Structural and Multidisciplinary Optimization (2023) 66:204
https://doi.org/10.1007/s00158-023-03663-0

RESEARCH PAPER

# A Hilbertian projection method for constrained level set-based topology optimisation

Zachary J. Wegert[1] · Anthony P. Roberts[1] · Vivien J. Challis[1]

Received: 1 May 2023 / Revised: 20 July 2023 / Accepted: 23 August 2023 / Published online: 8 September 2023

## Abstract

We present an extension of the projection method proposed by Challis et al. (Int J Solids Struct 45(14–15):4130–4146, 2008) for constrained level set-based topology optimisation that harnesses the Hilbertian velocity extension-regularisation framework. Our Hilbertian projection method chooses a normal velocity for the level set function as a linear combination of (1) an orthogonal projection operator applied to the extended optimisation objective shape sensitivity and (2) a weighted sum of orthogonal basis functions for the extended constraint shape sensitivities. This combination aims for the best possible first-order improvement of the optimisation objective in addition to first-order improvement of the constraints. Our formulation utilising basis orthogonalisation naturally handles linearly dependent constraint shape sensitivities. Furthermore, use of the Hilbertian extension-regularisation framework ensures that the resulting normal velocity is extended away from the boundary and enriched with additional regularity. Our approach is generally applicable to any topology optimisation problem to be solved in the level set framework. We consider several benchmark constrained microstructure optimisation problems and demonstrate that our method is effective with little-to-no parameter tuning. We also find that our method performs well when compared to a Hilbertian sequential linear programming method.

**Keywords** Level set method · Topology optimisation · Constraints · Hilbertian projection method

## 1 Introduction

The field of topology optimisation has enjoyed rapid growth owing to improved computing power, new optimisation techniques, and application to a wide range of design problems (Bendsøe and Sigmund 2004; Deaton and Grandhi 2013; Sigmund and Maute 2013). Classical computational methods for topology optimisation include density-based methods (Bendsøe 1989; Rozvany et al. 1992) in which the design variables are material densities of elements/nodes in a mesh, and level set-based methods (Wang et al. 2003; Allaire et al. 2004) in which the boundary of the shape is implicitly tracked as the zero level set of a higher dimensional level set function. Conventional level set methods rely on the Hamilton–Jacobi evolution equation to update the design according to a normal velocity field defined on the boundary (e.g. Wang et al. 2003; Allaire et al. 2004). An important aspect of these methods is extending the normal velocity away from the boundary. To this end, the Hilbertian velocity extension-regularisation framework, which is well known in the context of level set methods (see discussion by Allaire et al. (2021b)), can be used to generate a velocity field that guarantees a descent direction and has additional regularity (smoothness) over the whole computational domain.

Topology optimisation problems often include multiple constraints. In density-based topology optimisation (Bendsøe 1989; Rozvany et al. 1992), the application of constraints is usually straightforward and handled by the optimisation algorithm (e.g. Method of Moving Asymptotes (Svanberg 1987)). In the context of conventional level set-based methods, applying constraints is more complicated. The augmented Lagrangian method (Nocedal and Wright 2006; Birgin and Martínez 2009) is a classical approach for constrained optimisation problems. It converts a constrained optimisation problem into a sequence of unconstrained problems that are a combination of the classical Lagrangian method and quadratic

Responsible Editor: Shikui Chen

✉ Zachary J. Wegert
zach.wegert@hdr.qut.edu.au

[1] School of Mathematical Sciences, Queensland University of Technology, 2 George St, Brisbane, QLD 4000, Australia

penalty method. In a level set framework, applying the method is straightforward: the shape sensitivity of the augmented Lagrangian is used to inform the normal velocity for the Hamilton–Jacobi equation (e.g. Guo et al. 2014; Allaire et al. 2016; Cao et al. 2021). However, the difficulty associated with tuning the accompanying parameters is problem dependent and scales with the number of constraints (Allaire et al. 2021b). The level set sequential linear programming method (SLP) (Dunning and Kim 2015; Dunning et al. 2015) involves linearising the optimisation problem into a number of sub-problems that are then solved using a linear programming method (e.g. the simplex method (Kambampati et al. 2020)). For level set-based topology optimisation, applying SLP is fairly straightforward except for implementing appropriate trust region constraints (Dunning and Kim 2015). The projection method (Wang and Wang 2004; Challis et al. 2008) projects the objective shape sensitivity onto a space that will leave the constraints unchanged and combines this with constraint shape sensitivities. This approach has been used to successfully design material microstructures subject to isotropy constraints (Challis et al. 2008) but has not been widely adopted in the literature. However, similar methods have more recently been proposed in the literature by Barbarosie et al. (2020) and Feppon et al. (2020) for level set topology optimisation. The projection method and these two recent works are examples of general null space gradient methods (e.g. Nocedal and Wright 2006).

It is natural to consider methods of constrained optimisation that take advantage of the Hilbertian extension-regularisation framework. For example, Allaire et al. (2021b) recently presented an SLP method in the Hilbertian framework. In this paper we revisit the projection method from Challis et al. (2008) and combine it with the Hilbertian extension-regularisation procedure. Our method constructs an orthogonal basis that spans the set of extended constraint shape sensitivities and an orthogonal projection operator that projects onto the set perpendicular to the extended constraint shape sensitivities. We then define the normal velocity for the level set function as a linear combination of the orthogonal projection operator applied to the extended objective function shape sensitivity and a weighted sum of basis functions for the extended constraint shape sensitivities. This normal velocity is naturally extended onto the bounding domain and endowed with additional regularity due to the Hilbertian extension-regularisation. Whilst our method is similar to other recently proposed approaches (Barbarosie et al. 2020; Feppon et al. 2020), our formulation utilising an orthogonal basis provides significant benefits.

To demonstrate our presented Hilbertian projection method we consider several linear elastic microstructure optimisation (i.e. inverse homogenisation) problems with multiple constraints. The constraints naturally arise under the enforcement of symmetries for the effective material properties, such as isotropy. Irrespective of optimisation method, microstructure optimisation has been used successfully for a range of design problems including linear elastic materials with extremal properties (e.g. Gibiansky and Sigmund 2000; Andreassen et al. 2014), multifunctional composites (e.g. Challis et al. 2008), auxetic materials (e.g. Vogiatzis et al. 2017), piezoelectric materials (e.g. Silva et al. 1998; Wegert et al. 2022), and multi-material composites (e.g. Zhuang et al. 2010; Faure et al. 2017). In this work, we consider maximising the bulk modulus with and without isotropy constraints and the design of auxetic and multi-phase materials. We compare our Hilbertian projection method with a Hilbertian sequential linear programming (SLP) method (Sect. 5.3.2, Allaire et al. 2021b) and show that the Hilbertian projection method is able to successfully handle these optimisation problems with little-to-no parameter tuning.

The remainder of the paper is as follows. In Sect. 2, we discuss the mathematical background for the level set method, Hilbertian extension-regularisation procedure, and linear elastic microstructure optimisation. In Sect. 3, we formulate the Hilbertian projection method and compare our formulation to the null space method presented by Feppon et al. (2020). In Sect. 4, we discuss our numerical implementation. In Sect. 5, we present and discuss the example optimisation problems and results. Finally, in Sect. 6, we present our concluding remarks.

## 2 Mathematical background

In this section we give a brief introduction to the level set method for topology optimisation and shape derivatives. We then discuss the Hilbertian extension-regularisation framework. We conclude by describing linear elastic microstructure optimisation for single- and multi-phase materials.

### 2.1 The level set method

Level set methods track the boundary of a domain $\Omega$ inside a bounding domain $D \subset \mathbb{R}^d$ implicitly via the zero level set of a function $\phi : D \to \mathbb{R}$ (Sethian 1996; Osher and Fedkiw 2006). For a domain $\Omega$ inside a bounding domain $D$, the level set function $\phi$ is typically defined as

$$\begin{cases} \phi(\boldsymbol{x}) < 0 \text{ if } \boldsymbol{x} \in \Omega, \\ \phi(\boldsymbol{x}) = 0 \text{ if } \boldsymbol{x} \in \partial\Omega, \\ \phi(\boldsymbol{x}) > 0 \text{ if } \boldsymbol{x} \in D\backslash\bar{\Omega}. \end{cases} \tag{1}$$

Using this definition and assuming that the interface may evolve in time, a material derivative of $\phi$ on $\partial\Omega$ gives

$$\frac{\partial\phi}{\partial t}(t,\boldsymbol{x}) + v(t,\boldsymbol{x})|\nabla\phi(t,\boldsymbol{x})| = 0, \tag{2}$$

where $v$ is the normal velocity of the interface. In practice, the above is solved over the whole bounding domain $D$ instead of only on the interface $\partial\Omega$ by extending the velocity $v$ away from the boundary. Assuming that the time interval $(0, T)$ is small so that the velocity does not vary in time gives the Hamilton–Jacobi evolution equation (Sethian 1996; Osher and Fedkiw 2006; Allaire et al. 2021b):

$$\begin{cases} \frac{\partial\phi}{\partial t}(t,\boldsymbol{x}) + v(\boldsymbol{x})|\nabla\phi(t,\boldsymbol{x})| = 0, \\ \phi(0,\boldsymbol{x}) = \phi_0(\boldsymbol{x}), \\ \boldsymbol{x} \in D,\ t \in (0,T), \end{cases} \tag{3}$$

where $\phi_0(\boldsymbol{x})$ is the initial condition for $\phi$ at $t = 0$.

It is often useful to reinitialise the level set function as the signed distance function $d_\Omega$. This ensures that the level set function is neither too steep nor too flat near the boundary of $\Omega$ (Osher and Fedkiw 2006). The signed distance function may be defined as follows (Allaire et al. 2021b):

$$d_\Omega(\boldsymbol{x}) = \begin{cases} -d(\boldsymbol{x},\partial\Omega) & \text{if } \boldsymbol{x} \in \Omega, \\ 0 & \text{if } \boldsymbol{x} \in \partial\Omega, \\ d(\boldsymbol{x},\partial\Omega) & \text{if } \boldsymbol{x} \in D\backslash\bar{\Omega}, \end{cases} \tag{4}$$

where $d(\boldsymbol{x},\partial\Omega) := \min_{\boldsymbol{p}\in\partial\Omega} |\boldsymbol{x}-\boldsymbol{p}|$ is the minimum Euclidean distance from $\boldsymbol{x}$ to the boundary $\partial\Omega$. Several methods are available for constructing the signed distance function and the reader is referred to Osher and Fedkiw (2006) and Allaire et al. (2021b) and the references therein for a detailed discussion. In this work we use the following *reinitialisation equation* (Peng et al. 1999; Osher and Fedkiw 2006) to reinitialise a pre-existing level set function $\phi_0(\boldsymbol{x})$ as the signed distance function:

$$\begin{cases} \frac{\partial\phi}{\partial t}(t,\boldsymbol{x}) + S(\phi_0(\boldsymbol{x}))(|\nabla\phi(t,\boldsymbol{x})| - 1) = 0, \\ \phi(0,\boldsymbol{x}) = \phi_0(\boldsymbol{x}), \\ \boldsymbol{x} \in D, t > 0. \end{cases} \tag{5}$$

Here, $S$ is the sign function and Eq. 5 is solved until close to steady state. Similar numerical schemes may be used to solve for Hamilton–Jacobi evolution and reinitialisation (Eqs. 3 and 5).

## 2.2 Shape derivatives

To find a normal velocity $v$ that reduces some functional $J(\Omega)$ via solution of the Hamilton–Jacobi equation (Eq. 3) we use the notion of shape derivatives. We recall the following useful results from Allaire et al. (2004, 2021b).

Suppose that we consider smooth variations of the domain $\Omega$ of the form $\Omega_{\boldsymbol{\theta}} = (\boldsymbol{I} + \boldsymbol{\theta})(\Omega)$, where $\boldsymbol{\theta} \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$. Then the following definition and lemma follow:

**Definition 1** (Allaire et al. 2004) The shape derivative of $J(\Omega)$ at $\Omega$ is defined as the Fréchet derivative in $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ at $\boldsymbol{\theta}$ of the application $\boldsymbol{\theta} \to J(\Omega_{\boldsymbol{\theta}})$, i.e.

$$J(\Omega_{\boldsymbol{\theta}})(\Omega)) = J(\Omega) + J'(\Omega)(\boldsymbol{\theta}) + \mathrm{o}(\boldsymbol{\theta}) \tag{6}$$

with $\lim_{\boldsymbol{\theta}\to 0} \frac{|\mathrm{o}(\boldsymbol{\theta})|}{\|\boldsymbol{\theta}\|} = 0$, where the shape derivative $J'(\Omega)$ is a continuous linear form on $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$.

**Lemma 1** (Allaire et al. 2004) *Let $\Omega$ be a smooth-bounded open set and $f \in W^{1,1}(\mathbb{R}^d)$. Define*

$$J(\Omega) = \int_\Omega f \,\mathrm{d}\Omega. \tag{7}$$

*Then $J$ is differentiable at $\Omega$ and*

$$J'(\Omega)(\boldsymbol{\theta}) = \int_\Omega \operatorname{div}(\boldsymbol{\theta} f) \,\mathrm{d}\Omega = \int_{\partial\Omega} f\, \boldsymbol{\theta}\cdot\boldsymbol{n} \,\mathrm{d}\Gamma, \tag{8}$$

*for any $\boldsymbol{\theta} \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$.*

Céa's formal method (Céa 1986) can be applied to find the shape derivative of a functional $J$ that depends on fields that satisfy specified state equations (e.g. Allaire et al. 2004, 2021b). The method relies on defining a Lagrangian functional $\mathcal{L}$ that satisfies the two following properties:

1. The state equations are generated by stationarity of $\mathcal{L}$ under variations of the fields.
2. $\mathcal{L}$ is equal to the functional of interest $J$ at the solution to the state equations.

