Shape and Topology Optimization subject to 3D Nonlinear Magnetostatics - Part II: Numerics

Peter Gangl[1], Kevin Sturm[2] and Joachim Schöberl[2]
[1]Institute of Applied Mathematics, Graz University of Technology, Austria
[2] Institute for Analysis and Scientific Computing, TU Wien, Austria

AANMPDE Strobl 2019

July 4, 2019

TU Graz

Outline

Outline

Design Optimization of a 3D Electric Motor

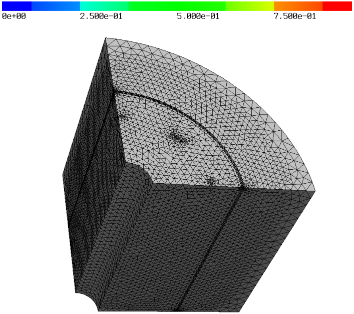
Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot n - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in V : \int_{\mathrm{D}} \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v + \epsilon u \cdot v \mathrm{d}x = \langle F, v \rangle \quad \text{for all } v \in V.$$

Here,

- u magnetic vector potential ($B = \mathrm{curl} u$)
- $\nu_\Omega(x, s) := \chi_\Omega(x)\nu_2(s) + \chi_{D \setminus \overline{\Omega}}(x)\nu_1(s)$,
- $\langle F, v \rangle := \int_{\Omega_c} J_i \cdot v \, \mathrm{d}x + \int_{\Omega_{mag}} \nu_{mag} M \cdot \mathrm{curl} v \mathrm{d}x$
- $V := H_0(\mathrm{D}, \mathrm{curl})$
- $\epsilon > 0$ regularization parameter
- $\nu_1, \nu_2 : \mathbb{R}_0^+ \to \mathbb{R}^+$ with $s \mapsto \nu_i(s)s$, $i = 1, 2$ is Lipschitz continuous and strongly monotone



Design Optimization of a 3D Electric Motor

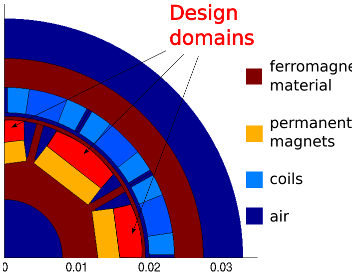
Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot n - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in V : \int_D \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v + \epsilon u \cdot v \mathrm{d}x = \langle F, v \rangle \quad \text{for all } v \in V.$$

Here,

- u magnetic vector potential ($B = \mathrm{curl} u$)
- $\nu_\Omega(x, s) := \chi_\Omega(x)\nu_2(s) + \chi_{D\setminus\overline{\Omega}}(x)\nu_1(s)$,
- $\langle F, v \rangle := \int_{\Omega_c} J_i \cdot v \, \mathrm{d}x + \int_{\Omega_{mag}} \nu_{mag} M \cdot \mathrm{curl} v \mathrm{d}x$
- $V := H_0(\mathrm{D}, \mathrm{curl})$
- $\epsilon > 0$ regularization parameter
- $\nu_1, \nu_2 : \mathbb{R}_0^+ \to \mathbb{R}^+$ with $s \mapsto \nu_i(s)s$, $i = 1, 2$ is Lipschitz continuous and strongly monotone



Design Optimization of a 3D Electric Motor

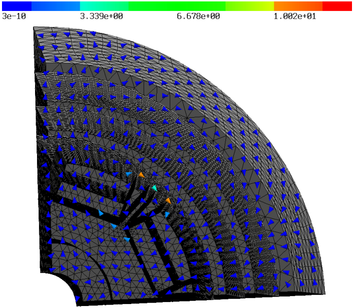
Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot n - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in V : \int_{\mathrm{D}} \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v + \epsilon u \cdot v \mathrm{d}x = \langle F, v \rangle \quad \text{for all } v \in V.$$

Here,

- u magnetic vector potential ($B = \mathrm{curl} u$)
- $\nu_\Omega(x,s) := \chi_\Omega(x)\nu_2(s) + \chi_{D \setminus \overline{\Omega}}(x)\nu_1(s)$,
- $\langle F, v \rangle := \int_{\Omega_c} J_i \cdot v \, \mathrm{d}x + \int_{\Omega_{mag}} \nu_{mag} M \cdot \mathrm{curl} v \mathrm{d}x$
- $V := H_0(\mathrm{D}, \mathrm{curl})$
- $\epsilon > 0$ regularization parameter
- $\nu_1, \nu_2 : \mathbb{R}_0^+ \to \mathbb{R}^+$ with $s \mapsto \nu_i(s)s$, $i = 1, 2$ is Lipschitz continuous and strongly monotone



Outline

A generic (shape) optimization algorithm

Minimization problem

$$\min_{\Omega \in \mathcal{A}} \mathcal{J}(\Omega)$$

where $\mathcal{J} : \mathcal{A} \to \mathbb{R}$, $\mathcal{A}$ set of shapes

Differentiability

Descent direction

Optimization step

A generic (shape) optimization algorithm

Minimization problem

$$\min_{\Omega \in \mathcal{A}} \mathcal{J}(\Omega)$$

where $\mathcal{J} : \mathcal{A} \to \mathbb{R}$, $\mathcal{A}$ set of shapes

Differentiability

$$d\mathcal{J}(\Omega; X)$$

Shape derivative at $\Omega \in \mathcal{A}$ in direction X

Descent direction

Optimization step

A generic (shape) optimization algorithm

Minimization problem

$$\min_{\Omega \in \mathcal{A}} \mathcal{J}(\Omega)$$

where $\mathcal{J} : \mathcal{A} \to \mathbb{R}$, $\mathcal{A}$ set of shapes

Differentiability

$$d\mathcal{J}(\Omega; X)$$

Shape derivative at $\Omega \in \mathcal{A}$ in direction X

Descent direction

For $\Omega^0 \in \mathcal{A}$, find X s.t. $d\mathcal{J}(\Omega^0; X) < 0$ by choosing positive definite $b(\cdot, \cdot)$ and solving

