# Optimizing Magnetic Confinement Devices for Fusion Plasmas

David Bindel

12 Nov 2024

Department of Computer Science
Cornell University

# Who?

Simons Collaboration: "Hidden Symmetries and Fusion Energy"

`https://hiddensymmetries.princeton.edu/`

A collaboration of plasma physicists and mathematicians from:

Princeton, NYU, Maryland, IPP Greifswald, Warwick, CU Boulder, Cornell, UW Madison, EPFL, ANU, UT Austin, U Arizona.

(along with many unfunded collaborators)

- Phase 0: Aug 2017-Aug 2018
- Phase 1: Sep 2018-Aug 2022
- Phase 2: Sep 2022-Aug 2025

# Some Phase 0 recollections

| | |
|---|---|
| 2017-08-22 | Email from Antoine Cerfon, “would you be interested in participating in these initial conversations?” |
| 2017-09-01 | Initial conversation |
| 2017-10-04 | LOI submitted |
| 2017-12-06 | First two-day proposal meeting |
| 2018-01-31 | Second two-day proposal meeting |
| 2018-02-15 | Proposal submitted |
| 2018-04-18 | Panel pitch (Bhattacharjee, MacKay, Bindel) |
| 2018-05-30 | Award announced to collaboration (recommended change in title to add Fusion Energy). |


I have no idea what you're talking about...
...so here's a bunny with a pancake on its head.

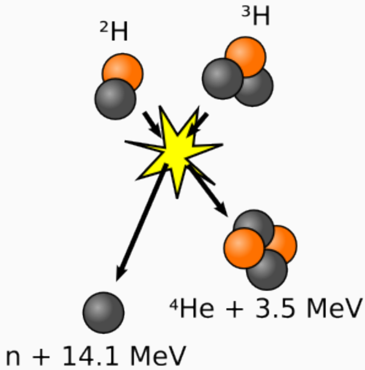
# “Fusion for a 5 Year Old”

*At the risk of sounding like a broken record, I will lobby for the addition of a paragraph in the introduction of the proposal that describes magnetically confined fusion as if it were being explained to a five year old.*
*– Mike O’Neill (2018-02-07)*

*“Adiabatic invariants of Hamiltonian mechanics” is well beyond the level of sophistication that should be included in the intro, in my opinion.*
*– Response to a proposed revision (2018-02-08)*

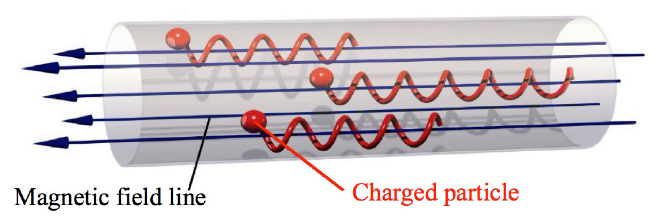
Ad: *Introduction to Stellarators* by Imbert-Gerard, Paul, Wright (`https://arxiv.org/abs/1908.05360`, coming to SIAM)

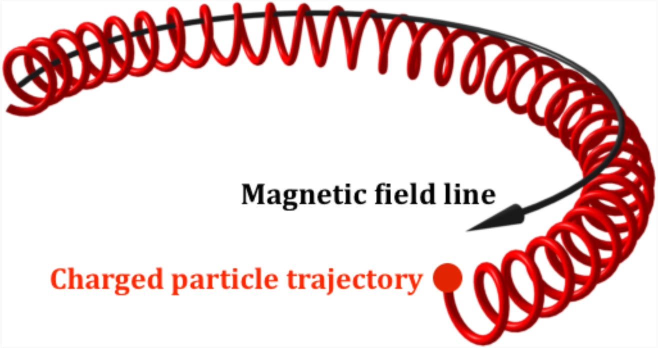
# “Fusion for a 5 Year Old”




Lawson: Need combination of high density, temperature, energy confinement time

# Magnetic confinement basics

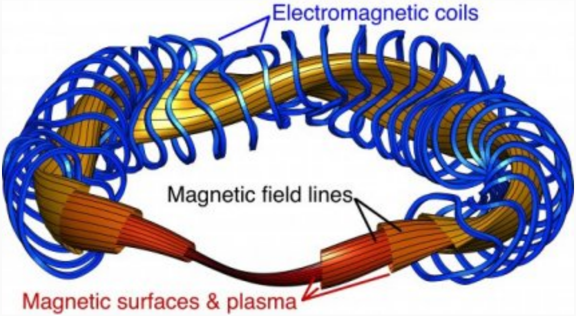


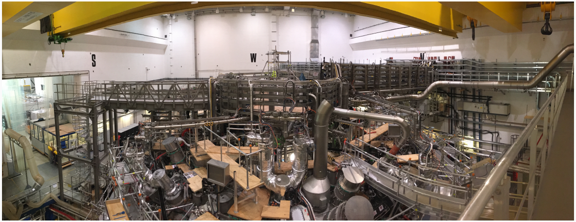



# Magnetic confinement basics

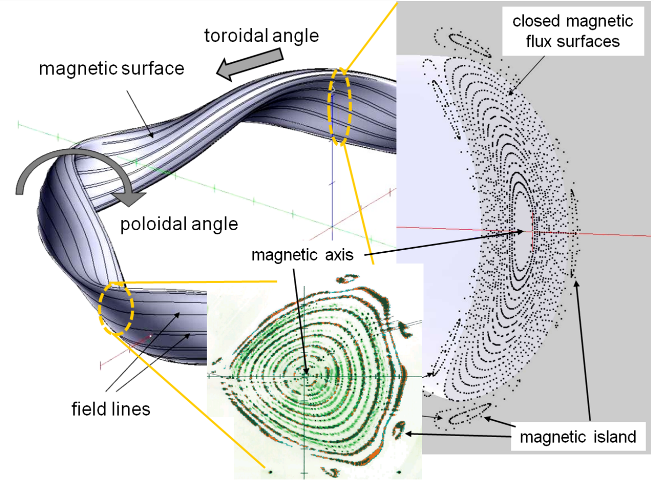

- Ensure drift in and out averages to zero.
- Tokamaks: axisymmetric field (requires plasma current)
- Stellarators: use a "hidden symmetry"