Once these properties are satisfied the shape derivative of the functional of interest can be found using Lemma 1 (Allaire et al. 2004).

## 2.3 Hilbertian extension-regularisation

To infer a descent direction from $J'(\Omega)$ we utilise the Hilbertian extension-regularisation method as discussed by Allaire et al. (2021b). This involves solving an identification problem over a Hilbert space $H$ on $D$ with inner product $\langle\cdot,\cdot\rangle_H$: Find $g_\Omega \in H$ such that

$$\langle g_\Omega, w\rangle_H = -J'(\Omega)(w\boldsymbol{n}) \ \forall w \in H. \tag{9}$$

For an unconstrained optimisation problem the resulting field $g_\Omega$ is the extended shape sensitivity that is used to evolve the interface with $\boldsymbol{\theta} = \tau g_\Omega \boldsymbol{n}$, where $\tau > 0$ is sufficiently small.

The Hilbertian extension-regularisation method provides two important benefits: it naturally extends the shape sensitivity from $\partial\Omega$ onto the bounding domain $D$ and ensures a descent direction for $J(\Omega)$ with additional regularity (i.e. $H$ as opposed to $L^2(\partial\Omega)$) (Allaire et al. 2021b). As discussed by Allaire et al. (2021b), this may be viewed as an analogue to the sensitivity filtering used in density-based topology optimisation algorithms.

A common choice for the Hilbert space $H$ is $H^1(D)$ with the inner product

$$\langle u, v\rangle_H = \beta^2 \int_D \nabla u \cdot \nabla v \,\mathrm{d}\Omega + \int_D uv \,\mathrm{d}\Omega, \tag{10}$$

where $\beta$ is the so-called regularisation length scale (e.g. Allaire et al. 2016; Feppon et al. 2019; Allaire et al. 2021a). For microstructure optimisation, we use the periodic Sobolev space $H = H^1_{\text{per}}(D)$ and use the inner product defined in Eq. 10.

## 2.4 Linear elastic microstructure optimisation

In this section we briefly discuss computational homogenisation and topology optimisation in the context of periodic microstructure design.

The state equations for linear elastic homogenisation over a domain $\Omega$ contained in a representative volume element (RVE) $D \subset \mathbb{R}^d$ under an applied strain field $\bar{\varepsilon}_{ij}$ are (e.g. Yvonnet 2019)

$$-\sigma_{ij,i} = 0 \text{ in } \Omega, \tag{11}$$

$$\sigma_{ij} n_j = 0 \text{ on } \partial\Omega, \tag{12}$$

$$\sigma_{ij} = C_{ijkl}\varepsilon_{kl}, \tag{13}$$

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right), \tag{14}$$

$$\frac{1}{\mathrm{Vol}(D)} \int_\Omega \varepsilon_{kl} \,\mathrm{d}\Omega = \bar{\varepsilon}_{kl}, \tag{15}$$

where $\sigma_{ij}$ is the stress tensor, $\varepsilon_{ij} = \varepsilon_{ij}(\boldsymbol{u})$ is the $D$-periodic strain field with displacement $\boldsymbol{u}$, and $C_{ijkl}$ is the spatially dependent elasticity tensor. Note that in the above we use summation notation for indices and comma notation for derivatives.

To compute the homogenised stiffness tensor $\bar{C}_{ijkl}$ of a periodic material, the above state equations are solved over $\Omega$ for three ($d = 2$) or six ($d = 3$) different combinations of macroscopic strain fields. These macroscopic strain fields are applied by decomposing the strain into the constant macroscopic strain field and fluctuation strain field as $\varepsilon_{ij} = \bar{\varepsilon}_{ij} + \tilde{\varepsilon}_{ij}$. The macroscopic strain fields are then given by the unique components of $\bar{\varepsilon}^{(kl)}_{ij} = \frac{1}{2}\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)$ in $k$ and $l$. For example, in two dimensions, the unique macroscopic strains are $\bar{\varepsilon}^{(11)}_{ij}$, $\bar{\varepsilon}^{(22)}_{ij}$, and $\bar{\varepsilon}^{(12)}_{ij}$. The notation $\tilde{\varepsilon}^{(kl)}_{ij}$ is used to denote the strain field fluctuation arising from the applied strain field $\bar{\varepsilon}^{(kl)}_{ij}$.

In practice, Eqs. 11–15 are solved using a finite element method and the weak formulation given here:

**Weak form 1** *For each unique constant macroscopic strain field* $\bar{\varepsilon}^{(kl)}_{ij}$, *find* $\tilde{\boldsymbol{u}}^{(kl)} \in H^1_{\text{per}}(\Omega)^d$ *such that*

$$\begin{aligned}&\int_\Omega C_{pqrs}\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})\varepsilon_{pq}(\boldsymbol{v}) \,\mathrm{d}\Omega \\ &= -\int_\Omega C_{pqrs}\bar{\varepsilon}^{(kl)}_{rs}\varepsilon_{pq}(\boldsymbol{v}) \,\mathrm{d}\Omega \ \ \forall \boldsymbol{v} \in H^1_{\text{per}}(\Omega)^d,\end{aligned} \tag{16}$$

where $\varepsilon_{ij}(\boldsymbol{v}) = \frac{1}{2}\left(v_{i,j} + v_{j,i}\right)$.

### 2.4.1 Single-phase problems

For single-phase problems (one solid and a void phase), once the solution $\tilde{\boldsymbol{u}}^{(ij)}$ to Weak Form 1 has been found for each unique macroscopic strain $\bar{\varepsilon}^{(ij)}_{pq}$, the resulting homogenised stiffness tensor may be computed via (Yvonnet 2019)

$$\bar{C}_{ijkl}(\Omega) = \int_\Omega C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)}) + \bar{\varepsilon}^{(ij)}_{pq})\bar{\varepsilon}^{(kl)}_{rs} \,\mathrm{d}\Omega, \tag{17}$$

assuming that $\mathrm{Vol}(D) = 1$.

To evaluate Weak Form 1 and the homogenised stiffness tensor above, we utilise the ersatz material approximation. This method, which is classical in the literature (e.g. Allaire et al. 2004), fills the void phase with a soft material so that the state equations can be resolved without a body-fitted mesh. To this end, for small $\varepsilon_{\text{void}}$ we take

$$C_{ijkl}(\boldsymbol{x}) = \begin{cases} C_{ijkl} & \text{if } \boldsymbol{x} \in \Omega, \\ \varepsilon_{\text{void}} C_{ijkl} & \text{if } \boldsymbol{x} \in D \setminus \Omega, \end{cases} \tag{18}$$

and relax integration to be over $D$. We can provide a smooth approximation to Eq. 18 using a smoothed Heaviside function $H_\eta$

$$H_\eta(\phi) = \begin{cases} 0 & \text{if } \phi < -\eta, \\ \frac{1}{2} + \frac{\phi}{2\eta} + \frac{1}{2\pi}\sin\left(\frac{\pi\phi}{\eta}\right) & \text{if } |\phi| \leq \eta, \\ 1 & \text{if } \phi > \eta, \end{cases} \tag{19}$$

where $\eta$ is half the length of the small transition region of $H_\eta(\phi)$ between 0 and 1. Equation 18 can then be replaced with

$$C_{ijkl}(\phi) = C_{ijkl}(1 - H_\eta(\phi)) + \varepsilon_{\text{void}} C_{ijkl} H_\eta(\phi). \tag{20}$$

It is important to note that the ersatz material approximation is consistent (Allaire et al. 2021b). That is, as $\varepsilon_{\text{void}} \to 0$, the approximation becomes exact.

We conclude this section by stating the shape derivative of the homogenised stiffness tensor:

**Lemma 2** *The shape derivative of Eq. 17 is given by*

$$\begin{aligned} \bar{C}'_{ijkl}(\Omega)(\boldsymbol{\theta}) = \int_{\partial\Omega} & C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)}) + \bar{\varepsilon}_{pq}^{(ij)}) \\ & \times (\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)}) + \bar{\varepsilon}_{rs}^{(kl)})\, \boldsymbol{\theta} \cdot \boldsymbol{n} \,\mathrm{d}\Gamma. \end{aligned} \tag{21}$$

***Proof*** See Appendix 1.

### 2.4.2 Multi-phase problems

For multi-phase problems, we utilise the colour level set method in which up to $2^M$ phases can be represented by the sign of $M$ level set functions (Wang and Wang 2004; Allaire et al. 2014a). For example, in the case of four phases $\Omega_1, \Omega_2, \Omega_3$ and $\Omega_4$ with two level set functions $\phi_1$ and $\phi_2$, we have

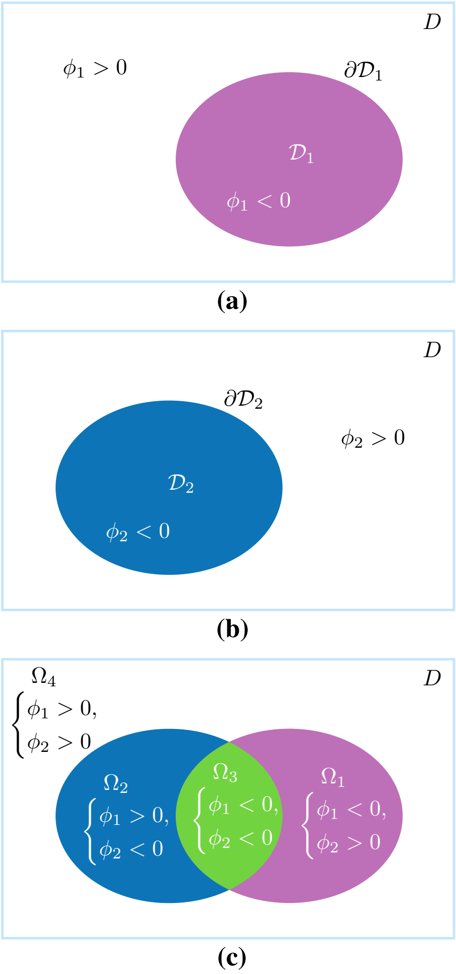
$$\begin{cases} \phi_1 < 0 \;\&\; \phi_2 > 0 \text{ if } \boldsymbol{x} \in \Omega_1, \\ \phi_1 > 0 \;\&\; \phi_2 < 0 \text{ if } \boldsymbol{x} \in \Omega_2, \\ \phi_1 < 0 \;\&\; \phi_2 < 0 \text{ if } \boldsymbol{x} \in \Omega_3, \\ \phi_1 > 0 \;\&\; \phi_2 > 0 \text{ if } \boldsymbol{x} \in \Omega_4. \end{cases} \tag{22}$$

We further denote the domains associated with each level set function $\phi_1$ and $\phi_2$ as $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively.

Figure 1 shows an illustration of this case.

In a similar way to Eq. 20 we can interpolate the value of the stiffness tensor between these domains via

$$\begin{aligned} & C_{ijkl}(d_{\mathcal{D}_1}, d_{\mathcal{D}_2}) \\ & = C_{ijkl,1}(1 - H_\eta(d_{\mathcal{D}_1}))H_\eta(d_{\mathcal{D}_2}) \\ & + C_{ijkl,2}H_\eta(d_{\mathcal{D}_1})(1 - H_\eta(d_{\mathcal{D}_2})) \\ & + C_{ijkl,3}(1 - H_\eta(d_{\mathcal{D}_1}))(1 - H_\eta(d_{\mathcal{D}_2})) \\ & + C_{ijkl,4}H_\eta(d_{\mathcal{D}_1})H_\eta(d_{\mathcal{D}_2}), \end{aligned} \tag{23}$$



**(a)**

**(b)**

**(c)**

**Fig. 1** An illustration of colour level sets with two level set functions and four phases. **a** and **b** show the domain $\mathcal{D}_1$ and $\mathcal{D}_2$ represented via the level set function $\phi_1$ and $\phi_2$, respectively. **c** shows the colour representation of $\Omega_1, \Omega_2, \Omega_3$, and $\Omega_4$ for different signs of $\phi_1$ and $\phi_2$

where $C_{ijkl,\alpha}$ is the elasticity tensor for the phase occupying $\Omega_\alpha$. In this multi-phase case we have replaced the level set functions $\phi_1$ and $\phi_2$ with $d_{\mathcal{D}_1}$ and $d_{\mathcal{D}_2}$ denoting their respective signed distance functions. This change facilitates shape differentiation. Unlike the situation discussed by Allaire et al. (2016), periodicity of $D$ ensures that this replacement is valid provided the level set functions are reinitialised often. Additional care should then be taken regarding the calculation of certain quantities and their shape derivatives. Namely, the homogenised stiffness tensor (Eq. 17) becomes

$$\begin{aligned} & \bar{C}_{ijkl}(\mathcal{D}_1, \mathcal{D}_2) \\ & = \int_D C_{pqrs}(d_{\mathcal{D}_1}, d_{\mathcal{D}_2})(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)}) + \bar{\varepsilon}_{pq}^{(ij)})\bar{\varepsilon}_{rs}^{(kl)} \,\mathrm{d}\Omega. \end{aligned} \tag{24}$$

The volume of $\Omega_1$ is given by

$$\mathrm{Vol}_{\Omega_1}(\mathcal{D}_1, \mathcal{D}_2) = \int_D (1 - H_\eta(d_{\mathcal{D}_1}))H_\eta(d_{\mathcal{D}_2}) \,\mathrm{d}\Omega. \tag{25}$$

Similar expressions are used for $\Omega_2$, $\Omega_3$, and $\Omega_4$.

