Shape Optimization using Wavelet-BEM

Karsten Eppler
and
Helmut Harbrecht

K. Eppler
Institut für Mathematik
Technische Universität Berlin / Germany

H. Harbrecht
Institut für Informatik und Praktische Mathematik
Christian–Albrechts–Universität Kiel / Germany

Overview

- Introduction to shape optimization
- BIE formulation
- Wavelet based fast solution of the BIE
- A shape problem from linear elasticity
- Electromagnetic shaping
- Identification of voids or inclusions

Shape optimization

Problem:

$$J(\Omega) = \int_\Omega j\big(\mathbf{x}, u_\Omega(\mathbf{x})\big) d\mathbf{x} \to \min,$$

where

$$\begin{aligned} -\Delta u_\Omega &= f \quad \text{in } \Omega, \\ u_\Omega &= g \quad \text{on } \Gamma, \end{aligned}$$

subject to

$$\begin{aligned} C_i(\Omega) &= \int_\Omega h_i(\mathbf{x}) d\mathbf{x} \le c_i, \qquad i = 1, 2, \ldots, m, \\ C_i(\Gamma) &= \int_\Gamma h_i(\mathbf{x}) d\mathbf{x} \le c_i, \qquad i = m+1, m+2, \ldots, n. \end{aligned}$$

security set $D \subseteq \mathbb{R}^2 : \Omega \subseteq D$

$$\left.\begin{array}{l} j : D \times \mathbb{R} \times \mathbb{R} \to \mathbb{R} \\ f, g, h_i : D \to \mathbb{R} \end{array}\right\} \text{ are sufficiently smooth functions}$$

Applications:

- optimal shape design: electromagnets, wings, walls, ...
- identification of inclusions or obstacles
- computation of free (stationary) surfaces

Domain variation

The directional derivative of a shape functional $J(\Omega)$ is defined via

$$\boxed{\nabla J(\Omega)[\mathbf{V}] = \lim_{\varepsilon \to 0} \frac{J(\Omega_\varepsilon[\mathbf{V}]) - J(\Omega)}{\varepsilon}}$$

- Method of mappings/perturbation of identity (Murat/Simon):

  $$\Omega_\varepsilon[\mathbf{V}] = \{\mathbf{x} + \varepsilon \mathbf{V}(\mathbf{x}) : \mathbf{x} \in \Omega\}$$

  with some smooth function $\mathbf{V} : \Omega \to \mathbb{R}^2$

- Speed method (Sokolowski/Zolesio)

Differentiating the PDE

$$\boxed{\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, & -\Delta u_\varepsilon[\mathbf{V}] &= f \quad \text{in } \Omega_\varepsilon[\mathbf{V}], \\ u &= g \quad \text{on } \Gamma, & u_\varepsilon[\mathbf{V}] &= g \quad \text{on } \Gamma_\varepsilon[\mathbf{V}]. \end{aligned}}$$

- Material derivative $\dot{u}$:
  Transport $u_\varepsilon[\mathbf{V}]$ to $u^\varepsilon[\mathbf{V}]$ living on Ω and compute

  $$\dot{u}[\mathbf{V}] = \lim_{\varepsilon \to 0} \frac{u^\varepsilon[\mathbf{V}] - u}{\varepsilon}.$$

- Local shape derivative du:
  The derivative $du[\mathbf{V}]$ is defined pointwise by

  $$du(\mathbf{x})[\mathbf{V}] = \lim_{\varepsilon \to 0} \frac{u_\varepsilon[\mathbf{V}](\mathbf{x}) - u(\mathbf{x})}{\varepsilon}, \quad \mathbf{x} \in \Omega.$$

$$\boxed{\dot{u} = du + \mathbf{V} \nabla u}$$

Shape differential calculus

Consider the shape functional $J(\Omega) = \int_\Omega j(u, \mathbf{x}) d\mathbf{x}$

with state function

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega \\ u &= g && \text{on } \Gamma \end{aligned}$$

and adjoint state function

$$\begin{aligned} -\Delta p &= \frac{\partial j}{\partial u}(u, \mathbf{x}) && \text{in } \Omega \\ p &= 0 && \text{on } \Gamma. \end{aligned}$$

The gradient requires only functions on the boundary

$$\begin{aligned} \nabla J(\Omega)[\mathbf{V}] &= \int_\Gamma \langle \mathbf{V}, \mathbf{n} \rangle j(u, \mathbf{x}) d\sigma_{\mathbf{x}} + \int_\Omega du[\mathbf{V}] \frac{\partial j}{\partial u}(u, \mathbf{x}) d\mathbf{x} \\ &= \int_\Gamma \langle \mathbf{V}, \mathbf{n} \rangle \left\{ j(u, \mathbf{x}) - \frac{\partial p}{\partial \mathbf{n}} \frac{\partial (g-u)}{\partial \mathbf{n}} \right\} d\sigma_{\mathbf{x}}, \end{aligned}$$

since the local shape derivative satisfies

$$\begin{aligned} -\Delta du[\mathbf{V}] &= 0 && \text{in } \Omega, \\ du[\mathbf{V}] &= \langle \mathbf{V}, \mathbf{n} \rangle \frac{\partial (g-u)}{\partial \mathbf{n}} && \text{on } \Gamma. \end{aligned}$$

This result is consistent to the following theorem:

Theorem (Hadamard):

Provided that the shape functional $J(\Omega)$ is shape differentiable, the directional derivative $\nabla J(\Omega)[\mathbf{V}]$ is a scalar quantity defined on the boundary $\partial\Omega$.

Boundary variation

Boundary is descibed by a parametrization

$$\Gamma = \left\{\gamma(t) = \left[\begin{smallmatrix} x(t) \\ y(t) \end{smallmatrix}\right] : t \in [0,1]\right\}.$$

Boundary variation is defined via

$$\Gamma_\varepsilon[\mathbf{V}] = \{\gamma(t) + \varepsilon\mathbf{V}(t) : t \in [0,1]\}.$$

- Smooth fields (Kirsch/Kress/Potthast): $\boxed{\mathbf{V} : [0,1] \to \mathbb{R}^2}$
  $\rightsquigarrow$ no unique variation $\mathbf{V}$ per perturbed boundary
- Polar coordinates/starshaped domains (Eppler): $\boxed{\mathbf{V}(t) = r(t)\mathbf{e}_r(t)}$
  $\rightsquigarrow$ unique variation $\mathbf{V}$ per perturbed boundary
- Normal direction (Hadamard/Pironneau/Fuji): $\boxed{\mathbf{V} = \rho(t)\mathbf{n}(t)}$
  $\rightsquigarrow \Gamma \in C^\alpha \Rightarrow \Gamma_\varepsilon[\mathbf{V}] \in C^{\alpha-1}$ even if $\rho \in C^\infty_{\text{per}}([0,1])$

Algorithmical consequences

- only boundary updates are practicable
- boundary integral representation of the shape Hessian
- one has to compute the following quantities

$$\begin{aligned} \nabla J(\Omega) &: \quad \frac{\partial u}{\partial \mathbf{n}}, \frac{\partial p}{\partial \mathbf{n}} \\ \nabla^2 J(\Omega) &: \quad \frac{\partial du[\mathbf{V}]}{\partial \mathbf{n}}, \frac{\partial dp[\mathbf{V}]}{\partial \mathbf{n}}, \frac{\partial^2 u}{\partial \mathbf{n}^2}, \frac{\partial^2 p}{\partial \mathbf{n}^2}, \frac{\partial^2 u}{\partial \mathbf{n}\partial \mathbf{t}}, \frac{\partial^2 p}{\partial \mathbf{n}\partial \mathbf{t}} \end{aligned}$$

