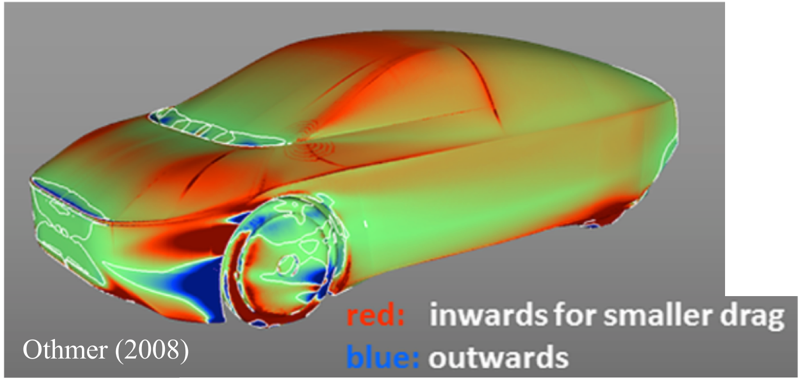
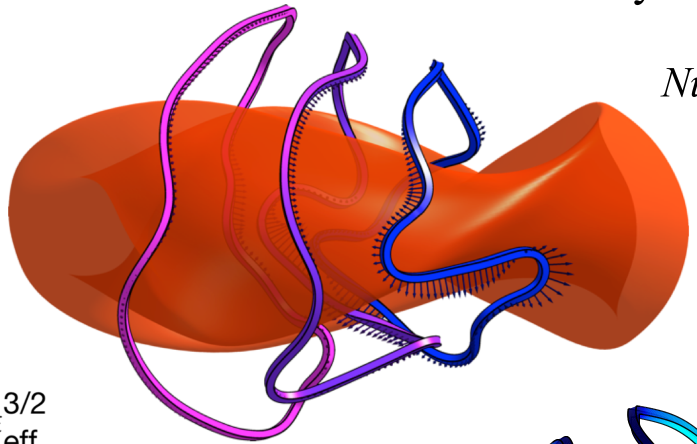
Computing local sensitivity & tolerances of stellarators using shape gradients

Matt Landreman, Elizabeth Paul, Thomas Antonsen Jr

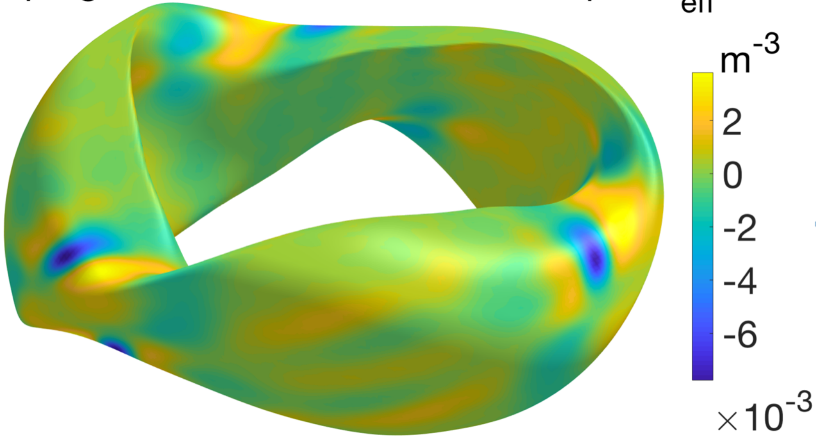
University of Maryland

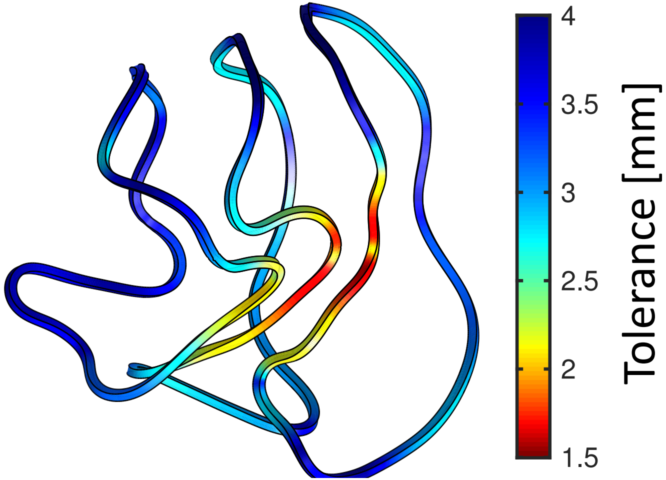
Nuclear Fusion ***58***
076023 (2018)





Shape gradient for neoclassical transport $\epsilon_{\text{eff}}^{3/2}$





The shape gradient is a new (to fusion) way to think about derivatives involving shapes.

- Derivatives involving shapes are central to stellarator optimization.
- These derivatives also encode tolerances, which have been a leading driver of cost:

  “The largest driver of the project cost growth were the accuracy requirements.”

  Strykowsky et al, *Engineering Cost & Schedule Lessons Learned on NCSX,* (2009).

- Compared to ‘parameter derivatives’, shape gradients have 2 advantages:
  - *Spatially local*
  - *Independent of parameterization and discretization*

Outline: Understanding local sensitivity & tolerances of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from existing codes.
- Fast computation of shape gradients via adjoint methods.
- Coil tolerances
- Magnetic sensitivity and tolerances

Historically, we have represented derivatives of shapes using parameter derivatives.

Let f denote any figure of merit, e.g. rotational transform $\iota = 1/q$, neoclassical transport, etc.

Parameter derivatives: Example: $\partial f / \partial R^c_{m,n}$ and $\partial f / \partial Z^s_{m,n}$

where $R^c_{m,n}$ and $Z^s_{m,n}$ parameterize the plasma boundary shape:

$$R(\theta,\zeta) = \sum_{m,n} R^c_{m,n} \cos(m\theta - n\zeta),$$

$$Z(\theta,\zeta) = \sum_{m,n} Z^s_{m,n} \sin(m\theta - n\zeta)$$

Historically, we have represented derivatives of shapes using parameter derivatives.

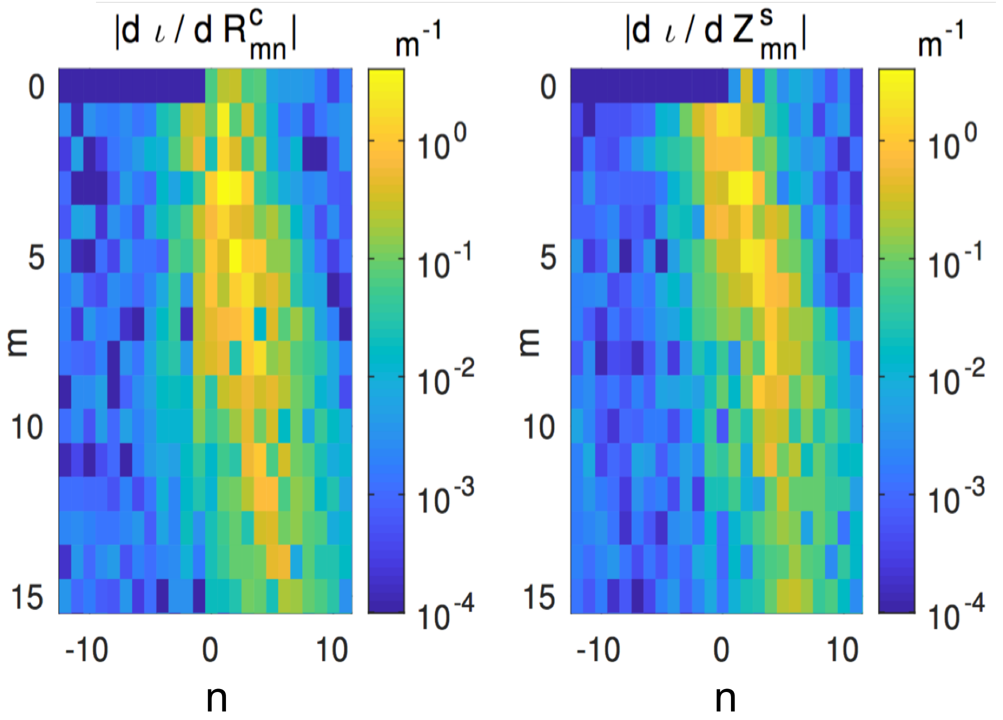
Let f denote any figure of merit, e.g. rotational transform $\iota = 1/q$, neoclassical transport, etc.

Parameter derivatives: Example: $\partial f / \partial R^c_{m,n}$ and $\partial f / \partial Z^s_{m,n}$

where $R^c_{m,n}$ and $Z^s_{m,n}$ parameterize the plasma boundary shape:

$$R(\theta,\zeta) = \sum_{m,n} R^c_{m,n} \cos(m\theta - n\zeta),$$

$$Z(\theta,\zeta) = \sum_{m,n} Z^s_{m,n} \sin(m\theta - n\zeta)$$



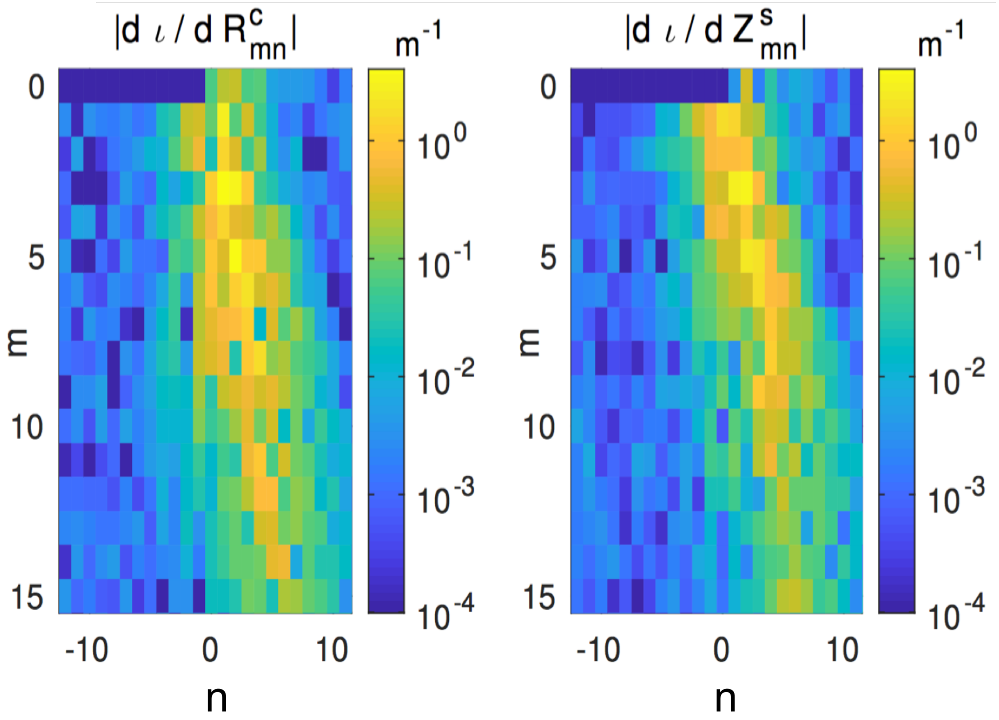
Historically, we have represented derivatives of shapes using parameter derivatives.

Let f denote any figure of merit, e.g. rotational transform $\iota = 1/q$, neoclassical transport, etc.

Parameter derivatives: Example: $\partial f / \partial R^c_{m,n}$ and $\partial f / \partial Z^s_{m,n}$
where $R^c_{m,n}$ and $Z^s_{m,n}$ parameterize the plasma boundary shape:

$$R(\theta,\zeta) = \sum_{m,n} R^c_{m,n} \cos(m\theta - n\zeta),$$

$$Z(\theta,\zeta) = \sum_{m,n} Z^s_{m,n} \sin(m\theta - n\zeta)$$



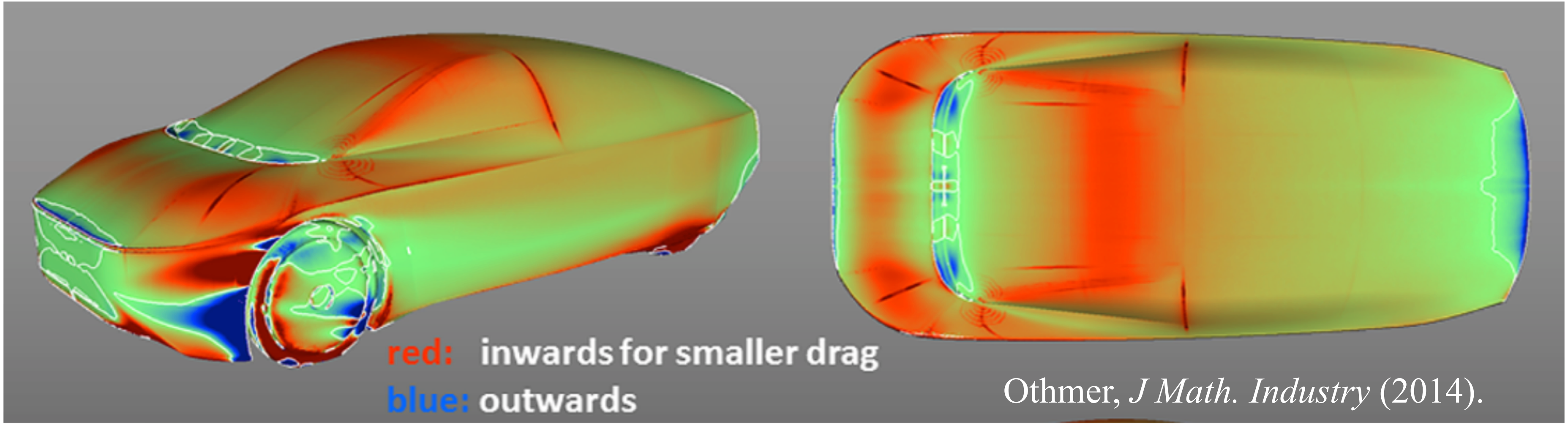
- Successfully used in STELLOPT to design NCSX, etc.
- Computable by finite differencing any code.