Integration over the whole cell $D$ in Eqs. 24 and 25 with dependence on the signed distance functions differs from the single-phase case where integration is over the domain $\Omega$ occupied by the solid phase (see Eq. 17). Shape differentiability of the signed distance function and a coarea formula can be used in this multi-phase case to derive the shape derivative. We utilise the "approximate" formula discussed by Allaire et al. (2014a). This assumes that the Heaviside smoothing parameter $\eta$ is small and that the principal curvatures of $\partial\Omega$ vanish.

**Lemma 3** *The approximate shape derivatives of Eqs. 24 and 25 under variation of the domain $\mathcal{D}_1$ by $\boldsymbol{\theta}_1$ are*

$$\begin{aligned}&\bar{C}'_{ijkl}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\\ &\quad\approx -\int_{\partial\mathcal{D}_1}\frac{\partial C_{pqrs}}{\partial g}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\\ &\qquad\qquad\times(\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})+\bar{\varepsilon}^{(kl)}_{rs})\,\boldsymbol{\theta}_1\cdot\boldsymbol{n}\,\mathrm{d}\Gamma,\end{aligned}\tag{26}$$

*where $g = H_\eta(d_{\mathcal{D}_1})$ and*

$$\mathrm{Vol}'_{\Omega_1}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\approx\int_{\partial\mathcal{D}_1}H_\eta(d_{\mathcal{D}_2})\,\boldsymbol{\theta}_1\cdot\boldsymbol{n}\,\mathrm{d}\Gamma.\tag{27}$$

*Analogous expressions follow for $\boldsymbol{\theta}_2$ and $\Omega_2$, $\Omega_3$, and $\Omega_4$.*

***Proof*** See Appendix 2.

We note that comparisons between the "true" formula, "Jacobian-free" formula (zero principal curvatures), and "approximate" formula have been discussed for compliance elsewhere in the literature (Allaire et al. 2014a, 2016). It suffices to mention that Allaire et al. (2016) found that the "approximate" formula does not capture the distortion that arises due to the ray integration and approximation of the principal curvatures in the "true" formula.

## 3 Hilbertian projection method

The Hilbertian framework yields a descent direction $\boldsymbol{\theta}=\tau g_\Omega\boldsymbol{n}$ for unconstrained optimisation problems. However, for constrained optimisation problems such as

$$\begin{aligned}&\min_{\Omega\in\mathcal{U}_{\mathrm{ad}}}\ J(\Omega)\\ &\ \text{s.t.}\ \ C_i(\Omega)=0,\ i=1,\dots,N,\\ &\qquad a(\boldsymbol{u},\boldsymbol{v})=l(\boldsymbol{v}),\ \forall\boldsymbol{v}\in V,\end{aligned}\tag{28}$$

the choice of $\boldsymbol{\theta}$ is more difficult.

In the literature a variety of optimisation methods deal with this problem but few of these take advantage of the Hilbertian framework. Allaire et al. (2021b) recently presented a sequential linear programming (SLP) method in the Hilbertian framework. The projection method uses orthogonal projections to evolve the design in a direction that aims for best possible improvement of the objective functional whilst improving the constraint functionals (Challis et al. 2008). In the following, we present a Hilbertian extension of the projection method for constrained topology optimisation.

### 3.1 Preliminaries

We proceed by first solving the following set of scalar Hilbertian extension-regularisation problems over $H$ for an objective functional $J(\Omega)$ and constraint functionals $C_i(\Omega)$:

*Find $g_\Omega\in H$ and $\mu_{\Omega i}\in H$ such that*

$$\langle g_\Omega, v\rangle_H=-J'(\Omega)(v\boldsymbol{n}),\ \forall v\in H,\ \text{and}\tag{29}$$

$$\langle \mu_{\Omega i}, v\rangle_H=-C_i'(\Omega)(v\boldsymbol{n}),\ \forall v\in H\tag{30}$$

*for all $i=1,\dots,N$, with inner product $\langle\cdot,\cdot\rangle_H$ and norm $\|\cdot\|_H=\sqrt{\langle\cdot,\cdot\rangle_H}$.*

Next we use Gram–Schmidt orthogonalisation to remove linearly dependent constraints from the set $\{\mu_{\Omega i}\}_{i=1}^N$ to obtain the set $\{\mu_{\Omega p}\}_{p=1}^{\bar{N}}$, where $\bar{N}\leq N$. We use $\{\bar{\mu}_{\Omega p}\}$ to denote the corresponding orthogonal basis that spans the set $C\subset H$ of extended constraint shape sensitivities. The basis $\{\bar{\mu}_{\Omega p}\}$ can be used to construct an orthogonal projection operator $P_{C^\perp}$ that projects the shape sensitivity $g_\Omega$ onto the set $C^\perp$ perpendicular to the set of extended constraint shape sensitivities. We define this operator as

$$P_{C^\perp}g_\Omega=g_\Omega-\sum_{p=1}^{\bar{N}}\frac{\langle\bar{\mu}_{\Omega p},g_\Omega\rangle_H}{\|\bar{\mu}_{\Omega p}\|_H^2}\bar{\mu}_{\Omega p}.\tag{31}$$

Then, by construction, evolving the level set function using normal velocity $P_{C^\perp}g_\Omega$ would to first order improve the objective functional $J(\Omega)$ whilst leaving the constraint functionals $C_i(\Omega)$ unchanged. On the other hand, the set of basis functions $\{\bar{\mu}_{\Omega p}\}$ describes directions that to first order improve the constraint functionals $C_i(\Omega)$.

### 3.2 Formulation

The Hilbertian projection method can then be formulated as follows: For some rate parameter $\lambda\in\mathbb{R}$, suppose we choose $v_\Omega\in H$ in the deformation field $\boldsymbol{\theta}=\tau v_\Omega\boldsymbol{n}$ so that $J(\Omega)$ and $C_i(\Omega)$ decrease via (Challis et al. 2008)

$$\begin{cases} J'(\Omega)(v_\Omega \boldsymbol{n}) = \text{ min possible,} \\ C_i'(\Omega)(v_\Omega \boldsymbol{n}) = -\lambda C_i. \end{cases} \tag{32}$$

It is important to note that we purposefully pose the former requirement as "min possible" so that the objective functional may increase when required to improve constraints. Furthermore, linearly dependent constraints in the optimisation problem need to be consistent to ensure that the second line of Eq. 32 is well posed. Specifying constraints that have linearly dependent shape sensitivities but for which the directions of improvement are in contradiction would violate this requirement.

In the Hilbertian framework, Eq. 32 may be rewritten as

$$\begin{cases} \langle g_\Omega, v_\Omega \rangle_H = \text{ max possible,} \\ \langle \mu_{\Omega i}, v_\Omega \rangle_H = \lambda C_i. \end{cases} \tag{33}$$

Note that the change in sign comes from the application of Eqs. 29 and 30. We choose the following linear combination for $v_\Omega$:

$$v_\Omega = \sqrt{1 - \sum_{p=1}^{\bar{N}} \alpha_p^2} \frac{P_{C^\perp} g_\Omega}{\|P_{C^\perp} g_\Omega\|_H} + \sum_{p=1}^{\bar{N}} \alpha_p \frac{\bar{\mu}_{\Omega p}}{\|\bar{\mu}_{\Omega p}\|_H}, \tag{34}$$

where $\alpha_p \in \mathbb{R}$ are determined using $\langle \mu_{\Omega p}, v_\Omega \rangle_H = \lambda C_p$ for $p = 1, \dots, \bar{N}$. This generates a lower-triangular linear system of the form

$$\lambda C_p = \sum_{l=1}^{p-1} \alpha_l \frac{\langle \bar{\mu}_{\Omega l}, \mu_{\Omega p} \rangle_H}{\|\bar{\mu}_{\Omega l}\|_H} + \alpha_p \|\bar{\mu}_{\Omega p}\|_H, \tag{35}$$

which can easily be solved via forward substitution (Challis et al. 2008).

To first order, the first term of Eq. 34 improves the objective whilst leaving the constraints unchanged due to orthogonality, whilst the second term improves the constraints with extended shape sensitivities that contribute to the basis $\{\bar{\mu}_{\Omega p}\}$. In the numerical examples, we have observed satisfaction of all constraints at convergence of the optimisation algorithm, including those that have linearly dependent shape sensitivities. The square root term of Eq. 34 is included to facilitate a balance between improving the objective and constraints.

### 3.3 Parameters

As discussed by Challis et al. (2008), the rate parameter $\lambda$ should be chosen to ensure $1 - \sum_{p=1}^{\bar{N}} \alpha_p^2 \geq 0$ and $\sum_{p=1}^{\bar{N}} \alpha_p^2 \geq \alpha_{\text{min}}^2$. The new parameter $\alpha_{\text{min}}$ then controls the balance between improving the objective or constraints. For example, $\alpha_{\text{min}} = 1$ ignores the objective function in Eq. 28 and instead solves a constraint satisfaction problem. As a result, the method only has a single parameter $\alpha_{\text{min}}$, whilst $\lambda$ is dictated by the inequalities above. In general, we find that $\alpha_{\text{min}}$ does not require fine tuning and unless otherwise stated we choose $\alpha_{\text{min}}^2 = 0.1$.

### 3.4 Comparison to null space methods

Our formulation is similar to null space methods recently developed by Barbarosie et al. (2020) and Feppon et al. (2020), both of which present a similar formulation. In the following we discuss some differences between our Hilbertian projection method and the null space method proposed by Feppon et al. (2020).

Most notably, our formulation makes use of an orthogonal basis for the set of extended constraint shape sensitivities. This avoids a possibly expensive matrix inversion that appears in the algorithm presented by (Feppon et al. 2020). Our use of the orthogonal basis also avoids reliance on the linear independence constraint qualification (LICQ) condition (Sect. 2.1, Feppon et al. 2020). Our method therefore naturally handles linearly dependent constraint shape sensitivities. Such dependencies often appear in microstructure optimisation problems when symmetries are imposed on the effective material properties (e.g. Sects. 5.2 and 5.4 below). Multi-phase level set-based topology optimisation via the colour level set method (Wang and Wang 2004; Allaire et al. 2014a) can also give rise to such linear dependency (e.g. Sect. 5.4 below). The ability of the Hilbertian projection method to naturally handle these situations gives the user more freedom in how topology optimisation problems are posed and avoids additional special treatment of linearly dependent constraint sensitivities.

For equality constrained problems when LICQ is satisfied, both the null space method (Feppon et al. 2020) and our Hilbertian project method give equivalent directions for improvement of the objective and violated constraints, up to coefficients $\alpha_p$ and constant $\lambda$. However, the method of attaining this improvement is quite different. In this work, the second term of Eq. 34 is a linear combination of the orthogonal basis of the set of extended constraint shape sensitivities. The coefficients $\alpha_p$ are chosen to improve constraints exponentially, as per the second requirement in Eq. 33. The null space instead uses a Gauss–Newton direction for ensuring exponential decay of violated constraints (Lemma 2.5 and Proposition 2.6, Feppon et al. 2020). This again relies on LICQ and possibly expensive matrix inversion discussed above.

Finally, unlike the null space methods proposed by Barbarosie et al. (2020) and Feppon et al. (2020), the Hilbertian projection method as formulated above is unable to handle inequality constraints. Such an extension could be considered in the future using slack variables or by adopting the procedure used by either Barbarosie et al. (2020) or Feppon et al. (2020). Interestingly, in terms of the total length covered to reach the optimum, Feppon et al. (2020) found that their dual quadratic programming method for handling inequality constraints yielded equivalent performance when compared to the method of slack variables. However, using slack variables introduces additional computational cost and possibly further parameter tuning (Feppon et al. 2020). For our implementation this additional cost should be small owing to the use of orthogonalisation. As such the slack variable method would be an appropriate first recourse for implementing inequality constraints within the Hilbertian projection method. This would be similar to the approach taken by Schropp and Singer (2000).

## 4 Numerical implementation

In the following we describe the numerical implementation of our topology optimisation algorithm. We first discuss the resolution of state equations and Hamilton–Jacobi-type equations followed by an overview of the optimisation algorithm. We finish with a brief discussion of the Hilbertian SLP method which is compared to our presented Hilbertian projection method.

### 4.1 Resolving state and Hamilton–Jacobi-type equations

To solve the state equations and the Hilbertian extension-regularisation problems we use the finite element package *Gridap* (Badia and Verdugo 2020; Verdugo and Badia 2022) in the programming language *Julia*. In particular, we discretise the periodic domain $D \subset \mathbb{R}^d$ into $n^d$ linear quadrilateral ($d = 2$) or hexahedral ($d = 3$) elements with element width $\Delta x$ and discretise the level set function at the nodes of the triangulation. To reduce computational cost when solving the state equations, we remove any elements that are completely void phase and leave a strip of ersatz material near the phase interface. The resulting linear systems for the state equations and Hilbertian extension-regularisation problems are then solved using a direct method in 2D or a GPU-based Jacobi pre-conditioned conjugate gradient method in 3D.

