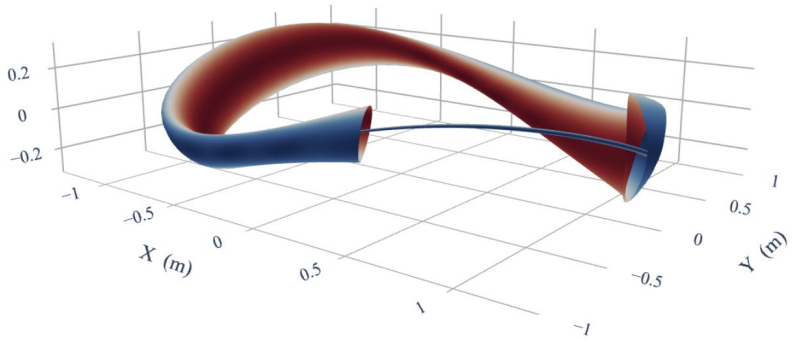
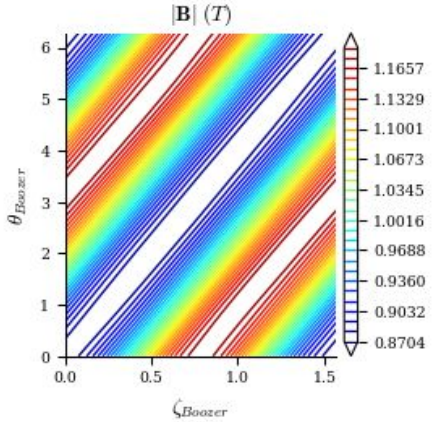
Stellarator Equilibrium and Optimization with DESC

Rory Conlin[1], Joao Biu[2], Daniel Dudt[1,3], Yigit Gunsur Elmacioglu[1], Rogerio Jorge[4], Rahul Gaur[1], Patrick Kim[5], Kian Orr[1], Dario Panici[1], Eduardo Rodriguez[6], Francisco Saenz[1], Jonathan Schilling[7], Kaya Unalmis[1], Egemen Kolemen[1]

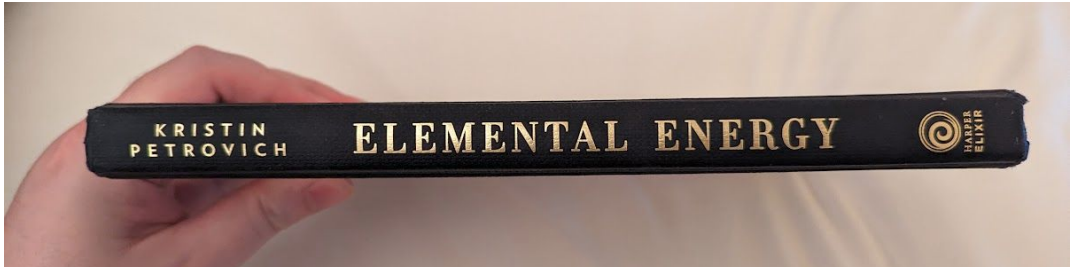


1. PU MAE
2. IST Lisbon
3. Thea Energy
4. U. Wisconsin
5. PU Astrophysics
6. IPP Greifswald
7. Proxima Fusion



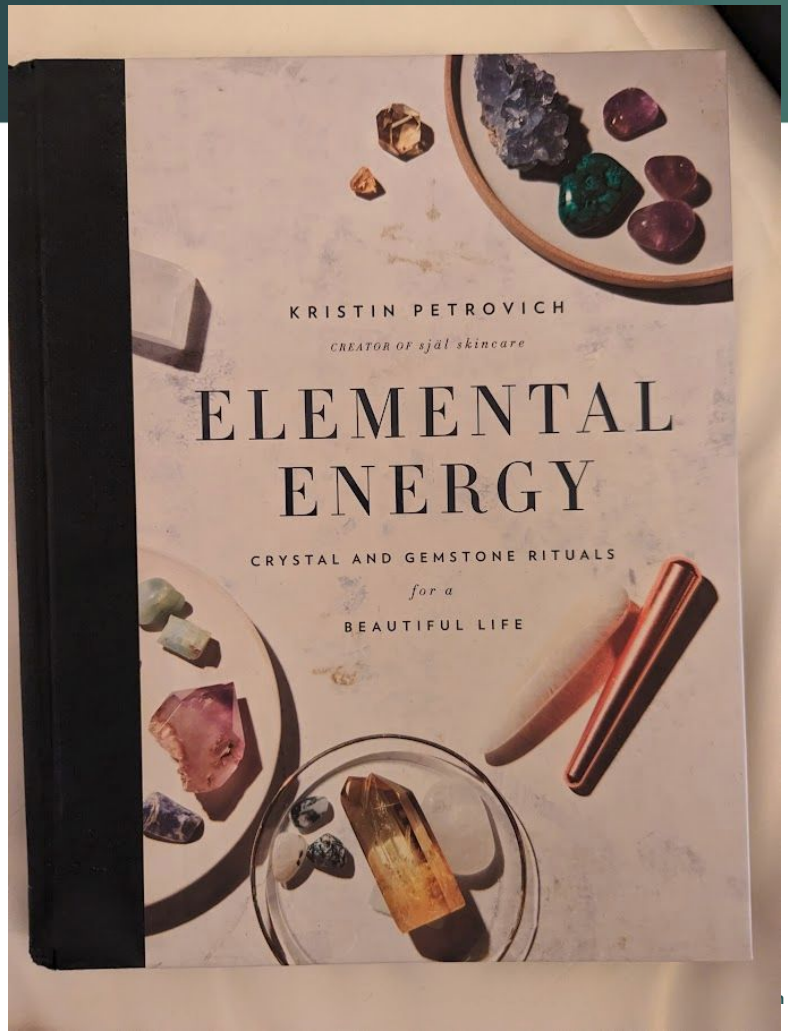
Plasma Control

Book on Elemental Energy in hotel - must really know their clientele

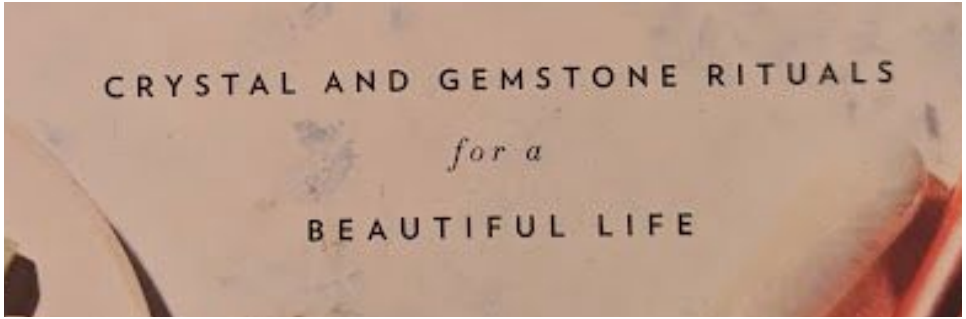


Very excited to read all about fusion and fission processes!

Plasma Control



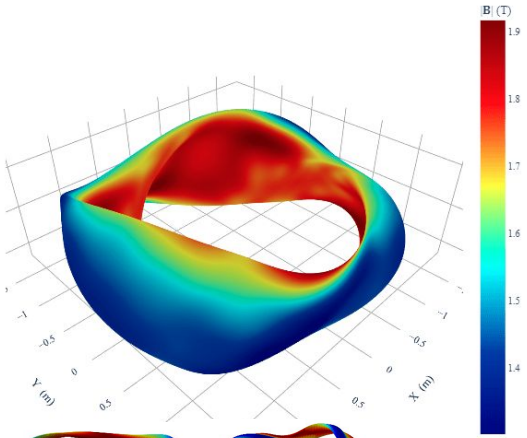
Or maybe not...



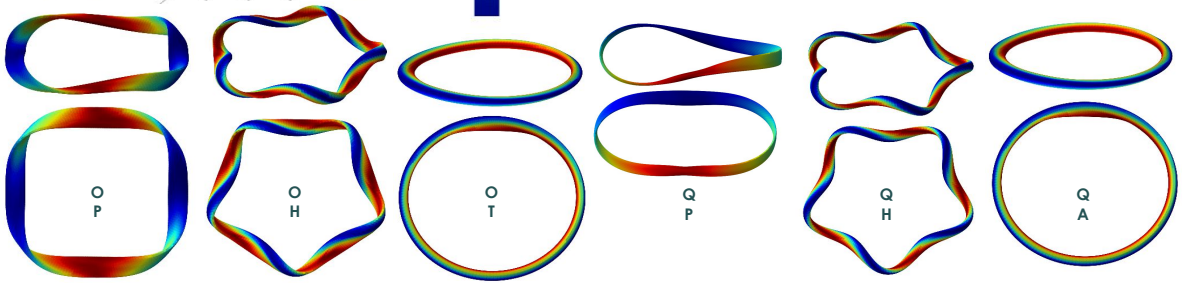
Plasma Control

DESC is a new(ish) tool for stellarator optimization

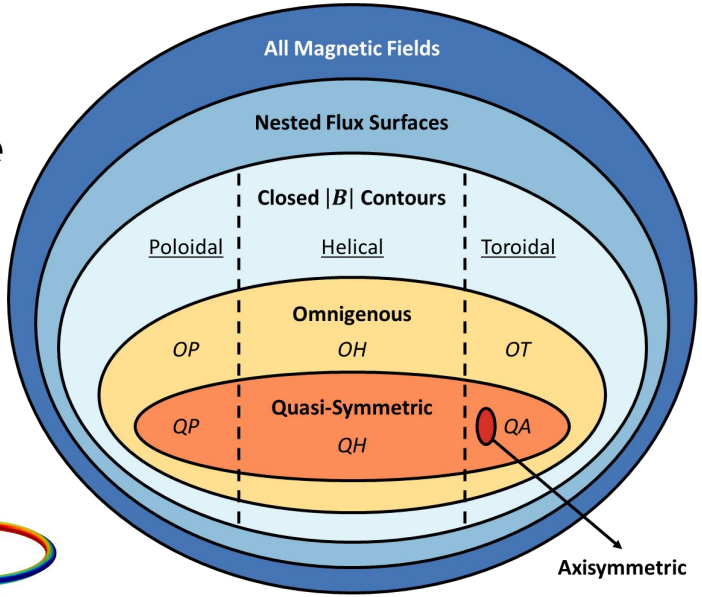
Accurate Equilibria



- Stellarator equilibria are complicated
- Design space is much larger than tokamaks



Fast Optimization



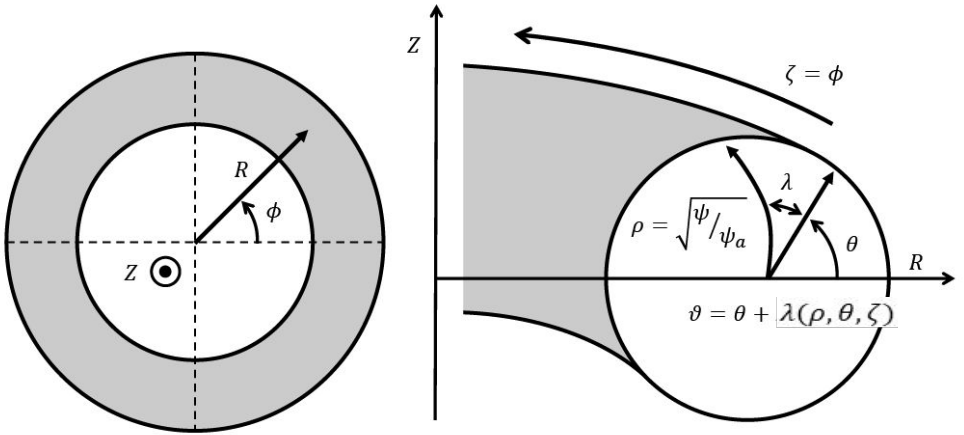
Stellarator Equilibrium and Optimization - DESC

- 3D Ideal MHD Equilibrium Code
- Assumes Nested Flux Surfaces
- Inverse Equilibrium Problem
- Minimizes Force Error Directly

$$\boldsymbol{F} = \boldsymbol{J} \times \boldsymbol{B} - \nabla p = 0$$

- Pseudospectral Code
- Python, AD, GPU-capable

3D Spectral Representation of $\mathbf{x} = (R, \lambda, Z)$ using Fourier-Zernike Basis



Preliminary: DESC represents solution in Fourier-Zernike Basis

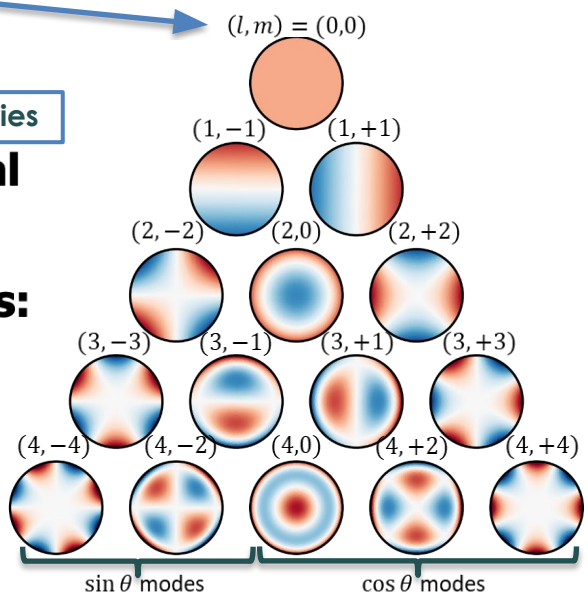
- $$X(\rho,\theta,\zeta) = \sum_{lmn} X_{lmn} Z_l^m(\rho,\theta)\mathcal{F}^n(\zeta)$$

  spectral coefficients; Zernike polynomials; Fourier series

- **Periodic boundary conditions for poloidal & toroidal angles**
- **Satisfies analyticity conditions at the magnetic axis:**

$$f(\rho,\theta) = \sum_m \rho^m \left(a_{m,0} + a_{m,2}\rho^2 + \cdots\right)\cos(m\theta) + \sum_m \rho^m \left(b_{m,0} + b_{m,2}\rho^2 + \cdots\right)\sin(m\theta)$$

- **Exponential convergence (if solution exists and is smooth)**



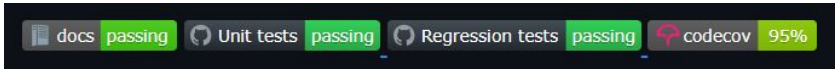
Boyd et al., *J. Comput. Phys.* (2011)
Lewis et al., *J. Math. Phys.* (1990)

DESC was developed from scratch with healthy coding practices in mind

- Open source Python code repository

  `https://github.com/PlasmaControl/DESC`

- Well documented, both in the code and external documentation
- Continuous Integration to test new code



- Modular structure enables custom applications and facilitates adding new capabilities
- Easy to install and start using `pip install desc-opt`
- Growing user + developer base around the world

Plasma Control

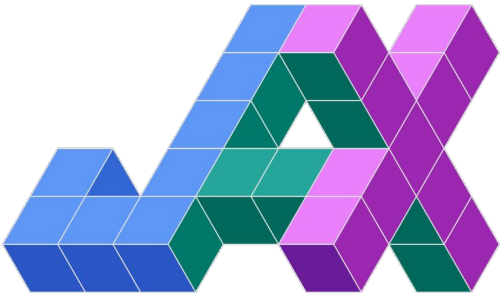
Uses JAX for automatic differentiation and JIT compilation

- **JAX is developed by Google, using the same backend as TensorFlow**
- **Automatic differentiation provides exact derivatives of arbitrary order**

  Jacobian matrix required for Newton method: $x_{n+1} = x_n - \left(\frac{\partial f}{\partial x}\right)^{-1} f(x)$

  - **yields derivative with SINGLE function call (no need for finite differences or manually writing out analytic derivatives)**
- **Just-in-time (JIT) compilation improves speed and memory usage using Accelerated Linear Algebra (XLA)**
- **Runs on both CPU & GPU**
- **Easy to implement**

