Computers and Structures 259 (2022) 106696

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

A worst-case approach to topology optimization for maximum stiffness under uncertain boundary displacement

Carl-Johan Thore

Division of Solid Mechanics, Department of Management and Engineering, Linköping University, Sweden

ARTICLE INFO

Article history:
Received 23 June 2021
Accepted 8 October 2021
Available online 28 October 2021

Keywords:
Topology optimization
Uncertainty
Eigenvalue
Matrix inequality
Non-smooth

ABSTRACT

We present a worst-case approach to topology optimization (TO) for maximum stiffness under boundary displacement parametrized by a matrix-valued scaling function times an uncertain vector giving its direction. The objective function in the TO problem is the minimum of the potential energy maximized over the set of boundary displacements, which in the absence of prescribed loads means maximizing the reaction loads arising from enforcing the boundary displacement. It is shown that the TO problem can be cast as the minimization of the maximum eigenvalue of a matrix depending on solutions to a small number of (linear elastic) state problems. Numerical solution of this potentially non-smooth problem using algorithms for smooth optimization, a non-linear semi-definite programming reformulation, and a non-smooth bundle method is discussed and tested.

1. Introduction

Topology optimization (TO) has become a popular tool for design of mechanical structures [10,13,24]. Topology-optimized structures are usually highly efficient under the conditions they are optimized for but may perform less than optimally, even poorly, under other conditions. To obtain structures that perform acceptably over a wide range of relevant scenarios it is thus necessary to account explicitly for uncertainty when formulating a TO problem. Among the many sources of uncertainty in mechanical design we have here chosen to consider boundary displacements. More precisely, we treat TO for maximum stiffness in linear elasticity with partially prescribed displacement on parts of the boundary of the design domain (partially prescribed meaning that the location is assumed known but the direction of the displacement is uncertain). To achieve a robust structure which is a stiff as possible we seek a design which minimize the minimum (average) reaction load obtained as the boundary displacement vary in an uncertainty set (see problem (6) below). This approach is related to some earlier work on worst-case load-uncertainty, see for example [56,33,58], but prescribing displacements rather than loads make the derivation and analysis somewhat different. In addition we also show well–posedness of the continuum problem and demonstrate numerical solution using a non-smooth method.

Previous work on TO, and other forms of structural optimization, under uncertainty have treated uncertain loads [5–7,47,8,9,28,37,56,18,33,29,65,19,63,65,58,44,22,62], geometry [50,31,41] and material properties [1,11,2,3,43,35,36,66,34,20,52,27], using both worst-case-oriented and stochastic methods [9]. Unlike stochastic methods, such as reliability-based design optimization [39], worst-case-oriented methods assume nothing about the probability distribution of the uncertain data, save for the data being confined to a bounded uncertainty set. In practise, detailed and accurate probability distributions may be difficult to obtain, so this speaks in favour of using worst-case oriented methods. In addition, such methods guarantee structural performance for all data in the uncertainty set, whereas a corresponding stochastic method can only give the same guarantee with a certain probability. On the other hand, worst-case-oriented methods assume the ability to compute the worst response and this amounts to solving a non-convex optimization problem to global optimality – in general a very difficult task. When using worst-case oriented methods one is thus limited to problems with special structure, for example where computing the worst response can be formulated as an eigenvalue problem [58], where the uncertainty set is finite with a very small number of elements [50], or where the problem actually admits a convex formulation [27]. The problem treated in the present work belongs in the first category, for the objective in our TO problem can be cast as that of finding a maximum eigenvalue.

In the following we first present the problem in the infinite-dimensional continuum setting and show existence of a solution for a general description of the uncertain data. This description is then specialized in Section 2.2, whereafter we proceed with the finite element (FE) discretization, and eventually end up with the

E-mail address: carl-johan.thore@liu.se

https://doi.org/10.1016/j.compstruc.2021.106696
0045-7949/

non-smooth problem (19) of minimizing the maximum eigenvalue-function of a matrix depending on the displacement. One way to handle non-smoothness numerically which appears popular at least in the engineering literature, is to simply ignore it and just apply a smooth optimization solver; this approach is tested below. In addition, we also try solving a smooth, non-linear semidefinite programming formulation and present some preliminary results from applying a simple non-smooth bundle method.

2. Continuum formulation

We present here the continuum version of the problem[1], formulated initially as the minimization of the maximum of the negative of the potential energy over the uncertainty set. (Depending on the reader's background it may be helpful to at least initially compare with the corresponding finite element matrix problem derived in Section 3, where starting with problem (17) one can use (13)–(16) to arrive at (19)).

The design domain $\Omega \subset \mathbb{R}^d, d = 1, 2$ or 3, is an open, bounded and connected domain with Lipschitz boundary $\partial\Omega$ with a subset $\Gamma_u, |\Gamma_u| > 0$, on which the displacement is prescribed. We follow the density-based approach to TO [13] and let the design be described by a function ρ, ideally only taking values 0 (no material) or 1 (material) in each point in the design domain. The design is restricted to the weakly* sequentially compact set $\mathscr{H} \subset L^\infty(\Omega)$. To ease notation we do not consider prescribed (non-zero) surface or body loads and assume that all d components of the displacement is prescribed on Γ_u.

Since there are no prescribed loads, the potential energy of the system is just one half of the strain-energy, i.e.,

$$\Pi(\rho; \boldsymbol{v}) = \frac{1}{2} a(\rho; \boldsymbol{v}, \boldsymbol{v}), \tag{1}$$

where, assuming small deformations and a linearly elastic material,

$$a(\rho; \boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \boldsymbol{E}(\rho)\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) \, \mathrm{d}V,$$

in which $\boldsymbol{\varepsilon}(\boldsymbol{u}) = (\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\mathsf{T})/2$ and: is the Frobenius inner product. The set of admissible displacements is given by

$$\boldsymbol{V}(\boldsymbol{u}_0) = \left\{ \boldsymbol{v} \in \boldsymbol{H}^1(\Omega) \mid \boldsymbol{v} = \boldsymbol{u}_0 \text{ on } \Gamma_u \right\},$$

where $\boldsymbol{u}_0 \in \boldsymbol{H}^{1/2}(\Omega)$. We make the non-restrictive assumptions that the constitutive tensor $\boldsymbol{E}(\cdot)$ is such that $a(\rho; \cdot, \cdot)$ is coercive on $\boldsymbol{V}(\boldsymbol{0})$ for every admissible design; that $\Pi(\cdot; \boldsymbol{v})$ is weakly* sequentially continuous on $\mathscr{H}$; and that it is monotone increasing in the sense that if $\rho_2 > \rho_1$ at a point in Ω, then at that point,

$$\boldsymbol{E}(\rho_2)\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon} > \boldsymbol{E}(\rho_1)\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}, \tag{2}$$

for every symmetric $d \times d$-matrix $\boldsymbol{\varepsilon}$. In practice, the weak* continuity can be ensured by using any of the standard regularization techniques in TO [16,42,30]. The monotonicity property, which is natural to assume, together with suitable constraints in $\mathscr{H}$, ensures that implicit penalization schemes such as SIMP or RAMP will work as intended [45]; see Section 2.1.1.

We first consider the problem of minimizing the potential energy, i.e. $\min_{\boldsymbol{v}\in\boldsymbol{V}(\boldsymbol{u}_0)} \Pi(\rho; \boldsymbol{v})$. Since the strictly convex function $\Pi(\rho, \cdot)$ is radially unbounded for every $\rho \in \mathscr{H}$ and $\boldsymbol{V}(\boldsymbol{u}_0)$ is weakly closed this problem has unique solution $\boldsymbol{u}$, characterized by the existence of a reaction load $\boldsymbol{\lambda} \in \boldsymbol{H}^{-1/2}(\Omega)$ such that equilibrium (principle of virtual work) and the boundary conditions hold, i.e. $\boldsymbol{u}$ solves

$$\begin{aligned} &a(\rho; \boldsymbol{u}, \boldsymbol{v}) = \langle \boldsymbol{\lambda}, \boldsymbol{v} \rangle_{\Gamma_u}, \quad \forall \boldsymbol{v} \in \boldsymbol{H}^1(\Omega) \\ &\boldsymbol{u} \in \boldsymbol{V}(\boldsymbol{u}_0) \end{aligned} \tag{3}$$

where if $\boldsymbol{\lambda}$ happens to live in $\boldsymbol{L}^2(\Gamma_u)$, the duality pairing

$$\langle \boldsymbol{\lambda}, \boldsymbol{v} \rangle_{\Gamma_u} = \int_{\Gamma_u} \boldsymbol{\lambda}^\mathsf{T} \boldsymbol{v} \, \mathrm{d}s.$$

Now consider the following design problem for a given function $\boldsymbol{u}_0$:

$$\min_{\rho\in\mathscr{H}} \left\{ - \min_{\boldsymbol{v}\in\boldsymbol{V}(\boldsymbol{u}_0)} \Pi(\rho; \boldsymbol{v}) \right\} = \min_{\rho\in\mathscr{H}} -\frac{1}{2} \langle \boldsymbol{\lambda}(\rho), \boldsymbol{u}_0 \rangle_{\Gamma_u}, \tag{4}$$

where the equality follows by substituting (3) into (1). From this follows then the interpretation that we maximize a weighted average of the reaction load at the boundary. In the numerical examples in Section 5 we refer to (the approximation of) $\frac{1}{2}\langle \boldsymbol{\lambda}(\rho), \boldsymbol{u}_0 \rangle_{\Gamma_u}$ and its worst-case counterpart as "compliance".

2.1. The worst-case problem

Introducing uncertain boundary displacements, confined to a set $\boldsymbol{D} \subset \boldsymbol{H}^{1/2}(\Omega)$, leads to the following version of (4):

$$\min_{\rho\in\mathscr{H}} \max_{\boldsymbol{u}_0\in\boldsymbol{D}} \left\{ - \min_{\boldsymbol{v}\in\boldsymbol{V}(\boldsymbol{u}_0)} \Pi(\rho; \boldsymbol{v}) \right\} = \min_{\rho\in\mathscr{H}} \underbrace{\max_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} - \Pi(\rho; \boldsymbol{v})}_{\equiv c(\rho)}, \tag{5}$$

where the equality follows by noting that $-\min\{-f\} = \max f$, and the set $\boldsymbol{W}(\boldsymbol{u}_0) = \boldsymbol{D} \times \boldsymbol{V}(\boldsymbol{u}_0)$. A solution ρ^* (if any) to problem (5) is robust in the sense that, by definition of maximum (or supremum if the max is not achieved),

$$-\Pi(\rho^*, \tilde{\boldsymbol{v}}) \leqslant c(\rho^*) = \max_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} - \Pi(\rho^*; \boldsymbol{v}) \quad \forall (\tilde{\boldsymbol{u}}_0, \tilde{\boldsymbol{v}}) \in \boldsymbol{W}(\tilde{\boldsymbol{u}}_0);$$

i.e., no response will be worse than $c(\rho^*)$ for any of the data in $\boldsymbol{W}$.

