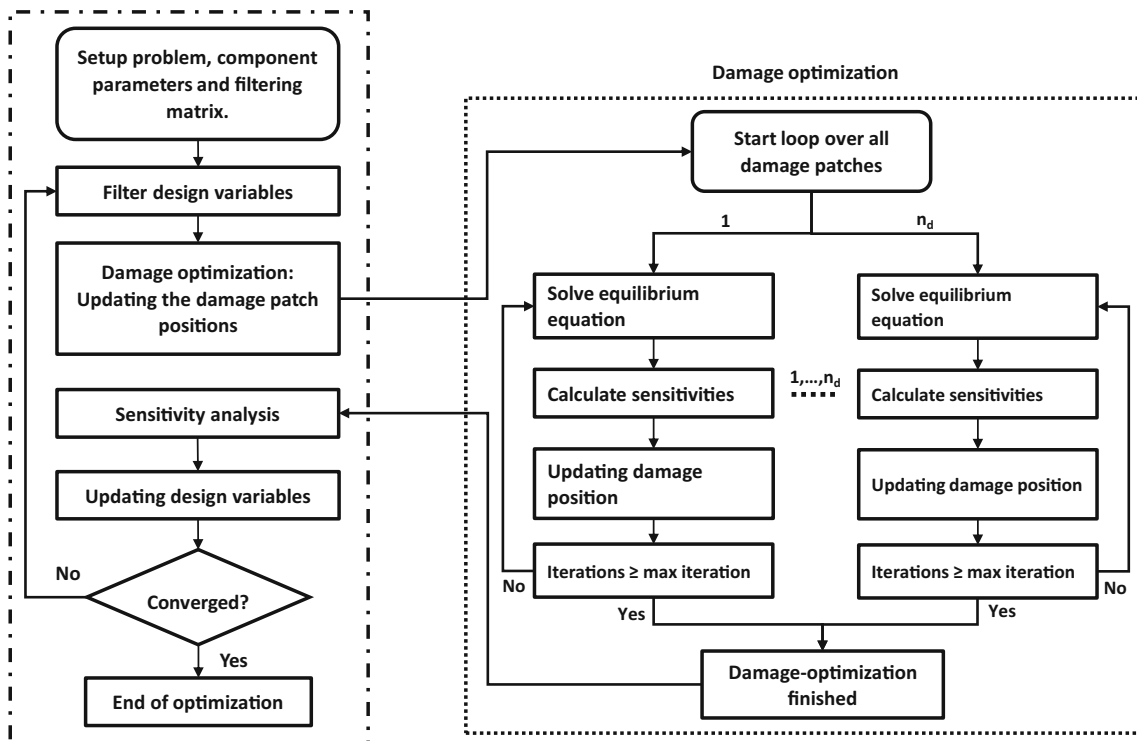


Fig. 6 Flowchart of the optimization algorithm with the inner and outer optimization problem

We note that a somewhat similar algorithm was proposed in da Silva et al. (2019) for TO under load uncertainty, and that the inner loop over all damage patches is almost trivial to parallelize if needed. The update of the position of the damage patch was done by the use of the method of moving asymptotes (Svanberg 1987) while the design was updated using the optimality criteria (OC) method (Christensen and Klarbring 2008) as implemented in Andreassen et al. (2011).

4 Numerical examples

The numerical examples in this section are based on the set-up shown in Fig. 7.

A “safe-zone” (the region to the right of the blue line in Fig. 7), where no damage can be present, is applied so as to not disconnect the force from the body (which would result in a very large compliance due to the non-zero Young’s modulus $E_{min} > 0$ of void regions). Damage patches of two different sizes that will be used are also shown.

The parameter γ in the KS-approximation (11) is taken as $5/C_{max}$ where C_{max} is the maximum compliance obtained for a given design (strictly speaking one should then consider C_{max} when computing derivatives). Due to big differences in the compliance over the course of the optimization, C_{max} is updated every 10th (outer) iteration. The parameter ψ in (1) starts at 2 and is doubled every 50th iteration until a final value of 16. After the 20 first iterations, the design does not change significantly and thus the positions of the MMCs are updated 4 times every iteration the first 20 iterations and thereafter only updated once per iteration to save computational time. The box constraint for each MMC is set to be $\pm 0.2L$ in both x - and y -direction starting from the initial position of the MMC. The initial positions for all MMCs are shown for each example below. The mesh size for the cantilever beam was 180×60 elements. All other parameters and settings can be found in Table 1.