$$\text{Find } X : b(X, W) = -d\mathcal{J}(\Omega; W) \ \forall W$$

Optimization step

A generic (shape) optimization algorithm

Minimization problem

$$\min_{\Omega \in \mathcal{A}} \mathcal{J}(\Omega)$$

where $\mathcal{J} : \mathcal{A} \to \mathbb{R}$, $\mathcal{A}$ set of shapes

Differentiability

$$d\mathcal{J}(\Omega; X)$$

Shape derivative at $\Omega \in \mathcal{A}$ in direction X

Descent direction

For $\Omega^0 \in \mathcal{A}$, find X s.t. $d\mathcal{J}(\Omega^0; X) < 0$ by choosing positive definite $b(\cdot, \cdot)$ and solving

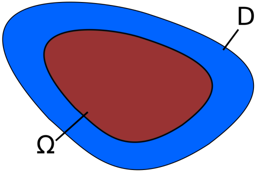
$$\text{Find } X : b(X, W) = -d\mathcal{J}(\Omega; W) \ \forall W$$

Optimization step

Choose a step size $t \in \mathbb{R}$ such that for update $\Omega^0 \to T_t^X(\Omega^0) =: \Omega^1$ it holds $\mathcal{J}(\Omega^1) < \mathcal{J}(\Omega^0)$.

Shape Derivative

The shape derivative of a shape function $\mathcal{J}$ describes the sensitivity of $\mathcal{J}$ when the shape Ω is perturbed by the action of a vector field $X \in C_0^1(D, \mathbb{R}^N)$ into $\Omega_t := \{x + tX(x) | x \in \Omega\}$.



Shape Derivative

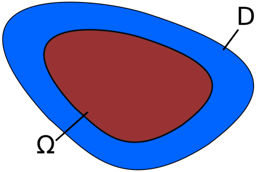
The shape derivative of a shape function $\mathcal{J}$ describes the sensitivity of $\mathcal{J}$ when the shape Ω is perturbed by the action of a vector field $X \in C_0^1(D, \mathbb{R}^N)$ into $\Omega_t := \{x + tX(x) | x \in \Omega\}$.

Definition

Shape derivative in direction $X \in C_0^1(D, \mathbb{R}^d)$ is defined by

$$d\mathcal{J}(\Omega; X) := \frac{d}{dt}\mathcal{J}(\Omega_t)|_{t=0} = \lim_{t \searrow 0} \frac{\mathcal{J}(\Omega_t) - \mathcal{J}(\Omega)}{t}$$

if $X \mapsto d\mathcal{J}(\Omega; X)$ exists and is linear and continuous on $C_0^1(D, \mathbb{R}^d)$.



Shape Derivative

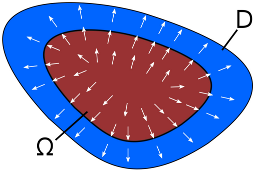
The shape derivative of a shape function $\mathcal{J}$ describes the sensitivity of $\mathcal{J}$ when the shape Ω is perturbed by the action of a vector field $X \in C_0^1(D, \mathbb{R}^N)$ into $\Omega_t := \{x + tX(x) | x \in \Omega\}$.

Definition

Shape derivative in direction $X \in C_0^1(D, \mathbb{R}^d)$ is defined by

$$d\mathcal{J}(\Omega; X) := \frac{d}{dt}\mathcal{J}(\Omega_t)|_{t=0} = \lim_{t \searrow 0} \frac{\mathcal{J}(\Omega_t) - \mathcal{J}(\Omega)}{t}$$

if $X \mapsto d\mathcal{J}(\Omega; X)$ exists and is linear and continuous on $C_0^1(D, \mathbb{R}^d)$.



Shape Derivative

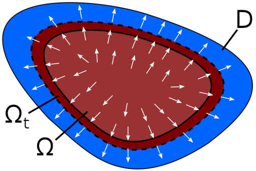
The shape derivative of a shape function $\mathcal{J}$ describes the sensitivity of $\mathcal{J}$ when the shape Ω is perturbed by the action of a vector field $X \in C_0^1(D, \mathbb{R}^N)$ into $\Omega_t := \{x + tX(x) | x \in \Omega\}$.

Definition

Shape derivative in direction $X \in C_0^1(D, \mathbb{R}^d)$ is defined by

$$d\mathcal{J}(\Omega; X) := \frac{d}{dt}\mathcal{J}(\Omega_t)|_{t=0} = \lim_{t \searrow 0} \frac{\mathcal{J}(\Omega_t) - \mathcal{J}(\Omega)}{t}$$

if $X \mapsto d\mathcal{J}(\Omega; X)$ exists and is linear and continuous on $C_0^1(D, \mathbb{R}^d)$.