But,
- Not unique: coordinate-dependent,
- Nonlocal: awkward for engineering.

The shape gradient is a complementary way to express derivatives involving shapes.

For surfaces, the shape gradient $= S$ where $\delta f = \int d^2 a \left(\delta \mathbf{r} \cdot \mathbf{n} \right) S.$

Unit normal



Othmer, *J Math. Industry* (2014).

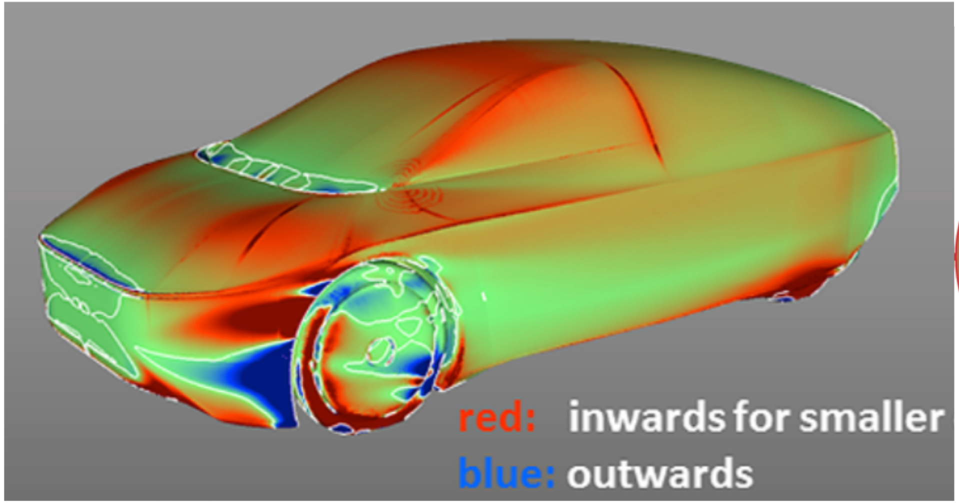
- Local (real-space, not Fourier-space). More useful for engineering.
- Independent of coordinates & discretization used to represent surface.

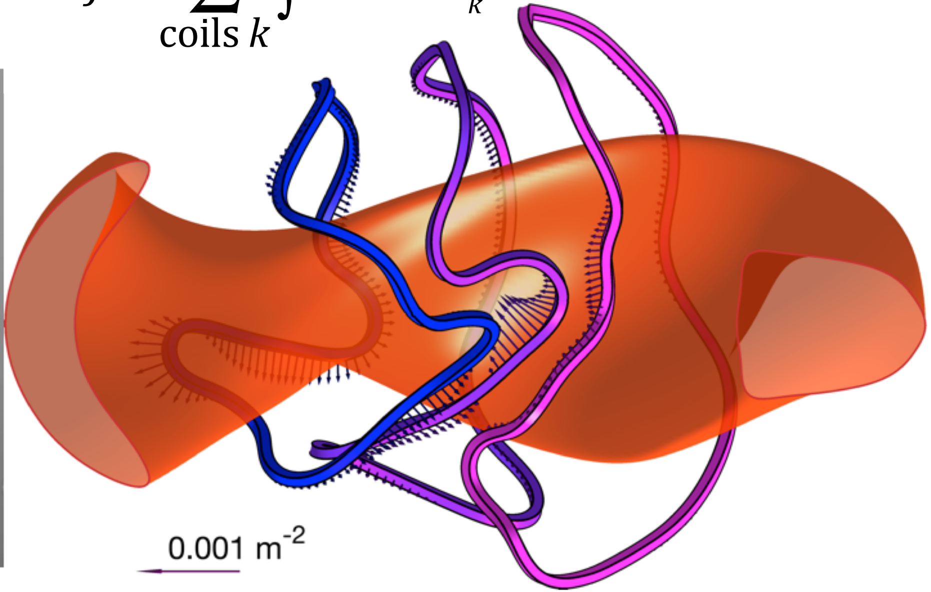
The shape gradient is a complementary way to express derivatives involving shapes.

For surfaces, the shape gradient $= S$ where $\delta f = \int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S.$

Unit normal

For coils, the shape gradient $= \mathbf{S}_k$ where $\delta f = \sum_{\text{coils } k} \int d\ell \; \delta \mathbf{r} \cdot \mathbf{S}_k.$





- Local (real-space, not Fourier-space). More useful for engineering.
- Independent of coordinates & discretization used to represent surface.

The shape gradient representation can be expected to exist for many shape functionals.

Derivative of a function

of n numbers $f\left(r_1, r_2, \ldots, r_n\right)$:

$$\delta f = \sum_{j=1}^{n} \frac{\partial f}{\partial r_j} \delta r_j$$

$n \to \infty$:

$$f = f\left[r\left(\ell\right)\right], \quad \delta f = \int d\ell \underbrace{\frac{\delta f}{\delta r}}_{S} \delta r$$

This is an instance of the "Riesz representation theorem":

Any linear functional can be written as an inner product with some element of the appropriate space.

Outline: Understanding local sensitivity & tolerances of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from existing codes.
- Fast computation of shape gradients via adjoint methods.
- Coil tolerances
- Magnetic sensitivity and tolerances

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

Discretize shape gradient:

$$S_X(\vartheta) = S_{X,0}^c + \sum_{m=1} \left[S_{X,m}^c \cos(m\vartheta) + S_{X,m}^s \sin(m\vartheta) \right] \qquad \& \; S_Y, S_Z$$

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

Discretize shape gradient:

$$S_X(\vartheta) = S_{X,0}^c + \sum_{m=1} \left[S_{X,m}^c \cos(m\vartheta) + S_{X,m}^s \sin(m\vartheta) \right] \qquad \& \; S_Y, S_Z$$

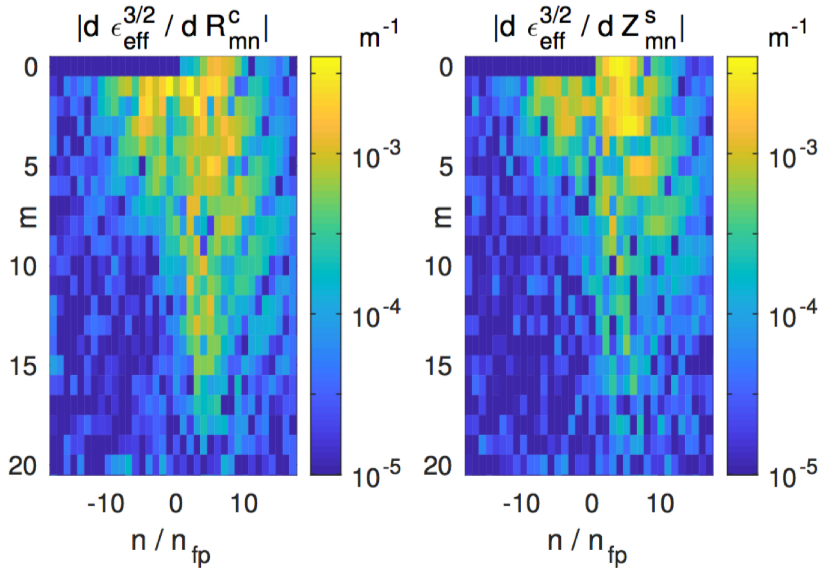
$$\int d\ell \, \delta\mathbf{r} \cdot \mathbf{S} = \delta f \qquad \Rightarrow \qquad \text{Solve } \int d\ell \, \frac{\partial \mathbf{r}}{\partial p_j} \cdot \mathbf{S} = \frac{\partial f}{\partial p_j} \text{ for } \mathbf{S}.$$

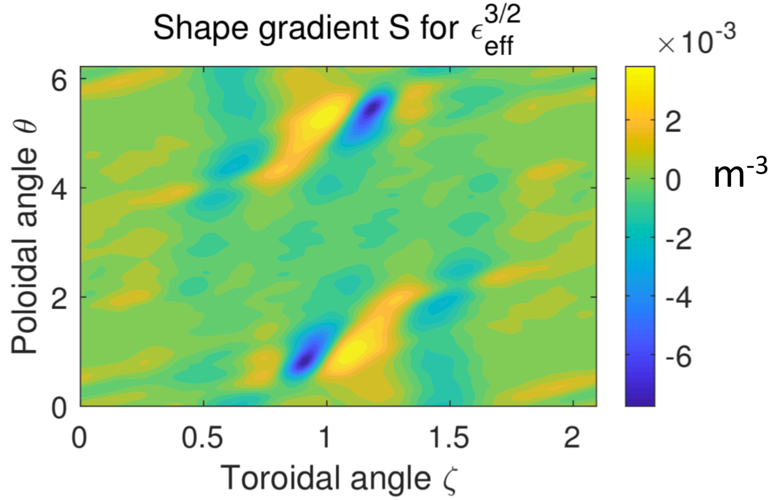
(Square linear system)

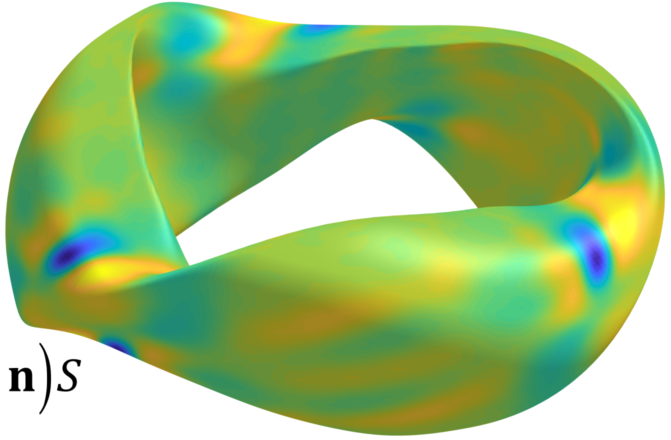
Example: Neoclassical transport $\varepsilon_{eff}^{3/2}$ at r/a=0.5

Shape gradient for boundary surface:

Parameter derivatives from
STELLOPT + VMEC + NEO:



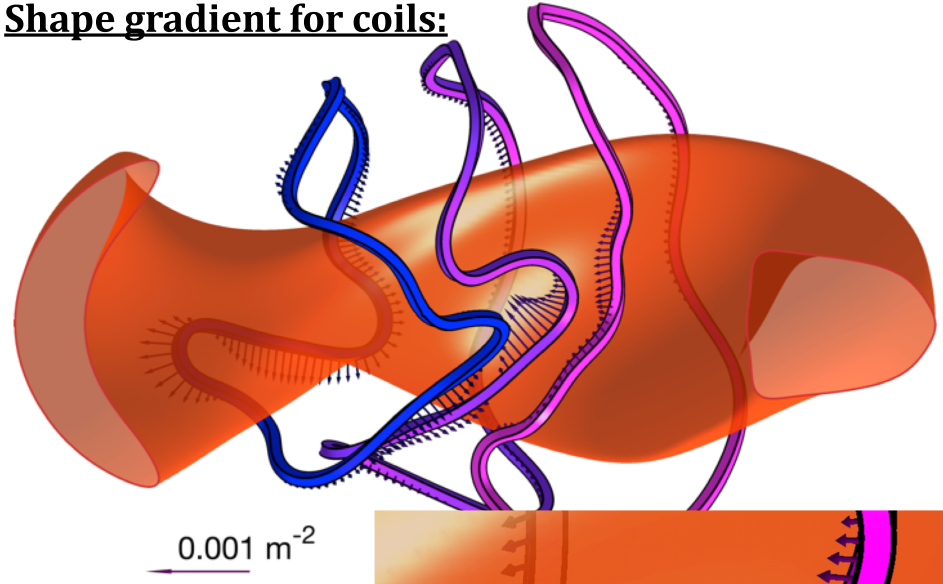


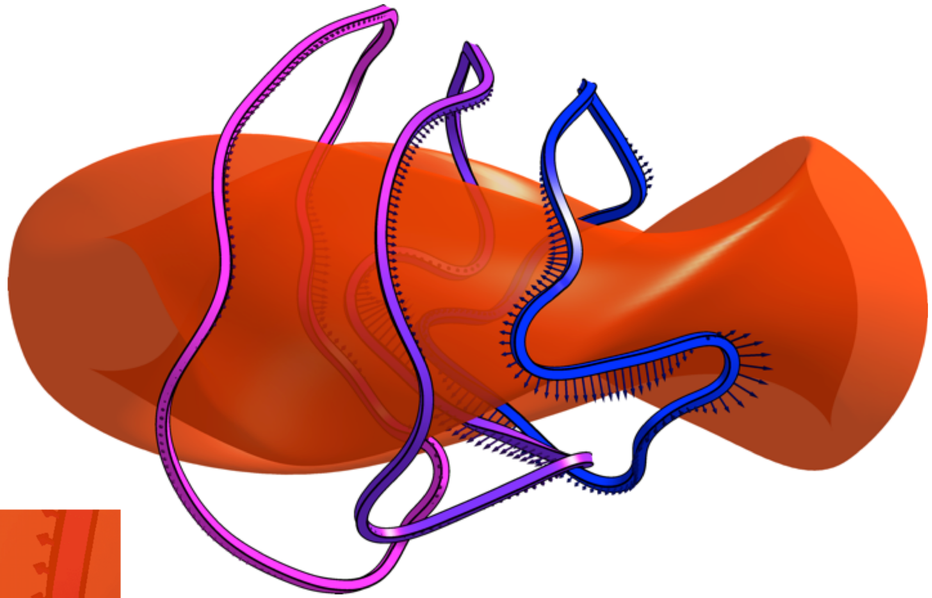


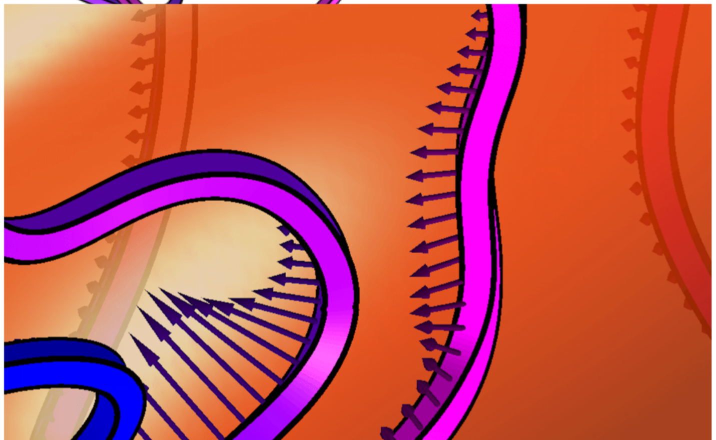
$$\delta f = \int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S$$