# Stellarator Concept and Practice



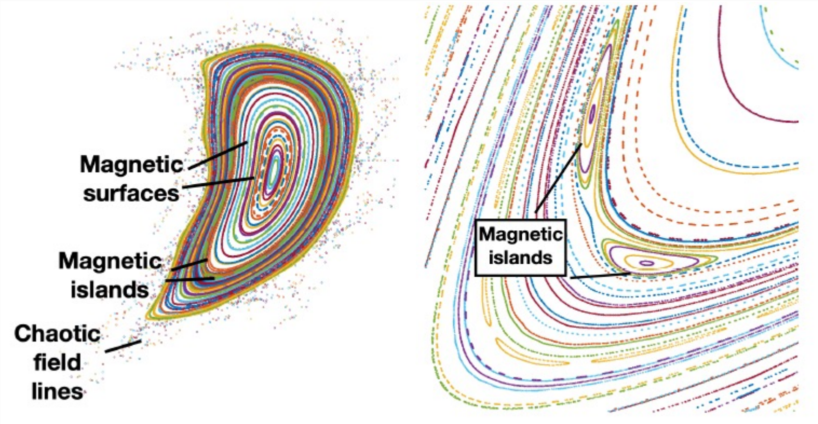



# Wendelstein 7-X Poincaré Plots



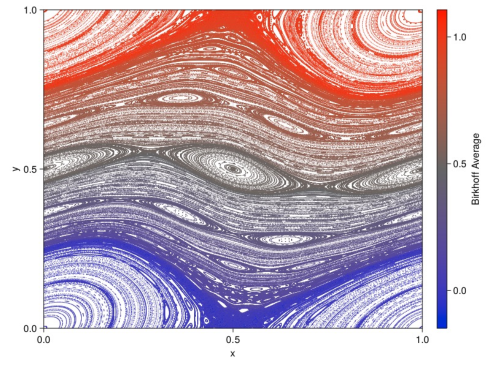

https://commons.wikimedia.org/wiki/File:Stellarator_magnetic_field.png

# Poincaré Features (NCSX)




"An Introduction to Stellarators" (2020)
Imbert-Gerard, Paul, and Wright.
`https://arxiv.org/abs/1908.05360`

# Digression: A Non-Stellarator Test Problem



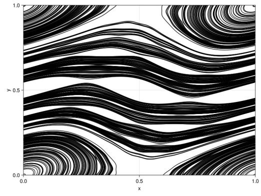

Illustrate with standard (Chirikov-Taylor) map

$$x_{t+1} = x_t + y_{t+1} \bmod 1$$

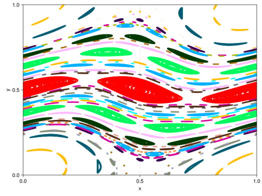
$$y_{t+1} = y_t - \frac{0.7}{2\pi} \sin(2\pi x_t)$$

# Plan in Pictures

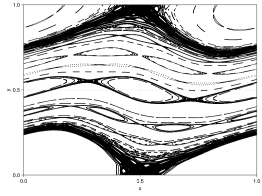
Circles




Islands




Chaos




- Iterating gives a Poincaré plot showing
  - X and O points (hyperbolic and elliptic periodic points)
  - Invariant circles and island chains (quasiperiodic orbits)
  - Chaos
- Goal: Identify these structures cheaply and automatically

# Processing Poincaré Plots

1. Make a Poincaré plot and eyeball it
2. Parameterization method
3. Form a function with invariant level sets
    - Birkhoff averaging
    - Weighted Birkhoff averaging
    - Adaptive weighted Birkhoff (*)
    - Learned labels (*)
4. Model dynamics for a field line (*)

# Parameterization method

Goal: $z : \mathbb{T} \to \mathbb{R}^2$ s.t.

$$F(z(\theta)) = z(\theta + \omega).$$

Discretize via Fourier:

$$\hat{z}(\theta) = \sum_{n=-m}^{m} \hat{z}_n \exp(2\pi i n \theta)$$

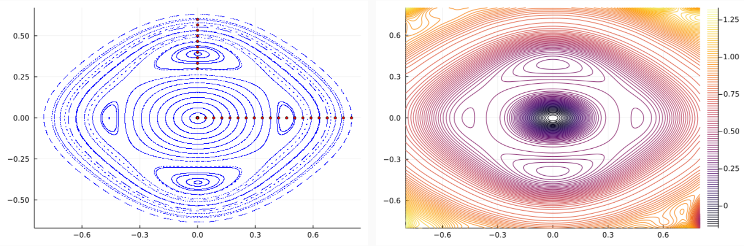
Solve nonlinear least squares problem

$$\min \sum_{i=0}^{N-1} \|z(i/N) - F(z(i/N + \omega))\|^2$$

with two additional constraints (phase + which circle).

Usually combine with continuation (e.g. from fixed point of $F$).

# Learned Labels




Goal: Find (non-constant) $h$ s.t. $h \circ F = h$.

