A tutorial on adjoint methods and applications to stellarator design

Elizabeth Paul

PRINCETON UNIVERSITY

March 2, 2021

U.S. DEPARTMENT OF ENERGY

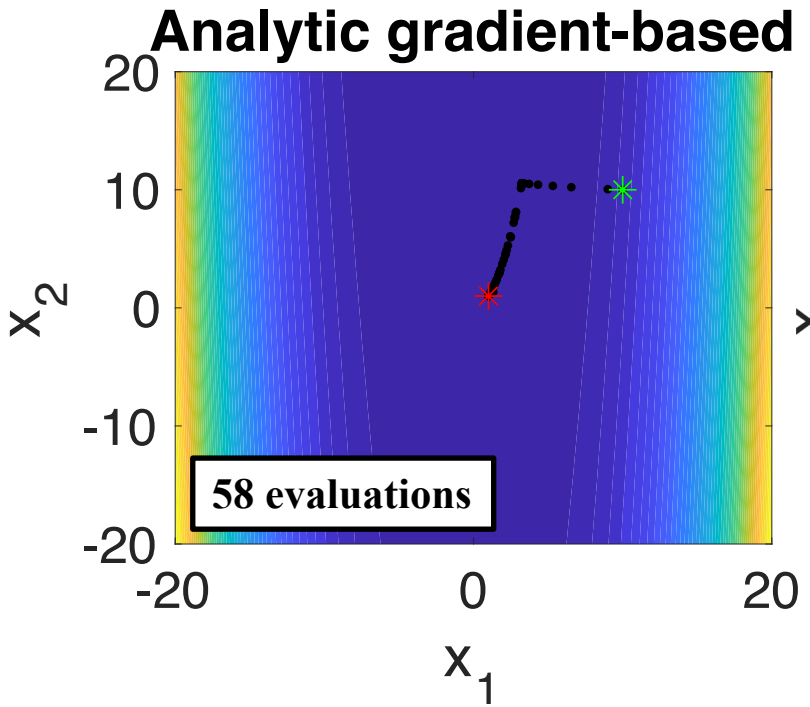
ARCS ADVANCING SCIENCE IN AMERICA

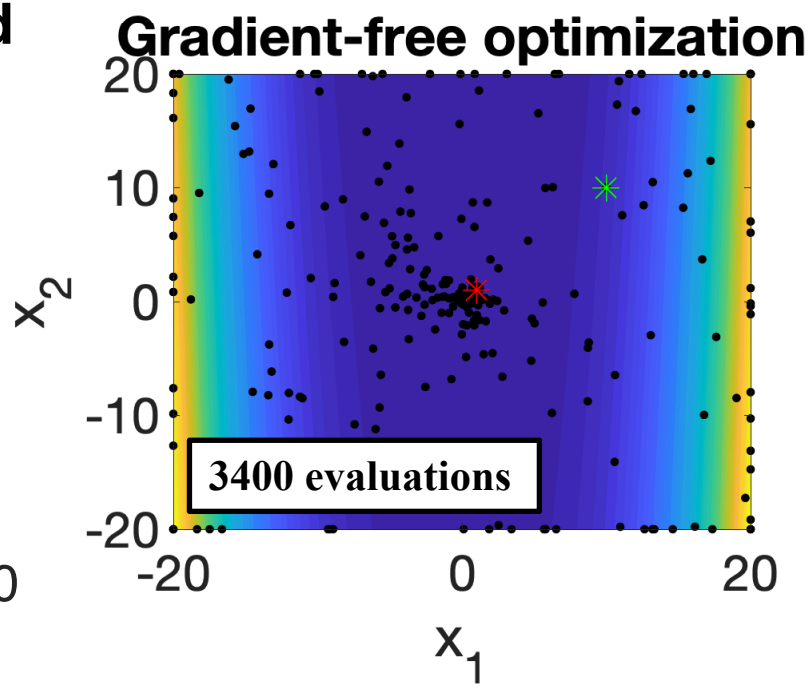
Outline

1. Introduction
2. Discrete adjoint method – “discretize then adjoint”
3. Continuous adjoint method – “adjoint then discretize”
4. Error correction
5. Stellarator applications

Gradient information is valuable for optimization

Minimization of 2D Rosenbrock function

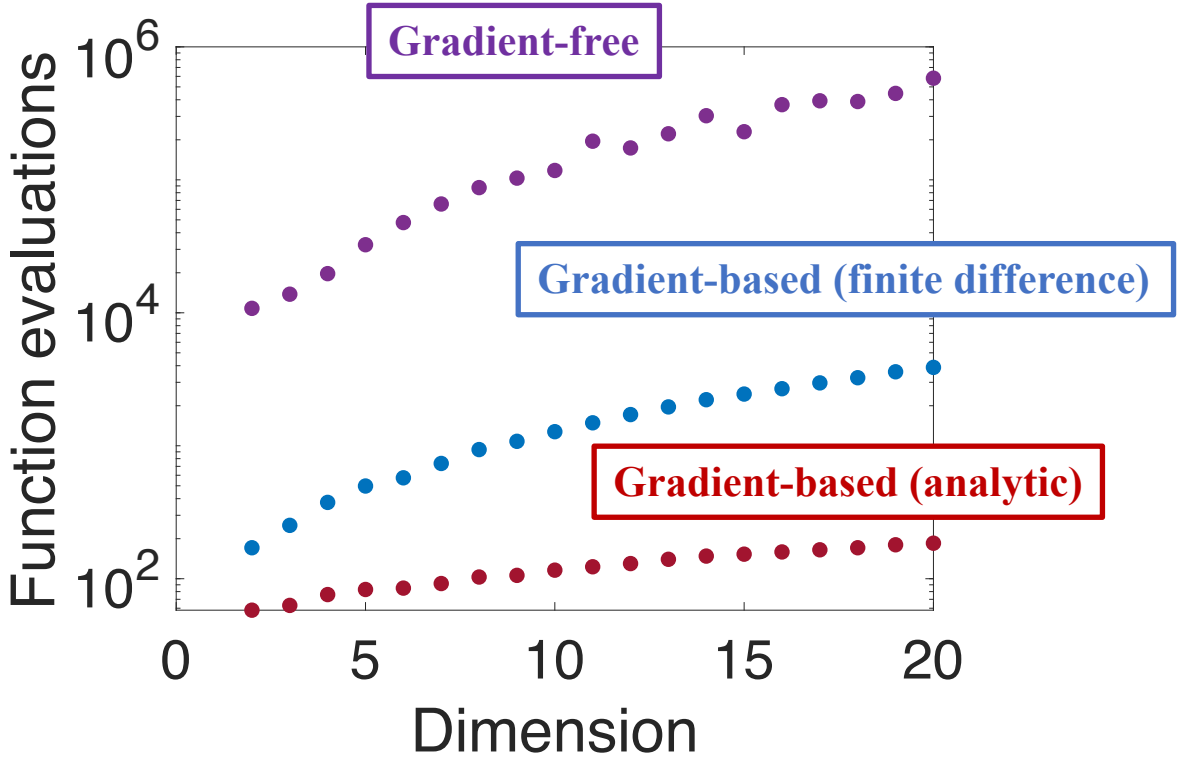




$$f(\{x_i\}) = \sum_{i=1}^{N-1} 100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2$$

Gradient information is valuable for optimization

Especially in high-dimensional spaces



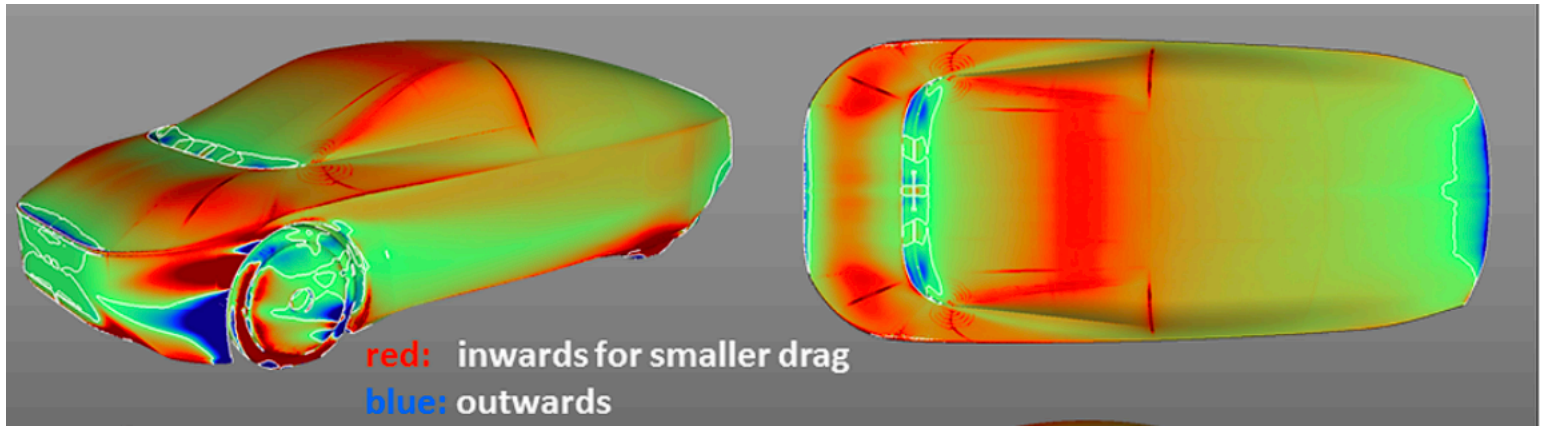
- Finite-difference
- Analytic derivatives
- Automatic differentiation
- Adjoint methods

What is an adjoint method?

