Cornell University

Stellarator Optimization

David Bindel, Silke Glas, Ariel Kellison,
Curran Muhlberger, Misha Padidar

Cornell University

smg374@cornell.edu

April 22, 2020

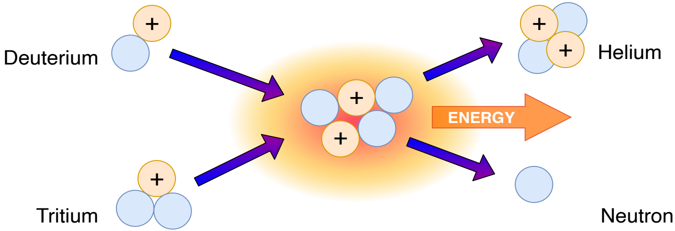
Thermonuclear Fusion

- Fusion of deuterium and tritium (isotopes of hydrogen)
- Collision of ions "fuse" releasing 17.6 MeV, a neutron, and helium

$$D + T \rightarrow n(14.1 MeV) + He(3.5 MeV)$$

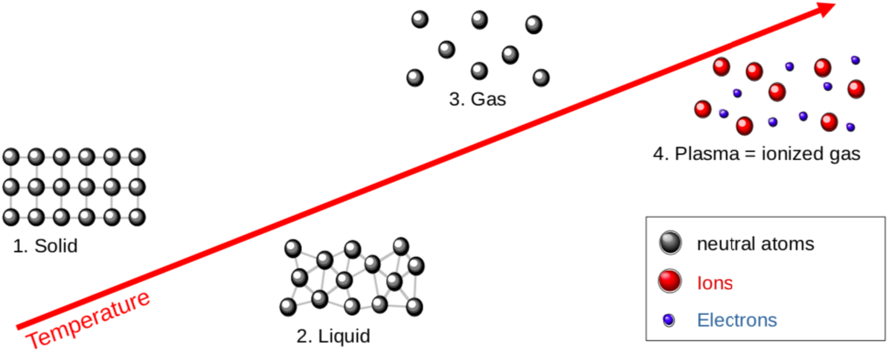
- Only possible when Lawson criteria is fulfilled, i.e,

$$nT\tau_E > 3 \times 10^2 1 m^{-3} keVs$$



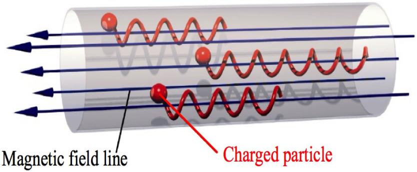
What is a Plasma?

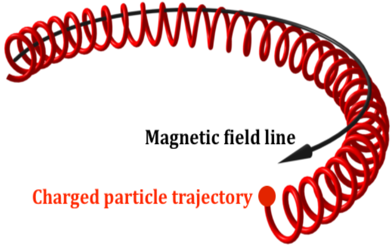
- At high temperatures needed any gas becomes plasma



Principle of Magnetic Confinement

- Charged particles are confined by magnetic fields
- Uniform field: particle moves in helical path
- Nonuniform field: particle moves *approximately* in helical path



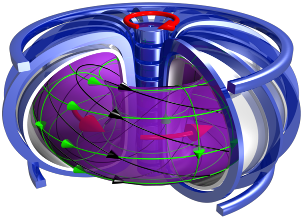


How is the Drift handled?

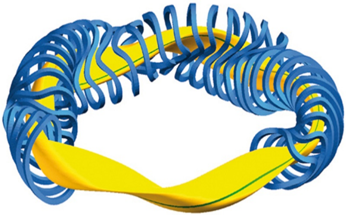
- Handling drift leads to different devices
- Tokamak: add electric field to plasma produced by transformer
- Stellarator: use more sophisticated geometry

Two main devices for generating fusion energy:

Tokamak

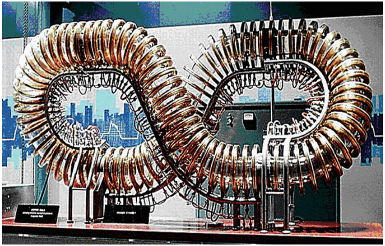


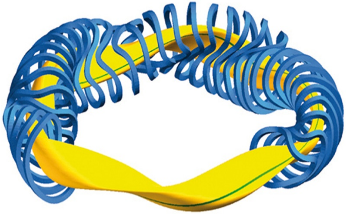
Stellarator



The Stellarator

- Stellarator from latin stella (engl. star),
- Concept of stellarator by Lyman Spitzer in 1951,
- Due to superior performance of tokamaks abandoned in 1960's,
- Return of stellarator (mostly) due to modern computational methods.