For the Hamilton–Jacobi evolution equation and signed distance reinitialisation equation (Eqs. 3 and 5), we use standard first-order Godunov upwind finite difference schemes (Peng et al. 1999; Allaire et al. 2004; Osher and Fedkiw 2006) that have been implemented on GPUs using *CUDA.jl* (Besard et al. 2018). For the Hamilton–Jacobi evolution equation we use $\lfloor n/10 \rfloor$ or $\lfloor n/3 \rfloor$ number of time steps in two dimensions or three dimensions, respectively. We are more conservative in three dimensions because we have less elements along each axial direction. For the reinitialisation equation we iterate until reaching a stopping condition

$$\|\phi^q - \phi^{q-1}\|_\infty < 5 \times 10^{-5}, \tag{36}$$

where $q$ is the iteration number. In addition, for the sign function $S$ we use the common approximation

$$S(\phi) = \frac{\phi}{\sqrt{\phi^2 + |\nabla\phi|^2 \Delta x^2}} \tag{37}$$

that applies on a Cartesian grid with square elements of side length $\Delta x$ (Osher and Fedkiw 2006).

### 4.2 Algorithm overview

In Algorithm 1 we present our optimisation algorithm that is based on the theory discussed in Sects. 2 and 3. Algorithm 1 is similar to Algorithm 5 presented by Allaire et al. (2021b), with the addition of a line search method for determining the Courant–Friedrichs–Lewy (CFL) coefficient $\gamma$ for solving the Hamilton–Jacobi evolution equation (e.g. Allaire et al. 2021a) with time step (Osher and Fedkiw 2006)

$$\Delta t = \frac{\gamma \Delta x}{\|v_\Omega\|_\infty}. \tag{38}$$

Note that we omit the indices on $\gamma$ that appear in Algorithm 1 for sake of clarity. In general, the line search method helps to remove oscillations in the optimisation history and improve convergence. For the stopping criterion we require that the current objective value compared to the previous five is stationary and that the constraints are satisfied within specified tolerances.

The Hilbertian projection method is implemented using the package *DoubleFloats.jl* that gives a machine epsilon of roughly $5 \times 10^{-32}$. This prevents accumulation of round-off error when generating the projection operator that can affect the optimisation history. All other computations are completed in standard double precision.

**Algorithm 1** Optimisation Algorithm

```
Initialisation: Initialise $\Omega^0$ inside a computational domain $D$ with mesh $\mathcal{T}$ and a level set function $\phi^0$.
 1: Find the initial solution $\boldsymbol{u}^{(ij)}$ to the homogenisation problem for each unique $\bar{\varepsilon}^{(ij)}$.
 2: for $q = 1, \dots, q_{\max}$ do
 3:     Calculate the shape sensitivity of the objective $J(\Omega^{q-1})$ and constraints $C_i(\Omega^{q-1})$.
 4:     Solve scalar Hilbertian extension-regularisation problems for the objective and constraints with length scale $\beta$.
 5:     Apply the Hilbertian projection method to find $v_\Omega$.
 6:     for $k = 1, \dots, k_{\max}$ do
 7:         Solve H-J evolution equation with CFL coefficient $\gamma^{q-1,k}$ to find new domain $\bar{\Omega}^k$ with associated level set function $\bar{\phi}^k$.
 8:         Solve reinitialisation equation.
 9:         Solve state equations for linear elastic homogenisation and calculate the new objective $J_{\text{new}}$.
10:         if $J_{\text{new}} < J(\Omega^{q-1}) + \xi|J(\Omega^{q-1})|$ or $\gamma^{q-1,k} < \gamma_{\min}$ then
11:             Increase the CFL coefficient:
                    $$\gamma^{q,k} = \min(\delta_{\text{inc}}\gamma^{q-1,k}, \gamma_{\max}).$$
12:             Accept the new iteration with $\Omega^q = \bar{\Omega}^k$ and $\phi^q = \bar{\phi}^k$.
13:             Break inner loop.
14:         else
15:             Decrease the CFL coefficient:
                    $$\gamma^{q-1,k+1} = \max(\delta_{\text{dec}}\gamma^{q-1,k}, \gamma_{\min}).$$
16:             Reject the new iteration and continue loop.
17:         end if
18:     end for
19:     if $|J(\Omega^q) - J(\Omega^{q-j})| \leq \epsilon_1|J(\Omega^q)|, \forall j = 1, \dots, j_{\max}$ and $|C_i(\Omega^q)| < \epsilon_2\ \forall i$ then
20:         return $\Omega^q$ and $\phi^q$.
21:     end if
22: end for
```

Table 1 gives the parameter values used for all the optimisation examples unless otherwise stated in Sect. 5.

### 4.3 Comparison to sequential linear programming (SLP)

We compare the Hilbertian projection method to the Hilbertian SLP method presented by Allaire et al. (2021b) (Sect. 5.3.2). We make two adjustments to the method. Firstly, to match our formulation we replace the inequality constraints with equality constraints. In addition, we change the trust region constraints for the constraint functionals to be

$$|\lambda_i| \leq \frac{\Delta x}{2\|\boldsymbol{\theta}_i\|_H} \tag{39}$$

for $i = 1, \dots, N$. For several two-dimensional problems we find that this choice promotes convergence and better optimisation results. However, as we will discuss later, choosing trust region constraints is not straightforward for our example optimisation problems.

**Table 1** Parameter values used for optimisation examples

| Parameter | Value | Type |
|---|---|---|
| $q_{\max}$ | 1000 | Max iterations |
| $k_{\max}$ | 10 | Max line search iterations |
| $\Delta x$ | $1/n$ | Mesh spacing |
| $\varepsilon_{\text{void}}$ | 0.001 | Ersatz material coeff |
| $\eta$ | $1.5\Delta x$ | Heaviside smoothing |
| $\beta$ | $4\Delta x$ | Hilbertian ext.-reg |
| $\alpha^2_{\min}$ | 0.1 | Hilbertian proj. meth |
| $\lambda$ | 0.5 | Hilbertian proj. meth |
| $\gamma_{\min}$ | 0.001 | H-J equation |
| $\gamma_{\max}$ | 0.1 | H-J equation |
| $\gamma_{\text{reinit}}$ | 0.1 | Reinit. equation |
| $\xi$ | 0.005 | Line search |
| $\delta_{\text{inc}}, \delta_{\text{dec}}$ | 1.1, 0.7 | Line search |
| $\epsilon_1, \epsilon_2$ | 0.01, 0.0001 | Stopping criteria |
| $j_{\max}$ | 5 | Stopping criteria |

Any variation in the parameters from these values will be specifically stated in the relevant example problem section

We implement Hilbertian SLP by adjusting line 5 of Algorithm 1 accordingly. To solve the resulting linearised optimisation problem we use the Julia packages *JuMP.jl* (Dunning et al. 2017) and *Ipopt.jl* (Wächter and Biegler 2006).

## 5 Example problems

In the following we give the optimisation results for several example problems that have been solved with both the Hilbertian projection method and Hilbertian SLP method.

### 5.1 Example 1: maximum bulk modulus

In this example we consider a bounding domain $D = [0, 1]^d$ that contains a solid phase and void phase. The solid phase is constructed from an isotropic medium with $E = 1$ and $\nu = 0.3$. Subject to a volume constraint $\text{Vol}(\Omega) = 1/2$, we maximise the effective bulk modulus $\bar{\kappa}(\Omega)$ of the material. The bulk modulus is a measure of stiffness to volumetric strain given by

$$\bar{\kappa} = \frac{1}{4}(\bar{C}_{1111} + \bar{C}_{2222} + 2\bar{C}_{1122}) \tag{40}$$

in two dimensions or

$$\bar{\kappa} = \frac{1}{9}(\bar{C}_{1111} + \bar{C}_{2222} + \bar{C}_{3333} \\ + 2(\bar{C}_{1122} + \bar{C}_{1133} + \bar{C}_{2233})) \tag{41}$$

in three dimensions. In other words, we seek to solve the optimisation problem:

$$\begin{aligned} \min_{\Omega\in\mathcal{U}_{\mathrm{ad}}} \quad & -\bar{\kappa}(\Omega) \\ \text{s.t.} \quad & \mathrm{Vol}(\Omega) = 1/2, \\ & a(\boldsymbol{u},\boldsymbol{v}) = l(\boldsymbol{v}),\ \forall \boldsymbol{v} \in V. \end{aligned} \tag{42}$$

The last line represents the satisfaction of the state equations.

In two dimensions we use a periodic starting structure with four equally spaced holes. For three dimensions the initial boundary between void and solid material is given by a Schwarz P minimal surface. It is well known that in two dimensions hole nucleation is not possible under Hamilton–Jacobi evolution (e.g. Allaire et al. 2004). For this reason we initialise the two-dimensional optimisation problems with more holes than required. Topological derivatives could be incorporated to rectify this, but this is outside of the scope of the current paper. We also note that different starting structures could be used provided they are periodic and have non-zero stiffness.

Figures 2 and 3 show the starting structures and optimisation results for two and three dimensions, respectively, for both the Hilbertian projection method and SLP. In addition, we compare the objective value with the Hashin–Shtrikman (HS) upper bound (Hashin and Shtrikman 1963). Table 2 shows a summary of the results.

In two dimensions both the Hilbertian projection method (Fig. 2b, d) and Hilbertian SLP (Fig. 2c, e) perform well, converging in 32 and 12 iterations, respectively, at 99.71% and 99.51% of the HS bound. In addition, the resulting structures are geometrically similar and match classical results in the literature (Sect. 2.10.3, Bendsøe and Sigmund 2004). It is worth noting that no topological changes occur in this example.

In three dimensions the Hilbertian projection method converges to 99.86% of the HS bound, whilst Hilbertian SLP converges to 71.07% of the bound. Over the course of the iteration history, topological changes occur for the Hilbertian projection method, whilst SLP fails to evolve from the initial topology. The resulting structure from the Hilbertian projection method matches other classical results (Sect. 2.10.3, Bendsøe and Sigmund 2004).

## 5.2 Example 2: maximum bulk modulus with isotropy

In Example 2, we consider the same problem setup as Example 1 with the addition of macroscopic isotropy constraints. This ensures that the resulting homogenised stiffness tensor is invariant under rotation.

In two dimensions the optimisation problem is given by

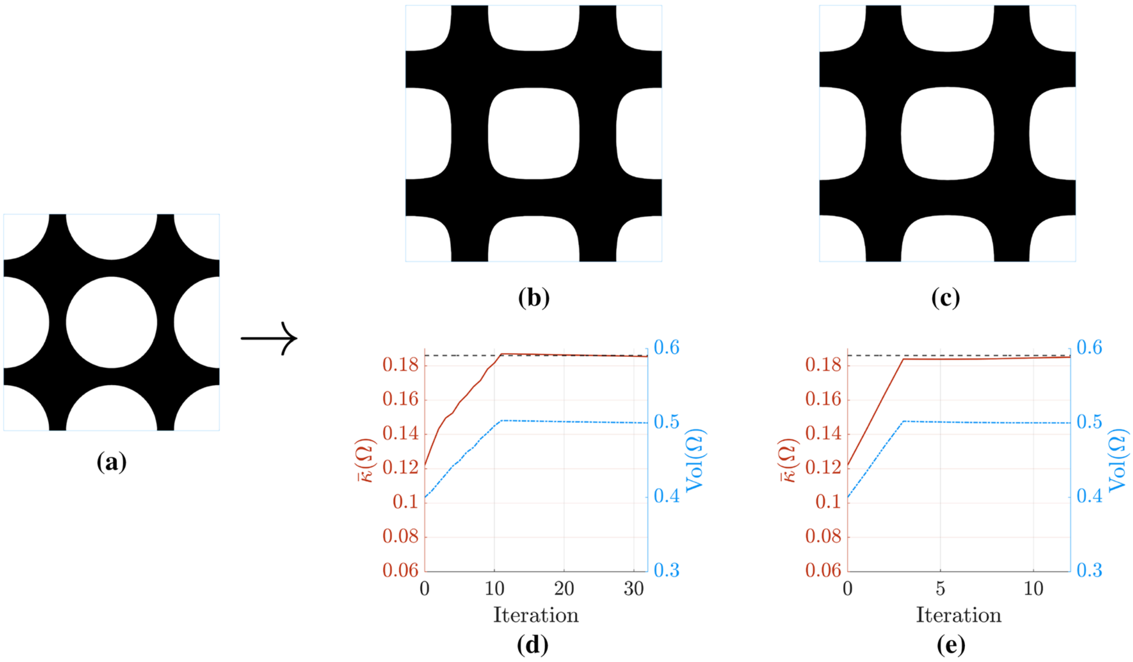


**Fig. 2** Two-dimensional optimisation results for Example 1: maximum bulk modulus. For the starting structure **a**, **b** and **c** show the final structures for the Hilbertian projection method and SLP, respectively, whilst **d** and **e** show the respective iteration histories. The Hashin–Shtrikman upper bound for the bulk modulus is given by the dashed black line