Using (3), problem (5) can be re-written as

$$\min_{\rho\in\mathscr{H}} \max_{\boldsymbol{u}_0\in\boldsymbol{D}} -\frac{1}{2} \langle \boldsymbol{\lambda}(\rho), \boldsymbol{u}_0 \rangle_{\Gamma_u} \tag{6}$$

Since, using (3), $-\langle \boldsymbol{\lambda}(\rho), \boldsymbol{u}_0 \rangle_{\Gamma_u} = -\langle \boldsymbol{\lambda}(\rho), \boldsymbol{u}(\rho) \rangle_{\Gamma_u} = -a(\rho, \boldsymbol{u}(\rho), \boldsymbol{u}(\rho)) \leqslant 0$, we see that the inner maximization problem in (6) consists essentially in trying to *minimize* the reaction load. This implies that the set $\boldsymbol{D}$ must be such that $\boldsymbol{u}_0$ is bounded away from the zero-function (the trivial solution); essentially $\boldsymbol{D}$ should provide a lower bound on (some norm of) $\boldsymbol{u}_0$. This may seem counter-intuitive at first, but a $\boldsymbol{u}_0$ which minimizes the reaction loads gives, loosely speaking, a direction in which the structure is the least stiff.

2.1.1. Existence, convexity and monotonicity

We show here three easy-to-prove but important properties of problem (5).

1. As for the existence of a solution to (5), recall that $\mathscr{H}$ is weakly* sequentially compact, so existence of a solution follows if we can show that c is weakly* sequentially lower semi-continuous on this set. Following [15, p. 261] we note that the epigraph

$$\begin{aligned} \mathrm{epi}(c) &= \{(\rho, \alpha) \in \mathscr{H} \times \mathbb{R} \,|\, \alpha \geqslant c(\rho)\} \\ &= \{(\rho, \alpha) \in \mathscr{H} \times \mathbb{R} \,|\, \alpha \geqslant -\Pi(\rho; \boldsymbol{v}), \ \forall (\boldsymbol{u}_0, \boldsymbol{v}) \in \boldsymbol{W}(\boldsymbol{u}_0)\} \\ &= \cap_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} \{(\rho, \alpha) \in \mathscr{H} \times \mathbb{R} \,|\, \alpha \geqslant -\Pi(\rho; \boldsymbol{v})\}, \end{aligned}$$

where the first equality is direct from the definition of epigraph, the second comes from the definition of c and the third from the fact that $\boldsymbol{W}(\boldsymbol{u}_0)$ does not depend on ρ. Now since $\Pi(\cdot; \boldsymbol{v})$ is (weakly*) continuous, $-\Pi(\cdot; \boldsymbol{v})$ is lower semi-continuous, hence its epigraph is closed. Then $\mathrm{epi}(c)$, being the intersection of

[1] See for example [17] for the functional analytic background.

closed sets, is closed, whence it follows that c is weakly* sequentially lower semi-continuous on $\mathscr{H}$.

2. If Π is affine in its first argument, then

$$c(\rho) = \max_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} -\Pi(\rho;\boldsymbol{v})$$

is convex (in ρ) by virtue of being the supremum of a set of affine functions. In practice one is more interested in the non-convex case, but we consider in the beginning of Section 5 some convex instances as well.

3. Finally, from inequality (2) follows that

$$\begin{aligned}\Pi(\rho_1,\boldsymbol{v}) &= \int_\Omega \boldsymbol{E}(\rho_1)\boldsymbol{\varepsilon}(\boldsymbol{v}):\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}V < \int_\Omega \boldsymbol{E}(\rho_2)\boldsymbol{\varepsilon}(\boldsymbol{v}):\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}V\\ &= \Pi(\rho_2,\boldsymbol{v})\end{aligned}$$

so that

$$\begin{aligned}\rho_2 > \rho_1 \text{ a.e in } \Omega \Rightarrow &\max_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} -\Pi(\rho_1;\boldsymbol{v})\\ &> \max_{(\boldsymbol{u}_0,\boldsymbol{v})\in\boldsymbol{W}(\boldsymbol{u}_0)} -\Pi(\rho_2;\boldsymbol{v}),\end{aligned}$$

i.e. adding material results in a smaller objective value in (5). This means that Assumption 1 in [45] holds, and we can expect the usual SIMP or RAMP penalization schemes to work in the sense of yielding close to binary-valued solutions for sufficiently large values of the penalization parameters (in practise regularization such as filters [16] always leaves some regions of intermediate ρ-values).

Remark 1. A common stochastic approach to TO under uncertainty is to minimize a linear combination of the expected value and the standard deviation of the chosen response. Depending on the relative weight put on the standard deviation, such an objective may fail to be monotone in the above sense – in the words of Toriis [59] it lacks "physical consistency" – and implicit penalization schemes will then not work as intended. □.

2.2. Parametrization of the uncertain data

Having shown that problem (5) is well-posed in the sense of having at least one solution, we now turn to the question of computational tractability. In practise, the choice of the uncertainty set is limited by the requirement that a global maximizer can be found in reasonable time; if not, we don't have a worst-case approach (though optimized designs would still likely be more robust than if uncertainty is not accounted for at all).

We shall herein take the boundary displacements to be of the form

$$\boldsymbol{u}_0(\boldsymbol{x}) = \boldsymbol{Q}(\boldsymbol{x})\boldsymbol{\delta} \quad \text{on } \Gamma_u, \tag{7}$$

where $\boldsymbol{Q}:\Gamma_u\to\mathbb{R}^{d\times s}$ is a fixed, piecewise continuous matrix-valued function. The vector $\boldsymbol{\delta}$, which does not depend on position $\boldsymbol{x}$, is subjected to the constraint $\|\boldsymbol{\delta}\|\geqslant 1$, where $\|\cdot\|$ is the Euclidean norm. Fig. 1, in which $d=2$ and $s=2$, shows a simple example of how $\boldsymbol{Q}$ could be defined. In the figure,

$$\boldsymbol{Q}(x,y) = \begin{cases}\boldsymbol{0} \text{ if } x=0, y\in[0,H]\\ \boldsymbol{I} \text{ if } x=L, y\in[0,H],\end{cases}$$

where $\boldsymbol{I}$ is a 2×2 identity matrix. That $\boldsymbol{Q}$ is taken as the zero-matrix on the left side means that we are confident that the domain is rigidly attached to something whose movement is negligible.

With boundary displacement on the form (7), problem (5) is specialized to

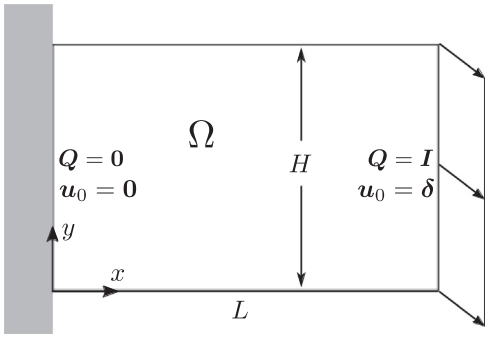

Fig. 1. Design domain with uncertain boundary displacement on the right side. The top and bottom side of the domain are unconstrained (traction-free).

$$\begin{aligned}\min_{\rho\in\mathscr{H}}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1}\left\{-\min_{\boldsymbol{v}\in\boldsymbol{V}(\boldsymbol{\delta})}\Pi(\rho;\boldsymbol{v})\right\} &= \min_{\rho\in\mathscr{H}}\\ &\times \max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} -\frac{1}{2}a\left(\rho;\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{Q}}\boldsymbol{\delta},\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{Q}}\boldsymbol{\delta}\right)\end{aligned} \tag{8}$$

where $\boldsymbol{V}(\boldsymbol{\delta}) = \left\{\boldsymbol{v}\in\boldsymbol{H}^1(\Omega)\mid \boldsymbol{v}=\boldsymbol{Q}(\boldsymbol{x})\boldsymbol{\delta} \text{ on } \Gamma_u\right\}$ and the equality is obtained by writing the solution to the innermost problem as $\boldsymbol{u}=\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{Q}}\boldsymbol{\delta}$, where the function $\tilde{\boldsymbol{Q}}$ is such that $\tilde{\boldsymbol{Q}}\boldsymbol{\delta}=\boldsymbol{Q}\boldsymbol{\delta}$ on Γ_u and $\tilde{\boldsymbol{u}}\in\boldsymbol{V}(\boldsymbol{0})$ satisfies

$$a\left(\rho;\tilde{\boldsymbol{u}},\boldsymbol{v}\right) = -a\left(\rho;\tilde{\boldsymbol{Q}}\boldsymbol{\delta},\boldsymbol{v}\right), \quad \forall\boldsymbol{v}\in\boldsymbol{V}(\boldsymbol{0}). \tag{9}$$

3. FE-discretization

Given a mesh $\{\Omega_e\}_{e=1}^m$, where $m=m(h)$ and h is the largest diameter of an element, such that $\overline{\Omega}=\cup\overline{\Omega_e}$, and $\mathscr{H}_h\subset\mathscr{H}$, we consider the following FE discretized version of (8):

$$\min_{\rho\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} -\frac{1}{2}a_h\left(\rho;\tilde{\boldsymbol{u}}_h+\tilde{\boldsymbol{Q}}_h\boldsymbol{\delta},\tilde{\boldsymbol{u}}_h+\tilde{\boldsymbol{Q}}_h\boldsymbol{\delta}\right), \tag{10}$$

where $\boldsymbol{u}_h=\tilde{\boldsymbol{u}}_h+\tilde{\boldsymbol{Q}}\boldsymbol{\delta}_h$, and $\tilde{\boldsymbol{u}}_h\in\boldsymbol{V}_h(\boldsymbol{0})$ solves

$$a_h\left(\rho;\tilde{\boldsymbol{u}},\boldsymbol{v}\right) = -a_h\left(\rho;\tilde{\boldsymbol{Q}}_h\boldsymbol{\delta},\boldsymbol{v}\right), \quad \forall\boldsymbol{v}\in\boldsymbol{V}_h(\boldsymbol{0}), \tag{11}$$

in which $a_h(\rho;\cdot,\cdot)$, with the subscript h being motivated by for example numerical quadrature, is assumed to be (for sufficiently small h) coercive on the Lagrange FE-approximation space $\boldsymbol{V}_h(\boldsymbol{0})\subset\boldsymbol{V}(\boldsymbol{0})$. With

$$\tilde{\boldsymbol{u}}_h = \sum_{i:\boldsymbol{x}_i\in\Omega\setminus\Gamma_u} N_i(\boldsymbol{x})\boldsymbol{u}_i \quad \text{and} \quad \tilde{\boldsymbol{Q}}_h\boldsymbol{\delta} = \sum_{i:\boldsymbol{x}_i\in\Gamma_u} N_i(\boldsymbol{x})\underbrace{\boldsymbol{Q}(\boldsymbol{x}_i)\boldsymbol{\delta}}_{\equiv\boldsymbol{d}_{0i}}, \tag{12}$$

where $\boldsymbol{x}_i$ is the position of node i and N_i are the basis functions of $\boldsymbol{V}_h$, we find that the vector $\boldsymbol{d}$ collecting the n degrees of freedom of $\boldsymbol{u}_h$ can be written as

$$\boldsymbol{d} = \boldsymbol{A}\tilde{\boldsymbol{d}}+\boldsymbol{B}\boldsymbol{d}_0, \tag{13}$$

where $\tilde{\boldsymbol{d}}$ contains the degrees of freedom of $\tilde{\boldsymbol{u}}_h$; $\boldsymbol{A}\in\mathbb{R}^{n\times(n-p)}$ and $\boldsymbol{B}\in\mathbb{R}^{n\times p}$ have full rank and satisfy $\boldsymbol{A}^T\boldsymbol{B}^T=\boldsymbol{0}$ and $\boldsymbol{B}^T\boldsymbol{B}=\boldsymbol{I}$; and, with the $\boldsymbol{d}_{0i}$-s defined in (12) and numbering the nodes on Γ_u as $1,2,\ldots,$

$$\boldsymbol{d}_0 = \begin{pmatrix}\boldsymbol{d}_{01}\\ \boldsymbol{d}_{02}\\ \vdots\end{pmatrix} = \begin{pmatrix}\boldsymbol{Q}(\boldsymbol{x}_1)\\ \boldsymbol{Q}(\boldsymbol{x}_2)\\ \vdots\end{pmatrix}\boldsymbol{\delta} = \boldsymbol{Q}\boldsymbol{\delta}. \tag{14}$$

Here and onwards, $\boldsymbol{Q} \in \mathbb{R}^{p\times s}$ is a matrix containing values of the function $\boldsymbol{Q}$ (recall Section 2.2) sampled at the nodes on Γ_u.