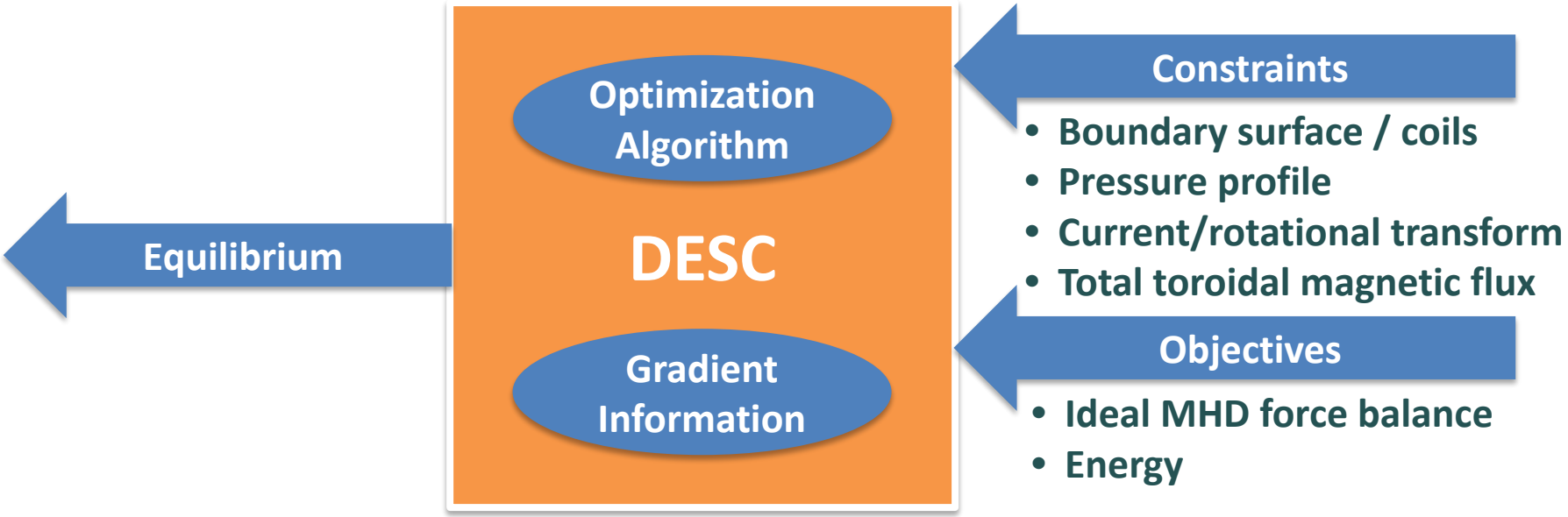
```
import jax.numpy as jnp
```



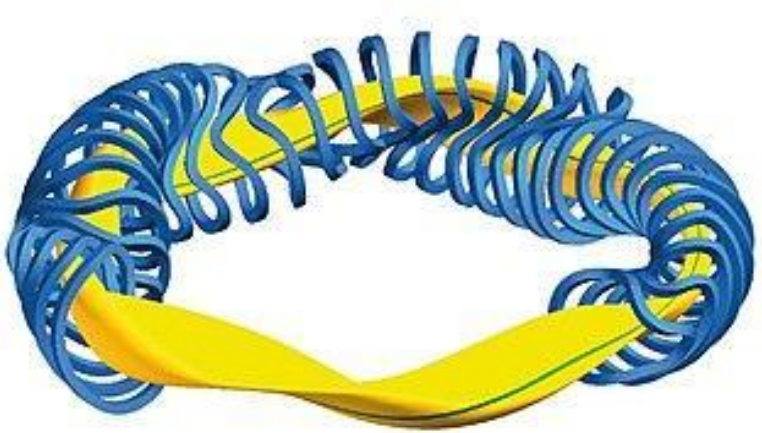
https://github.com/google/jax

Plasma Control

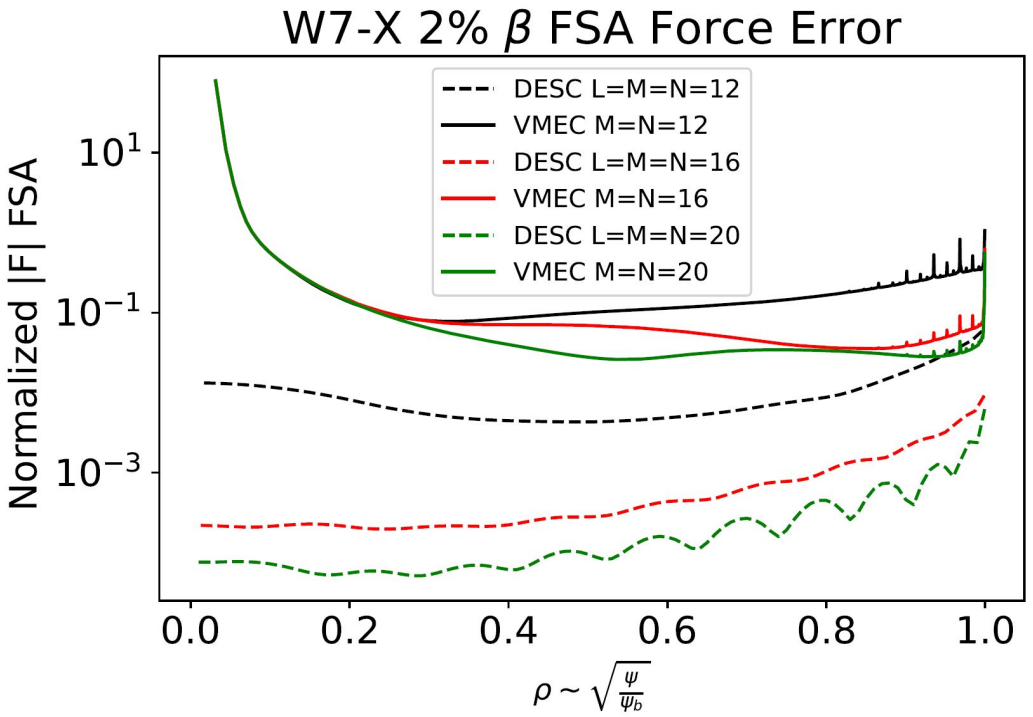
The Stellarator Equilibrium Problem



DESC spectral methods yield more accurate equilibrium solutions



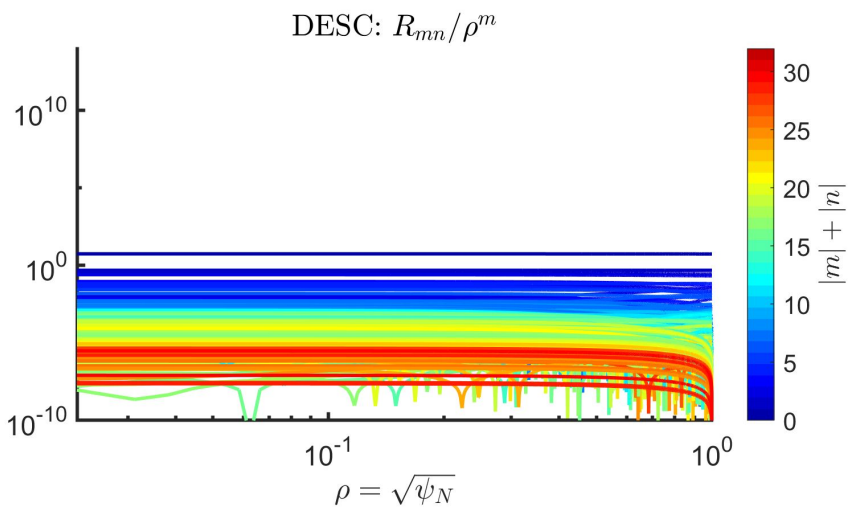
W7-X

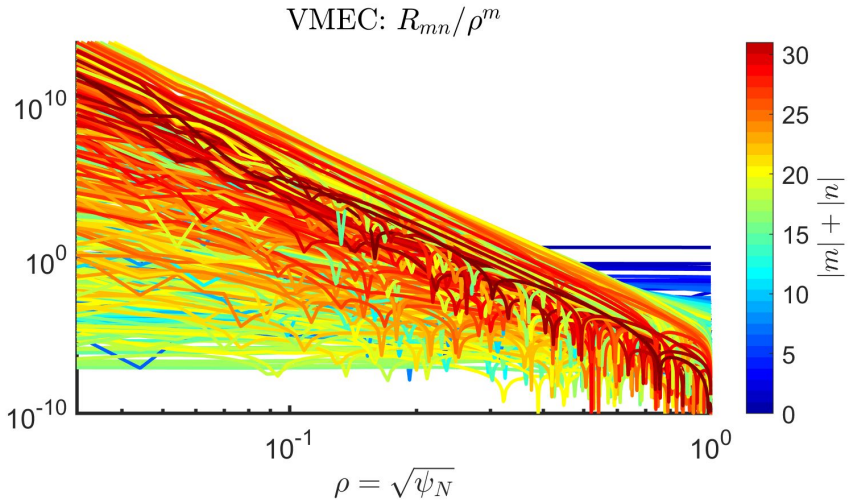


Plasma Control

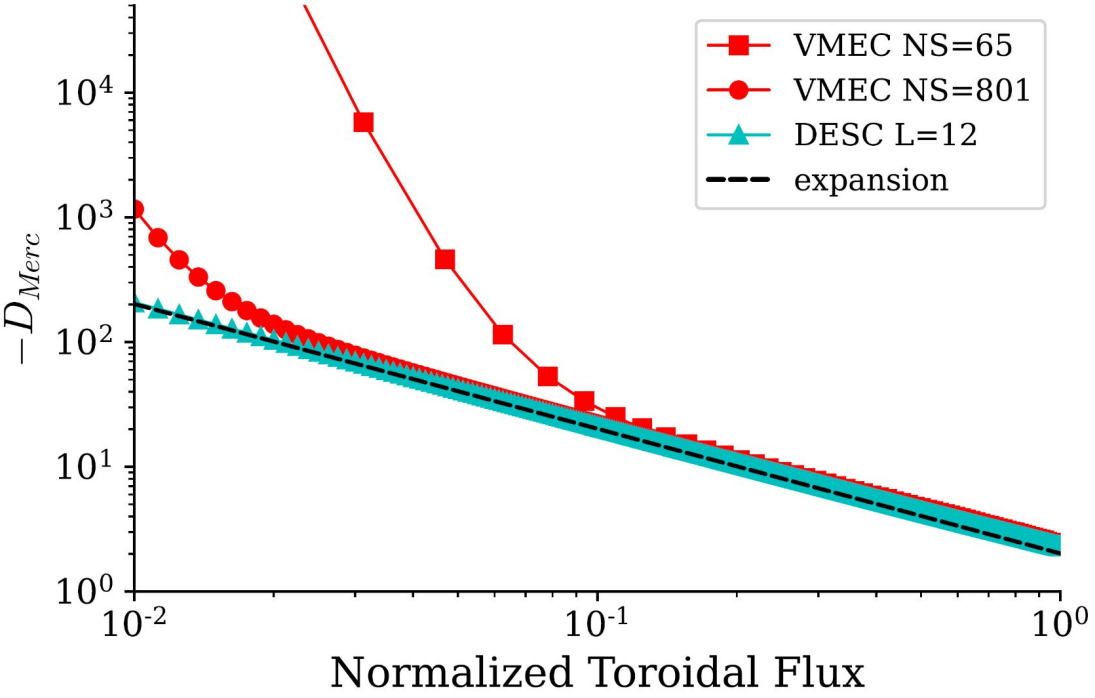
VMEC and DESC Analytic Constraint Near Axis

- **DESC coefficients obey constraint inherently due to Zernike basis**
- **VMEC Fourier coefficients do not**
  - **Unphysical modes in VMEC solution Fourier spectrum**





Accurately resolving the magnetic axis is important for stability calculations



VMEC requires high radial resolution to resolve axis

Run times:

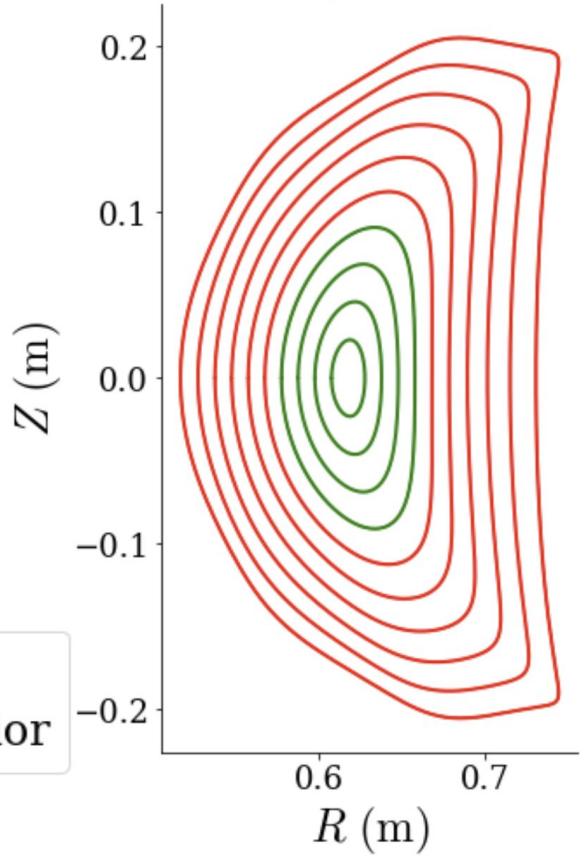
- **DESC = 0.2 GPU-hours (NVIDIA A100)**
- **VMEC = 5.2 CPU-hours (AMD Opteron 6276)**

Landreman & Sengupta, *J. Plasma Phys.* (2019)

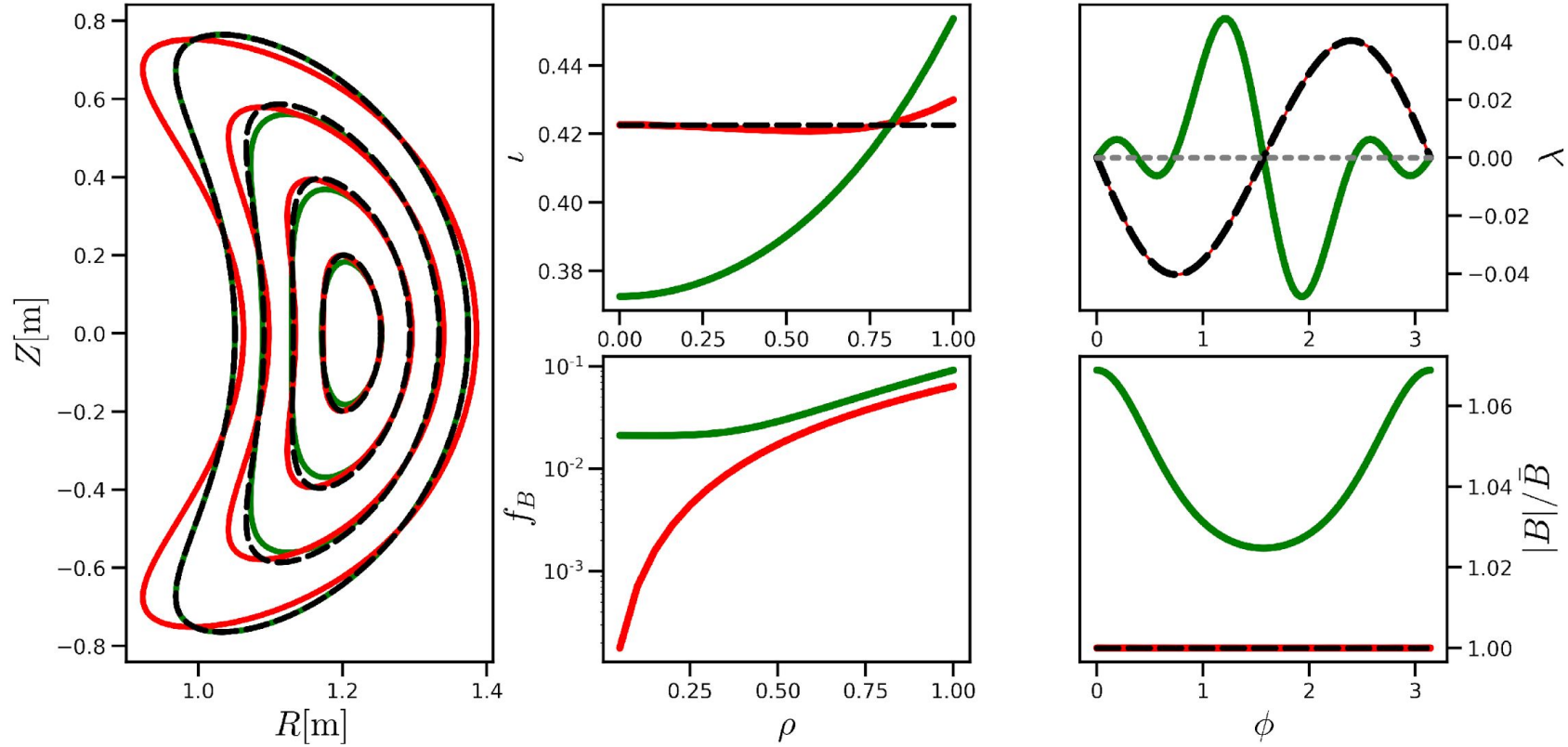
Near-Axis Expansion (NAE) Constraints in DESC (D. Panici, PU; E. Rodriquez, IPP)

NAE equilibrium evaluated at r =0.1

- Constrain global equilibrium by NAE behavior as $\varrho \to 0$
  - Use information from NAE where it is most valid
  - Avoid singular behavior present when evaluating at large r
- Map NAE coefficients to Fourier-Zernike modes of DESC to fix $O(\varrho^0)$ (axis) and $O(\varrho^1)$ behavior



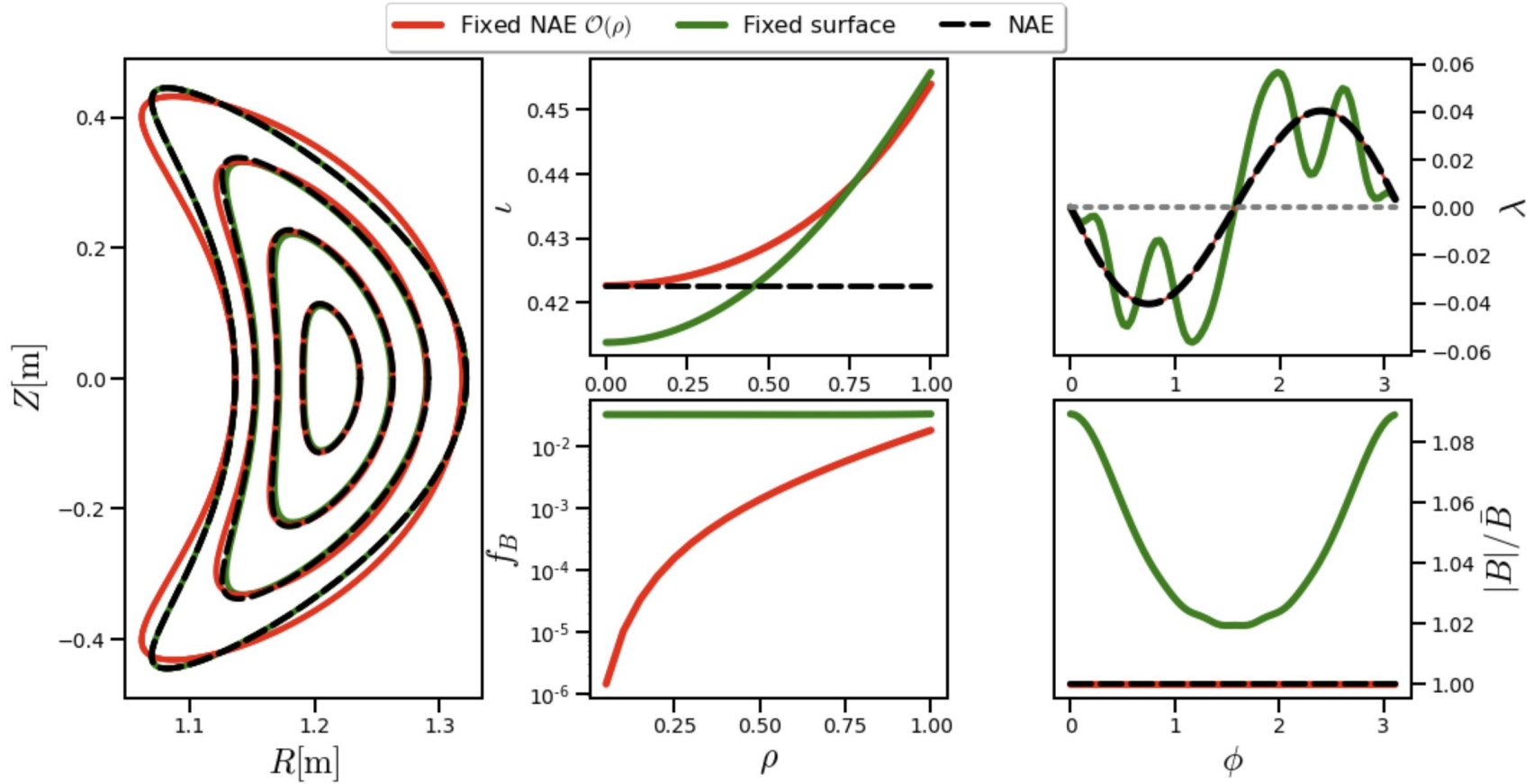
QA $O(\rho^1)$ NAE-constrained Equilibrium



pyQSC based on Precise QA from (Landreman and Paul 2022)

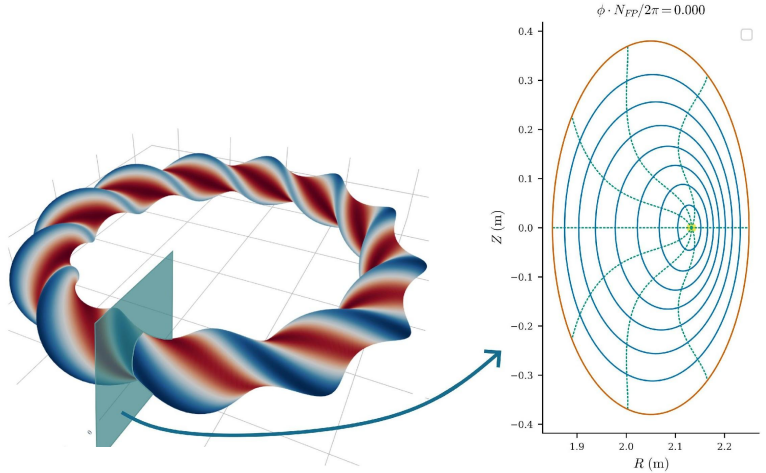
Plasma Control

QA O(ϱ^2) NAE-constrained Equilibrium

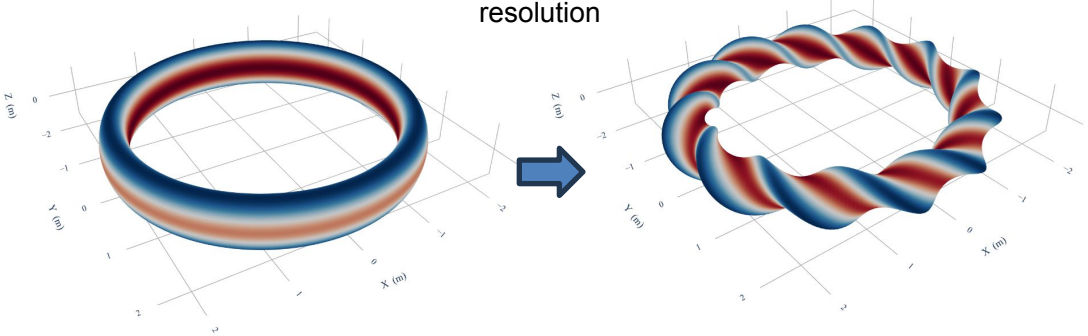


Poincare Boundary Condition (Y. Gunsur Elmacioglu, PU)

- **Aim: Solving 3D MHD Equilibrium of a stellarator with a given toroidal cross section instead of giving last closed flux surface (LCFS)**



Increase toroidal resolution

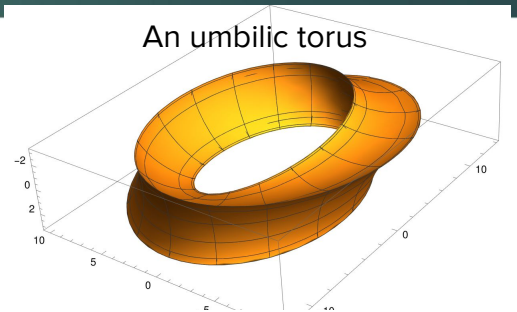


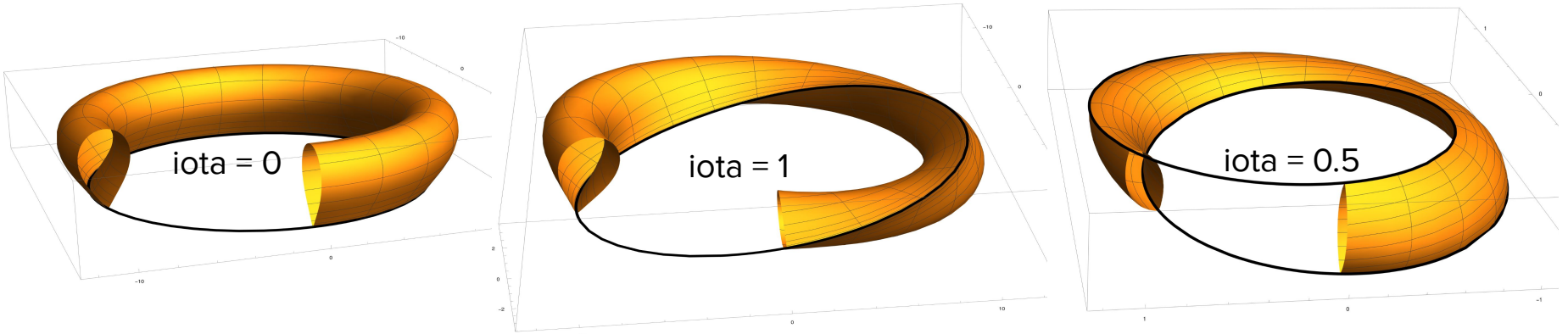
- **Method: Start with axisymmetric initial guess of same cross-section. Increase toroidal resolution, and minimize force error.**

Plasma Control

Umbilic Stellarator Design and Analysis (R. Gaur, PU)

- "Umbilic" shapes are shapes with a single side (think Mobius strips)
- Possible implication for stellarators
  - Natural locations for X-point or island divertor
  - long connection length -> reduce heat flux to divertor
- Capability implemented in DESC to investigate umbilic topologies

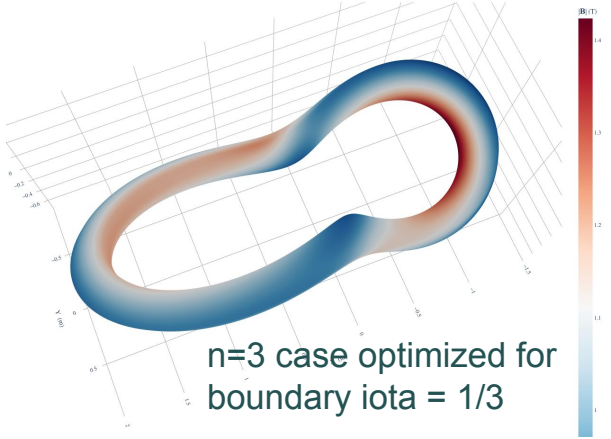


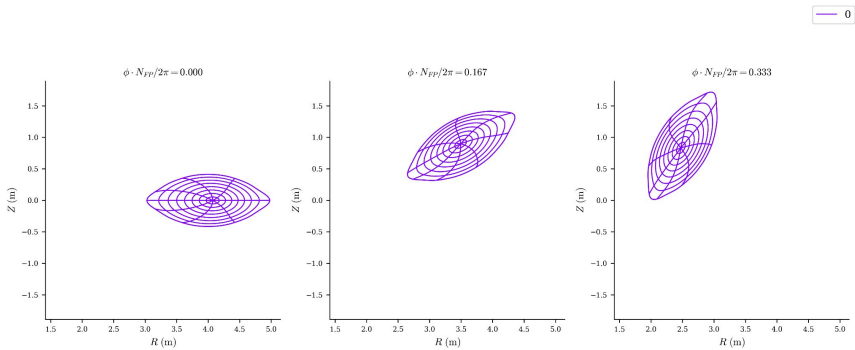


1. Field line stays on the X-point. By rotating the X point, we can control the boundary iota
2. By Increasing the poloidal and toroidal resolution in DESC, we can makes the boundary sharper