Discretize via favorite ansatz, e.g. $h = \sum_{j=1}^{m} c_j \phi(\|x - x_j\|)$.
Define $h(x_j) = y_j$ and $h(F(x_j)) = y_j'$, solve (for example)

$$\text{minimize } \frac{\eta}{2} y^T K^{-1} y + \frac{1}{2}\|y - \tilde{y}\|^2 \text{ s.t. } y_i = y_i'$$

to encourage $h$ smooth, non-constant, invariant under $F$.

Ruth and Bindel, `https://arxiv.org/abs/2312.00967`

# Birkhoff Average

Consider $f : \Omega \to \Omega$ symplectic, $h \in \mathcal{C}^\infty(\Omega)$

Define *Birkhoff average*:

$$\mathcal{B}_K[h](x) = \frac{1}{K+1}\sum_{k=0}^{K}(h \circ F^k)(x).$$

Birkhoff-Khinchin: for $h \in \mathcal{L}^1$, converges a.e. to conditional expectation of an invariant measure on an invariant set.

Error behavior $\mathcal{B}_K[h](x) - \bar{h}(x)$?

- Invariant circle/island? $O(K^{-1})$
- Chaos? $O(K^{-1/2})$

Rates signal regular vs chaotic ("stochastic") trajectories.

# Birkhoff Average

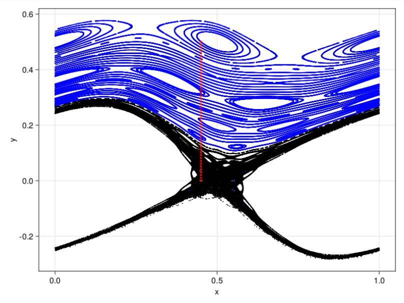
Ideas:

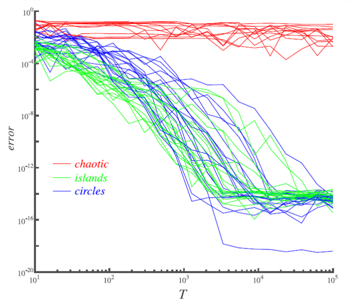
- Invariant sets as level sets of Birkhoff average
- Convergence rates as signal of regularity vs chaos

Converges in the long run – but in the long run, we are all dead. (with apologies to Keynes)

Related: Learn a continuous, nonconstant $\bar{h}$ s.t. $\bar{h} = \bar{h} \circ F$. Can do pretty well with kernel interpolation ansatz – a topic for another talk.

# Weighted Birkhoff average




Sander and Meiss, Physica D, 411 (2020) p. 132569;
Das, Sander, and Yorke, Nonlinearity, 30 (2017), pp. 4111-4140

Weighting accelerates regular convergence to super-algebraic:

$$\mathcal{WB}_K[h](x) = \sum_{k=0}^{K} w_{k,K}(h \circ F^k)(x).$$

# Signal Processing Perspective

Parameterize $z(\theta)$ for invariant circle

$$F(z(\theta)) = z(\theta + \omega), \quad z(\theta) = \sum_{n \in \mathbb{Z}} \hat{z}_n \exp(2\pi i n \theta)$$

Trajectory $z_t = z(\omega t)$ has series expansion

$$z_t = \sum_{n \in \mathbb{Z}} \hat{z}_n \xi^{nt}, \quad \xi = \exp(2\pi i \omega)$$

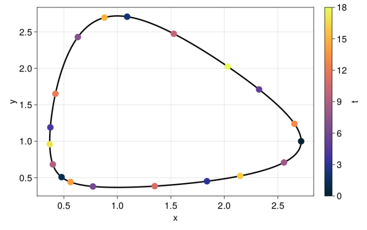
Observables $h_t = h(z_t)$ can be similarly expanded

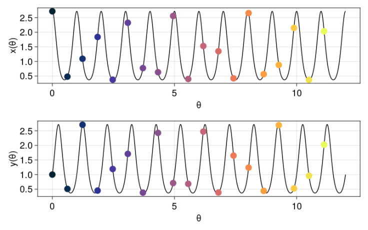
$$h_t = \sum_{n \in \mathbb{Z}} \hat{h}_n \xi^{nt}, \quad \bar{h} = \hat{h}_0$$

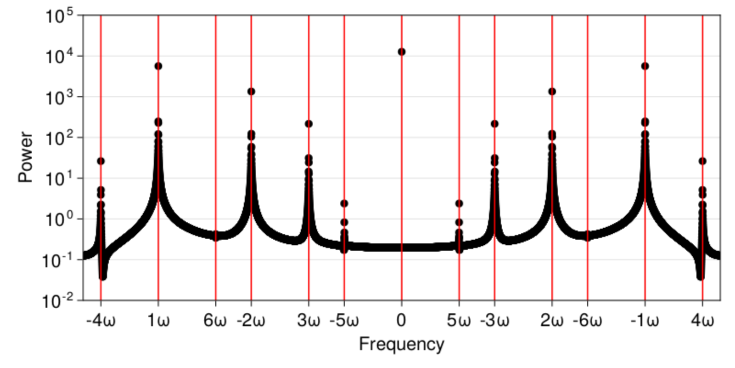
Weighted Birkhoff starting from $x_0$

$$\mathcal{B}_K[h](x_0) = \sum_{n \in \mathbb{Z}} \hat{h}_n p_K(\xi^n), \quad p_K(z) = \sum_{k=0}^{K} w_{k,K} z^k$$