Using (13), the matrix version of (11) can now be written as

$$\tilde{\boldsymbol{K}}\tilde{\boldsymbol{d}} = -\hat{\boldsymbol{K}}\boldsymbol{d}_0, \tag{15}$$

where the stiffness matrix (sometimes called the reduced stiffness matrix) $\tilde{\boldsymbol{K}} = \boldsymbol{A}^\mathsf{T}\boldsymbol{K}\boldsymbol{A}$, the "thin" $(n-p)\times p$ matrix $\hat{\boldsymbol{K}} = \boldsymbol{A}^\mathsf{T}\boldsymbol{K}\boldsymbol{B}$, and $\boldsymbol{K}$ has entries $K_{ij} = a_h(\rho; N_i, N_j)$. It follows from the positive definiteness of $a_h(\rho;\cdot,\cdot)$ that $\boldsymbol{K}$ is positive definite on the null-space of the constraints. Since $\boldsymbol{A}$ has full rank it then follows that $\tilde{\boldsymbol{K}}$ is positive definite, hence invertible, so that

$$\tilde{\boldsymbol{d}} = -\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}}\boldsymbol{d}_0. \tag{16}$$

Letting $\boldsymbol{\rho} \in \mathscr{H}_h$ collect the degrees of freedom of the approximate design $\rho \in \tilde{\mathscr{H}}_h$, and using (13), (15) and (16), the matrix version of (10) can now be formulated as

$$\begin{aligned}\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} -\frac{1}{2}\boldsymbol{d}^\mathsf{T}\boldsymbol{K}\boldsymbol{d} &= \min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} \\ &\quad -\frac{1}{2}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}}\boldsymbol{d}_0 + \boldsymbol{B}\boldsymbol{d}_0\right)^\mathsf{T}\boldsymbol{K}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}}\boldsymbol{d}_0 + \boldsymbol{B}\boldsymbol{d}_0\right) \\ &= \min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} \\ &\quad -\boldsymbol{d}_0^\mathsf{T}\underbrace{\frac{1}{2}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}}\boldsymbol{d}_0 + \boldsymbol{B}\right)^\mathsf{T}\boldsymbol{K}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}} + \boldsymbol{B}\right)}_{=\boldsymbol{H}(\boldsymbol{\rho})}\boldsymbol{d}_0,\end{aligned} \tag{17}$$

where, for every admissible design, the matrix $\boldsymbol{H}(\boldsymbol{\rho})$ is clearly symmetric and positive definite. Using (14) we thus obtain

$$\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|\geqslant 1} -\boldsymbol{\delta}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{H}(\boldsymbol{\rho})\boldsymbol{Q}\boldsymbol{\delta} = \min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|=1} -\boldsymbol{\delta}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{H}(\boldsymbol{\rho})\boldsymbol{Q}\boldsymbol{\delta}, \tag{18}$$

where the equality follows from $-\boldsymbol{H}(\boldsymbol{\rho})$ being negative definite (the objective is thus monotone decreasing in $\|\boldsymbol{\delta}\|$ so that a maximizer must have the smallest allowable norm). This problem may, using the standard variational characterization of eigenvalues, and introducing $\boldsymbol{M} = -\boldsymbol{Q}^\mathsf{T}\boldsymbol{H}\boldsymbol{Q}$, be recast as

$$\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\lambda_{max}(\boldsymbol{M}(\boldsymbol{\rho})), \tag{19}$$

where $\lambda_{max}(\boldsymbol{M}(\boldsymbol{\rho}))$, and the associated eigenvector(s), solves

$$\boldsymbol{M}(\boldsymbol{\rho})\boldsymbol{\delta} = \lambda\boldsymbol{\delta}, \quad \|\boldsymbol{\delta}\| = 1. \tag{20}$$

To solve problem (19) we need to construct the matrix $\boldsymbol{M} = \boldsymbol{M}(\boldsymbol{\rho})$. To this end we recall the definition of $\boldsymbol{H}$ from (17) and write

$$\boldsymbol{M} = -\boldsymbol{Q}^\mathsf{T}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}} + \boldsymbol{B}\right)^\mathsf{T}\boldsymbol{K}\left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}} + \boldsymbol{B}\right)\boldsymbol{Q} = -\boldsymbol{W}^\mathsf{T}\boldsymbol{K}\boldsymbol{W}, \tag{21}$$

where

$$\boldsymbol{W} = \left(-\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}} + \boldsymbol{B}\right)\boldsymbol{Q} = -\boldsymbol{A}\tilde{\boldsymbol{K}}^{-1}\hat{\boldsymbol{K}}\boldsymbol{Q} + \boldsymbol{B}\boldsymbol{Q} = \boldsymbol{A}\tilde{\boldsymbol{W}} + \boldsymbol{B}\boldsymbol{Q},$$

in which the matrix $\tilde{\boldsymbol{W}}$ satisfies

$$\tilde{\boldsymbol{K}}(\boldsymbol{\rho})\tilde{\boldsymbol{W}} = -\hat{\boldsymbol{K}}(\boldsymbol{\rho})\boldsymbol{Q}. \tag{22}$$

The cost for evaluating $\boldsymbol{M}(\boldsymbol{\rho})$ in (19) is thus essentially that of solving s linear systems with the same stiffness matrix. Note that each column i of $\tilde{\boldsymbol{W}}$ can be interpreted as a (reduced) displacement vector for the load $-\hat{\boldsymbol{K}}\boldsymbol{Q}_{\cdot i}$, where $\boldsymbol{Q}_{\cdot i}$ is the i:th column of $\boldsymbol{Q}$.

Remark 2. We have here described the theory in terms of a "reduced" stiffness matrix $\tilde{\boldsymbol{K}}$. In practise one can of course also work with a "full-size" matrix, handling the Dirichlet conditions by for example zeroing rows and columns of the prescribed degrees of freedom and placing ones in the corresponding diagonal position of the stiffness matrix (this is the approach used for the 3D examples below). Note also that the use of Lagrange FEs is not required for our approach. □

3.1. Retrieving the worst-case displacement

The worst-case boundary displacement(s) for a given design is $\boldsymbol{d}_{0,worst} = \boldsymbol{Q}\boldsymbol{\delta}_{max}$, where $\boldsymbol{\delta}_{max} = \boldsymbol{\delta}_{max}(\boldsymbol{\rho})$ is an eigenvector associated with the maximum eigenvalue in (20). In case the multiplicity of this eigenvalue is greater than one, the eigenvectors are not unique and there are infinitely many $\boldsymbol{d}_0$:s that give the same worst response. Given $\boldsymbol{d}_{0,worst}$, the worst-case displacement is obtained using (13) as $\boldsymbol{d}_{worst} = \boldsymbol{A}\tilde{\boldsymbol{d}}_{worst} + \boldsymbol{B}\boldsymbol{d}_0$, where $\tilde{\boldsymbol{d}}_{worst}$ satisfies $\tilde{\boldsymbol{K}}\tilde{\boldsymbol{d}}_{worst} = -\hat{\boldsymbol{K}}\boldsymbol{d}_{0,worst}$.

4. Numerical solution

For the numerical solution of problem (19) we shall, in Sections 4.1, 4.2 and 4.3, consider three methods.

Common to the considered methods is the use of first-order sensitivity information, a necessity in TO [51]. To this end, note that since $\boldsymbol{M}$ is smooth in $\boldsymbol{\rho}$, the maximum-eigenvalue function is Lipschitz-continuous, hence admits a subdifferential given by [46, Theorem 3]

$$\partial\lambda_{max} = \left\{\boldsymbol{s}\in\mathbb{R}^{n_d} \mid s_i = \boldsymbol{P}^T\frac{\partial\boldsymbol{M}}{\partial\rho_i}\boldsymbol{P} : \boldsymbol{U},\ \boldsymbol{U} = \boldsymbol{U}^T \in \mathbb{R}^{t\times t},\ \mathrm{tr}\boldsymbol{U} = 1,\ \boldsymbol{U}\succeq\boldsymbol{0}\right\}, \tag{23}$$

in which the vectors $\boldsymbol{s}$ are subgradients, n_d is the number of design variables, and t is the multiplicity of the maximum eigenvalue. The columns of $\boldsymbol{P}\in\mathbb{R}^{s\times t}$ are (orthonormal) eigenvectors associated with λ_{max}. In the case $t = 1$, giving $\boldsymbol{U} = 1$, $\partial\lambda_{max}$ reduces to the usual gradient whose i:th component is

$$[\nabla\lambda_{max}]_i = \boldsymbol{p}^\mathsf{T}\frac{\partial\boldsymbol{M}}{\partial\rho_i}\boldsymbol{p},$$

where $\boldsymbol{p}$ is the associated eigenvector. Recalling (21) one finds after some calculations that

$$\frac{\partial\boldsymbol{M}}{\partial\rho_i} = -\boldsymbol{W}^\mathsf{T}\frac{\partial\boldsymbol{K}}{\partial\rho_i}\boldsymbol{W},$$

so that, with $\boldsymbol{u} = \boldsymbol{W}\boldsymbol{p}$,

$$[\nabla\lambda_{max}]_i = -\boldsymbol{u}^\mathsf{T}\frac{\partial\boldsymbol{K}}{\partial\rho_i}\boldsymbol{u}. \tag{24}$$

For definiteness we now take the approximate design to be element-wise constant, with elemental values collected in $\boldsymbol{\rho}\in\mathbb{R}^m$, and subjected to an upper bound γV on its total volume. Here V is the volume of the design domain and $\gamma\in(0,1)$. The set of admissible designs in (19) is thus given by

$$\mathscr{H}_h = \left\{\boldsymbol{\rho}\in\mathbb{R}^m \mid 0\leqslant\rho_e\leqslant 1,\ e = 1,\ldots,m,\ \sum_{e=1}^m\rho_e V_e\leqslant\gamma V\right\}. \tag{25}$$

The system of constraints defining $\mathscr{H}_h$ in (25) satisfies Slater's constraint qualifier [4, Def. 5.38] whence a solution $\boldsymbol{\rho}$ to (19) must satisfy [21]

$$\partial\lambda_{max}(\boldsymbol{\rho}) + \kappa\left(\sum_{e=1}^m\rho_e V_e - \gamma V\right) + \sum_{e=1}^m\left[\gamma_e\rho_e + \gamma_{e+m}(\rho_e - 1)\right]\in 0$$

for some $\kappa\geqslant 0$ and $\gamma_e, \gamma_{e+m}\geqslant 0, e = 1,\ldots,m$. If λ_{max} happens to be differentiable at a solution then this reduces to the standard Karush-Kuhn-Tucker optimality conditions.