- BIE formulation seems to be very attractive

General setup

Consider the shape functional

$$J(\Omega) = \int_\Omega h(\mathbf{x})u(\mathbf{x}) + h_0(\mathbf{x})d\mathbf{x}$$

with the state function u and its adjoint p solving

$$\begin{aligned} \Delta u = f \text{ in } \Omega, &\qquad u = g \text{ on } \Gamma, \\ \Delta p = h \text{ in } \Omega, &\qquad p = 0 \text{ on } \Gamma. \end{aligned}$$

$$\begin{aligned}
J(\Omega) &= \int_\Omega (N_h\, f\ + h_0)d\mathbf{x} + \int_\Gamma g\,\frac{\partial N_h}{\partial \mathbf{n}}\,d\sigma - \int_\Gamma \frac{\partial u}{\partial \mathbf{n}}\,N_h\,d\sigma, \\
\nabla J(\Omega)[dr] &= \int_0^{2\pi} dr\,r\Big[(hg+h_0)\big|_\Gamma + \frac{\partial p}{\partial \mathbf{n}}\frac{\partial (g-u)}{\partial \mathbf{n}}\Big]\,d\phi, \\
\nabla^2 J(\Omega)[dr_1, dr_2] &= \int_0^{2\pi} dr_1 dr_2\Big[(hg+h_0)\big|_\Gamma - \frac{\partial p}{\partial \mathbf{n}}\frac{\partial (g-u)}{\partial \mathbf{n}}\Big] \\
&\quad + dr_1 dr_2 r\frac{\partial}{\partial \mathbf{e}_r}\Big[(hg+h_0)\big|_\Gamma - \frac{\partial p}{\partial \mathbf{n}}\cdot\frac{\partial (g-u)}{\partial \mathbf{n}}\Big] \\
&\quad + r\,dr_1\Big[\frac{\partial p}{\partial \mathbf{n}}\cdot\frac{\partial du[dr_2]}{\partial \mathbf{n}} - \frac{\partial dp[dr_2]}{\partial \mathbf{n}}\cdot\frac{\partial (g-u)}{\partial \mathbf{n}}\Big]d\phi,
\end{aligned}$$

where the local shape derivatives $du[dr_2]$ and $dp[dr_2]$ solve

$$\begin{aligned} \Delta du[dr_2] = 0 \text{ in } \Omega, &\qquad du[dr_2] = +dr_2\,\langle \mathbf{e}_r, \mathbf{n}\rangle\,\big[\tfrac{\partial (g-u)}{\partial \mathbf{n}}\big] \text{ on } \Gamma, \\ \Delta dp[dr_2] = 0 \text{ in } \Omega, &\qquad dp[dr_2] = -dr_2\,\langle \mathbf{e}_r, \mathbf{n}\rangle\,\tfrac{\partial p}{\partial \mathbf{n}} \qquad \text{ on } \Gamma. \end{aligned}$$

BIE formulation

Consider u satisfying the boundary value problem

$$\begin{aligned} \Delta u &= f \qquad \text{in } \Omega, \\ u &= g \qquad \text{on } \Gamma. \end{aligned}$$

Let $u = v + N_f$, where N_f is a Newton potential with $\Delta N_f = f$ and v denotes the harmonical function with

$$\begin{aligned} \Delta v &= 0 \qquad\qquad \text{in } \Omega, \\ v &= g - N_f \qquad \text{on } \Gamma. \end{aligned}$$

$$\boxed{\frac{\partial u}{\partial \mathbf{n}} = \frac{\partial v}{\partial \mathbf{n}} + \frac{\partial N_f}{\partial \mathbf{n}} \quad \text{with} \quad \mathcal{V}\frac{\partial v}{\partial \mathbf{n}} = \Big(\frac{1}{2} + \mathcal{K}\Big)v,}$$

where
$$\begin{aligned} (\mathcal{V}g)(\mathbf{x}) &:= -\frac{1}{2\pi}\int_\Gamma \log\|\mathbf{x}-\mathbf{y}\| g(\mathbf{y})d\sigma_{\mathbf{y}}, \\ (\mathcal{K}g)(\mathbf{x}) &:= -\frac{1}{2\pi}\int_\Gamma \frac{\langle \mathbf{y}-\mathbf{x}, \mathbf{n}_{\mathbf{y}}\rangle}{\|\mathbf{x}-\mathbf{y}\|^2} g(\mathbf{y})d\sigma_{\mathbf{y}}. \end{aligned}$$

$$\boxed{\begin{aligned} \frac{\partial^2 u}{\partial \mathbf{t}^2} &= +\frac{\partial^2 g}{\partial \mathbf{t}^2} - \kappa\left(\frac{\partial v}{\partial \mathbf{n}} - \frac{\partial (g - N_f)}{\partial \mathbf{n}}\right) \\ \frac{\partial^2 u}{\partial \mathbf{n}^2} &= -\frac{\partial^2 g}{\partial \mathbf{t}^2} + \kappa\left(\frac{\partial v}{\partial \mathbf{n}} - \frac{\partial (g - N_f)}{\partial \mathbf{n}}\right) + f \end{aligned}}$$

$$\boxed{\begin{aligned} &\frac{\partial^2 u}{\partial \mathbf{n}\partial \mathbf{t}} = \frac{\partial^2 v}{\partial \mathbf{n}\partial \mathbf{t}} + \frac{\partial^2 N_f}{\partial \mathbf{n}\partial \mathbf{t}} \quad \text{with} \\ &\mathcal{V}\frac{\partial^2 v}{\partial \mathbf{n}\partial \mathbf{t}} = \left(\frac{1}{2} + \mathcal{K}\right)\frac{\partial (g - N_f)}{\partial \mathbf{t}} \\ &\qquad + \left[\frac{1}{2} + \mathcal{K}, \frac{\partial}{\partial \mathbf{t}}\right](g - N_f) - \left[\mathcal{V}, \frac{\partial}{\partial \mathbf{t}}\right]\frac{\partial v}{\partial \mathbf{n}} \end{aligned}}$$

Approximation of the domain

Starshaped domains can be identified with infinite Fourier series:

$$\Omega \iff r(\phi) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos n\phi + a_{-n} \sin n\phi$$

Replace the infinite shape optimization problem by a finite one:

$$\Omega_{N_r} \iff r(\phi) = \frac{a_0}{2} + \sum_{n=1}^{N_r} a_n \cos n\phi + a_{-n} \sin n\phi$$

$\rightsquigarrow$ exponentially convergent approximation of the optimal domain

For a shape functional $J(\Omega_{N_r})$ the shape gradient is given by

$$\nabla J(\Omega_{N_r}) = \begin{bmatrix} \nabla J(\Omega_{N_r})[\cos N_r\phi] \\ \nabla J(\Omega_{N_r})[\cos(N_r-1)\phi] \\ \vdots \\ \nabla J(\Omega_{N_r})[\sin N_r\phi] \end{bmatrix}$$

and its hessian by

$$\nabla^2 J(\Omega_{N_r}) = \begin{bmatrix} \nabla^2 J(\Omega_{N_r})[\cos N_r\phi, \cos N_r\phi] & \cdots & \nabla^2 J(\Omega_{N_r})[\cos N_r\phi, \sin N_r\phi] \\ \nabla^2 J(\Omega_{N_r})[\cos(N_r-1)\phi, \cos N_r\phi] & \cdots & \nabla^2 J(\Omega_{N_r})[\cos(N_r-1)\phi, \sin N_r\phi] \\ \vdots & & \vdots \\ \nabla^2 J(\Omega_{N_r})[\sin N_r\phi, \cos N_r\phi] & \cdots & \nabla^2 J(\Omega_{N_r})[\sin N_r\phi, \sin N_r\phi] \end{bmatrix}$$

Optimization algorithms

Augmented Lagrangian (equality constraints):

$$L_\alpha(\Omega, \boldsymbol{\lambda}) = J(\Omega) + \boldsymbol{\lambda}^T \begin{bmatrix} C_1(\Omega)-c_1 \\ \vdots \\ C_m(\Omega)-c_m \\ C_{m+1}(\Gamma)-c_{m+1} \\ \vdots \\ C_n(\Gamma)-c_n \end{bmatrix} + \frac{\alpha}{2} \left\| \begin{bmatrix} C_1(\Omega)-c_1 \\ \vdots \\ C_m(\Omega)-c_m \\ C_{m+1}(\Gamma)-c_{m+1} \\ \vdots \\ C_n(\Gamma)-c_n \end{bmatrix} \right\|^2$$

with update rule

$$\boldsymbol{\lambda}^{(k)} = \boldsymbol{\lambda}^{(k-1)} + \alpha \begin{bmatrix} C_1(\Omega)-c_1 \\ \vdots \\ C_m(\Omega)-c_m \\ C_{m+1}(\Gamma)-c_{m+1} \\ \vdots \\ C_n(\Gamma)-c_n \end{bmatrix} \quad \longrightarrow \quad \boldsymbol{\lambda}^\star$$

and $\alpha \geq 0$ fixed but appropriately chosen.