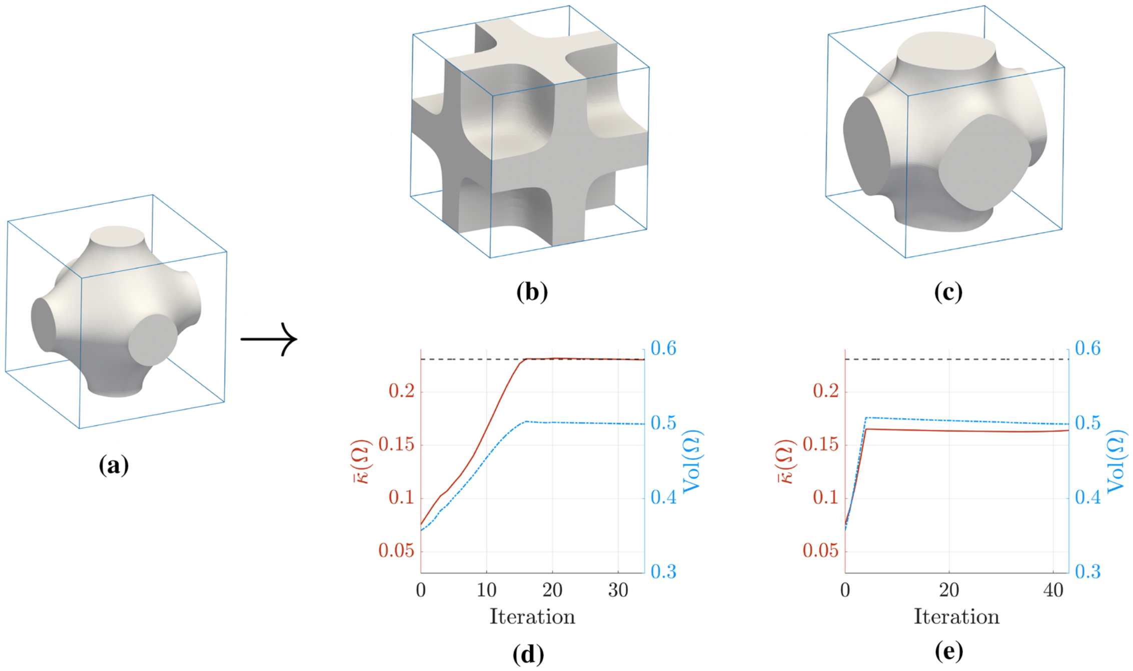


**Fig. 3** Three-dimensional optimisation results for Example 1: maximum bulk modulus. For the starting structure **a**, **b** and **c** show the final structures for the Hilbertian projection method and SLP, respectively, whilst **d** and **e** show the respective iteration histories. The Hashin–Shtrikman upper bound for the bulk modulus is given by the dashed black line

**Table 2** Summary of optimisation results for Example 1: maximum bulk modulus

| Method | $d$ | Fig | $\bar{\kappa}$ | HS bound | Vol | Iters |
|---|---|---|---|---|---|---|
| Proj | 2 | 2b, d | 0.1854 | 0.1860 | 0.5001 | 32 |
| SLP | 2 | 2c, e | 0.1850 | 0.1860 | 0.5001 | 12 |
| Proj | 3 | 3b, d | 0.2304 | 0.2308 | 0.5000 | 34 |
| SLP | 3 | 3c, e | 0.1640 | 0.2308 | 0.5000 | 43 |

$$\begin{aligned} \min_{\Omega\in\mathcal{U}_{\mathrm{ad}}} \quad & -\bar{\kappa}(\Omega) \\ \text{s.t.} \quad & \mathrm{Vol}(\Omega) = 1/2, \\ & C_i(\Omega) = 0, \; i = 1, \dots, 6, \\ & a(\boldsymbol{u}, \boldsymbol{v}) = l(\boldsymbol{v}), \; \forall \boldsymbol{v} \in V. \end{aligned} \tag{43}$$

The constraints $C_i(\Omega)$ are given by

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_1 = \bar{C}_{1111} - \bar{\kappa} - \bar{\mu}, \tag{44}$$

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_2 = \bar{C}_{2222} - \bar{\kappa} - \bar{\mu}, \tag{45}$$

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_3 = \sqrt{2}(\bar{C}_{1122} - \bar{\kappa} + \bar{\mu}), \tag{46}$$

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_4 = 2\bar{C}_{1112}, \tag{47}$$

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_5 = 2\bar{C}_{2212}, \tag{48}$$

$$\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}C_6 = 2(\bar{C}_{1212} - \bar{\mu}), \tag{49}$$

where $\bar{\mu}$ is the isotropic shear modulus given by

$$\bar{\mu} = \frac{1}{8}(\bar{C}_{1111} + \bar{C}_{2222}) - \frac{1}{4}\bar{C}_{1122} + \frac{1}{2}\bar{C}_{1212}. \tag{50}$$

Analogous expressions appear in three dimensions and the number of isotropy constraints increases from 6 to 21 (Challis 2009). It should be noted that the term $\sqrt{4\bar{\kappa}^2 + 8\bar{\mu}^2}$ appears as a normalisation coefficient and is considered constant for the purpose of the shape differentiation. Furthermore, owing to the symmetry of the isotropy constraints the extended constraint shape sensitivities have a nullity of two. Our method handles these with no special treatment.

To visualise the behaviour of the isotropy constraints in the iteration history, we define the effective anisotropy

$\bar{\mathcal{A}}$ to be the sum of squares of the violation of these constraints. That is,

$$\bar{\mathcal{A}}(\Omega) = \sqrt{\sum C_i(\Omega)^2}. \tag{51}$$

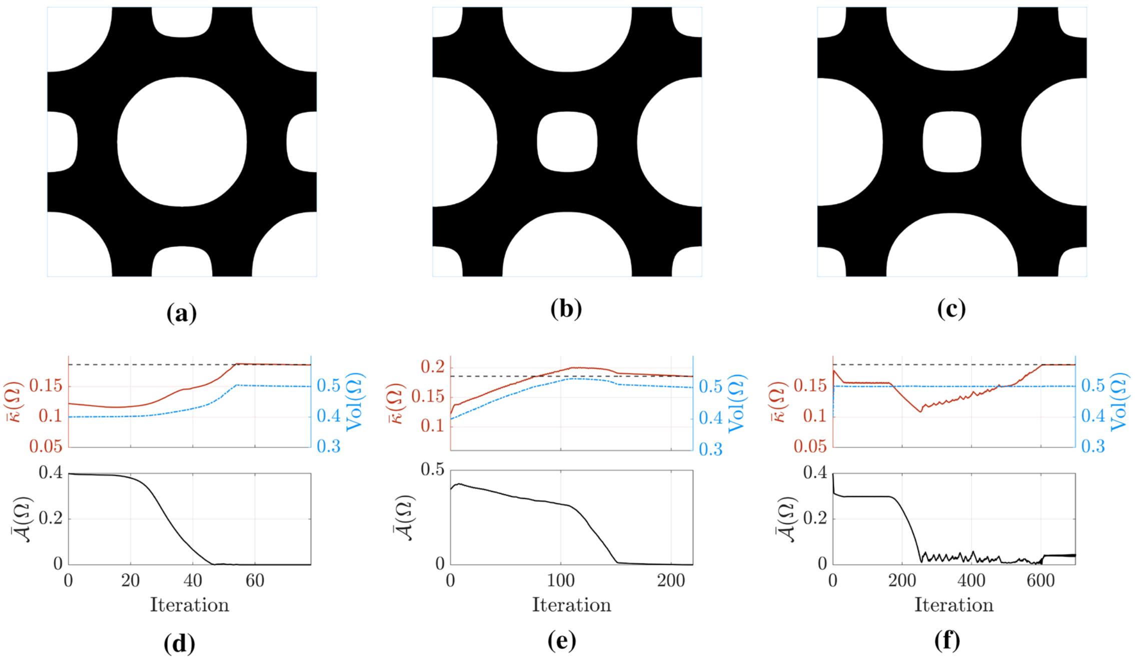
For the two-dimensional problem we implement both the full set of isotropy constraints as well as the single constraint $\bar{\mathcal{A}} = 0$. We find that the SLP method struggles with the full set of constraints. For this reason we instead use $\bar{\mathcal{A}} = 0$ as the isotropy constraint for the SLP method. For the Hilbertian projection method we find that the full set of constraints is more effective. This matches previous literature (e.g. Challis et al. 2008; Challis 2009).

Figure 4 shows the two-dimensional results for the Hilbertian projection method with the full set of constraints and single anisotropy measure constraint, and SLP with the anisotropy measure constraint. We omit the starting structure as this is the same as previously (Fig. 2a). Figure 5 shows the three-dimensional results for the Hilbertian projection method and SLP. Table 3 shows a summary of the results for this example.

In two dimensions, the Hilbertian projection method performs well for the full set of constraints and the single anisotropy measure constraint. The number of iterations is markedly lower when using the full set of constraints (78 vs. 220). This is unsurprising because including the individual symmetry constraints enables the projection method to improve each constraint separately (Challis et al. 2008; Challis 2009). The final optimisation values with the full set or single constraint are very similar, being 99.68% and 99.69% of the HS bound, whilst the final structures are geometrically identical under a periodic shift of half the cell edge length along each coordinate direction. The resulting structures match other classical results (Sect. 2.10.3, Bendsøe and Sigmund 2004). In contrast, the SLP method does not manage to reduce the measure of anisotropy to zero and reaches the maximum number of iterations. However, the final structure obtained with SLP (Fig. 4c) is very close to those obtained with the Hilbertian projection method (Fig. 4a and b). We suspect that SLP fails to converge due to the difficulty associated with choosing the trust region constraints.

For the three-dimensional results, we find that the Hilbertian projection method with the full set of constraints converges in 53 iterations to 99.12% of the HS bound. The final structure again matches known results from the literature (Sect. 2.10.3, Bendsøe and Sigmund 2004). For SLP, the optimisation algorithm fails to converge in 1000 iterations and the final structure is clearly not optimal.

These results demonstrate that the Hilbertian projection method is able to handle a large number of constraints (22 in 3 dimensions) without requiring any parameter tuning of $\alpha_{\min}$ or $\lambda$. In addition, the optimisation histories for the



(a) (b) (c)

(d) (e) (f)

**Fig. 4** Two-dimensional optimisation results for Example 2: maximum bulk modulus with isotropy. **a**–**c** show the final structures for the Hilbertian projection method with the full set of constraints, single anisotropy constraint, and SLP with the single anisotropy constraint, respectively, whilst **d**–**f** show the respective iteration histories. The Hashin-Shtrikman upper bound for the bulk modulus is given by the dashed black line. Due to very little change between iteration 600 and 1000, the upper bound on the $x$-axis in **f** has been reduced to 700

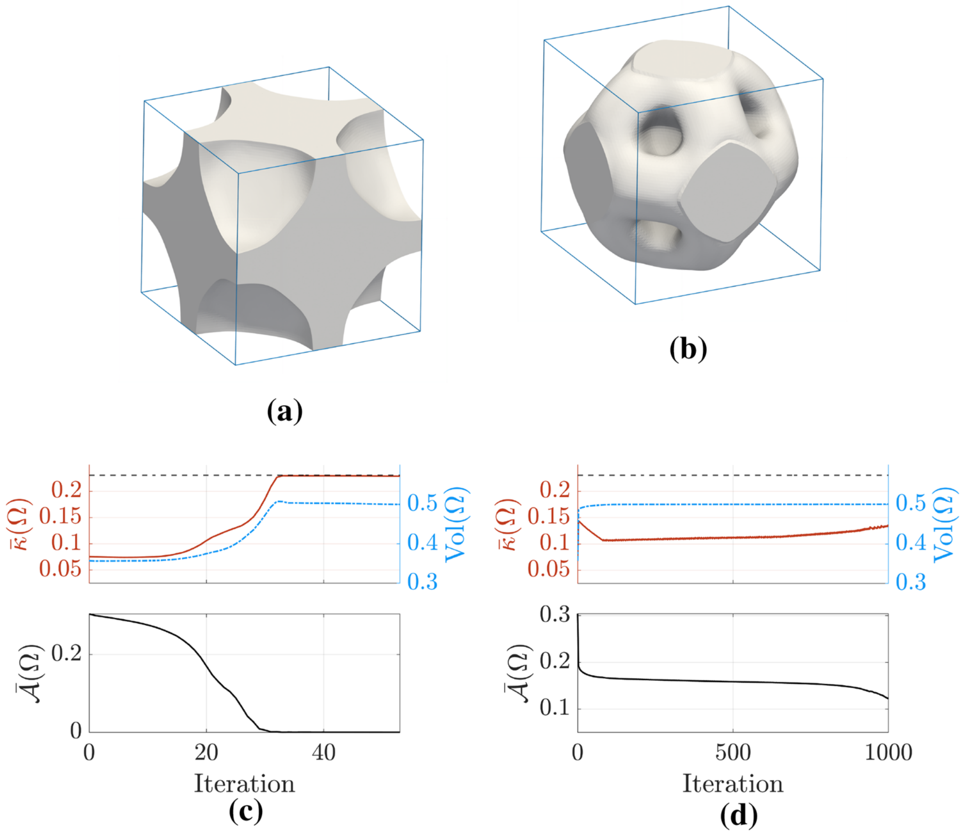


**Fig. 5** Three-dimensional optimisation results for Example 2: maximum bulk modulus with isotropy. **a** and **b** show the final structures for the Hilbertian projection method and SLP, respectively, whilst **c** and **d** show the respective iteration histories. The Hashin–Shtrikman upper bound for the bulk modulus is given by the dashed black line

**Table 3** Summary of optimisation results for Example 2: maximum bulk modulus with isotropy

| Method | $d$ | Iso. const. type | Fig | $\bar{\kappa}$ | HS bound | $\bar{\mathcal{A}}$ | Vol | Iters |
|---|---|---|---|---|---|---|---|---|
| Proj | 2 | Full set | 4a, d | 0.1854 | 0.1860 | 0.0001 | 0.5000 | 78 |
| Proj | 2 | Aniso. meas | 4b, e | 0.1854 | 0.1860 | 0.0001 | 0.5000 | 220 |
| SLP | 2 | Aniso. meas | 4c, f | 0.1856 | 0.1860 | 0.0363 | 0.5010 | 1000* |
| Proj | 3 | Full set | 5a, c | 0.2287 | 0.2308 | 0.0001 | 0.5000 | 53 |
| SLP | 3 | Aniso. meas | 5b, d | 0.1295 | 0.2308 | 0.1348 | 0.5007 | 1000* |

The asterisk denotes failure to converge

Hilbertian projection method with the full set of isotropy constraints (Figs. 4d and 5c) are smooth and converge fairly quickly.