Shape Derivative (Example 1)

Let $\Omega \subset D$, $f \in C^1(\overline{D})$ and

$$\mathcal{J}(\Omega) := \int_\Omega f(x)\,\mathrm{d}x.$$

Shape Derivative (Example 1)

Let $\Omega \subset D$, $f \in C^1(\overline{D})$ and

$$\mathcal{J}(\Omega) := \int_\Omega f(x)\,\mathrm{d}x.$$

Using the transformation $y = T_t(x) = x + tX(x)$, we get

$$\begin{aligned} d\mathcal{J}(\Omega; X) &= \frac{d}{dt}\mathcal{J}(\Omega_t)|_{t=0} \\ &= \frac{d}{dt}\left(\int_{\Omega_t} f(y)\,\mathrm{d}y\right)|_{t=0} \\ &= \frac{d}{dt}\left(\int_{\Omega} f(T_t(x))\,\mathrm{det}(DT_t(x))\,\mathrm{d}x\right)|_{t=0} \end{aligned}$$

Shape Derivative (Example 1)

Let $\Omega \subset D$, $f \in C^1(\overline{D})$ and

$$\mathcal{J}(\Omega) := \int_\Omega f(x)\,\mathrm{d}x.$$

Using the transformation $y = T_t(x) = x + tX(x)$, we get

$$\begin{aligned} d\mathcal{J}(\Omega; X) &= \frac{d}{dt}\mathcal{J}(\Omega_t)|_{t=0} \\ &= \frac{d}{dt}\left(\int_{\Omega_t} f(y)\,\mathrm{d}y\right)|_{t=0} \\ &= \frac{d}{dt}\left(\int_{\Omega} f(T_t(x))\,\det(DT_t(x))\,\mathrm{d}x\right)|_{t=0} \end{aligned}$$

Noting that $\frac{d}{dt}\det(DT_t(x))|_{t=0} = \mathrm{div}(X)$ and $\frac{d}{dt}T_t(x)|_{t=0} = X$, we get further

$$d\mathcal{J}(\Omega; X) = \int_\Omega f(x)\mathrm{d}iv(X) + \nabla f \cdot X\,\mathrm{d}x$$

Shape Derivative (Example 2)

We consider the PDE-constrained shape optimization problem

$$\min_{\Omega} J(u) := \int_{\Omega} |u - u_d|^2 \, \mathrm{d}x$$

$$\text{s.t. } u \in H_0^1(\Omega) : \int_{\Omega} \nabla u \cdot \nabla v \mathrm{d}x = \int_{\Omega} f v \, \mathrm{d}x \; \forall v \in H_0^1(\Omega)$$

Shape Derivative (Example 2)

We consider the PDE-constrained shape optimization problem

$$\min_{\Omega} J(u) := \int_{\Omega} |u - u_d|^2 \, \mathrm{d}x$$

$$\text{s.t. } u \in H_0^1(\Omega) : \int_{\Omega} \nabla u \cdot \nabla v \mathrm{d}x = \int_{\Omega} f v \, \mathrm{d}x \; \forall v \in H_0^1(\Omega)$$

Defining u_Ω as the unique solution to the PDE constraint, we define the reduced functional $\mathcal{J}(\Omega) := J(u_\Omega)$.

Shape Derivative (Example 2)

We consider the PDE-constrained shape optimization problem

$$\min_{\Omega} J(u) := \int_{\Omega} |u - u_d|^2 \, \mathrm{d}x$$
$$\text{s.t. } \; u \in H_0^1(\Omega) : \int_{\Omega} \nabla u \cdot \nabla v \mathrm{d}x = \int_{\Omega} fv \, \mathrm{d}x \; \forall v \in H_0^1(\Omega)$$

Defining u_Ω as the unique solution to the PDE constraint, we define the reduced functional $\mathcal{J}(\Omega) := J(u_\Omega)$.
For the shape derivative, we introduce the perturbed Lagrangian

$$G(t, \varphi, \psi) = \int_{\Omega} \xi(t) |\varphi - u_d^t|^2 \mathrm{d}x + \int_{\Omega} (DT_t^{-T} \nabla \varphi) \cdot (DT_t^{-T} \nabla \psi) \xi(t) \, dx - \int_{\Omega} \xi(t) f^t \psi \, dx$$

where we used the notation $\xi(t) := \det(DT_t(x))$, $u_d^t = u_d \circ T_t$ and $f^t = f \circ T_t$.

Shape Derivative (Example 2)

We consider the PDE-constrained shape optimization problem

$$\min_{\Omega} J(u) := \int_{\Omega} |u - u_d|^2 \, \mathrm{d}x$$

$$\text{s.t. } \; u \in H_0^1(\Omega) : \int_{\Omega} \nabla u \cdot \nabla v \mathrm{d}x = \int_{\Omega} fv \, \mathrm{d}x \;\forall v \in H_0^1(\Omega)$$

Defining u_Ω as the unique solution to the PDE constraint, we define the reduced functional $\mathcal{J}(\Omega) := J(u_\Omega)$.
For the shape derivative, we introduce the perturbed Lagrangian

$$G(t, \varphi, \psi) = \int_{\Omega} \xi(t) |\varphi - u_d^t|^2 \mathrm{d}x + \int_{\Omega} (DT_t^{-T} \nabla \varphi) \cdot (DT_t^{-T} \nabla \psi) \xi(t) \, dx - \int_{\Omega} \xi(t) f^t \psi \, dx$$

where we used the notation $\xi(t) := \det(DT_t(x))$, $u_d^t = u_d \circ T_t$ and $f^t = f \circ T_t$.
For u^t the pull-back of the solution of the perturbed problem, it obviously holds that

$$\mathcal{J}(\Omega_t) = G(t, u^t, \psi)$$

for all ψ, thus $d\mathcal{J}(\Omega; X) = \frac{d}{dt} \mathcal{J}(\Omega_t)|_{t=0} = \frac{d}{dt} G(t, u^t, \psi)|_{t=0}$.