- starting point: choose initial guesses $\Omega^{(0)}$, $\boldsymbol{\lambda}^{(0)}$
- outer iteration: for $k = 1, 2, \ldots$ solve

  $$\Omega^{(k)} = \arg\min L_\alpha(\Omega, \boldsymbol{\lambda}^{(k-1)})$$

  with initial guess $\Omega^{(k,0)} := \Omega^{(k-1)}$ and update $\boldsymbol{\lambda}^{(k)}$
- inner iteration: for $l = 1, 2, \ldots$ update

  $$\Omega^{(k,l)} = \Omega^{(k,l-1)} + h\delta\Omega$$

  with descent direction $\delta\Omega$ from gradient/hessian information and h from line search
- line search: set $h := 1$; while

  $$L_\alpha(\Omega^{(k,l-1)} + h\delta\Omega, \boldsymbol{\lambda}^{(k)}) \geq L_\alpha(\Omega^{(k,l-1)}, \boldsymbol{\lambda}^{(k)})$$

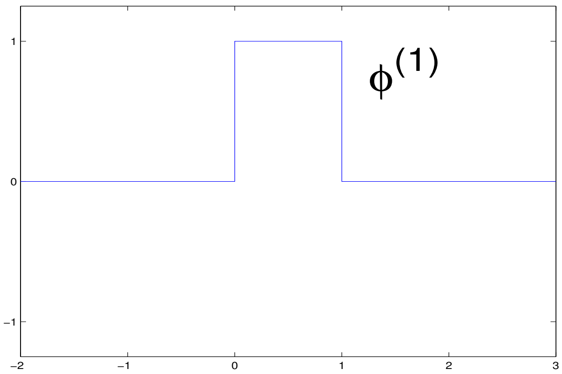
  reduce h.

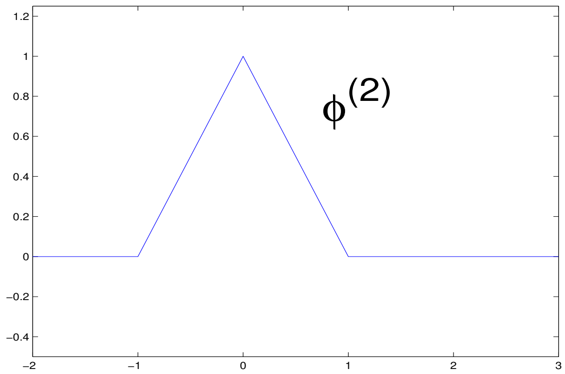
BEM formulation

Boundary integral equation: $\mathcal{V}\frac{\partial v}{\partial \mathbf{n}} = \Big(\frac{1}{2} + \mathcal{K}\Big)v$

with
$$(\mathcal{V}g)(\mathbf{x}) := -\frac{1}{2\pi}\int_\Gamma \log\|\mathbf{x}-\mathbf{y}\| g(\mathbf{y})d\sigma_{\mathbf{y}}$$
$$(\mathcal{K}g)(\mathbf{x}) := -\frac{1}{2\pi}\int_\Gamma \frac{\langle \mathbf{y}-\mathbf{x}, \mathbf{n}_{\mathbf{y}}\rangle}{\|\mathbf{x}-\mathbf{y}\|^2} g(\mathbf{y})d\sigma_{\mathbf{y}}$$

Discretization via $N_\Gamma = 2^L$ piecewise constants $\Phi_L^{(1)}$ and piecewise linears $\Phi_L^{(2)}$, respectively:





Ansatz: $\frac{\partial v}{\partial \mathbf{n}} = \boldsymbol{\sigma}_L^\phi \Phi_L^{(1)}$

Galerkin system:

$$\boxed{\mathbf{V}_L^\phi \boldsymbol{\sigma}_L^\phi = (\tfrac{1}{2}\mathbf{B}_L^\phi + \mathbf{K}_L^\phi)(\mathbf{G}_L^\phi)^{-1}\mathbf{v}_L^\phi}$$

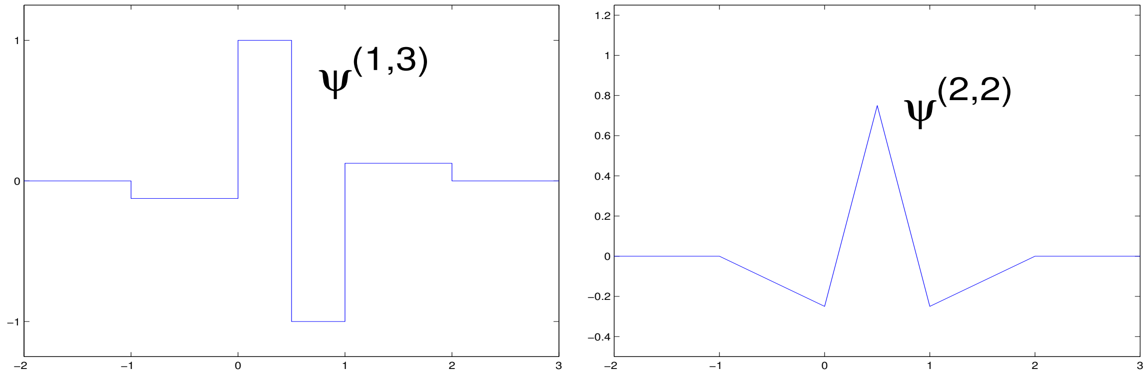
where
$$\mathbf{V}_L^\phi = (\mathcal{V}\Phi_L^{(1)}, \Phi_L^{(1)})_{L^2(\Gamma)}, \quad \mathbf{B}_L^\phi = \big(\Phi_L^{(2)}, \Phi_L^{(1)}\big)_{L^2(\Gamma)},$$
$$\mathbf{K}_L^\phi = \big(\mathcal{K}\Phi_L^{(2)}, \Phi_L^{(1)}\big)_{L^2(\Gamma)}, \quad \mathbf{G}_L^\phi = \big(\Phi_L^{(2)}, \Phi_L^{(2)}\big)_{L^2(\Gamma)},$$
$$\mathbf{v}_L^\phi = (v, \Phi_L^{(2)})_{L^2(\Gamma)}.$$

Wavelet Galerkin scheme

Single-scale Galerkin scheme:

$$\left.\begin{array}{l}\bullet \text{ densely populated system matrices} \\ \bullet\ \operatorname{cond}(\mathbf{V}_L^\phi) \sim N_\Gamma\end{array}\right\} \to \text{ complextity } \geq \mathcal{O}(N_\Gamma^2)$$

Change of bases: use piecewise constants wavelets $\Psi_L^{(1,3)}$ and piecewise linear wavelets $\Phi_L^{(2,2)}$:



Ansatz: $\dfrac{\partial v}{\partial \mathbf{n}} = \boldsymbol{\sigma}_L^\psi \Psi_L^{(1)}$

Galerkin system:

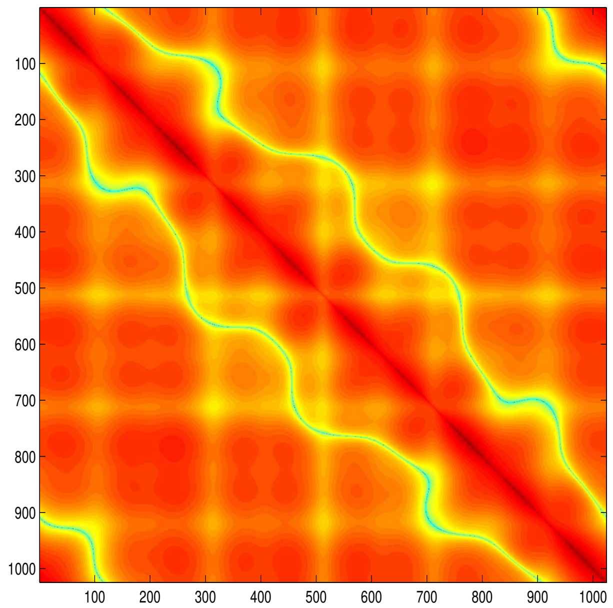
$$\boxed{\mathbf{V}_L^\psi \boldsymbol{\sigma}_L^\psi = (\tfrac{1}{2}\mathbf{B}_L^\psi + \mathbf{K}_L^\psi)(\mathbf{G}_L^\psi)^{-1}\mathbf{v}_L^\psi}$$

where
$$\begin{aligned}\mathbf{V}_L^\psi &= (\mathcal{V}\Psi_L^{(1)}, \Psi_L^{(1)})_{L^2(\Gamma)}, & \mathbf{B}_L^\psi &= \big(\Psi_L^{(2)}, \Psi_L^{(1)}\big)_{L^2(\Gamma)}, \\ \mathbf{K}_L^\psi &= \big(\mathcal{K}\Psi_L^{(2)}, \Psi_L^{(1)}\big)_{L^2(\Gamma)}, & \mathbf{G}_L^\psi &= \big(\Psi_L^{(2)}, \Psi_L^{(2)}\big)_{L^2(\Gamma)}, \\ \mathbf{v}_L^\psi &= (v, \Psi_L^{(2)})_{L^2(\Gamma)}.\end{aligned}$$