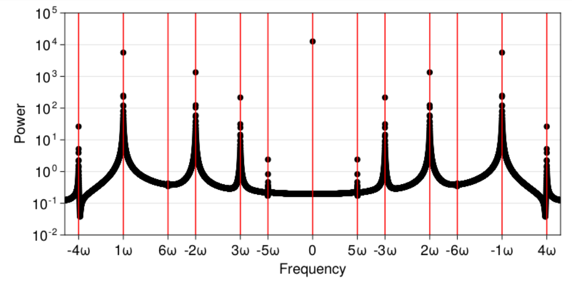
# Signal Processing Perspective

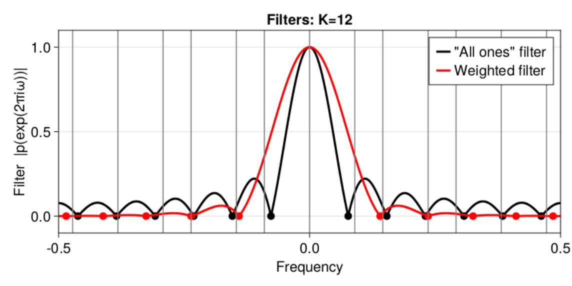


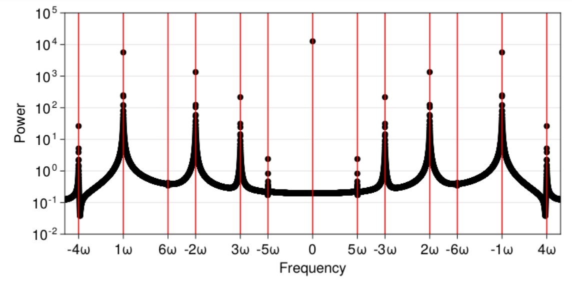



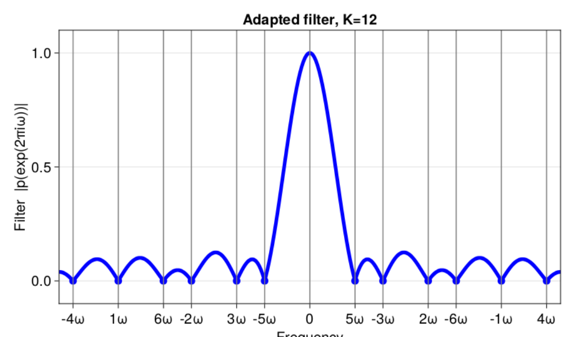
# Signal Processing Perspective



# Signal Processing Perspective: Adaptive Filtering



# Adaptive Filtering

Series for $h_t = h(z_t)$

$$h_t = \sum_{n\in\mathbb{Z}} \hat{h}_n \xi^{nt}$$

Filtered/accelerated series with polynomial $p_K$:

$$\mathcal{AWB}_K[h](x_t) = \sum_{n\in\mathbb{Z}} \hat{h}_n \xi^{nt} p_K(\xi^n) \to \hat{h}_n$$

How do we adaptively choose the filter polynomial?

Desiderata for this to work:

- Fast enough decay of $\hat{h}_n$
- "Sufficiently irrational" $\omega$ (Diophantine condition)

# (Vector) Reduced Rank Extrapolation

Assume

$$h_t = \hat{h}_0 + \sum_{n \neq 0} \lambda_n^t \quad \text{(e.g. } \lambda_n = \xi^n\text{)}$$

Difference sequence removes mean:

$$u_t = h_{t+1} - h_t = \sum_{n \neq 0} (\lambda_n - 1)\hat{h}_m \lambda_m^t$$

Seek coeffs $c_k$ to minimize

$$\sum_{t=0}^{T-1} \left( \sum_{k=0}^{K} c_k u_{k+t} \right)^2 \text{ s.t. } \sum_{k=0}^{K} c_k = 1.$$

Accelerated series is

$$\tilde{h}_t = \sum_{k=0}^{K} c_k h_{k+t}.$$

# Details

- Can (and do) use vector observables
- Rectangular Hankel matrix $\implies$ fast matvecs via FFT
- Solve least squares problem with LSQR
- Constrain for time reversibility $\implies$ palindromic polynomial:

  $$c_j = c_{K-j}$$

  Roots come in inverse pairs (generally on unit circle)
- Measure convergence adaptively via residual

# (Vector) Reduced Rank Extrapolation

Standard vector RRE convergence (Sidi, *Vector Extrapolation Methods with Applications*): if $|\lambda_j|$ are in descending order, error for $K$th extrapolated average goes like

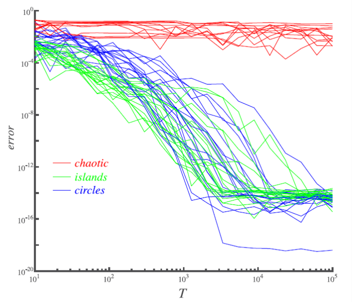
$$\hat{h}_{0,K} - \hat{h}_0 = O(\lambda_{K+1}^{2K}).$$

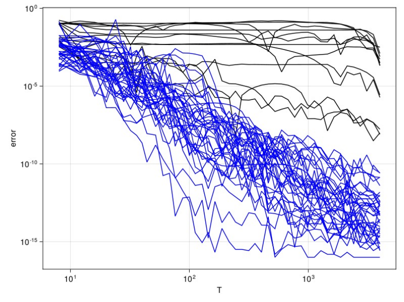
*But* for us everything is on the unit circle!

Alternate analysis gives super-algebraic convergence given

- Enough smoothness of circle (decay of $|\hat{h}_n|$ with $|n|$)
- "Sufficient irrationality" (Diophantine condition) so $\xi_n$ doesn't get too close to 1 too fast.