- Method of computing derivative of objective depending on solution of system of equations
- Requires solving additional linear equation in which adjoint operator appears

- Useful introductory references
  - M. Giles et al, *Flow, turbulence and combustion 65* (2000).
  - R. Errico, *Bulletin of the American Meteorological Society* 78 (1997).
  - R. Plessix, *Geophys. J. Int.* 167 (2006).

OpenFOAM



Discrete adjoint approach – "discretize then adjoint"

Linear system constrained optimization

$$\overleftrightarrow{\boldsymbol{L}}\boldsymbol{u} = \boldsymbol{b}$$

$$\min_{\Omega} \chi^2(\Omega, \boldsymbol{u}(\Omega))$$

- $\boldsymbol{u}$ = state variables
- Ω = parameters
- χ^2 = cost function

Discrete adjoint approach – "discretize then adjoint"

Linear system constrained optimization

$$\overleftrightarrow{\boldsymbol{L}}\boldsymbol{u} = \boldsymbol{b}$$

$$\min_{\Omega} \chi^2(\Omega, \boldsymbol{u}(\Omega))$$

- $\boldsymbol{u}$ = state variables
- Ω = parameters
- χ^2 = cost function

Variations with linear constraint

$$\frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} + \overleftrightarrow{\boldsymbol{L}}\frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega} = \frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \cdot \frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega}$$

Explicit dependence Implicit dependence

Requires N_Ω solutions of linear system
(finite-difference or analytic derivatives)

Discrete adjoint approach – "discretize then adjoint"

Variations with linear constraint

$$\frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} + \overleftrightarrow{\boldsymbol{L}}\frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega} = \frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \cdot \frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \cdot \left(\overleftrightarrow{\boldsymbol{L}}^{-1} \left(\frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega} - \frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} \right) \right)$$

Discrete adjoint approach – "discretize then adjoint"

Variations with linear constraint

$$\frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} + \overleftrightarrow{\boldsymbol{L}}\frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega} = \frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \cdot \frac{\partial \boldsymbol{u}(\Omega)}{\partial \Omega}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \cdot \left(\overleftrightarrow{\boldsymbol{L}}^{-1} \left(\frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega} - \frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} \right) \right)$$

"Rearrange parenthesis"

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \left(\left(\overleftrightarrow{\boldsymbol{L}}^{-1}\right)^T \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}} \right) \cdot \left(\frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega} - \frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega}\boldsymbol{u} \right)$$

Adjoint equation

$$\boxed{\overleftrightarrow{\boldsymbol{L}}^T \boldsymbol{q} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}}}$$

Discrete adjoint approach – "discretize then adjoint"

The adjoint approach – solve two linear systems

$$\overleftrightarrow{\boldsymbol{L}}\boldsymbol{u} = \boldsymbol{b} \qquad \overleftrightarrow{\boldsymbol{L}}^T \boldsymbol{q} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \boldsymbol{q} \cdot \left(\frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega} - \frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega} \boldsymbol{u} \right)$$

Derivatives computed analytically

Discrete adjoint approach – "discretize then adjoint"

The adjoint approach – solve two linear systems

$$\overleftrightarrow{\boldsymbol{L}}\boldsymbol{u} = \boldsymbol{b} \qquad \overleftrightarrow{\boldsymbol{L}}^T \boldsymbol{q} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \boldsymbol{u}}$$

$$\frac{\partial \chi^2(\Omega, \boldsymbol{u}(\Omega))}{\partial \Omega} = \frac{\partial \chi^2(\Omega, \boldsymbol{u})}{\partial \Omega} + \boldsymbol{q} \cdot \left(\frac{\partial \boldsymbol{b}(\Omega)}{\partial \Omega} - \frac{\partial \overleftrightarrow{\boldsymbol{L}}(\Omega)}{\partial \Omega} \boldsymbol{u} \right)$$

Derivatives computed analytically

Finite difference	Adjoint
$\frac{2}{3} N_\Omega N^3$	$2 N_\Omega N^2 + \frac{2}{3} N^3$

Continuous adjoint approach – "adjoint then discretize"

General PDE-constrained optimization

$$L(\Omega, u) = 0 \;\rightarrow \langle q, L(\Omega, u)\rangle = 0 \text{ for all } q$$

$$\min_{\Omega} \chi^2(\Omega, u(\Omega))$$

- u = state variables
- Ω = parameters
- q = test function
- χ^2 = cost function
- $\langle \ldots \rangle$ = appropriate inner product

Continuous adjoint approach – "adjoint then discretize"

General PDE-constrained optimization

$$L(\Omega, u) = 0 \;\rightarrow \langle q, L(\Omega, u)\rangle = 0 \text{ for all } q$$

$$\min_{\Omega} \chi^2(\Omega, u(\Omega))$$

- u = state variables
- Ω = parameters
- q = test function
- χ^2 = cost function
- $\langle \ldots \rangle$ = appropriate inner product

Variations with PDE constraint

$$\delta L(\Omega, u; \delta\Omega) + \delta L\big(\Omega, u; \delta u(\Omega; \delta\Omega)\big) = 0$$

$$\delta\chi^2(\Omega, u; \delta u) = \langle \widehat{\chi^2}, \delta u\rangle \rightarrow \delta\chi^2(\Omega, u(\Omega); \delta\Omega) = \delta\chi^2(\Omega, u; \delta\Omega) + \langle \widehat{\chi^2}, \delta u(\Omega; \delta\Omega)\rangle$$

Explicit dependence Implicit dependence

Requires N_Ω solutions of PDE
(finite-difference or analytic derivatives)

Continuous adjoint approach – Lagrangian functional

Enforce constraint with Lagrange multiplier

$$\mathcal{L}(\Omega, u, q) = \chi^2(\Omega, u) + \langle q, L(\Omega, u) \rangle$$

Continuous adjoint approach – Lagrangian functional

Enforce constraint with Lagrange multiplier

$$\mathcal{L}(\Omega, u, q) = \chi^2(\Omega, u) + \langle q, L(\Omega, u)\rangle$$

$\delta\mathcal{L}(\Omega, u, q; \delta q) = 0 \rightarrow$ $L(\Omega, u) = 0$ "State equation"

$\delta\mathcal{L}(\Omega, u, q; \delta u) = 0 \rightarrow$ $\delta\chi^2(\Omega, u; \delta u) + \langle q, \delta L(\Omega, u; \delta u)\rangle = 0$ "Adjoint equation"

$\delta\mathcal{L}(\Omega, u, q; \delta\Omega) = 0 \rightarrow$ $\delta\chi^2(\Omega, u; \delta\Omega) + \langle q, \delta L(\Omega, u; \delta\Omega)\rangle = 0$ "Optimality"

Use the adjoint variable (q) to cancel the implicit dependence

Continuous adjoint approach – Lagrangian functional

$$\delta\mathcal{L}(\Omega, u, q; \delta u) = 0 \rightarrow \ \delta\chi^2(\Omega, u; \delta u) + \langle q, \delta L(\Omega, u; \delta u)\rangle = 0$$

Linearize operator and cost function

$$\delta L(\Omega, u; \delta u) = L_0 \delta u \qquad \delta\chi^2(\Omega, u; \delta u) = \langle \widehat{\chi^2}, \delta u\rangle$$

Continuous adjoint approach – Lagrangian functional

$$\delta\mathcal{L}(\Omega, u, q; \delta u) = 0 \rightarrow \ \delta\chi^2(\Omega, u; \delta u) + \langle q, \delta L(\Omega, u; \delta u)\rangle = 0$$

Linearize operator and cost function

$$\delta L(\Omega, u; \delta u) = L_0 \delta u \qquad \delta\chi^2(\Omega, u; \delta u) = \langle \widehat{\chi^2}, \delta u\rangle$$

Use adjoint property

$$\begin{aligned}\delta\mathcal{L}(\Omega, u, q; \delta u) &= \langle \widehat{\chi^2}, \delta u\rangle + \langle q, L_0 \delta u\rangle \\ &= \langle \delta u, \widehat{\chi^2} + L_0^\dagger q\rangle\end{aligned}$$

Solve adjoint problem

$$\boxed{\widehat{\chi^2} + L_0^\dagger q = 0}$$

Continuous adjoint approach – Lagrangian functional

$$\delta\mathcal{L}(\Omega, u, q; \delta u) = 0 \rightarrow \ \delta\chi^2(\Omega, u; \delta u) + \langle q, \delta L(\Omega, u; \delta u)\rangle = 0$$

