University of Basel

Boundary Element Methods for Shape Optimisation in Homogenisation

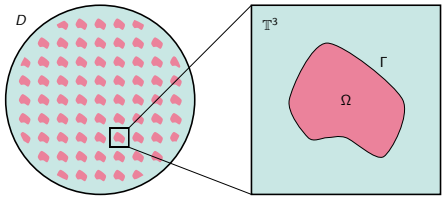
Helmut Harbrecht, Michael Multerer, Remo von Rickenbach

Department of Mathematics and Computer Science, University of Basel

Bernoullis Tafelrunde, 27 September 2021

Motivation

- Shape Optimisation has many applications in the industry.
- A bone implant should have the same material properties as a real bone.
- We can think of a bone as a periodic scaffold structure.
- How shall we choose the shapes of the holes?



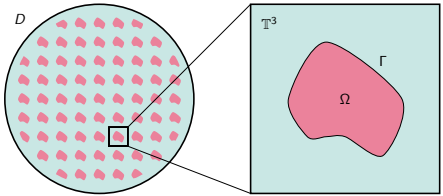
Outline

1. Motivation
2. Homogenisation
3. Shape Optimisation
4. Boundary Elements
5. Implementation
6. Numerical Examples

Homogenisation

- Define $\mathbb{T}^3 = \mathbb{R}^3 / \left[-\frac{1}{2}, \frac{1}{2}\right]^3$.
- Let $\mathbf{A} : \mathbb{T}^3 \to \mathbb{R}^{3\times 3}$, $\varepsilon > 0$ and $\mathbf{A}^\varepsilon(\boldsymbol{x}) = \mathbf{A}\left(\frac{\boldsymbol{x}}{\varepsilon}\right)$.
- For a domain $D \subseteq \mathbb{R}^3$, we are interested in the solution u^ε of the partial differential equation

$$\begin{cases} -\operatorname{div}\left(\mathbf{A}^\varepsilon \nabla u^\varepsilon\right) = f & \text{in } D, \\ u^\varepsilon = 0 & \text{on } \partial D. \end{cases}$$

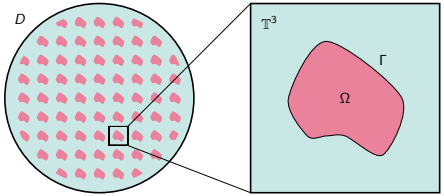


Homogenisation

- Homogenisation treats the behaviour of u^ε in the limit case $\varepsilon \to 0$.
- There exists a homogenised problem

$$\begin{cases} -\operatorname{div}\left(\mathbf{A}_0 \nabla u_0\right) = f & \text{in } D, \\ u_0 = 0 & \text{on } \partial D. \end{cases}$$

with a homogenised solution u_0 such that $u^\varepsilon \rightharpoonup u_0$ weakly in $H^1(D)$.



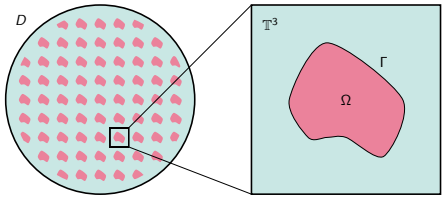
Homogenisation

The effective material tensor $\mathbf{A}_0 = \left[a_{i,j}\right]_{i,j=1}^{3}$, is given by the entries

$$a_{i,j} = \int_{\mathbb{T}^3} \left\langle \mathbf{A}(\boldsymbol{e}_i + \nabla w_i),\ \boldsymbol{e}_j + \nabla w_j \right\rangle \mathrm{d}V,$$

where for $i = 1, 2, 3$, w_i is the solution of the cell problem

find $w_i \in H^1_{per}(\mathbb{T}^3)$ s.t. $\operatorname{div}\left(\mathbf{A}(\boldsymbol{e}_i + \nabla w_i)\right) = 0.$



Homogenisation

- For a scaffold structure, we define $\mathbf{A} = \mathbb{1}_{\mathbb{T}^3 \setminus \overline{\Omega}} \mathbf{I}$.
- Then, the cell problem is equivalent to the exterior Neumann problem

$$\begin{cases} \Delta w_i = 0 & \text{in } \mathbb{T}^3 \setminus \overline{\Omega}, \\ \partial_{\boldsymbol{n}} w_i = -n_i & \text{on } \Gamma = \partial\Omega. \end{cases}$$

- The compatibility condition is satisfied, as

$$\int_\Gamma -n_i \, \mathrm{d}A = -\int_\Gamma \langle \boldsymbol{e}_i, \boldsymbol{n} \rangle \, \mathrm{d}A = -\int_\Omega \operatorname{div}(\boldsymbol{e}_i) \, \mathrm{d}V = 0.$$

- The effective material tensor only depends on the size and the shape of Ω.

Shape Optimisation

For a desired material tensor $\mathbf{B}$, we define the shape functional

$$J(\Omega) = \frac{1}{2}\left\|\mathbf{A}_0(\Omega) - \mathbf{B}\right\|_F^2 = \frac{1}{2}\sum_{i,j=1}^{3}\left(a_{i,j}(\Omega) - b_{i,j}\right)^2.$$

Goal: Optimise the shape of Ω such that $J(\Omega)$ is minimal.

It is crucial to know the behaviour of $J(\Omega)$ when Ω is slightly deformed.