Table 1 All parameters used in the optimization

Parameter	Value
Sampling points	$n_n = 16$
Density penalization	$q_1 = 3$
Heaviside exponent	$q_2 = 2$
Filter radius	$R = 0.05333 L$
Heaviside steepness	$\beta = 5$
ϕ exponent	$P_\phi = 6$
OC Move limit	0.05
Maximum OC iterations	400
Damage map components	360×120
Projection level for eroded design	$\eta_{eroded} = 0.7$
Projection level for nominal design	$\eta_{nominal} = 0.5$
Nominal Young’s modulus	$E_0 = 1$
Poisson’s ratio	0.3
Void Young’s modulus	$E_{min} = 10^{-9}$
Projection steepness	$\psi \in \{2, 4, 6, 8, 16\}$
Length parameter	$L=1$
Thickness of design domain	1
Element type	Quad4
MMA parameter asyinit	1
MMA parameter asyincr	1.4
MMA parameter asydecr	0.7

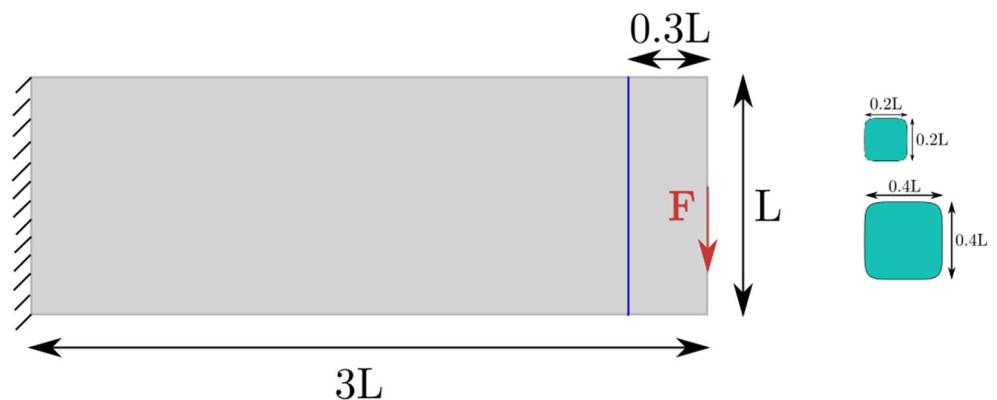
As initial guess for the design, we use in all cases a uniform density-distribution satisfying the volume and box constraints.

4.1 Damage patch 0.2L

4.1.1 Nominal design with damage

In Fig. 8 (top), one can see how the design looks when it is optimized not considering damage and in the middle plot one can see the worst damage in the structure. To assess the performance of the structure, a so-called damage map,

Fig. 7 Design domain with loads and boundary conditions. To the right are examples of damage patches of two different sizes



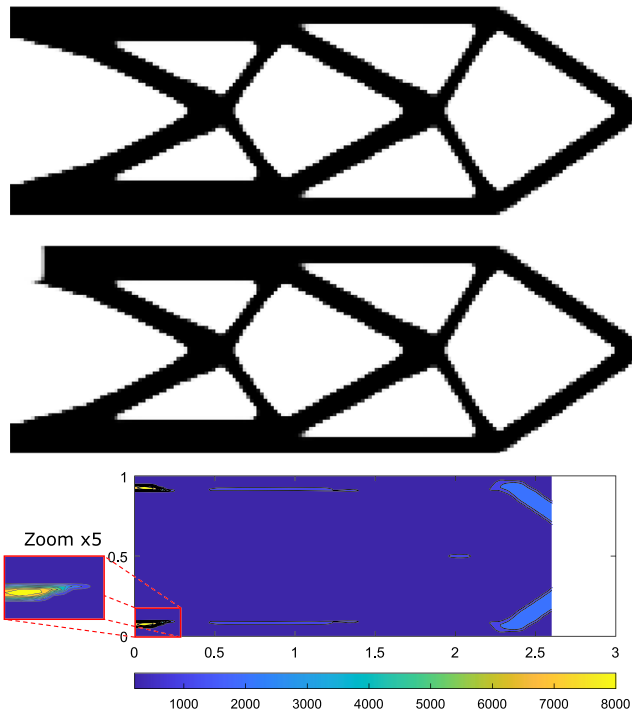


Fig. 8 Top: optimized design. Middle: the design with the worst damage. Bottom: damage map for the design

Fig. 8, is constructed by using 360×120 damage patches evenly spread throughout the domain. This yields a map where one can see how the worst compliance varies as a function of the centroid position of a damage patch. The nominal compliance of the nominal design is 202.4 and the compliance for the worst damage is 8627.96. A damage to the optimized structure not considering damage is thus highly detrimental for the performance.

4.1.2 Optimized design with stationary damage

To compare to earlier work in Jansen et al. (2014) and Zhou and Fleury (2016), an optimization is performed with stationary damage patches. The optimization was performed with 60×180 damage patches uniformly distributed in the domain. The top plot in Fig. 9 shows the optimized design, and the middle plot shows the design with the worst damage obtained from the damage map in the lowermost plot. The undamaged compliance was 245.41 and the worst compliance found during the optimization was 371.28. This is significantly lower than what was found from the damage map where the worst compliance found was 497.46. This shows that even if a large amount of stationary damage patches is used, even a slight change in the position of the damage patches can damage the structure significantly more. Below, we will see that this effect can be avoided