Linearize operator and cost function

$$\delta L(\Omega, u; \delta u) = L_0 \delta u \qquad \delta\chi^2(\Omega, u; \delta u) = \langle \widehat{\chi^2}, \delta u\rangle$$

Use adjoint property

$$\begin{aligned}\delta\mathcal{L}(\Omega, u, q; \delta u) &= \langle \widehat{\chi^2}, \delta u\rangle + \langle q, L_0 \delta u\rangle \\ &= \langle \delta u, \widehat{\chi^2} + L_0^\dagger q\rangle\end{aligned}$$

Solve adjoint problem

$$\boxed{\widehat{\chi^2} + L_0^\dagger q = 0}$$

Implicit dependence disappears!

$$\delta\mathcal{L}(\Omega, u, q; \delta\Omega) = \delta\chi^2(\Omega, u; \delta\Omega) + \langle q, \delta L(\Omega, u; \delta\Omega) + \cancel{\delta L(\Omega, u; \delta u(\delta\Omega))}\rangle$$

Derivatives computed analytically

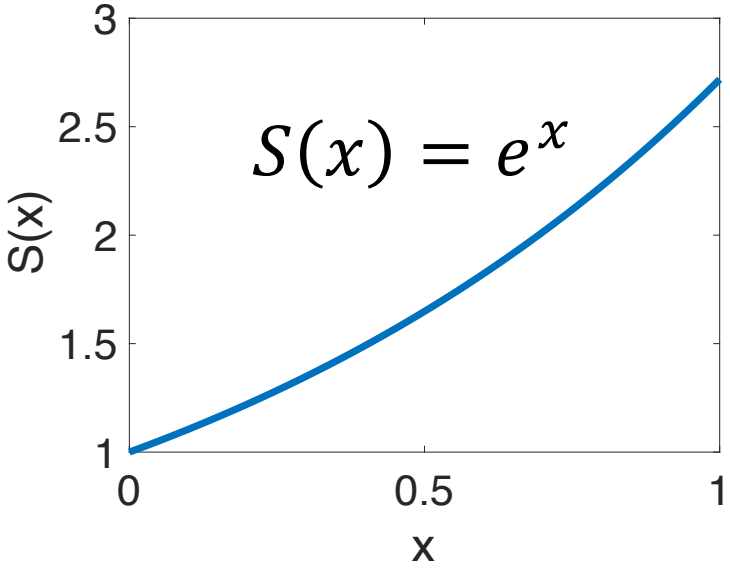
Example – Advection-Diffusion Equation

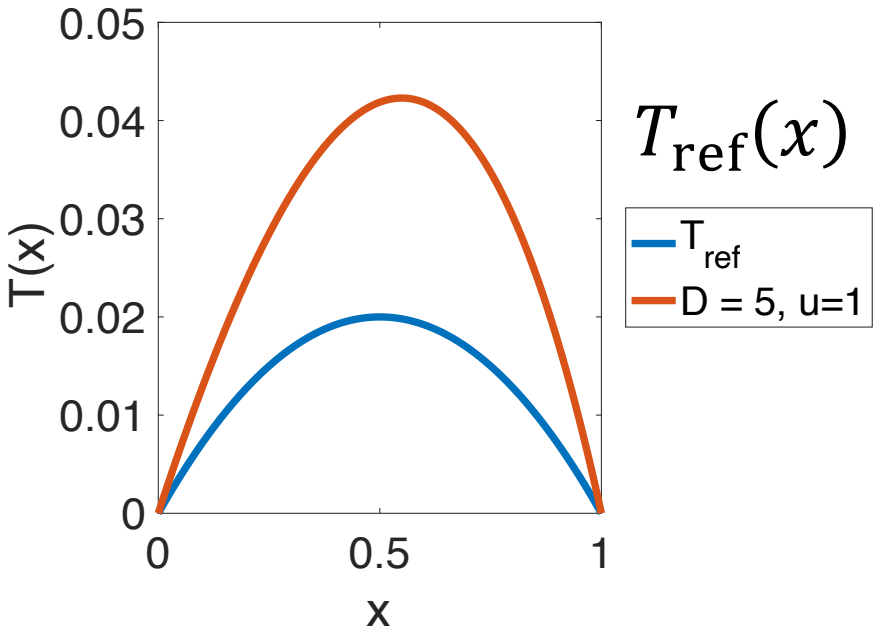
$$F(T, D) = S(x) - u\frac{dT}{dx} + \frac{d}{dx}\left(D\frac{dT}{dx}\right) = 0$$

$$T(0) = T(L) = 0$$

- State variable: $T(x)$ (Temperature)
- Design variable: D (diffusion coefficient)
- Velocity: u = const.
- Heat source: $S(x)$ (prescribed)

Cost functional: $\chi^2(T, D) = \frac{1}{2}\int_0^L (T - T_{\text{ref}})^2 dx$





$$T_{\text{ref}}(x) = 0.08(x - x^2)$$

Example – Advection-Diffusion Equation

1. *Express ODE constraint with Lagrangian functional*

$$\mathcal{L}(T, q, D) = \chi^2(T, D) + \int_0^L dx\, q(x)\, F(T, D)$$

$$= \chi^2\,(T, D) + \int_0^L dx\, q(x)\, \left(S(x) - u\frac{dT}{dx} + \frac{d}{dx}\left(D\frac{dT}{dx}\right)\right)$$

Example – Advection-Diffusion Equation

1. *Express ODE constraint with Lagrangian functional*

$$\mathcal{L}(T,q,D) = \chi^2(T,D) + \int_0^L dx\, q(x)\, F(T,D)$$
$$= \chi^2\,(T,D) + \int_0^L dx\, q(x)\, \left(S(x) - u\frac{dT}{dx} + \frac{d}{dx}\left(D\frac{dT}{dx}\right)\right)$$

2. *Seek stationary point of Lagrangian*

$$0 = \delta\mathcal{L}(T,q,D;\delta T) = \int_0^L (T - T_{\mathrm{ref}})\,\delta T(x) + q(x)\left(u\frac{d\delta T}{dx} + \frac{d}{dx}\left(D\frac{d\delta T}{dx}\right)\right)dx$$
$$= \int_0^L \delta T(x)\,\left((T - T_{\mathrm{ref}}) - \frac{dq(x)}{dx}u + \frac{d}{dx}\left(D\frac{dq}{dx}\right)\right)dx$$
$$+[u\delta T q + D\delta T'(x)q - D\delta T q'(x)]_0^L$$

Example – Advection-Diffusion Equation

$$0 = \delta\mathcal{L}(T, q, D; \delta T) = \int_0^L \delta T(x) \left((T - T_{\text{ref}}) + \frac{dq(x)}{dx} u + \frac{d}{dx}\left(D \frac{dq}{dx} \right) \right) dx$$

$$+ [u \delta T q + D \delta T'(x) q \; - D \delta T q'(x)]_0^L$$

3. *Define adjoint problem*

$$(T - T_{\text{ref}}) + \frac{dq(x)}{dx} u + \frac{d}{dx}\left(D \frac{dq}{dx} \right) = 0$$

$$q(0) = q(L) = 0$$

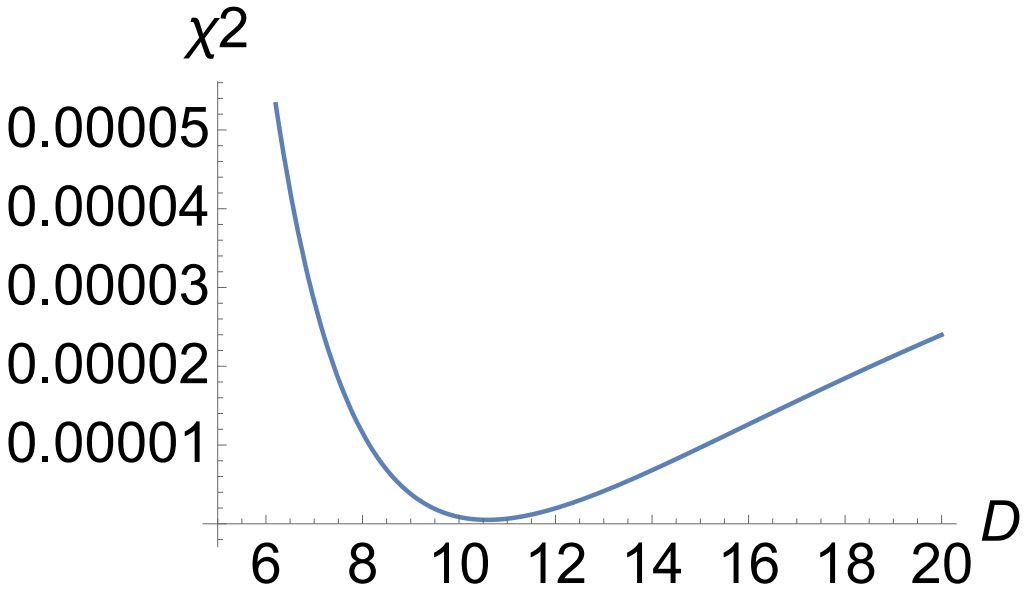
`backward in time/reverse'