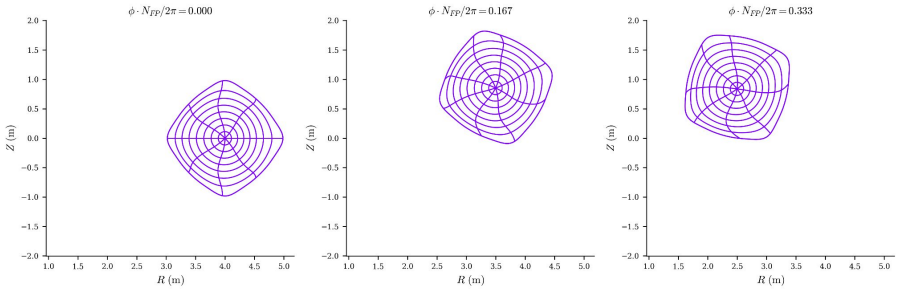
Umbilic equilibria for resonant diverter design

- **We can use an analytical representation to create umbilic boundaries with n (n > 1) sharp edges**
- **DESC imports this boundary and generates a fixed boundary equilibrium with a rational iota = m/n on the boundary**
- **This should make the sharp edges behave like X-points with rational iota values. Potential applications to resonant divertors**





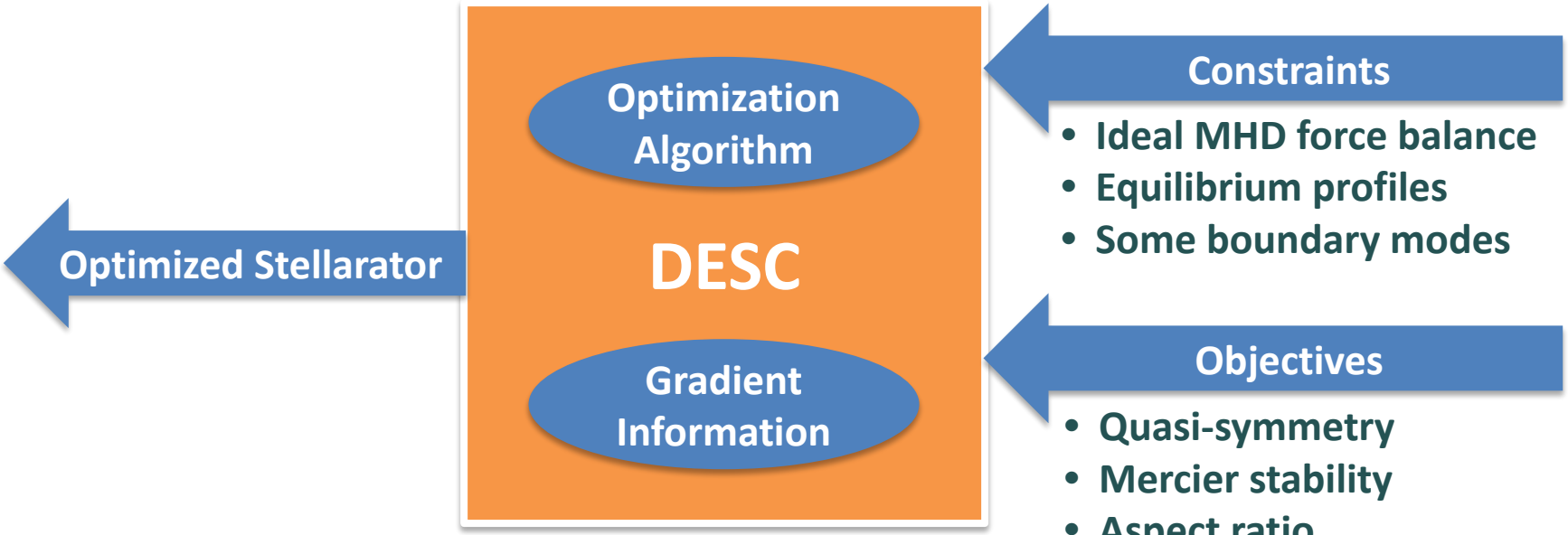
n=2



n=4

Plasma Control

The Stellarator Optimization Problem



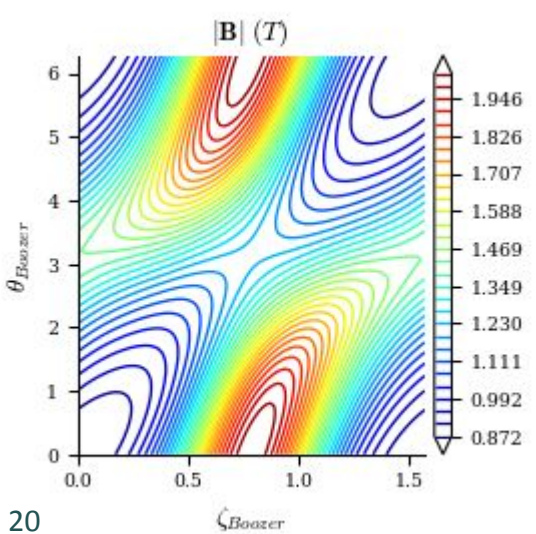
Stellarator Optimization in DESC - Example

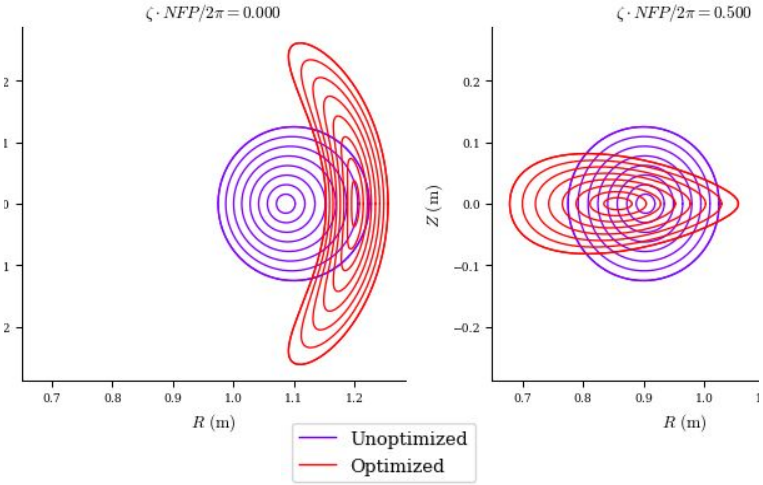
Optimizing for "Quasi Symmetry" - Proxy metric for particle confinement

- **In Quasisymmetry, |B| is 2D fxn on a given flux surface**

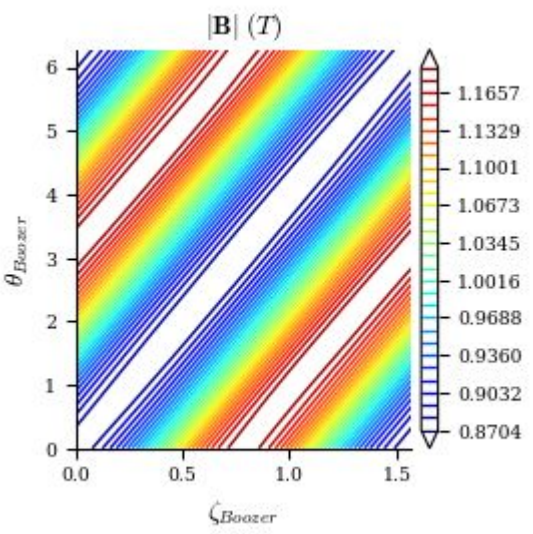
$$|B| = |B|(\rho, M\theta_B - N\zeta_B)$$

Unoptimized |B|

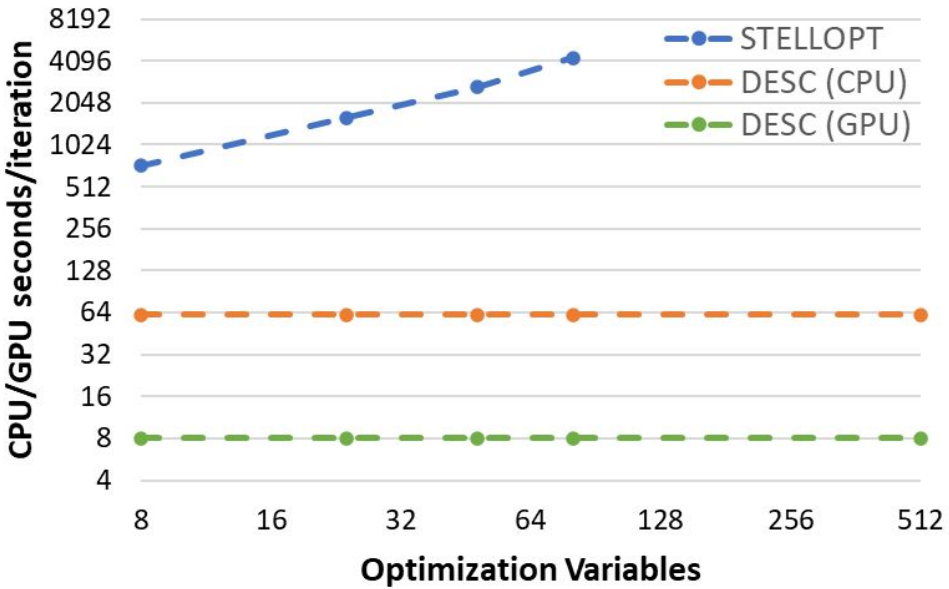




Optimized |B|



DESC Allow Much Faster Stellarator Optimization



- **Only require <u>one</u> equilibrium solve per optimization iteration**

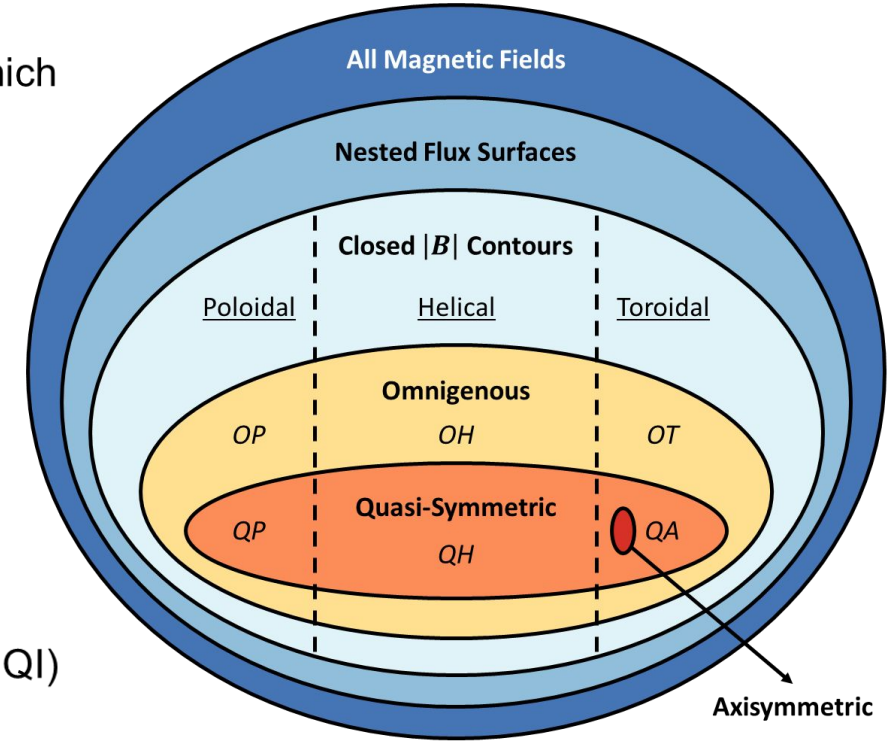
Optimization Code	Computation Time
STELLOPT (8 CPUs)	~ 2 hours
STELLOPT (16 CPUs)	~ 1.5 hours
STELLOPT (32 CPUs)	~ 1 hour
DESC (1 CPU)	< 30 minutes
DESC (1 GPU)	< 10 minutes

General Omnigenity: confined particles without Quasi-Symmetry

- **Omnigenity:** the class of magnetic fields in which the bounce-averaged radial drifts of trapped particles vanish
  - $B = |\boldsymbol{B}|$ contours are closed curves
  - B_{max} contour is straight in Boozer coords
  - "Bounce distances" between consecutive B points are independent of the field line α

- Previous omnigenous equilibria were limited to:
  - Quasi-Axisymmetry (QA)
  - Quasi-Helical symmetry (QH)
  - Omnigenity with Poloidal contours (OP aka QI)

- General omnigenity = larger design space!



Plasma Control

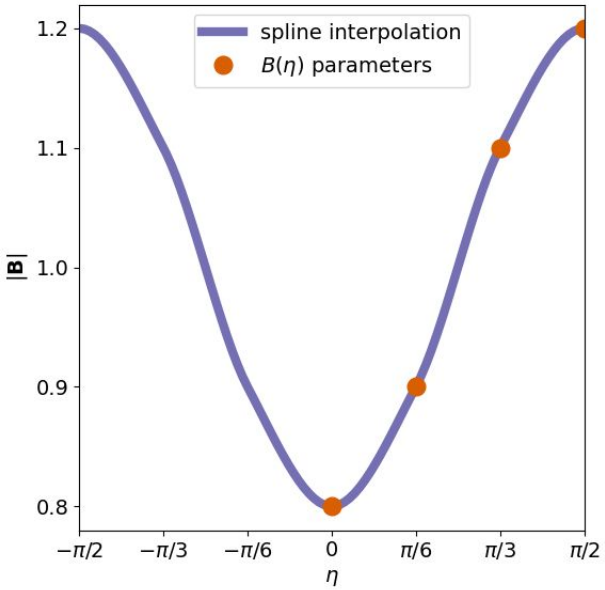
Omnigenity Optimization Through Omnigenous Phase Space Definition (D. Dudt, PU/Thea)

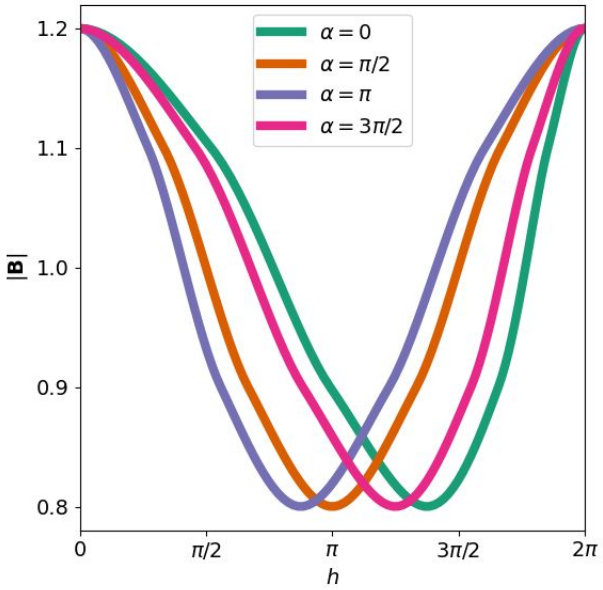
- **Idea: Create a model that can describe any omnigenous field of interest**
- **Then, in optimization we can penalize the difference between the target omnigenous field and the equilibrium's field**

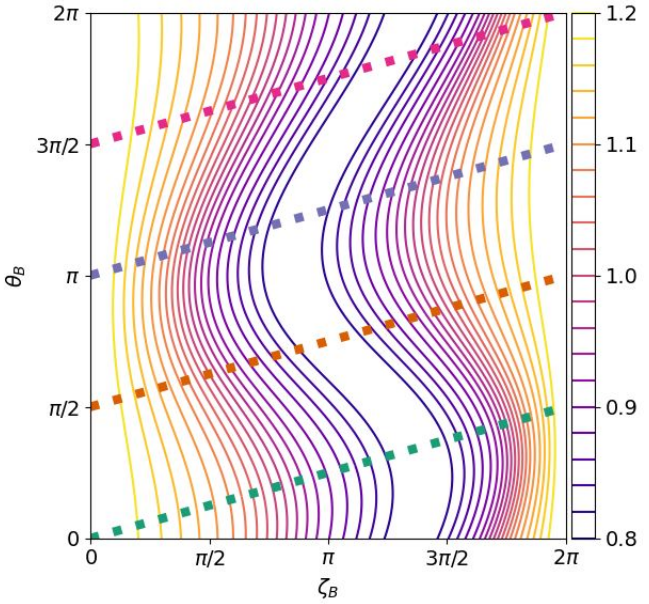
$$\boldsymbol{f_{OM}} = B_{eq}(\alpha,\eta) - B_{OM}(\alpha,\eta)$$

- **Because we parametrize the omnigenous field, we are free to:**
  - **Keep parametrization the same (target a specific omnigenous field)**
  - **Allow the parametrization to be part of the optimization (optimize for SOME omnigenous field)**

General omnigenity optimization is implemented in DESC



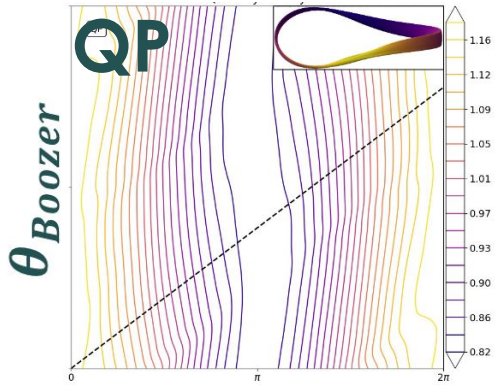


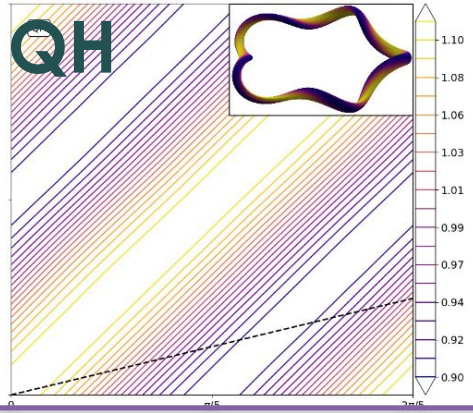


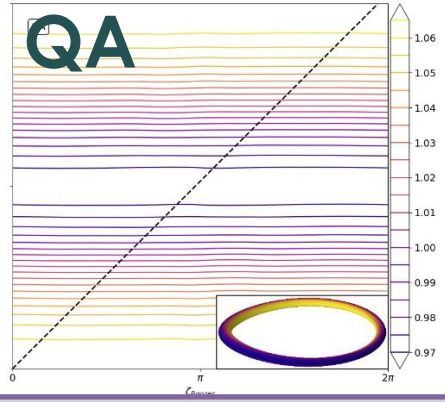
1. **Define a target magnetic well "shape" (in computational coordinate η)**
2. **Define a target "shift" on each field line (preserves constant bounce distances)**
3. **Optimize to minimize the errors: $B_{equilibrium} - B_{target}$**

Plasma Control

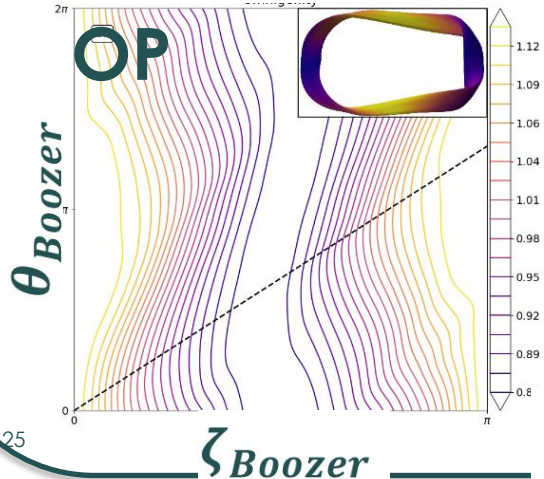
DESC can find equilibria with any omnigenity type

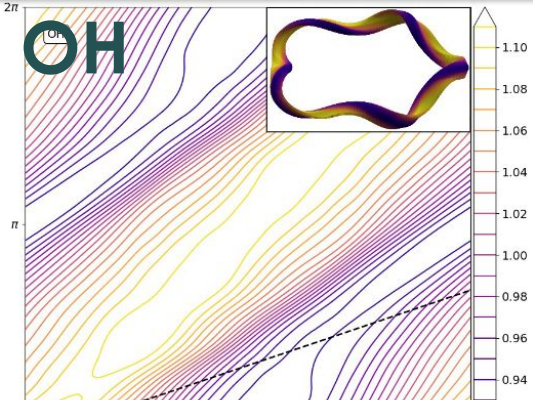


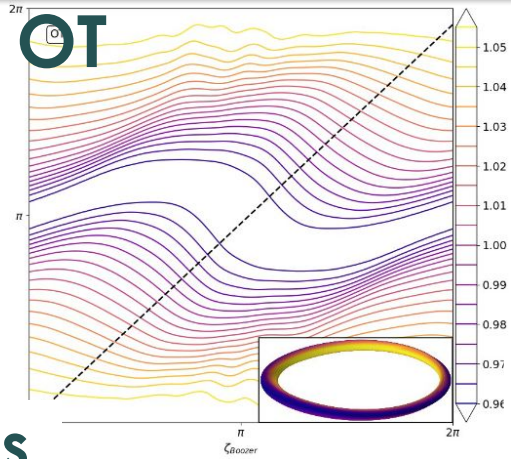




QS Fields



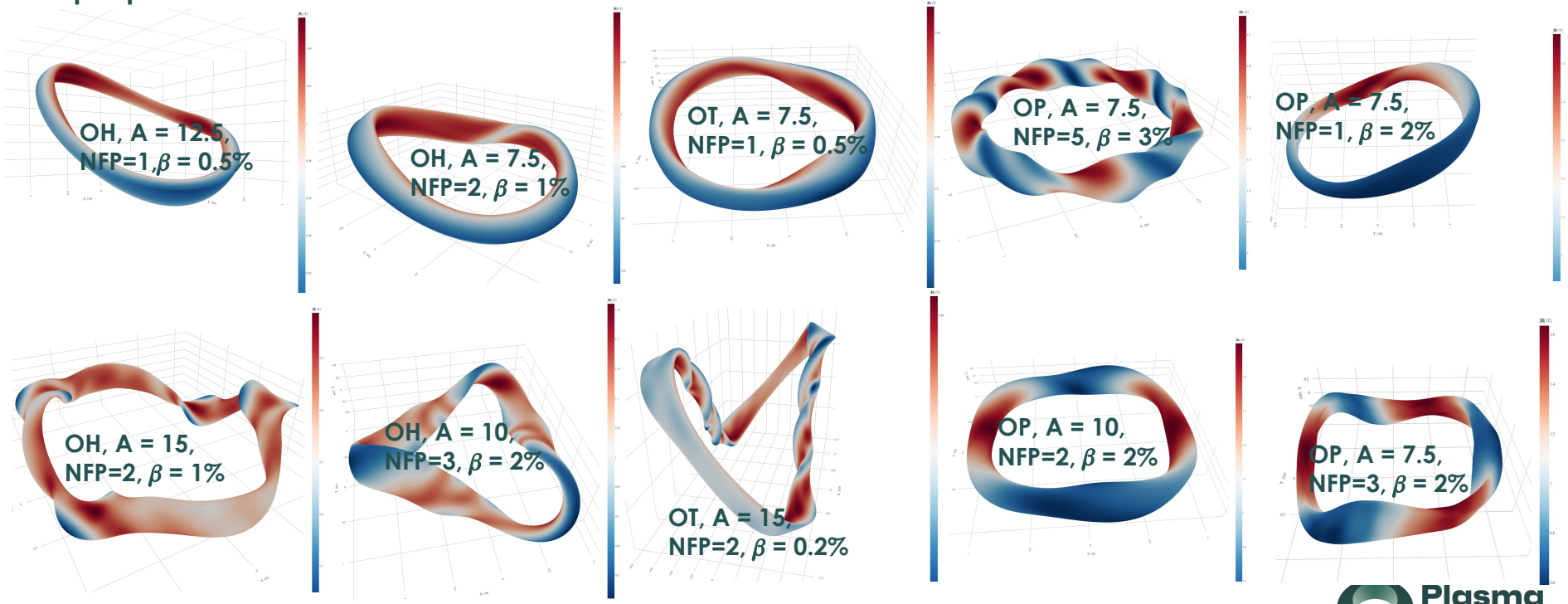




Omnigenous Fields

Database of omnigenous equilibria (R. Gaur, PU)