# Weighted Birkhoff vs RRE

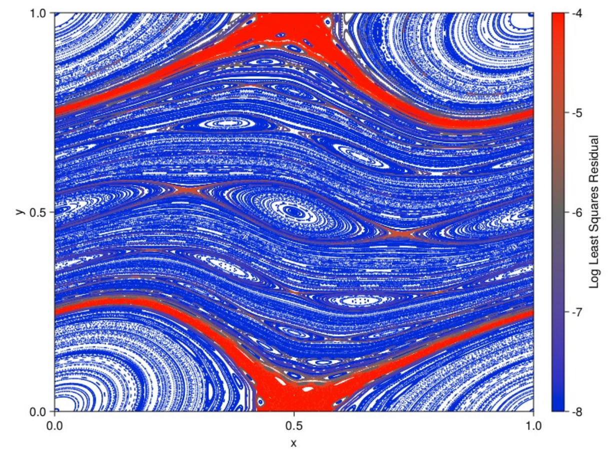



Still good for classification.convergence slightly faster than weighted Birkhoff.

# Residuals and Regularity

Use least squares residual to judge “circleness.”




(Hard cases near rational rotational transform)

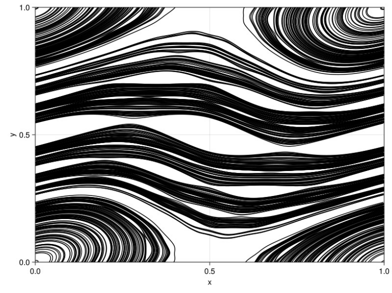
# Post-Processing (Filter Diagonalization)

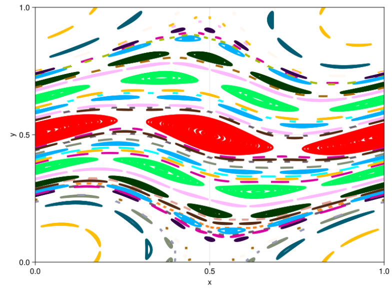
Why use the RRE model just for averaging?

1. Form filter polynomial with coefficients $c$
2. Find natural frequencies / polynomial roots
3. Sort by contribution to signal
4. Of 10 most contributing frequencies, identify rationals (Sander & Meiss)
    - Yes: island chain — RRE on $q$th step
    - No: call largest the rotational transform
5. Project signal onto Fourier modes

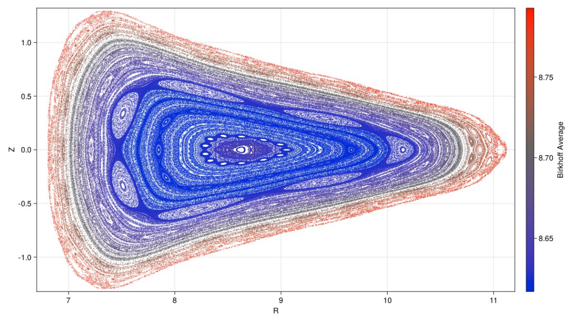
Get shape and characteristics of circles and islands.

# Island Identification



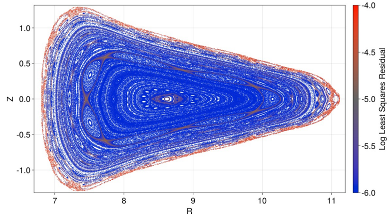
# Wistell Stellarator Configuration



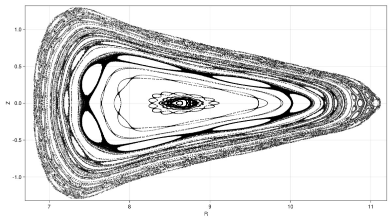

- 1000 random trajectories (via RK4 on interpolated $B$ field)
- $K_{\mathsf{max}} = 300, T_{\mathsf{max}} = 900$
- Residual tolerance = $10^{-6}$
- Rational tolerance = $10^{-6}$

# Wistell Analysis

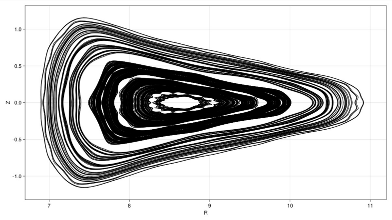
## Residual



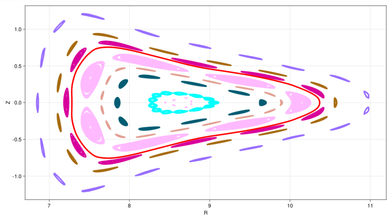

## Chaos




## Circles




## Islands



# Pros and Cons

- Extrapolation pros
    - Classifies chaos vs regular trajectories
    - Recovers invariant circles/islands
    - No need for continuation or initial guesses
    - Parallelizable over trajectories
- Cons
    - Problems near low-order rationals
    - Linear algebra adds extra cost vs weighted Birkhoff
- Higher dimensions?
    - Relevant beyond field line flow (guiding center approx)
    - Invariant sets are more complicated
    - The "model the trajectory" philosophy should still work

Ruth and Bindel, `https://arxiv.org/abs/2403.19003`

# Reset

We were talking about *optimizing* stellarators...

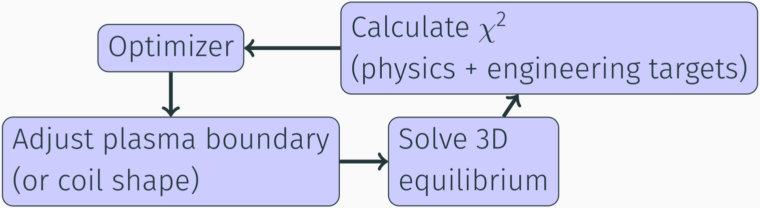
## Stellarator Quality Measures

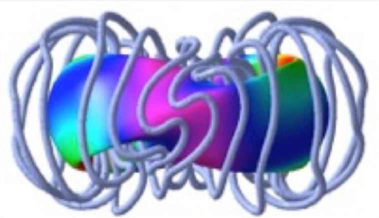
What makes an "optimal" stellarator?