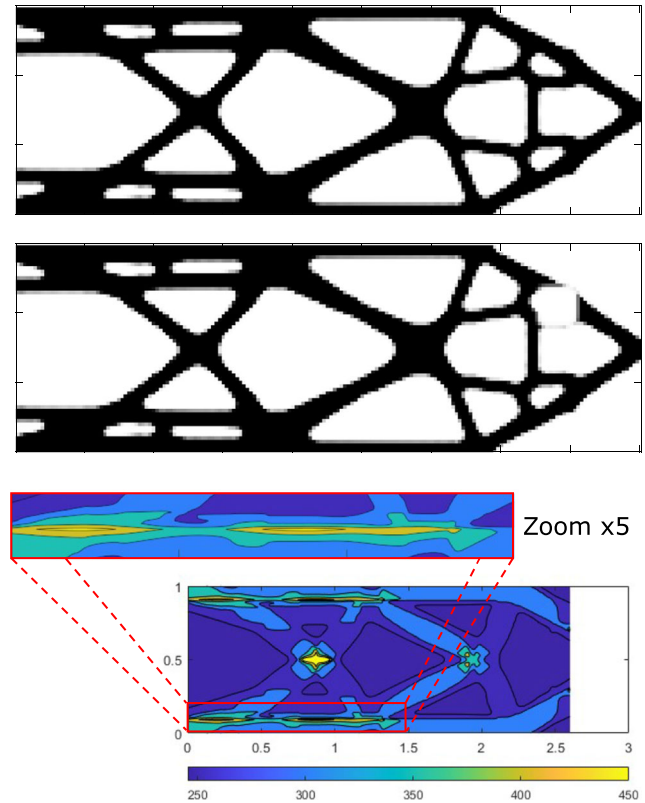


Fig. 9 Top: design optimized with stationary damage patches. Middle: design with the worst damage. Bottom: damage map

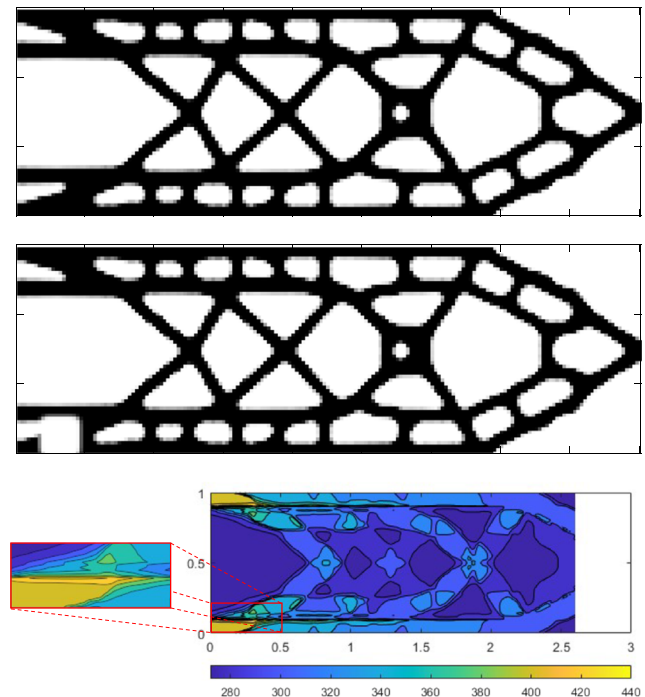


Fig. 10 Top: optimized design with 5×12 moving damage patches. Middle: design with the worst damage. Bottom: damage map

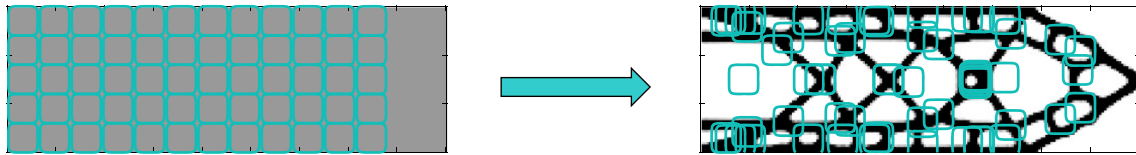


Fig. 11 Initial and final density distribution with the corresponding damage patch distribution (turquoise squares) superimposed