- Database of thousands of omnigenous equilibria created with DESC
- This will help us understand how omnigenity (and its subset quasi-symmetry) changes with different input parameters



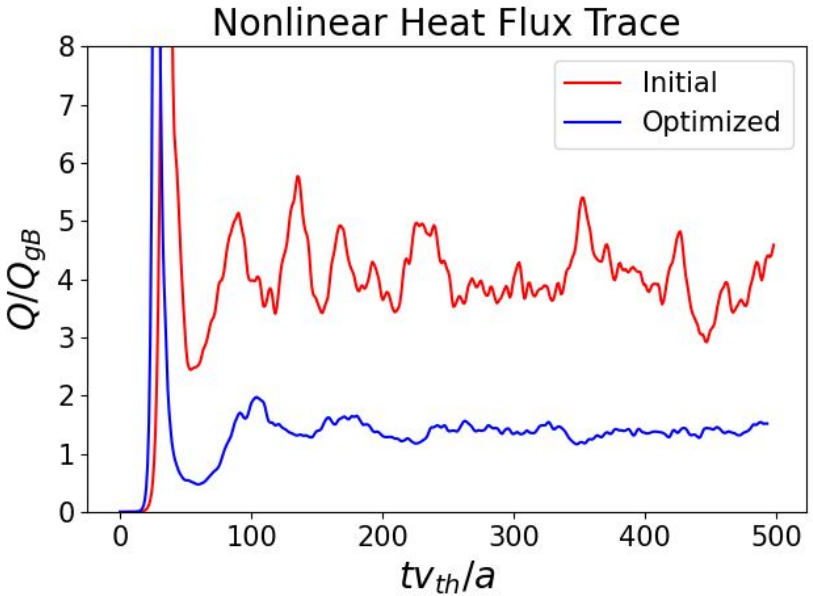
Turbulence + QS Optimization Using GX + DESC (P. Kim, PU)

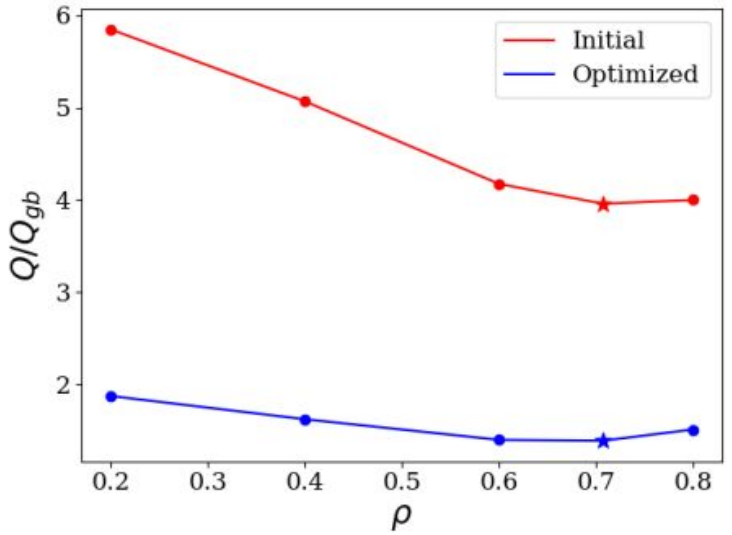
- GX + DESC coupling enables direct optimization of nonlinear heat fluxes with good quasi-symmetry.
- Why use “exact” finite differences for noisy objective?
- SPSA algorithm allows for gradient approximation for noisy objectives in only 2 function evals.
  - “Finite difference in all directions at once”
- Asymptotically unbiased estimate

$$\hat{\mathbf{g}}_{\mathbf{n}}\left(\mathbf{x}_{\mathbf{n}}\right)_i = \frac{f\left(\mathbf{x}_{\mathbf{n}} + \mathbf{c}_{\mathbf{n}}\right) - f\left(\mathbf{x}_{\mathbf{n}} - \mathbf{c}_{\mathbf{n}}\right)}{2c_{ni}},$$

Optimize for single flux tube, reduce nonlinear heat flux ~3x across entire volume

- Optimize for QH + nonlinear heat flux at s=0.5
- Reduces heat flux 3x over multiple field lines, multiple surfaces

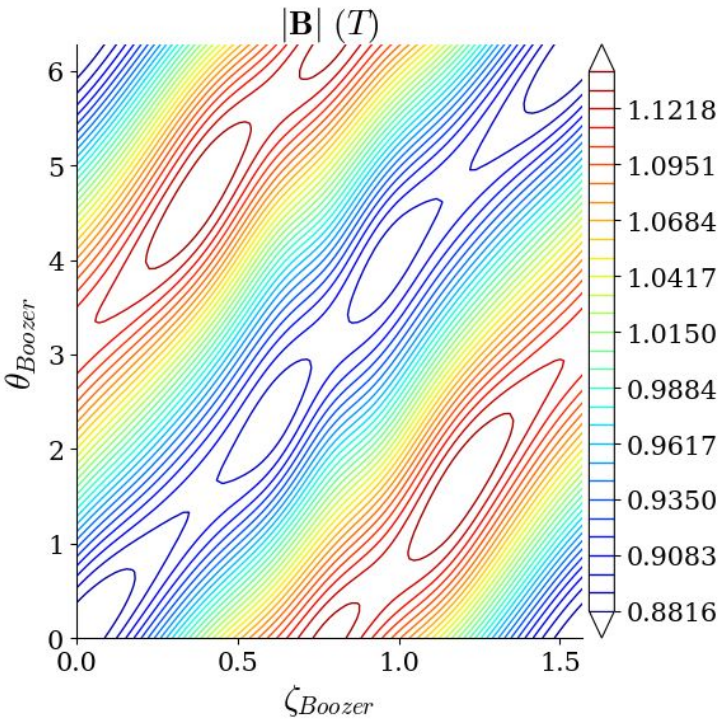




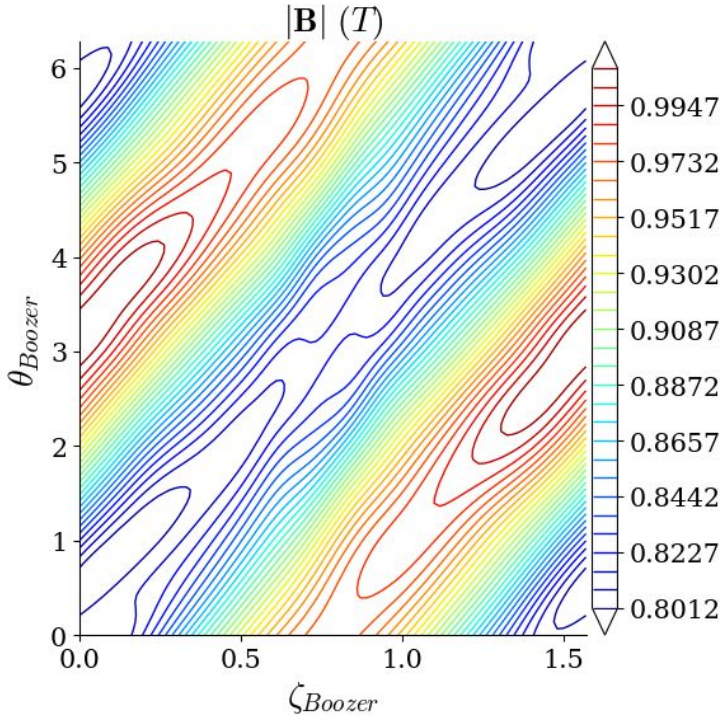
28

Minimal sacrifice of QS for reduced turbulence

QS only



QS + Turbulence

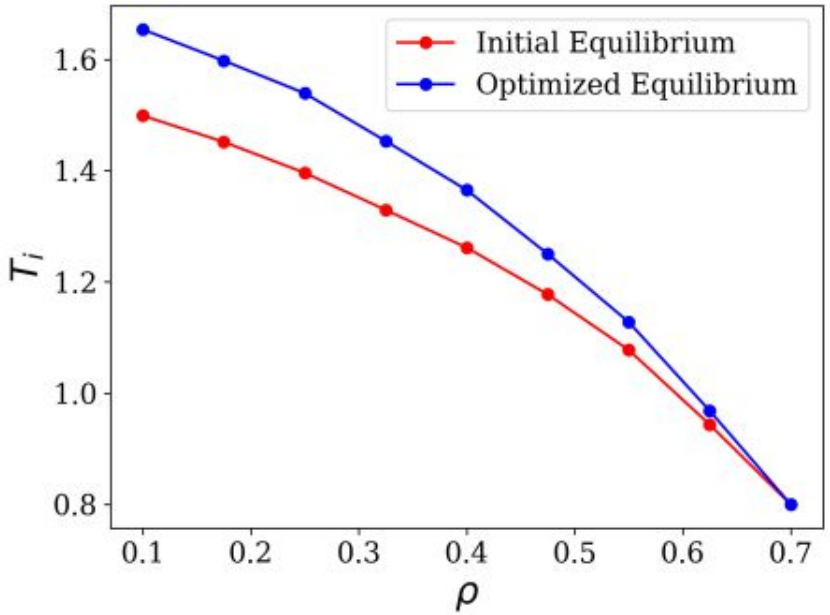


Initial

Optimized

Despite reduced heat flux, only marginal improvement in T_i

- Transport simulations done with Trinity 3D
- Increase core T_i by ~10%
- Less than expected, may need to optimize at lower a/L_T
- Self consistency is important!



GPU Allows Direct Optimization of Fast Particles (J. Biu, IST Lisbon)

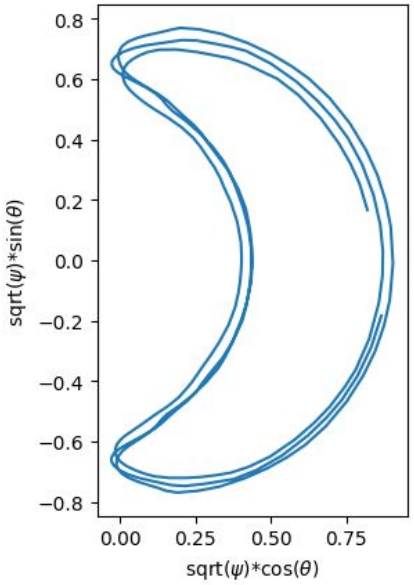
- GPU has advantage in doing the same compute many times
- *Integrate lots of particle trajectories in DESC (guiding center equations of motion)*
- JAX autodiff: Jacobian of particle trajectories wrt equilibrium

$$\dot{\psi} = \frac{m}{qB^3}\left(\frac{v_\perp^2}{2} + v_\parallel^2\right)\boldsymbol{B}\times\nabla B\cdot\nabla\psi$$

$$\dot{\theta} = \frac{v_\parallel}{B}\boldsymbol{B}\cdot\nabla\theta\frac{m}{qB^3}\left(\frac{v_\perp^2}{2} + v_\parallel^2\right)\boldsymbol{B}\times\nabla B\cdot\nabla\theta$$

$$\dot{\zeta} = \frac{v_\parallel}{B}\boldsymbol{B}\cdot\nabla\zeta$$

$$\dot{v}_\parallel = -\frac{v_\perp^2}{2B}\left(\frac{\boldsymbol{B}}{B} + \frac{mv_\perp^2}{2qB^3}\frac{1}{v_\parallel}\boldsymbol{B}\times\nabla B\right)\cdot\nabla B$$

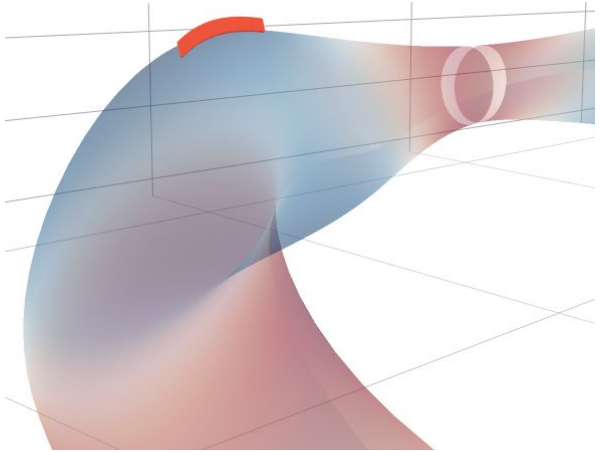


Plasma Control

Direct Optimization for Fast Particle Confinement

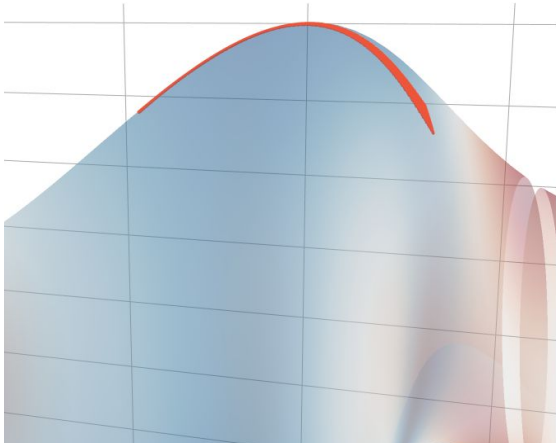
- Integrate Guiding Center EoM directly
- Consider entire trajectory, not just when particle leaves - more info per particle
- Optimize the equilibrium from particle's trajectories using JAX autodiff

Starting Equilibrium

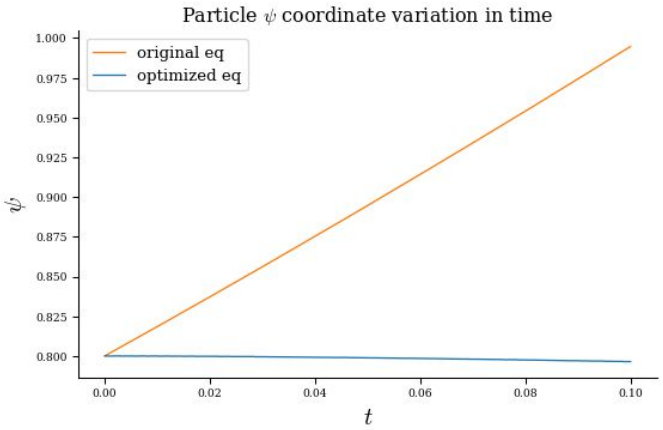


Particle drifting from flux surface ψ

Optimized Equilibrium



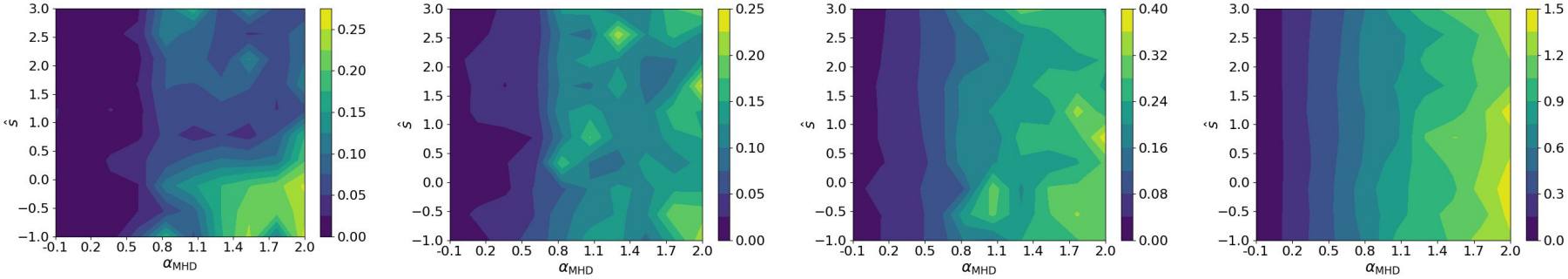
Particle confined in flux surface ψ



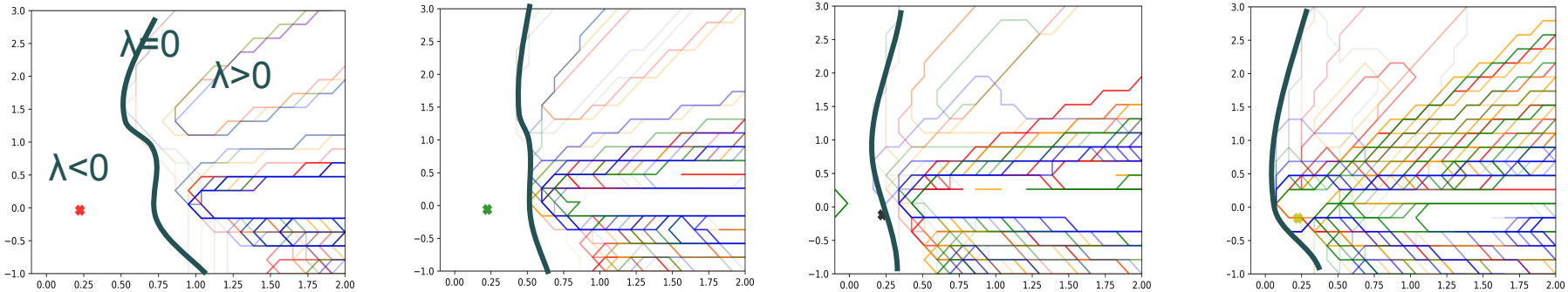
Plasma Control

Strong correlation between ideal and kinetic ballooning instability (R. Gaur, PU)

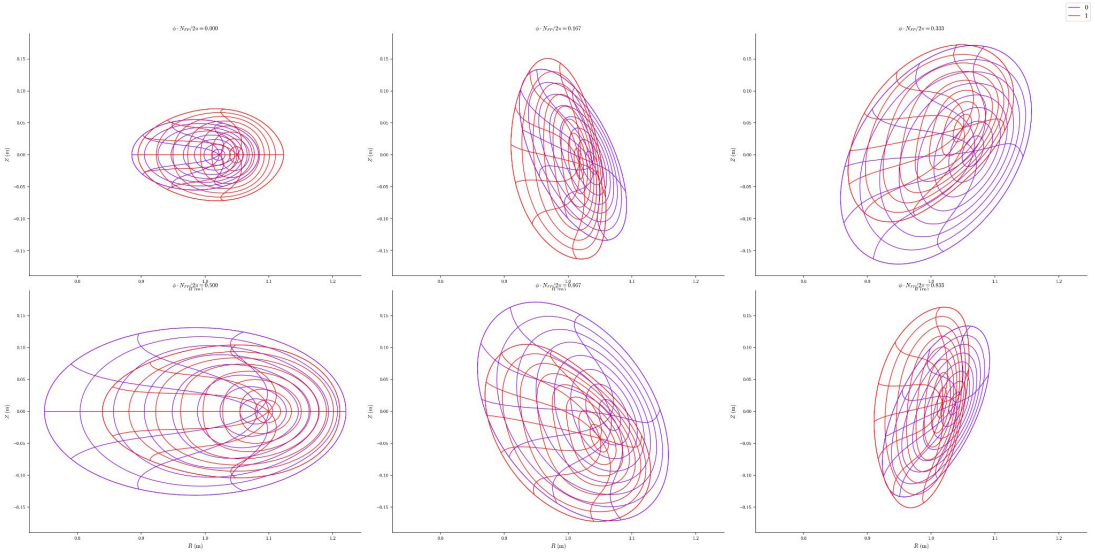
Maximum KBM growth rate (λ_KBM) s-alpha scan for a high-beta W7X equilibrium at four radii