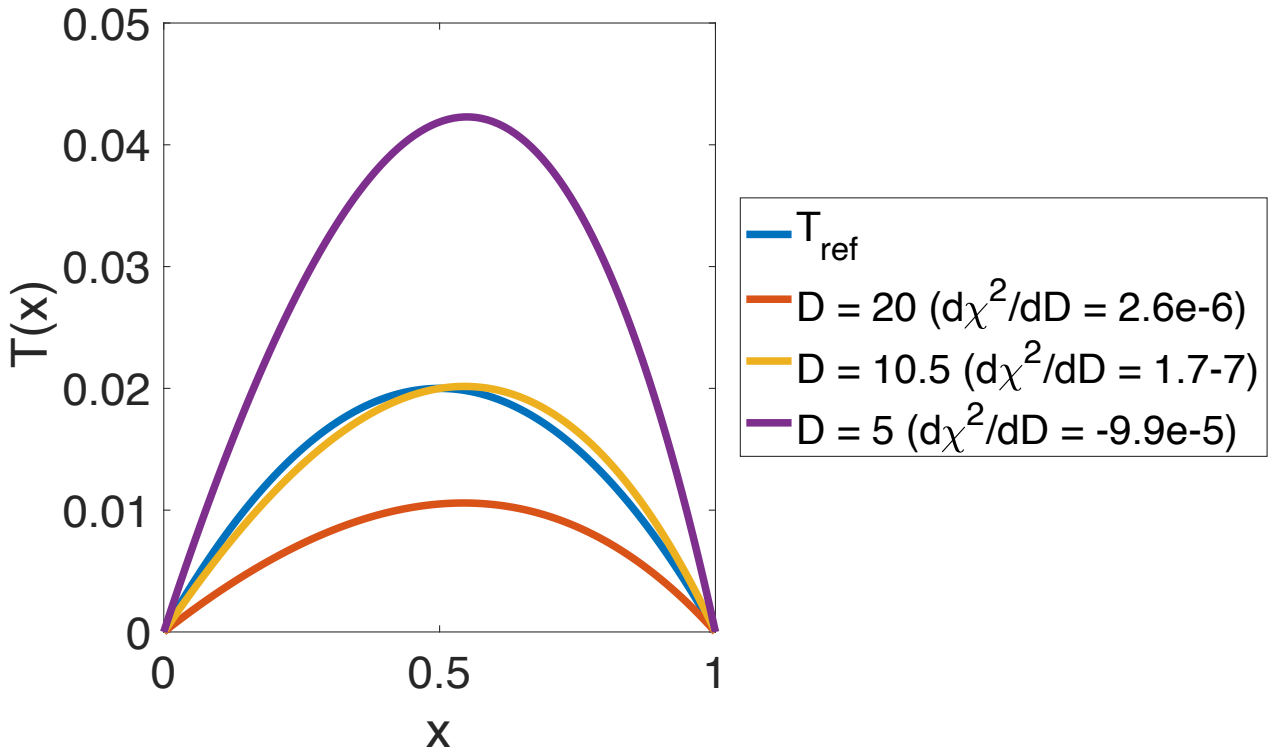
Example – Advection-Diffusion Equation

4. *Compute parameter derivative*

$$\delta\chi^2(T, D; \delta D) = \delta\mathcal{L}(T, q, D; \delta D) + \cancel{\delta\mathcal{L}(T, q\, D,; \delta T(\delta D))}$$

$$= \int_0^L dx\; q(x)\; \left(\frac{d}{dx}\left(\delta D\, \frac{dT}{dx}\right)\right)$$





General considerations – discrete vs. continuous approaches

Discrete approach

- Convenient to implement if PDE is solved with linear system or Newton method
- Only approximation is tolerance of state equation and adjoint solve
- Matrix properties of state and adjoint equation similar (e.g. eigenspectrum, LU factorization)

Continuous approach

- More convenient to implement for specialized solvers (e.g. singular or shock solutions)
- Accuracy depends on discretization of inner product
- Adjoint equation is independent of discretization
- Adjoint equation depends on choice of inner product

Adjoint discretization error correction

Using the same adjoint solution obtained for sensitivity calculations, we can correct (some of) the discretization error in objectives of interest

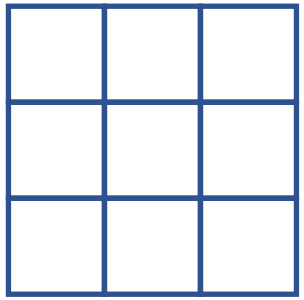
Continuous PDE $$Lu = b$$

Discretized on mesh with typical spacing *H* $$L_H u_H = b_H$$

Objective evaluated on mesh *H* $$\chi_H^2 = \left\langle u_H, \left(\widetilde{\chi^2}\right)^H \right\rangle_H$$

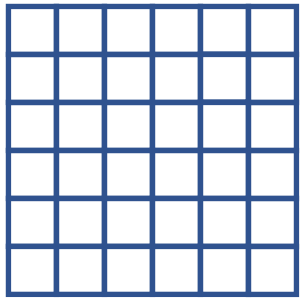
We'd like to know objective on fine mesh *h* (but too expensive to compute) $$\chi_h^2 = \left\langle u_h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h$$

Adjoint discretization error correction



We'd like to know (but don't want to compute) $\chi_h^2 = \left\langle u_h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h$

$$L_h u_h = b_h$$

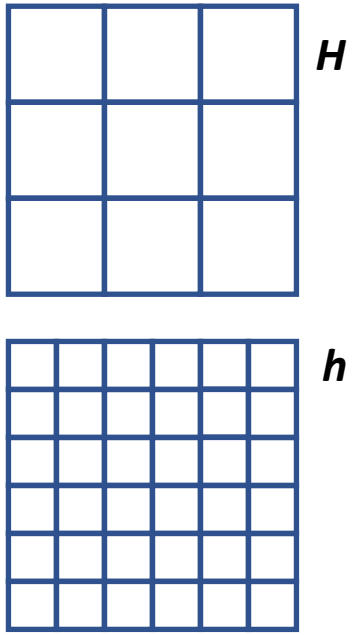


We can write this in terms of *interpolated* coarse-mesh solution, $(u_H)^h$

$$\chi_h^2 = \left\langle (u_H)^h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h + \left\langle \underbrace{\left(u_h - (u_H)^h\right)}_{L_h^{-1}\left(b_h - L_h (u_H)^h\right)}, \left(\widetilde{\chi^2}\right)^h \right\rangle_h$$

Venditti et al, *14th Comput. Fluid Dyn. Conf.*, (1999).
Pierce et al, *Journal of Computational Physics* 200, 769 (2004).

Adjoint discretization error correction



$$\chi_h^2 = \left\langle (u_H)^h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h + \left\langle L_h^{-1}\left(b_h - L_h (F_H)^h\right), \left(\widetilde{\chi^2}\right)^h \right\rangle_h$$

Use adjoint property

$$\chi_h^2 = \left\langle (u_H)^h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h + \left\langle \left(b_h - L_h (u_H)^h\right), \underbrace{\left(L_h^{-1}\right)^\dagger \left(\widetilde{\chi^2}\right)^h}_{q_h} \right\rangle_h$$

Requires solution of adjoint equation on fine mesh

$$L_h^\dagger q_h = \left(\widetilde{\chi^2}\right)^h$$

Adjoint discretization error correction

Assumption: interpolated coarse mesh adjoint solution gives us some information about fine mesh adjoint solution

$$L_H^\dagger q_H = \left(\widetilde{\chi^2}\right)^H \longrightarrow (q_H)^h \approx q_h$$

$$\chi_h^2 = \left\langle (u_H)^h, \left(\widetilde{\chi^2}\right)^h \right\rangle_h + \overbrace{\left\langle b_h - L_h (u_H)^h, (q_H)^h \right\rangle}^{\text{Computable correction}}$$

$$+ \underbrace{\left\langle b_h - L_h (u_H)^h, q_h - (q_H)^h \right\rangle}_{\text{Uncorrectable (hopefully small) error}}$$