Shape Optimisation

- Let $\tilde{\boldsymbol{f}} : \left[-\frac{1}{2}, \frac{1}{2}\right]^3 \to \left[-\frac{1}{2}, \frac{1}{2}\right]^3$ be a sufficiently smooth vector field with compact support, and $\boldsymbol{f} : \mathbb{T}^3 \to \mathbb{T}^3$ be the periodic continuation of $\tilde{\boldsymbol{f}}$.
- We define the perturbation of the identity as

$$T_{t,\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{x} + t\boldsymbol{f}(\boldsymbol{x}).$$

  For $t > 0$ small enough, $T_{t,\boldsymbol{f}}$ is a diffeomorphism that preserves each cell.
- We define the shape derivative of $a_{i,j}$ in the direction of $\boldsymbol{f}$ as

$$a'_{i,j}[\boldsymbol{f}](\Omega) = \left[\frac{\mathrm{d}}{\mathrm{d}t} a_{i,j}\big(T_{t,\boldsymbol{f}}(\Omega)\big)\right]_{t=0} = \lim_{t\to 0} \frac{a_{i,j}\big(T_{t,\boldsymbol{f}}(\Omega)\big) - a_{i,j}(\Omega)}{t}.$$

- Consequently,

$$J'[\boldsymbol{f}](\Omega) = \sum_{i,j=1}^{3} a'_{i,j}[\boldsymbol{f}](\Omega)\big(a_{i,j}(\Omega) - b_{i,j}\big).$$

Shape Optimisation

Remember the solutions of the cell problems w_i. By defining $\phi_i = x_i + w_i$, it holds:

Theorem (Dambrine and Harbrecht, 2020)

The coefficient $a_{i,j}(\Omega)$ and its shape derivative are given by

$$a_{i,j}(\Omega) = \int_{\mathbb{T}^3 \setminus \overline{\Omega}} \left\langle \nabla \phi_i,\ \nabla \phi_j \right\rangle \, \mathrm{d}V,$$

$$a'_{i,j}[\boldsymbol{f}](\Omega) = -\int_{\Gamma} \left\langle \nabla_\Gamma \phi_i,\ \nabla_\Gamma \phi_j \right\rangle \langle \boldsymbol{f}, \boldsymbol{n} \rangle \, \mathrm{d}A,$$

where ∇_Γ is the tangential gradient.

Boundary Elements

Remember that we have to solve the exterior Neumann problems

$$\begin{cases} \Delta w_i = 0 & \text{in } \mathbb{T}^3 \setminus \overline{\Omega}, \\ \partial_{\boldsymbol{n}} w_i = -n_i & \text{on } \Gamma = \partial\Omega. \end{cases}$$

- In every iteration step, the shape of the domain Ω changes.
- The values of w_i will be needed on the boundary $\Gamma = \partial\Omega$ only.
- The resulting shape might not be convex.

Solution: Use Boundary Elements.

Boundary Elements: Theoretical Background

Definition

Let $d \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^d$, and let $\mathcal{L}$ be an elliptic, second order differential operator. The **fundamental solution** or **Green's function** is a function that satisfies the distributional equation

$$\mathcal{L}_{\boldsymbol{y}} G(\boldsymbol{x}, \boldsymbol{y}) = \delta_{\mathbf{0}}(\boldsymbol{x} - \boldsymbol{y}).$$

Examples:

$$\mathcal{L} = -\Delta, \qquad G(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} -\frac{1}{2\pi} \log\left(\|\boldsymbol{x} - \boldsymbol{y}\|\right), & d = 2, \\ \frac{1}{4\pi\|\boldsymbol{x}-\boldsymbol{y}\|}, & d = 3. \end{cases}$$

$$\mathcal{L} = -\Delta - \kappa^2 \mathcal{I}, \qquad G(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} \frac{i}{4} H_0^{(1)}\left(\kappa\|\boldsymbol{x} - \boldsymbol{y}\|\right), & d = 2, \\ \frac{\exp\left(i\kappa\|\boldsymbol{x}-\boldsymbol{y}\|\right)}{4\pi\|\boldsymbol{x}-\boldsymbol{y}\|}, & d = 3. \end{cases}$$

Boundary Elements: Theoretical Background

Let $\Omega \subseteq \mathbb{R}^3$ with a Lipschitz boundary $\Gamma = \partial\Omega$. Consider the interior Laplace problem with Dirichlet boundary conditions

$$\begin{cases} \Delta u = 0 & \text{in } \Omega, \\ u = g & \text{on } \Gamma. \end{cases}$$

Green's representation formula gives us

$$\begin{aligned} u(\boldsymbol{x}) &= -\int_\Omega \big(\Delta_{\boldsymbol{y}} G\big)(\boldsymbol{x},\boldsymbol{y}) u(\boldsymbol{y}) \,\mathrm{d}V_{\boldsymbol{y}} \\ &= \int_\Gamma G(\boldsymbol{x},\boldsymbol{y}) \big(\partial_{\boldsymbol{n}} u\big)(\boldsymbol{y}) \,\mathrm{d}A_{\boldsymbol{y}} - \int_\Gamma \big(\partial_{\boldsymbol{n}_{\boldsymbol{y}}} G\big)(\boldsymbol{x},\boldsymbol{y}) u(\boldsymbol{y}) \,\mathrm{d}A_{\boldsymbol{y}}, \qquad \boldsymbol{x} \in \Omega. \end{aligned}$$