### 5.3 Example 3: auxetic materials

In this example we consider two-dimensional minimum volume auxetic materials with a Poisson's ratio of −0.5. For the problem setup, we consider a bounding domain $D = [0, 1]^2$ that contains a solid phase and void phase. As previously mentioned, the solid phase is constructed from an isotropic material with $E = 1$ and $\nu = 0.3$.

To obtain an effective Poisson's ratio $\bar{\nu} = -0.5$, we require that $\bar{C}_{1111} = \bar{C}_{2222}$ and prescribe a value to $\bar{C}_{1111}$ and $\bar{C}_{1122}$ so that the effective Poisson's ratio $\bar{\nu}$ given by

$$\bar{\nu} = \frac{\bar{C}_{1122}}{\bar{C}_{1111}} \tag{52}$$

gives the required $\bar{\nu} = -0.5$.

It may be noted that this is similar to the approach taken by Vogiatzis et al. (2017). However, they instead minimise a weighted sum of the square of the difference between $\bar{C}_{ijkl}$ and its prescribed value subject to a volume constraint.

For the purpose of this example we choose $\bar{C}_{1111} = 0.1$, which results in $\bar{C}_{1122} = -0.05$. The resulting optimisation problem is then

$$
\begin{aligned}
\min_{\Omega\in\mathcal{U}_{\mathrm{ad}}} \ & \mathrm{Vol}(\Omega) \\
\text{s.t.} \ & \bar{C}_{1111}(\Omega) = 0.1, \\
& \bar{C}_{2222}(\Omega) = 0.1, \\
& \bar{C}_{1122}(\Omega) = -0.05, \\
& \bar{C}_{1112}(\Omega) = 0, \\
& \bar{C}_{2212}(\Omega) = 0, \\
& a(\boldsymbol{u},\boldsymbol{v}) = l(\boldsymbol{v}),\ \forall \boldsymbol{v} \in V.
\end{aligned} \tag{53}
$$

For this example we use $\gamma_{\max} = 0.05$ and $\alpha^2_{\min} = 0.5$. This means that the optimiser favours improvement of the constraints rather than the objective and does not move too quickly to avoid disconnecting in the first few iterations.

Figure 6 shows the starting structure (Fig. 6a) and optimisation results (Fig. 6b and c) for this problem using the Hilbertian projection method. We use a periodic starting structure with sixteen equally spaced holes. The method converges in 61 iterations with a final volume of 0.3159 and Poisson's ratio of −0.4998. In contrast, SLP fails because the optimiser prioritises the objective leading to a disconnected solid phase and the algorithm is unable to recover.

## 5.4 Example 4: multi-phase materials

For our final example we consider two-dimensional multi-phase maximum bulk modulus problems with and without isotropy constraints. We consider a bounding domain $D = [0, 1]^2$ that contains two solid phases and void. The solid phase contained in $\Omega_2$ is an isotropic material with $E = 1$ and $\nu = 0.3$, whilst $\Omega_3$ contains isotropic material with $E = 1/2$ and $\nu = 0.3$. The void phase is contained in $\Omega_1$ and $\Omega_4$. As previously, most void phase is removed from the mesh (see Sect. 4) whilst any material close to the interface is specified as weak material with $E = 10^{-3}$ and $\nu = 0.3$.

We consider two optimisation problems. The first is maximum bulk modulus subject to volume constraints on $\Omega_2$ and $\Omega_3$ given by

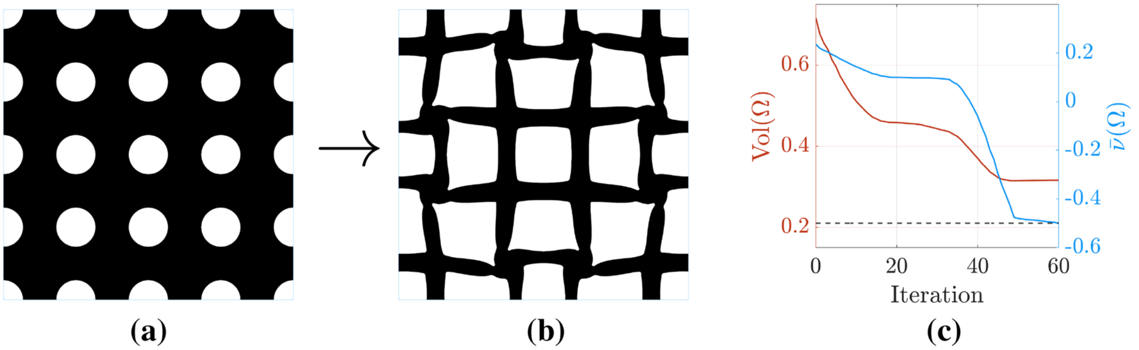
$$
\begin{aligned}
\min_{\mathcal{D}_1,\mathcal{D}_2\in\mathcal{U}_{\mathrm{ad}}} \ & -\bar{\kappa}(\mathcal{D}_1,\mathcal{D}_2) \\
\text{s.t.} \ & \mathrm{Vol}_{\Omega_2}(\mathcal{D}_1,\mathcal{D}_2) = 1/4, \\
& \mathrm{Vol}_{\Omega_3}(\mathcal{D}_1,\mathcal{D}_2) = 1/4, \\
& a(\boldsymbol{u},\boldsymbol{v}) = l(\boldsymbol{v}),\ \forall \boldsymbol{v} \in V.
\end{aligned} \tag{54}
$$

The second is maximum bulk modulus subject to macroscopic isotropy constraints and volume constraints on $\Omega_2$ and $\Omega_3$ given by

$$
\begin{aligned}
\min_{\mathcal{D}_1,\mathcal{D}_2\in\mathcal{U}_{\mathrm{ad}}} \ & -\bar{\kappa}(\mathcal{D}_1,\mathcal{D}_2) \\
\text{s.t.} \ & \mathrm{Vol}_{\Omega_2}(\mathcal{D}_1,\mathcal{D}_2) = 1/4, \\
& \mathrm{Vol}_{\Omega_3}(\mathcal{D}_1,\mathcal{D}_2) = 1/4, \\
& C_i(\mathcal{D}_1,\mathcal{D}_2) = 0,\ i = 1,\dots,6, \\
& a(\boldsymbol{u},\boldsymbol{v}) = l(\boldsymbol{v}),\ \forall \boldsymbol{v} \in V,
\end{aligned} \tag{55}
$$

where the constraints $C_i$ are as in Example 2. For SLP, we use the single anisotropy constraint and for the Hilbertian projection method we use the full set of isotropy constraints. For these examples we use $\gamma_{\max} = 0.05$ so that the optimiser does not evolve the designs too quickly. It should be noted that for the case of only volume constraints the extended constraint shape sensitivities have a nullity of one (in $\boldsymbol{\theta}_1$) and zero (in $\boldsymbol{\theta}_2$) owing to the structure of the shape derivatives for the volume constraints. For the case of volume and isotropy constraints the extended constraint shape sensitivities have a nullity of three (in $\boldsymbol{\theta}_1$) and two (in $\boldsymbol{\theta}_2$) due to the underlying symmetry of the shape derivatives for the volume and isotropy constraints. Our method handles these with no special treatment.

We initialise with two overlapping level set functions that give a starting structure completely comprised of the less stiff material ($\Omega_3$) and the void phase. Regions of stiffer material ($\Omega_2$) are readily generated during the optimisation via independent evolution of the two level set functions. We use a starting structure of four equally spaced holes for the optimisation problem without isotropy constraints and



(a) (b) (c)

**Fig. 6** Optimisation results for Example 3: auxetic materials. For the starting structure **a**, **b** shows the final structure, whilst **c** shows the iteration history for the Hilbertian projection method. The desired Poisson's ratio of −0.5 is given by the dashed black line

nine equally spaced holes for the problem with isotropy constraints.

Figures 7 and 8 show the optimisation results and history without and with isotropy constraints, respectively. We denote the stiff and less stiff material phase by blue and green, respectively, whilst the smooth interface is given by the dark green overlap. We include the Hashin–Shtrikman–Walpole (HSW) upper bound (Walpole 1966) for this problem as the black-dashed line in these figures. Table 4 shows a summary of the results.

For the case of no isotropy constraints, both the Hilbertian projection method and SLP converge to roughly 96% of the HSW bound, whilst the resulting structures (Fig. 7b and c) are geometrically similar apart from the thin interface that presents in the results for the Hilbertian projection method. The difference between our results and the theoretical upper bound is likely due to the use of the approximate formula for the shape derivative. Indeed, results from Allaire et al. (2014a) showed that the approximate formula yields slightly less optimal results than the true or Jacobian-free counterpart. The iteration history for the Hilbertian projection method (Fig. 7d) is fairly smooth, whilst the history for SLP (Fig. 7e) moves rapidly at the beginning of the optimisation to satisfy the volume constraints and increase the objective. As previously noted, this is likely due to the trust region constraints that, in this case, need to be chosen to be more conservative.

With the addition of isotropy constraints, we again find that the Hilbertian projection method works well without significant parameter tuning. In 148 iterations, the optimisation algorithm is able to satisfy all constraints and obtain a final objective value that is 93.44% of the HS bound. In contrast, SLP does not converge within 1000 iterations.

## 6 Conclusion

In this paper we have presented a Hilbertian extension of the projection method for constrained level set-based topology optimisation. At its core, the method relies on the Hilbertian extension-regularisation method in which a set of identification problems are solved over a Hilbert space $H$ on $D$ with inner product $\langle\cdot,\cdot\rangle_H$. This procedure naturally extends shape sensitivities onto the bounding domain $D$ and enriches them with the regularity of $H$. For a constrained optimisation problem the projection method framework aims for the best first-order improvement of the objective in addition to first-order improvement of the constraints. These requirements for the projection method may then be reposed in the Hilbertian framework in terms of $H$ and $\langle\cdot,\cdot\rangle_H$. We satisfy these reposed requirements by defining the normal velocity of the level set function as a linear combination of the orthogonal projection operator applied to the extended objective shape sensitivity and basis functions for the extended constraint

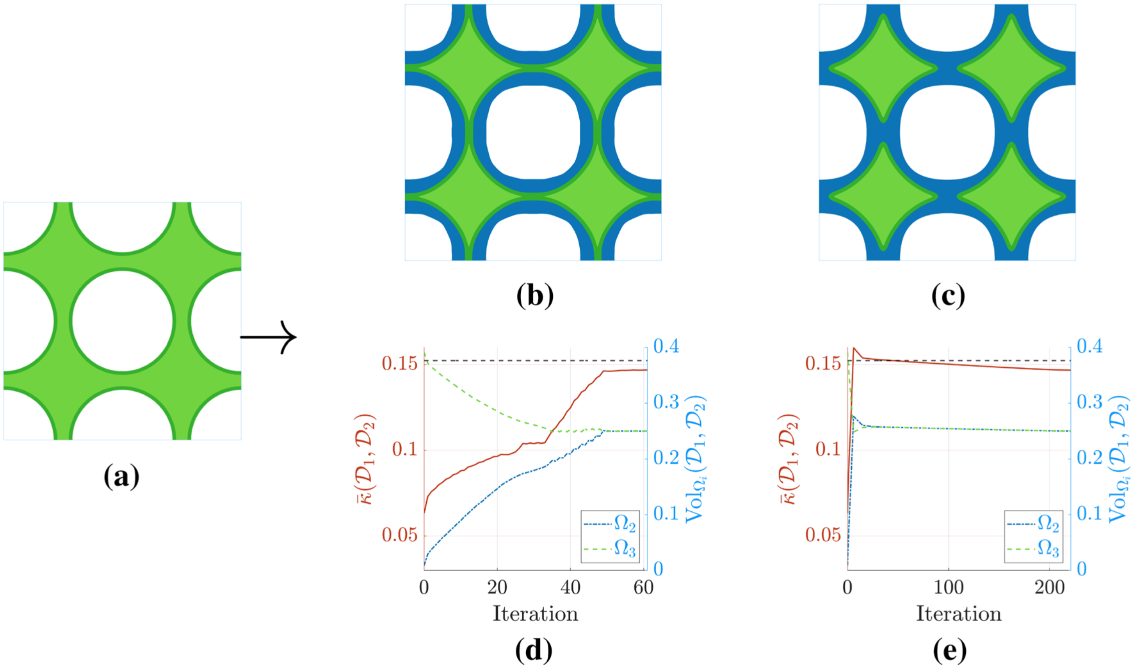


**Fig. 7** Optimisation results for Example 4: maximum bulk modulus multi-phase materials without isotropy constraints. For the starting structure **a**, **b** and **c** show the final structures for the Hilbertian projection method and SLP, respectively, whilst **d** and **e** show the respective iteration histories. The Hashin–Shtrikman–Walpole upper bound for the bulk modulus is given by the dashed black line. Note that the volume constraint for each phase is 0.25