Error correction example – Legendre spectral method

State equation

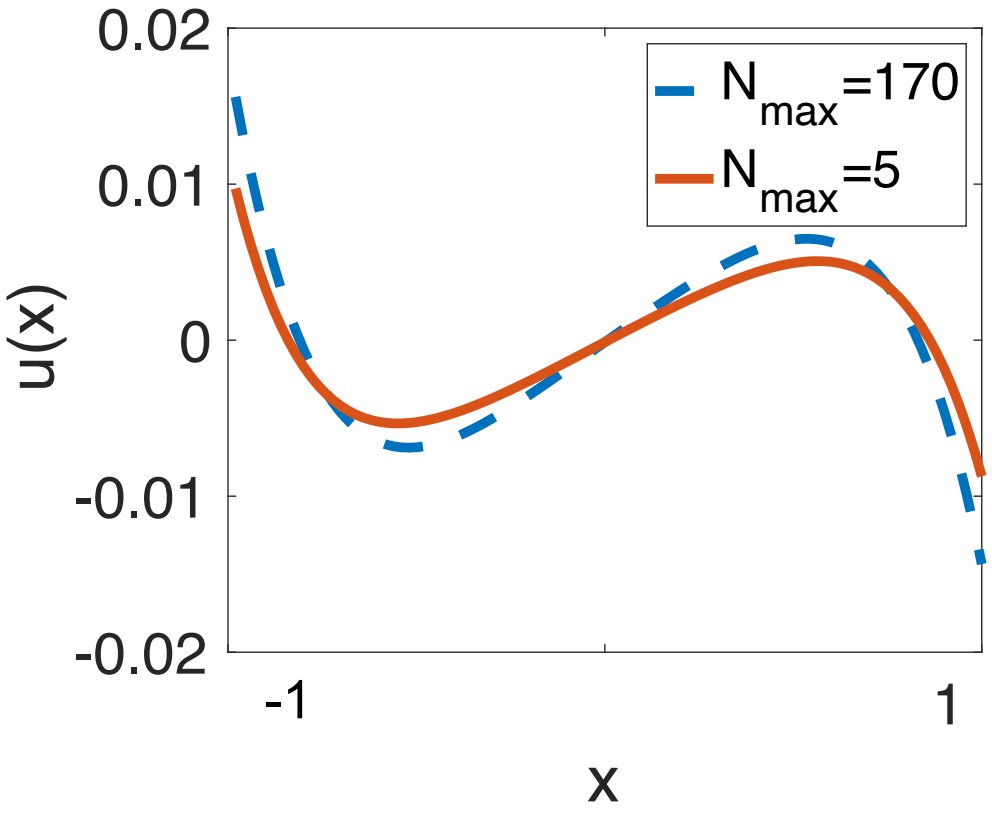
$$A(1+x^2)u(x) + B(1-x^2)\frac{du(x)}{dx} + C\frac{d}{dx}\left((1-x^2)\frac{du}{dx}\right) = \sin(\pi x)$$

Cost functional

$$\chi^2(u) = \int_{-1}^{1} dx\, u(x)\sin(\pi x)$$

Compute solution with Legendre polynomials

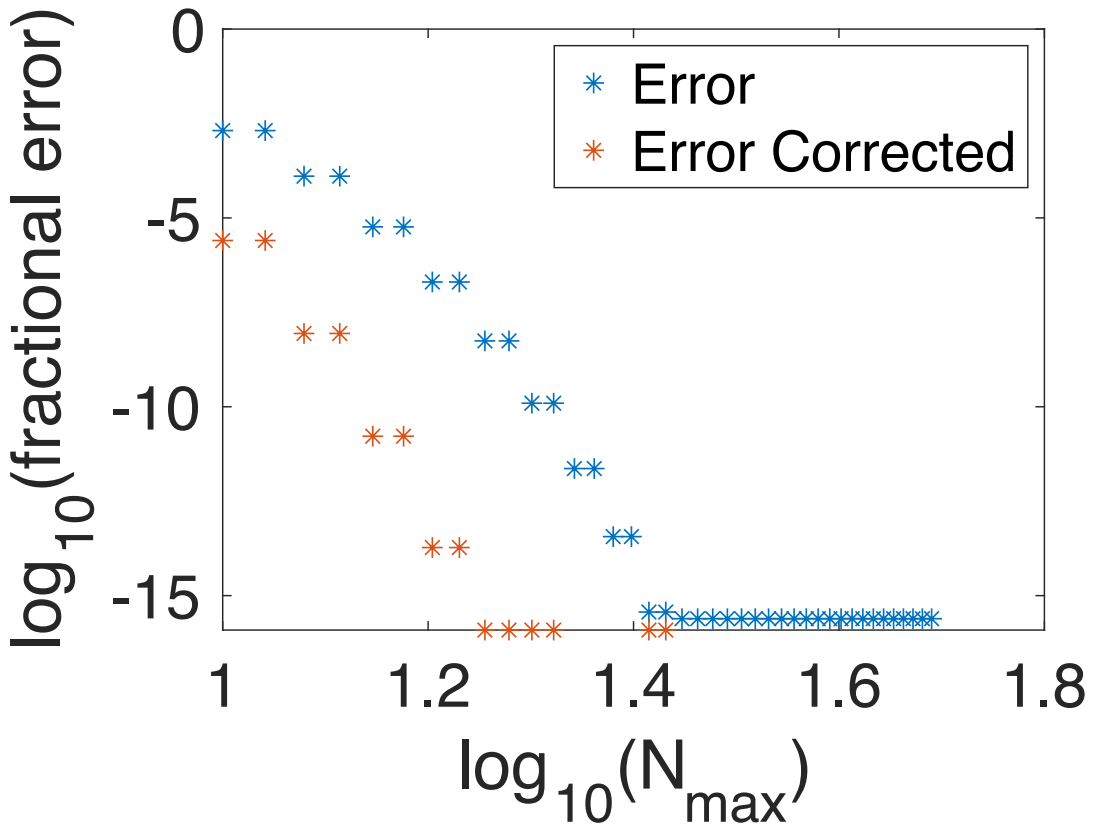
$$u_H(x) = \sum_{n}^{N_{\max}^H} u_n P_n(x)$$



Error correction example – Legendre spectral method

Adjoint equation

$$A(1+x^2)q(x) - B\frac{d}{dx}\left((1-x^2)q(x)\right) + C\frac{d}{dx}\left((1-x^2)\frac{dq}{dx}\right) = \sin(\pi x)$$



"interpolation" = series truncation

$$u_h(x) = \sum_{n}^{N_{\max}^h} u_n P_n(x)$$

$$(u_H(x))^h = \sum_{n}^{N_{\max}^H} u_n P_n(x)$$

Applications for stellarator design

Linearized MHD interpretation of shape derivatives

- Objective: $f(\boldsymbol{B}(S_{\text{plasma}}))$

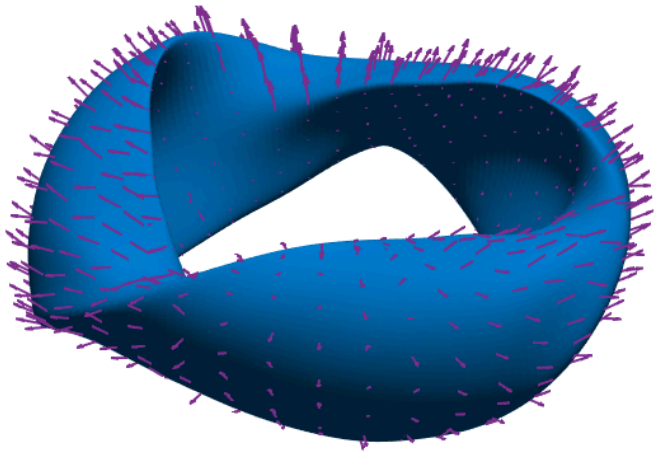
- MHD equilibrium with specified $p(\psi), \iota(\psi)$, and S_{plasma}

$$0 = (\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0\nabla p$$

- Perturbation with fixed $\iota(\psi)$ and $p(\psi)$ determined from $\boldsymbol{\xi}_1$

$$\delta\boldsymbol{B}_1(\boldsymbol{\xi}_1) = \nabla\times(\boldsymbol{\xi}_1\times\boldsymbol{B})$$
$$\delta p(\boldsymbol{\xi}_1) = -\boldsymbol{\xi}_1\cdot\nabla p$$

Displacement
$(\delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}})$



Unperturbed boundary

Linearized MHD interpretation of shape derivatives

- Objective: $f(\boldsymbol{B}(S_{\text{plasma}}))$

- MHD equilibrium with specified $p(\psi), \iota(\psi)$, and S_{plasma}

$$0 = (\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0\nabla p$$

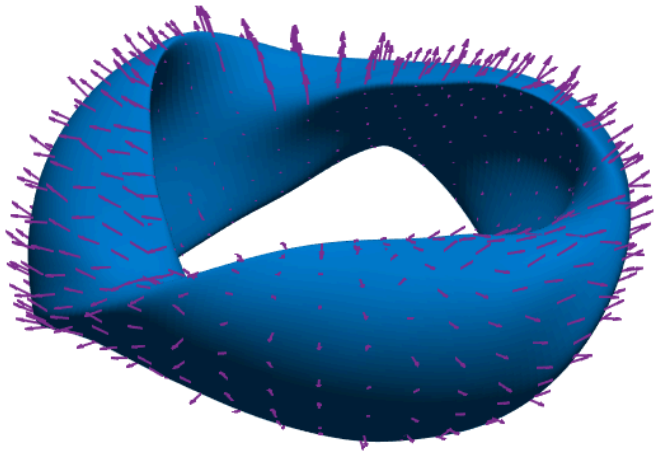
- Perturbation with fixed $\iota(\psi)$ and $p(\psi)$ determined from $\boldsymbol{\xi}_1$

$$\delta\boldsymbol{B}_1(\boldsymbol{\xi}_1) = \nabla\times(\boldsymbol{\xi}_1\times\boldsymbol{B})$$
$$\delta p(\boldsymbol{\xi}_1) = -\boldsymbol{\xi}_1\cdot\nabla p$$