Boundary Elements: Theoretical Background

Using the trace properties, one obtains

$$\mathcal{V}\partial_{\boldsymbol{n}} u = \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) u|_{\Gamma},$$

with

$$\mathcal{V}\varrho(\boldsymbol{x}) = \int_{\Gamma} G(\boldsymbol{x},\boldsymbol{y})\varrho(\boldsymbol{y})\ \mathrm{d}A_{\boldsymbol{y}}, \qquad \mathcal{K}\varrho(\boldsymbol{x}) = \int_{\Gamma} \partial_{\boldsymbol{n}_{\boldsymbol{y}}} G(\boldsymbol{x},\boldsymbol{y})\varrho(\boldsymbol{y})\ \mathrm{d}A_{\boldsymbol{y}}.$$

Having $\partial_{\boldsymbol{n}} u$, one can calculate $u(\boldsymbol{x})$ for every $\boldsymbol{x} \in \Omega$ according to Green's representation formula.

Boundary Elements: Theoretical Background

- On a Lipschitz domain in three dimensions, the operator $\mathcal{V} : H^{-1/2+s}(\Gamma) \to H^{1/2+s}(\Gamma)$ is continuous for $|s| \leq \frac{1}{2}$, and invertible for $s = 0$.
- On a Lipschitz domain in three dimensions, the operator $\mathcal{K} : H^{1/2+s}(\Gamma) \to H^{1/2+s}(\Gamma)$ is continuous for $|s| \leq \frac{1}{2}$, but $\ker\left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) = \operatorname{span}\{\mathbb{1}\}$.
- In a periodic setting, the fundamental solution satisfies $-\Delta_{\boldsymbol{y}} G(\boldsymbol{x}, \boldsymbol{y}) = \delta_{\boldsymbol{0}}(\boldsymbol{x} - \boldsymbol{y}) - 1$.
- In a periodic setting, the fundamental solution cannot be represented explicitly, but as the series

$$G(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{4\pi\|\boldsymbol{x} - \boldsymbol{y}\|} + \frac{\|\boldsymbol{x} - \boldsymbol{y}\|^2}{6} + \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \alpha_n^m \phi_n^m(\boldsymbol{x} - \boldsymbol{y}).$$

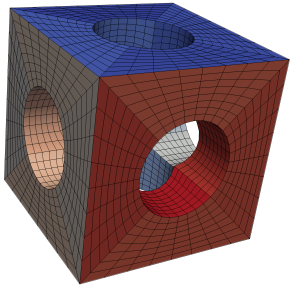
Here, $\alpha_n^m \in \mathbb{C}$ are scalar coefficients and ϕ_n^m denote the solid spherical harmonics.

Boundary Elements: Discretisation

Definition

For an admissible decomposition $\mathcal{T} = \{T_1, ..., T_N\}$ of Γ, with nodes $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_M\}$, we define the **ansatz space** $\boldsymbol{S_h^1(\Gamma)} = \text{span}\{\psi_1, ..., \psi_M\}$, where

$$\psi_i(\boldsymbol{x}) = \begin{cases} 1, & \boldsymbol{x} = \boldsymbol{x}_i, \\ 0, & \boldsymbol{x} = \boldsymbol{x}_j,\ i \neq j, \\ \text{transformed bilinear}, & \text{else.} \end{cases}$$



Boundary Elements: Discretisation

Remember that we want to solve

$$\mathcal{V}t = \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) g,$$

where $t = \partial_{\boldsymbol{n}} u$ and $g = u\big|_{\Gamma}$ are the given Dirichlet data.

Boundary Elements: Discretisation

Remember that we want to solve

$$\mathcal{V}t = \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) g,$$

where $t = \partial_{\boldsymbol{n}} u$ and $g = u\big|_\Gamma$ are the given Dirichlet data.

We can pose the variational problem

find $t \in L^2(\Gamma)$, such that

$$\langle \mathcal{V}t,\ \psi\rangle_\Gamma = \left\langle \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) g,\ \psi \right\rangle_\Gamma \qquad \text{for any } \psi \in L^2(\Gamma).$$

Boundary Elements: Discretisation

Remember that we want to solve

$$\mathcal{V}t = \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) g,$$

where $t = \partial_{\boldsymbol{n}} u$ and $g = u\big|_{\Gamma}$ are the given Dirichlet data.

If $Q_h : L^2(\Gamma) \to S_h^1(\Gamma)$ denotes the $L^2(\Gamma)$-orthogonal projection, we solve the Galerkin problem

$$\text{find } t_h \in S_h^1(\Gamma), \text{ such that}$$
$$\langle \mathcal{V} t_h,\ \psi_i \rangle_{\Gamma} = \left\langle \left(\mathcal{K} + \frac{1}{2}\mathcal{I}\right) Q_h g,\ \psi_i \right\rangle_{\Gamma}, \qquad i = 1, ..., M.$$