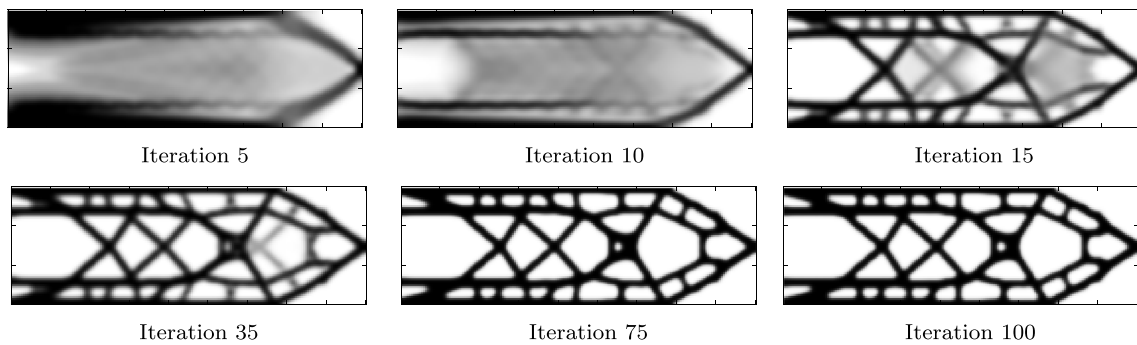


Fig. 12 The evolution of the design during the first 100 design updates

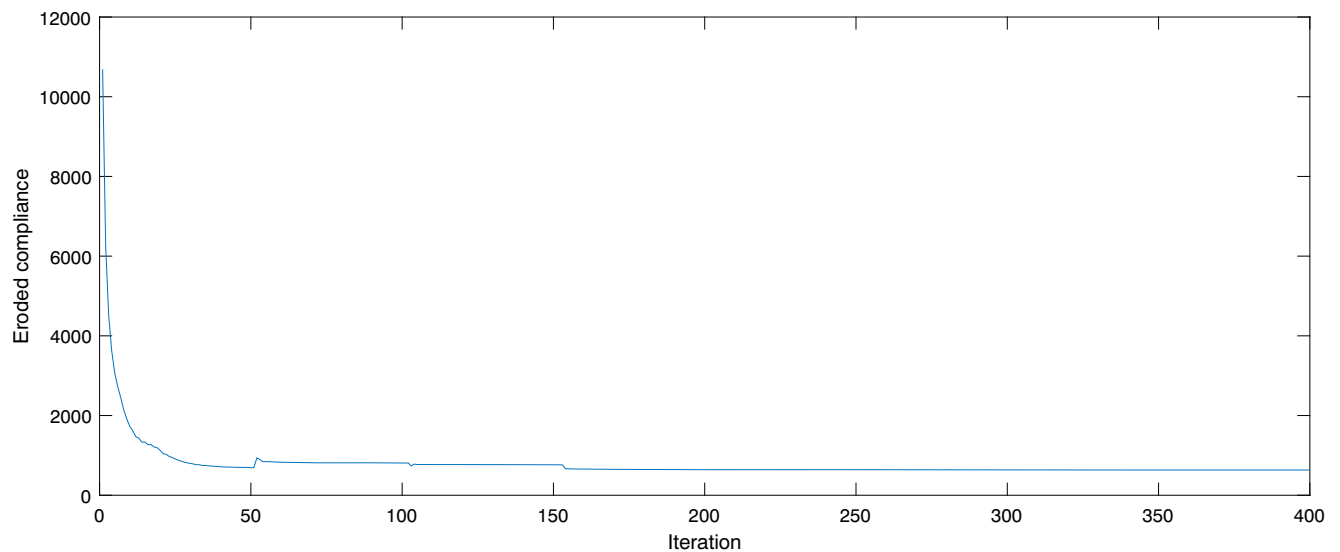


Fig. 13 The eroded compliance as a function of the iterations during the optimization

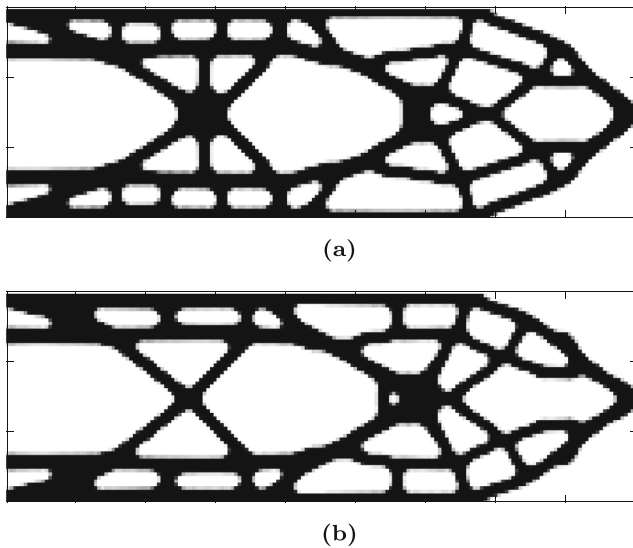


Fig. 14 Designs for $3 \times 10 = 30$ (a) and $5 \times 24 = 120$ (b) moving damage patches

almost entirely by using a much smaller number of moving damage patches.

4.1.3 Optimized design with moving damage

Figure 10 shows an optimized design for an initial damage distribution of 5×12 damage patches. As can be seen in the middle plot, the worst damage is located near the support at the bottom left part of the domain. The corresponding damage plot is seen in the bottom of Fig. 10.

The initial and final distribution of damage patches for the design in Fig. 10 can be seen in Fig. 11 superimposed onto the respective design. The evolution of the design can be seen in Fig. 12

The objective function (the maximum eroded compliance) evolution can be seen in Fig. 13. The general appearance is smooth with jumps at 50, 100, and 150 iterations due

to the update of the steepness-parameter ψ for the Heaviside projection filter defined in (1).

In Fig. 14a, one can see the design obtained when $3 \times 10 = 30$ uniformly distributed damage patches were considered during the optimization and in Fig. 14b one can see the design when $5 \times 24 = 120$ damage patches were considered during the optimization. The undamaged and damaged compliance can be seen in Table 2.