- Approximates field symmetries (which measures?)
- Satisfies macroscopic and local stability
- Divertor fields for particle and heat exhaust
- Minimizes collisional and energetic particle transport
- Minimizes turbulent transport
- Satisfies basic engineering constraints (cost, size, etc)

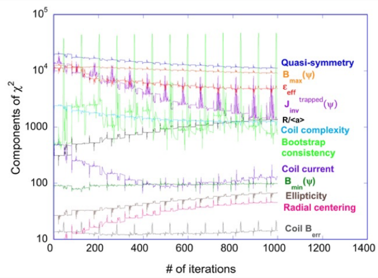
Each objective involves different approximations, uncertainties, and computational costs.

# How Do We Optimize? (STELLOPT Approach)



$$r(\phi,\theta) + iz(\phi,\theta) = \sum \alpha_{m,n} e^{i(m\phi - n\theta)}$$



# Challenges

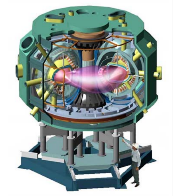
1. Costly and "black box" physics computations
    - Each step: MHD equilibrium solve, transport, coil design, …
    - Several times per step for finite-difference gradients
2. Managing tradeoffs
    - How do we choose the weights in the $\chi^2$ measure? By gut?
    - Varying the weights does not expose tradeoffs sensibly
3. **Dealing with uncertainties**
    - What you simulate $\neq$ what you build!
4. Global search
    - How to avoid getting stuck in local minima?

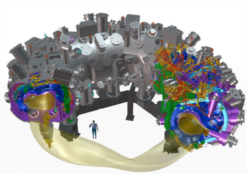
# Optimization Under Uncertainty

Low construction tolerances:

- NCSX: 0.08%
- Wendelstein 7-X: 0.1% – 0.17%

Higher tolerances as coil opt goal!

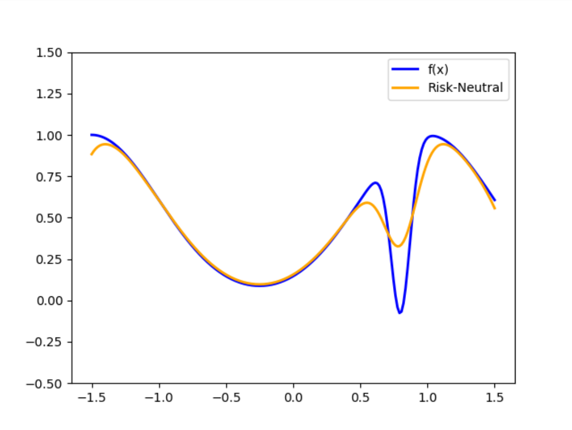




Also want tolerance to

- Changes to control parameters
- Uncertainty in physics or model

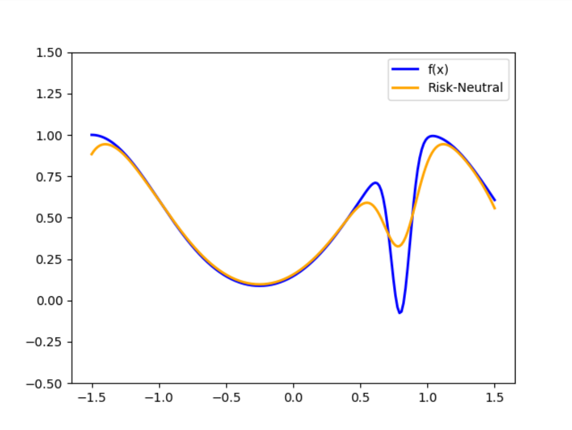
# Risk-neutral OUU




Want efficient OUU in $\sim 200$ dimensions

$$\min_{x \in \Omega} \mathbb{E}_U[f(x - U)]$$

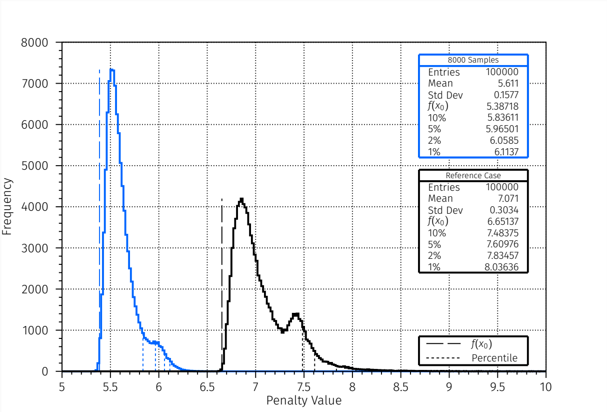
# Risk-neutral OUU




Want efficient OUU in $\sim 200$ dimensions

$$\min_{x \in \Omega} \mathbb{E}_U[f(x - U)]$$

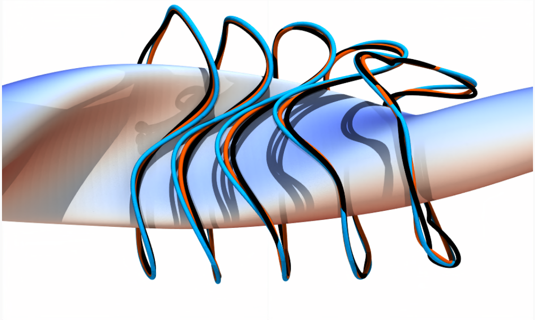
# (Recent) Prior: Monte Carlo Approach




**Robustness & mean perf greatly improved** (w/ $\sim 10^8$ evals)

J.-F. Lobsien, M. Drevlak, T. Kruger, S. Lazerson, C. Zhu, T. S. Pedersen, *Improved performance of stellarator coil design optimization*, Journal of Plasma Physics, 2020.