Example: Neoclassical transport $\varepsilon_{eff}^{3/2}$ at r/a=0.5

Shape gradient for coils:







Arrows show $\mathbf{S}_k$ where

$$\delta f = \sum_{\text{coils } k} \int d\ell \; \delta\mathbf{r} \cdot \mathbf{S}_k .$$

Outline: Understanding local sensitivity & tolerances of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from existing codes.
- Fast computation of shape gradients via adjoint methods.
- Coil tolerances
- Magnetic sensitivity and tolerances

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

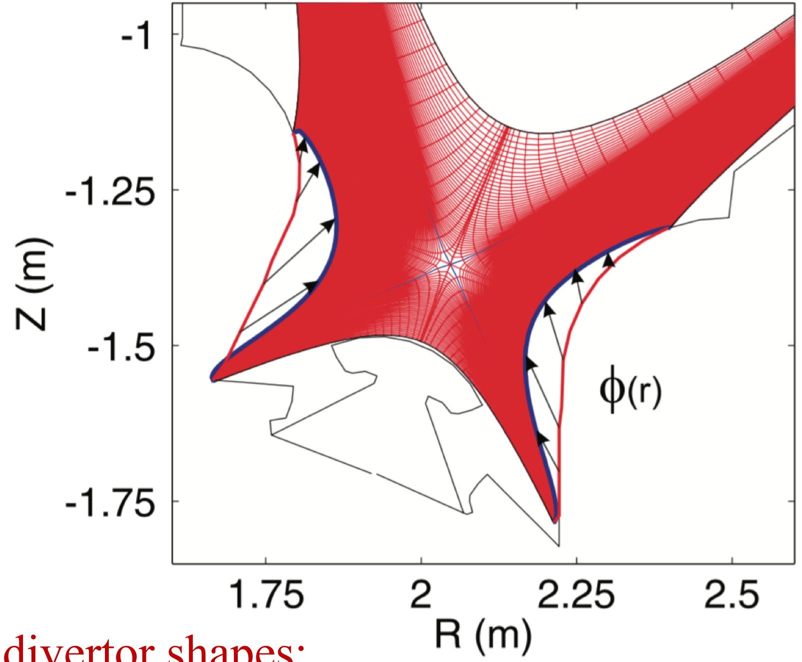
Adjoint methods:

- You can get the derivative of a code result with respect to all N parameters with only 1 (not N) extra calculation.
- Requires some theory work and code modifications.

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

<u>Adjoint methods:</u>

- You can get the derivative of a code result with respect to <u>all</u> *N* parameters with only 1 (not *N*) extra calculation.
- Requires some theory work and code modifications.



Recently used for optimizing tokamak divertor shapes:

- W. Dekeyser, Ph.D. thesis, KU Leuven (2014).
- W. Dekeyser et al, *Nucl. Fusion* 54, 073022 (2014).
- M. Baelmans, et al, *Nucl. Fusion* 57, 036022 (2017).

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

Antonsen, Paul, & ML, PO5.00005 Wed 2:48pm

Self-adjointness of linearized MHD:

$$\int_\Omega \left(\xi^{(2)} \cdot \mathbf{F}^{(1)} - \xi^{(1)} \cdot \mathbf{F}^{(2)} \right) = \int_{\partial\Omega} \left[\left(\mathbf{n} \cdot \xi^{(1)} \right) \left(\mathbf{B} \cdot \delta\mathbf{B}^{(2)} \right) - \left(\mathbf{n} \cdot \xi^{(2)} \right) \left(\mathbf{B} \cdot \delta\mathbf{B}^{(1)} \right) \right]$$

where $\mathbf{F}^{(j)} = \mathbf{J}^{(j)} \times \mathbf{B} + \mathbf{J} \times \mathbf{B}^{(j)} - \nabla p^{(j)}$, $\quad \mathbf{B}^{(j)} = \nabla \times \left(\xi^{(j)} \times \mathbf{B} \right)$, $\quad \mu_0 \mathbf{J}^{(j)} = \nabla \times \mathbf{B}^{(j)}$, $\quad p^{(j)} + \xi^{(j)} \cdot \nabla p = 0.$

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

Antonsen, Paul, & ML, PO5.00005 Wed 2:48pm

Self-adjointness of linearized MHD:

$$\int_{\Omega}\left(\xi^{(2)}\cdot\mathbf{F}^{(1)}-\xi^{(1)}\cdot\mathbf{F}^{(2)}\right)=\int_{\partial\Omega}\left[\left(\mathbf{n}\cdot\xi^{(1)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right)-\left(\mathbf{n}\cdot\xi^{(2)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(1)}\right)\right]$$

where $\mathbf{F}^{(j)}=\mathbf{J}^{(j)}\times\mathbf{B}+\mathbf{J}\times\mathbf{B}^{(j)}-\nabla p^{(j)}$, $\quad \mathbf{B}^{(j)}=\nabla\times\left(\xi^{(j)}\times\mathbf{B}\right)$, $\quad \mu_0\mathbf{J}^{(j)}=\nabla\times\mathbf{B}^{(j)}$, $\quad p^{(j)}+\xi^{(j)}\cdot\nabla p=0.$

`Real' perturbation $\xi^{(1)}$: $\mathbf{n}\cdot\xi^{(2)}\Big|_{\partial\Omega}\neq 0,\quad \mathbf{F}^{(1)}=0.$

Adjoint perturbation $\xi^{(2)}$: $\mathbf{n}\cdot\xi^{(2)}\Big|_{\partial\Omega}=0,\quad \mathbf{F}^{(2)}\neq 0.$

$$\Rightarrow\quad \int_{\Omega}\xi^{(1)}\cdot\mathbf{F}^{(2)}=\int_{\partial\Omega}\left(\mathbf{n}\cdot\xi^{(1)}\right)\underbrace{\left(-\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right)}_{\text{Shape gradient}}$$

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

Antonsen, Paul, & ML, PO5.00005 Wed 2:48pm

Self-adjointness of linearized MHD:

$$\int_{\Omega}\left(\xi^{(2)}\cdot\mathbf{F}^{(1)}-\xi^{(1)}\cdot\mathbf{F}^{(2)}\right)=\int_{\partial\Omega}\left[\left(\mathbf{n}\cdot\xi^{(1)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right)-\left(\mathbf{n}\cdot\xi^{(2)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(1)}\right)\right]$$

where $\mathbf{F}^{(j)}=\mathbf{J}^{(j)}\times\mathbf{B}+\mathbf{J}\times\mathbf{B}^{(j)}-\nabla p^{(j)}, \qquad \mathbf{B}^{(j)}=\nabla\times\left(\xi^{(j)}\times\mathbf{B}\right), \qquad \mu_0\mathbf{J}^{(j)}=\nabla\times\mathbf{B}^{(j)}, \qquad p^{(j)}+\xi^{(j)}\cdot\nabla p=0.$

\`Real' perturbation $\xi^{(1)}$: $\left.\mathbf{n}\cdot\xi^{(2)}\right|_{\partial\Omega}\neq 0, \quad \mathbf{F}^{(1)}=0.$

Adjoint perturbation $\xi^{(2)}$: $\left.\mathbf{n}\cdot\xi^{(2)}\right|_{\partial\Omega}=0, \quad \mathbf{F}^{(2)}\neq 0.$

$$\Rightarrow \int_{\Omega}\xi^{(1)}\cdot\mathbf{F}^{(2)}=\int_{\partial\Omega}\left(\mathbf{n}\cdot\xi^{(1)}\right)\underbrace{\left(-\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right)}_{\text{Shape gradient}}$$

$\Rightarrow$ If perturbations to a figure of merit f can be written $\delta f=\int_{\Omega}\xi\cdot\left(\text{something}\right)$, adjoint calculation is given by perturbing the equilibrium by the "something".

For certain figures of merit, shape gradients can be computed extremely efficiently using adjoint methods.

Antonsen, Paul, & ML, PO5.00005 Wed 2:48pm

Self-adjointness of linearized MHD:

$$\int_\Omega \left(\xi^{(2)} \cdot \mathbf{F}^{(1)} - \xi^{(1)} \cdot \mathbf{F}^{(2)}\right) = \int_{\partial\Omega} \left[\left(\mathbf{n}\cdot\xi^{(1)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right) - \left(\mathbf{n}\cdot\xi^{(2)}\right)\left(\mathbf{B}\cdot\delta\mathbf{B}^{(1)}\right)\right]$$

where $\mathbf{F}^{(j)} = \mathbf{J}^{(j)}\times\mathbf{B} + \mathbf{J}\times\mathbf{B}^{(j)} - \nabla p^{(j)}$, $\quad \mathbf{B}^{(j)} = \nabla\times\left(\xi^{(j)}\times\mathbf{B}\right)$, $\quad \mu_0 \mathbf{J}^{(j)} = \nabla\times\mathbf{B}^{(j)}$, $\quad p^{(j)} + \xi^{(j)}\cdot\nabla p = 0.$

`Real' perturbation $\xi^{(1)}$: $\left.\mathbf{n}\cdot\xi^{(2)}\right|_{\partial\Omega} \neq 0, \quad \mathbf{F}^{(1)} = 0.$

Adjoint perturbation $\xi^{(2)}$: $\left.\mathbf{n}\cdot\xi^{(2)}\right|_{\partial\Omega} = 0, \quad \mathbf{F}^{(2)} \neq 0.$

$$\Rightarrow \int_\Omega \xi^{(1)}\cdot\mathbf{F}^{(2)} = \int_{\partial\Omega}\left(\mathbf{n}\cdot\xi^{(1)}\right)\underbrace{\left(-\mathbf{B}\cdot\delta\mathbf{B}^{(2)}\right)}_{\text{Shape gradient}}$$

$\Rightarrow$ If perturbations to a figure of merit f can be written $\delta f = \int_\Omega \xi\cdot\left(\text{something}\right)$, adjoint calculation is given by perturbing the equilibrium by the "something".

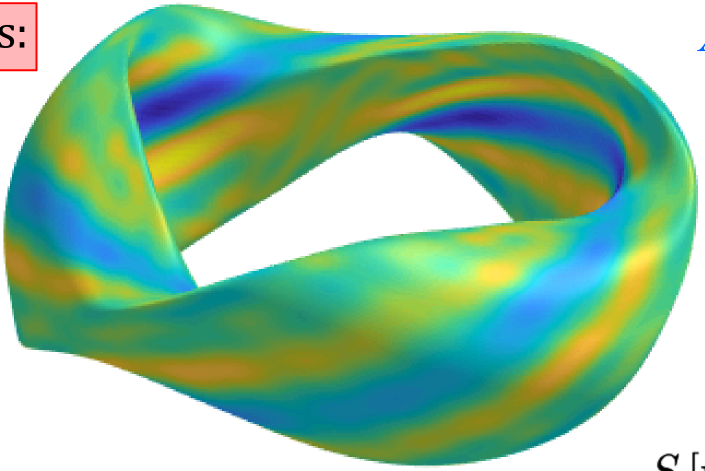
Can be generalized to include perturbations that change ι, & vacuum region + coils.

Adjoint calculations for several figures of merit have now been demonstrated.