The principle appearance, as well as the compliance, of the designs in Figs. 9, 10, 14b and a is similar. If more damage patches were used, then the worst damage found from the optimization is closer to the worst damage found from the damage map. For the 0.2L damage patch, the compliance from the optimization is still close to the worst damage even when only 30 damage patches were used: 431.79 compared to 453.22.

The computational time required for 5×12 moving patches was approximately 23.4 min and for 60×180 stationary patches it was approximately 5542 min. This is thus roughly 240 times faster. This can be compared with it being 180 times more damage patches in the example with the stationary patches. Some additional computational times can be seen in Table 2. The optimization ran on a non-optimized code using MATLAB R1029b on a Intel core i9-9900K @3.60 GHz processor with 64 GB of RAM and using a sparse direct solver for the equilibrium equations. The bottleneck in the calculation is the number times the equilibrium equation needs to be solved.

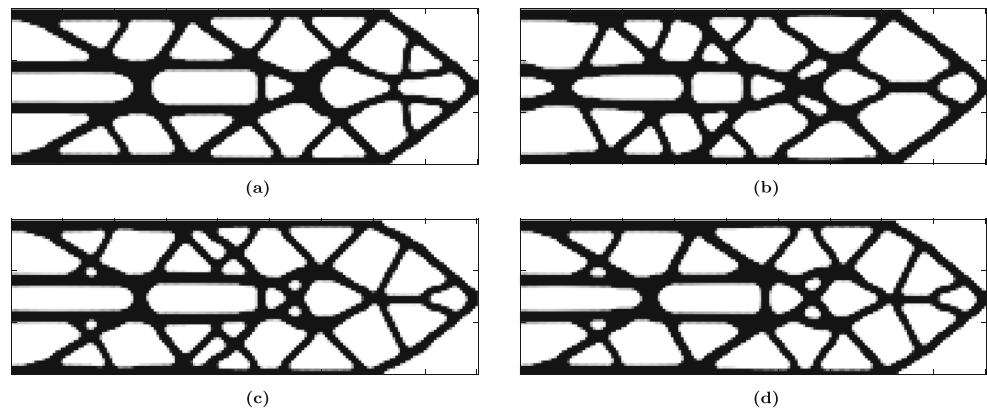
4.2 Damage patch 0.4L

Figure 15 shows designs obtained with a damage patch of size 0.4L. In Fig. 15a, we see a design for 60×180 stationary damage patches, in Fig. 15b, one with 3×10 moving damage patches, in Fig. 15c one with 5×12 moving damage patches, and in Fig. 15d one with 5×24 moving damage patches. The designs here as well are similar to each

Table 2 Compliance for the cantilever beam example obtained with different settings for the damage size and number of patches

Damage patches in optimization	Damage size	Undamaged compliance	Worst compliance, optimization	worst compliance, damage map	Time (min)
n/a	0.2L	202.4	n/a	8627.96	~ 3
60×180 (stationary)	0.2L	245.41	371.28	497.46	~ 5542
3×10	0.2L	275.34	431.79	453.22	~ 15.8
5×12	0.2L	271.87	445.16	445.7	~ 23.4
5×24	0.2L	262.17	457.61	457.7	~ 45
n/a	0.4L	208.28	n/a	9649.31	–
60×180 (stationary)	0.4L	321.75	765.69	947.55	–
3×10	0.4L	347.33	637.23	844.14	–
5×12	0.4L	346.48	841.53	857.77	–
5×24	0.4L	331.61	832.03	842.28	–

Fig. 15 Four designs obtained when using 0.4L damage patches. Top left: 60×180 stationary damage patches. Top right: 3×10 moving damage patches. Bottom left: 5×12 moving damage patches. Bottom right: 5×24 moving damage patches



other and have the same principle design. The compliance for the damaged structure obtained from the damage maps is similar as well except for the optimization with 60×180 stationary damage patches. The compliance for the undamaged structures and the compliance for the damaged structure can be found in Table 2. As one can see in Fig. 15, the designs are very similar, but once again the design with stationary damage patches in Fig. 15a has the worst damaged compliance. Here, the moving damage patches make for much better designs at lower computational cost.

4.3 Summary

A summary of data for all the designs considered above, including the nominal and worst compliance found during the optimization and the worst compliance found from the damage map, is given in Table 2. As can be seen in the table, the compliance for the structures optimized using a 0.2L damage patch is almost 20 times better than that for the nominal design. The worst compliance found during the optimization is also close to the worst compliance found from the damage map. For the 0.4L damage, the compliance optimized with damage is more than 10 times better than that for the nominal design. Worth noting is that when a low number of damage patches were used, $3 \times 10 = 30$, the worst compliance was not found during the optimization but the structure is still just as damage tolerant as the designs where the worst damage found during the optimization was close to the worst compliance from the damage map.

5 Additional examples and extensions

To further investigate the capability of the algorithm, some additional examples, namely the MBB beam and the L-bracket, are considered. For these examples, the MMC-damage formulation is the same as in the previous section where only the centroid is used as design variable for the inner problem.