4.1. NLP-formulation

The first approach to a numerical solution is to simply ignore the potential non-differentiability and treat (19) as a standard, smooth optimization problem using (24) for the derivatives. A motivation for this could be that while theoretical analyses [48,57] suggest that non-smoothness is in fact expected to occur precisely at optimal solutions, this is only so if there is enough freedom in choosing the design. Since this freedom depends not only on the number of elements in $\boldsymbol{\rho}$ and explicit constraints, but also on the shape of the design domain, loads and other boundary conditions, it may be argued that in general in TO there isn't enough freedom to obtain non-smoothness (for example, depending on the aspect ratio of the design-domain it may be impossible to obtain a design performing equally well for every possible loading direction).

4.2. Non-linear semidefinite programming

Introducing an auxiliary variable z, problem (19) may be recast as the following smooth, non-linear semi-definite programming (SDP) problem (c.f. [58]):

$$\begin{aligned}&\min_{\boldsymbol{\rho}\in\mathscr{H}_h, z\leqslant 0} z\\ &\text{s.t. } z\boldsymbol{I}-\boldsymbol{M}(\boldsymbol{\rho})\succeq\boldsymbol{0},\end{aligned}\tag{26}$$

where $\boldsymbol{I}$ is an identity matrix and "$\succeq\boldsymbol{0}$" means that the matrix to the left should be positive semi-definite. Unfortunately, to the best of our knowledge, the only software tailored for this type of problem is PenLab/PENNON [26], which requires exact second-derivative information and whose development appears to have ceased. An alternative is to reformulate (26) as a smooth NLP, for example based on Cholesky- [58] or LDL-factorizations [61,14]. Using the fact that a matrix is positive semi-definite if and only if it admits a Cholesky factorization we consider the following version of (26):

$$\begin{aligned}&\min_{\boldsymbol{\rho}\in\mathscr{H}_h, z\leqslant 0, \boldsymbol{L}\in\mathbb{L}^s} z\\ &\text{s.t. } z\boldsymbol{I}-\boldsymbol{M}(\boldsymbol{\rho})=\boldsymbol{L}\boldsymbol{L}^{\mathsf{T}},\end{aligned}\tag{27}$$

where $\mathbb{L}^s$ is the space of real-valued, lower triangular $s\times s$-matrices with non-negative diagonal entries. Unfortunately we now have a set of non-affine equality constraints, so the feasible set will always be non-convex even if the original problem is convex (which it is not in TO). On the other hand, every local minimum in (26) is a local minimum in (27) and vice versa (cf. [58]), so the non-convexity is not necessarily a practical issue.

Another drawback of formulation (27) is that it cannot be solved efficiently with the Method of Moving Asymptotes (MMA) [54], arguably one of the most popular optimization methods for TO. Therefore we use below the code IPOPT [64]. An advantage of this is that unlike the standard MMA, IPOPT is a globally convergent method, meaning that convergence to a stationary point is guaranteed under mild assumptions.

4.2.1. Sequential linear SDP

Inspired by sequential convex programming methods such as MMA one may consider solving a sequence of convex approximations of (26). One way to achieve a convex approximating problem is to make a linear approximation of each component function in the matrix inequality in (26) (MMA-type approximations of the component functions would yield a non-convex matrix constraint), leading to a linear, hence convex matrix inequality. In detail, defining $\boldsymbol{A}(\boldsymbol{x})\equiv z\boldsymbol{I}-\boldsymbol{M}(\boldsymbol{\rho})$ and linearizing around a point $\boldsymbol{x}^k=(z^k,\boldsymbol{\rho}^k)$ gives the convex problem

$$\begin{aligned}&\min_{\Delta\boldsymbol{x}=(\Delta\boldsymbol{\rho},\Delta z)} z^k+\Delta z\\ &\text{s.t. } \begin{cases}\boldsymbol{A}(\boldsymbol{x}^k)+\sum_{i=1}^{m+1}\frac{\partial\boldsymbol{A}}{\partial x_i}(\boldsymbol{x}^k)\Delta x_i\succeq\boldsymbol{0}\\ \boldsymbol{\rho}^k+\Delta\boldsymbol{\rho}\in\mathscr{H}_h,\ z^k+\Delta z\leqslant 0.\end{cases}\end{aligned}\tag{28}$$

(In practise one has to impose, for example, "move limits" on $\Delta\boldsymbol{x}$ to ensure convergence.) Linear SDP-problems of the type (28) can be solved by specialized software such as SeDuMi [53], SDPT3 [60] or MOSEK. Unfortunately these codes are not efficient for the kind of large-scale problems one has in TO, hence this a approach is not considered in the numerical examples below.

4.3. Non-smooth optimization

A third option is to actually treat (19) using methods designed for non-smooth optimization. Such methods can, roughly speaking, be divided into two classes [38]: subgradient and bundle methods. Inspired by [40,67] we consider here a so-called proximal bundle method, many details of which are described in [32, Chapter 15].

The basic idea of our bundle method is to solve a sequence, $k=1,2,\ldots$, of approximating problems of the form

$$\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\left\{f_k(\boldsymbol{\rho})+\frac{\mu_k}{2}\|\boldsymbol{\rho}-\boldsymbol{\rho}_k\|^2\right\},\tag{29}$$

where the parameter $\mu_k>0$ is used to vary the cost of moving from the current, so-called stability center $\boldsymbol{\rho}_k$. The model function f_k, approximating the maximum-eigenvalue function $f=f(\boldsymbol{\rho})=\lambda_{max}(\boldsymbol{M}(\boldsymbol{\rho}))$, should be such that (29) is simple to solve. Here we take

$$f_k(\boldsymbol{\rho})=f(\boldsymbol{\rho}_k)+\max_{i\in J_k}\left[-\beta_{ik}+\boldsymbol{s}_i^{\mathsf{T}}(\boldsymbol{\rho}-\boldsymbol{\rho}_k)\right],$$

where $\boldsymbol{s}_i$ is one of the subgradients from $\partial\lambda_{max}(\boldsymbol{\rho}_i)$ defined in (23) and $J_k\subset\{1,\ldots,k\}$. The so-called subgradient locality measure β_{ik} is defined as [38]

$$\beta_{ik}=\max\left\{|f(\boldsymbol{\rho}_k)-f(\boldsymbol{\rho}_i)+\boldsymbol{s}_i^{\mathsf{T}}(\boldsymbol{\rho}_i-\boldsymbol{\rho}_k)|,\gamma\|\boldsymbol{\rho}_k-\boldsymbol{\rho}_i\|\right\},\tag{30}$$

where $\gamma\geqslant 0$ is user-specified parameter (which can be taken as zero in the convex case).

The bundle method tested herein can be described as follows:

1. Given $\gamma\geqslant 0, M>0, \ell_{max}>0$, initial guess $\boldsymbol{\rho}_0$, and $\boldsymbol{s}_0\in\partial\lambda_{max}(\boldsymbol{\rho}_0)$, set $k=0$. Then
2. Solve (29) for a given μ_k. Denote by $\hat{\boldsymbol{\rho}}$ the solution to this problem.
3. If stopping criteria satisfied, STOP (with $\hat{\boldsymbol{\rho}}$ as the solution). Otherwise go to Step 4.
4. If $f(\hat{\boldsymbol{\rho}})-f(\boldsymbol{\rho}_k)\leqslant -M|f(\boldsymbol{\rho}_k)-f_k(\hat{\boldsymbol{\rho}})|$, set stability center $\boldsymbol{\rho}_{k+1}=\hat{\boldsymbol{\rho}}$ and $k=k+1$ and go to Step 2. Otherwise set $\boldsymbol{\rho}_{k+1}=\boldsymbol{\rho}_k$ and go to Step 5.
5. If $|J_k|=\ell_{max}$, remove the two elements with the largest β_{ik} from the bundle and update J_k.
6. Add $\boldsymbol{s}_i\in\partial\lambda_{max}(\hat{\boldsymbol{\rho}})$ and β_{ik} to the bundle. Then set $k=k+1$ and go to Step 2.

In Step 4 of this algorithm we check whether the actual decrease of f is sufficiently large compared to the predicted change (we use here the absolute value because for non-convex problems $f(\boldsymbol{\rho}_k)-f_k(\hat{\boldsymbol{\rho}})$ may become negative). If so, we trust the model and just move the stability center. If not, we "refine" the model by adding, in Step 6, one more element to the bundle and solve the approximating problem again.

The choice of the stabilization parameter μ_k in Step 2 can have a significant impact on the performance of the algorithm – setting it too small can lead to short steps and slow convergence, but setting it too large can lead to non-convergence. The strategy used in our numerical example is to start with a "small" (relative to $\lambda_{max}(\boldsymbol{\rho}_0)$) value which is increased by 10 percent if the objective value at the new trial point $\hat{\boldsymbol{\rho}}$ is greater that the objective value at the current stability center, i.e. if the objective, contrary to our desire, increases. In the numerical examples we set $\mu_0 = 10^{-4}\lambda_{max}(\boldsymbol{\rho}_0)$. We note for potential future work that various procedures for adaptive updating of μ_k have been proposed [49,38].

Besides evaluating the original objective function, the most computationally costly part of the algorithm is the solution of problem (29). This problem can be reformulated as the smooth, quadratic optimization problem

$$\begin{aligned} &\min_{(\boldsymbol{\rho},r)\in\mathscr{H}_h\times\mathbb{R}} \; r+\mu_k\|\boldsymbol{\rho}-\boldsymbol{\rho}_k\|^2 \\ &\text{s.t. } r \geqslant f(\boldsymbol{\rho}_k)-\beta_{ik}+\boldsymbol{s}_i^{\mathsf{T}}(\boldsymbol{\rho}-\boldsymbol{\rho}_k), \quad \forall i\in J_k, \end{aligned} \tag{31}$$

from which we derive the Lagrange dual problem

$$\begin{aligned} &\max_{\boldsymbol{\lambda}\geqslant\boldsymbol{0}} \; -\boldsymbol{b}^{\mathsf{T}}\boldsymbol{\lambda}+\sum_{i=1}^{m}\min_{0\leqslant\rho_i\leqslant 1}\{\tfrac{1}{2}\mu_k\rho_i^2-\mu_k\rho_{ki}\rho_i+\boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{A}_i\rho_i\} \\ &\text{s.t. } \boldsymbol{A}_n^{\mathsf{T}}\boldsymbol{\lambda}=-1, \end{aligned} \tag{32}$$

which is solved numerically to obtain solutions to (29). In (32), $\boldsymbol{A}$ and $\boldsymbol{b}$ are defined by the volume constraint in $\mathscr{H}_h$ and the explicit linear constraints in (31). The gradient of the objective function in the dual problem, call it $\tilde{L}$, with respect to $\boldsymbol{\lambda}$ is

$$\nabla\tilde{L}=-\boldsymbol{b}+\sum_{i=1}^{m}\boldsymbol{A}_i\rho_i(\boldsymbol{\lambda}),$$

where $\rho_i(\boldsymbol{\lambda})=\min\{1,\max\{0,-(\mu_k\rho_{ki}+\boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{A}_i)/\mu_k\}\}$ is the unique solution to the i:th minimization problem in (32). Together with Step 5 in the above algorithm which limits the size of the bundle, this dual method becomes efficient also for large-scale problems.

5. Numerical examples

Some numerical examples are given here to show the effect of accounting for uncertain boundary displacement and give an indication as to how the numerical solution methods discussed in Section 4 works. As mentioned in connection with problem (4), we will refer to objective function values as "compliance".