Boundary Elements: Discretisation

Let us define

$$\mathbf{K}_h = \left[\left\langle \mathcal{K}\psi_j, \psi_i \right\rangle_\Gamma\right]_{i,j=1}^{M}, \qquad \mathbf{V}_h = \left[\left\langle \mathcal{V}\psi_j, \psi_i \right\rangle_\Gamma\right]_{i,j=1}^{M},$$
$$\mathbf{M}_h = \left[\left\langle \psi_j, \psi_i \right\rangle_\Gamma\right]_{i,j=1}^{M}, \qquad \boldsymbol{g}_h = \left[\left\langle g, \psi_i \right\rangle_\Gamma\right]_{i=1}^{M}.$$

Then, the Galerkin variational formulation is equivalent to

$$\mathbf{V}_h \boldsymbol{t}_h = \left(\mathbf{K}_h + \frac{1}{2}\mathbf{M}_h\right)\mathbf{M}_h^{-1}\boldsymbol{g}_h.$$

Boundary Elements: Neumann Problem

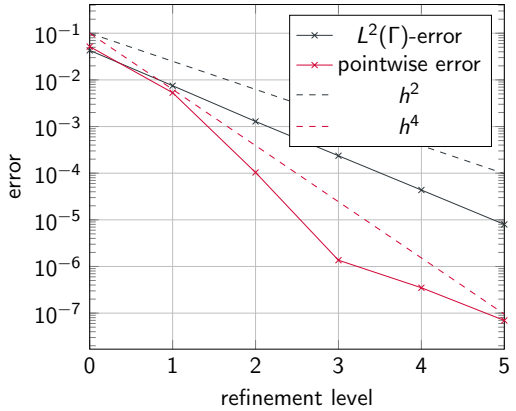
- For an exterior Neumann problem, the resulting equation is $\left(\mathcal{K} - \frac{1}{2}\mathcal{I}\right) u\big|_\Gamma = \mathcal{V}\partial_{\boldsymbol{n}} u$.
- The operator $\left(\mathcal{K} - \frac{1}{2}\mathcal{I}\right) : H^s(\Gamma) \to H^s(\Gamma)$ is continuously invertible for $s = 0, 1/2$.
- To make the solution of an interior Neumann problem unique, one can factor out the constant, i.e., set

$$\tilde{t}_h = t_h - \frac{\langle t_h,\ \mathbb{1}\rangle_\Gamma}{\|\mathbb{1}\|^2_{L^2(\Gamma)}}\mathbb{1}, \qquad \tilde{\boldsymbol{t}}_h = \boldsymbol{t}_h - \frac{\boldsymbol{t}_h^T \mathbf{M}_h \mathbf{1}_h}{\mathbf{1}_h^T \mathbf{M}_h \mathbf{1}_h}\mathbf{1}_h.$$

Boundary Elements: Convergence

Using the described procedure, one can show that

$$\left\|u - u_h\right\|_{L^2(\Gamma)} \leq Ch^2 \|u\|_{H^2(\Gamma)}, \qquad \left|u(\boldsymbol{x}) - u_h(\boldsymbol{x})\right| \leq Ch^4 \|u\|_{H^2(\Gamma)}, \quad \boldsymbol{x} \in \Omega.$$



Boundary Elements: Remarks

- The equation is solved on the boundary only.
- No volume mesh has to be generated, or updated, respectively.
- Singularities arising from the geometry are treated in a natural way.
- One dimension lower, but full matrices instead.
- There exist fast boundary element methods to overcome the full population.

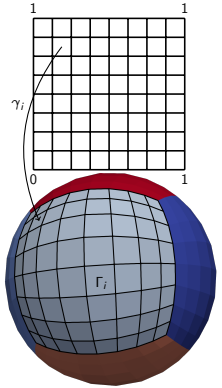
Implementation

We use the **B**oundary **E**lement **M**ethod **B**ased **E**ngineering **L**ibrary **BEMBEL**.

- Written in `C++`, compatible with `Eigen3`.
- Fast multipole method, combined with $\mathcal{H}^2$-matrices.
- Parametric surface representation.

Implementation: Shape Representation

- Decomposition into surface patches Γ_i with parametrisations γ_i.
- On each element, the parametrisation γ_i is approximated by a piecewise polynomial.
- In the case of a quintic polynomial, we achieve an $L^2\big([0,1]^2\big)$-accuracy of $\mathcal{O}(h^6)$.



Implementation: Shape Discretisation

- Take a correlation kernel $C(r)$, for example a Matérn kernel.
- Calculate the eigenfunctions of the corresponding Hilbert-Schmidt operator.
- This gives us a deformation basis, with some spacial relation to the geometry.
  - A Sphere delivers spherical harmonics.
  - Each geometry delivers another basis
- Calculate the deformation functions by a quintic interpolation.