Shape Derivative (Example 2)

Moreover, it can be shown that[1] $\frac{d}{dt}G(t, u^t, \psi)|_{t=0} = \frac{\partial}{\partial t}G(0, u, p)$ where u and p are the unperturbed state and adjoint state, thus:

$$d\mathcal{J}(\Omega; X) = \frac{\partial}{\partial t}G(0, u, p).$$

[1] K. Sturm, SIAM J. Control Optim., 2015

Shape Derivative (Example 2)

Moreover, it can be shown that[1] $\frac{d}{dt} G(t, u^t, \psi)|_{t=0} = \frac{\partial}{\partial t} G(0, u, p)$ where u and p are the unperturbed state and adjoint state, thus:

$$d\mathcal{J}(\Omega; X) = \frac{\partial}{\partial t} G(0, u, p).$$

$$\begin{aligned}\frac{\partial}{\partial t} G(t, \varphi, \psi)|_{t=0} =& \frac{\partial}{\partial t} \left(\int_\Omega \xi(t) |\varphi - u_d^t|^2 \mathrm{d}x \right. \\ & \left. + \int_\Omega (DT_t^{-T} \nabla \varphi) \cdot (DT_t^{-T} \nabla \psi) \xi(t) \, dx - \int_\Omega \xi(t) f^t \psi \, dx \right) |_{t=0}\end{aligned}$$

[1] K. Sturm, SIAM J. Control Optim., 2015

Shape Derivative (Example 2)

Moreover, it can be shown that[1] $\frac{d}{dt} G(t, u^t, \psi)|_{t=0} = \frac{\partial}{\partial t} G(0, u, p)$ where u and p are the unperturbed state and adjoint state, thus:

$$d\mathcal{J}(\Omega; X) = \frac{\partial}{\partial t} G(0, u, p).$$

$$\begin{aligned}
\frac{\partial}{\partial t} G(t, \varphi, \psi)|_{t=0} =& \frac{\partial}{\partial t} \Big(\int_\Omega \xi(t) |\varphi - u_d^t|^2 \mathrm{d}x \\
&+ \int_\Omega (DT_t^{-T} \nabla \varphi) \cdot (DT_t^{-T} \nabla \psi) \xi(t) \, dx - \int_\Omega \xi(t) f^t \psi \, dx \Big) |_{t=0} \\
=& \int_\Omega \mathrm{div}(X) |\varphi - u_d|^2 - 2(\varphi - u_d) \nabla u_d \cdot X \, dx \\
&+ \int_\Omega (\mathrm{div}(X) I - DX - DX^\top) \nabla \varphi \cdot \nabla \psi \, dx \\
&- \int_\Omega (\nabla f \cdot X + f \mathrm{div}(X)) \psi \, dx.
\end{aligned}$$

[1] K. Sturm, SIAM J. Control Optim., 2015

Automatic Shape Differentiation in NGSolve

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \underbrace{\int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x}_{=:G_1(t)} + \underbrace{\int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\,dx}_{=:G_2(t)} - \underbrace{\int_\Omega \xi(t) f^t p\,dx}_{=:G_3(t)},$$

$T_t(x) = x + tX(x),\ F_t = DT_t = I + tDX,\ u_d^t = u_d \circ T_t,\ f^t = f \circ T_t.$

Automatic Shape Differentiation in NGSolve

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \underbrace{\int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x}_{=:G_1(t)} + \underbrace{\int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\,dx}_{=:G_2(t)} - \underbrace{\int_\Omega \xi(t) f^t p\,dx}_{=:G_3(t)},$$

$T_t(x) = x + tX(x)$, $F_t = DT_t = I + tDX$, $u_d^t = u_d \circ T_t$, $f^t = f \circ T_t$.

Procedure:

$$\frac{dG_1(t)}{dt} = \frac{dG_1}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \underbrace{\frac{dG_1}{du_d}\frac{du_d}{dy}}_{dG_1/dy}\underbrace{\frac{dT_t}{dt}}_{X} \qquad = \frac{dG_1}{d\xi}\mathrm{div}(X) + \frac{dG_1}{dy}\cdot X$$

Automatic Shape Differentiation in NGSolve

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \underbrace{\int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x}_{=:G_1(t)} + \underbrace{\int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\,dx}_{=:G_2(t)} - \underbrace{\int_\Omega \xi(t) f^t p\,dx}_{=:G_3(t)},$$

$$T_t(x) = x + tX(x),\ F_t = DT_t = I + tDX,\ u_d^t = u_d \circ T_t,\ f^t = f \circ T_t.$$

Procedure:

$$\frac{dG_1(t)}{dt} = \frac{dG_1}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \underbrace{\frac{dG_1}{du_d}\frac{du_d}{dy}}_{dG_1/dy}\underbrace{\frac{dT_t}{dt}}_{X} \qquad = \frac{dG_1}{d\xi}\mathrm{div}(X) + \frac{dG_1}{dy}\cdot X$$

$$\frac{dG_2(t)}{dt} = \frac{dG_2}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \frac{dG_2}{dF_t}\underbrace{\frac{dF_t}{dt}}_{DX} \qquad = \frac{dG_2}{d\xi}\mathrm{div}(X) + \frac{dG_2}{dF_t}DX$$

Automatic Shape Differentiation in NGSolve

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \underbrace{\int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x}_{=:G_1(t)} + \underbrace{\int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\,dx}_{=:G_2(t)} - \underbrace{\int_\Omega \xi(t) f^t p\,dx}_{=:G_3(t)},$$

$T_t(x) = x + tX(x)$, $F_t = DT_t = I + tDX$, $u_d^t = u_d \circ T_t$, $f^t = f \circ T_t$.