Furthermore, the MMC formulation can easily be modified to include both a change in length, L_i , and thickness, t_i , as well as rotation, θ_i , of the components (c.f.

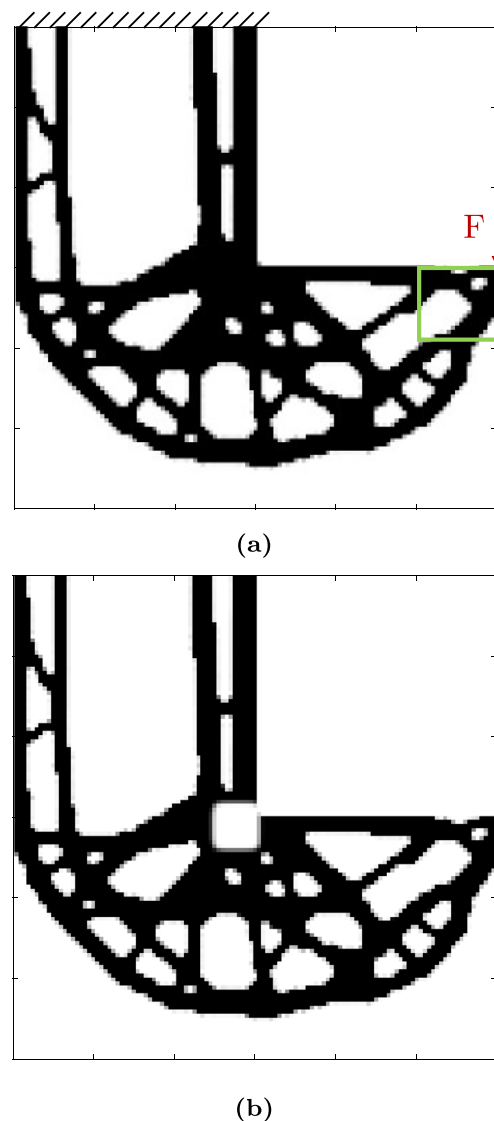


Fig. 16 The optimized L-bracket (a) and the worst damage (b)

Figure 1). The added variables do make the optimization problem increasingly non-convex and it becomes difficult to verify, with, e.g. a damage map, if the worst damage is found. However, it is clear that the optimized designs are more fail-safe than an optimized design without regarding damage.

If nothing else is stated the parameters are the same as, found in Table 1.

5.1 L-bracket with 0.2L damage patches

For the L-bracket shown in Fig. 16a, a box (in green) was added as a safe zone around the point where the force was applied. This was done to keep the damage from disconnecting the force from the structure. The design domain was $2L \times 2L$ with an inactive zone of $L \times L$ in the top right. The number of elements was 120×120 . The initial damage patches were distributed evenly in the domain and a total of 47 damage patches were considered. The same type of damage patches as in the 0.2L damage cantilever setup was used. The optimized structure can be seen in Fig. 16a, and the worst damage at the inward corner can be seen in Fig. 16b. The compliance for the undamaged structure was 118.66 and for the damaged structure it was 236.23.

5.2 MBB beam with 0.2L damage patches

The other example is the MBB beam with the same setup as for the L-bracket. For the MBB beam, two safe zones (in green) had to be applied to not disconnect the force as well as the support in the bottom right in Fig. 17a since this too would make the compliance tend to infinity. The number of elements was 120×40 . The total number of damage patches was 72 which were evenly distributed. The optimized structure can be seen in Fig. 17a and the worst damage, located slight below the center of the domain, can be seen in Fig. 17b. The compliance for the undamaged structure was 283.67 and for the damaged structure it was 481.12.

5.3 Varying patch size

In this example, the MMC is allowed to change its position as well as its length, L and width, t . To not get too long, thin damage patches which can cut too many structural members, a bound is put on the length $L_i \in [0.05, 0.2]$ as well as the thickness $t_i \in [0.05, 0.2]$. Without these bounds, a sufficiently long patch could cut away all the material at the supports and the compliance would tend to infinity in the same manner as disconnecting the force would. To maintain a sensible size of the damage, a bound on the area which it is allowed to cover was added. That is $L_i t_i \leq \text{MaxArea}$ was added into problem (7). This bound is always fulfilled when

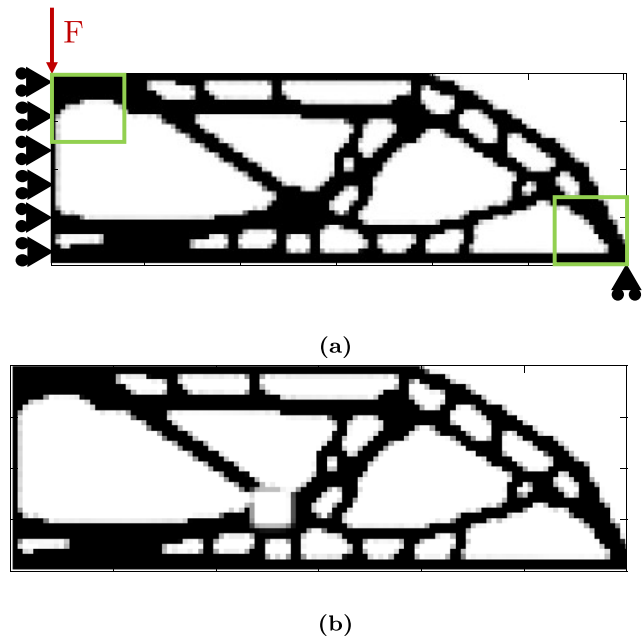


Fig. 17 The optimized MBB beam (a) and the worst damage (b)