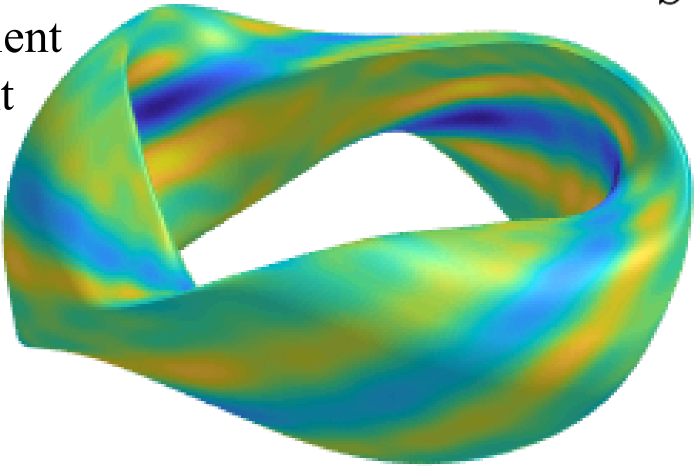
$\boxed{f = \iota \text{ on axis:}}$

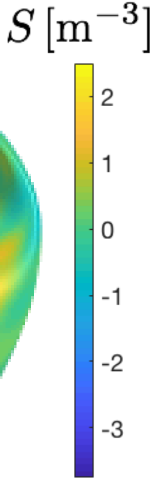
Antonsen, Paul, & ML, PO5.00005 Wed 2:48pm

Shape gradient from finite differences:



Shape gradient from adjoint method:



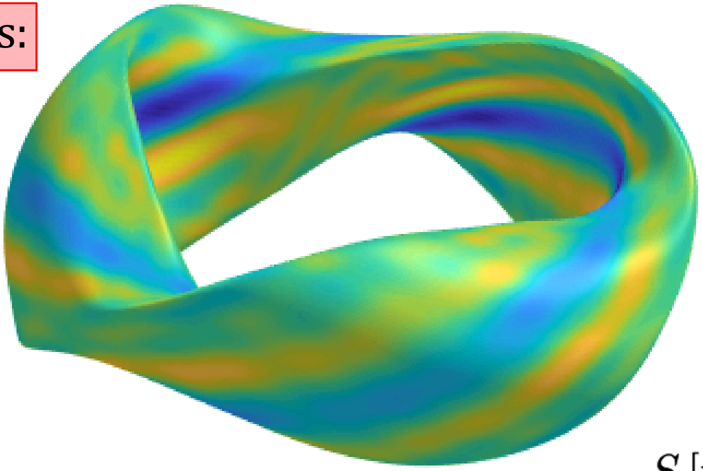


Only 2 VMEC calls!!

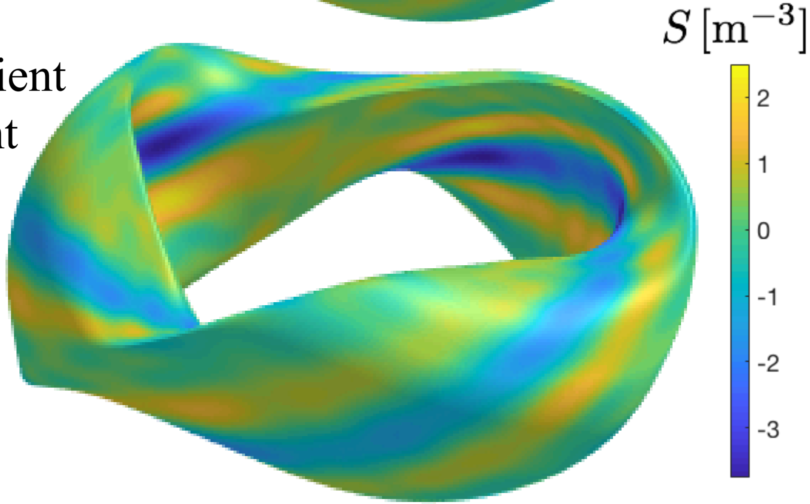
Adjoint calculations for several figures of merit have now been demonstrated.

$f = \iota$ on axis:

Shape gradient from finite differences:

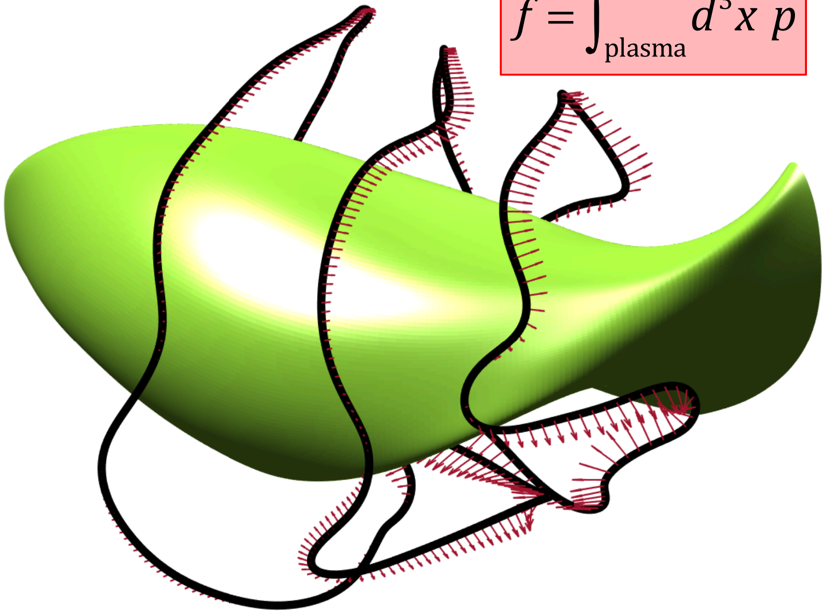


Shape gradient from adjoint method:



Only 2 VMEC calls!!

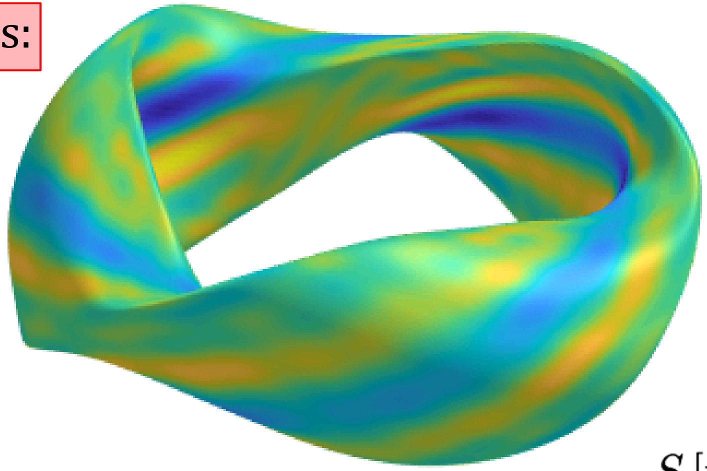
$$f = \int_{\text{plasma}} d^3x\ p$$



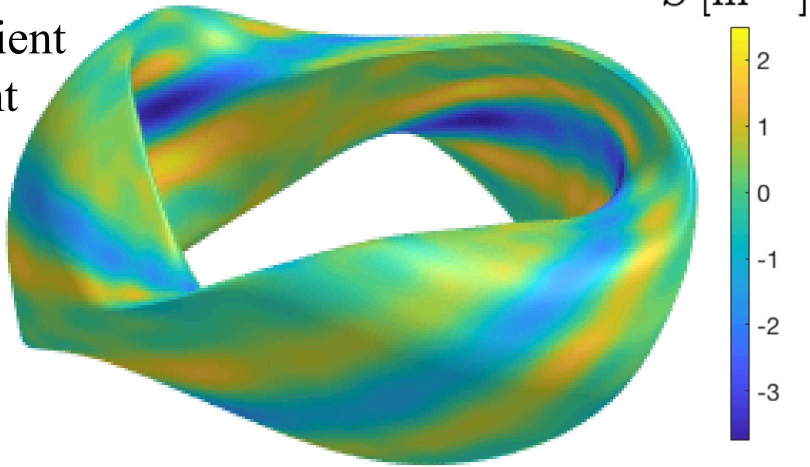
Adjoint calculations for several figures of merit have now been demonstrated.

$f = \iota$ on axis:

Shape gradient from finite differences:

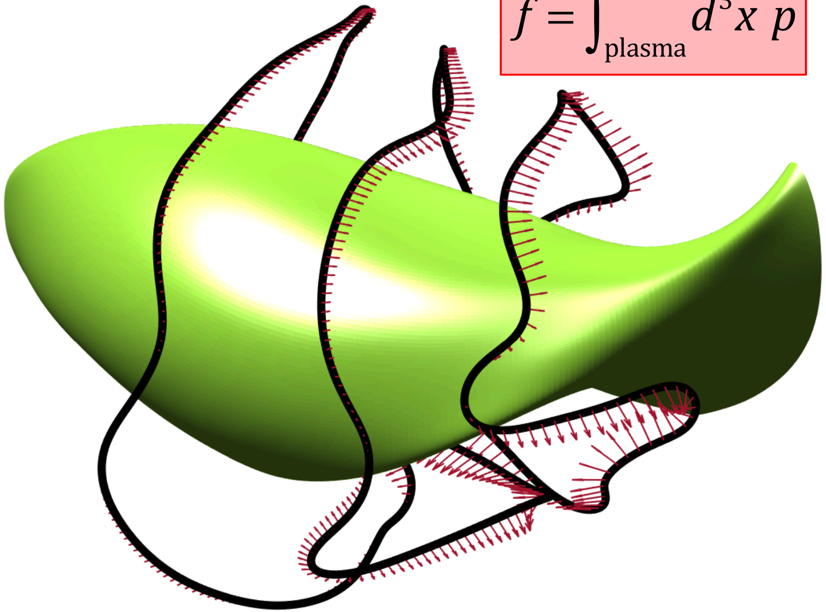


Shape gradient from adjoint method:



Only 2 VMEC calls!!

$$f = \int_{\text{plasma}} d^3x\, p$$



If we could include an arbitrary pressure tensor in VMEC or a perturbed-MHD code, we could use an adjoint to quickly compute the shape gradient of ε_{eff}.

Outline: Understanding local sensitivity & tolerances of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from existing codes.
- Fast computation of shape gradients via adjoint methods.
- Coil tolerances
- Magnetic sensitivity and tolerances

Coil tolerances can be computed from the shape gradient.

Choose an acceptable Δf & any weight $w(\ell) \geq 0$.

$$\text{Let } T(\ell) = \frac{w\,\Delta f}{\sum \int d\ell\, w|\mathbf{S}|}.$$

If $|\delta \mathbf{r}| \leq T$,

$$|\delta f| \leq \int d\ell |\mathbf{S} \cdot \delta \mathbf{r}| \leq \int d\ell |\mathbf{S}||\delta \mathbf{r}| \leq \int d\ell |\mathbf{S}| T = \Delta f.$$

Coil tolerances can be computed from the shape gradient.

Choose an acceptable Δf & any weight $w(\ell) \geq 0$.

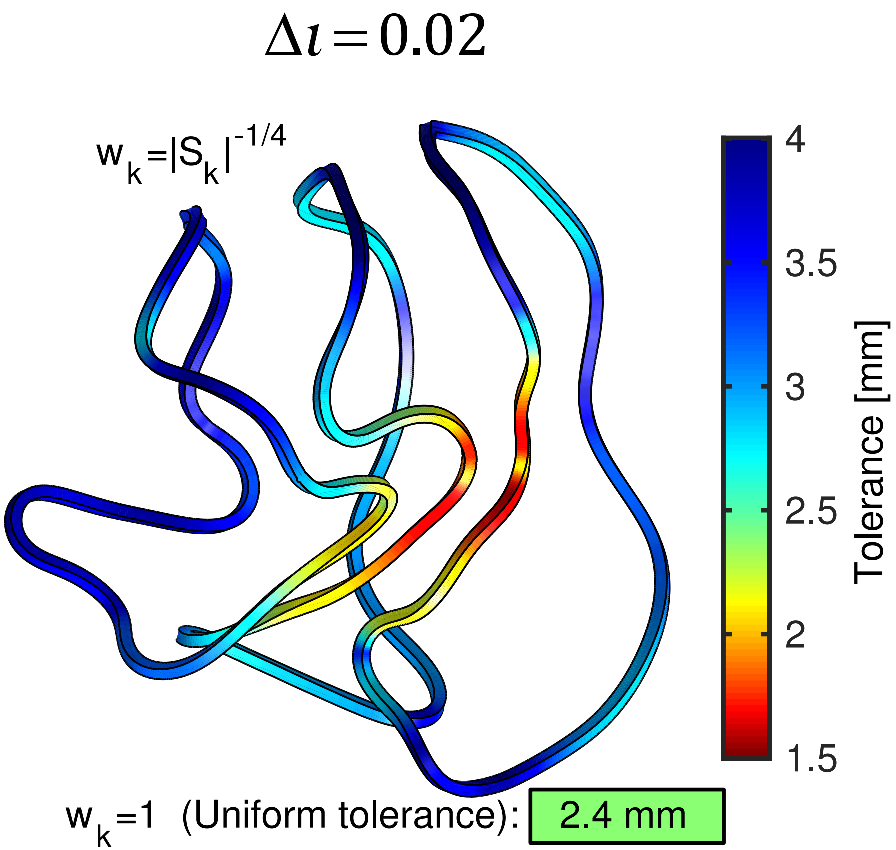
$$\text{Let } T(\ell) = \frac{w\,\Delta f}{\sum \int d\ell\, w|\mathbf{S}|}.$$

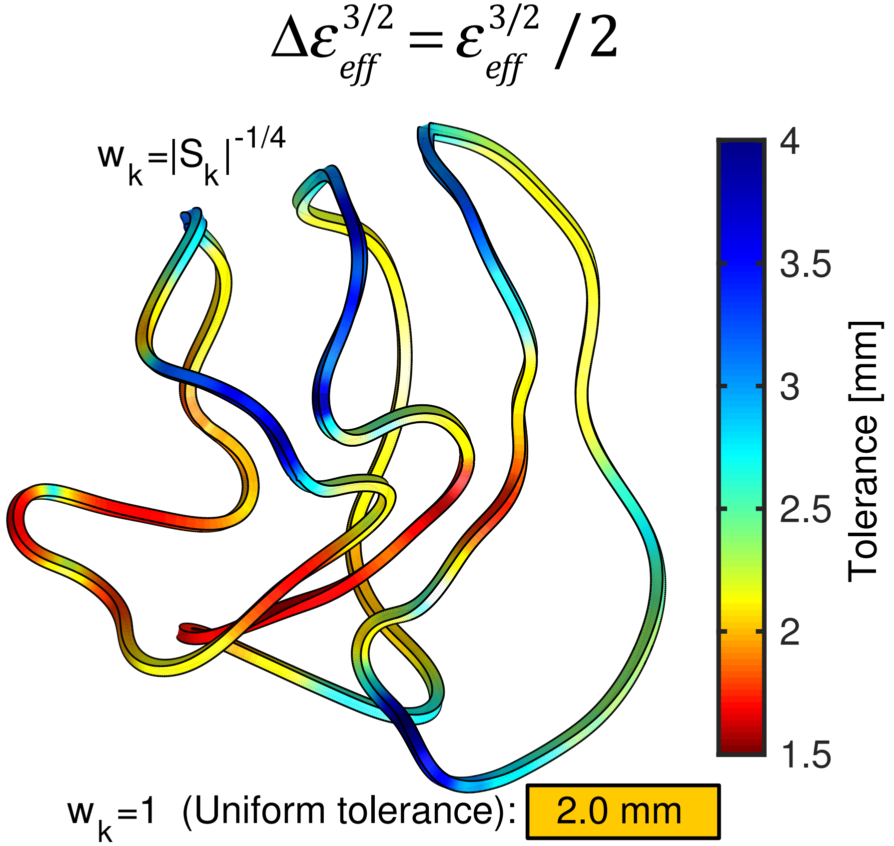
If $|\delta \mathbf{r}| \leq T$,

$$|\delta f| \leq \int d\ell |\mathbf{S}\cdot\delta\mathbf{r}| \leq \int d\ell |\mathbf{S}||\delta\mathbf{r}| \leq \int d\ell |\mathbf{S}| T = \Delta f.$$

Conservative: a bound on the worst possible outcome.