Outlook:

1. Motivation
2. Stellarator Optimization Status Quo
3. Dealing with Uncertainties
   - Short Introduction to Bayesian Optimization
   - Risk-Neutral Optimization
   - Risk-Averse Optimization
4. Conclusion

What Makes a Good Stellarator?

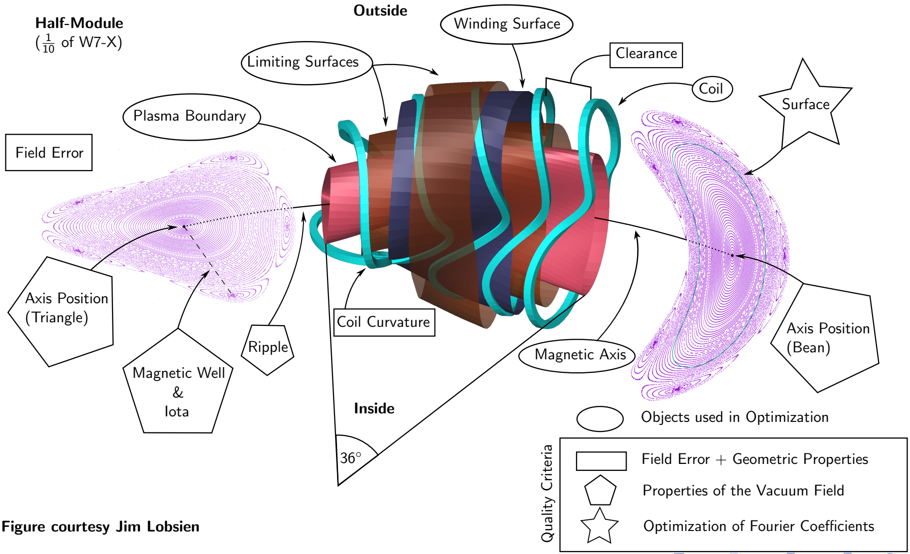


Figure courtesy Jim Lobsien

Optimization Methods

a) Building Coils and see what happens

b) Coil optimization first

c) Standard Optimization: Two Step Method:

1. Optimize plasma boundary
2. Optimize coils to achieve plasma boundary

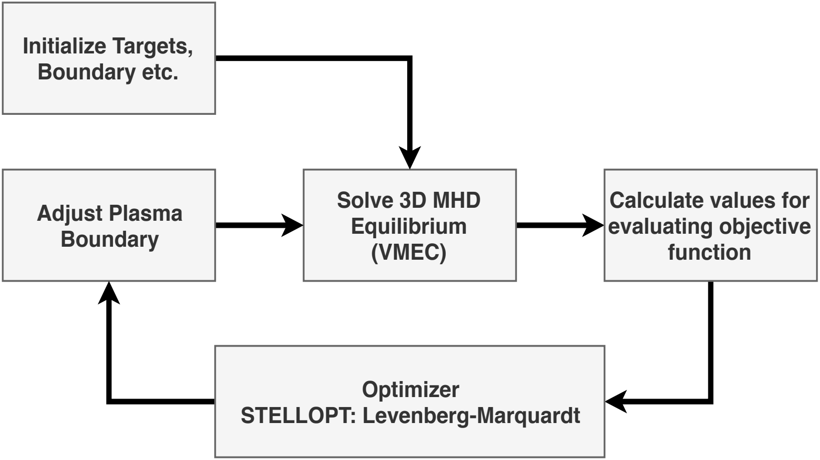
Pros:

- Learn about interplay of boundary/parameters
- Less numerical expensive than directly evaluating the coils

Cons:

- There might be no set of coils which can reproduce the magnetic field
- One needs two optimization procedures for coils

Optimization with STELLOPT



STELLOPT Approach

Goal: Design Magnetohydrodynamics (MHD) equilibrium

- Possible parameters for boundary: $C \subset \mathbb{R}^n$
- Physics/engineering properties: $F : C \to \mathbb{R}^m$
- Target vector: $F^* \in \mathbb{R}^m$

Target function: Minimize χ^2 objective over C, i.e.,

$$\chi^2(x) = \sum_{k=1}^{m} \frac{J_k(x)}{\sigma_k^2}, \quad J_k(x) = (F_k(x) - F_k^*)^2$$

Solve via Levenberg-Marquardt, genetic algorithms
(avoids gradient information apart from finite differences)

Challenges

1) Costly and "black box" physics computations

- Each step: MHD equilibrium solve, coil design,
- Several times per step for finite-difference gradients

2) Managing tradeoffs

- How do we choose the weights in the χ^2 measure?
- Varying the weights does not expose tradeoffs sensibly

3) Dealing with uncertainties

- What you simulate $\neq$ what you build!