For purpose of comparison, to see that the robust formulation leads to designs which differ in some meaningful way from non-robust designs, we solve some problems with a fixed $\boldsymbol{\delta}_0$ such that $\|\boldsymbol{\delta}_0\|=1$. For such data problem (18) reduces to

$$\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\max_{\boldsymbol{\delta}:\|\boldsymbol{\delta}\|=1,\boldsymbol{\delta}=\boldsymbol{\delta}_0}\boldsymbol{\delta}^{\mathsf{T}}\boldsymbol{M}(\boldsymbol{\rho})\boldsymbol{\delta}=\min_{\boldsymbol{\rho}\in\mathscr{H}_h}\boldsymbol{f}^{\mathsf{T}}\boldsymbol{H}(\boldsymbol{\rho})\boldsymbol{f}, \tag{33}$$

where $\boldsymbol{f}=\boldsymbol{Q}\boldsymbol{\delta}_0$.

In the following we use SIMP [12] for penalization and a linear density filter [16] with radius R for regularization. Assuming an isotropic material, the global stiffness matrix is written as $\boldsymbol{K}(\boldsymbol{\rho})=\sum_{e=1}^{m}(E_{min}+\tilde{\rho}_e(\boldsymbol{\rho})^q(E-E_{min})\boldsymbol{K}_e)$, where $\tilde{\rho}_e(\boldsymbol{\rho})$ is a filtered variable, the SIMP parameter $q\geqslant 1$ and $\boldsymbol{K}_e$ is an element stiffness matrix. The constant E_{min} is introduced to avoid singular stiffness matrices and is set to 10^{-9} in the examples. The Young's modulus E is set to 10, the Poisson's ratio to 0.3 and the maximum volume fraction of solid material in (25) to $\gamma=0.4$. In all the examples we use as initial guess $\boldsymbol{\rho}_0$ a constant, feasible design. For the 2D (plane stress) examples, the displacement is approximated using bi-linear, square FEs.

The optimization problems are solved using MMA [55], IPOPT [64] or the bundle method described in Section 4.3. Unless explicitly stated, we use default settings for IPOPT and MMA. For IPOPT we used a limited-memory BFGS approximation of the Hessian (of the Lagrangian) with 60 correction pairs. The dual problem (32) in the bundle method is solved using the active-set algorithm of the Matlab (R2021a) function `fmincon` with default settings. The 2D state problems are solved using the sparse direct solver CHOLMOD [23].

5.1. Example 1

Figure 2 shows the set-up, based on an L-shaped design domain, for the first example. The thickness of the domain (into the paper) is $0.1L$, with L set to 1 throughout. A mesh consisting of 36834 FEs is used. Non-zero displacement, with $\boldsymbol{Q}=\boldsymbol{I}_{2x2}$, is prescribed on the upper edge of the tip over a small length of $0.01L$. The problems in this subsection are solved using MMA.

For reference we show in Fig. 3 some designs obtained for fixed $\boldsymbol{\delta}$:s, i.e. solutions to problem (33). In this case, to make sure that what we see is not an effect of ending up in a poor stationary point, we have set the SIMP parameter to $q=1$ to get a convex problem (we also do not use the density filter in this case). The blue arrows in the left plot indicates the direction used in the optimization and the red the worst (the one which maximizes the objective function) direction for the optimized design. The right column in Fig. 3 shows how the compliance varies if the direction of $\boldsymbol{\delta}$ is changed, with the minimum attained for the direction used in the optimization (the compliance is π-periodic in the angle, so strictly speaking it is only necessary to check it for angles in $[0,\pi)$).

Figure 4 shows an optimized design with uncertain boundary displacement whose direction is allowed to vary as indicated by the circle in Fig. 2. The design was obtained by applying MMA to the eigenvalue formulation (19). One notices that the design is quite similar to the design in the top of Fig. 3, the main differences showing near the loading and at the upper legs, but looking at how the compliance varies (Fig. 4, right), the robust design has a slightly smaller variability. Compared to the design in the bottom row of Fig. 3 the difference in variability is even more significant.

The stopping criteria for the optimizations illustrated in Fig. 3 and Fig. 4 was a maximum of 1000 MMA iterations. The iteration history in Fig. 5, which is essentially flat after 100 iterations, indi-

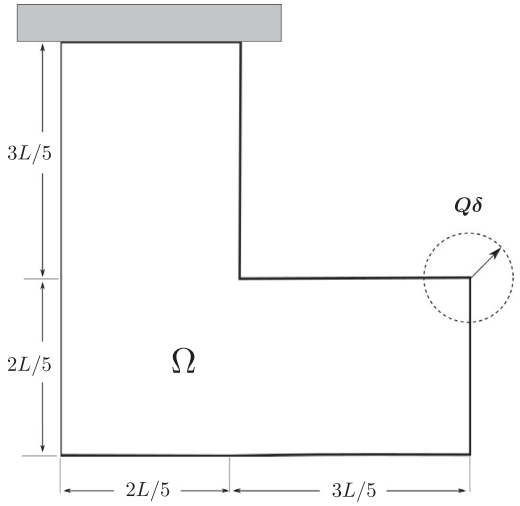

Fig. 2. L-shaped design domain. The upper part is fixed and non-zero displacement is prescribed on the upper edge at the tip.

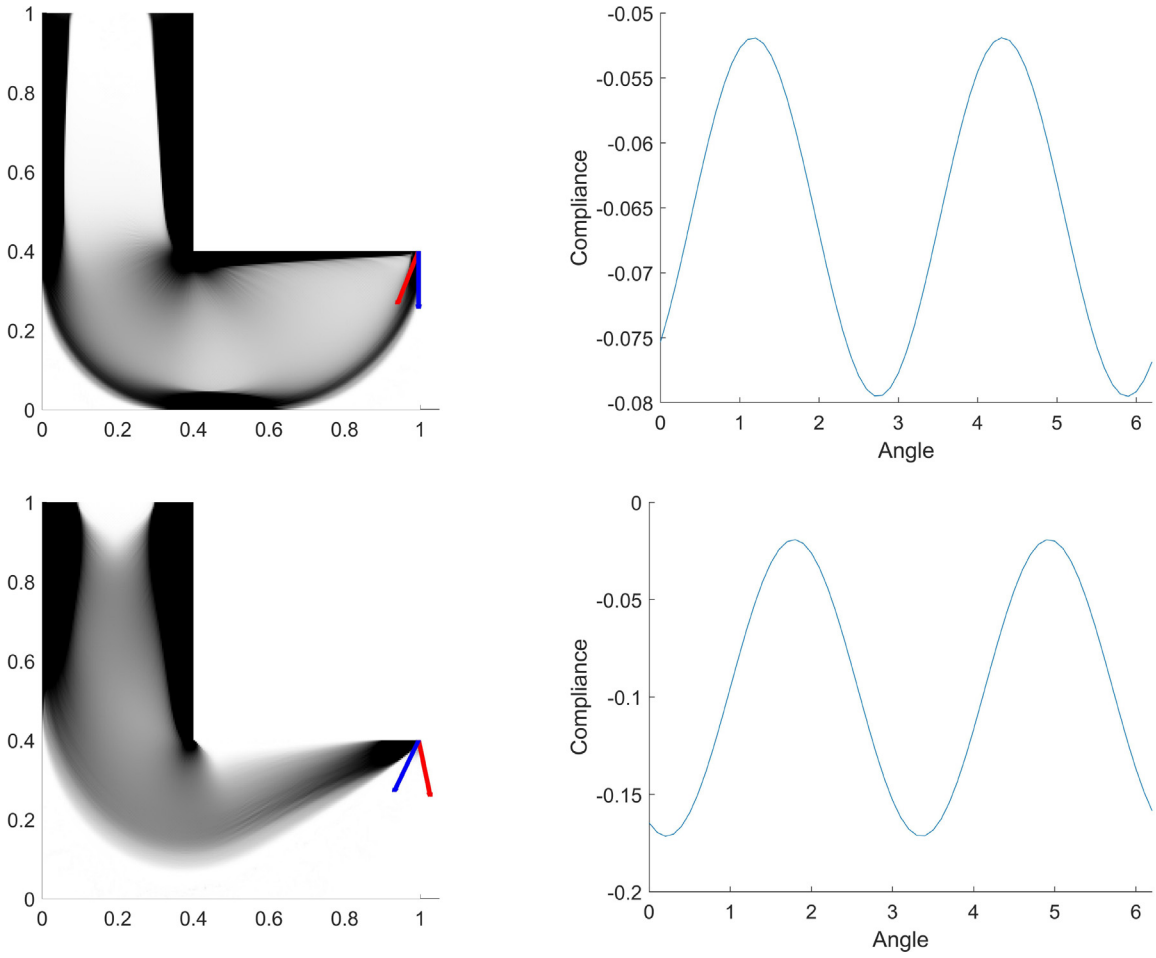

Fig. 3. Designs optimized for a fixed $\boldsymbol{\delta}$ (blue arrows). Top row: $\boldsymbol{\delta} = (0, -1)$. Bottom row: $\boldsymbol{\delta} = \left(2/\sqrt{5}\right)(-1/2, -1)$. Blue arrows indicate the fixed direction used in the optimization and red arrows shows the worst direction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

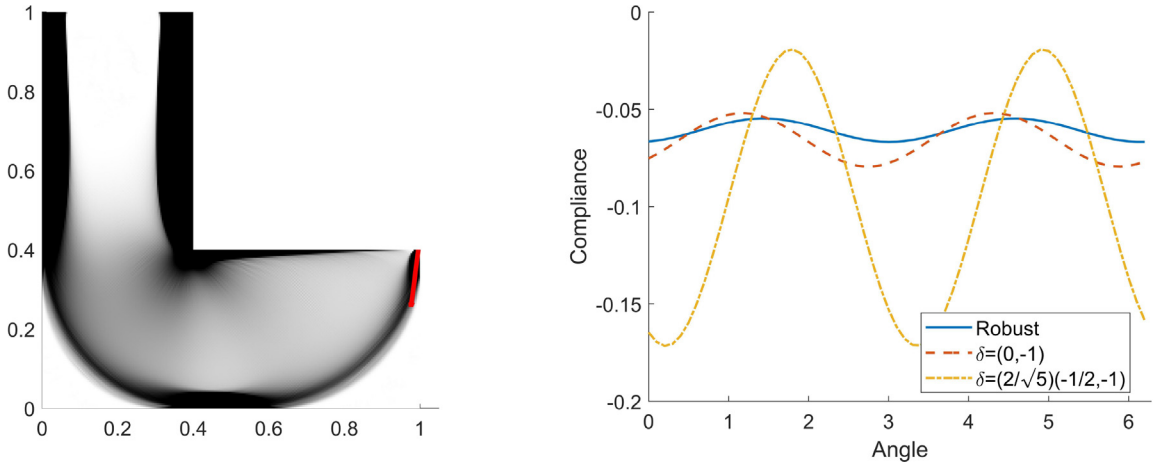

Fig. 4. Left: Design optimized under uncertainty. The load arrow shows the worst-case direction. Right: Compliance for the designs in Fig. 3 and Fig. 4 (Robust).