# Our Approach: fast TuRBO-ADAM

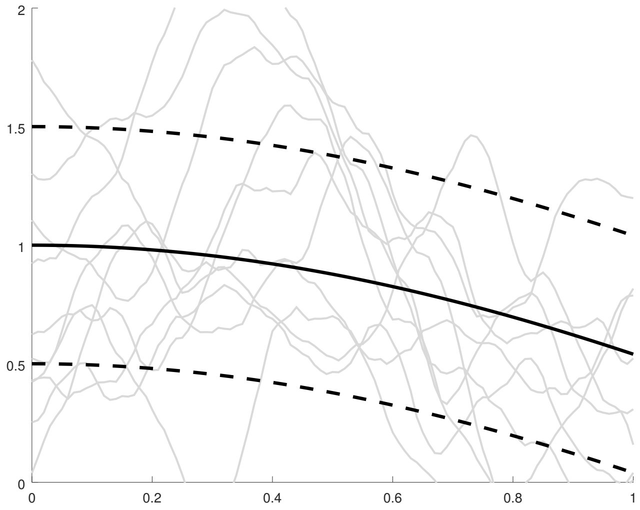


Black: ref; red: TuRBO-ADAM 10mm; blue: TuRBO-ADAM 20mm.

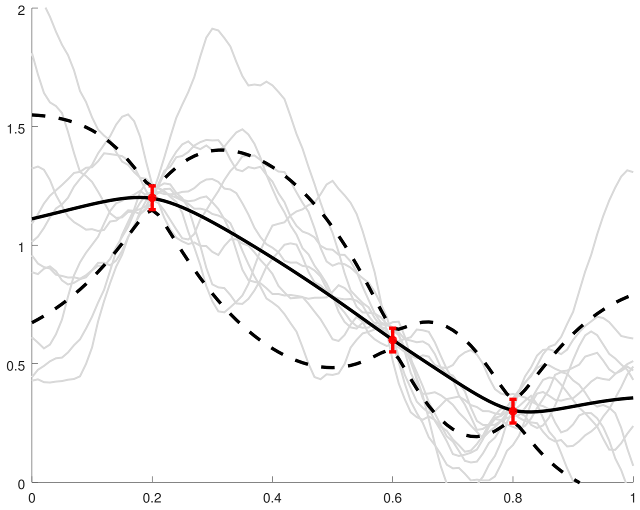
Evaluate objective with FOCUS from PPPL.

- Global search with modified TuRBO
- Local refinement with ADAM with control variate

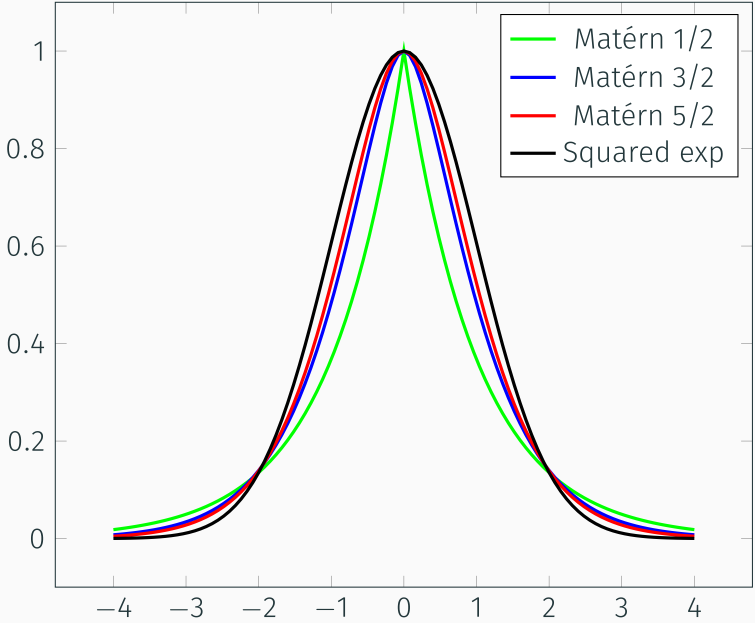
Costs about 0.01% the evaluation budget.


2
1.5
1
0.5
0
0
0.2
0.4
0.6
0.8
1

# Being Bayesian



# Matérn and SE kernels



# Gaussian Processes (GPs)

Our favorite continuous distributions over

$$\begin{array}{lll} \mathbb{R}: & \text{Normal}(\mu, \sigma^2), & \mu, \sigma^2 \in \mathbb{R} \\ \mathbb{R}^n: & \text{Normal}(\mu, C), & \mu \in \mathbb{R}^n, C \in \mathbb{R}^{n \times n} \\ \mathbb{R}^d \to \mathbb{R}: & \text{GP}(\mu, k), & \mu : \mathbb{R}^d \to \mathbb{R},\ k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R} \end{array}$$

More technically, define GPs by looking at finite sets of points:

$$\forall X = (x_1, \ldots, x_n),\ x_i \in \mathbb{R}^d,$$

$$\text{have } f_X \sim N(\mu_X, K_{XX}), \text{ where}$$

$$f_X \in \mathbb{R}^n, \quad (f_X)_i \equiv f(x_i)$$

$$\mu_X \in \mathbb{R}^n, \quad (\mu_X)_i \equiv \mu(x_i)$$

$$K_{XX} \in \mathbb{R}^{n \times n}, \quad (K_{XX})_{ij} \equiv k(x_i, x_j)$$

When $X$ is unambiguous, we will sometimes just write $K$.