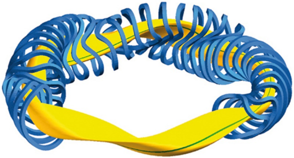
4) Global search

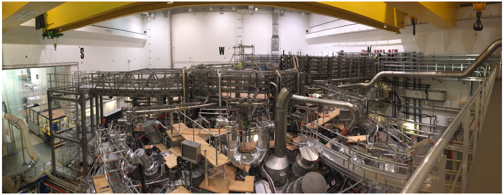
- How to avoid getting stuck in local minima?

Dealing with Uncertainties

What you simulate $\neq$ what you build!

- Errors in coil building process can lead to loss of fusion energy
- Need: Method that increases construction tolerances without compromising performance





Left: W7-X (sketch); right: W7-X (real).

What are we optimizing?

Single objective optimization:

$$\text{minimize } f(x) \text{ s.t. } x \in \Omega \subset \mathbb{R}^d$$

Assume

- Ω compact (and simple — e.g. a box)
- $f : \Omega \to \mathbb{R}$, f is expensive to evaluate
- We think the true f has some smoothness

Later: constraints, non-smoothness, multi-objective, derivatives, etc

What are we optimizing?

Single objective optimization:

$$\text{minimize } f(x - U) \text{ s.t. } x \in \Omega \subset \mathbb{R}^d, U \text{ random variable}$$

Assume

- Ω compact (and simple — e.g. a box)
- f is expensive to evaluate (and maybe noisy)
- We think the true f has some smoothness

Later: constraints, non-smoothness, multi-objective, derivatives, etc

Measures of Uncertainty in this Talk

Setting:

- Investor has two investments to choose
- What does he choose if he uses:

Measures of Uncertainty in this Talk

Setting:

- Investor has two investments to choose
- What does he choose if he uses:

Risk-Neutral Optimization:

- Picks investment with highest return
- Risk is not taken into account

Measures of Uncertainty in this Talk

Setting:

- Investor has two investments to choose
- What does he choose if he uses:

Risk-Neutral Optimization:

- Picks investment with highest return
- Risk is not taken into account

Risk-Averse Optimization:

- More conservative approach
- Prefers lower return with less risk

Short Introduction to Bayesian Optimization

Surrogate Optimization

Basic Idea: replace expensive $f(x)$ by $\hat{f}(x)$ using data

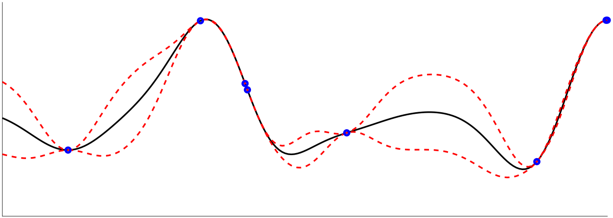
Wishlist:

- $\hat{f}(x)$ should be cheap/cheaper to evaluate
- $\hat{f}(x)$ should be able to give measure of uncertainty

Being Bayesian for the rest of the talk, i.e,

- Works well for input dimensions ≤ 10

Gaussian Process (GP)



- A Gaussian process (GP) is a distribution over functions, written as:

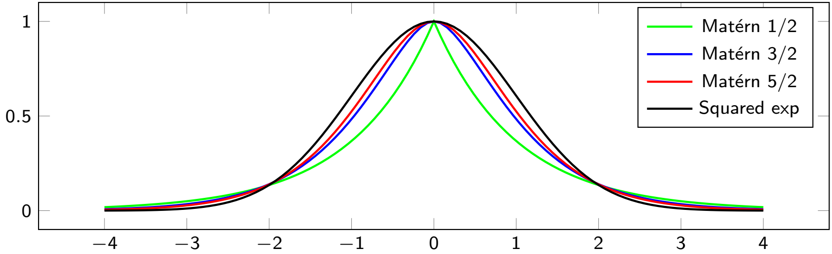
$$\mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')).$$

- Mean: $\mu : \mathbb{R}^d \to \mathbb{R}$, often zero, constant or low-degree polynomial.
- Covariance: $k(\mathbf{x}, \mathbf{x}') : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. Assumed $k(\mathbf{x}, \mathbf{x}')$ is PD kernel.
- For data points $\mathbf{x}$, $y(\mathbf{x}) \sim \mathcal{N}(\mu_X, K_{XX})$, $(\mu_X)_i = \mu(\mathbf{x}_i)$, $(K_{XX})_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$.