Procedure:

$$\frac{dG_1(t)}{dt} = \frac{dG_1}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \underbrace{\frac{dG_1}{du_d}\frac{du_d}{dy}}_{dG_1/dy}\underbrace{\frac{dT_t}{dt}}_{X} \qquad = \frac{dG_1}{d\xi}\mathrm{div}(X) + \frac{dG_1}{dy}\cdot X$$

$$\frac{dG_2(t)}{dt} = \frac{dG_2}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \frac{dG_2}{dF_t}\underbrace{\frac{dF_t}{dt}}_{DX} \qquad = \frac{dG_2}{d\xi}\mathrm{div}(X) + \frac{dG_2}{dF_t}DX$$

$$\frac{dG_3(t)}{dt} = \frac{dG_3}{d\xi}\underbrace{\frac{d\xi}{dt}}_{div(X)} + \underbrace{\frac{dG_3}{df}\frac{df}{dy}}_{dG_3/dy}\underbrace{\frac{dT_t}{dt}}_{X} \qquad = \frac{dG_3}{d\xi}\mathrm{div}(X) + \frac{dG_3}{dy}\cdot X$$

Automatic Shape Differentiation in NGSolve

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \underbrace{\int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x}_{=:G_1(t)} + \underbrace{\int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\, dx}_{=:G_2(t)} - \underbrace{\int_\Omega \xi(t) f^t p\, dx}_{=:G_3(t)},$$

$T_t(x) = x + tX(x)$, $F_t = DT_t = I + tDX$, $u_d^t = u_d \circ T_t$, $f^t = f \circ T_t$.

$$\frac{d}{dt}\left(G^{u,p}(t)\right)|_{t=0} = \frac{dG^{u,p}}{d\xi}\mathrm{div}(X) + \frac{dG^{u,p}}{dF}DX + \frac{dG^{u,p}}{dy}\cdot X$$

Automatic Shape Differentiation in NGSolve

$$G^{u,p}(t) := \int_\Omega \xi(t)|u - u_d^t|^2 \mathrm{d}x + \int_\Omega (F_t^{-T}\nabla u)\cdot(F_t^{-T}\nabla p)\xi(t)\, dx - \int_\Omega \xi(t) f^t p\, dx$$

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt}\left(G^{u,p}(t)\right)_{t=0} = \frac{dG^{u,p}}{d\xi}\mathrm{div}(X) + \frac{dG^{u,p}}{dF}DX + \frac{dG^{u,p}}{dy}\cdot X$$

Code in NGSolve:

```
def Cost(u):
    return (u-gfud)*(u-gfud) * xi * dx

def Equation(u,v):
    return (alpha*(Finv.trans*grad(u))*(Finv.trans*grad(v))-f*v) * xi * dx

a = BilinearForm(fes, symmetric=True)
a += Equation(u,v)

G = Cost(gfu) + Equation(gfu,gfp)   #gfu/gfp solutions to state/adjoint eqn

dJOmega = LinearForm(VEC)
dJOmega += G.Derive(xi, div(X))
dJOmega += G.Derive(F, Grad(X))
dJOmega += G.Derive(x, X[0]) + G.Derive(y, X[1])
```

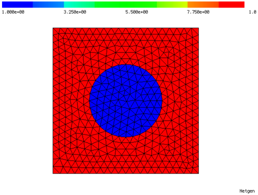
Numerical Example: Nonlinear Transmission

Model Problem 1

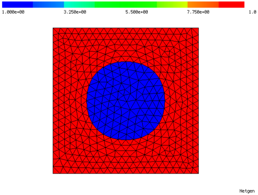
$$\min_{\Omega} J(u) := \int_D |u - u_d|^2 \, \mathrm{d}x$$

$$\text{s.t. } u \in H_0^1(D) : \int_D \alpha_\Omega(|\nabla u|) \nabla u \cdot \nabla v \mathrm{d}x = \int_D f v \, \mathrm{d}x \; \forall v \in H_0^1(D)$$

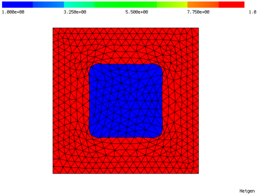
where $\alpha_\Omega(|\nabla u|) = \chi_\Omega(x)\alpha_0(|\nabla u|) + \chi_{D\setminus\Omega}(x)\alpha_1$, $\Omega \subset D$ and given $u_d, f \in C^1(\mathbb{R}^2)$.



iteration 0



iteration 50



iteration 200

Automatic Shape Differentiation in NGSolve: $H(curl)$ setting (nonlinear)

Covariant transformation:

$$u \circ T_t = (DT_t)^{-T} \hat{u}$$

$$(\text{curl}(u)) \circ T_t = \frac{1}{\det(DT_t)} DT_t \, \hat{\text{curl}}(\hat{u})$$

$$d\mathcal{J}(\Omega; X) = \frac{d}{dt} \left(G^{u,p}(t)\right)|_{t=0}$$

where

$$G^{u,p}(t) := \int_{\Omega_g} \xi(t) \left| \frac{1}{\xi(t)} F_t \text{curl} u \cdot \boldsymbol{n} - (B_d^n)^t \right|^2 \, \mathrm{d}x$$
$$+ \int_D \xi(t) \nu_\Omega \left(\left| \frac{1}{\xi(t)} F_t curl(u) \right| \right) \left(\frac{1}{\xi(t)} F_t curl(u) \right) \cdot \left(\frac{1}{\xi(t)} F_t curl(v) \right) \mathrm{d}x$$
$$- \int_{\Omega_{mag}} \xi(t) \nu_{mag} \, M^t \left(\frac{1}{\xi(t)} F_t curl(v) \right) \mathrm{d}x,$$

$T_t(x) = x + tX(x)$, $F_t = DT_t = I + tDX$, $(B_d^n)^t = B_d^n \circ T_t$, $M^t = M \circ T_t$.