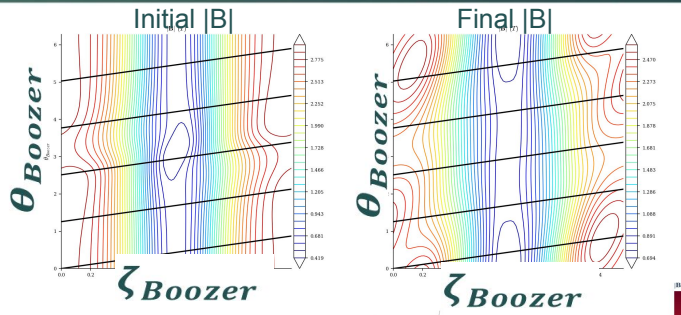


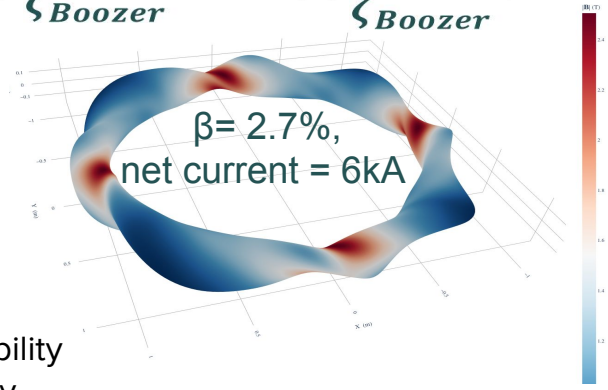
Ideal-ballooning marginal stability boundary for the same equilibrium $\frac{d}{d\theta}\left(g\frac{dX}{d\theta}\right) + cX = \lambda f X, \lambda = -\gamma^2$

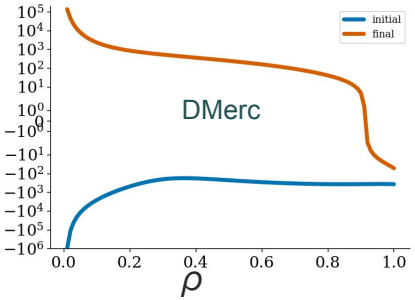


Finding ideal ballooning, Mercier stable, omnigenous equilibria with enhanced KBM stability (OP case)









Objectives:

- Ballooning stability
- Mercier stability
- Omnigenity
- Elongation
- Aspect Ratio
- Curvature

Plasma Control

Improving methods for optimization with constraints

Current optimization methods used in the field: Sum of Squares

Combine equality + inequality constraints

$$\min_x f(x) + w_1 [g_{eq}(x)]^2 + w_2 \max(0, g_{ineq}(x)])^2$$

Limitations:

- **Hard to guess a-priori what weights should be**
- **Only satisfies constraints approximately**
- **Max term for inequality constraints is non-smooth**

Better methods: Augmented Lagrangian

- **Combination of traditional Lagrangian + quadratic penalty**

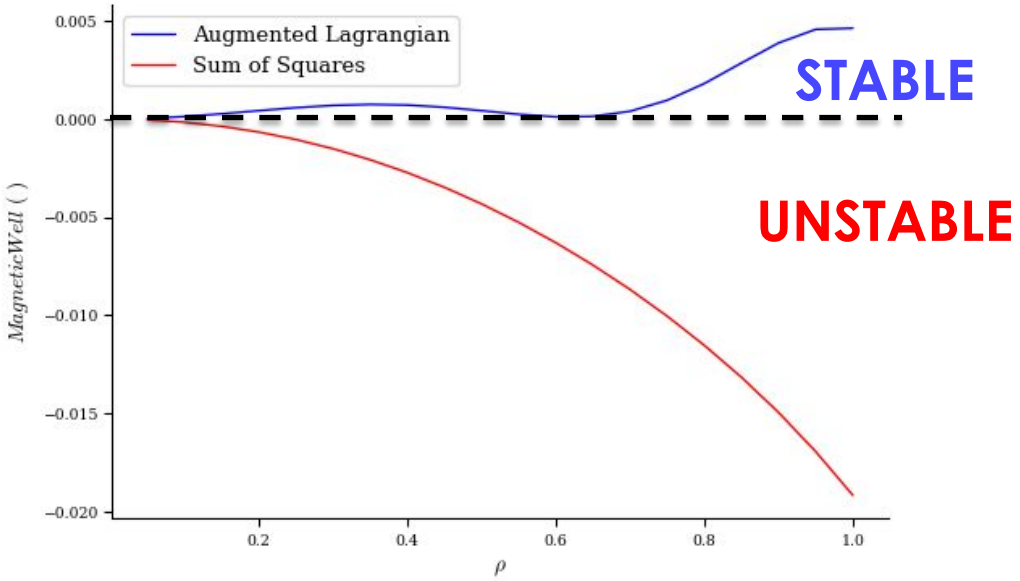
$$\mathcal{L}(x, \lambda, \mu) = f(x) + \lambda^T \mathbf{g}(x) + \mu g^2(x)$$

- **Doesn't introduce any non-smooth terms**
- **No need to guess correct weights through expensive trial and error**
- **"Exact" method - doesn't need $\mu \to$ infinity**
- **Solve sequence of subproblems for varying μ, λ - algorithm automatically chooses values**
- **Provides estimate of true Lagrange multipliers - useful information about trade-offs**

- **Open source packages available (LANCELOT, GALAHAD, NLopt, etc). Also python/JAX version implemented in DESC**

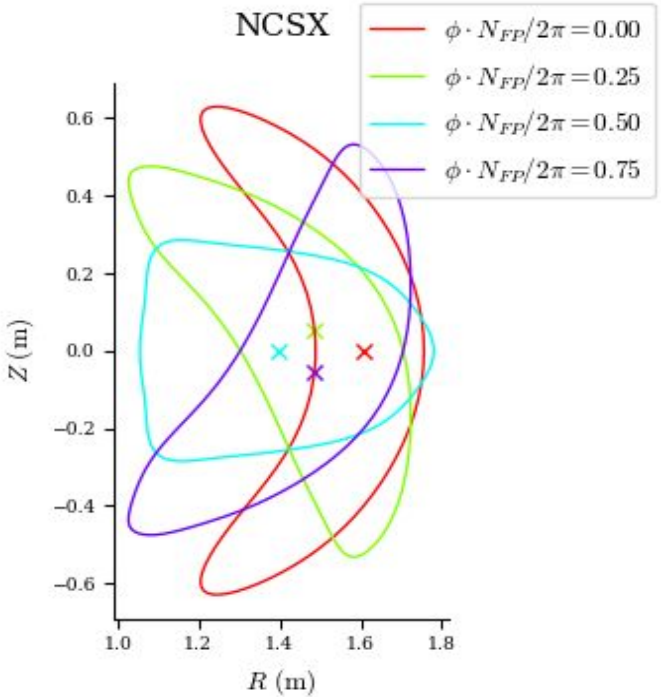
Augmented Lagrangian takes guesswork out of penalty terms

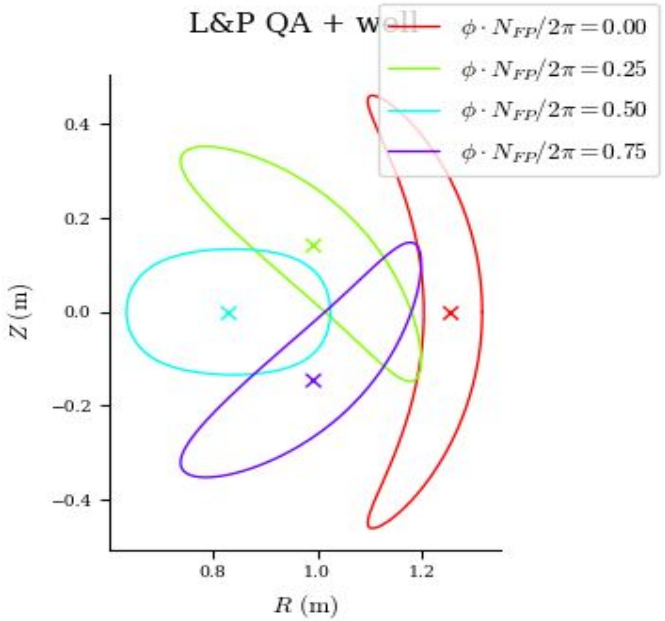
- **Simple quadratic penalty fails to give stable equilibrium, even for large values of weight**
- **Instead applying inequality constraint w/ augmented Langrangian gives magnetic well > 0**



Plasma Control

Can we de-bean optimized stellarators?





- NCSX + L&P precise QA: Magnetic well, but large concave regions
- Can be hard to create concave shaping with coils far from plasma

Can we do better?

Plasma Control

Application of Constrained Optimization in DESC

Minimize f_{QS} ("two term" metric)

$$f_C = (M\iota - N)(\mathbf{B} \times \nabla\psi) \cdot \nabla B - (MG + NI)\mathbf{B} \cdot \nabla B$$

subject to:

Force balance: $\mathbf{J} \times \mathbf{B} - \nabla p = 0$

Iota: $0.43 < \iota(\rho) < 0.5$

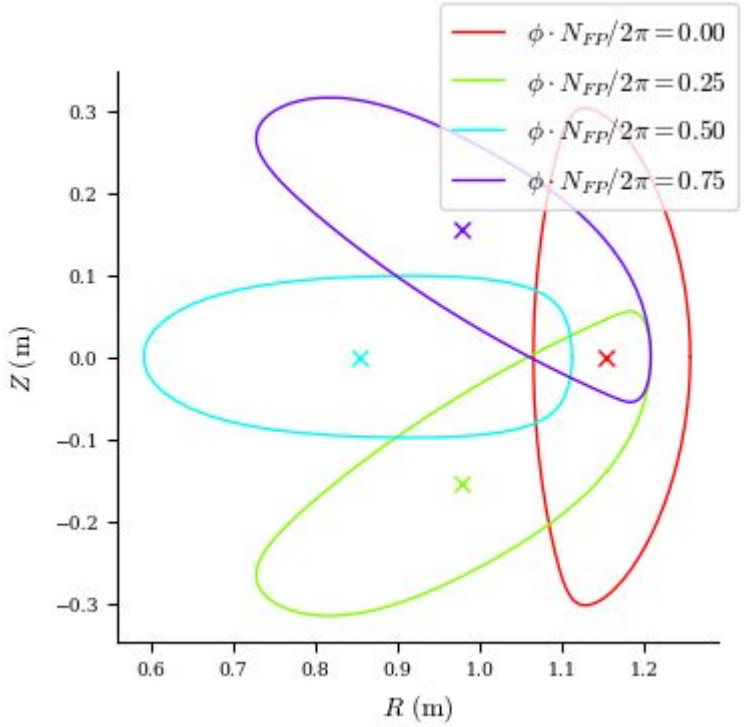
Major/Minor radii: $R_0 = 1\ \mathrm{m}$, $R_0/a = 6$

Magnetic well: $\dfrac{\partial_\rho V(2\mu_0 \partial_\rho p + \partial_\rho \langle B^2 \rangle)}{V\langle B^2 \rangle} > 0$

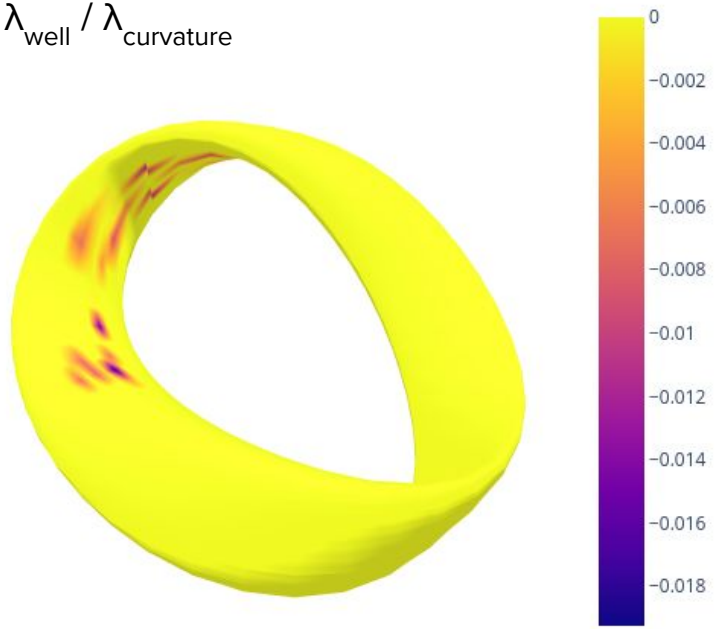
Mean Curvature: $H = (\kappa_1 + \kappa_2)/2 < 0$

κ_1, κ_2 eigenvalues of second fundamental form

Avoids concavity! But MHD unstable - Lagrange Multipliers say why

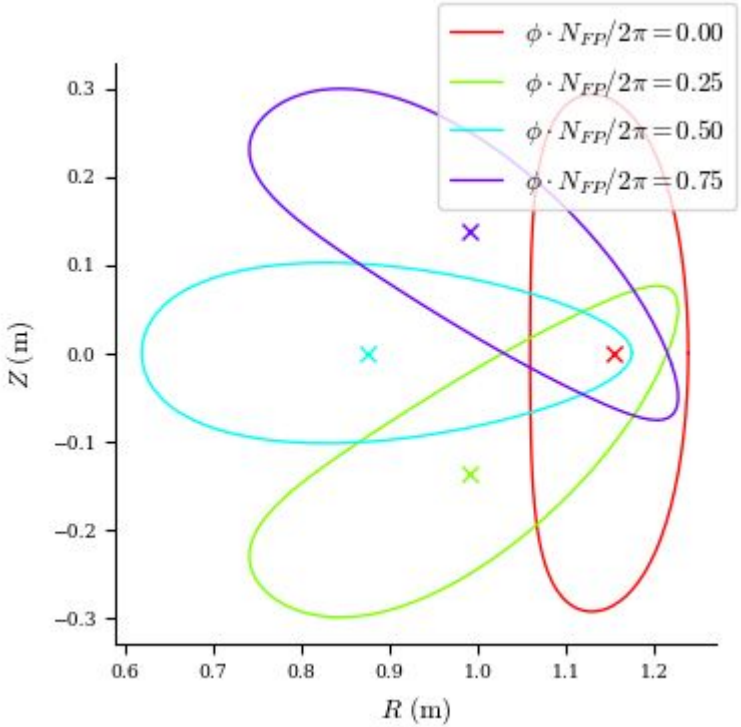


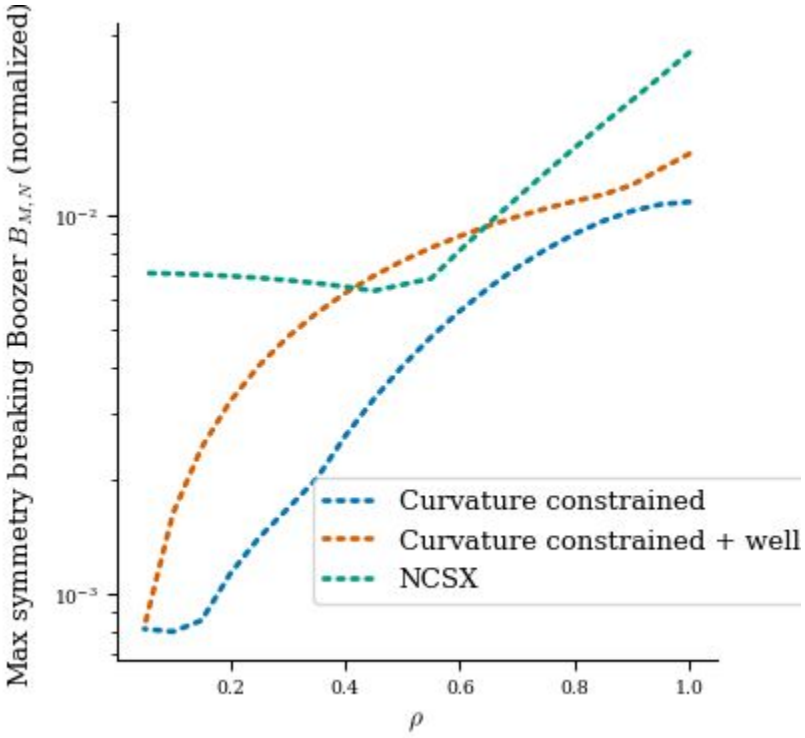
λ_{well} / $\lambda_{curvature}$



Plasma Control

Relaxing binding constraint achieves MHD stability with minimal sacrifice



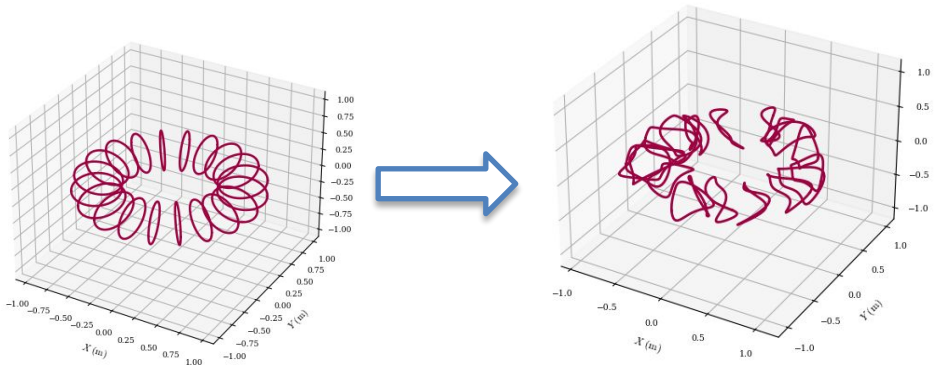


Stage 2- Coil Optimization

- **During the fixed-boundary equilibrium solve (and optimization) (in DESC), the boundary surface of the equilibrium is assumed to be a flux surface (so B·n=0)**
  - **however, DESC has no knowledge of the coils external to this equilibrium, so we must find coils that make this true**
- **Problem then: Find coils to minimize normal field on surface**

$$\chi_B^2 = \int \mathrm{d}^2 a \; B_{\text{normal}}^2$$

Example Coil Optimization



Plasma Control

REGCOIL Algorithm Implemented in DESC (D. Panici, PU)

- **Using surface current distributions on a specified winding is an efficient approach to the coil-finding problem[4,5]**

$$\boldsymbol{K} = \boldsymbol{n} \times \nabla \Phi \qquad \Phi(\theta', \zeta') = \Phi_{sv}(\theta', \zeta') + \frac{G\zeta'}{2\pi} + \frac{I\theta'}{2\pi}$$

Surface Current Density | Unit Normal to Winding Surface | Current Potential | G: Surface Net Poloidal Current | I: Surface Net Toroidal Current

- **Then minimization of quadratic flux becomes a linear (in Φ_{sv}) least-squares problem, after expanding in Fourier Series (I,G, and other terms are known)**

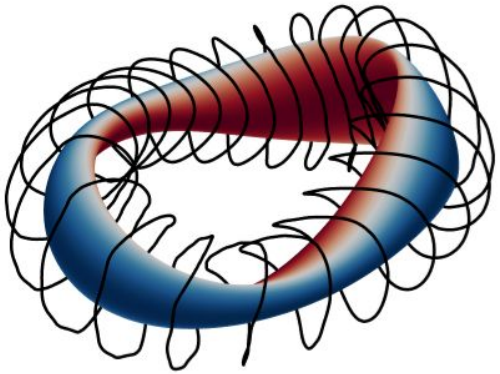
$$\chi_B^2 = \int \mathrm{d}^2 a \; B^2_{\text{normal}} \qquad \Phi_{sv} = \sum_{m,n} \Phi_{sv}^{mn} sin(m\theta' - n\zeta') \qquad B_n = B_n^{ext} + B_n^{pl} + B_n^{GI} + A\Phi_{sv}^{mn}$$

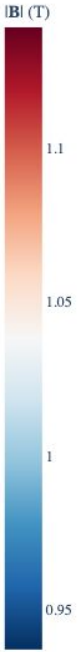
- **However, can lead to poor solutions without regularization -> REGCOIL adds regularization to the problem**

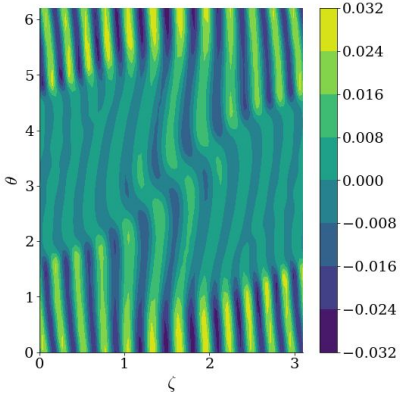
$$\chi_K^2 = \int \mathrm{d}^2 a' \; K(\theta', \zeta')^2$$

Plasma Control

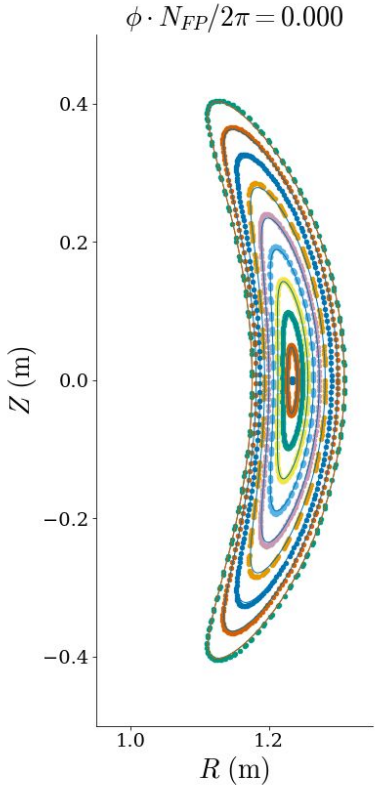
Modular Coil Optimization for precise QA in DESC with REGCOIL





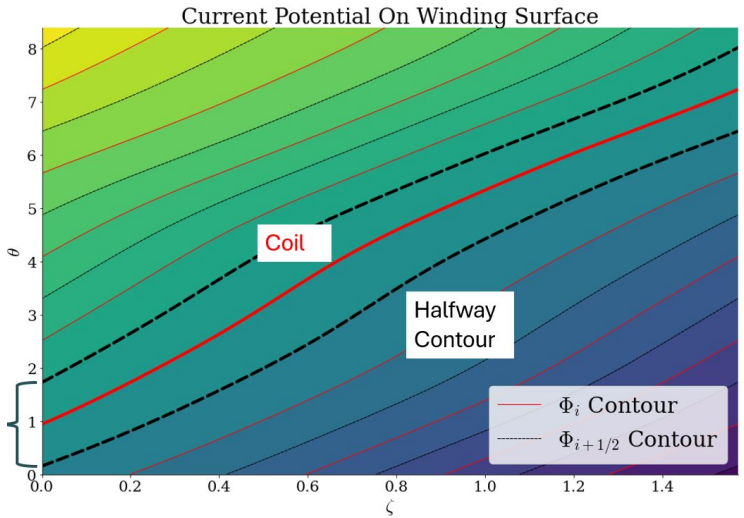


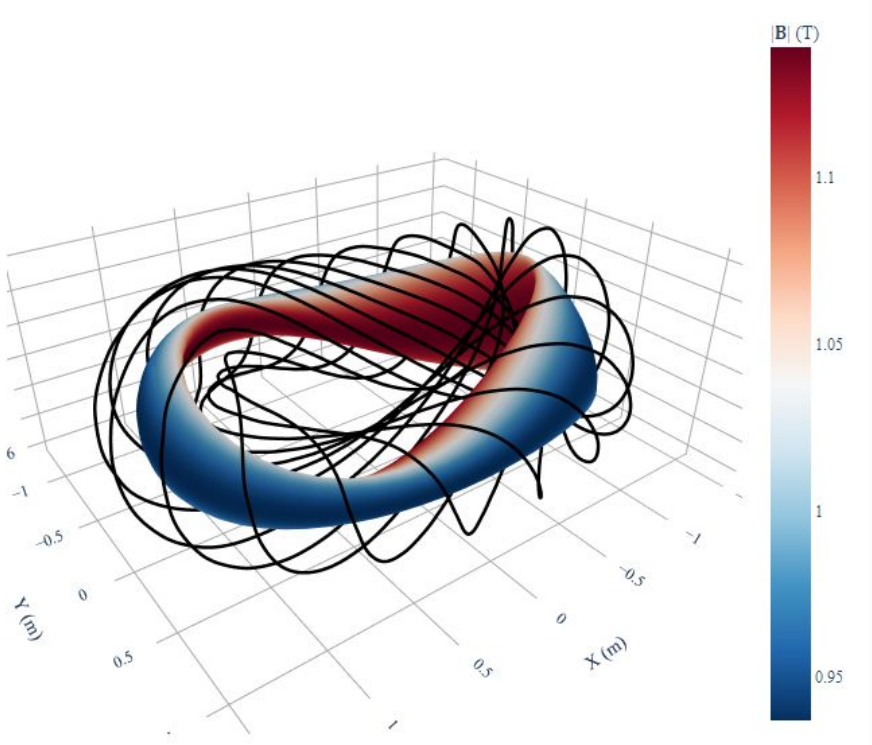
Biot-Savart Law Field Tracing



Plasma Control

Helical Coilset in DESC





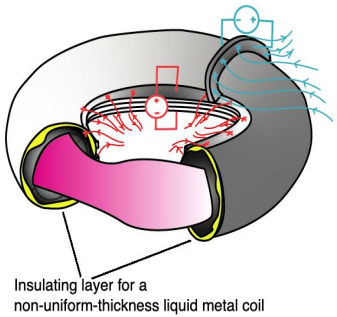
REGCOIL Algorithm implemented in DESC to find surface currents

Algorithms to discretize into coils also implemented

Filamentary coils coming soon! (next week?)