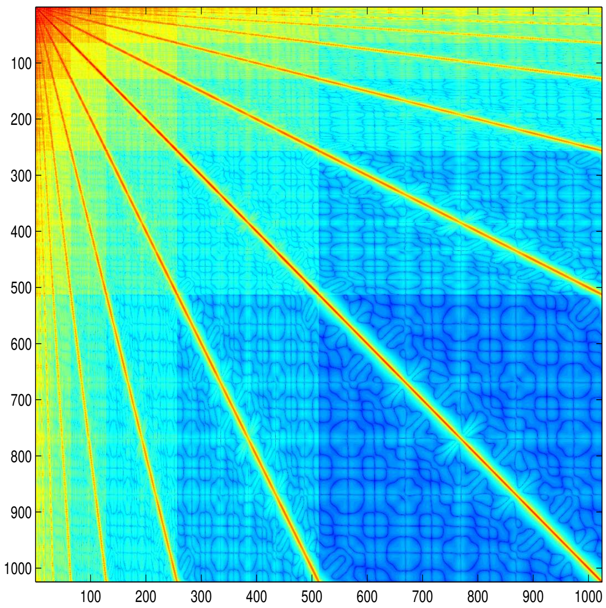
Wavelet Galerkin scheme (Beylkin/Coifman/Rokhlin, 1991
Dahmen/Prößdorf/Schneider, 1993++):

$$\left.\begin{array}{l}\bullet \text{ sparse matrix approximation} \\ \bullet\ \operatorname{diag}(\mathbf{V}_L^\psi) \text{ preconditioner of } \mathbf{V}_L^\psi\end{array}\right\} \to \text{ complextity } \mathcal{O}(N_\Gamma)$$

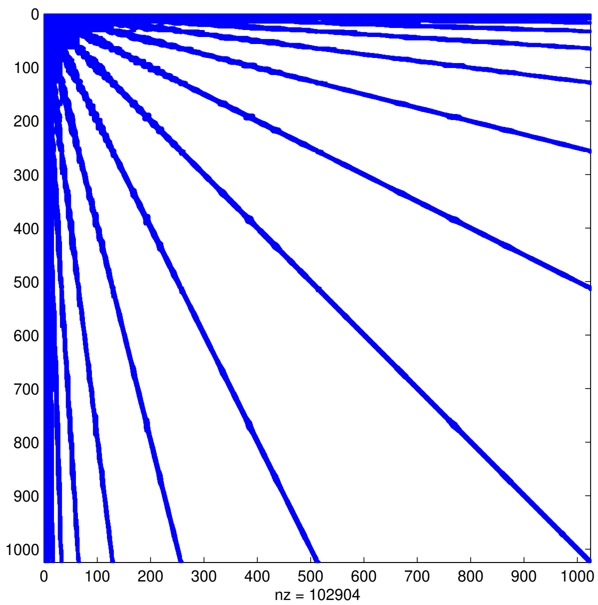
Matrix compression



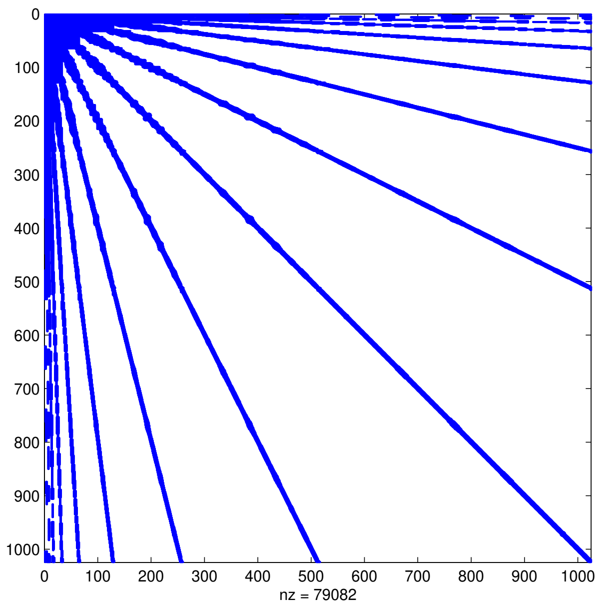
single-scale basis



wavelet basis



1st compression



1st + 2nd compression

Requirements for a Newton step

Ansatz:

$$\frac{\partial v}{\partial \mathbf{n}} = \boldsymbol{\sigma}_L^{\psi}\Psi_L^{(1)}, \qquad \frac{\partial^2 v}{\partial \mathbf{n}\partial \mathbf{t}} = \widetilde{\boldsymbol{\sigma}}_L^{\psi}\Psi_L^{(1)}$$

System matrices:

$$\begin{aligned}
\mathbf{V}_L^{\psi} &= (\mathcal{V}\Psi_L^{(1)}, \Psi_L^{(1)})_{L^2(\Gamma)}, & \widetilde{\mathbf{V}}_L^{\psi} &= \big(\big[\mathcal{V}, \tfrac{\partial}{\partial \mathbf{t}}\big]\Psi_L^{(1)}, \Psi_L^{(1)}\big)_{L^2(\Gamma)} \\
\mathbf{K}_L^{\psi} &= (\mathcal{K}\Psi_L^{(2)}, \Psi_L^{(1)})_{L^2(\Gamma)}, & \widetilde{\mathbf{K}}_L^{\psi} &= \big(\big[\mathcal{K}, \tfrac{\partial}{\partial \mathbf{t}}\big]\Psi_L^{(2)}, \Psi_L^{(1)}\big)_{L^2(\Gamma)}
\end{aligned}$$

Mass matrices:

$$\mathbf{B}_L^{\psi} = \big(\Psi_L^{(2)}, \Psi_L^{(1)}\big)_{L^2(\Gamma)}, \qquad \mathbf{G}_L^{\psi} = \big(\Psi_L^{(2)}, \Psi_L^{(2)}\big)_{L^2(\Gamma)}$$

Data vectors:

$$\mathbf{v}_L^{\psi} = (v, \Psi_L^{(2)})_{L^2(\Gamma)}, \qquad \widetilde{\mathbf{v}}_L^{\psi} = \big(\tfrac{\partial v}{\partial \mathbf{t}}, \Psi_L^{(2)}\big)_{L^2(\Gamma)}$$

- Solve the Galerkin system

$$\boxed{\mathbf{V}_L^{\psi}\boldsymbol{\sigma}_L^{\psi} = (\tfrac{1}{2}\mathbf{B}_L^{\psi} + \mathbf{K}_L^{\psi})(\mathbf{G}_L^{\psi})^{-1}\mathbf{v}_L^{\psi}}$$

to compute the normal derivative of the state function and the N_r local shape derivatives

- Solve the Galerkin system

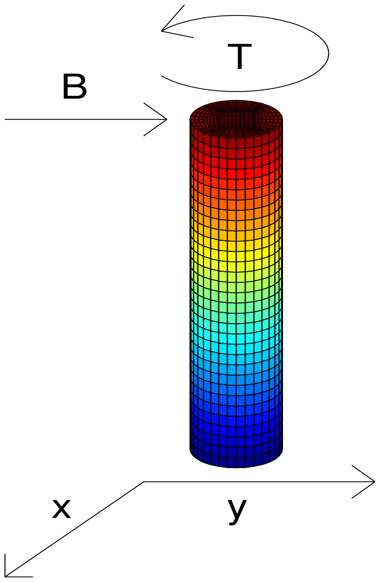
$$\boxed{\mathbf{V}_L^{\psi}\widetilde{\boldsymbol{\sigma}}_L^{\psi} = \big(\tfrac{1}{2}\mathbf{B}_L^{\psi} + \mathbf{K}_L^{\psi}\big)\big(\mathbf{G}_L^{\psi}\big)^{-1}\widetilde{\mathbf{v}}_L^{\psi} + \widetilde{\mathbf{K}}_L^{\psi}\big(\mathbf{G}_L^{\psi}\big)^{-1}\mathbf{v}_L^{\psi} + \widetilde{\mathbf{V}}_L^{\psi}\mathbf{v}_L^{\psi}}$$