Coil tolerances can be computed from the shape gradient.





Outline: Understanding local sensitivity & tolerances of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from existing codes.
- Fast computation of shape gradients via adjoint methods.
- Coil tolerances
- Magnetic sensitivity and tolerances

A magnetic sensitivity S_B can be computed from the shape gradient.

Define S_B by $\mathbf{B}_0 \cdot \nabla S_B = \langle S \rangle - S$.

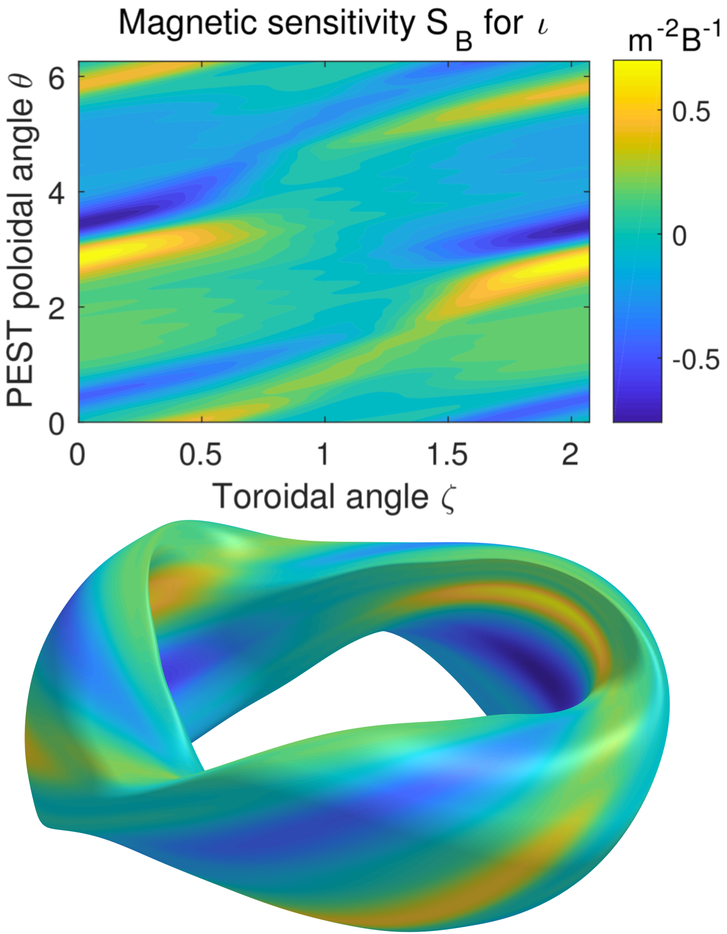
Substitute into $\delta f = \int d^2 a \, S \delta \mathbf{r} \cdot \mathbf{n}$.

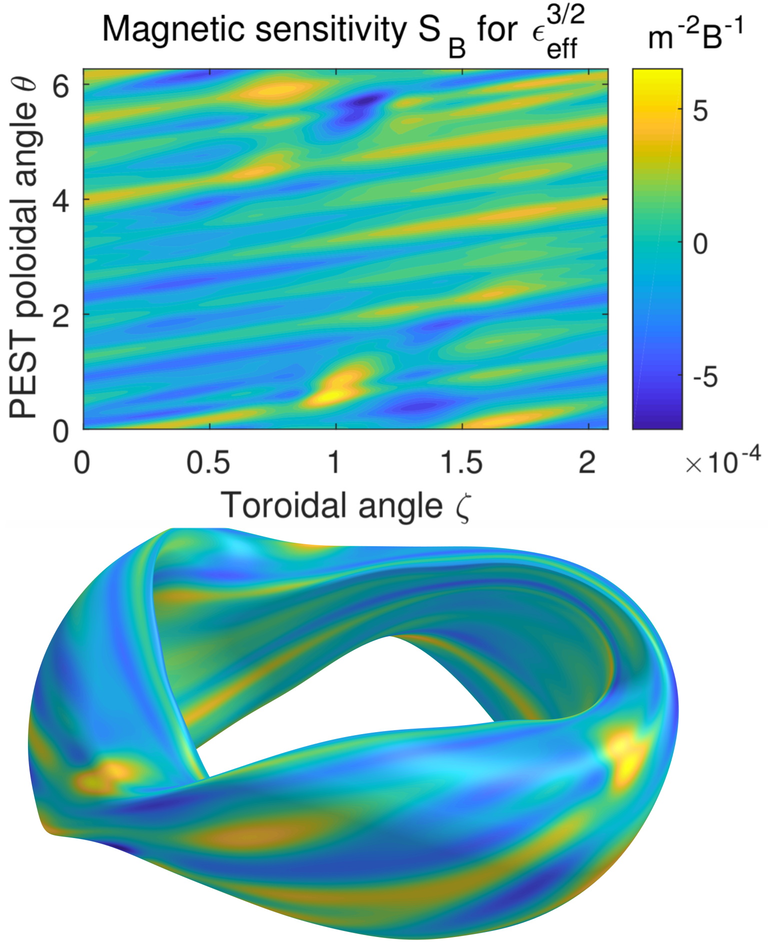
After some algebra …

$$\Rightarrow \quad \delta f = \int d^2 a \, S_B \delta \mathbf{B} \cdot \mathbf{n}.$$

A magnetic sensitivity S_B can be computed from the shape gradient.

$$\delta f = \int d^2a\, S_B \delta \mathbf{B}\cdot\mathbf{n}$$





A magnetic tolerance T_B can be computed from the magnetic sensitivity.

Choose an acceptable Δf & any weight $W\left(\theta,\zeta\right)\geq 0.$

$$\text{Let } T_B\left(\theta,\zeta\right)=\frac{W\,\Delta f}{\int d^2a\; W\left|S_B\right|}.$$

$$\text{If } \left|\delta\mathbf{B}\cdot\mathbf{n}\right|\leq T_B,$$

$$\begin{aligned}\left|\delta f\right| &\leq \int d^2a\left|S_B\right|\left|\delta\mathbf{B}\cdot\mathbf{n}\right| \\ &\leq \int d^2a\left|S_B\right|T_B \\ &\leq \Delta f.\end{aligned}$$

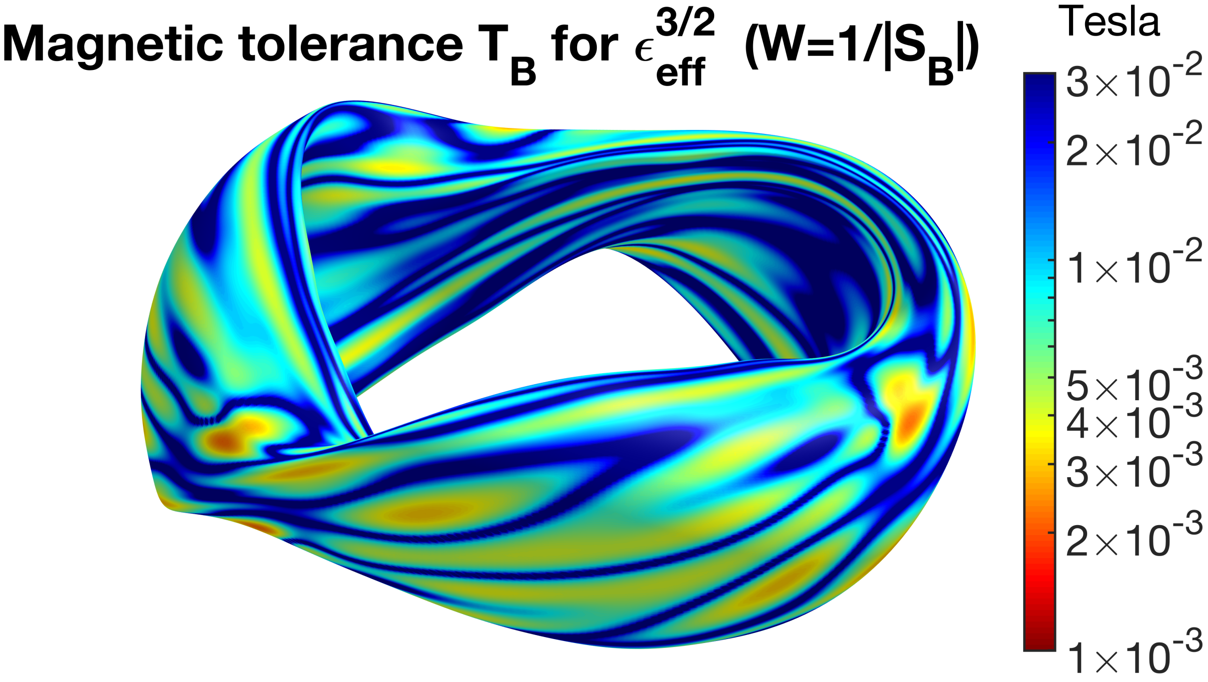
A magnetic tolerance T_B can be computed from the magnetic sensitivity.

Choose an acceptable Δf & any weight $W(\theta,\zeta)\geq 0$.

Let $T_B(\theta,\zeta)=\dfrac{W\,\Delta f}{\int d^2a\, W\left|S_B\right|}$.

If $\left|\delta\mathbf{B}\cdot\mathbf{n}\right|\leq T_B$,

$$\begin{aligned}\left|\delta f\right| &\leq \int d^2a\left|S_B\right|\left|\delta\mathbf{B}\cdot\mathbf{n}\right|\\ &\leq \int d^2a\left|S_B\right|T_B\\ &\leq \Delta f.\end{aligned}$$



W=1 (Alternative uniform tolerance): 6.1×10^{-3} T

Conclusions

Shape gradients provide ***local*** sensitivity & tolerance information which could inform

- How accurately and rigidly the coils should be built,
- Where coils should be connected to support structure,
- Where sources of error fields like coil leads should be located.

<u>Future work:</u>

- Shape gradients for island width.
- Develop adjoint methods for more figures of merit. Need 3D equilibrium or stability code with arbitrary pressure anisotropy.
- Target tolerances in STELLOPT to increase them.

Landreman & Paul, Nuclear Fusion ***58*** *076023 (2018),*

Antonsen, Paul, & Landreman, PO5.00005 Wed 2:48pm

Extra slides

For some shape functionals, the shape gradient can be computed analytically.

Example: Given $\mathbf{B}(\mathbf{r})$, vary surface to minimize $f = \underbrace{\frac{1}{2}\left[\int d^2a\left(\mathbf{B}\cdot\mathbf{n}\right)^2\right]}_{\text{"Quadratic flux" [1]}} + \lambda \underbrace{\int d^3x}_{\text{Volume}}$

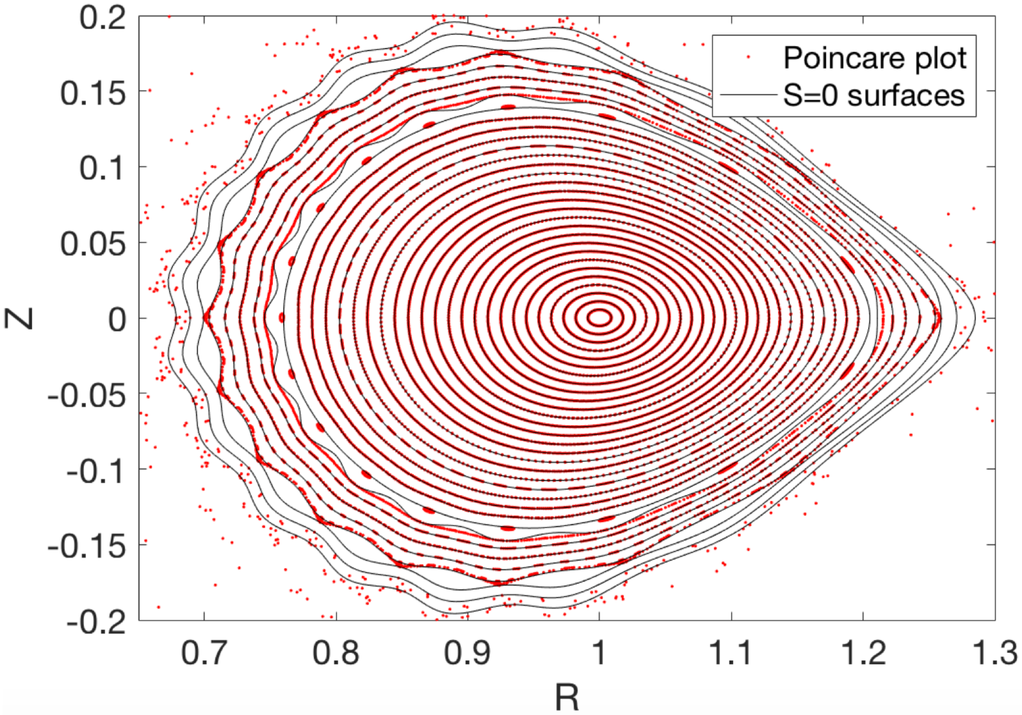
Perturb position vector $\mathbf{r}(\theta,\zeta)$.