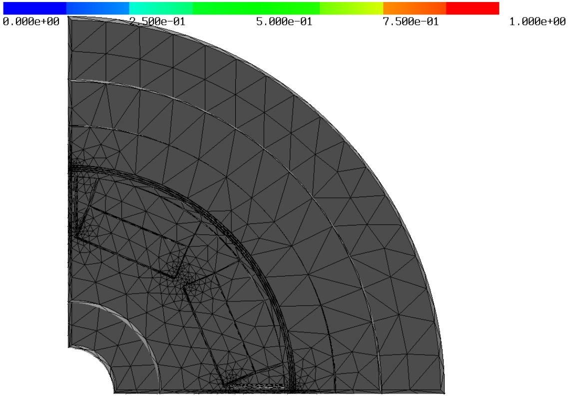
$$\frac{d}{dt}\left(G^{u,p}(t)\right) = \frac{dG^{u,p}}{d\xi} \text{div}(X) + \frac{dG^{u,p}}{dF} DX + \frac{dG^{u,p}}{dy} \cdot X$$

First Results Electric Motor (proof of concept)

Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot \boldsymbol{n} - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in \mathbf{V} : \int_D \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v \mathrm{d}x = \langle \boldsymbol{F}, v \rangle \quad \text{for all } v \in \mathbf{V}.$$

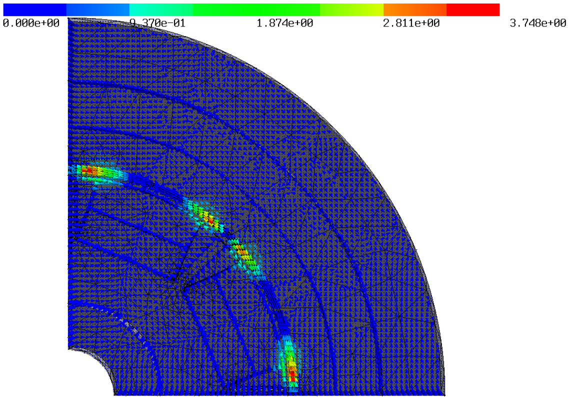


First Results Electric Motor (proof of concept)

Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot \boldsymbol{n} - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in \mathbf{V} : \int_{\mathrm{D}} \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v \mathrm{d}x = \langle \boldsymbol{F}, v \rangle \quad \text{for all } v \in \mathbf{V}.$$

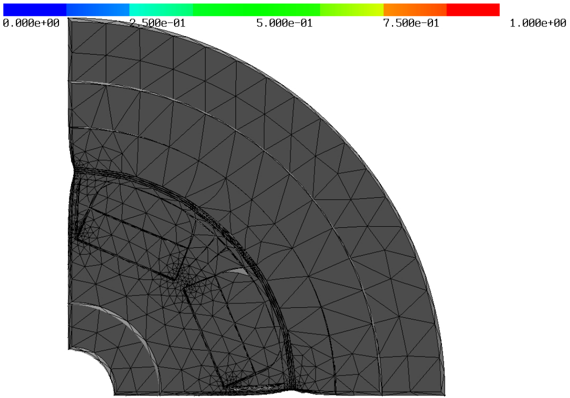


First Results Electric Motor (proof of concept)

Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot \boldsymbol{n} - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in \mathbf{V} : \int_{\mathrm{D}} \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v \mathrm{d}x = \langle \boldsymbol{F}, v \rangle \quad \text{for all } v \in \mathbf{V}.$$

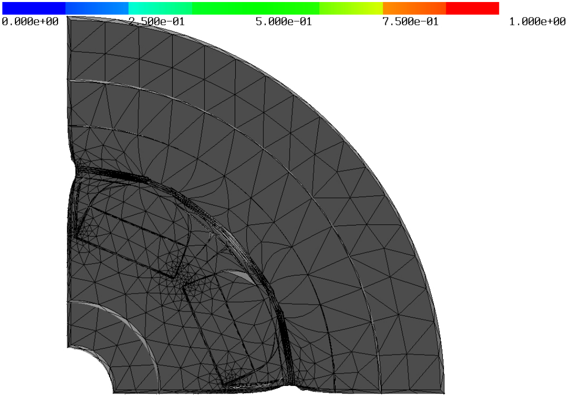


First Results Electric Motor (proof of concept)

Quasilinear Model Problem

$$\min_{\Omega \in \mathcal{A}(D)} J(u) = \int_{\Omega_g} |\mathrm{curl} u \cdot \boldsymbol{n} - B_d^n|^2 \mathrm{d}x$$

$$\text{s.t. } u \in \mathbf{V} : \int_{\mathrm{D}} \nu_\Omega(x, |\mathrm{curl} u|) \mathrm{curl} u \cdot \mathrm{curl} v \mathrm{d}x = \langle \boldsymbol{F}, v \rangle \quad \text{for all } v \in \mathbf{V}.$$



Outline

Topological Derivative: Definition

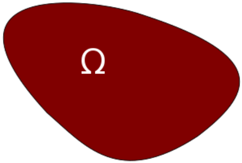
Idea:

Sensitivity of $\mathcal{J} = \mathcal{J}(\Omega) = J(\Omega, u(\Omega))$ w.r.t. insertion of hole $\omega_\varepsilon = x_0 + \varepsilon\,\omega$ (ω e.g. dots unit disk)