Single Coil Winding Surface (F. Saenz, PU)

- **Using surface current distributions on a fixed winding surface to find electrical conductivity-voltage fields**



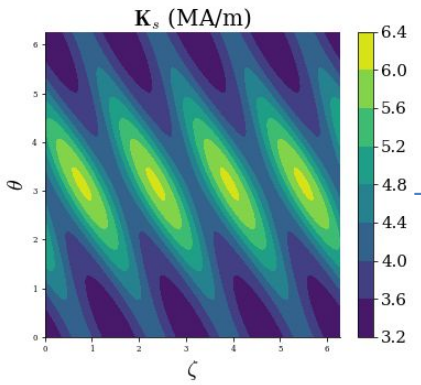
$$\Phi(\theta', \zeta') = \Phi_{sv}(\theta', \zeta') + \frac{G\zeta'}{2\pi} + \frac{I\theta'}{2\pi}$$

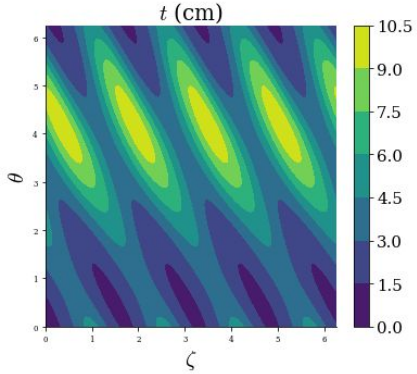
Current Potential

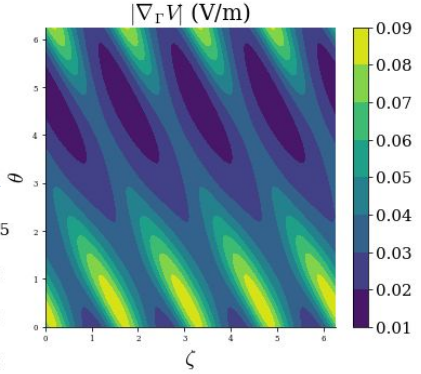
$$\mathbf{K}_s = \mathbf{n} \times \nabla_\Gamma \Phi = \sigma \nabla_\Gamma V$$

$\mathbf{K}_s$: Surface Current Density; $\mathbf{n}$: Unit Normal to Winding Surface; σ: Thickness-average conductivity; $\nabla_\Gamma V$: Voltage distribution

- **Thickness-average conductivity is inversely proportional to the local thickness *t***
  - Find thickness distributions for coil made of a single material.





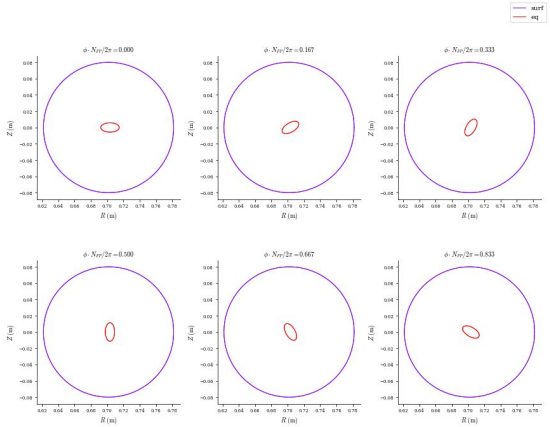


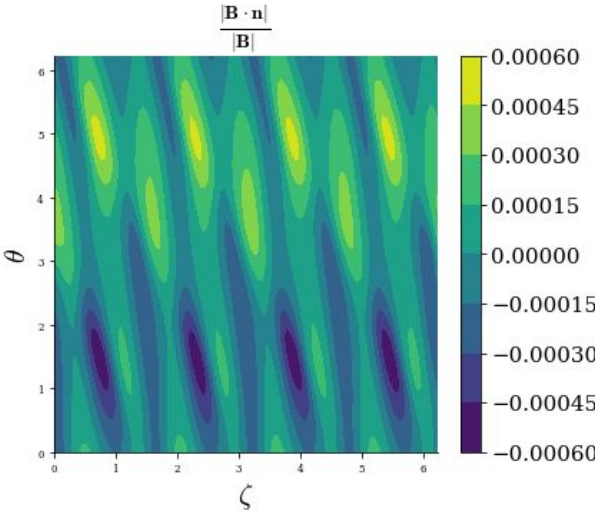
- **Instead of thickness distribution, the winding surface could be split into small patches of different conductivity.**

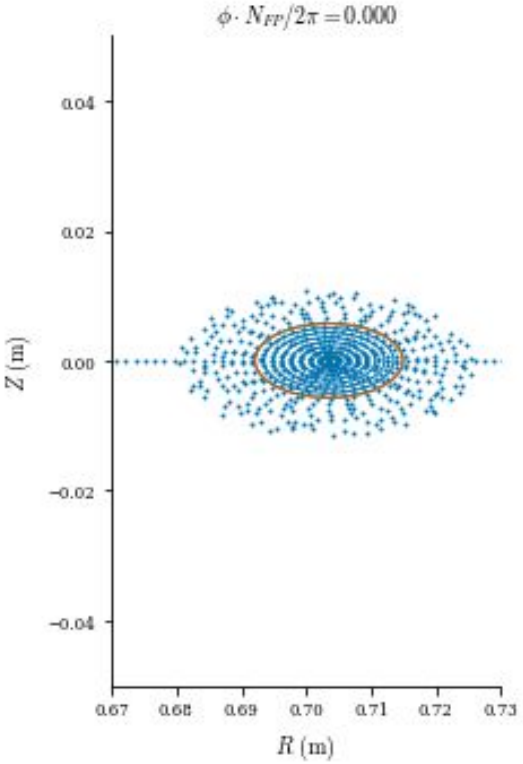
Plasma Control

Field Line Tracing from thickness/voltage fields

- **Single coil winding surface made of copper.**
  - Electrical conductivity: 58 MS/m
  - Max. current density: 50 MA/m^2
- **Flux surfaces obtained!**
  - Field calculated directly with Biot-Savart Law







Free Boundary DESC

Free boundary constraints:

Flux surface:

$$\mathbf{B} \cdot \mathbf{n} = 0$$

Pressure balance:

$$B_{\text{out}}^2 - B_{\text{in}}^2 - 2\mu_0 p = 0$$

Tangential jump:

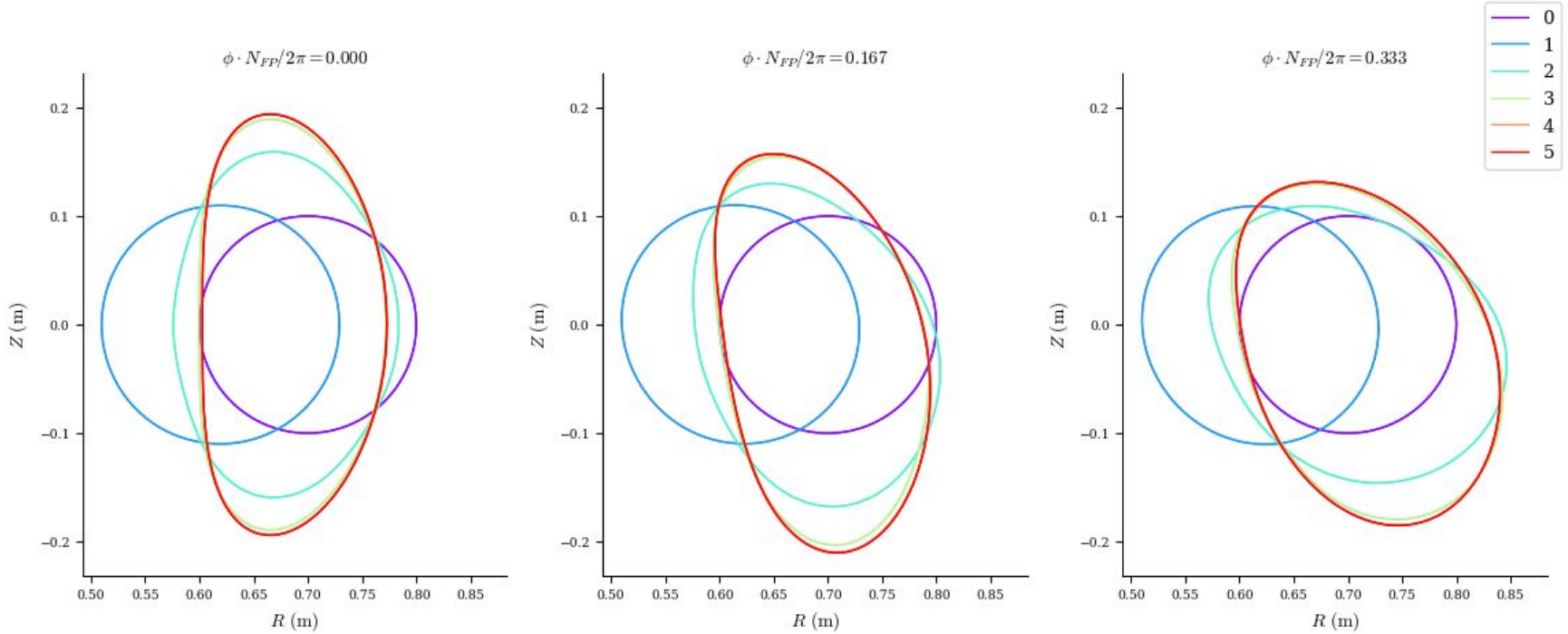
$$\mathbf{n} \times [\mathbf{B}_{\text{out}} - \mathbf{B}_{\text{in}}] - \mu_0 \mathbf{K} = 0$$

B_{in} = B from fixed boundary DESC

B_{out} = B_{ext} + B_{VC} + B_K

- **Parameterize unknown sheet current K, along with R, Z**
- **Use high order singular integration scheme (Malhotra et al 2019) to compute virtual casing + sheet current field B_{VC} + B_K**
- **Minimize all 3 equations simultaneously**
- **Flexible choices for external field**
  - **MGRID**
  - **Direct biot-savart from coils**
  - **Winding surface current potential**
  - **Dommaschk potentials**
  - **Any python callable**

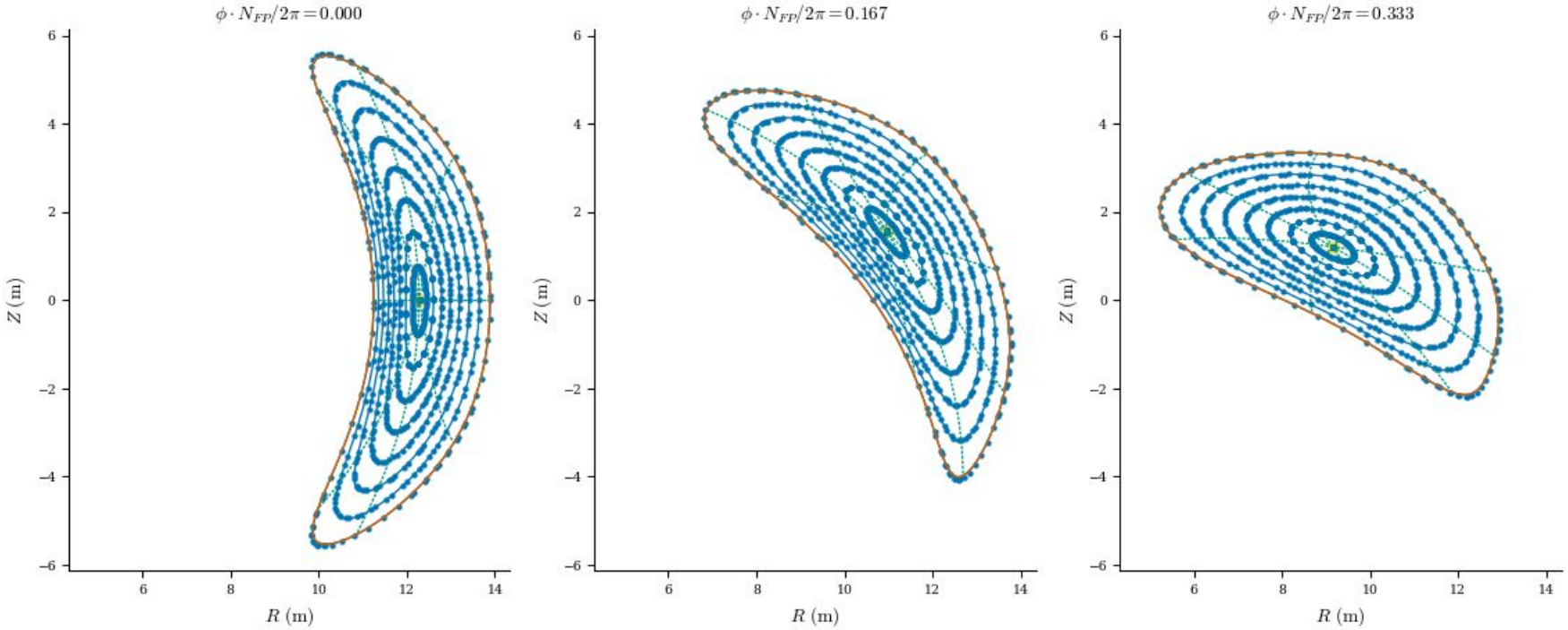
Converges for simple cases in as little as 5 steps



Plasma Control

Agrees with field line tracing for vacuum equilibria

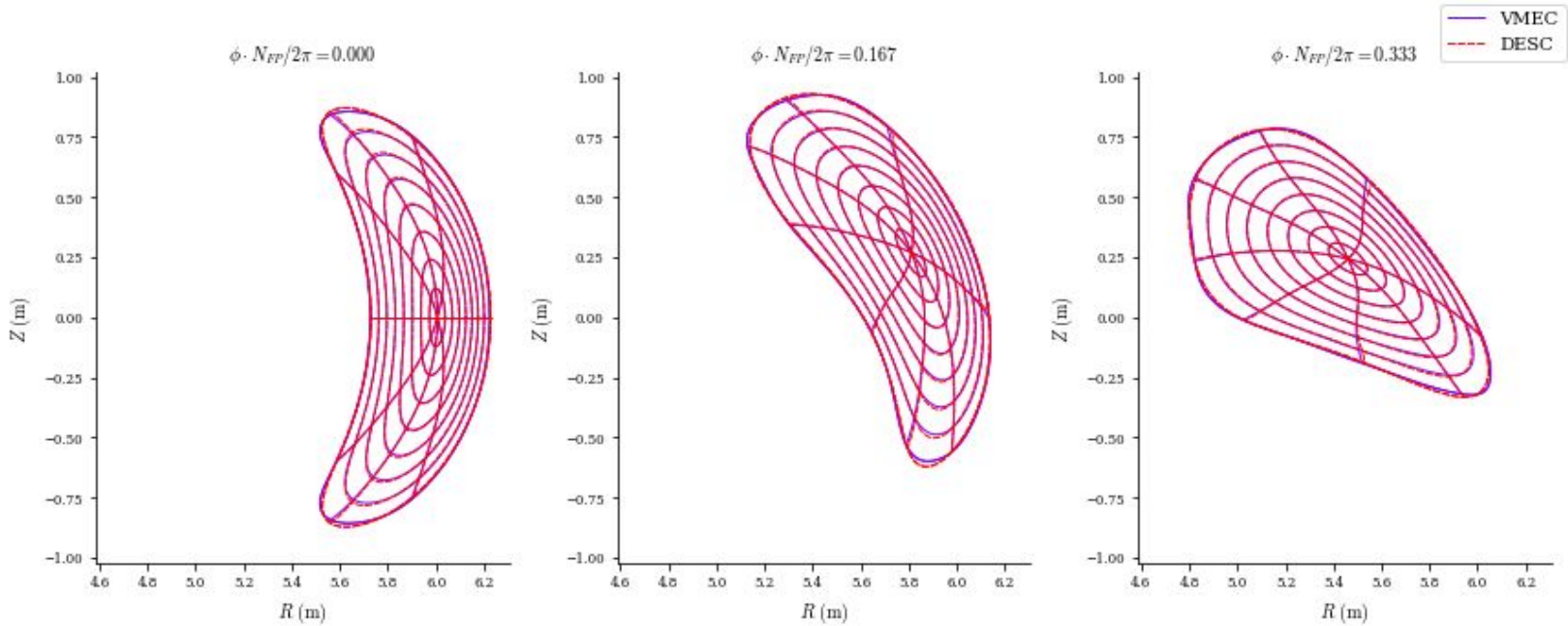
L&P precise QA, external field using direct Biot-Savart from coils



Plasma Control

Agrees with VMEC at finite beta

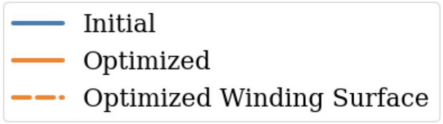
W7-X at β=2%, external field from MGRID

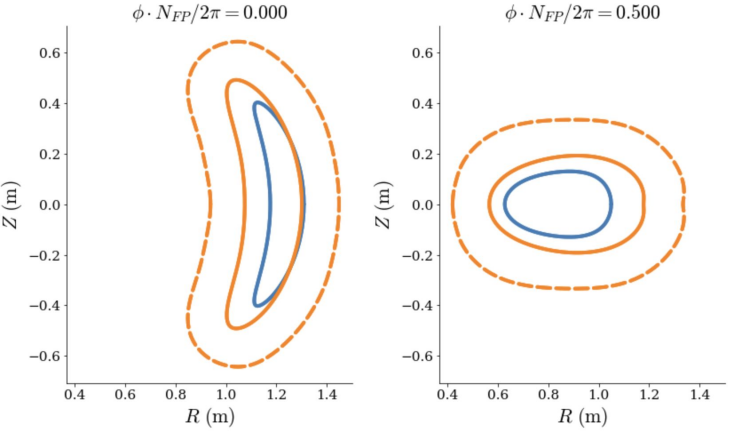


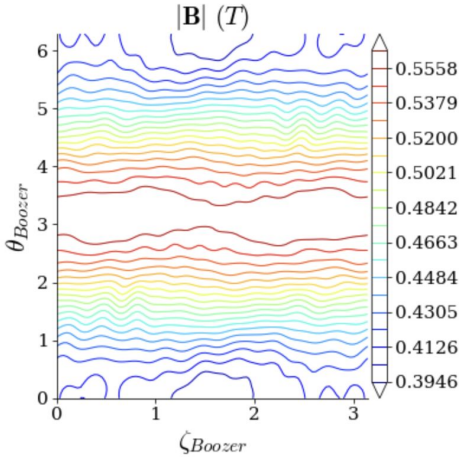
Plasma Control

Single-Stage Optimization with Winding Surface

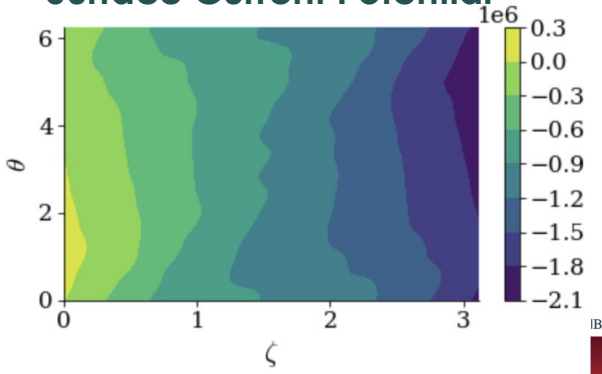
- Starting from precise QA, optimized smaller aspect ratio with QS target, along with quadratic flux target from surface current on winding surface, which was allowed to vary
- Aspect ratio lowered from 6 -> 4, with moderate QS degradation
- Max Bn error of ~7% after optimization