Implementation: Shape Calculus

Lemma (Harbrecht, Multerer, and v. R., 2021)

It holds

$$a_{i,j}(\Omega) = \delta_{i,j}\big(1 - |\Omega|\big) - \int_\Gamma w_j \langle \boldsymbol{e}_i, \boldsymbol{n}\rangle \,\mathrm{d}A,$$

$$a'_{i,j}[\boldsymbol{f}](\Omega) = -\int_\Gamma \Big[\big\langle \boldsymbol{e}_i + \nabla_\Gamma w_i,\ \boldsymbol{e}_j + \nabla_\Gamma w_j\big\rangle - \langle \boldsymbol{e}_i, \boldsymbol{n}\rangle\langle \boldsymbol{n}, \boldsymbol{e}_j\rangle\Big]\langle \boldsymbol{f}, \boldsymbol{n}\rangle \,\mathrm{d}A.$$

- It holds
$$|\Omega| = \int_\Omega \operatorname{div}\left(\frac{\boldsymbol{x}}{3}\right) \,\mathrm{d}V = \frac{1}{3}\int_\Gamma \langle \boldsymbol{x}, \boldsymbol{n}\rangle \,\mathrm{d}A.$$
- For $\boldsymbol{b} \in \operatorname{span}\{\boldsymbol{n}\}^\perp$, and $\boldsymbol{c} \in \mathbb{R}^3$, we have $\langle \boldsymbol{b}, \boldsymbol{c}\rangle = \langle \boldsymbol{n} \times \boldsymbol{b},\ \boldsymbol{n} \times \boldsymbol{c}\rangle$. This allows us to use the surface curl instead of the tangential gradient.

Implementation: Shape Update

- Let us define

$$g(\boldsymbol{y}) = J\big((\mathbf{I} + y_1\boldsymbol{f}_1 + ... + y_N\boldsymbol{f}_N)(\Omega)\big).$$

  We use the direction of the steepest descent

$$\boldsymbol{d} = -\nabla g(\mathbf{0}) = -\big[J'[\boldsymbol{f}_1](\Omega), ..., J'[\boldsymbol{f}_N](\Omega)\big]^T.$$

- If $g(\boldsymbol{d}) < g(\mathbf{0})$, set

$$\Omega_{\text{new}} = \big(\mathbf{I} + d_1\boldsymbol{f}_1 + ... + d_N\boldsymbol{f}_N\big)(\Omega).$$

  If not, decrease the step size.

Implementation: Shape Update

- In practice, we calculate the quadratic polynomial $h(t)$, which interpolates $g(t\boldsymbol{d})$ at the points
$$\big(0,\ g(\mathbf{0})\big), \quad \big(0,\ g'(\mathbf{0})\big), \quad \big(1,\ g(\boldsymbol{d})\big).$$
Afterwards, we use its minimum $t^* \in (0,1)$ as the new step size.
- An evaluation of the shape functional is expensive, so we do not want to apply an Armijo line search.
- The BFGS method was also applied, but did not perform well.

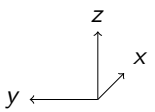
Numerical Examples

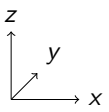
- The initial deformation basis is used throughout the whole process.
- In all of the subsequent examples, the optimisation was stopped if $J(\Omega) < 1\mathrm{e}{-5}$.
- The calculations were carried out on the fourth refinement level, resulting in

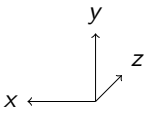
$$N \sim N_p\left(2^4 + 1\right)^2$$

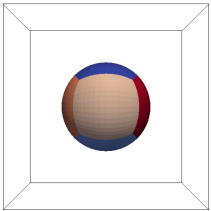
  unknowns.
- If not stated otherwise, $N = 16$ deformation functions were used.

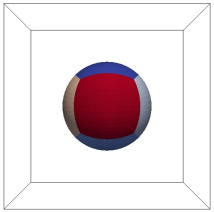
Numerical Examples: Sphere

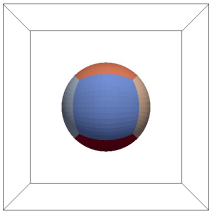






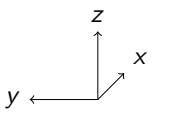


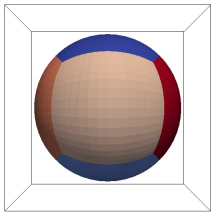


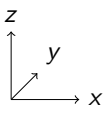


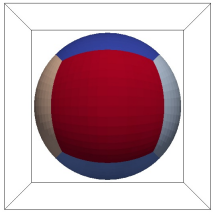
Desired Tensor $\mathbf{B} = 0.9\mathbf{I}$, Initial Guess $\mathcal{B}_{0.3}(0)$, Iterations: 3.

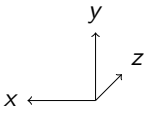
Numerical Examples: Sphere

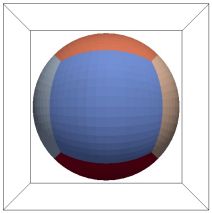






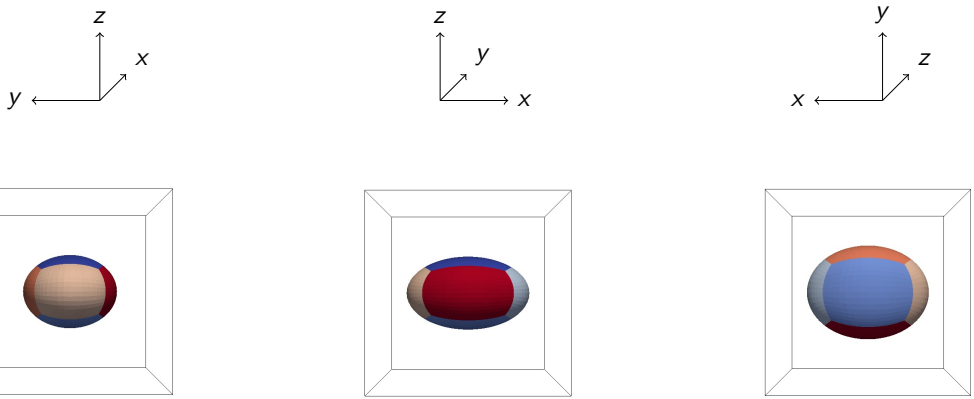






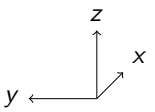
Desired Tensor $\mathbf{B} = 0.6\mathbf{I}$, Initial Guess $\mathcal{B}_{0.3}(0)$, Iterations: 6.