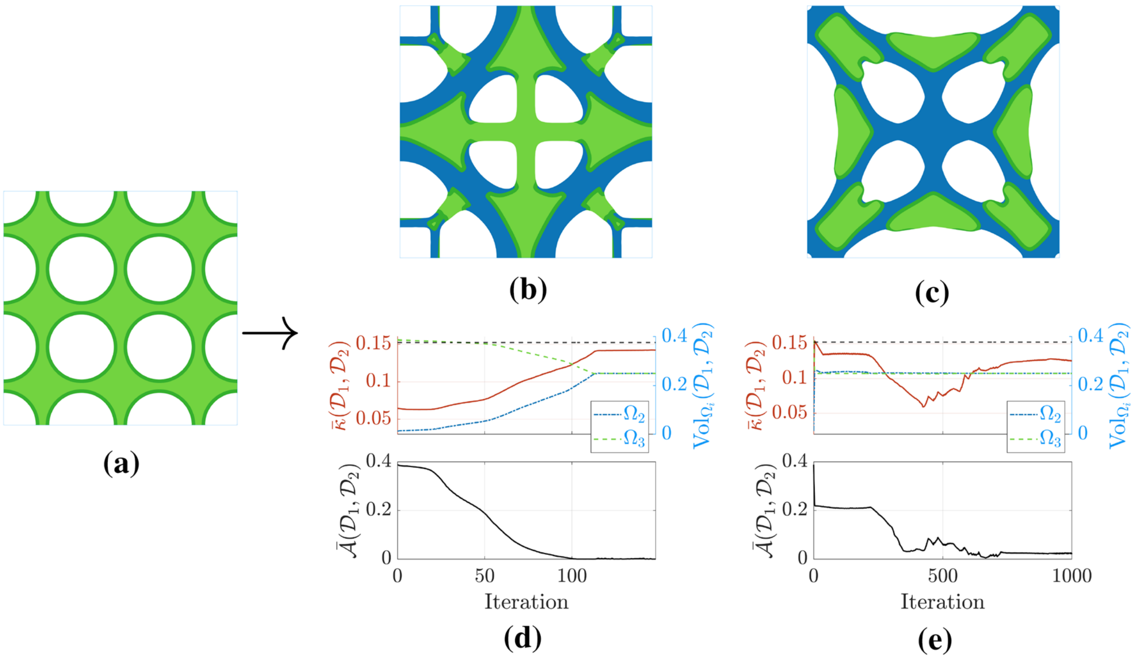


**Fig. 8** Optimisation results for Example 4: maximum bulk modulus multi-phase materials with isotropy constraints. For the starting structure **a**, **b** and **c** show the final structures for the Hilbertian projection method and SLP, respectively, whilst **d** and **e** show the respective iteration histories. The Hashin–Shtrikman–Walpole upper bound for the bulk modulus is given by the dashed black line. Note that the volume constraint for each phase is 0.25

**Table 4** Summary of optimisation results for Example 4: two-dimensional maximum bulk modulus multi-phase materials

| Method | Iso. const. type | Fig | $\bar{\kappa}$ | HSW bound | $\bar{A}$ | $\mathrm{Vol}_{\Omega_2}$ | $\mathrm{Vol}_{\Omega_3}$ | Iters |
|---|---|---|---|---|---|---|---|---|
| Proj | NA | 7a, d | 0.1469 | 0.1524 | N.A | 0.2501 | 0.2500 | 61 |
| SLP | NA | 7b, e | 0.1468 | 0.1524 | N.A | 0.2500 | 0.2501 | 221 |
| Proj | Full set | 8b, d | 0.1424 | 0.1524 | 0.0001 | 0.2500 | 0.2500 | 148 |
| SLP | Aniso. meas | 8c, e | 0.1254 | 0.1524 | 0.0236 | 0.2506 | 0.2497 | 1000* |

The asterisk denotes failure to converge

shape sensitivities. Owing to the Hilbertian extension-regularisation of shape sensitivities, the chosen normal velocity is already extended onto the bounding domain $D$ and endowed with the regularity of $H$.

To demonstrate the Hilbertian projection method for constrained level set-based topology optimisation we have solved several example microstructure optimisation problems with multiple constraints. We showed that the Hilbertian projection method successfully handled all of these optimisation problems with little-to-no tuning of the parameter $\alpha_{\min}$ that controls the balance between improving the objective and constraints. The Hilbertian projection method also naturally handles linearly dependent constraint shape sensitivities. Such linear dependencies often appear in microstructure optimisation and multi-phase optimisation problems.

We found that our method performs well when compared to a Hilbertian sequential linear programming (SLP) method (Allaire et al. 2021b). For problems only involving volume constraint(s), both methods converged to appropriate optimised microstructures. However, SLP did not successfully solve some of the more complex example optimisation problems. These results demonstrate the capacity of the Hilbertian projection method and likely other projection/null space methods (e.g. Barbarosie et al. 2020; Feppon et al. 2020) for solving constrained level set-based topology optimisation problems.

Applying multiple constraints is challenging in level set-based topology optimisation owing to the reliance on implicitly defined domains. Alongside other recent work (Barbarosie et al. 2020; Feppon et al. 2020; Allaire et al. 2021b), our proposed Hilbertian projection method makes significant progress towards improving the capacity of conventional level set-based methods for constrained topology optimisation. Furthermore, due to its generality, the Hilbertian projection method is not confined to microstructure optimisation. It may be applied to any topology optimisation problem to be solved in a level set framework, including macroscopic or multi-physics problems. Inequality constraints could likely be incorporated into the method using slack variables. In addition, the method is not confined to Eulerian level set methods and can be readily applied to Lagrangian or body-fitted level set methods (e.g. Allaire et al. 2014b). These extensions could be considered in the future work.

## Appendix 1

***Proof of Lemma 2*** We proceed via Céa's method (Céa 1986): suppose we define the Lagrangian $\mathcal{L}$ to be

$$\begin{aligned}&\mathcal{L}(\Omega,\tilde{\boldsymbol{u}}^{(ij)},\tilde{\boldsymbol{u}}^{(kl)})\\&=\int_\Omega C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\bar{\varepsilon}^{(kl)}_{rs}\ \mathrm{d}\Omega\\&+\int_\Omega C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})\ \mathrm{d}\Omega.\end{aligned}\tag{A1}$$

We do not include auxiliary fields as it turns out that the problem is self-adjoint.

Under a variation $\delta\tilde{\boldsymbol{u}}^{(ij)}$ of $\tilde{\boldsymbol{u}}^{(ij)}$ with $ij\neq kl$, the corresponding variation of $\mathcal{L}$ is

$$\begin{aligned}\frac{\delta\mathcal{L}}{\delta\tilde{\boldsymbol{u}}^{(ij)}}&=\int_\Omega C_{pqrs}\delta\tilde{u}^{(ij)}_{p,q}\bar{\varepsilon}^{(kl)}_{rs}\ \mathrm{d}\Omega\\&\quad+\int_\Omega C_{pqrs}\delta\tilde{u}^{(ij)}_{p,q}\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})\ \mathrm{d}\Omega\\&=\int_\Omega C_{pqrs}(\bar{\varepsilon}^{(kl)}_{rs}+\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)}))\delta\tilde{u}^{(ij)}_{p,q}\ \mathrm{d}\Omega\\&=\int_\Omega\sigma^{(kl)}_{pq}\delta\tilde{u}^{(ij)}_{p,q}\ \mathrm{d}\Omega\\&=-\int_\Omega\sigma^{(kl)}_{pq,q}\delta\tilde{u}^{(ij)}_{p}\ \mathrm{d}\Omega+\int_{\partial\Omega}\sigma^{(kl)}_{pq}n_q\delta\tilde{u}^{(ij)}_{p}\ \mathrm{d}\Gamma\end{aligned}$$

where symmetry of material coefficients has been used along with integration by parts. Requiring $\mathcal{L}$ to be stationary gives the state equations for stress under loading $\bar{\varepsilon}^{kl}_{rs}$. In particular, allowing arbitrary $\delta\tilde{u}^{(ij)}_p$ within $\Omega$ gives $\sigma^{(kl)}_{pq,q}=0$ in $\Omega$ and allowing arbitrary $\delta\tilde{u}^{(ij)}_p$ on $\partial\Omega$ gives $\sigma^{(kl)}_{pq}n_q=0$ on $\partial\Omega$.

Next, under a variation $\delta\tilde{\boldsymbol{u}}^{(kl)}$ of $\tilde{\boldsymbol{u}}^{(kl)}$ with $ij\neq kl$, the corresponding variation of $\mathcal{L}$ is

$$\begin{aligned}\frac{\delta\mathcal{L}}{\delta\tilde{\boldsymbol{u}}^{(kl)}}&=\int_\Omega C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\delta\tilde{u}^{(kl)}_{r,s}\ \mathrm{d}\Omega\\&=\int_\Omega\sigma^{(ij)}_{rs}\delta\tilde{u}^{(kl)}_{r,s}\ \mathrm{d}\Omega\\&=-\int_\Omega\sigma^{(ij)}_{rs,s}\delta\tilde{u}^{(kl)}_{r}\ \mathrm{d}\Omega+\int_{\partial\Omega}\sigma^{(ij)}_{rs}n_s\delta\tilde{u}^{(kl)}_{r}\ \mathrm{d}\Gamma\end{aligned}$$

where symmetry of material coefficients has been used along with integration by parts. Requiring $\mathcal{L}$ to be stationary gives the state equations for stress under loading $\bar{\varepsilon}^{(ij)}_{pq}$. It should be noted that when $ij=kl$, we can apply the product rule which results in the state equations for stress under a constant macroscopic strain $\bar{\varepsilon}^{(ij)}_{pq}$.

Together, the above results give the state equations for the fields $\tilde{\boldsymbol{u}}^{(ij)}$ and $\tilde{\boldsymbol{u}}^{(kl)}$, as required.

We also require that the Lagrangian $\mathcal{L}$ equals the objective at the solution to the state equations. Indeed, at the solution to the state equations, we obtain

$$\mathcal{L}(\Omega)=\bar{C}_{ijkl}(\Omega)\tag{A2}$$

as required.

The shape derivative of $\mathcal{L}$ at fixed $\tilde{\boldsymbol{u}}^{(ij)}$ and $\tilde{\boldsymbol{u}}^{(kl)}$ can then be calculated using Lemma 1 to be

$$\begin{aligned}C'(\Omega)(\boldsymbol{\theta})&=\mathcal{L}'(\Omega)(\boldsymbol{\theta})|_{\tilde{\boldsymbol{u}}^{(ij)},\tilde{\boldsymbol{u}}^{(kl)}}\\&=\int_{\partial\Omega}C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\\&\quad\times(\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})+\bar{\varepsilon}^{(kl)}_{rs})\ \boldsymbol{\theta}\cdot\boldsymbol{n}\ \mathrm{d}\Gamma.\end{aligned}\tag{A3}$$

## Appendix 2

***Proof of Lemma 3*** Similarly to Appendix 1, suppose we define the Lagrangian $\mathcal{L}$ to be

$$\begin{aligned}&\mathcal{L}(\mathcal{D}_1,\mathcal{D}_2,\tilde{\boldsymbol{u}}^{(ij)},\tilde{\boldsymbol{u}}^{(kl)})\\&=\int_D C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\bar{\varepsilon}^{(kl)}_{rs}\ \mathrm{d}\Omega\\&+\int_D C_{pqrs}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})\ \mathrm{d}\Omega\end{aligned}\tag{B4}$$

where $C_{pqrs}=C_{pqrs}(d_{\mathcal{D}_1},d_{\mathcal{D}_2})$. As previously, stationarity of $\mathcal{L}$ under variations $\delta\tilde{\boldsymbol{u}}^{(ij)}$ and $\delta\tilde{\boldsymbol{u}}^{(kl)}$ retrieve the state equations and at the solution to the state equations $\mathcal{L}(\Omega)=\bar{C}_{ijkl}(\Omega)$. The given Lagrangian therefore satisfies the requirements for Céa's method.

Using differentiability of the signed distance function $d_{\mathcal{D}_i}$ (Lemma 2.4, Proposition 2.5, Allaire et al. 2014a) and the chain rule, we have

$$\begin{aligned}&\bar{C}'_{ijkl}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\\&=\int_D d'_{\mathcal{D}_1}(\boldsymbol{\theta}_1)\frac{\partial C_{pqrs}}{\partial d_{\mathcal{D}_1}}(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})\\&\qquad\times(\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})+\bar{\varepsilon}^{(kl)}_{rs})\,\mathrm{d}\Omega\end{aligned}\tag{B5}$$

where $d'_{\mathcal{D}_i}$ is the shape derivative of the signed distance function for $\boldsymbol{x}\in D\setminus\Sigma$ given by

$$d'_{\mathcal{D}_1}(\boldsymbol{x})=-\boldsymbol{\theta}(p_{\partial\mathcal{D}_1}(\boldsymbol{x}))\cdot\boldsymbol{n}(p_{\partial\mathcal{D}_1}(\boldsymbol{x}))\tag{B6}$$

where $p_{\partial\mathcal{D}_1}(\boldsymbol{x})$ is the projection of a point $x$ onto the boundary $\partial\mathcal{D}_1$ and $\Sigma$ is the set of points in the skeleton of $\partial\mathcal{D}_1$ (Definition 2.3, Allaire et al. 2014a).

Using this and the Jacobian-free coarea formula (Corollary 2.13, Equation 2.15, Allaire et al. 2014a) results in

$$\begin{aligned}&\bar{C}'_{ijkl}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\\&=-\int_{\partial\mathcal{D}_1}\boldsymbol{\theta}_1\cdot\boldsymbol{n}\int_{\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D}H'_\eta(d_{\mathcal{D}_1})\frac{\partial C_{pqrs}}{\partial g}\\&\quad\times(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq})(\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})+\bar{\varepsilon}^{(kl)}_{rs})\,\mathrm{d}z\,\mathrm{d}\Gamma\end{aligned}\tag{B7}$$

where $g(x)=H_\eta(x)$.