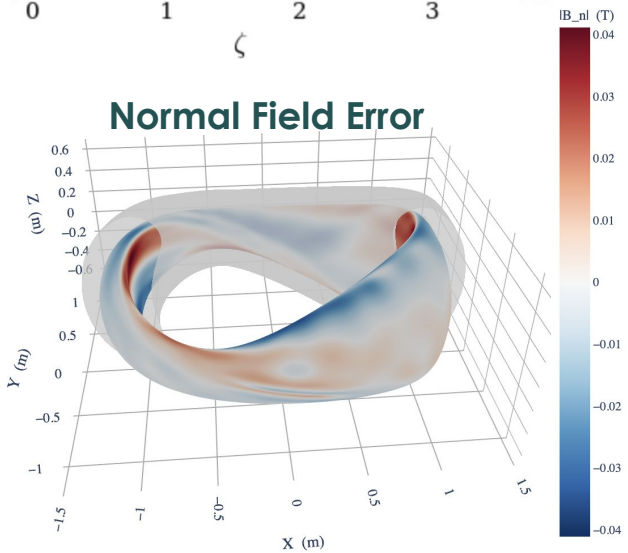




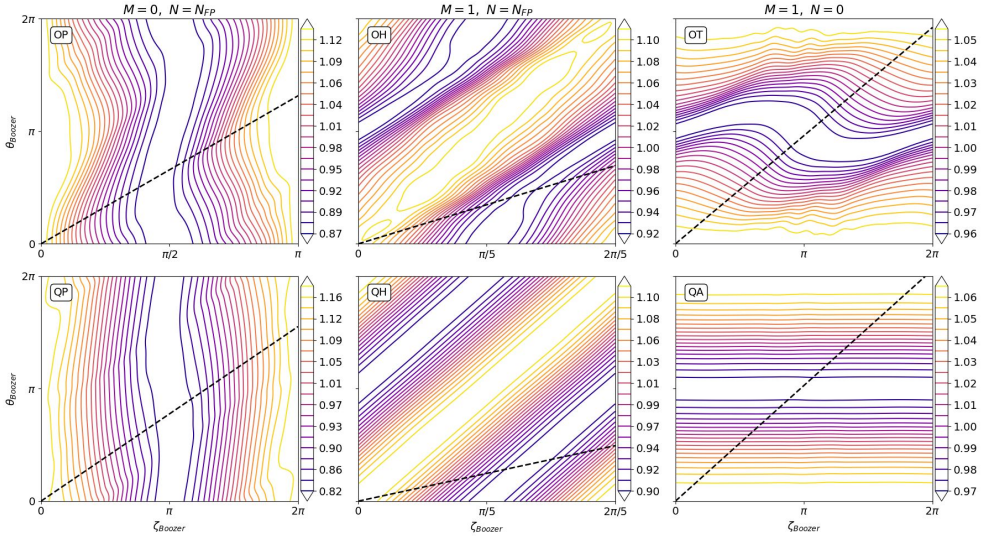
Surface Current Potential

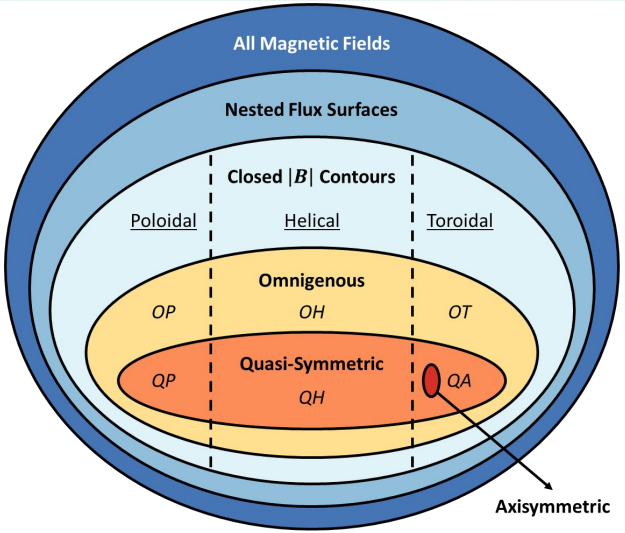


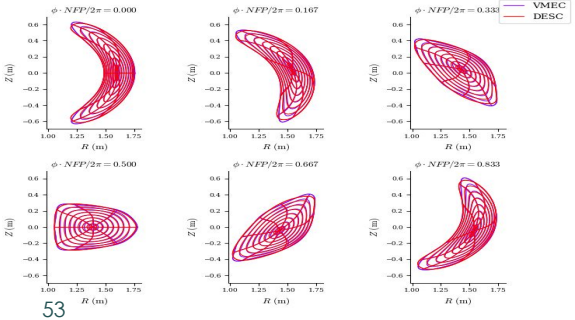
Normal Field Error

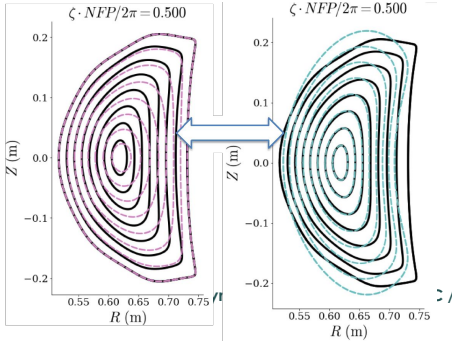


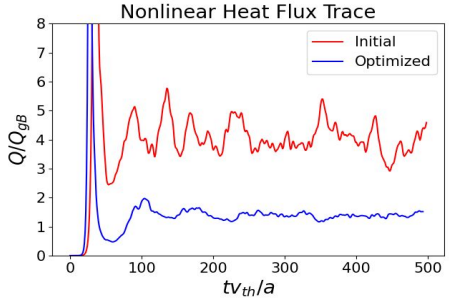
Conclusions: DESC multi-objective optimization applied to turbulence optimization, near axis, free surface, coil design, & general omnigenity











Plasma Control

Upcoming & Ongoing work

- Single stage optimization
- Optimization w/ filamentary coils
- Fast calculation of bounce integrals
  - ε_{eff}
  - Γ_c, Γ_α
- Equilibrium reconstruction
- Self consistent transport
- Generalized coordinates (radial, toroidal)
- Lots of software improvements
  - MPI, multi-node
- Many more…

Postdoc opportunities available: ekolemen@princeton.edu

Stellarator Optimization with DESC Automatic Differentiation (AD)

- Once an equilibrium solution is found, it may not be "good" in the sense of some physics objective g(x,c) (stability, particle confinement, etc)
- So, want to optimize the inputs to the problem to find solutions with improved objective values

Equilibrium Solution $x^* = [R_{lmn}, Z_{lmn}, \lambda_{lmn}]$ Inputs $c = [p, i, R_b, Z_b]$ Satisfy Force Balance $f(x^*, c) = 0$

- Want to optimize some objective g(x,c) wrt the inputs c
- Need derivative information!!

- Conventionally, must use finite differences and change c one element at a time, and resolve
- Takes len(c) equilibrium solves -> Expensive
- Finite differences are inaccurate

$$\frac{\partial g}{\partial c}$$

- DESC AD with JAX gives fast, accurate derivative information
- Obtain necessary derivatives with one equilibrium solve!

Plasma Control

Stellarator Optimization - Can optimize parameters while maintaining the equilibrium constraint

- Let $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{c})$ be a physics/engineering objective that we wish to optimize

Taylor expansion:

$$\boldsymbol{g}(\boldsymbol{x}+\Delta\boldsymbol{x}, \boldsymbol{c}+\Delta\boldsymbol{c}) = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{c}) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\Delta\boldsymbol{x} + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{c}}\Delta\boldsymbol{c}$$

$\boldsymbol{g}(\boldsymbol{x}+\Delta\boldsymbol{x}, \boldsymbol{c}+\Delta\boldsymbol{c}) \to 0$

equilibrium perturbation

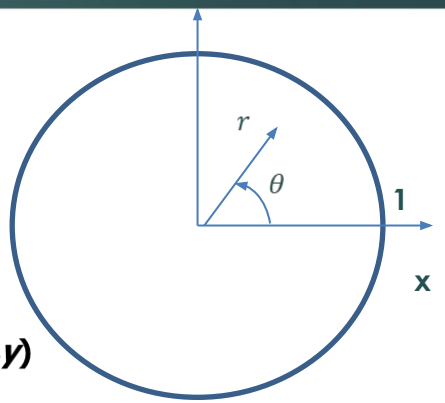
least-squares problem:

$$\left[\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right)^{-1}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{c}} - \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{c}}\right]\Delta\boldsymbol{c} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{c})$$

- Yields the optimal perturbation to improve the objective $\boldsymbol{c}^* = \boldsymbol{c} + \Delta\boldsymbol{c}$
- Extended to higher-order approximations with little additional cost
- Used in an adaptive Gauss-Newton trust-region optimization method

Analyticity Constraint at Polar Axis Proof

- **Assume $f(r,\theta)$ is a physical scalar, regular at r=0**
- **Expand in a Fourier Series: $\sum_{m=-\infty}^{\infty} a_{m(r)} e^{im\theta} = \sum_{-\infty}^{\infty} f_m(r,\theta)$**
  - **Where the Fourier coefficients are a function of polar radius *r***
- **Assume each $f_m(r,\theta)$ is a regular function of *(x,y)* at r=0**
- **Notice that $e^{im\theta}$ is NOT regular at r=0 (it is multi-valued)**
- **But, $\left[re^{\pm im\theta}\right]^{|m|} = [x \pm iy]^{|m|}$ is a regular function of (*x,y*) b/c it is a <u>polynomial</u> in (*x,y*)**
- **We can rewrite $f(r,\theta)$ as**



[1](Lewis and Bellan, 1990)

$$f_m(r,\theta) = a_m(r)e^{im\theta}$$

$$= \frac{a_m(r)}{r^{|m|}} r^{|m|} e^{i\theta m}$$

$$= \frac{a_m(r)}{r^{|m|}} \left[re^{\pm i\theta}\right]^{|m|} \begin{cases} + & m > 0 \\ - & m < 0 \end{cases}$$

58

Regular at r=0

$$f_m(r,\theta) = \frac{a_m(r)}{r^{|m|}} [x \pm iy]^{|m|}$$

Must be regular at r=0!

Regular at r=0

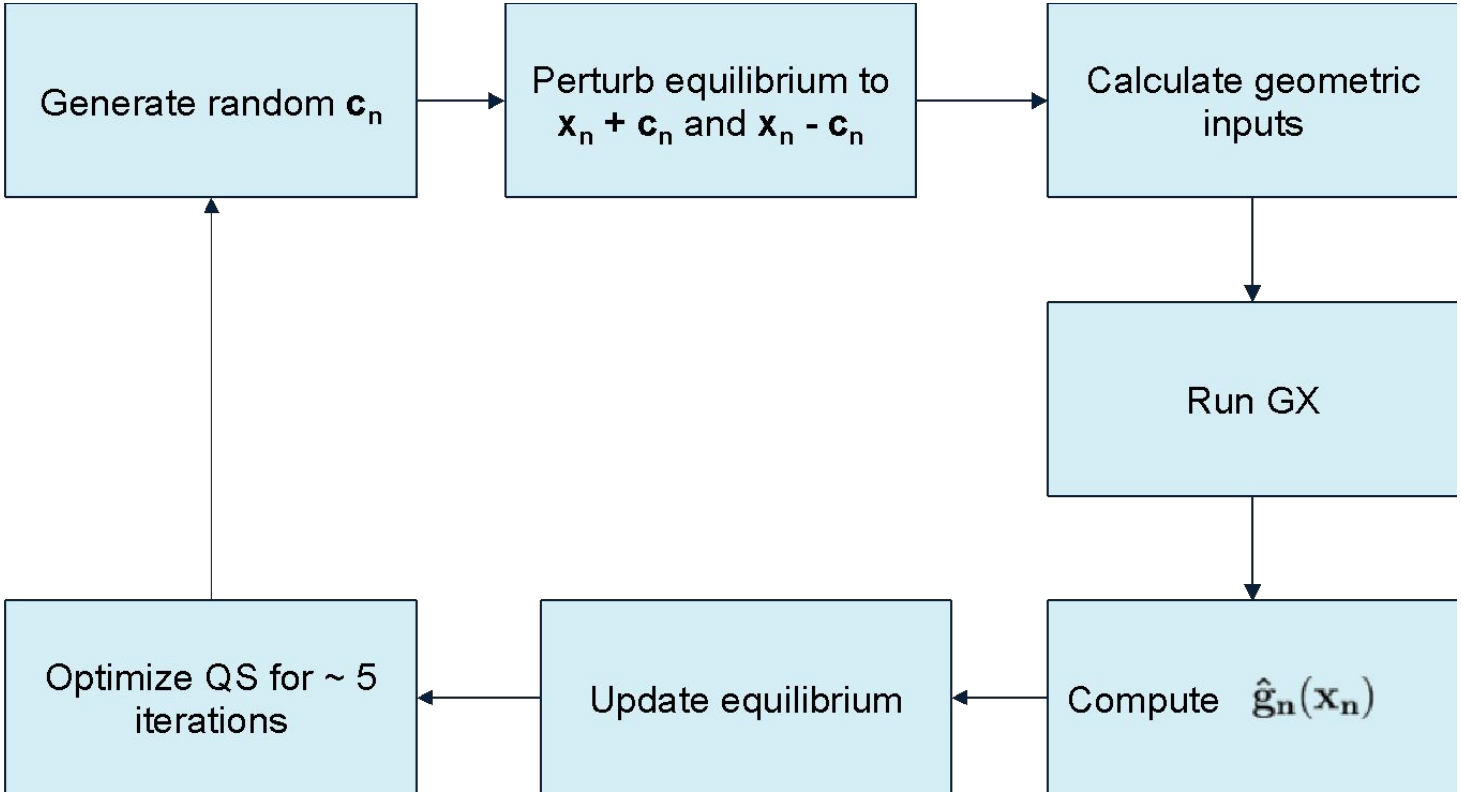
$$\lim_{r\to 0} \frac{a_m(r)}{r^{|m|}} < \infty$$

$a_m(r)$ must scale at least as $r^{|m|}$

$$a_m(r) \sim r^{|m|} + r^{|m|+2} \ldots$$

Plasma Control

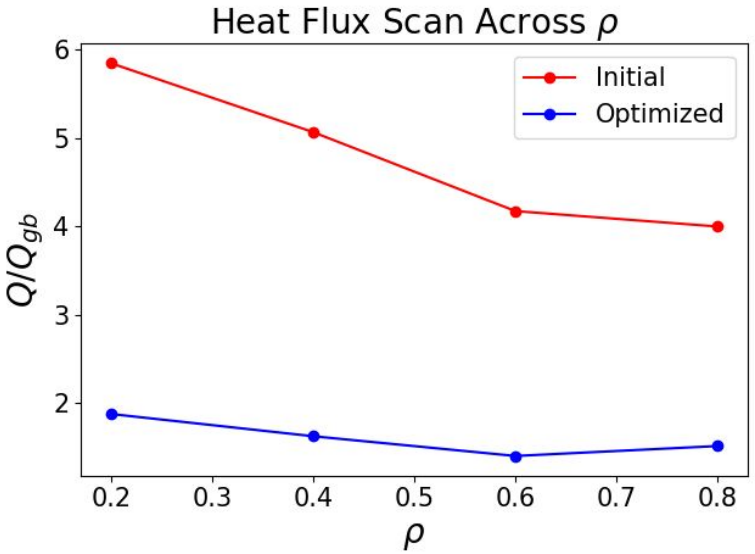
Simultaneous Perturbation Stochastic Approximation

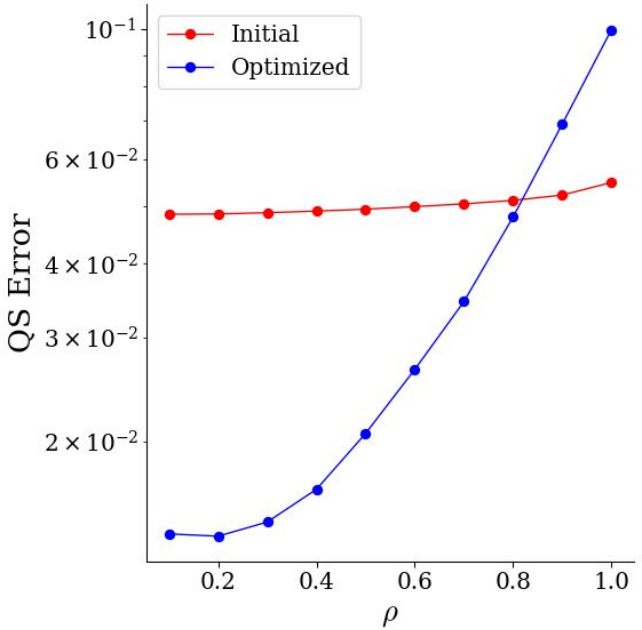


Plasma Control

Flux Surface Scan

- **Only heat flux at s = ψ/ψ_b = 0.5 is optimized.**
- **Heat flux is reduced over several different flux surfaces.**
- **QS error is also reduced over most of the volume**

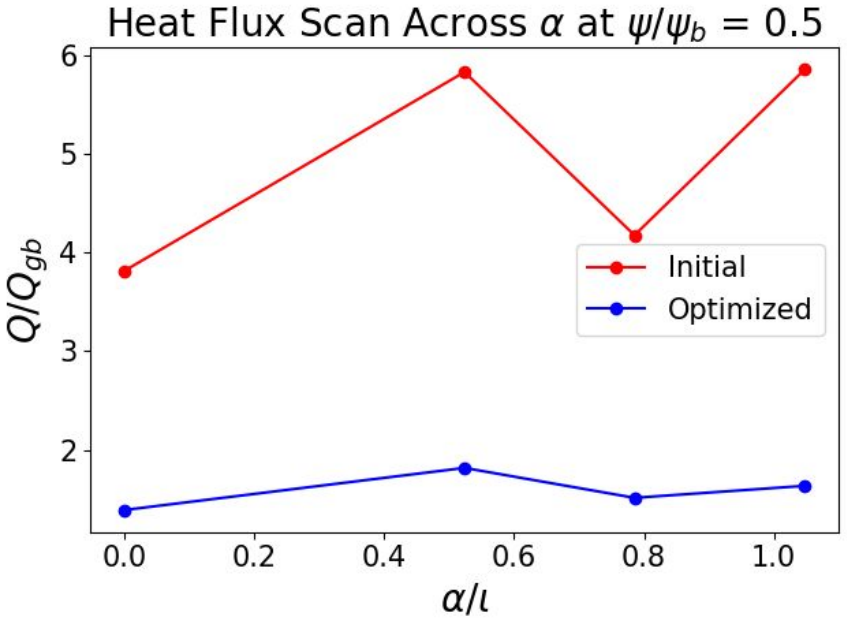




Plasma Control

Field-Line Label Scans

- **Only simulated at α = 0 field line.**
- **Scans show decreased heat fluxes across several field-lines.**

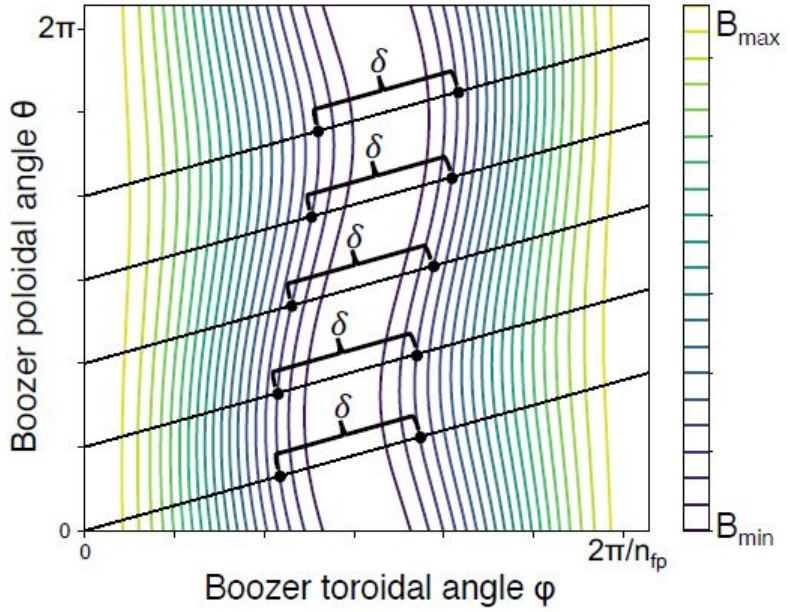


Plasma Control

Omnigenous magnetic fields

Quasi-Isodynamic (QI) magnetic fields = omnigenous magnetic fields with constant |B| contours that close poloidally

Particles in omnigenous magnetic fields have no net radial drifts



(Goodman 2022)

Conditions for Omnigenity:

- B_{max} is a straight contour in Boozer coordinates
- Constant "bounce distance" δ between consecutive points of equal B on each field line α

$$\delta = \sqrt{\Delta\theta_B^2 + \Delta\zeta_B^2} \propto \Delta\zeta_B \qquad \frac{\partial\delta}{\partial\alpha} = 0$$

Plasma Control

Omnigenity is parametrized by a coordinate mapping

- Map between computational and Boozer coordinates:

helicity = (M, N)

$$h(\rho, \alpha, \eta) = h(\theta_B, \zeta_B)$$

$$2\eta + \pi + \sum_{l=0}^{L_\rho} \sum_{m=0}^{M_\eta} \sum_{n=-N_\alpha}^{N_\alpha} x_{lmn} T_l(2\rho - 1) \mathcal{F}_m(\eta) \mathcal{F}_{nN_{FP}}(\alpha) = \begin{cases} N\zeta_B & \text{for } M = 0 \\ -\theta_B + \frac{N}{M}\zeta_B & \text{for } M \neq 0 \end{cases}$$

ρ = flux surface label
α = field line label
η = coord along field line

free parameters

Chebyshev polynomials

Fourier series

- Constant bounce distances are preserved:

$$\delta \propto \Delta h = h(\rho, +\eta, \alpha) - h(\rho, -\eta, \alpha)$$

$$= 4\eta + \sum_{l=0}^{L_\rho} \sum_{m=0}^{M_\eta} \sum_{n=-N_\alpha}^{N_\alpha} x_{lmn} \left[T_l(2\rho - 1) \mathcal{F}_{nN_{FP}}(\alpha) \underbrace{[\mathcal{F}_m(+\eta) - \mathcal{F}_m(-\eta)]}_{= 0 \text{ because } \sum_{m \geq 0} \mathcal{F}_m(\eta) \text{ is an even function of } \eta} \right]$$

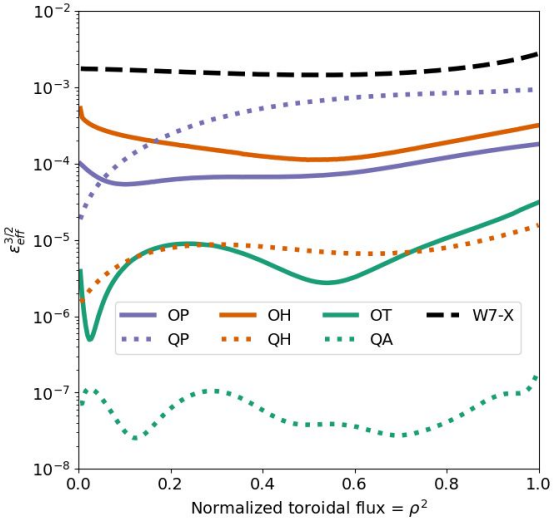
$$= 4\eta$$

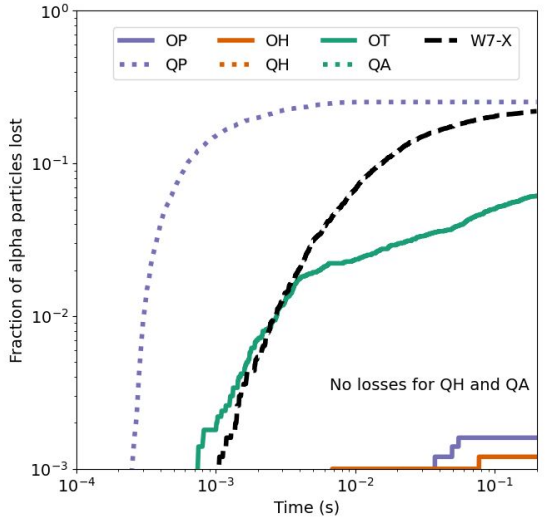
$$\therefore \frac{\partial \delta}{\partial \alpha} = 0$$

Plasma Control

Solutions show good particle confinement!