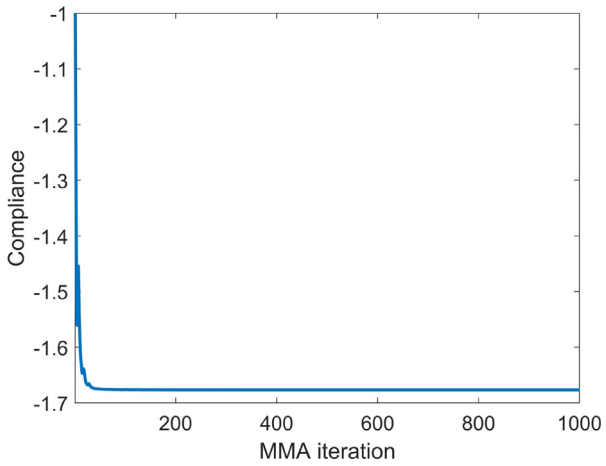

Fig. 5. Iteration history for the design in Fig. 4.

cate that in practise one could probably have used much fewer iterations in this particular case.

5.2. Example 2

We here consider the same setup as before, but now with SIMP parameter $q = 3$ to approximate TO, and the filter radius set to $R = 0.01L$ (increasing the filter radius leads, as expected, to designs with larger gray regions and fewer structural members which still, however, exhibits the robust behaviour of essentially constant compliance for all angles illustrated in the following figures).

5.2.1. Eigenvalue and SDP formulations

Figure 6 shows designs obtained using MMA (top) and IPOPT (middle) on the eigenvalue formulation (19), and IPOPT (bottom) on the SDP formulation (27). It is interesting to note that unlike for the case $q = 1$ (see Fig. 4) we get here designs, at least the middle and bottom ones, which have essentially the same compliance for every loading direction, meaning that the maximum eigenvalue has multiplicity 2. It is perhaps a bit surprising that this is seen here but not for $q = 1$ since theoretical analyses [48,57] indicate that the more degrees of freedom one has when minimizing a maximum eigenvalue, the more likely one is to see multiplicity greater than 1 at an optimal solution.

Looking at the iteration histories in Fig. 6 it should be noted that an MMA iteration requires evaluating the objective and constraints exactly once, whereas IPOPT may evaluate these functions several times during an iteration when performing a line search – this is a price one has to pay to have a globally convergent method. Critical to the performance of our optimization problem is the number of times the equilibrium Eqs. (22) are solved. For the top design in Fig. 6, using MMA on the eigenvalue formulation, the number of equilibrium solves is 2000; for the middle design, using IPOPT on the eigenvalue formulation, the number is 7022; and for the bot-

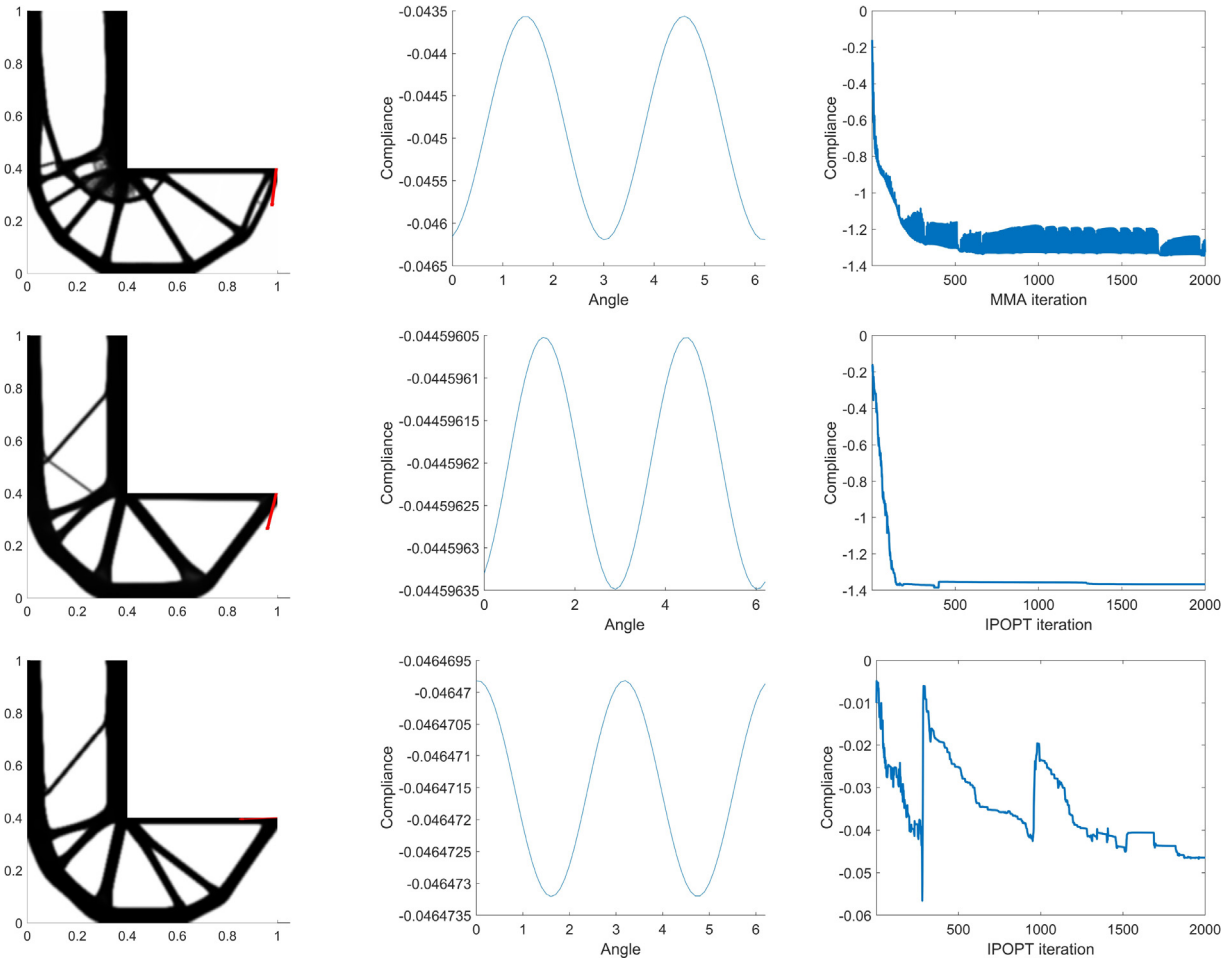

Fig. 6. Topology optimized design under uncertainty together with compliance variation and iteration history. Top: Eigenvalue formulation with MMA. Middle: Eigenvalue formulation with IPOPT. Bottom: SDP formulation with IPOPT (here "Compliance" in the iteration history is the value of the z-variable in (27)).

tom design, using IPOPT on the SDP formulation this number is 2325.

The highly oscillatory behaviour exhibited by MMA (Fig. 6, top right), which may be contrasted with the smooth convergence in Fig. 5, can be avoided by adjusting the parameters `asyinit`, `asydecr` and `asyincr` [55], or potentially switching to the globally convergent version of MMA [54]; either way one expects the number of equilibrium solves required to reach an optimum (or at least stationary point) to go up (we in fact tested GCMMA [55], with default settings, on this problem, but stopped after 18000 equilibrium solves with a design still having mostly intermediate $\boldsymbol{\rho}$-values). The relatively smooth convergence history for IPOPT on the eigenvalue formulation (Fig. 6, middle right), suggests that it might be possible to improve the performance by setting looser stopping criteria. This is not the case for IPOPT on the SDP formulation (Fig. 6, bottom right), where, as suggested by the oscillatory convergence history, setting a much looser stopping criteria frequently led to premature stopping at designs with large regions of intermediate $\boldsymbol{\rho}$-values.

5.2.2. Non-smooth formulation

Figure 7 shows results from using the bundle method of Section 4.3 to solve the same problem as in Fig. 6. As for the parameters in the algorithm we first set $\gamma = 0.1/m$ and $M = 0.25$ and take as stopping criteria a maximum of 1000 bundle iterations (similarly to MMA, one bundle iteration requires evaluating the objective and constraints exactly once). The three designs in Fig. 7 are obtained using the maximum number of elements in the bundle set to $\ell_{max} = 1$ (top), $\ell_{max} = 30$ (middle) and $\ell_{max} = 60$ (bottom). As can be seen, setting $\ell_{max} = 1$ (in which case we don't really have a "bundle") yields a design with large regions of intermediate $\boldsymbol{\rho}$-values, whereas $\ell_{max} = 30$ or $\ell_{max} = 60$ results in nice looking, close to binary-valued designs, with $\ell_{max} = 60$ giving a slightly better optimal value. The two designs differ a bit from those in Fig. 6 (middle and bottom), but are quite similar in terms of compliance.

In addition to μ_k and ℓ_{max}, our bundle method has two more parameters: M and γ. Intuitively, decreasing M makes it easier to satisfy the check whether to update the stability center in Step

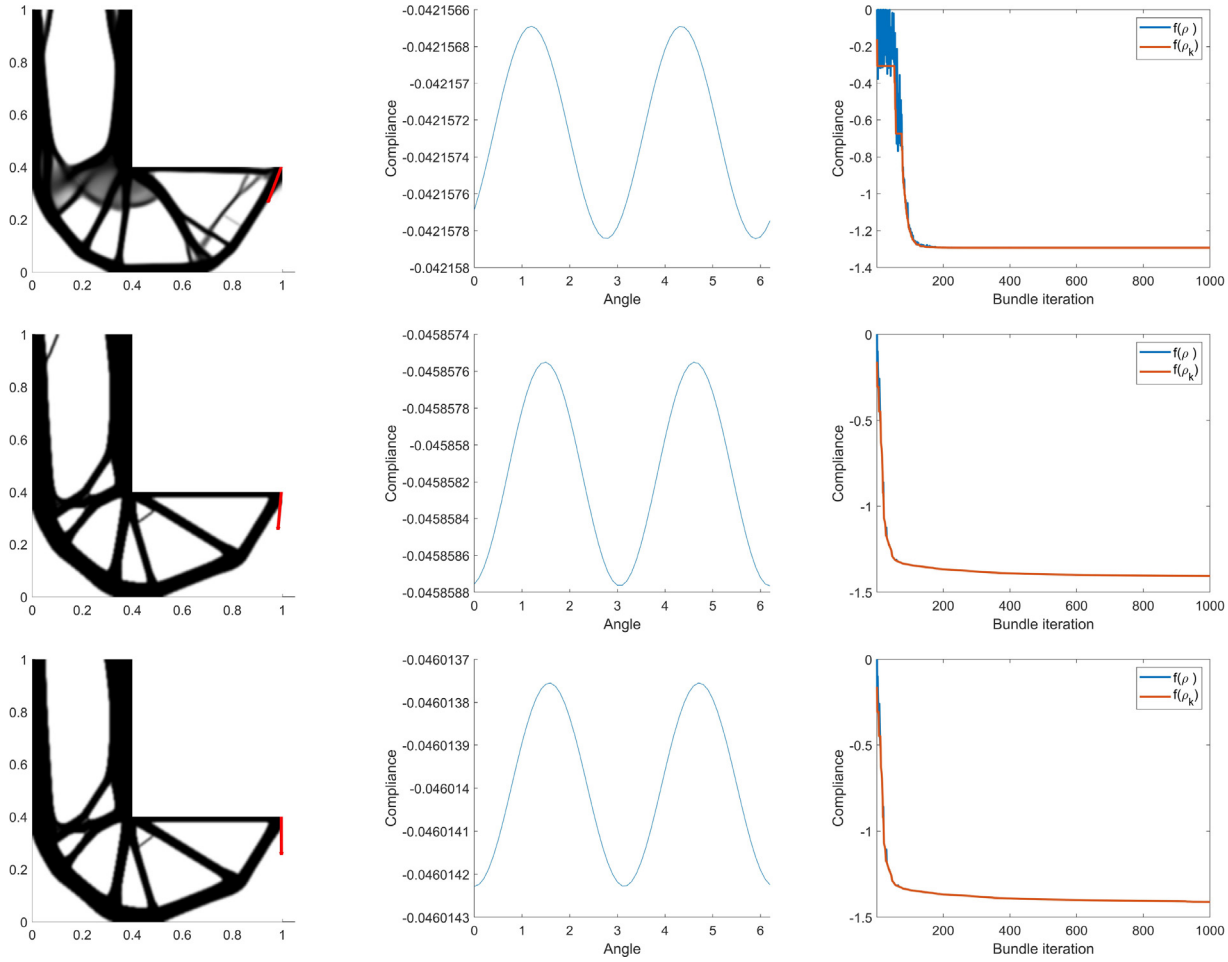

Fig. 7. Designs obtained using the bundle method. Left: Designs. Middle: Compliance variation for the optimized design. Right: Convergence history showing the compliance at the trial point $\hat{\boldsymbol{\rho}}$ (which may or may not be accepted as the next stability center) and the stability center $\boldsymbol{\rho}_k$.