- Perturbed equilibrium with specified $\delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}}|_{S_{\text{plasma}}}$ satisfies

$$\boldsymbol{F}(\boldsymbol{\xi}_1) = (\nabla\times\boldsymbol{B})\times\delta\boldsymbol{B}_1 + (\nabla\times\delta\boldsymbol{B}_1)\times\boldsymbol{B} - \mu_0\boldsymbol{\nabla}\delta p(\boldsymbol{\xi}_1) = 0$$
$$\boldsymbol{\xi}_1\cdot\hat{\boldsymbol{n}}\Big|_{S_{\text{plasma}}} = \delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}}\Big|_{S_{\text{plasma}}}$$

Displacement
$(\delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}})$



Unperturbed boundary

Linearized MHD interpretation of shape derivatives

- Objective: $f(\boldsymbol{B}(S_{\text{plasma}}))$

- MHD equilibrium with specified $p(\psi), \iota(\psi)$, and S_{plasma}

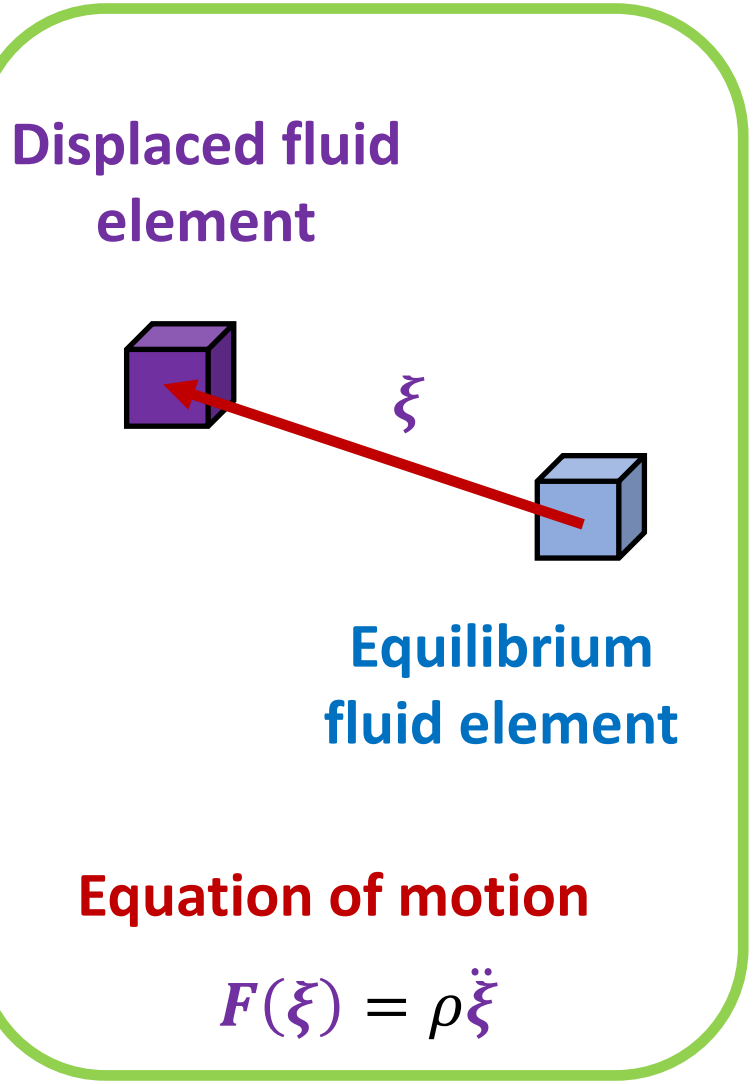
$$0 = (\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0\nabla p$$

- Perturbation with fixed $\iota(\psi)$ and $p(\psi)$ determined from $\boldsymbol{\xi}_1$

$$\delta\boldsymbol{B}_1(\boldsymbol{\xi}_1) = \nabla\times(\boldsymbol{\xi}_1\times\boldsymbol{B})$$
$$\delta p(\boldsymbol{\xi}_1) = -\boldsymbol{\xi}_1\cdot\nabla p$$

- Perturbed equilibrium with specified $\delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}}|_{S_{\text{plasma}}}$ satisfies

$$\boldsymbol{F}(\boldsymbol{\xi}_1) = (\nabla\times\boldsymbol{B})\times\delta\boldsymbol{B}_1 + (\nabla\times\delta\boldsymbol{B}_1)\times\boldsymbol{B} - \mu_0\boldsymbol{\nabla}\delta p(\boldsymbol{\xi}_1) = 0$$
$$\boldsymbol{\xi}_1\cdot\hat{\boldsymbol{n}}\Big|_{S_{\text{plasma}}} = \delta\boldsymbol{x}\cdot\hat{\boldsymbol{n}}\Big|_{S_{\text{plasma}}}$$



Self-adjointness of MHD force operator

Self-adjoint property of MHD

$$\langle \xi_1 , \xi_2 \rangle = \int_{V_{\text{plasma}}} d^3x \, \xi_1 \cdot \xi_2$$

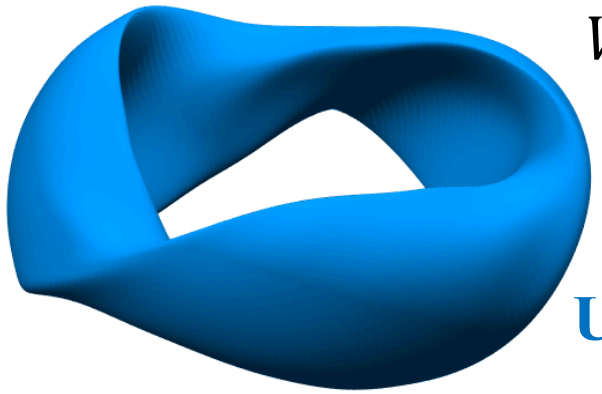
For $\langle \xi, \xi \rangle < \infty$, $\xi \cdot \hat{\boldsymbol{n}} = 0$,

$$\langle \xi_1 , \boldsymbol{F}(\xi_2) \rangle = \langle \boldsymbol{F}(\xi_1), \xi_2 \rangle$$

Variational Principle

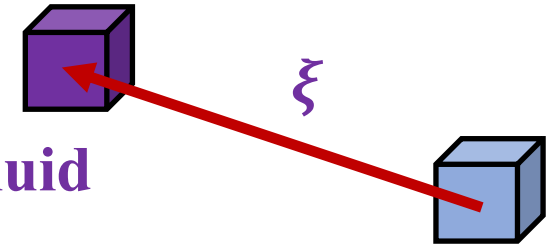
$$L[\xi] = \frac{1}{2}\langle \dot{\xi}, \dot{\xi} \rangle + \frac{1}{2}\langle \boldsymbol{F}(\xi), \xi \rangle$$

$$\delta L[\xi; \delta\xi] = 0 \leftrightarrow \boldsymbol{F}(\xi) = \rho \ddot{\xi}$$



V_{plasma}

Unperturbed boundary



Displaced fluid element

Equilibrium fluid element

Computing MHD shape gradient with adjoint approach

$$\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2) = f(\boldsymbol{B}) + \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_2 \cdot [(\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0 \nabla p\,]$$

1. Shape derivative of figure of merit

$$\delta f\big(S_{\text{plasma}}; \delta\boldsymbol{B}(\boldsymbol{\xi}_1), \delta p(\boldsymbol{\xi}_1)\big) = \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_1 \cdot \boldsymbol{A}_1 + \int_{S_{\text{plasma}}} d^2x\, \hat{\boldsymbol{n}} \cdot \boldsymbol{\xi}_1 A_2$$

Computing MHD shape gradient with adjoint approach

$$\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2) = f(\boldsymbol{B}) + \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_2 \cdot [(\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0 \nabla p\,]$$

1. Shape derivative of figure of merit

$$\delta f\left(S_{\text{plasma}}; \delta\boldsymbol{B}(\boldsymbol{\xi}_1), \delta p(\boldsymbol{\xi}_1)\right) = \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_1 \cdot \boldsymbol{A}_1 + \int_{S_{\text{plasma}}} d^2x\, \hat{\boldsymbol{n}} \cdot \boldsymbol{\xi}_1 A_2$$

2. Solve "adjoint" equation ($\delta\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2; \delta\boldsymbol{B}(\boldsymbol{\xi}_1)) = 0$)

$$\boldsymbol{F}(\boldsymbol{\xi}_2) + \boldsymbol{A}_1 = 0$$

$$\boldsymbol{\xi}_2 \cdot \hat{\boldsymbol{n}}|_{S_{\text{plasma}}} = 0$$

Computing MHD shape gradient with adjoint approach

$$\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2) = f(\boldsymbol{B}) + \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_2 \cdot [(\boldsymbol{\nabla}\times\boldsymbol{B})\times\boldsymbol{B} - \mu_0 \nabla p\,]$$