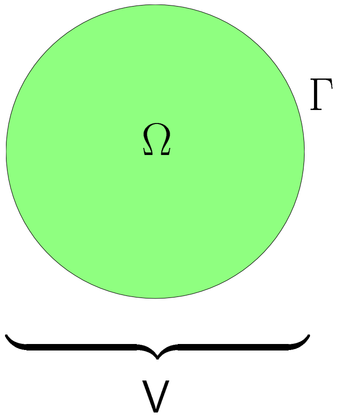
to compute the second order derivatives of the state function

A shape problem from linear elasticity

Given: Elastic cylindrical bar (homogeneous, isotropic) with cross section $\Omega \in \mathbb{R}^2$

Problem: Maximize torsional rigidity under constraints on volume and bending rigidity (with respect to y-direction)





$$\boxed{\begin{array}{c} J(\Omega) = -T(\Omega) \to \min \quad \text{subject to} \\ V(\Omega) \leq V_0, \quad B(\Omega) \geq B_0, \quad \mathbf{x}_b(\Omega) = \mathbf{0} \end{array}}$$

Bending rigidity: $$B(\Omega) = \int_\Omega y^2 \, d\mathbf{x}$$

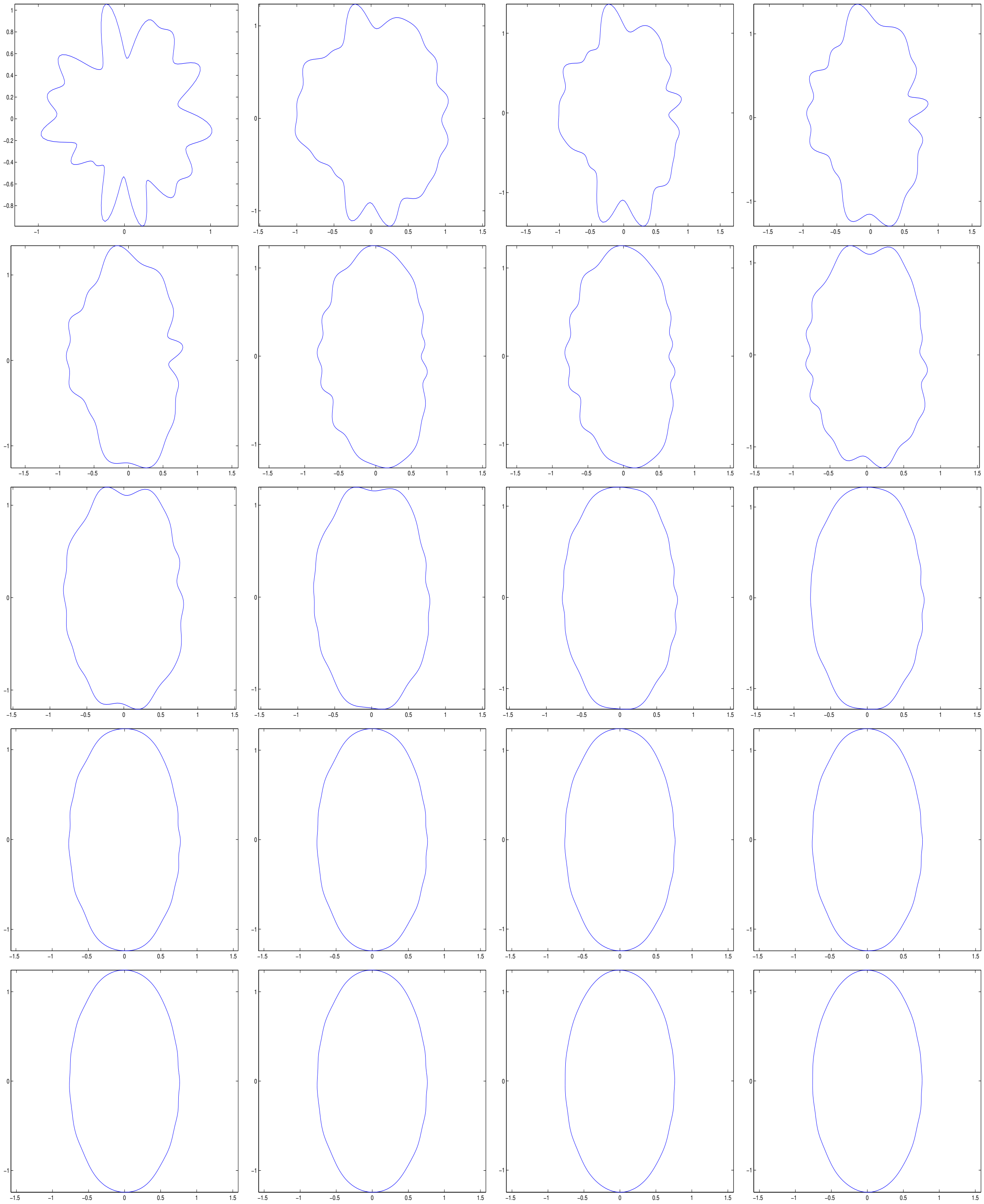
Volume: $$V(\Omega) = \int_\Omega d\mathbf{x}$$

Torsional rigidity: $$T(\Omega) = 2 \int_\Omega u(\mathbf{x}) \, d\mathbf{x}$$

where $$\boxed{\begin{array}{rl} -\triangle u = 2 & \text{in } \Omega \\ u = 0 & \text{on } \Gamma \end{array}}$$

Barycentre: $$\mathbf{x}_b(\Omega) = \left[\int_\Omega x \, d\mathbf{x}, \int_\Omega y \, d\mathbf{x} \right]^T$$

Numerical results



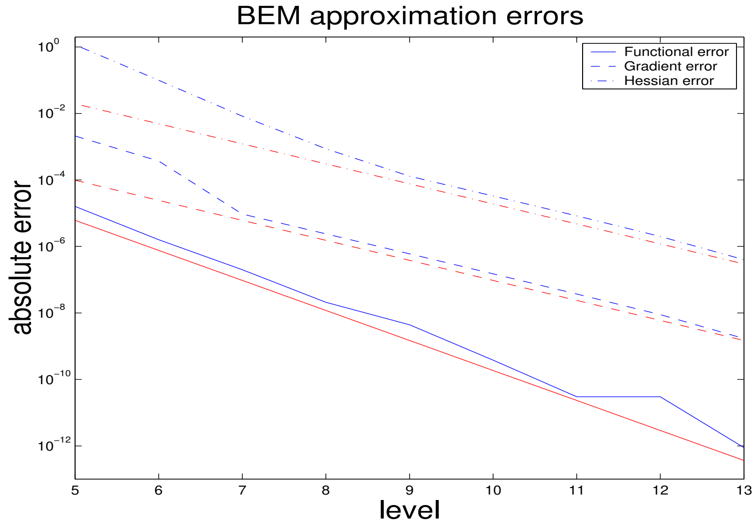
Quasi-Newton method for the penalty functional,
$N_\Gamma = 256$, $N_r = 16$, cpu-time $= 45$ sec.,
l^2-error of the Fourier-coefficients: 0.031737.

Numerical Results

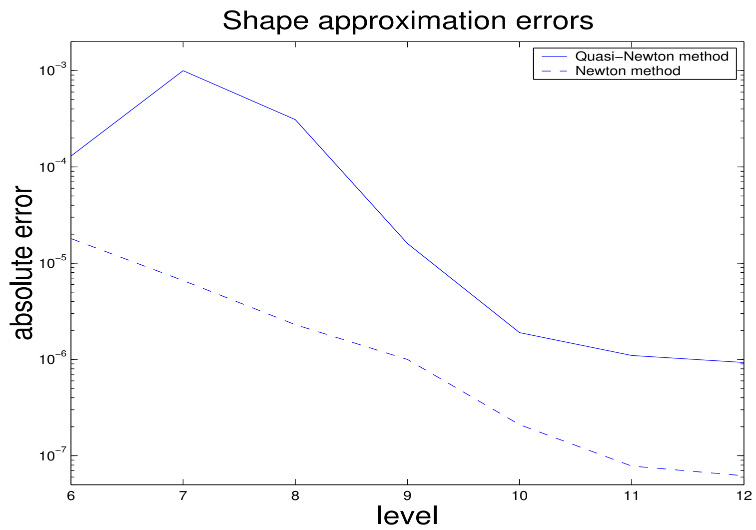
- Approximation errors of the functional data:

$$|J(\Omega) - \overline{J(\Omega)}| = \mathcal{O}(h_L^3),$$

$$\|\nabla J(\Omega) - \overline{\nabla J(\Omega)}\| = \mathcal{O}(h_L^2), \quad \|\nabla^2 J(\Omega) - \overline{\nabla^2 J(\Omega)}\| = \mathcal{O}(h_L^2)$$