4 in the algorithm, hence the method should become less "conservative". As for γ, entering in (30), setting it too large might lead to a poor model of the objective function. Our experience is however that the method is not highly sensitivity to (reasonably large) variations of these parameters, though the non-convexity of the problem implies that slight differences may be seen in optimized designs. Fig. 8 shows designs with $\ell_{max} = 30$ and varying M and γ. In all cases, the convergence-behaviour is similar to the one illustrated by the middle right plot in Fig. 7, i.e. smooth with rapid changes in the beginning. Some differences are seen between the designs but the optimal values are very similar: -0.045802 (left), -0.045384 (middle) and -0.045795 (right).

5.3. Example 3

To illustrate that the proposed TO problem formulation works also for larger 3D problems and how uncertainty might affect such designs we give in Fig. 9 two optimized designs from a 3D-version of the problem with the L-shaped domain using a mesh with 700000 eight-noded, tri-linear elements and a total of 2188053 degrees of freedom. The SIMP-parameter was set to $q = 3$, the filter radius to $R = 0.015$ and $\boldsymbol{Q} = \boldsymbol{I}_{3x3}$ over the middle part of the tip of the L. The multiplicity of the maximum eigenvalue in the robust design on the right is 2 (of 3).

Figure 10 shows (images taken from the software FreeCAD) stl-versions of the two designs. It is clear, that the robust and non-robust designs are quite different, with the robust design having a more box-like appearance to counter loading from all directions, whereas the non-robust design is clearly more adapted only to the downwards bending applied in the optimization.

The total runtime for the robust design was roughly 9.5 h for 800 MMA iterations on two 16-core Intel 6130 CPUs using the conjugate gradient method with an algebraic multi-grid preconditioner implemented in the code AMGCL [25]. Smoothed aggregation was used for coarsening and incomplete LU-factorization without fill-in for relaxation. Near null-space vectors of the stiffness matrix were provided for improved performance.

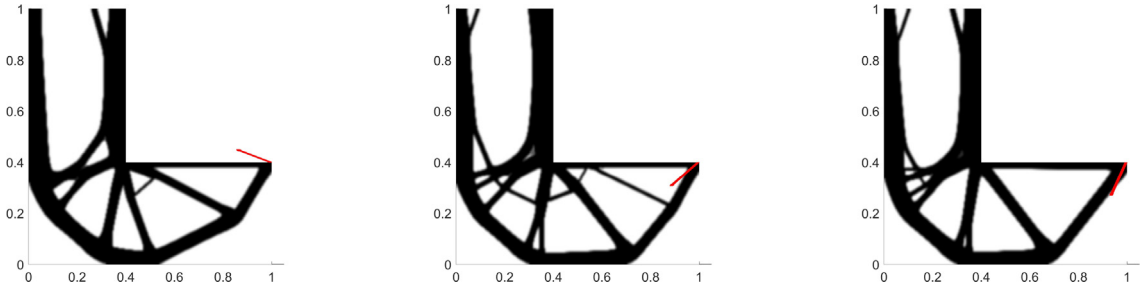

Fig. 8. Designs obtained using the same settings as in Fig. 7 (middle) but with varying M and γ. Left: $(M, \gamma) = (0.05, 0.1/m)$. Middle: $(M, \gamma) = (0.75, 0.1/m)$. Right: $(M, \gamma) = (0.25, 5/m)$.

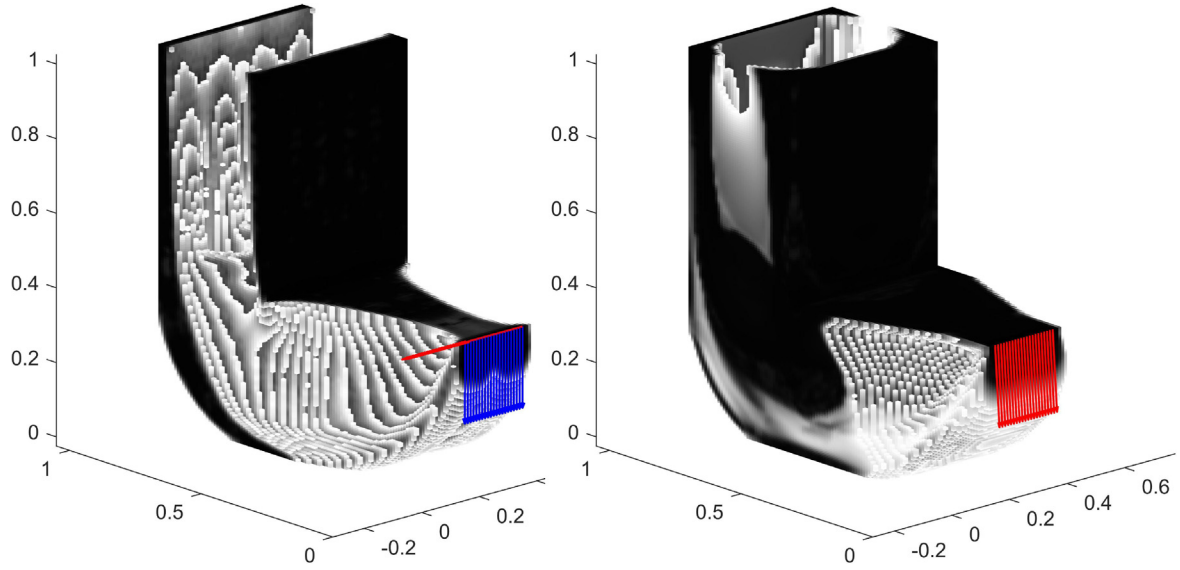

Fig. 9. 3D designs (tresholded at $\rho = 0.7$). Left: Design optimized for a fixed direction $(0, 0, -1)$ as blue arrows. Worst loading direction $(0, -1, 0)$ as red arrows. Right: Robust design. Worst loading direction $(-0.077, 0, -0.997)$ as red arrows. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

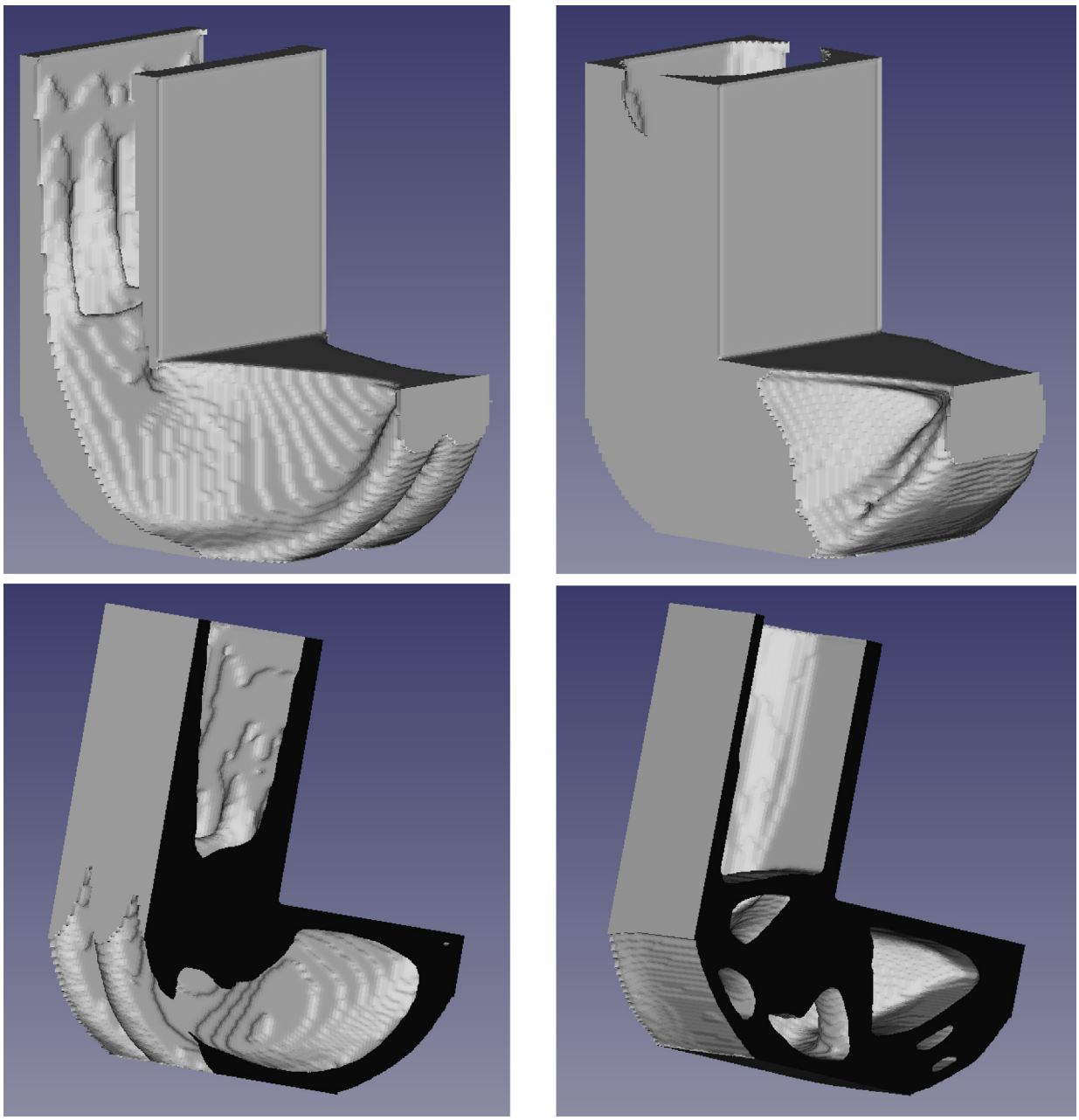

Fig. 10. Slightly smoothed stl-versions of the 3D designs from Fig. 9. Top and bottom left: Design optimized for a fixed direction. Top and bottom right: Robust design. A clipping plane at $y = 0.3$ has been inserted in the lower plots to show internal holes.