Definition

Let d denote the space dimension. The topological derivative of a domain-dependent functional $\mathcal{J}$ at a spatial point x_0 is defined as the quantity $\mathcal{T}(x_0)$ satisfying the topological asymptotic expansion

$$\mathcal{J}(\Omega_\varepsilon) - \mathcal{J}(\Omega) = \varepsilon^d \mathcal{T}(x_0) + o(\varepsilon^d).$$



Topological Derivative: Definition

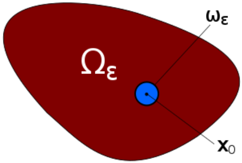
Idea:

Sensitivity of $\mathcal{J} = \mathcal{J}(\Omega) = J(\Omega, u(\Omega))$ w.r.t. insertion of hole $\omega_\varepsilon = x_0 + \varepsilon\,\omega$ (ω e.g. dots unit disk)

Definition

Let d denote the space dimension. The topological derivative of a domain-dependent functional $\mathcal{J}$ at a spatial point x_0 is defined as the quantity $\mathcal{T}(x_0)$ satisfying the topological asymptotic expansion

$$\mathcal{J}(\Omega_\varepsilon) - \mathcal{J}(\Omega) = \varepsilon^d \mathcal{T}(x_0) + o(\varepsilon^d).$$



Topological Derivative: Definition

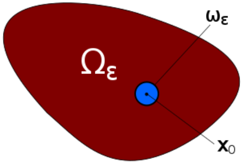
Idea:

Sensitivity of $\mathcal{J} = \mathcal{J}(\Omega) = J(\Omega, u(\Omega))$ w.r.t. insertion of hole $\omega_\varepsilon = x_0 + \varepsilon\,\omega$ (ω e.g. dots unit disk)

Definition

Let d denote the space dimension. The topological derivative of a domain-dependent functional $\mathcal{J}$ at a spatial point x_0 is defined as the quantity $\mathcal{T}(x_0)$ satisfying the topological asymptotic expansion

$$\mathcal{J}(\Omega_\varepsilon) - \mathcal{J}(\Omega) = \varepsilon^d \mathcal{T}(x_0) + o(\varepsilon^d).$$



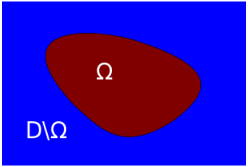
- $\mathcal{T}(x_0) < 0 \Longrightarrow \mathcal{J}(\Omega_\varepsilon) < \mathcal{J}(\Omega)$ for ε small enough

Model Problem

$$\min \; J(u) := \int_D |u - u_d|^2 \mathrm{d}x$$

$$\text{s.t. } u \in H_0^1(D) : \int_D \alpha_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x = \int_D f v \, \mathrm{d}x \;\forall v \in H_0^1(D)$$

$$\text{where } \alpha_\Omega(x) = \begin{cases} \alpha_0 & x \in \Omega \\ \alpha_1 & x \in D \setminus \overline{\Omega} \end{cases}, \text{ with } \alpha_0, \alpha_1 > 0.$$



Here, we have

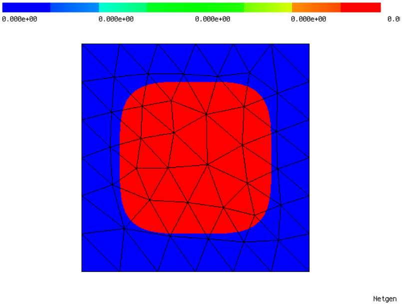
- $\mathcal{T}^{\Omega \to D\setminus\Omega}(x) = 2\alpha_0 \frac{\alpha_1 - \alpha_0}{\alpha_1 + \alpha_0} \pi \nabla u(x) \cdot \nabla p(x)$ for all $x \in \Omega$
- $\mathcal{T}^{D\setminus\Omega \to \Omega}(x) = 2\alpha_1 \frac{\alpha_0 - \alpha_1}{\alpha_0 + \alpha_1} \pi \nabla u(x) \cdot \nabla p(x)$ for all $x \in D \setminus \overline{\Omega}$

Solving an interface problem using XFEM

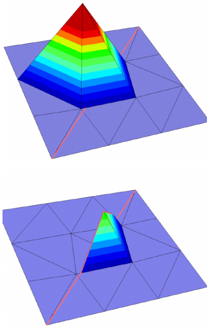
Interface problem

$$\begin{cases} -\nabla \cdot (\alpha_0 \nabla u) = f & \text{in} \quad \Omega, \\ -\nabla \cdot (\alpha_1 \nabla u) = f & \text{in} \quad D \setminus \overline{\Omega}, \\ [\![u]\!] = 0 & \text{on} \quad \partial\Omega, \\ [\![-\alpha \nabla u \cdot \mathbf{n}]\!] = 0 & \text{on} \quad \partial\Omega, \\ u = u_D & \text{on} \quad \partial D. \end{cases}$$

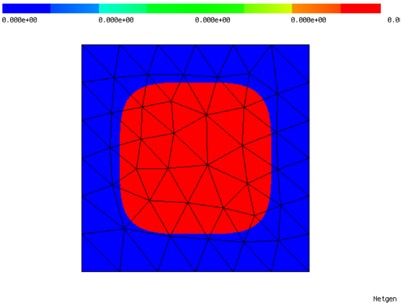
where $\alpha_0, \alpha_1 > 0$, $f \in C^1(\overline{D})$.