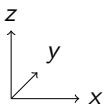
Numerical Examples: Sphere

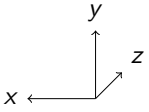


Desired Tensor $\mathbf{B} = \mathrm{diag}(0.9, 0.88, 0.86)$, Initial Guess $\mathcal{B}_{0.3}(0)$, Iterations: 17.

Numerical Examples: Sphere



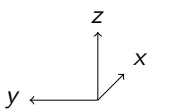


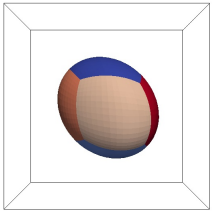


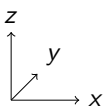
For the next example, we define the transformation matrix $\mathbf{T}$ as

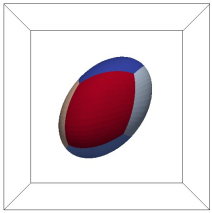
$$\mathbf{T} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} \end{bmatrix}.$$

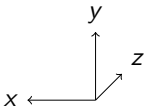
Numerical Examples: Sphere

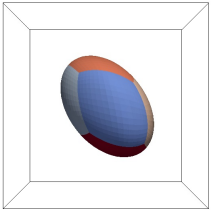






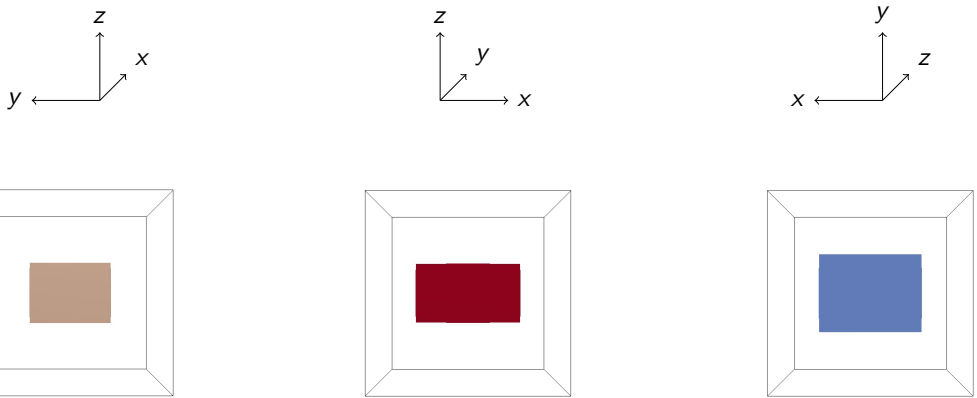






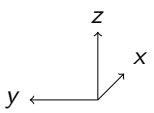
Desired Tensor $\mathbf{B} = \mathbf{T}\text{diag}(0.9, 0.88, 0.86)\mathbf{T}^T$, Initial Guess $\mathcal{B}_{0.3}(0)$, Iterations: 25.

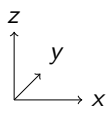
Numerical Examples: Cube

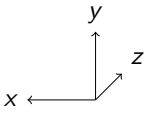


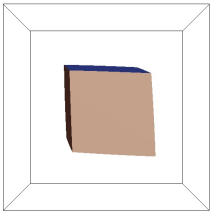
Desired Tensor $\mathbf{B} = \mathrm{diag}(0.9, 0.88, 0.86)$, Initial Guess $[-0.15, 0.15]^3$, Iterations: 13.

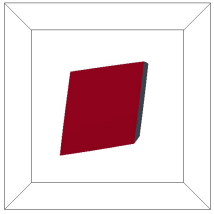
Numerical Examples: Cube

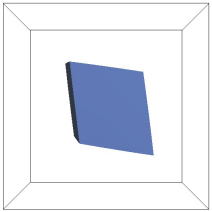






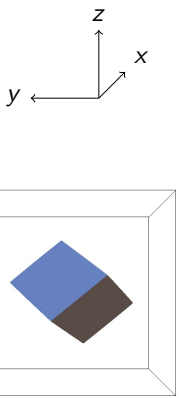


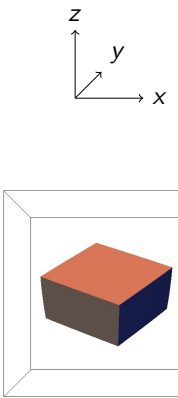


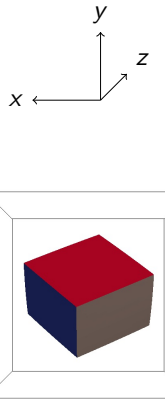


Desired Tensor $\mathbf{B} = \mathbf{T}\text{diag}(0.9, 0.88, 0.86)\mathbf{T}^T$, Initial Guess $[-0.15, 0.15]^3$, Iterations: 14.