Incorporating Assumptions

Key places to inject assumptions on f:

- Kernel choice
  - Standard choices (e.g., squared exponential, Matérn) work well
  - ... but better choices require less training data
  - Typically choose based on some belief about smoothness
- Mean field
  - Standard choices are constant or linear
  - Can bring in other shapes if more is known



Bayesian Optimization (BO)

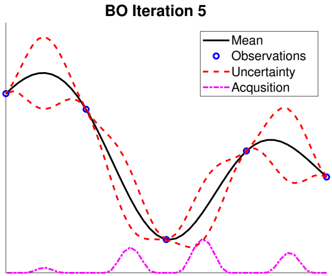


Figure: First building GP from observations (blue). Optimize acquisition function (magenta) for new sample point (green).

- Goal: given simple domain Ω, find the global minimum $\mathbf{x}^* \in \Omega$:

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \; , \; \forall \mathbf{x} \in \Omega.$$

- Build a Gaussian process that models f.
- Choose next evaluation point by maximizing acquisition function $\Lambda(\mathbf{x})$.

Bayesian optimization

- Acquisition function must balance exploration and exploitation
- Let y^* is current observed min. Popular acquisition functions:
  - Probability Improvement (PI):

$$\mathrm{PI}(\mathbf{x}) = P(f(x) \le y^*) = \Phi\left(\frac{y^* - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right).$$

  - Expected Improvement (EI):

$$\begin{aligned}\mathrm{EI}(\mathbf{x}) &= \mathbb{E}[\max(y^* - f(x), 0)] \\ &= (y^* - \mu(\mathbf{x}))\Phi\left(\frac{y^* - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right) + \sigma(\mathbf{x})\phi\left(\frac{y^* - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right).\end{aligned}$$

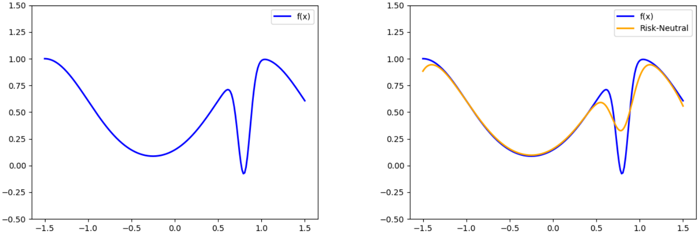
Risk-Neutral Optimization

Risk-Neutral Optimization

Perform well on average under perturbations.

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

- Not worried in general about bad perturbations
- Solution prefers shallower wide basins
- Interested in case when f is a GP



Recall Standard Bayesian Optimization

Let $f \sim (0, k(x, x'))$, i.e.,

- $m(x) = 0$
- $k(x, x') = \phi(\|x - x'\|)$

Observe f at n points

- $X = \{x_1, \ldots, x_n\}$,
- $f_X = \{f(x_1), \ldots, f(x_n)\}$
- Posterior is GP with $f^*|f_X \sim (m^*(x), k^*(x, x))$ with

$$m^*(x) = \sum_{i=1}^{n} c_i \phi(\|x - x_i\|), \qquad c_i = [\Phi_{XX}^{-1} f_X]_i, \quad d_{ij} = [\Phi_{XX}^{-1}]_{ij}$$

$$k^*(x, x') = \phi(x - x') - \sum_{i,j=1}^{N} d_{ij} \phi(||x - x_i||) \phi(||x' - x_j||)$$

Computing the Risk-Neutral Optimization function

Computing the expected value yields:

$$\bar{f}(x) = \mathbb{E}_U[f(x - U)] = \int_{\mathbb{R}} f^*(x - u)w(u)\,du = [f^* * w](x)$$

Convolution performs linear operation on GP $\rightsquigarrow \bar{f} \sim GP(\bar{m}(x), \bar{k}(x, x))$ with

$$\begin{aligned}
\bar{m}(x) &:= [m^* * w](x) = \sum_{i=1}^{N} c_i \bar{\phi}(\|x - x_i\|) \\
&= \sum_{i=1}^{N} c_i \int_{\mathbb{R}} \phi(\|x - u - x_i\|)w(u)\,du = \sum_{i=1}^{N} c_i[\phi * w](x) \\
\bar{k}(x, x') &:= [[k^* *_1 w] *_2 w](x, x') \\
&= \int_{\mathbb{R},\mathbb{R}} k^*(|||(x - u) - (x' - v)|||)w(u)w(v)\,du\,dv
\end{aligned}$$