Finally, the support of $H'_\eta(x)$ is $|x|<2\eta$, so the integral over $\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D$ is restricted to a tubular region about $\partial\mathcal{D}_1$ (Allaire et al. 2014a). Therefore, for small $\eta$, we may assume that

$$\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})(\boldsymbol{z})\approx\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})(\boldsymbol{y})\tag{B8}$$

and

$$d_{\mathcal{D}_2}(\boldsymbol{z})\approx d_{\mathcal{D}_2}(\boldsymbol{y})\tag{B9}$$

where $\boldsymbol{z}\in\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D$ and $\boldsymbol{y}\in\partial\mathcal{D}_1$. In addition, the derivative $\frac{\partial C_{pqrs}}{\partial g}(H_\eta(d_{\mathcal{D}_1}))$ is independent of $d_{\mathcal{D}_1}$ by Eq. 23. Equation B7 can therefore be written as

$$\begin{aligned}&\bar{C}'_{ijkl}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\\&\approx-\int_{\partial\mathcal{D}_1}\boldsymbol{\theta}_1\cdot\boldsymbol{n}\,\frac{\partial C_{pqrs}}{\partial g}\left(\varepsilon_{pq}(\tilde{\boldsymbol{u}}^{(ij)})+\bar{\varepsilon}^{(ij)}_{pq}\right)\\&\qquad\times\left(\varepsilon_{rs}(\tilde{\boldsymbol{u}}^{(kl)})+\bar{\varepsilon}^{(kl)}_{rs}\right)\\&\qquad\times\left[\int_{\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D}H'_\eta(d_{\mathcal{D}_1})\,\mathrm{d}z\right]\mathrm{d}\Gamma\end{aligned}\tag{B10}$$

Finally, it can be shown using elementary vector calculus that

$$\int_{\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D}H'_\eta(d_{\mathcal{D}_1}(z))\,\mathrm{d}z=1\tag{B11}$$

which completes this portion of the proof.

For Eq. 25, (Corollary 2.8, Allaire et al. 2014a) and the chain rule gives

$$\begin{aligned}&\mathrm{Vol}'_{\Omega_1}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\\&=\int_D d'_{\mathcal{D}_1}(\boldsymbol{\theta}_1)H'_\eta(d_{\mathcal{D}_1})H_\eta(d_{\mathcal{D}_2})\,\mathrm{d}\Omega\end{aligned}\tag{B12}$$

As previously mentioned, shape differentiability of $d_{\mathcal{D}_i}$ along with the Jacobian-free coarea formula gives

$$\begin{aligned}&\mathrm{Vol}'_{\Omega_1}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)=\int_{\partial\mathcal{D}_1}\boldsymbol{\theta}_1\cdot\boldsymbol{n}\\&\times\left[\int_{\mathrm{ray}_{\partial\mathcal{D}_1}(\boldsymbol{x})\cap D}H'_\eta(d_{\mathcal{D}_1})H_\eta(d_{\mathcal{D}_2})\,\mathrm{d}s\right]\mathrm{d}\Gamma\end{aligned}\tag{B13}$$

The prior approximations then give the result

$$\mathrm{Vol}'_{\Omega_1}(\mathcal{D}_1,\mathcal{D}_2)(\boldsymbol{\theta}_1)\approx\int_{\partial\mathcal{D}_1}\boldsymbol{\theta}_1\cdot\boldsymbol{n}\,H_\eta(d_{\mathcal{D}_2})\mathrm{d}\Gamma\tag{B14}$$

which concludes the proof.

**Acknowledgements** This work was supported by the Australian Research Council through the Discovery Grant scheme (DP220102759). Computational resources used in this work were provided by the eResearch Office, Queensland University of Technology. The first author is supported by a QUT Postgraduate Research Award and a Supervisor Top-Up Scholarship. The authors would also like to thank the anonymous reviewers for their constructive comments that have resulted in improvements to the manuscript.

**Author contributions** ZJW (zach.wegert@hdr.qut.edu.au) and VJC (vivien.challis@qut.edu.au) are co-corresponding authors of this paper.

**Funding** Open Access funding enabled and organized by CAUL and its Member Institutions.

## Declarations

**Conflicts of interest** The authors have no competing interests to declare that are relevant to the content of this article.

**Replication of results** As this work is a part of a new project, we do not provide the source code. However, we do provide Algorithm 1 to help readers reproduce results. In addition, we provide all parameter values along with an iteration history for all problems. Interested readers can contact the authors for further information.

## References

Allaire G, Jouve F, Toader AM (2004) Structural optimization using sensitivity analysis and a level-set method. J Comput Phys 194(1):363–393. https://doi.org/10.1016/j.jcp.2003.09.032

Allaire G, Dapogny C, Delgado G, Michailidis G (2014a) Multi-phase structural optimization via a level set method. ESAIM Control Optim Calc Var 20(2):576–611. https://doi.org/10.1051/cocv/2013076

Allaire G, Dapogny C, Frey P (2014b) Shape optimization with a level set based mesh evolution method. Comput Methods Appl Mech Eng 282:22–53. https://doi.org/10.1016/j.cma.2014.08.028

Allaire G, Jouve F, Michailidis G (2016) Thickness control in structural optimization via a level set method. Struct Multidisc Optim 53(6):1349–1382. https://doi.org/10.1007/s00158-016-1453-y

Allaire G, Bogosel B, Godoy M (2021a) Shape optimization of an imperfect interface: steady-state heat diffusion. J Optim Theory Appl 191(1):169–201. https://doi.org/10.1007/s10957-021-01928-6

Allaire G, Dapogny C, Jouve F (2021b) Shape and topology optimization. Elsevier, Amsterdam, pp 1–132. https://doi.org/10.1016/bs.hna.2020.10.004

Andreassen E, Lazarov BS, Sigmund O (2014) Design of manufacturable 3d extremal elastic microstructure. Mech Mater 69(1):1–10. https://doi.org/10.1016/j.mechmat.2013.09.018

Badia S, Verdugo F (2020) Gridap: An extensible finite element toolbox in Julia. J Open Source Softw 5(52):2520. https://doi.org/10.21105/joss.02520

Barbarosie C, Toader AM, Lopes S (2020) A gradient-type algorithm for constrained optimization with application to microstructure optimization. Discrete Contin Dyn Syst B 25(5):1729–1755. https://doi.org/10.3934/dcdsb.2019249

Bendsøe MP (1989) Optimal shape design as a material distribution problem. Struct Optim 1(4):193–202. https://doi.org/10.1007/BF01650949

Bendsøe MP, Sigmund O (2004) Topology optimization theory, methods and applications, 2nd edn. Springer, Berlin. https://doi.org/10.1007/978-3-662-05086-6

Besard T, Foket C, De Sutter B (2018) Effective extensible programming: unleashing Julia on GPUs. IEEE Trans Parallel Distrib Syst. https://doi.org/10.1109/TPDS.2018.2872064

Birgin EG, Martínez JM (2009) Practical augmented Lagrangian methods. Springer, Boston, pp 3013–3023. https://doi.org/10.1007/978-0-387-74759-0_517

Cao S, Wang H, Lu X, Tong J, Sheng Z (2021) Topology optimization considering porosity defects in metal additive manufacturing. Appl Sci 11(12):5578. https://doi.org/10.3390/app11125578

Céa J (1986) Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût. ESAIM Math Model Numer Anal 20(33):371–402. https://doi.org/10.1051/m2an/1986200303711

Challis VJ, Roberts AP, Wilkins AH (2008) Design of three dimensional isotropic microstructures for maximized stiffness and conductivity. Int J Solids Struct 45(14–15):4130–4146. https://doi.org/10.1016/j.ijsolstr.2008.02.025

Challis VJ (2009) Multi-property topology optimisation with the level-set method. The University of Queensland, Brisbane

Deaton J, Grandhi R (2013) A survey of structural and multidisciplinary continuum topology optimization: post 2000. Struct Multidisc Optim 49:1–38. https://doi.org/10.1007/s00158-013-0956-z

Dunning I, Huchette J, Lubin M (2017) Jump: a modeling language for mathematical optimization. SIAM Rev 59(2):295–320. https://doi.org/10.1137/15M1020575

Dunning PD, Kim HA (2015) Introducing the sequential linear programming level-set method for topology optimization. Struct Multidisc Optim 51(3):631–643. https://doi.org/10.1007/s00158-014-1174-z

Dunning PD, Brampton CJ, Kim HA (2015) Simultaneous optimisation of structural topology and material grading using level set method. Mater Sci Technol 31(8):884–894. https://doi.org/10.1179/1743284715Y.0000000022

Faure A, Michailidis G, Parry G, Vermaak N, Estevez R (2017) Design of thermoelastic multi-material structures with graded interfaces using topology optimization. Struct Multidisc Optim 56(4):823–837. https://doi.org/10.1007/s00158-017-1688-2

Feppon F, Allaire G, Bordeu F, Dapogny C (2019) Shape optimization of a coupled thermal fluid-structure problem in a level set mesh evolution framework. SeMA J 76(3):413–458. https://doi.org/10.1007/s40324-018-00185-4

Feppon F, Allaire G, Dapogny C (2020) Null space gradient flows for constrained optimization with applications to shape optimization. ESAIM Control Optim Cal Var 26:90. https://doi.org/10.1051/cocv/2020015

Gibiansky LV, Sigmund O (2000) Multiphase composites with extremal bulk modulus. J Mech Phys Solids 48(3):461–498. https://doi.org/10.1016/S0022-5096(99)00043-5

Guo X, Zhang W, Zhong W (2014) Explicit feature control in structural topology optimization via level set method. Comput Methods Appl Mech Eng 272:354–378. https://doi.org/10.1016/j.cma.2014.01.010

Hashin Z, Shtrikman S (1963) A variational approach to the theory of the elastic behaviour of multiphase materials. J Mech Phys Solids 11(2):127–140. https://doi.org/10.1016/0022-5096(63)90060-7

Kambampati S, Gray JS, Alicia Kim H (2020) Level set topology optimization of structures under stress and temperature constraints. Comput Struct 235(106):265. https://doi.org/10.1016/j.compstruc.2020.106265

Nocedal J, Wright SJ (2006) Numerical optimization, 2nd edn. Springer, New York

Osher S, Fedkiw R (2006) Level set methods and dynamic implicit surfaces. Applied mathematical sciences, 1st edn. Springer, Cham

Peng D, Merriman B, Osher S, Zhao H, Kang M (1999) A pde-based fast local level set method. J Comput Phys 155(2):410–438. https://doi.org/10.1006/jcph.1999.6345

Rozvany GIN, Zhou M, Birker T (1992) Generalized shape optimization without homogenization. Struct Optim 4:250–252. https://doi.org/10.1007/BF01742754

Schropp J, Singer I (2000) A dynamical systems approach to constrained minimization. Numer Funct Anal Optim 21(3–4):537–551. https://doi.org/10.1080/01630560008816971

Sethian JA (1996) Level set methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Cambridge University Press, Cambridge

Sigmund O, Maute K (2013) Topology optimization approaches: a comparative review. Struct Multidisc Optim 48(6):1031–1055. https://doi.org/10.1007/s00158-013-0978-6

Silva ECN, Fonseca JSO, Kikuchi N (1998) Optimal design of periodic piezocomposites. Comput Methods Appl Mech Eng 159(1–2):49–77. https://doi.org/10.1016/S0045-7825(98)80103-5

Svanberg K (1987) The method of moving asymptotes-a new method for structural optimization. Int J Numer Methods Eng 24(2):359–373. https://doi.org/10.1002/nme.1620240207

Verdugo F, Badia S (2022) The software design of gridap: a finite element package based on the julia JIT compiler. Comput Phys Commun 276(108):341. https://doi.org/10.1016/j.cpc.2022.108341

Vogiatzis P, Chen S, Wang X, Li T, Wang L (2017) Topology optimization of multi-material negative poisson's ratio metamaterials using a reconciled level set method. Comput Aided Des 83:15–32. https://doi.org/10.1016/j.cad.2016.09.009

Wächter A, Biegler L (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math Progr 106:25–57. https://doi.org/10.1007/s10107-004-0559-y

Walpole L (1966) On bounds for the overall elastic moduli of inhomogeneous systems-I. J Mech Phys Solids 14(3):151–162. https://doi.org/10.1016/0022-5096(66)90035-4

Wang MY, Wang X (2004) "Color" level sets: a multi-phase method for structural topology optimization with multiple materials. Comput Methods Appl Mech Eng 193(6–8):469–496. https://doi.org/10.1016/j.cma.2003.10.008

Wang MY, Wang X, Guo D (2003) A level set method for structural topology optimization. Comput Methods Appl Mech Eng 192(1–2):227–246. https://doi.org/10.1016/S0045-7825(02)00559-5

Wegert ZJ, Roberts AP, Challis VJ (2022) Multi-objective structural optimisation of piezoelectric materials. Int J Solids Struct 248(111):666. https://doi.org/10.1016/j.ijsolstr.2022.111666

Yvonnet J (2019) Computational homogenization of heterogeneous materials with finite elements. Solid mechanics and its applications, 1st edn. Springer, Cham. https://doi.org/10.1007/978-3-030-18383-7

Zhuang C, Xiong Z, Ding H (2010) Topology optimization of multi-material for the heat conduction problem based on the level set method. Eng Optim 42(9):811–831. https://doi.org/10.1080/03052150903443780