only varying the centroid and rotation. The maximal area was set to be equal to the area of a square damage patch with side lengths of $0.2L$. The amount of damage patches was set to 5×12 and they were evenly distributed throughout the domain. The optimized design can be seen in Fig. 18a. The design resembles the designs in Fig. 15 and this is due to the maximum length of the damage patch being the same as one of the sides of the larger damage patches even though the area covered is smaller in this example. The final damage distribution is shown in Fig. 18b.

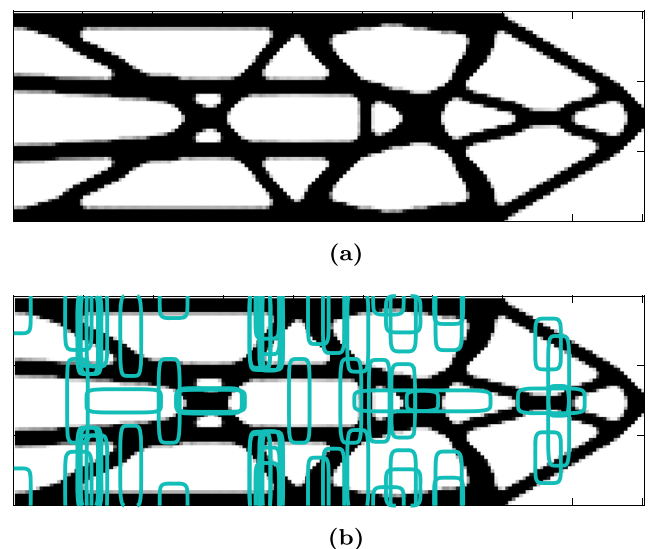


Fig. 18 The optimized cantilever beam with 4 damage variables (a) and the damage distribution (b)

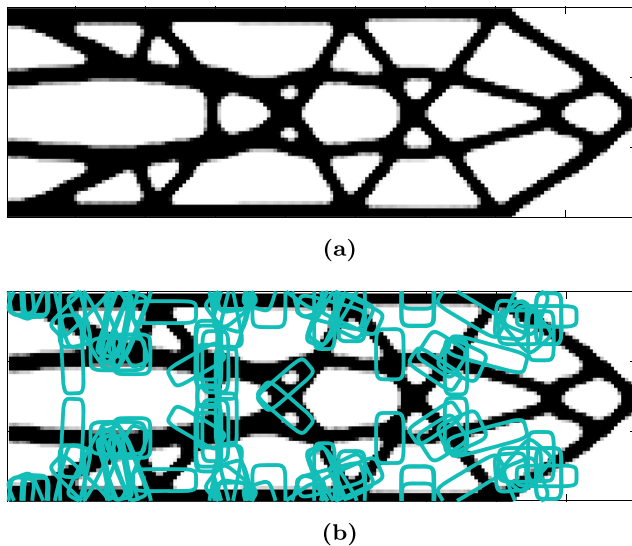


Fig. 19 The optimized cantilever beam with 5 variables (a) and the damage distribution (b)

5.4 Varying patch size and orientation

Additionally, one can let the components rotate as stated in (4). The same bounds as in the previous section are used with added bounds on the angle $\theta \in [-\pi/2, \pi/2]$ for each patch. The damage patches were initially distributed as crosses similar to Fig. 2 to try and capture as many bad damage locations as possible. There were 5×12 crosses evenly distributed in the domain resulting in 120 damage patches. Figure 19a shows the optimized structure and Fig. 19b shows the damage distribution. One can see that the damage patches would rather orientate in such a way that they cut as many structural members as possible and not along the structural beams. This is due to the fact that cutting a structural member with a thin damage is almost equally as bad as if the entire beam was removed (if it wasn't for E_{min} in (10), ensuring a small stiffness also in voids, it would be exactly as bad).

6 Concluding remarks

We have proposed a method for TO of fail-safe structures based on a standard density-based description of the design and a mesh-independent MMC representation of damage. The numerical examples show that the method can be effective for generating fail-safe designs.

When comparing the compliance found during the optimization to the damage map compliance in Sections 4.1 and 4.2, the worst damage was never truly found. However, for a sufficient amount of damage patches, it was close, and for the smaller damage size, it was closer than when using stationary patches. Furthermore, even though the worst

damage is not found, the structure still has more than a 10-fold improvement in terms of compliance compared to the nominal design, even for the design with 3×10 damage patches of size $0.4L$. For the smaller $0.2L$ damage size, the optimization proved successful at finding the worst damage with as little as 30 damage patches and yielded a 20-fold improvement in worst compliance compared to the nominal design, see Table 2.