Numerical Examples: Cube

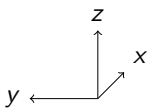


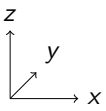


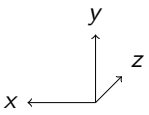


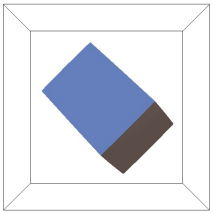
Desired Tensor $\mathbf{B} = \text{diag}(0.9, 0.88, 0.86)$, Initial Guess $\mathbf{T}[-0.15, 0.15]^3$, Iterations: 15.

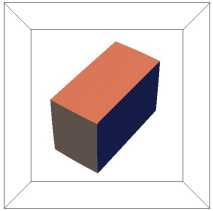
Numerical Examples: Cube

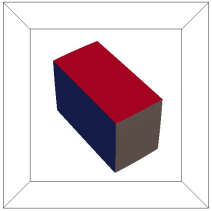












Desired Tensor $\mathbf{B} = \mathbf{T}\text{diag}(0.9, 0.88, 0.86)\mathbf{T}^T$, Initial Guess $\mathbf{T}[-0.15, 0.15]^3$, Iterations: 25.

Numerical Examples: Sphere and Cube

In the following examples, we use the initial guess

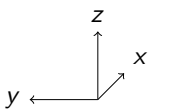
$$\mathcal{B}_{0.15}\left(\begin{bmatrix}-0.25\\-0.25\\-0.25\end{bmatrix}\right) \cup [0.175,\ 0.325]^3.$$

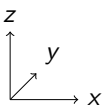
As the desired tensor, we use

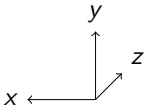
$$\mathbf{B} = \operatorname{diag}(1,\ 0.995,\ 0.99).$$

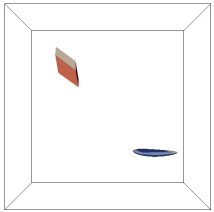
We also use a correlation length ℓ different from 1. If ℓ is small, the correlation is small.

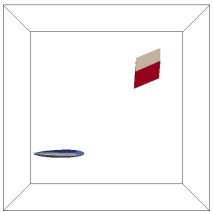
Numerical Examples: Sphere and Cube

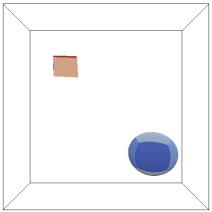






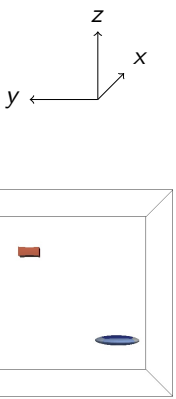


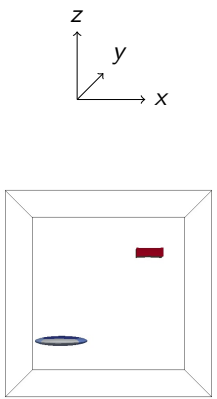


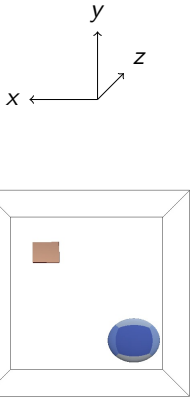


$\ell = 1$, Iterations: 12, Deformation Functions: 16.

Numerical Examples: Sphere and Cube







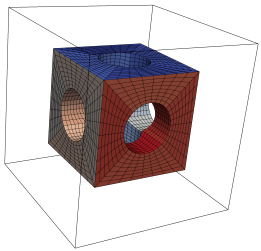
$\ell = \frac{1}{4}$, Iterations: 8, Deformation Functions: 50.

Numerical Examples: Toy

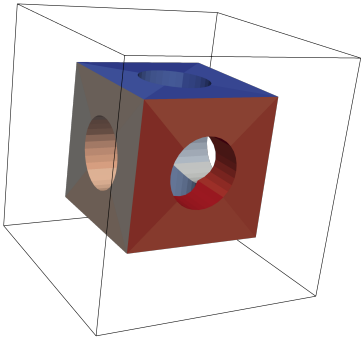
These calculations can be extended to higher-order ansatz spaces and more complex geometries.
In the following example, the initial toy box is contained in $[-0.3, 0.3]^3$. The desired tensor is

$$\mathbf{B} = \mathrm{diag}(0.82,\ 0.78,\ 0.74).$$

The calculations were performed with 200 piecewise (bi-)quadratic ansatz functions.