- Shape approximation errors: Newton vs. quasi-Newton method

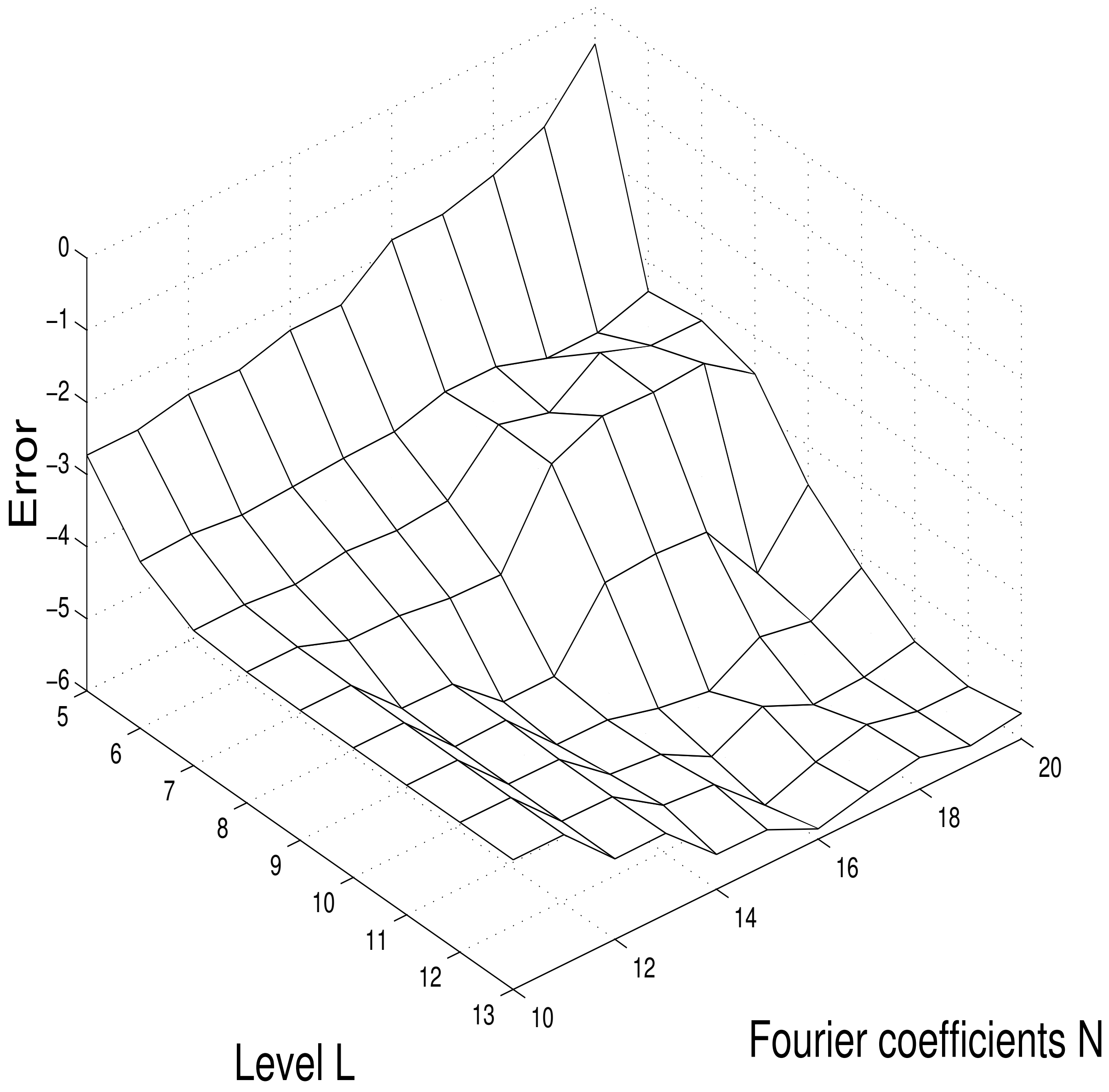


- Performance:

Quasi-Newton method		Newton method	
PDE-solutions	cpu-time	PDE-solutions	cpu-time
54.0	70	22.6	47.7

Numbers of PDE-solutions and cpu-times to achieve an error of approximation less than 1e-3

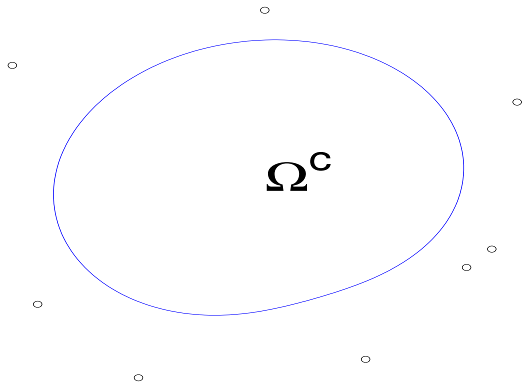
Asymptotic behaviour



The l^2-error of the Fourier coefficients vs. the number of unknowns N_r and L.

Electromagnetic shaping 2D

Given: cylindrical vertical column of liquid metal with cross section $\Omega^c \in \mathbb{R}^2$ and M vertical circular conductors with radii ε



Shape problem: (Pierre/Henrot/Roche/Sokolowski/Novruzi)

$$J(\Omega) = -\int_\Omega \|\nabla u\|^2 d\mathbf{x} + A\int_\Gamma 1 d\sigma \to \min,$$

where
$$\begin{aligned} -\Delta u = j &:= \textstyle\sum_{i=1}^M \frac{\alpha_i I}{\pi\varepsilon^2}\chi_{B_\varepsilon(\mathbf{x}_i)} && \text{in } \Omega,\\ u &= 0 && \text{on } \Gamma,\\ u &= \mathcal{O}(1) && \text{as } \|\mathbf{x}\| \to \infty,\\ \nabla u &= \mathcal{O}(\|\mathbf{x}\|^{-2}) && \text{as } \|\mathbf{x}\| \to \infty. \end{aligned}$$

subject to $|\Omega^c| = V_0$.

Integration by parts yields

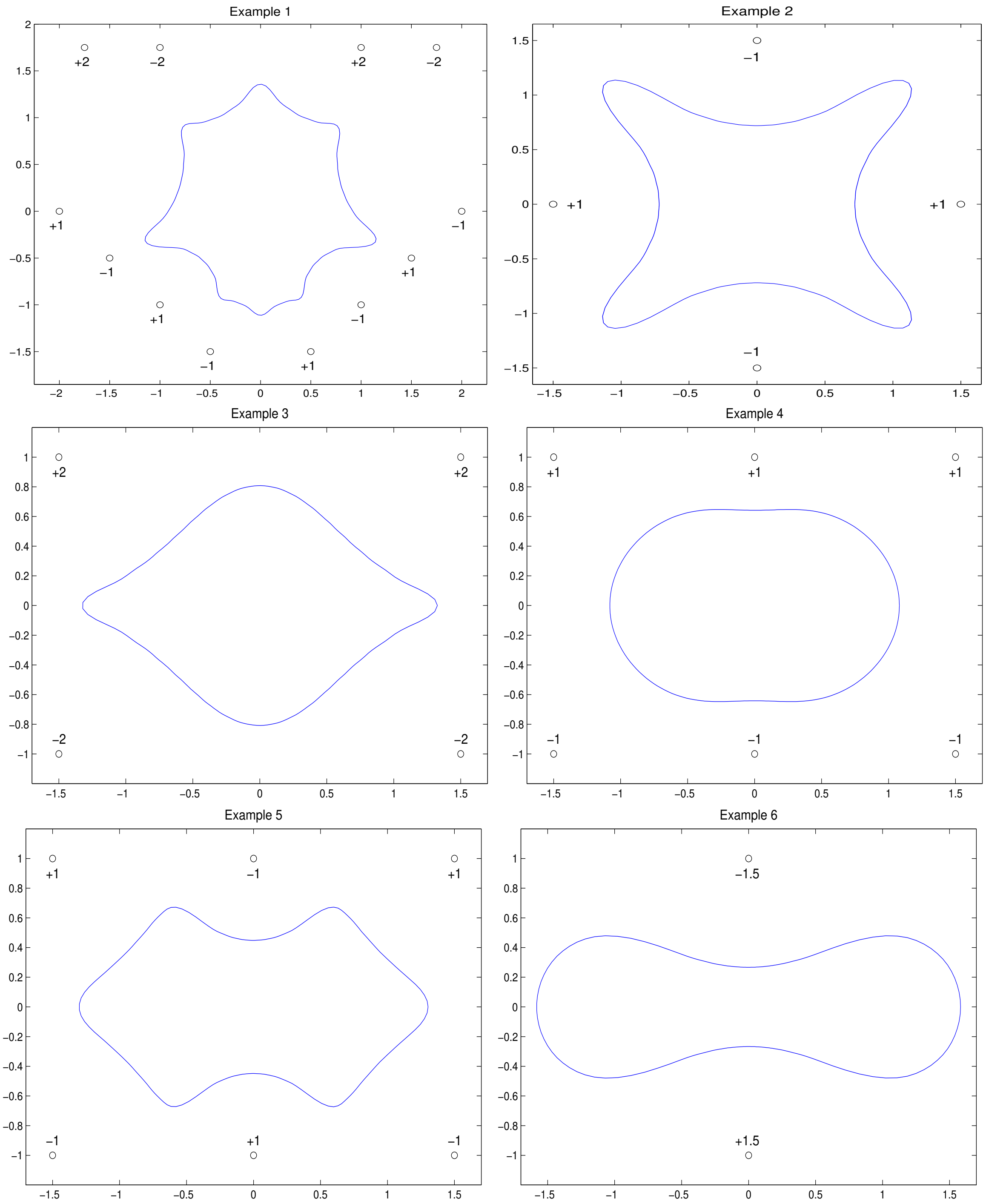
$$\begin{aligned} J(\Omega) &= -\int_\Omega j(\mathbf{x})u(\mathbf{x})d\mathbf{x} + A\int_\Gamma 1 d\sigma \\ &= C(\varepsilon) - \int_\Gamma N_j(\mathbf{x})\frac{\partial u}{\partial \mathbf{n}}(\mathbf{x})\, d\sigma + A\int_\Gamma 1 d\sigma \end{aligned}$$