# Being Bayesian

Now consider prior of $f \sim \mathrm{GP}(\mu, k)$, noisy measurements

$$f_X \sim y + \epsilon, \quad \epsilon \sim N(0, W), \qquad \text{typically } W = \sigma^2 I$$

Posterior is $f \sim \mathrm{GP}(\mu', k')$ with

$$\begin{aligned} \mu'(x) &= \mu(x) + K_{xX} c & \qquad \tilde{K} &= K_{XX} + W \\ k'(x, x') &= K_{xx'} - K_{xX} \tilde{K}^{-1} K_{Xx'} & \qquad c &= \tilde{K}^{-1}(y - \mu_X) \end{aligned}$$

The expensive bit: solves with $\tilde{K}$.

# Bayesian Optimization (BO)

Typical GP-based BO:

- Evaluate $f$ on initial sample in $\Omega$
- Condition a GP on sample data
- Until budget exhausted
  - Optimize *acquistion function* $\alpha(x)$ over $\Omega$
    (e.g. $\alpha_{\mathrm{EI}}(x) = E\left[[f(x_{\mathrm{best}}) - f(x)]_+\right]$ where $x_{\mathrm{best}}$ is best so far)
  - Evaluate at selected point
  - Update the GP model (including hyper-parameters)
- Standard cost: $O(n^3)$ per step (with $n$ data points)

# Multi-Start

Suppose $d$ large, but not too many minimizers:

- Choose $M$ starting points scattered over $\Omega$
- Run local minimizer (gradient descent, Newton, etc)
- Hope for at least one start per convergence basin

Q: How to allocate effort to different starts?

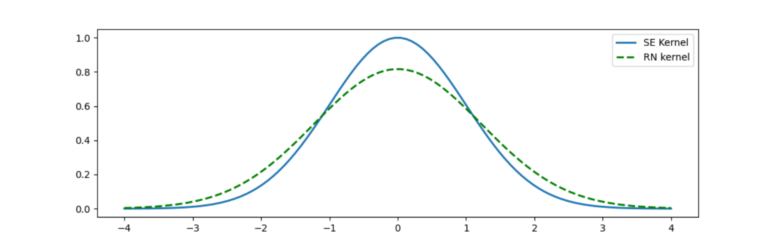
# TuRBO: Trust-Region BO

For high-d: combine local BO with multi-start strategy

- Rough global sampling at $M$ points
- Local GP models and trust-region around each point
- Thompson sampling to choose which local model (and trust region) to refine next

(Eriksson, Pearce, Gardner, Turner, Poloczek, 2019)

# TuRBO + OUU




- TuRBO builds GP models for $f(x)$ (nominal objective)
- Simple transform from GP for $f(x)$ to GP for $E_U[f(x+U)]$ (Beland and Nair, 2017)

Problem: TuRBO explores a lot — want more refinement

# Stochastic Gradient Descent (SGD)

Ordinary gradient descent is

$$x_{k+1} = x_k - \alpha_k \nabla \phi(x_k)$$

SGD is

$$x_{k+1} = x_k - \alpha_k g_k$$

where $g_k$ is a random draw, $E[g_k] = \nabla \phi(x_k)$.

For $\phi(x) = E_U[f(x+U)]$, use $g_k = \nabla f(x_k + u_k)$.

Convergence is slow ($O(1/m)$), but steps can be cheap.
Speed depends a lot on variance of gradient estimator.

# Adam + Control Variates

- Regular Adam: stochastic gradient algorithm with "adaptive momentum" for step size control. Use directions

$$g(x) = \nabla f(x + U)$$

  for a random draw $U$ (can also do mini-batch).

- Variance reduction with control variates (Wang, Chen, Smola, Xing, 2013)

$$\begin{aligned} g(x) &= \nabla f(x + U) + \alpha(\hat{g}(x) - E[\hat{g}(x)]) \\ \hat{g}(x) &= \nabla f(x) + HU. \end{aligned}$$

- True Hessian not avail, so set $H$ to be an approximate Hessian (BFGS approximation via gradients from Adam).

# Additional Information

*Multi-output GPs* model $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}^k$

- Model covariance over space and across outputs.
- Example: function values + derivatives

$$\mu^\nabla(\mathrm{x}) = \begin{bmatrix} \mu(x) \\ \nabla_x \mu(x) \end{bmatrix}, \quad k^\nabla(x, x') = \begin{bmatrix} k(x, x') & (\nabla_{x'} k(x, x'))^T \\ \nabla_x k(x, x') & \nabla^2 k(x, x') \end{bmatrix}$$

- Can also model multi-fidelity sims, etc

*Pro:* FOCUS provides gradients, easy to incorporate!
*Con:* Matrix dimensions scale like $n(d+1)$
*(Partial) Fix:* Variational inference (Bindel, Gardner, Huang, Padidar, Zhu, NeurIPS 2021)

# Some concluding notes

> *I was tense, I was nervous, I guess it just wasn't my night. Art Fleming gave the answers; oh, but I couldn't get the questions right.*
>
> — *Weird Al Yankovic, "I Lost on Jeopardy"*

Stellarator optimization is hard. Beyond formulating reasonable objectives, challenges include:

1. Costly and "black box" physics computations
2. Managing tradeoffs
3. Dealing with uncertainties
4. Global search

Many challenges/opportunities in the formulation – not unique to stellarators!

(And lots of interesting non-optimization problems, too!)