After some algebra...

$$\delta f = \int d^2a\, S\, \delta\mathbf{r}\cdot\mathbf{n}$$

where $S = \left(\mathbf{B}\cdot\mathbf{n}\right)^2 H + \mathbf{B}\cdot\nabla\left(\mathbf{B}\cdot\mathbf{n}\right) + \lambda$,

$H =$ mean curvature.

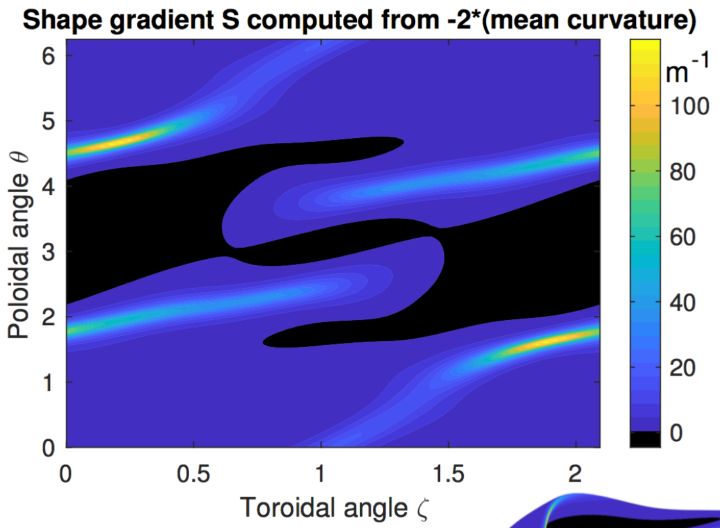


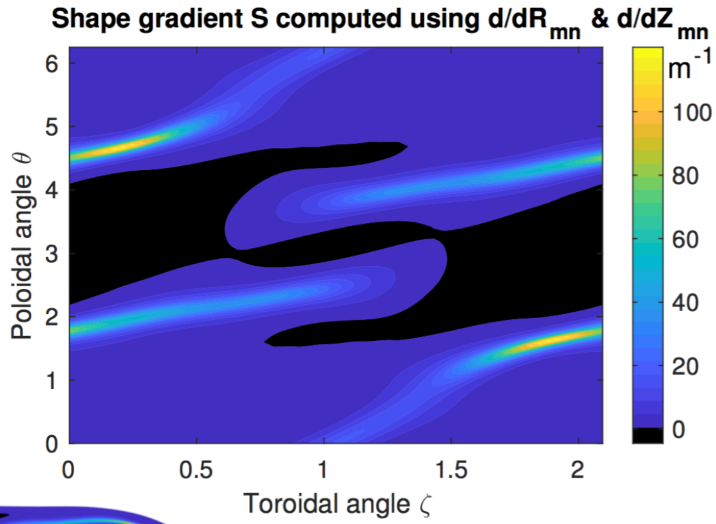
$S = 0$ at the optimum.

[1] Dewar, Hudson, Price (1994).

The algorithm for computing shape gradients can be verified by comparison to analytic theory.

Consider $f = \text{area.}$ Analytic result: $S = -2 \times \left(\text{mean curvature}\right)$



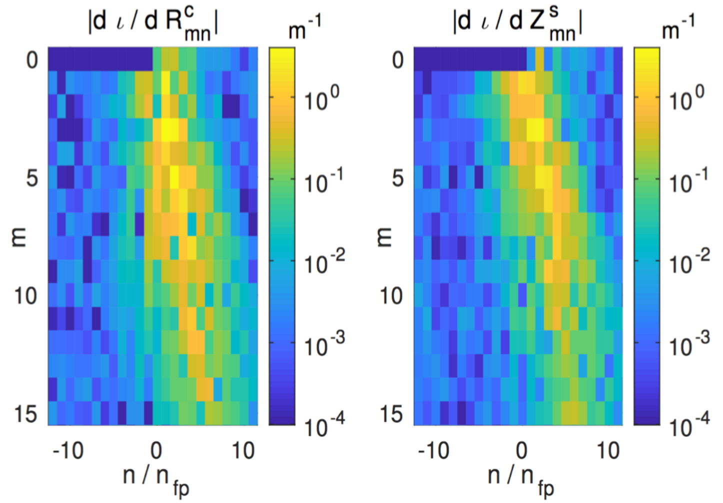


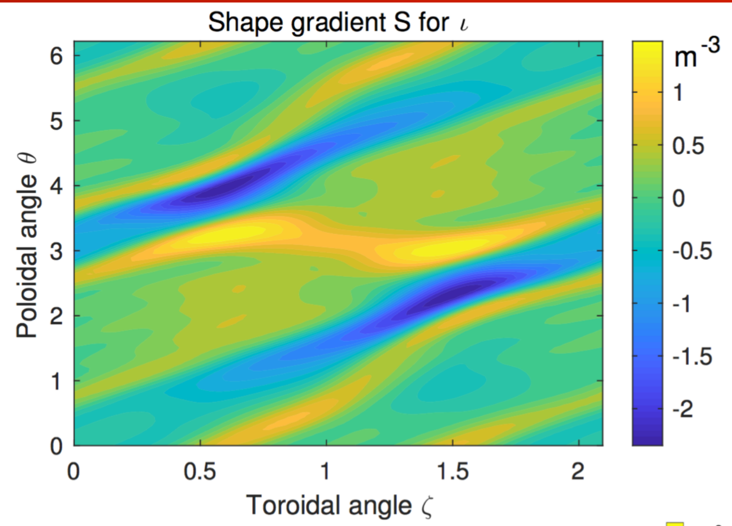
$$\delta f = \int d^2 a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S$$

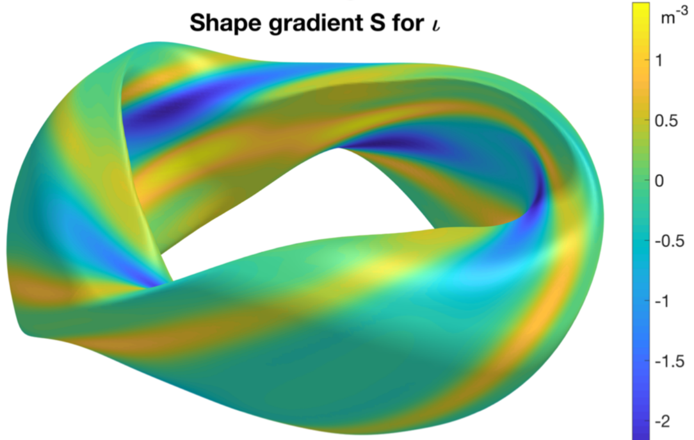
Example: Rotational transform at r/a=0.5

Shape gradient for boundary surface:

Parameter derivatives from STELLOPT/VMEC:



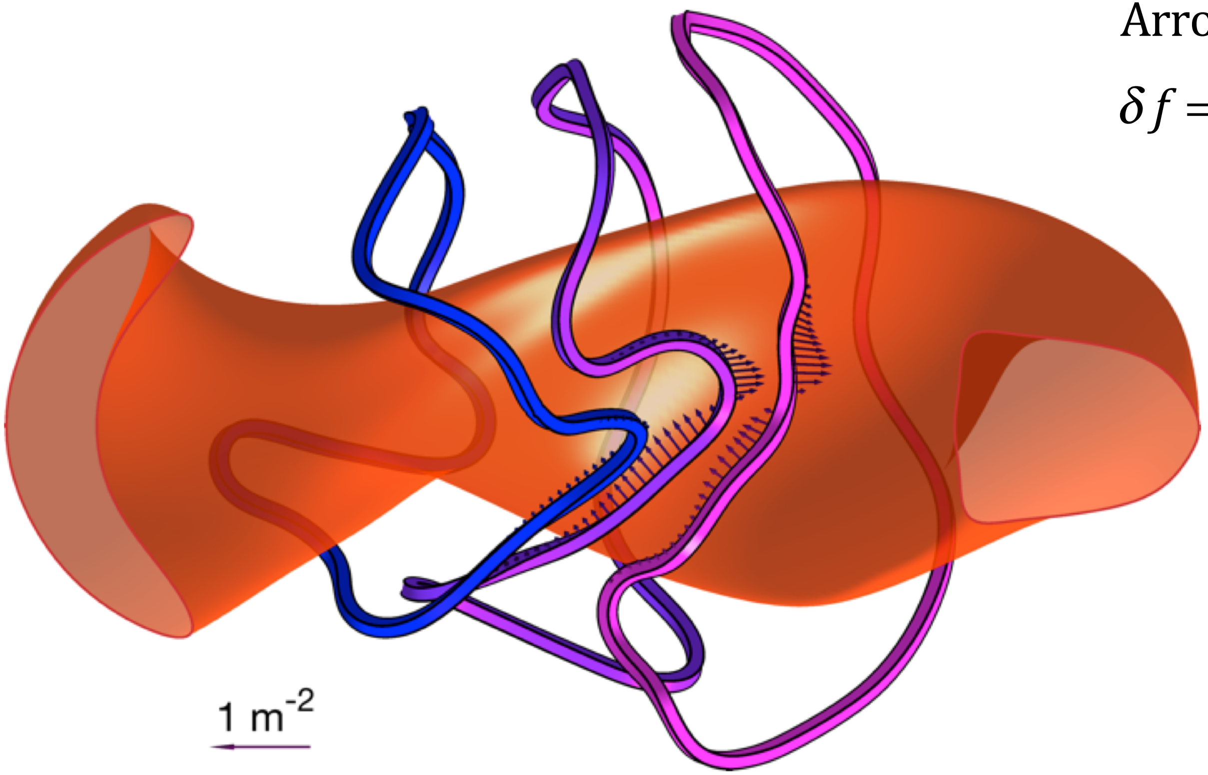




$$\delta f = \int d^2 a \left(\delta \mathbf{r} \cdot \mathbf{n} \right) S$$

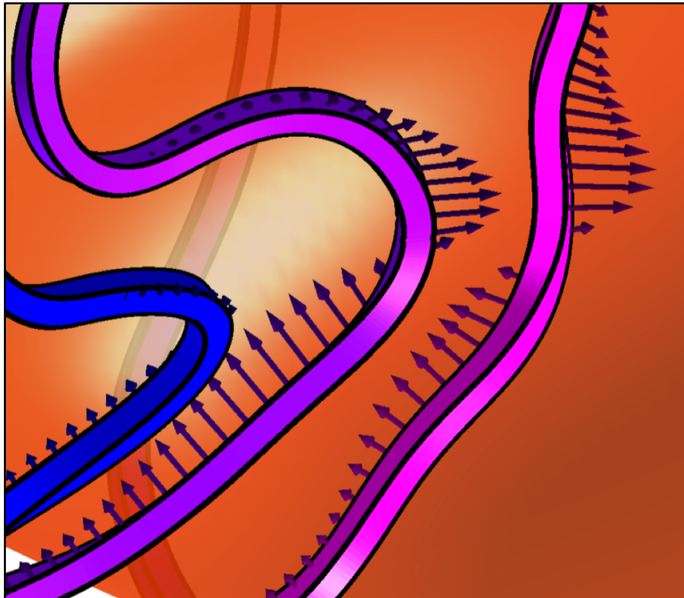
Example: Rotational transform at r/a=0.5

Shape gradient for coils:



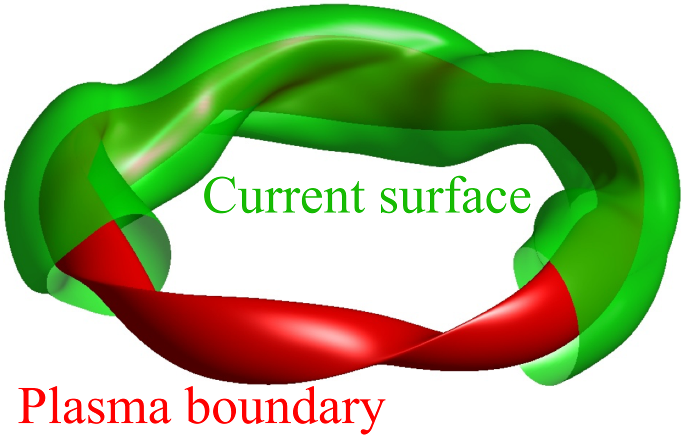
Arrows show $\mathbf{S}_k$ where

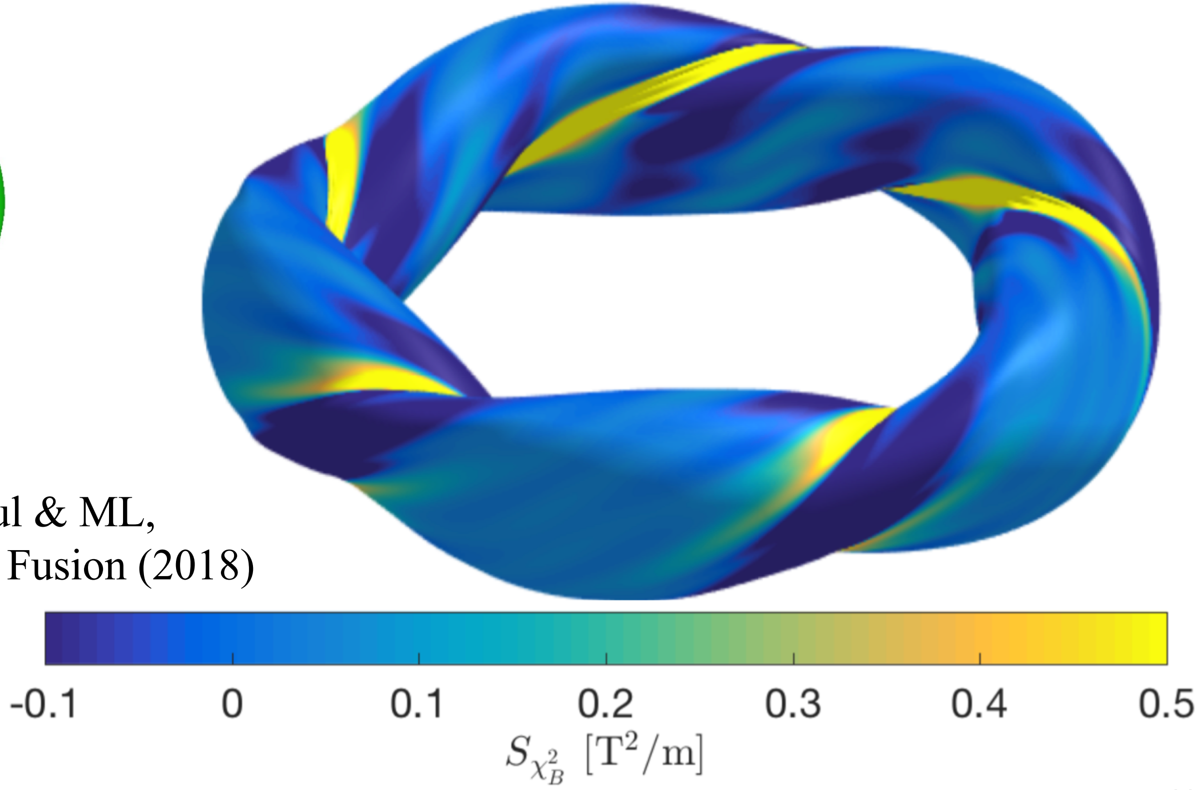
$$\delta f = \sum_{\text{coils } k} \int d\ell \; \delta \mathbf{r} \cdot \mathbf{S}_k .$$



The shape gradient can show where coils must be close to the plasma.

Shape gradient on a current surface for $f = \int d^2a \left(\mathbf{B}\cdot\mathbf{n}\right)^2$ given a fixed plasma boundary





E. J. Paul & ML,
Nuclear Fusion (2018)

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Plasma boundary shape:

Parameters p_j are $\left\{R^c_{m,n}, Z^s_{m,n}\right\}$.

Compute $\dfrac{\partial f}{\partial p_j}$ using finite differences, e.g. STELLOPT.

Discretize shape gradient: $S(\theta,\zeta) = \sum_q S_q \cos(m_q\theta - n_q\zeta)$

$$\int d^2a\,(\delta\mathbf{r}\cdot\mathbf{n})S = \delta f \quad \Rightarrow \quad \text{Solve } \int d^2a \frac{\partial \mathbf{r}}{\partial p_j}\cdot\mathbf{n}S = \frac{\partial f}{\partial p_j} \text{ for } S. \quad (1)$$

Linear system, not square.

Check that $\dfrac{\partial f}{\partial p_j}$ is in the column space of matrix.

If so, (1) can be solved for S_q using pseudo-inverse of matrix.

In some cases, shape gradients can be computed analytically.

<u>Integrals over a curve:</u>

If $f\left[C\right]=\int_C d\ell\, Q$ for some $Q\left(\mathbf{r}\right)$ and space curve C,

$$\Rightarrow \quad \delta f=\int_C d\ell\, \delta\mathbf{r}\cdot \underbrace{\left[\left(\overleftrightarrow{\mathbf{I}}-\mathbf{tt}\right)\cdot\nabla Q-q\kappa\mathbf{n}\right]}_{\mathbf{S}}$$

where $\kappa=$ curvature, $\mathbf{t}=$ tangent.

<u>Integrals over a surface:</u>

If $f\left[\partial\Omega\right]=\int_{\partial\Omega} d^2a\, Q$ for some $Q\left(\mathbf{r}\right)$ and surface $\partial\Omega$,

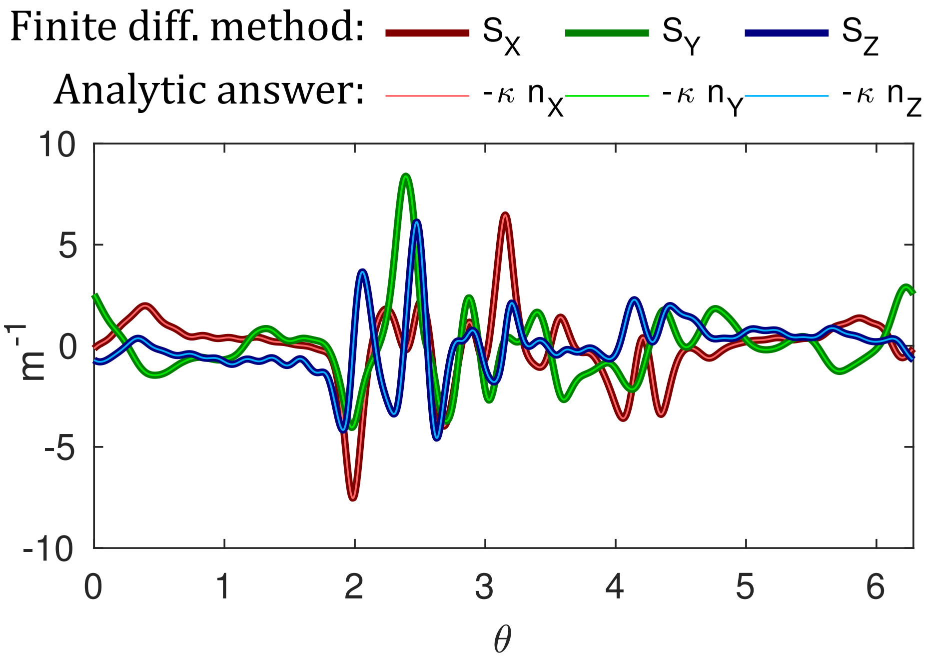
$$\Rightarrow \quad \delta f=\int_{\partial\Omega} d^2a\left(\delta\mathbf{r}\cdot\mathbf{n}\right)\underbrace{\left(\mathbf{n}\cdot\nabla Q-2QH\right)}_{S}$$

where $H=$ mean curvature.

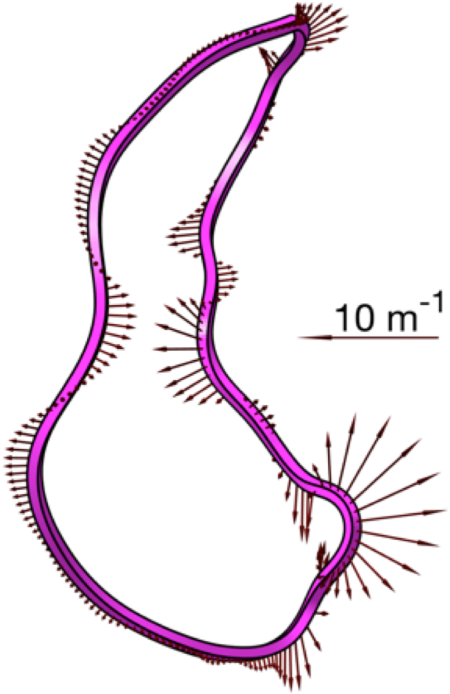
The algorithm for computing coil shape gradients can be verified by comparison to analytic theory.

Consider $f = \text{length}$. Analytic result: $\mathbf{S} = -\kappa \mathbf{n}$

(NCSX type-A coil)



Shape gradient **S**:



A magnetic sensitivity S_B can be computed from the shape gradient.

$$\mathbf{B}\cdot\nabla\psi=0 \qquad \Rightarrow \qquad \mathbf{B}_0\cdot\nabla\delta\psi+\delta\mathbf{B}\cdot\nabla\psi_0=0.$$

Also $\quad 0=d\psi=\delta\psi+\delta\mathbf{r}\cdot\nabla\psi_0.$

$$\Rightarrow \qquad \mathbf{B}_0\cdot\nabla\left(\delta\mathbf{r}\cdot\nabla\psi_0\right)=\delta\mathbf{B}\cdot\nabla\psi_0. \qquad (1)$$

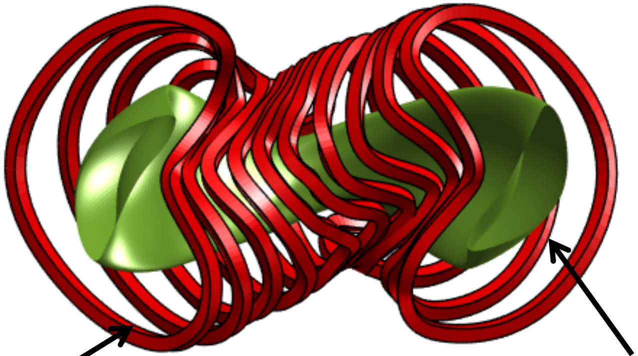
Define S_B by $\mathbf{B}_0\cdot\nabla S_B=\langle S\rangle-S. \qquad (2)$

Substitute $(2)\&(1)$ into $\delta f=\int d^2a\, S\delta\mathbf{r}\cdot\mathbf{n}$

$$\Rightarrow \qquad \delta f=\langle S\rangle\delta V+\int d^2a\, S_B\delta\mathbf{B}\cdot\mathbf{n}.$$

Perturbation to volume

3D MHD Toroidal Equilibrium



Coils

Assume good flux surfaces in plasma

$$-\nabla p + \frac{\mathbf{J}\times\mathbf{B}}{c} = 0$$

$$\nabla\times\mathbf{B} = \frac{4\pi}{c}\mathbf{J}$$

In vacuum

$$\nabla\times\mathbf{B} = \frac{4\pi}{c}\mathbf{J}_C$$

coil current

Poloidal flux

$$\mathbf{B} = \nabla\alpha\times\nabla\theta - \nabla\Phi_p(\alpha)\times\nabla\zeta$$

$$= \nabla\alpha\times\nabla\left(\theta - \iota(\alpha)\zeta\right)$$

Toroidal Flux

$$\iota(\alpha) = d\Phi_p(\alpha)/d\alpha$$

Rotational transform

Linear Perturbations to Equilibrium

$$\mathbf{J}_C \Rightarrow \mathbf{J}_C + \boxed{\delta\mathbf{J}_C}$$

$$\nabla p \Rightarrow \nabla p + \boxed{\nabla\cdot\delta\underline{\underline{\mathbf{P}}}}$$

$$\Phi_p(\alpha) \Rightarrow \Phi_p(\alpha) + \boxed{\delta\Phi_p(\alpha)}$$

$$\iota(\alpha) = d\Phi_p(\alpha)/d\alpha$$

Generalized Forces:

Changes in current/shape/location of coils

Added pressure tensor

Change in poloidal flux profile

$$\mathbf{A}_V \Rightarrow \mathbf{A}_V + \boxed{\delta\mathbf{A}_V}$$

$$\mathbf{B} \Rightarrow \mathbf{B} + \boxed{\nabla\times\left(\xi\times\mathbf{B} - \delta\Phi_P\nabla\zeta\right)}$$

$$I_T \Rightarrow I_T + \boxed{\delta I_T(\alpha)}$$

Generalized responses:

Changes in vacuum fields

Changes in magnetic field

Change in toroidal current profile

Generalized Forces and Responses

$$\text{Responses}\quad \begin{pmatrix} \delta \mathbf{A}_V \\ \xi \\ d\delta I_T / d\alpha \end{pmatrix} = \underline{\underline{O}} \begin{pmatrix} \delta \mathbf{J}_C \\ \nabla \cdot \delta \underline{\underline{P}} \\ \delta\Phi_P \end{pmatrix} \quad \text{Forces}$$

More generically, for two different perturbations

$$\delta x_i^{(1)} = \sum_j O_{ij}\, \delta F_j^{(1)} \qquad \delta x_i^{(2)} = \sum_j O_{ij}\, \delta F_j^{(2)}$$

Onsager Symmetry Gives

$$\sum_j \left\{ \delta x_i^{(1)} \delta F_i^{(2)} - \delta x_i^{(2)} \delta F_i^{(1)} \right\} = 0$$

Onsager Symmetry for MHD Equilibria (free boundary)

$$\int_{VP} d^3x\left(-\xi^{(1)}\cdot\nabla\cdot\left(\delta\underline{\underline{\mathbf{P}}}^{(2)}+\xi^{(2)}\cdot\nabla p\underline{\underline{\mathbf{1}}}\right)+\xi^{(2)}\cdot\nabla\cdot\left(\delta\underline{\underline{\mathbf{P}}}^{(1)}+\xi^{(1)}\cdot\nabla p\underline{\underline{\mathbf{1}}}\right)\right)$$

Pressure - Displacement

$$+\frac{2\pi}{c}\int_{VP} d\alpha\left(\delta\Phi_p^{(1)}\frac{d}{d\alpha}\delta I_T^{(2)}-\delta\Phi_p^{(2)}\frac{d}{d\alpha}\delta I_T^{(1)}\right)$$