- **Top:** Neoclassical collisional transport magnitude
  - Computed by NEO
- **Bottom:** Collisionless losses of fusion-born alpha particles
  - Computed by SIMPLE
  - Particles initialized at $\rho = 0.5$
  - Configurations scaled to a and B of ARIES-CS
- Reference case is W7-X at $\beta = 4\%$
- Precise QP is difficult (impossible?) to achieve
- Higher alpha particle losses for OT case might be due to wide banana orbits?

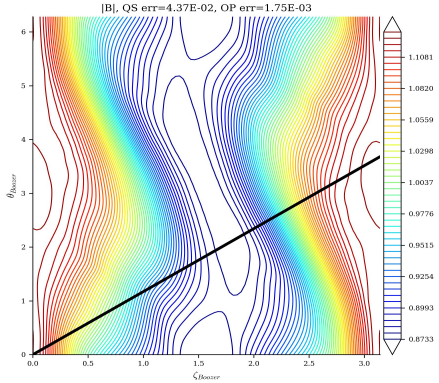




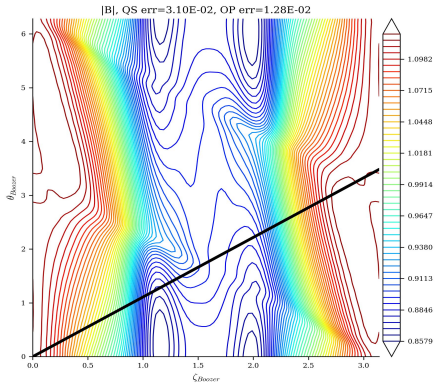
Creating a database of omnigenous equilibria using DESC

- Omnigenity is a favorable property that ensures confinement of thermal trapped particles in a stellarator.
- It is a superset of quasisymmetry (QS) and allows us to explore a much larger parameter space than QS
- Based on Dudt et al. (2023), we generate a database of omnigenous equilibria.
- We start with near-axis or global QA, QH, QI, and using them as seed equilibria, generate more quasisymmetric and omnigenous configurations
- Omnigenity requires closed |B| contours instead straight |B| contours in QS. QA, QH and QI -> OT, OH, OP.
- This provides a larger space of equilibria, advantageous for multi-objective, multi-constraint optimization

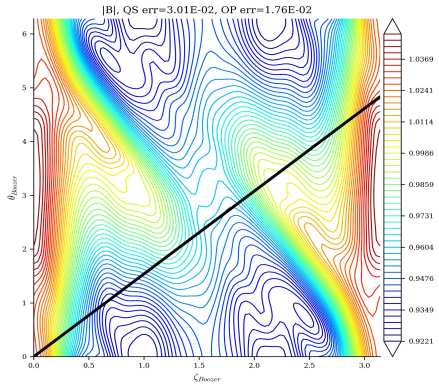
Generating OP and QI from known QI equilibria



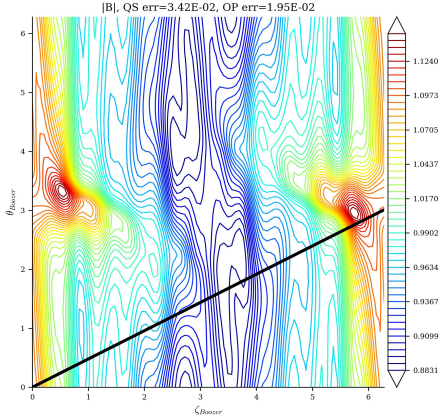
Dudt '23 OP (seed)



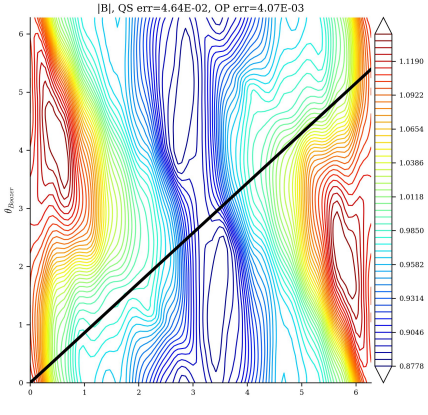
Derived OP equilibrium 1



Derived OP equilibrium 2



Plunk OP (seed equilibrium)



Derived OP equilibrium 1

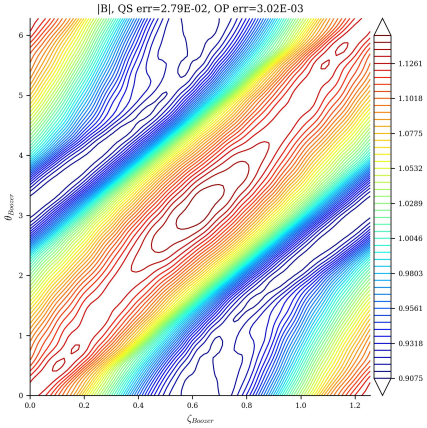
1. DESC is used to load a seed quasisymmetric equilibrium and generate new omnigenous equilibria.

2. Here we obtain OP equilibrium from known OP equilibria.

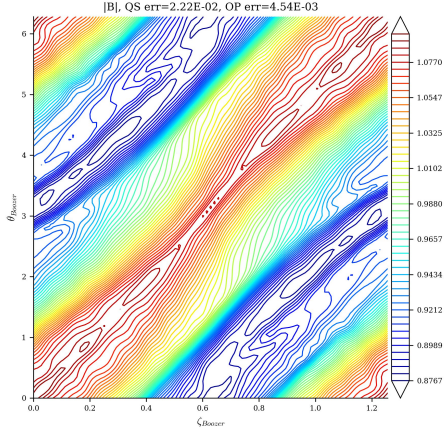
3. We fit the near-axis solution with DESC that allows imposing additional constraints

Plasma Control

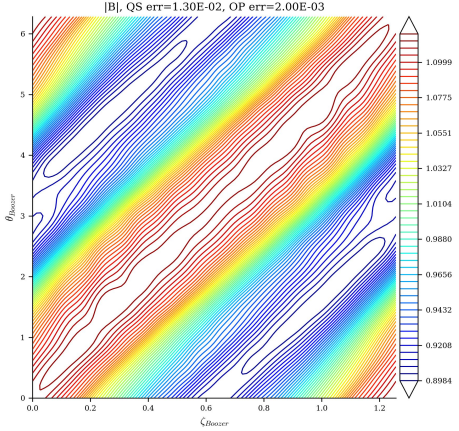
OH and OT from QH and QA equilibria



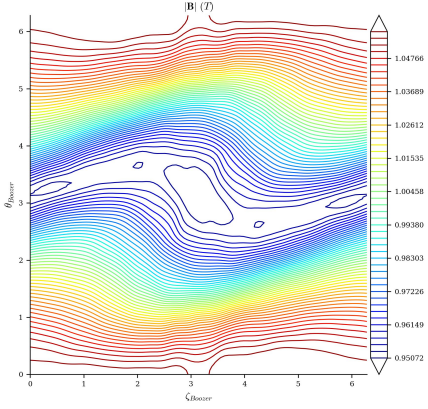
Dudt '23 OH from Landreman QH (seed)



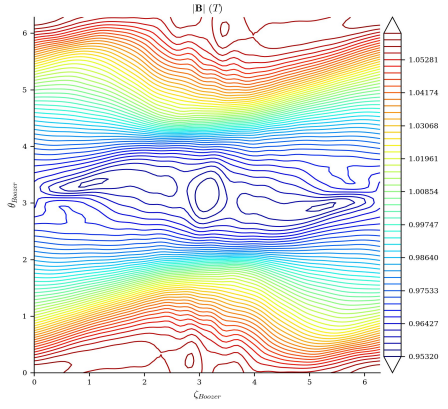
Derived OH equilibrium 1



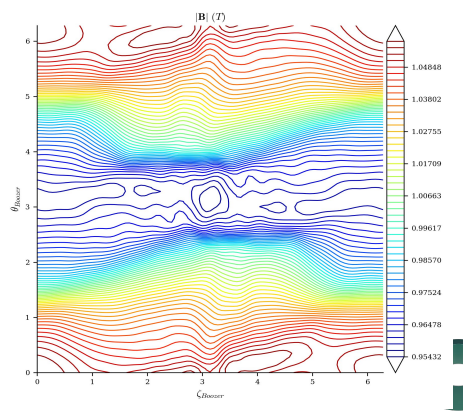
Derived OH equilibrium 2



Dudt '23 OT from PyQsc (seed)



Derived OT equilibrium 1



Derived OT equilibrium 2

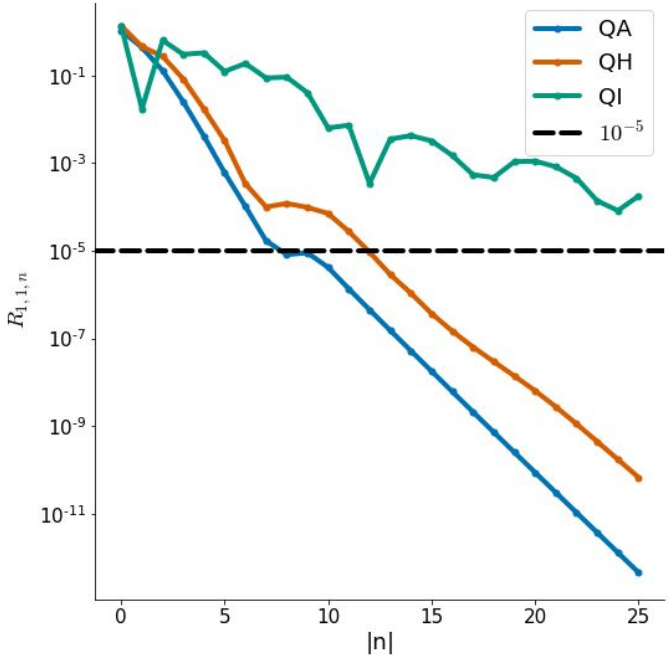
ch 21 2024

Plasma Control

Fitting NAE Behavior with Toroidal Fourier Series – O(ρ^1)

O(ρ) Coefficient $R_{1,1,n}$

$$R_1 = \mathcal{R}_{1,1}(\phi)\cos\theta + \mathcal{R}_{1,-1}(\phi)\sin\theta$$



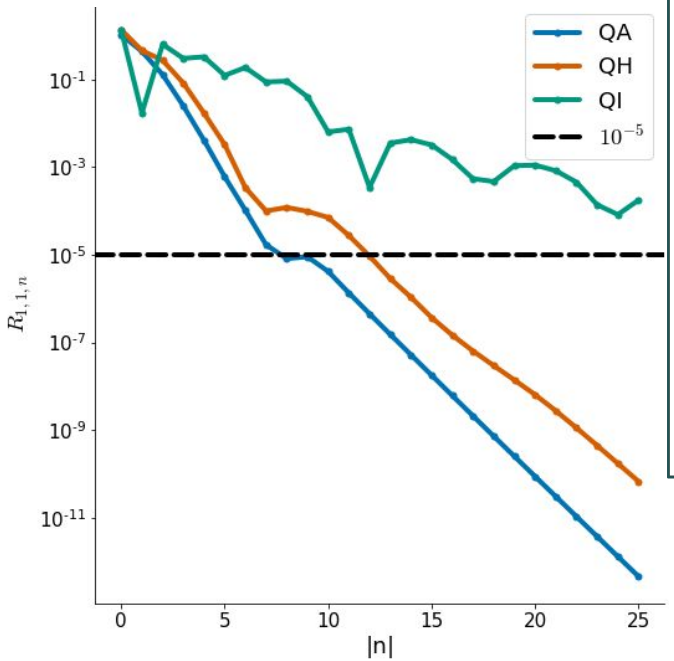
QA NAE behavior simplest to describe

QI NAE behavior very difficult to describe with cylindrical angle!

Plasma Control

Fitting NAE Behavior with Toroidal Fourier Series – O(ρ^2)

O(ρ) Coefficient $R_{1,1,n}$

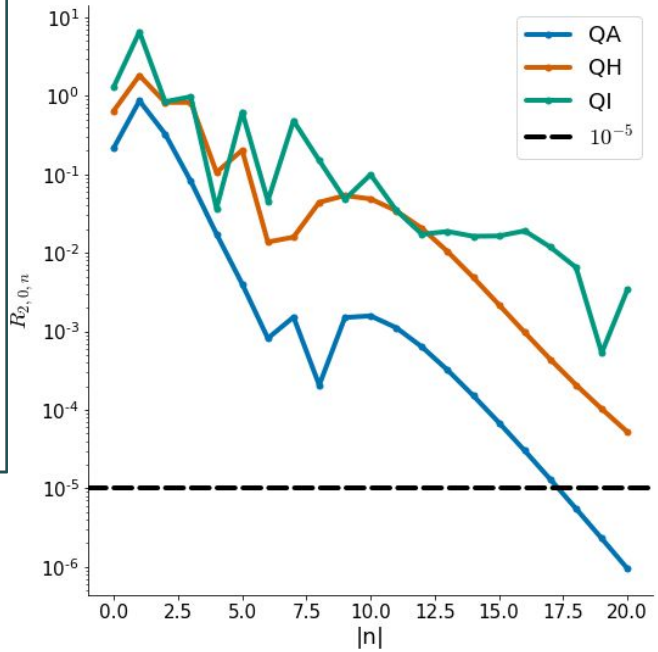


O(ρ) NAE behavior much easier to describe with cylindrical angle than O(ρ^2)

O(ρ^2) – constrained equilibria may require very high N_{tor} to capture behavior accurately

Generalized toroidal angle may help condense spectrum

O(ρ^2) Coefficient $R_{2,0,n}$



$$R_2 = \mathcal{R}_{2,0}(\phi) + \mathcal{R}_{2,2}(\phi)\cos 2\theta + \mathcal{R}_{2,-2}(\phi)\sin 2\theta$$

Plasma Control

$O(\rho^0)$ (axis) Constraint in DESC

NAE axis in pyQSC given as Fourier series in cylindrical toroidal angle ϕ:

$$R = R_0 + \sum_{n=1}^{N} (R_n^C \cos m\phi + R_n^S \sin m\phi) \qquad Z = \sum_{n=1}^{N} (Z_n^C \cos m\phi + Z_n^S \sin m\phi)$$

Constraint in DESC representation is simple: Evaluate DESC $R(\rho,\theta,\phi)$, $Z(\rho,\theta,\phi)$ at ρ=0 and match terms:

NAE Axis Coefficients

$$R_n^{C/S} = \sum_{k=0}^{\infty} (-1)^k R_{2k,0,\pm|n|}$$

$$Z_n^{C/S} = \sum_{k=0}^{\infty} (-1)^k Z_{2k,0,\pm|n|}$$

DESC Fourier-Zernike Coefficients

Plasma Control

O(ϱ^1) NAE Constraint in DESC

- After a short geometric derivation, one can derive (up to O(ϱ)) the R,Z position of a point on a flux surface from the NAE in terms of the cylindrical angle

$$\mathbf{r} \approx \mathbf{r}_0(\phi) + \rho R_1 \hat{\mathbf{R}} + \rho Z_1 \hat{\mathbf{Z}}$$

where

$$R_1 = \mathcal{R}_{1,1}(\phi)\cos\theta + \mathcal{R}_{1,-1}(\phi)\sin\theta \qquad Z_1 = Z_{1,1}(\phi)\cos\theta + Z_{1,-1}(\phi)\sin\theta$$

- And the coefficients are functions of the NAE X,Y coefficients and the Frenet-Serret basis vectors

- Then, equating the O(ϱ) coefficients in the DESC Fourier-Zernike basis with the above expressions yields:

(Identical expressions for Z as well)

NAE Coefficie

$$\mathcal{R}_{1,1,n} = -\sum_{k=1}^{M} (-1)^k k R_{2k-1,1,n},$$

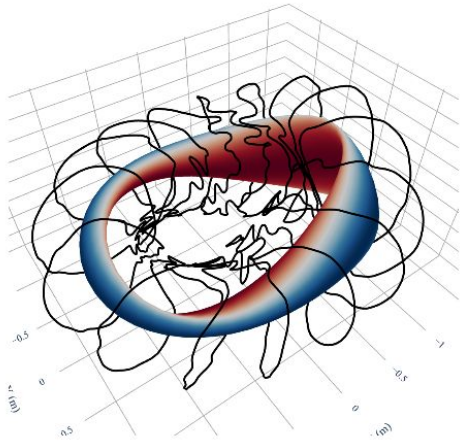
$$\mathcal{R}_{1,-1,n} = -\sum_{k=1}^{M} (-1)^k k R_{2k-1,-1,n},$$

DESC Fourier-Zernike Coefficients

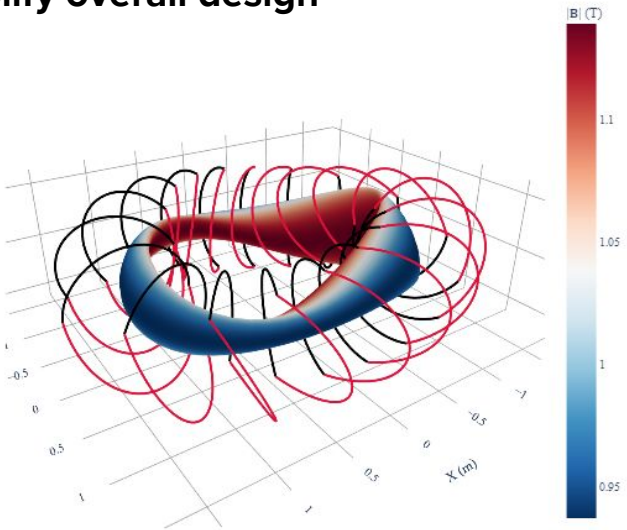
Plasma Control

Improvements to Coil Optimization - C^0 Coils?

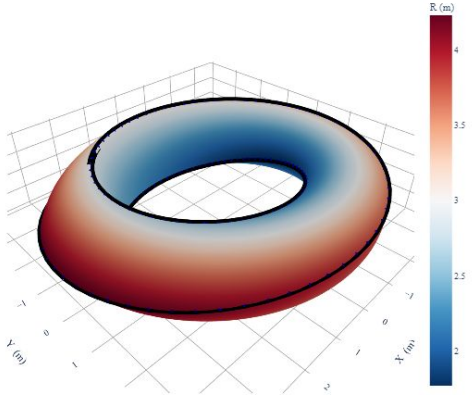
- Coil optimization represents coil geometries with Fourier series
  - Smooth curves, but can have tight curvature
  - Typically penalize things like length, curvature, torsion to ease engineering
- However, piecewise continuous (C^0) coils cannot be represented with Fourier
  - May be an overlooked design space for coils that are simpler to manufacture than arbitrarily shaped modular coils
  - Could use in conjunction with other types of coils to simplify overall design

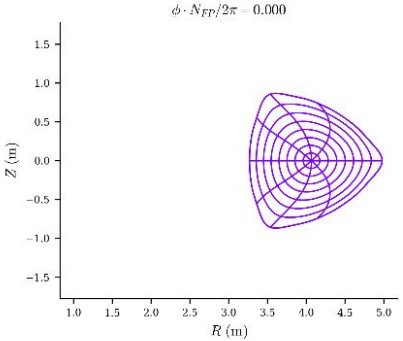


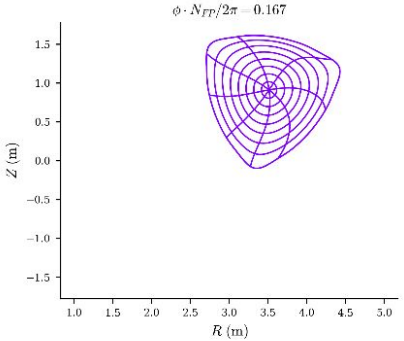
Simplify?

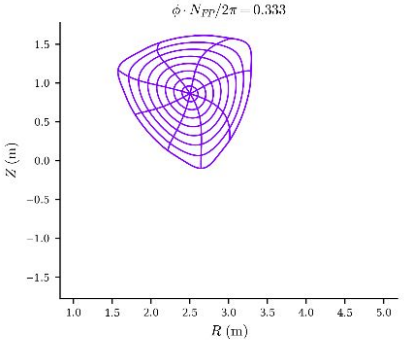


Umbilic figures backup









Plasma Control