with $N_j = \sum_{i=1}^M N_{ij}$ given by

$$N_{ij}(\mathbf{x}) = \begin{cases} \frac{\alpha_i}{4\pi}\left[1 - \frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{\varepsilon^2} - 2\log\varepsilon\right], & \mathbf{x} \in B_\varepsilon(\mathbf{x}_i),\\ -\frac{\alpha_i}{2\pi}\log\|\mathbf{x}-\mathbf{x}_i\|, & \mathbf{x} \notin B_\varepsilon(\mathbf{x}_i). \end{cases}$$

Electromagnetic shaping 2D

Example	iterations	$\|\nabla L_c^{(n)}(\Omega,\Omega^{(n)})\|$	λ	cpu-time
1	13	1.593e-08	3.010e-02	89 sec.
2	14	1.483e-07	7.573e-02	89 sec.
3	15	4.544e-08	3.541e-01	102 sec.
4	15	2.117e-07	9.738e-01	101 sec.
5	21	4.092e-07	8.856e-02	162 sec.
6	24	3.257e-07	2.191e-01	199 sec.



Electromagnetic shaping 3D

Given: bubble $\Omega \subseteq \mathbb{R}^3$ of liquid metal levitating in a magnetic field generatet by conductors

Problem: compute the free surface $\Gamma := \partial\Omega$ of the bubble of liquid metal

Formulation as shape optimization problem: (Pierre/Roche)

$$J(\Omega) = A \int_\Gamma 1 d\sigma + B \int_\Omega x_3 d\mathbf{x} - \int_{\Omega^c} \|\mathbf{B}\|^2 d\mathbf{x} \to \min,$$

where $\mathbf{B}$ satisfies the magnetostatic Maxwell equations

$$\begin{aligned} \nabla \times \mathbf{B} &= \mu_0 \mathbf{j} && \text{in } \Omega^c, \\ \nabla \mathbf{B} &= 0 && \text{in } \Omega^c, \\ \langle \mathbf{B}, \mathbf{n} \rangle &= 0 && \text{on } \Gamma, \\ \mathbf{B} &= \mathcal{O}(\|\mathbf{x}\|^{-2}) && \text{as } \|\mathbf{x}\| \to \infty, \end{aligned}$$

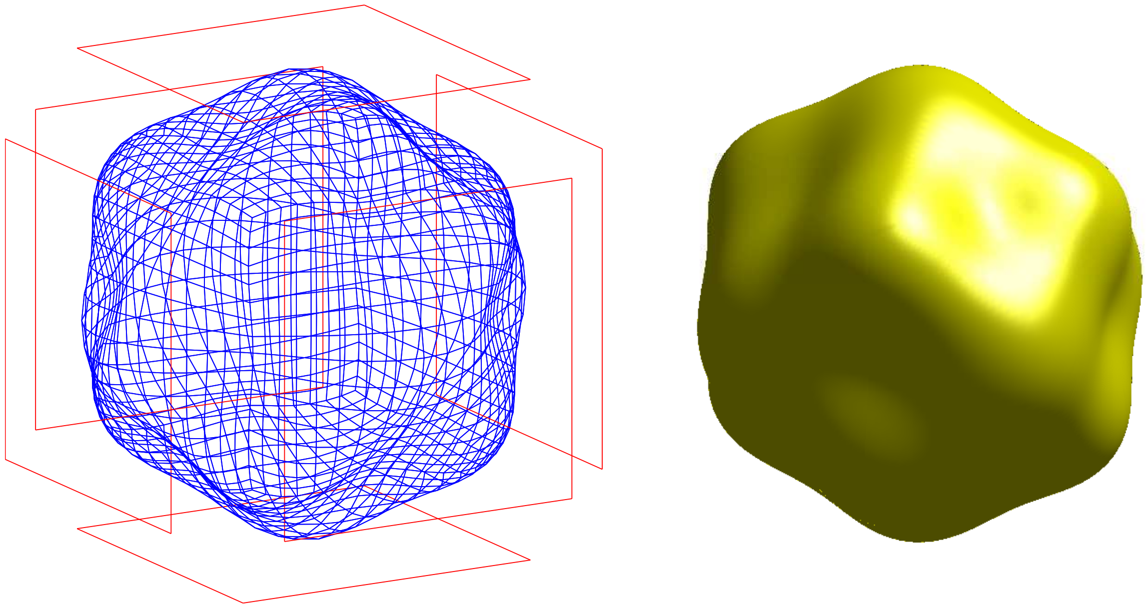
subject to $|\Omega| = V_0$.

Shape gradient:

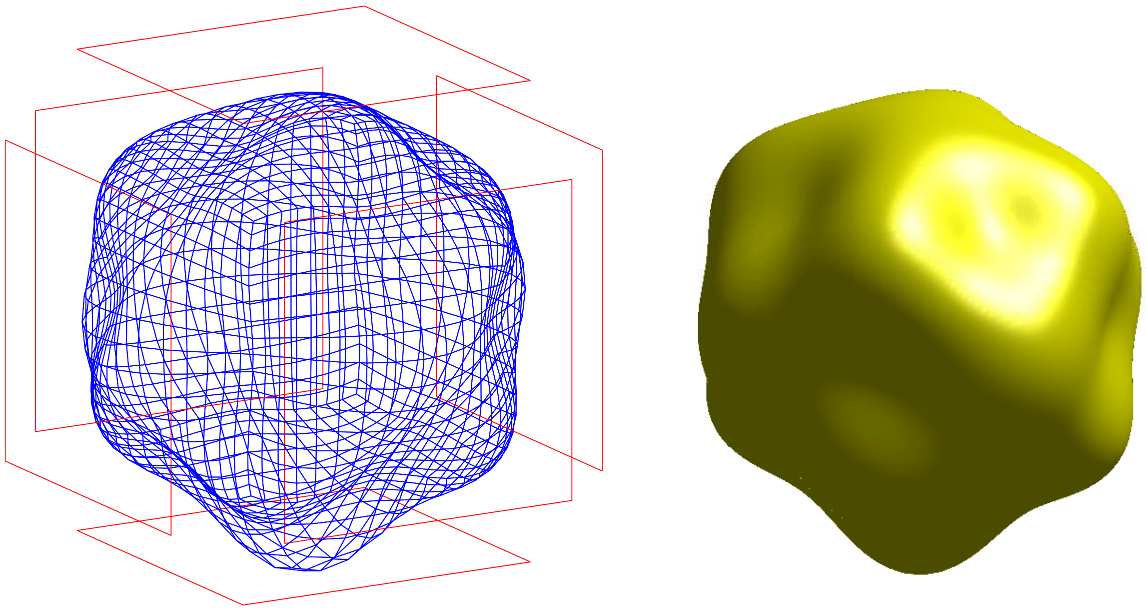
$$\nabla J(\Omega)[\mathbf{V}] = \int_\Gamma \langle \mathbf{V}, \mathbf{n} \rangle \left\{ 2A\mathcal{H} + Bx_3 + \|\mathbf{B}\|^2 \right\} d\sigma$$