As shown in Section 5, the proposed algorithm also works for different examples and it is easy to implement other MMC descriptions in the proposed framework. The structures optimized with additional damage variables, seen in Figs. 18a and 19a, resemble the structures obtained for a $0.4L$ damage patch seen in Fig. 15 even though each damage only occupies one fourth of the area. The reason is that the damage patches prefer to be as long as possible in one direction to cut as many structural members as possible. More variables added makes it more difficult to know if one has found the worst damage or not but the structures are more fail-safe none the less.

Acknowledgements We thank Krister Svanberg for sharing his MMA code.

Funding Open access funding provided by Linköping University. This work was financed by the Swedish Research Council under grant agreement no. 2019-04615.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Replication of results A complete description of the mathematical model together with numerical values for all model parameters is provided in the article.

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

- Ambrozkiwicz O, Kriegesmann B (2018) Adaptive strategies for fail-safe topology optimization. In: International conference on engineering optimization. Springer, pp 200–211

- Ambroziewicz O, Kriegesmann B (2020) Density-based shape optimization for fail-safe design. *Journal of Computational Design and Engineering*
- Andreassen E, Clausen A, Schevenels M, Lazarov B, Sigmund O (2011) Efficient topology optimization in MATLAB using 88 lines of code. *Struct Multidiscip Optim* 43
- Bendsoe M, Kikuchi N (1988) Generating optimal topologies in structural design using a homogenization method. *Comput Meth Appl Mech Eng* 72:197–224
- Bourdin B (2001) Filters in topology optimization. *Int J Numer Methods Eng* 50(9):2143–2158
- Christensen P, Klarbring A (2008) An introduction to structural optimization, vol 153. Springer Science & Business Media, Berlin
- Coniglio S, Morlier J, Gogu C, Amargier R (2019) Generalized geometry projection: A unified approach for geometric feature based topology optimization. *Arch Comput Methods Eng* 1–38
- da Silva G, Cardoso E, Beck A (2019) Non-probabilistic robust continuum topology optimization with stress constraints. *Struct Multidiscip Optim* 59(4):1181–1197
- Deaton J, Grandhi R (2014) A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Struct Multidiscip Optim* 49(1):1–38
- Dou S (2019) Porous structural design for additive manufacturing and improved damage tolerance. In: 13th world congress of structural and multidisciplinary optimization. Springer
- Guo X, Zhang W, Zhong W (2014) Doing topology optimization explicitly and geometrically—a new moving morphable components based framework. *J Appl Mech* 81(8)
- Jansen M, Lombaert G, Schevenels M, Sigmund O (2014) Topology optimization of fail-safe structures using a simplified local damage model. *Struct Multidiscip Optim* 49(4):657–666
- Klarbring A, Strömberg N (2012) A note on the min-max formulation of stiffness optimization including non-zero prescribed displacements. *Struct Multidiscip Optim* 45(1):147–149
- Kreisselmeier G, Steinhauser R (1979) Systematic control design by optimizing a vector performance index. *IFAC Proc Vol* 12(7):113–117. [https://doi.org/10.1016/S1474-6670\(17\)65584-8](https://doi.org/10.1016/S1474-6670(17)65584-8), iFAC Symposium on computer Aided Design of Control Systems, Zurich, Switzerland, 29–31 August
- Niu MCY (1997) Airframe stress analysis and sizing. Hong Kong Conmlit Press Limited
- Niu MCY (1998) Airframe structural design. Lockheed Aeronautical System Company
- Rietz A (2001) Sufficiency of a finite exponent in SIMP (power law) methods. *Struct Multidiscip Optim* 21(2):159–163
- Sigmund O, Petersson J (1998) Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Struct Optim* 16(1):68–75
- Stolpe M (2019) Fail-safe truss topology optimization. *Struct Multidiscip Optim* 60(4):1605–1618
- Svanberg K (1987) The method of moving asymptotes—a new method for structural optimization. *Int J Numer Methods Eng* 24(2):359–373
- Wang F, Lazarov B, Sigmund O (2011) On projection methods, convergence and robust formulations in topology optimization. *Struct Multidiscip Optim* 43(6):767–784
- Wang H, Liu J, Wen G, Xie Y (2020) The robust fail-safe topological designs based on the von Mises stress. *Finite Elem Anal Des* 171:103376
- Wu J, Aage N, Sigmund O (2018) Infill optimization for additive manufacturing - approaching bone-like porous structures. *IEEE Trans Visual Comput Graph* 24(2):1127–1140. <https://doi.org/10.1109/TVCG.2017.2655523>
- Zhang W, Yuan J, Zhang J, Guo X (2016) A new topology optimization approach based on Moving Morphable Components (MMC) and the ersatz material model. *Struct Multidiscip Optim* 53(6):1243–1260
- Zhou M, Fleury R (2016) Fail-safe topology optimization. *Struct Multidiscip Optim* 54(5):1225–1243

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