6. Concluding remarks

We have presented and showed well-posedness of a worst-case approach for TO under uncertain boundary displacement in a formulation leading to an optimization problem with a maximum eigenvalue as the objective function. For the numerical solution we tested three different approaches: (i) ignoring potential non-smoothness and applying a standard NLP-solver; (ii) reformulating the problem as a smooth, non-linear SDP problem; and (iii) applying a non-smooth bundle method. The first strategy actually worked in the sense of not leading to nonsensical designs (in particular IPOPT yielded a perfectly acceptable design in Fig. 6 (middle)), but just ignoring non-smoothness is not guaranteed to work in all cases. Between the SDP formulation and the bundle method, the latter (or other non-smooth methods) is perhaps more interesting for future work as it can potentially be applied to a broader class of problems not admitting SDP formulations. We note that our version of the bundle method is extremely simple, so robustness- and performance-wise we have only established a lower bound on what can be achieved with this type of algorithms.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research was funded by the Swedish Research Council, Grant No. 2019-0461. The 3D computations were enabled by the

Swedish National Infrastructure for Computing (SNIC) at the National Super Computer Centre partially funded by the Swedish Research Council through grant agreement No. 2018-05973. Thanks to Krister Svanberg for sharing his MMA code.

References

[1] Achtziger W, Bendsøe MP. Design for maximal flexibility as a simple computational model of damage. Struct Optim 1995;10(3):258–68.
[2] Achtziger W, Bendsøe MP. Optimal topology design of discrete structures resisting degradation effects. Struct Optim 1999;17(1):74–8.
[3] Achtziger W, Bendsøe MP, Taylor JE. An optimization problem for predicting the maximal effect of degradation of mechanical structures. SIAM J Optim 2000;10(4):982–98.
[4] Andréasson N, Evgrafov A, Patriksson M. An Introduction to Continuous Optimization. Studentlitteratur 2005.
[5] Ben-Haim Y. Convex models of uncertainty: Applications and implications. Erkenntnis 1994;41(2):139–56.
[6] Ben-Tal A, Nemirovski A. Robust truss topology design via semidefinite programming. SIAM J Optim 1997;7:991–1016.
[7] Ben-Tal A, Nemirovski A. Robust convex optimization. Math Oper Res 1998;23(4):769–805.
[8] Ben-Tal A, Nemirovski A. Structural design via semidefinite programming. In: Saigal R, Wolkowitcs H, Vandenberghe L, editors. Handbook on Semidefinite Programming. Kluwer; 2000.
[9] Ben-Tal A, El Ghaoui L, Nemirovski AS. Robust Optimization. Princeton Series in Applied Mathematics. Princeton University Press; 2009.
[10] Bendsøe MP, Kikuchi N. Generating optimal topologies in structural design using a homogenization method. Comput Methods Appl Mech Eng 1988;71(2):197–224.
[11] Bendsøe MP, Díaz AR. A method for treating damage related criteria in optimal topology design of continuum structures. Struct Optim 1998;16(2):108–15.
[12] Bendsøe MP, Sigmund O. Material interpolation schemes in topology optimization. Arch Appl Mech 1999;69:635–54.
[13] Bendsøe MP, Sigmund O. Topology Optimization: Theory, Methods and Applications. Springer-Verlag; 2003.
[14] Bogani C, Kocvara Michal, Stingl M. A new approach to the solution of the VTS problem with vibration and buckling constraints. In: 8th World Congress on Structural and Multidisciplinary Optimization; 2009.
[15] Bonnans JF, Shapiro A. Perturbation Analysis of Optimization Problems. Springer; 2000.
[16] Bourdin B. Filters in topology optimization. Int J Numer Meth Eng 2001;50(9):2143–58.
[17] Brezis H. Functional analysis, Sobolev spaces and partial differential equations. Springer Science & Business Media; 2010.
[18] Brittain K, Silva M, Tortorelli DA. Minmax topology optimization. Struct Multidiscipl Optim 2012;45(5):657–68.
[19] Canelas A, Carrasco M, López J. Application of the sequential parametric convex approximation method to the design of robust trusses. J Global Optim 2017;68:169–87.
[20] Changizi N, Jalalpour M. Robust topology optimization of frame structures under geometric or material properties uncertainties. Struct Multidiscipl Optim 2017;56(4):791–807.
[21] Clarke FH. Optimization and Nonsmooth Analysis. John Wiley & Sons; 1983.
[22] da Silva GA, Cardoso EL, Beck TA. Non-probabilistic robust continuum topology optimization with stress constraints. Struct Multidiscipl Optim 2018.
[23] Davis T. Direct Methods for Sparse Linear Systems. SIAM; 2006.
[24] Deaton J, Grandhi RV. A survey of structural and multidisciplinary continuum topology optimization: post 2000. Struct Multidiscipl Optim 2014;49(1):1–38.
[25] Demidov D. AMGCL: An efficient, flexible, and extensible algebraic multigrid implementation. Lobachevskii J Math 2019;40(5):535–46.
[26] Fiala J, Kočvara M, Stingl M. PENLAB: A MATLAB solver for nonlinear semidefinite optimization. pre-print; 2013.
[27] Greifenstein J, Stingl M. Topology optimization with worst-case handling of material uncertainties. Struct Multidiscipl Optim 2020;61:1377–97.
[28] Guo X, Bai W, Zhang W. Confidence extremal structural response analysis of truss structures under static load uncertainty via SDP relaxation. Comput Struct 2009;87(3–4):246–53.
[29] Guo X, Zhao X, Zhang W, Yan J, Sun G. Multi-scale robust design and optimization considering load uncertainties. Comput Methods Appl Mech Eng 2015;283:994–1009.
[30] Hägg L, Wadbro E. Nonlinear filters in topology optimization: existence of solutions and efficient implementation for minimum compliance problems. Struct Multidiscipl Optim 2017;55(3):1017–28.
[31] Hashimoto D, Kanno Y. A semidefinite programming approach to robust truss topology optimization under uncertainty in locations of nodes. Struct Multidiscipl Optim 2015;51:439–61.
[32] Hiriart-Urruty JB, Lemarechal C. Convex Analysis and Minimization Algorithms II. Springer; 1991.
[33] Holmberg E, Thore C-J, Klarbring A. Worst-case topology optimization of self-weight loaded structures using semi-definite programming. Struct Multidiscipl Optim 2015;52(5):915–28.
[34] Jalalpour M, Tootkaboni M. An efficient approach to reliability-based topology optimization for continua under material uncertainty. Struct Multidiscipl Optim 2016;53(4):759–72.
[35] Jansen M, Lombaert G, Diehl M, Lazarov BS, Sigmund O, Schevenels M. Robust topology optimization accounting for misplacement of material. Struct Multidiscipl Optim 2013;47(3):317–33.
[36] Jansen M, Lombaert G, Schevenels M, Sigmund O. Topology optimization of fail-safe structures using a simplified local damage model. Struct Multidiscipl Optim 2014;49(4):657–66.
[37] Kanno Y. An implicit formulation of mathematical program with complementarity constraints for application to robust structural optimization. J Oper Res Soc Jpn 2011;54(2):65.
[38] Karmitsa N, Bagirov A, Mäkelä MM. Comparing different nonsmooth minimization methods and software. Optimiz Methods Softw 2012;27(1):131–53.
[39] Kharmanda G, Olhoff N, Mohamed A, Lemaire M. Reliability-based topology optimization. Struct Multidiscipl Optimiz 2004;26(5):295–307.
[40] Kočvara M. Topology optimization with displacement constraints: a bilevel programming approach. Struct Optim 1997;14(4):256–63.
[41] Lazarov B, Wang F, Sigmund O. Length scale and manufacturability in density-based topology optimization. Arch Appl Mech 2016;86:1–30.
[42] Lazarov BS, Sigmund O. Filters in topology optimization based on helmholtz-type differential equations. Int J Numer Meth Eng 2011;86(6).
[43] Lazarov BS, Schevenels M, Sigmund O. Topology optimization considering material and geometric uncertainties using stochastic collocation methods. Struct Multidiscipl Optim 2012;46(4):597–612.
[44] Liu JT, Gea HC. Robust topology optimization under multiple independent unknown-but-bounded loads. Comput Methods Appl Mech Eng 2018;329:464–79.
[45] Martínez JM. A note on the theoretical convergence properties of the SIMP method. Struct Multidiscipl Optim 2005;29(4):319–23.
[46] Overton ML. Large-scale optimization of eigenvalues. SIAM J Optim 1992;2(1):88–120.
[47] Pantelides CP, Ganzerli S. Design of trusses under uncertain loads using convex models. J Struct Eng 1998;124(3):318–29.
[48] Pataki G. On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. Math Oper Res 1998;23(2):339–58.
[49] Schramm H, Zowe J. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. SIAM J Optim 1992;2:121–52.
[50] Sigmund O. Manufacturing tolerant topology optimization. Acta Mech Sin 2009;25(2):227–39.
[51] Sigmund O. On the usefulness of non-gradient approaches in topology optimization. Struct Multidiscipl Optim 2011;43.
[52] Stolpe M. Fail-safe truss topology optimization. Struct Multidiscipl Optim 2019. https://doi.org/10.1007/s00158-019-02295-7.
[53] Sturm JF. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. Optim Methods Softw 1999;11–12:625–53.
[54] Svanberg K. A class of globally convergent optimization methods based on conservative convex separable approximations. SIAM J Optim 2002;12(2):555–73.
[55] Svanberg K. MMA and GCMMA, versions September 2007; 2007. https://people.kth.se/krille/.
[56] Takezawa A, Nii S, Kitamura M, Kogiso N. Topology optimization for worst load conditions based on the eigenvalue analysis of an aggregated linear system. Comput Methods Appl Mech Eng 2011;200(25):2268–81.
[57] Thore C-J. Multiplicity of the maximum eigenvalue in structural optimization problems. Struct Multidiscipl Optim 2016;53:961–5.
[58] Thore C-J, Holmberg E, Klarbring A. A general framework for robust topology optimization under load-uncertainty including stress constraints. Comput Methods Appl Mech Eng 2017;319:1–18.
[59] Toriis AJ. Robust compliance-based topology optimization: A discussion on physical consistency. Comput Methods Appl Mech Eng 2019;352:110–36.
[60] Tütüncü RH, Toh KC, Todd MJ. Solving semidefinite-quadratic-linear programs using SDPT3. Math Program Ser B 2003;95:189–217.
[61] Vanderbei RJ, Benson HY. On formulating semidefinite programming problems as smooth convex nonlinear optimization problems. Technical report, Center for Discrete Mathematics &#38. Theoret Comput Sci 2000.
[62] Venini P. Topology optimization of dynamic systems under uncertain loads: An h∞-norm-based approach. ASME J Comput Nonlinear Dyn 2019;14.
[63] Venini P, Pingaro M. An innovative H_∞–norm based worst case scenario approach for dynamic compliance optimization with applications to viscoelastic beams. Struct Multidiscipl Optim 2016:1–26.
[64] Wächter A, Biegler LT. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. Math Program 2006;106:25–7.
[65] Zhang X, Kang Z, Zhang W. Robust topology optimization for dynamic compliance minimization under uncertain harmonic excitations with inhomogeneous eigenvalue analysis. Struct Multidiscipl Optim 2016:1–16.
[66] Zhou M, Fleury R. Fail-safe topology optimization. Struct Multidiscipl Optim 2016;54(5):1225–43.
[67] Zhou P, Du J, Lü Z. Topology optimization of freely vibrating continuum structures based on nonsmooth optimization. Struct Multidiscipl Optim 2017;56(3):603–18.