1. Shape derivative of figure of merit

$$\delta f\big(S_{\text{plasma}}; \delta\boldsymbol{B}(\boldsymbol{\xi}_1), \delta p(\boldsymbol{\xi}_1)\big) = \int_{V_{\text{plasma}}} d^3x\, \boldsymbol{\xi}_1 \cdot \boldsymbol{A}_1 + \int_{S_{\text{plasma}}} d^2x\, \hat{\boldsymbol{n}} \cdot \boldsymbol{\xi}_1 A_2$$

2. Solve "adjoint" equation ($\delta\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2; \delta\boldsymbol{B}(\boldsymbol{\xi}_1)) = 0$)

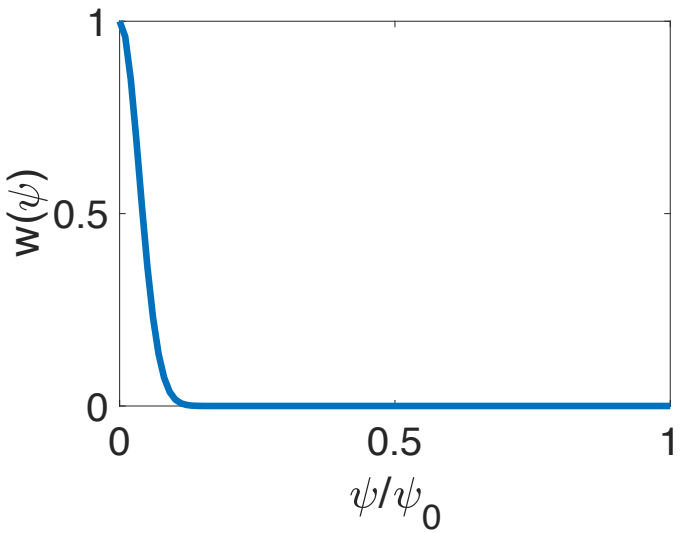
$$\boldsymbol{F}(\boldsymbol{\xi}_2) + \boldsymbol{A}_1 = 0$$
$$\boldsymbol{\xi}_2 \cdot \hat{\boldsymbol{n}}|_{S_{\text{plasma}}} = 0$$

3. Compute shape derivative of $f(\boldsymbol{B})$ *without perturbing* S_{plasma}

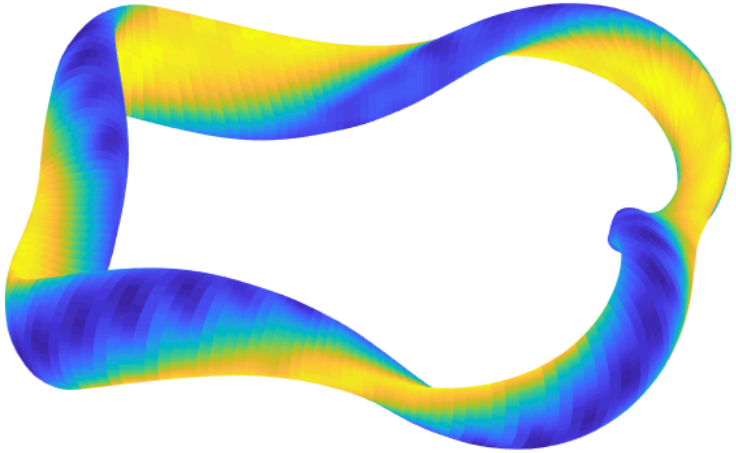
$$\delta f\big(S_{\text{plasma}}; \delta\boldsymbol{x}\big) = \delta\mathcal{L}(\boldsymbol{B}, \boldsymbol{\xi}_2; \delta\boldsymbol{x}) = \int_{S_{\text{plasma}}} d^2x\, \delta\boldsymbol{x} \cdot \hat{\boldsymbol{n}} \left(\frac{\delta\boldsymbol{B}(\boldsymbol{\xi}_2) \cdot \boldsymbol{B}}{\mu_0} + A_2\right)$$

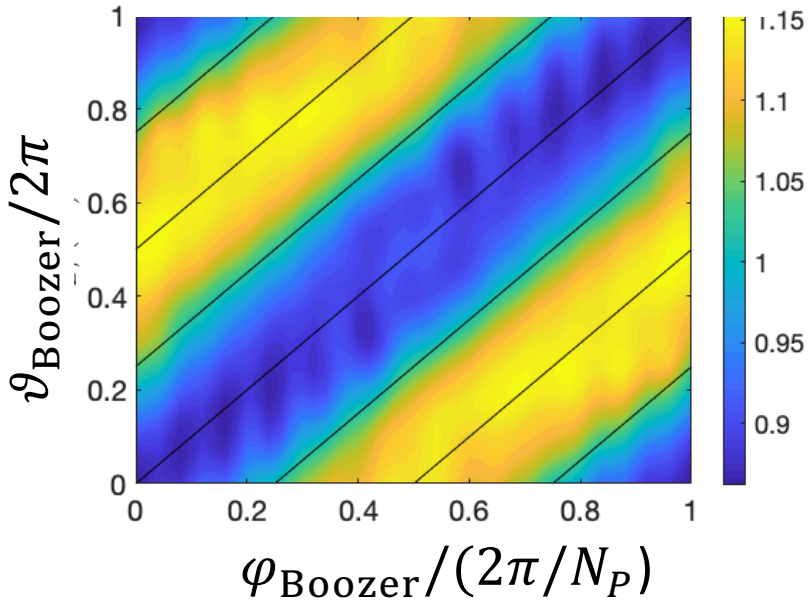
Example: Magnetic ripple shape gradient

$$f_R = \int_{V_{\text{plasma}}} d^3x \; \frac{1}{2} w(\psi)\left(B - \overline{B}\right)^2 \qquad \overline{B} = \frac{\int_{V_{\text{plasma}}} d^3x \, w(\psi) B}{\int_{V_{\text{plasma}}} d^3x \, w(\psi)}$$



HSX (*quasi-symmetric*) field strength





Example: Magnetic ripple shape gradient

$$\boldsymbol{F}(\boldsymbol{\xi}_2) + \nabla \cdot \boldsymbol{P} = 0$$

$$\boldsymbol{P} = p_{||}\, \hat{\boldsymbol{b}}\, \hat{\boldsymbol{b}} + p_{\perp}(\boldsymbol{I} - \hat{\boldsymbol{b}}\, \hat{\boldsymbol{b}})$$

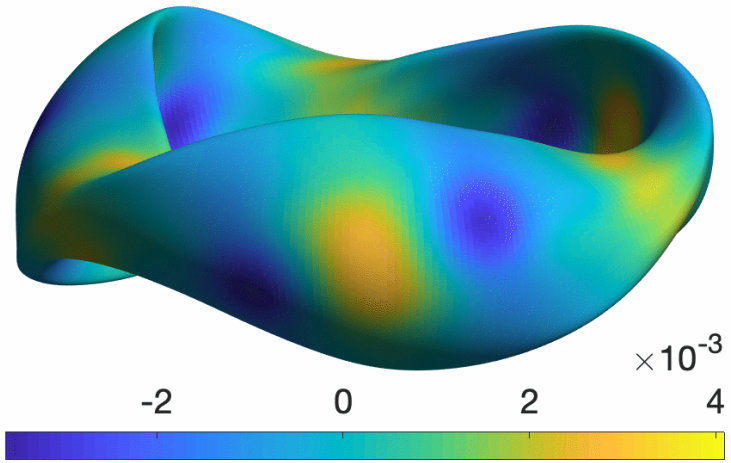
$\approx$

$$0 = \left(\nabla \times \boldsymbol{B}^{(1)}\right) \times \boldsymbol{B}^{(1)} - \mu_0 \nabla\left(p(\psi)\right)$$

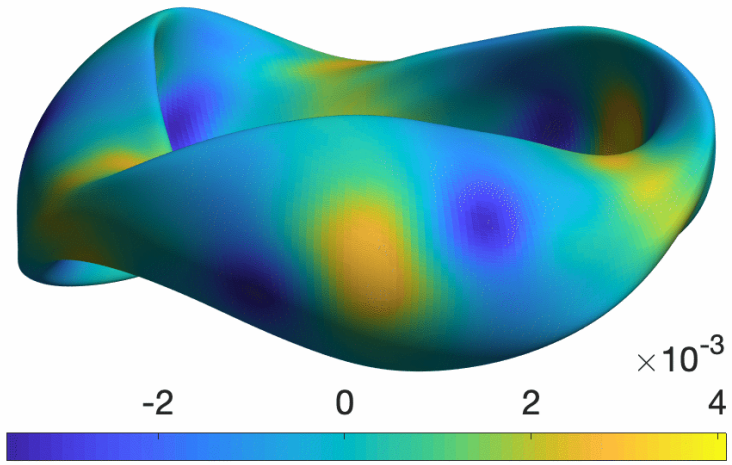
$$0 = \left(\nabla \times \boldsymbol{B}^{(2)}\right) \times \boldsymbol{B}^{(2)} - \mu_0 \nabla\left(p(\psi)\right) - \mu_0 \Delta_P \nabla \cdot \boldsymbol{P}$$

$$\delta \boldsymbol{B} = (\boldsymbol{B}^{(2)} - \boldsymbol{B}^{(1)})/\Delta_P$$