Electromagnetic shaping 3D

- Computation without gravitational forces ($B = 0$):



- Computation with gravitational forces ($B \neq 0$):

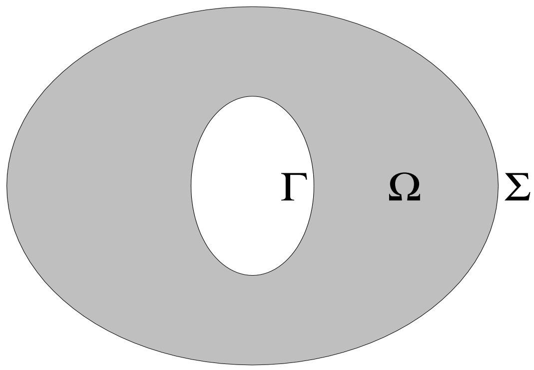


Identification of voids or inclusions

Given: Dirichlet data f and Neumann data g on Σ

Problem: seek Γ such that

$$\begin{aligned} \Delta u &= 0 && \text{in } \Omega, \\ u &= 0 && \text{on } \Gamma, \\ u &= f && \text{on } \Sigma, \\ \frac{\partial u}{\partial \mathbf{n}} &= g && \text{on } \Sigma. \end{aligned}$$



Shape problem:

$$J(\Omega) = \int_\Omega \|\nabla(v - w)\|^2 d\mathbf{x} \to \min,$$

where

$$\begin{aligned} \Delta v &= 0 & \Delta w &= 0 && \text{in } \Omega, \\ v &= 0 & w &= 0 && \text{on } \Gamma, \\ \frac{\partial v}{\partial \mathbf{n}} &= g & w &= f && \text{on } \Sigma. \end{aligned}$$

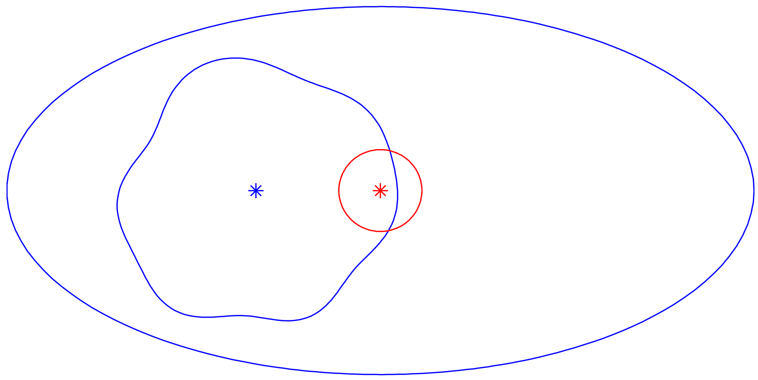
Shape gradient (Roche/Sokolowski):

$$\boxed{\nabla J(\Omega)[\mathbf{V}] = \int_\Gamma \langle \mathbf{V}, \mathbf{n} \rangle \left[\left(\frac{\partial v}{\partial \mathbf{n}} \right)^2 - \left(\frac{\partial w}{\partial \mathbf{n}} \right)^2 \right] d\sigma}$$

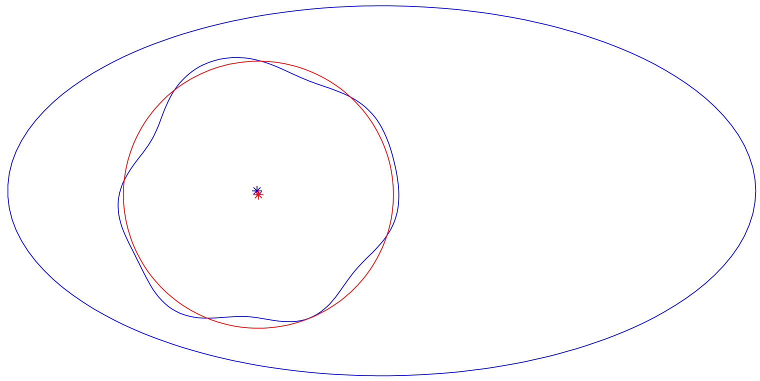
$\rightsquigarrow$ extremely ill posed problem

Identification of voids or inclusions

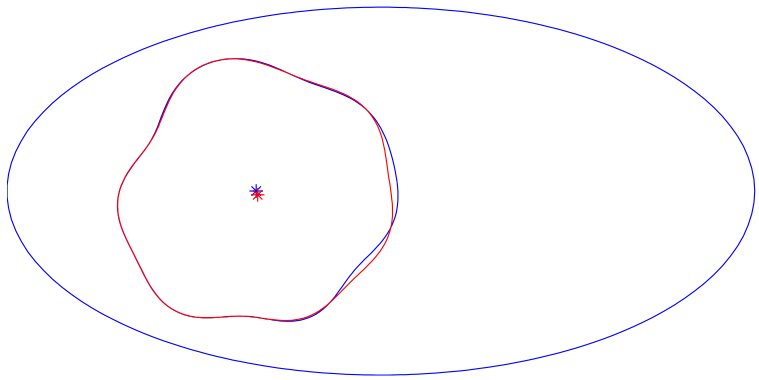
- Choose circle as initial guess:



- Shift and scale (20 quasi-Newton steps / 64 BEM-unknowns):

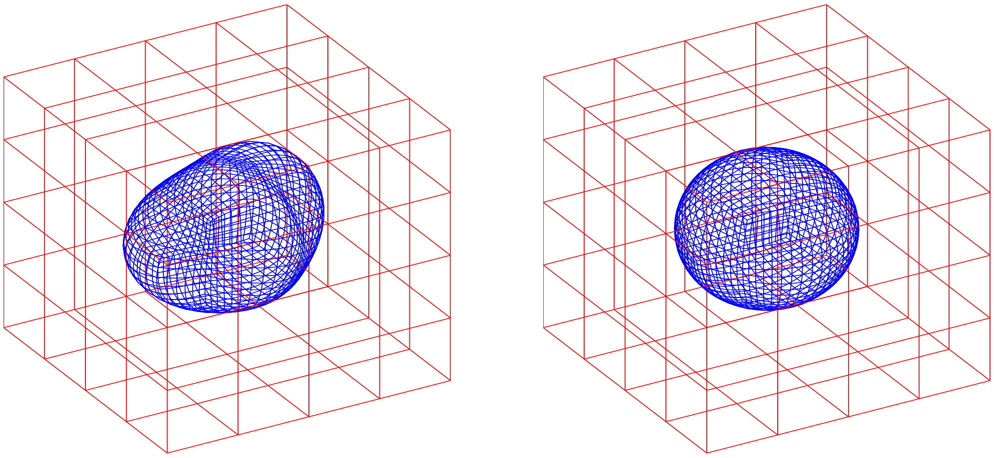


- Compute boundary (30 quasi-Newton steps / 512 BEM-unknowns):

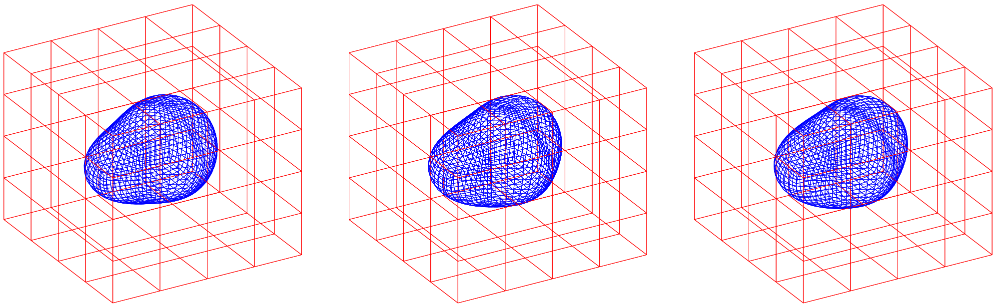


Identification of voids or inclusions

- Given problem and initial guess:



- After 10/20/30 quasi-Newton steps using 3468 BEM-unknowns:



- Functional error (blue) and shape error (red):