Rotational transform –Toroidal current

$$+\frac{1}{c}\int_{V-ext} d^3x\left(\delta\mathbf{J}_C^{(1)}\cdot\mathbf{A}_V^{(2)}-\delta\mathbf{J}_C^{(2)}\cdot\mathbf{A}_V^{(1)}\right)=0$$

Coil current – Vector potential

Onsager Symmetry for MHD Equilibria (given or fixed boundary)

$$\int_{VP} d^3x \left(-\xi^{(1)} \cdot \nabla \cdot \left(\delta \underline{\underline{\mathbf{P}}}^{(2)} + \xi^{(2)} \cdot \nabla p \underline{\underline{\mathbf{1}}} \right) + \xi^{(2)} \cdot \nabla \cdot \left(\delta \underline{\underline{\mathbf{P}}}^{(1)} + \xi^{(1)} \cdot \nabla p \underline{\underline{\mathbf{1}}} \right) \right)$$

Pressure - Displacement

$$-\frac{2\pi}{c} \int_{VP} d\alpha \left(\delta I_T^{(2)} \frac{d}{d\alpha} \delta\Phi_p^{(1)} - \delta I_T^{(1)} \frac{d}{d\alpha} \delta\Phi_p^{(2)} \right)$$

Rotational transform –Toroidal current

$$+\frac{1}{4\pi} \int_S d^2x \, \mathbf{n} \cdot \left(\xi^{(1)} \delta\mathbf{B}^{(2)} \cdot \mathbf{B} - \xi^{(2)} \delta\mathbf{B}^{(1)} \cdot \mathbf{B} \right) = 0$$

Surface displacement

Fixed boundary shape gradient

Figure of merit $F = \int_{VP} d^3x\, p(\alpha)$

$$dF(\boldsymbol{\xi}^{(1)}) = -\int_{VP} d^3x\, \boldsymbol{\xi}^{(1)} \cdot \nabla p + \int_{SP} d^2x\, (\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}) p$$

Forward (hard) problem

$$\left(\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}\right)_{SP} \neq 0 \qquad \delta\Phi_P^{(1)} = 0$$

$$\delta \underline{\underline{\boldsymbol{P}}}^{(1)} = -\boldsymbol{\xi}^{(1)} \cdot \nabla p \underline{\underline{\boldsymbol{1}}}$$

Adjoint (easy) problem

$$\left(\boldsymbol{\xi}^{(2)} \cdot \boldsymbol{n}\right)_{SP} = 0 \qquad \delta\Phi_P^{(2)} = 0$$

$$\delta \underline{\underline{\boldsymbol{P}}}^{(2)} = \left(-\boldsymbol{\xi}^{(2)} \cdot \nabla p + \delta p\right) \underline{\underline{\boldsymbol{1}}}$$

Use fixed boundary Onsager relation

$$\int_{VP} d^3x \left(-\boldsymbol{\xi}^{(1)} \cdot \nabla \delta p\right) = \frac{1}{4\pi} \int_{SP} d^2x \left(\boldsymbol{n} \cdot \boldsymbol{\xi}^{(1)}\right) \delta \boldsymbol{B}^{(2)} \cdot \boldsymbol{B}$$

$$\boxed{dF(\boldsymbol{\xi}^{(1)}) = \int_{SP} d^2x\, (\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}) \underbrace{\left(\frac{\delta \boldsymbol{B}^{(2)} \cdot \boldsymbol{B}}{4\pi\delta} + p\right)}_{S}}$$

Shape gradient (*S*) computed without

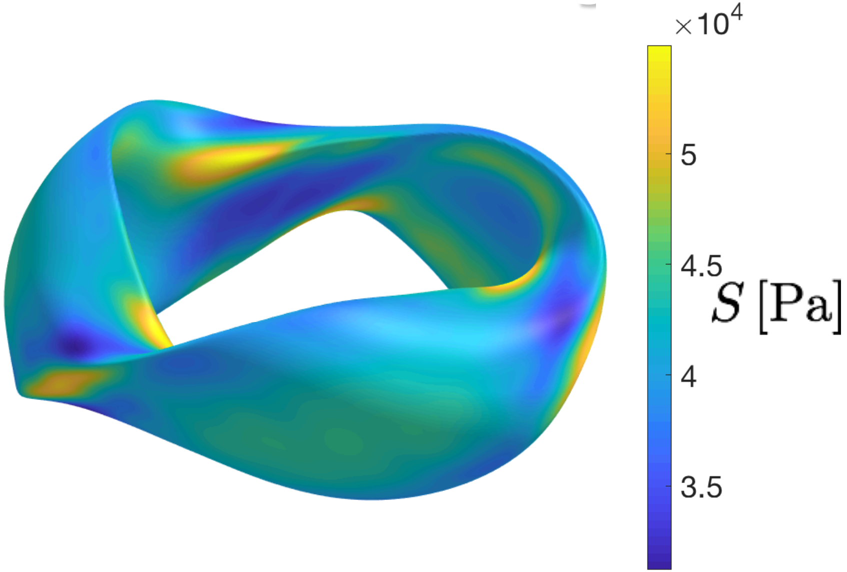
Computing the shape gradient the hard way

- NCSX LI383 equilibrium computed with VMEC[1] $\{R^c_{mn}, Z^s_{mn}\}$
- $\{\partial F/\partial R^c_{mn}, \partial F/\partial Z^s_{mn}\}$ applied by perturbing boundary harmonics
- S computed from $\left(\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}\right)_{SP}$ by solving a small linear system[2]

Forward problem

$$\left(\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}\right)_{SP} \neq 0 \quad \delta\Phi_P^{(1)} = 0$$

$$\delta P^{(1)} = -\boldsymbol{\xi}^{(1)} \cdot \nabla p \quad 1$$



$$F = \int_{VP} d^3x \, p(\alpha)$$

$$dF(\boldsymbol{\xi}^{(1)}) = \int_{SP} d^2x \left(\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}\right) S$$

[1]HIrshman and Whitman, 1983 Phys. Fluids 25 3553
[2]Landreman and Paul, 2018 Nucl. Fusion 58 076023

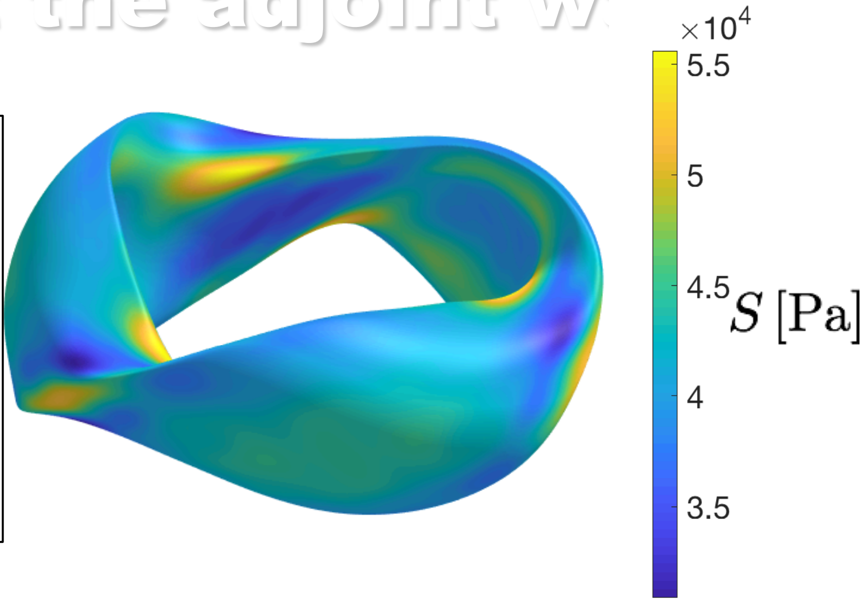
Computing the shape gradient the adjoint way

- NCSX LI383 equilibrium computed with VMEC[1]
- **Requires ≈1% CPU hours in comparison with numerical derivative**

Adjoint problem

$$\left(\boldsymbol{\xi}^{(2)} \cdot \boldsymbol{n}\right)_{SP} = 0 \qquad \delta \Phi_P^{(2)} = 0$$

$$\delta \underline{\underline{\boldsymbol{P}}}^{(2)} = \left(-\boldsymbol{\xi}^{(2)} \cdot \nabla p + \delta p\right) \underline{\underline{\mathbf{1}}}$$



$$F = \int_{VP} d^3x \, p(\alpha) \qquad dF(\boldsymbol{\xi}^{(1)}) = \int_{SP} d^2x \, (\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}) \left(\frac{\delta \boldsymbol{B}^{(2)} \cdot \boldsymbol{B}}{4\pi\delta} + p \right)$$

Free boundary shape gradient

Figure of merit $F = \int_{VP} d^3x\, p(\alpha)$

$$dF(\boldsymbol{\xi}^{(1)}) = -\int_{VP} d^3x\, \boldsymbol{\xi}^{(1)} \cdot \nabla p + \int_{SP} d^2x\, (\boldsymbol{\xi}^{(1)} \cdot \boldsymbol{n}) p$$

$p(\alpha_0) = 0$

Forward (hard) problem

$$\delta \boldsymbol{r}^{(1)}_{\text{coil}} \times \boldsymbol{t} \neq 0 \qquad \delta \Phi^{(1)}_P = 0$$

$$\delta \underline{\underline{\boldsymbol{P}}}^{(1)} = -\boldsymbol{\xi}^{(1)} \cdot \nabla p \underline{\underline{\boldsymbol{1}}}$$

$$\delta I^{(1)}_{\text{coil}} = 0$$

Adjoint (easy) problem

$$\delta \boldsymbol{r}^{(1)}_{\text{coil}} \times \boldsymbol{t} = 0 \qquad \delta \Phi^{(2)}_P = 0$$

$$\delta \underline{\underline{\boldsymbol{P}}}^{(2)} = \left(-\boldsymbol{\xi}^{(2)} \cdot \nabla p + \delta p\right) \underline{\underline{\boldsymbol{1}}}$$

$$\delta I^{(2)}_{\text{coil}} = 0$$

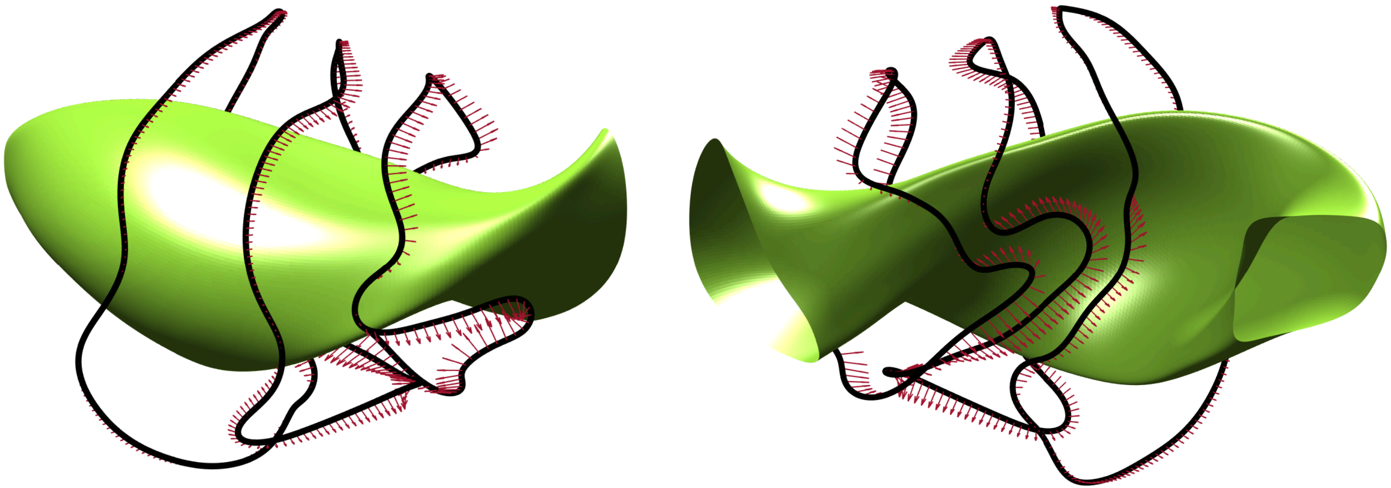
Use free boundary Onsager relation

$$\int_{VP} d^3x \left(-\boldsymbol{\xi}^{(1)} \cdot \nabla \cdot (\delta p \underline{\underline{\boldsymbol{1}}})\right) = \frac{4\pi}{c} \sum_{\text{coils}} I_{\text{coil}} \left(\oint dl \left(\delta \boldsymbol{r}^{(1)}_{\text{coil}} \times \boldsymbol{t}\right) \cdot \delta \boldsymbol{B}_2 \right)$$

$$\boxed{dF(\boldsymbol{\xi}^{(1)}) = \sum_{\text{coils}} \oint dl\, \delta \boldsymbol{r}^{(1)}_{\text{coil}} \cdot \underbrace{\left(\frac{4\pi}{c\delta} I_{\text{coil}} \boldsymbol{t} \times \delta \boldsymbol{B}_2 \right)}_{S}}$$

Coil shape gradient (*S*) computed without

Free boundary shape gradient



$$F = \int_{VP} d^3x\, p(\alpha)$$

$$dF(\boldsymbol{\xi}^{(1)}) = \sum_{\text{coils}} \oint dl\, \delta \boldsymbol{r}^{(1)}_{\text{coil}} \cdot \left(\frac{4\pi}{c\delta} I_{\text{coil}} \boldsymbol{t} \times \delta \boldsymbol{B}_2 \right)$$

- NCSX LI383 equilibrium computed with VMEC[1]
- $\delta \boldsymbol{B}_2$ computed at coil locations using DIAGNO[3]

[1]H.J. Gardner 1990 *Nucl. Fusion* 30 1417