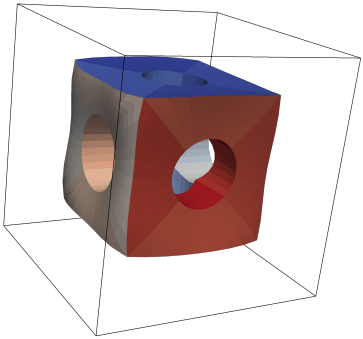


Numerical Examples: Toy



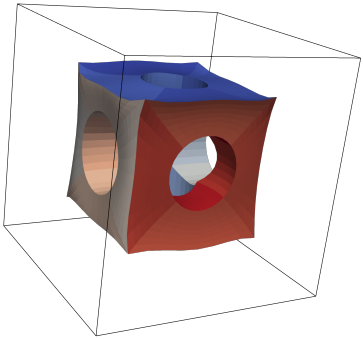
Iteration: 0, $J(\Omega) = 1.92\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



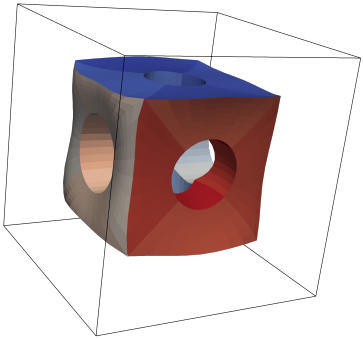
Iteration: 1, $J(\Omega) = 1.69\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



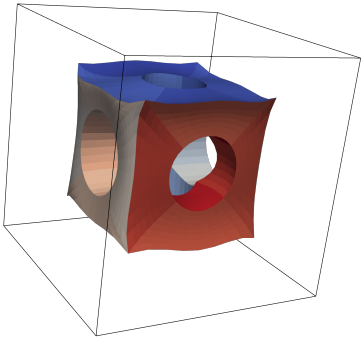
Iteration: 2, $J(\Omega) = 1.40\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



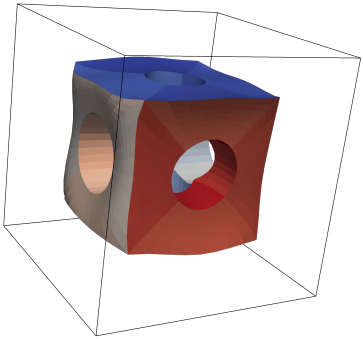
Iteration: 3, $J(\Omega) = 1.34\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



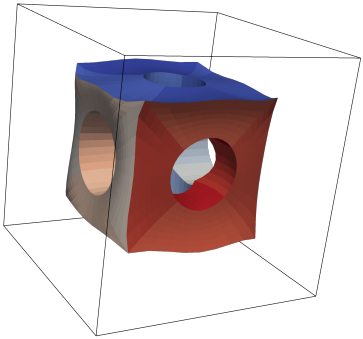
Iteration: 4, $J(\Omega) = 1.26\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



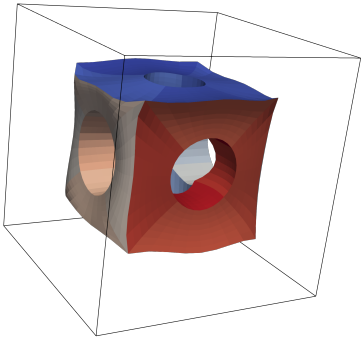
Iteration: 5, $J(\Omega) = 1.21\text{e}{-3}$, Step Size: 1.

Numerical Examples: Toy



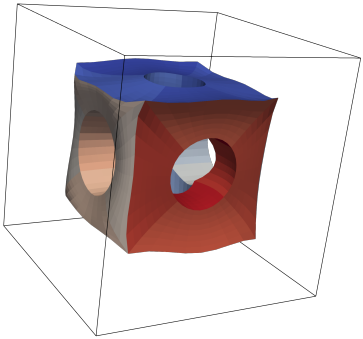
Iteration: 6, $J(\Omega) = 3.38\text{e}{-5}$, Step Size: 0.49.

Numerical Examples: Toy



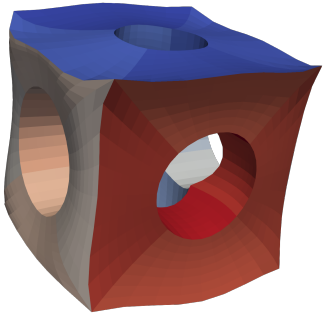
Iteration: 7, $J(\Omega) = 1.41\text{e}{-5}$, Step Size: 0.10.

Numerical Examples: Toy

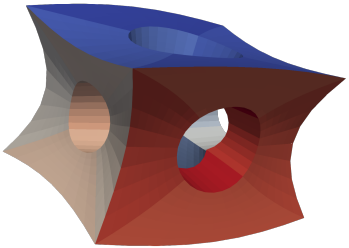


Iteration: 8, $J(\Omega) = 8.53\text{e}{-6}$, Step Size: 0.03.

Numerical Examples: Toy



200 deformation functions



50 deformation functions

Conclusion

- The final shape is not unique, but depends on the choice of the deformation functions and the initial shape.
- The shape optimisation procedure was successfully implemented with the boundary element method library **BEMBEL**.
- Boundary elements are useful for shape optimisation problems, as no mesh volume mesh has to be updated.
- The calculations can be extended to higher-order ansatz functions.
- The shape optimisation can be performed with a reference shape of an arbitrary topology.

Thank you for your attention!

M. Dambrine and H. Harbrecht. Shape optimization for composite materials and scaffold structures. *Multiscale Modeling & Simulation*, 18:1136–1152, 2020.

J. Dölz, H. Harbrecht, S. Kurz, M. Multerer, S. Schöps, and F. Wolf. Bembel: The fast isogeometric boundary element C++ library for Laplace, Helmholtz, and electric wave equation. *SoftwareX*, 11:100476, 2020.

H. Harbrecht, M. Multerer, and R. von Rickenbach. Isogeometric shape optimization for scaffold structures. ArXiv, May 2021. `https://arxiv.org/abs/2105.15158`.