Direct



Adjoint



Example: Magnetic ripple shape gradient

$$\boldsymbol{F}(\boldsymbol{\xi}_2) + \nabla \cdot \boldsymbol{P} = 0$$

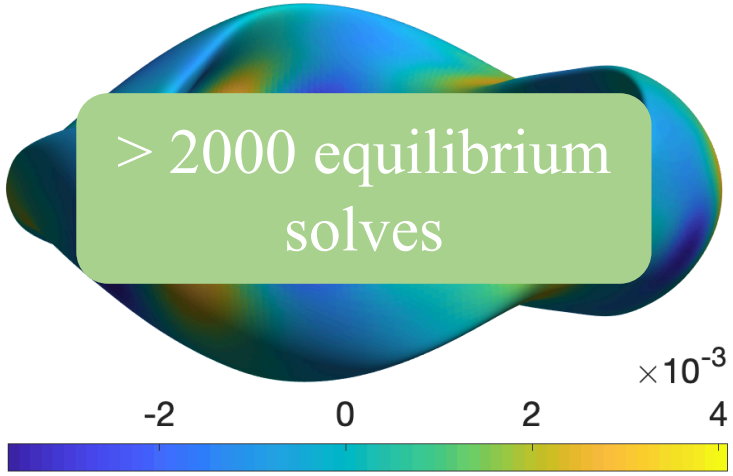
$$\boldsymbol{P} = p_{||}\, \hat{\boldsymbol{b}}\, \hat{\boldsymbol{b}} + p_{\perp}(\boldsymbol{I} - \hat{\boldsymbol{b}}\, \hat{\boldsymbol{b}})$$

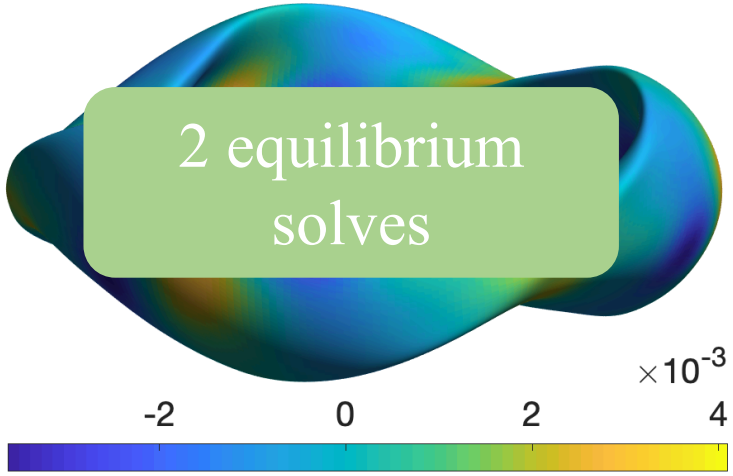
$$\approx$$

$$0 = \left(\nabla \times \boldsymbol{B}^{(1)}\right) \times \boldsymbol{B}^{(1)} - \mu_0 \nabla\left(p(\psi)\right)$$

$$0 = \left(\nabla \times \boldsymbol{B}^{(2)}\right) \times \boldsymbol{B}^{(2)} - \mu_0 \nabla\left(p(\psi)\right) - \mu_0 \Delta_P \nabla \cdot \boldsymbol{P}$$

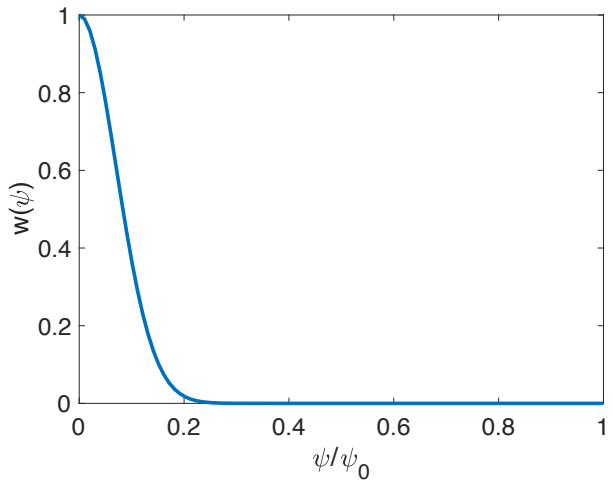
$$\delta \boldsymbol{B} = (\boldsymbol{B}^{(2)} - \boldsymbol{B}^{(1)})/\Delta_P$$



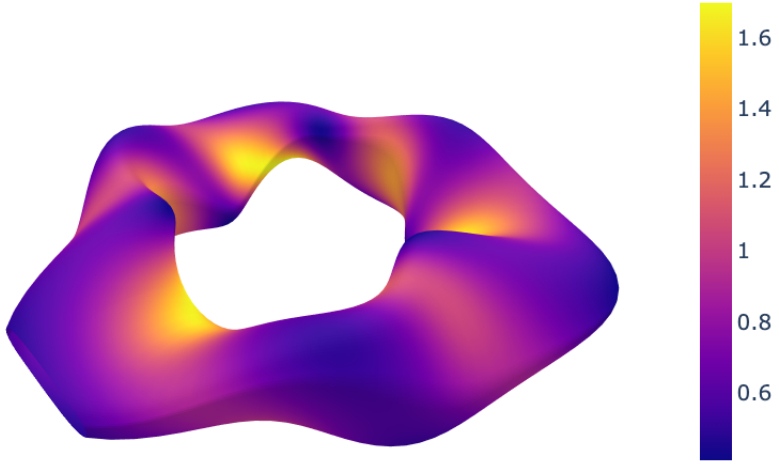


Optimization for near-axis quasisymmetry

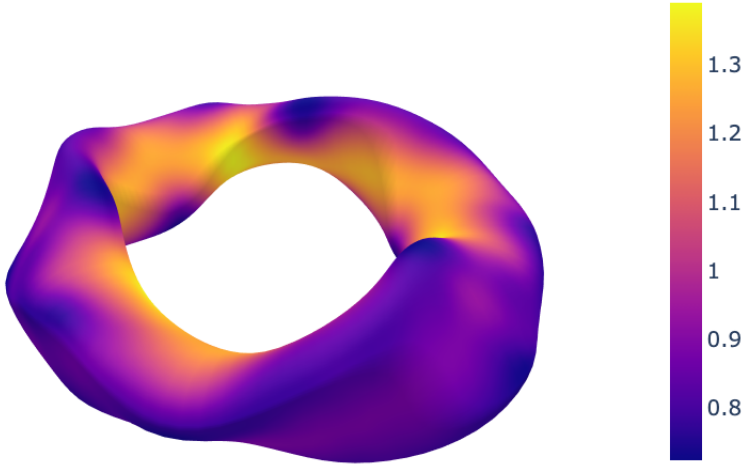
$$f(S_P) \approx \frac{\int_{V_P} d^3x\, w(\psi)(B - \bar{B})^2}{\int_{V_P} d^3x\, w(\psi) B^2}$$

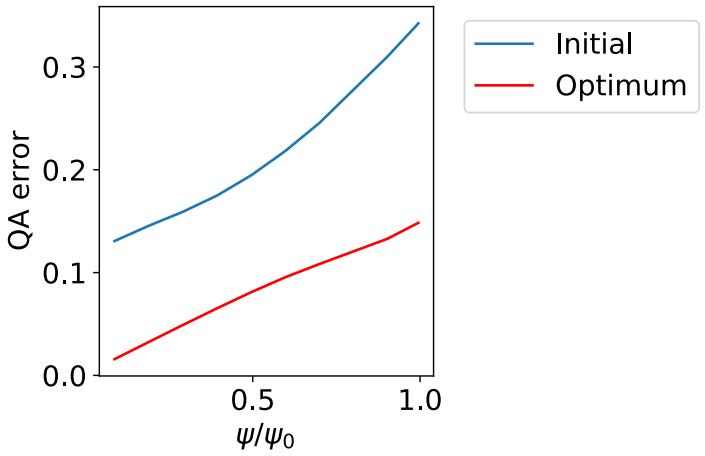


Initial *B*



Optimum *B*





Conclusions

- Adjoint methods are valuable for gradient-based optimization and sensitivity analysis
- Bonus: discretization error correction
- Stellarator applications
  - Current potential methods – discrete approach
    - E. J. Paul et al, Nuclear Fusion 58 (2018).
  - Neoclassical transport – discrete & continuous approaches
    - E. J. Paul et al, J. Plasma Phys. 85 (2019).
  - MHD shape gradients – continuous approach
    - E. J. Paul et al, accepted to J. Plasma Phys. (2021)
    - E. J. Paul et al, J. Plasma Phys. 86 (2020).
    - T. Antonsen et al, J. Plasma Phys. 85 (2019).

Mathematics → physics

Optimization for improved plasma confinement

Physics → mathematics

Insight into development of adjoint methods