Solving an interface problem using XFEM



Standard and cut-off basis function
(C. Lehrenfeld)



$$V_h^\Gamma = V_h + V_h^x$$

where

- V_h standard finite element space on background mesh
- V_h^x cut-off basis functions

Solving an interface problem using XFEM

Nitsche formulation: Find $u_h \in V_h^\Gamma$:

$$\sum_{i \in \{+,-\}} (\alpha_i \nabla u_h \nabla v_h)_{\Omega_i} + (\{\!\{-\alpha \nabla u_h \cdot n\}\!\}, [\![v_h]\!])_\Gamma + ([\![u_h]\!], \{\!\{-\alpha \nabla v_h \cdot n\}\!\})_\Gamma + \left(\frac{\lambda}{h}[\![u_h]\!], [\![v_h]\!]\right)_\Gamma = (f, v_h)_\Omega$$

for all $v_h \in V_h^\Gamma$

Here: average $\{\!\{\cdot\}\!\}$, jump $[\![\cdot]\!]$

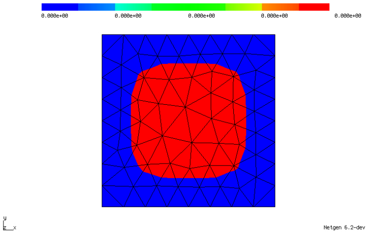
Solving an interface problem using XFEM

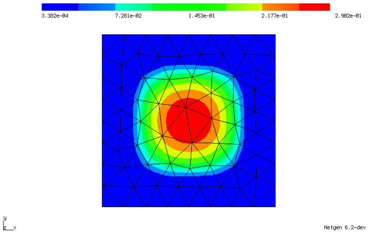
Nitsche formulation: Find $u_h \in V_h^\Gamma$:

$$\sum_{i \in \{+,-\}} (\alpha_i \nabla u_h \nabla v_h)_{\Omega_i} + (\{\!\{-\alpha \nabla u_h \cdot n\}\!\}, [\![v_h]\!])_\Gamma + ([\![u_h]\!], \{\!\{-\alpha \nabla v_h \cdot n\}\!\})_\Gamma + \left(\frac{\lambda}{h}[\![u_h]\!], [\![v_h]\!]\right)_\Gamma = (f, v_h)_\Omega$$

for all $v_h \in V_h^\Gamma$

Here: average $\{\!\{\cdot\}\!\}$, jump $[\![\cdot]\!]$





Solution by `ngsxfem`

Algorithm (Amstutz, Andrä 2006)

Represent design by level set function ψ:

$$\psi(x) > 0 \Leftrightarrow x \in \Omega_1$$
$$\psi(x) < 0 \Leftrightarrow x \in \Omega_2$$

Generalized topological derivative:

$$g_\psi(x) := \begin{cases} \mathcal{T}^{1\to 2}(x) & x \in \Omega_1 \\ -\mathcal{T}^{2\to 1}(x) & x \in \Omega_2 \end{cases}$$

Lemma

Sufficient local optimality condition: $\psi = g_\psi$

Proof:
Let $\hat{x} \in \Omega_1$. Then $0 < \psi(\hat{x}) = g_\psi(\hat{x}) = \mathcal{T}^{1\to 2}(\hat{x})$, thus, introducing material 2 at $\hat{x}$ will yield an increase of $\mathcal{J}$.
Analogous for $\hat{x} \in \Omega_2$. $\square$

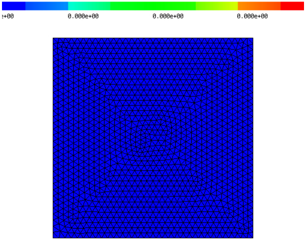
Idea

Idea: Fixed point iteration

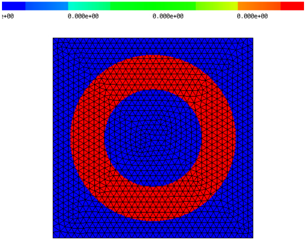
$$\psi_{k+1} = (1-s)\psi_k + s g_\psi$$

where $s \in [0,1]$ as large as possible such that $\mathcal{J}(\psi_{k+1}) < \mathcal{J}(\psi_k)$.

Example Topology Optimization

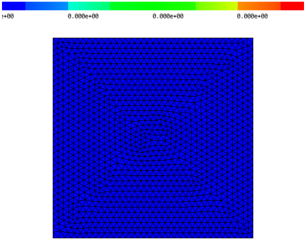


Initial Design

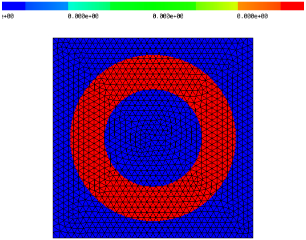


Desired Design

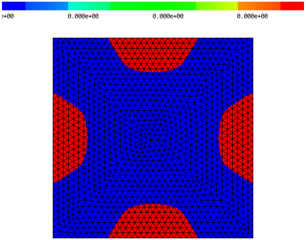
Example Topology Optimization



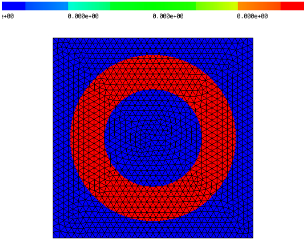
Initial Design



Desired Design



Design at iteration 2



Final Design

Conclusion and Outlook

Conclusion

- Automatic shape differentiation in NGSolve
- Application to 3D electric motor (nonlinear H(curl) setting)
- Topology optimization in NGSolve using `ngsxfem`

Outlook

- Numerical evaluation of topological derivative in nonlinear H(curl) setting
- Topology optimization of 3D electric motor

Conclusion and Outlook

Conclusion

- Automatic shape differentiation in NGSolve
- Application to 3D electric motor (nonlinear H(curl) setting)
- Topology optimization in NGSolve using `ngsxfem`

Outlook

- Numerical evaluation of topological derivative in nonlinear H(curl) setting
- Topology optimization of 3D electric motor

Thank you for your attention!