Example - Squared Exponential Kernel

Quantities $\bar{m}(x)$ and $\bar{k}(x, x')$ can be calculated analytically for some cases:

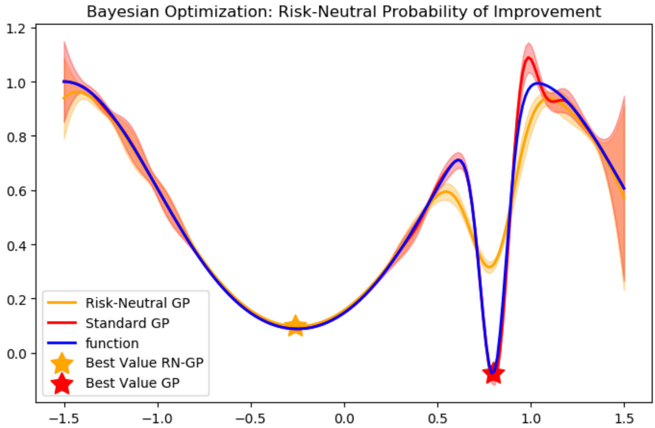
- No Monte-Carlo needed to evaluate the integral
- Feasible for, e.g., squared exponential kernel with hypers α, a

$$\phi(x - x') = \alpha \exp\left(-\frac{(\|x - x'\|)^2}{a^2}\right)$$

- and mean-zero gaussian distribution

$$w(u) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2} u^T \Sigma^{-1} u\right)$$

Risk-Neutral Bayesian Optimization



Risk-Averse Optimization

Mean-Variance

Mean-Variance penalizes variance as proxy for uncertainty, $\eta \in \mathbb{R}^+$

$$\min_{x \in \Omega} \left(\mathbb{E}_U[f(x-U)] + \eta \mathrm{Var}_U[f(x-U)] \right).$$

The objective function w.r.t. f^* reads

$$\min_{x \in \Omega} \left(\mathbb{E}_U[f(x-U)] + \eta \left(\mathbb{E}_U[f^2(x-U)] - \left(\mathbb{E}_U[f(x-U)]\right)^2 \right) \right),$$

with

$$\bar{f}(x) := \mathbb{E}_U[f(x-U)] = \int_{\mathbb{R}} f^*(x-u) w(u)\, du,$$

$$\hat{f}(x) := \mathbb{E}_U[f^2(x-U)] = \int_{\mathbb{R}} f^*(x-u)^2 w(u)\, du,$$

$$\bar{f}^2(x) := \left(\mathbb{E}_U[f(x-U)]\right)^2 = \left(\int_{\mathbb{R}} f^*(x-u) w(u)\, du \right)^2.$$

Computation of Third Part -1

- What we want: distribution of $\bar{f}^2(x)$

Way to find out:[1]

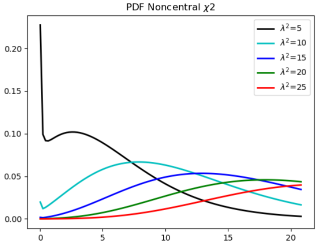
- Re-scale $\bar{f}(x)$ with $\sqrt{\bar{k}(x,x)}$, thus

$$Y := \bar{f}(x)/\sqrt{\bar{k}(x,x)} \sim \mathcal{N}(\lambda, 1),$$

with $\lambda := \frac{\bar{m}(x)}{\sqrt{\bar{k}(x,x)}}$

- For the squared variable Y^2, we get a noncentral χ^2 distribution with
  - 1 degree of freedom
  - noncentrality parameter λ^2

$$Y^2 \sim NC\chi^2(1, \lambda^2).$$



[1]Following A.K. Uhrenholt, B.S. Jensen, *Efficient Bayesian Optimization for target vector estimation*, Proceedings of AISTATS, 2019.

Computation of Third Part -2

But: only want distribution for $\bar{f}^2(x)$

$\rightsquigarrow$ variable transformation

$$p(\bar{f}^2(x)) = p(g(\bar{f}^2(x))) * |g'(\bar{f}^2(x))|$$

$\rightsquigarrow$ thus

$$\bar{f}^2(x) \sim NC\chi^2(Y^2|1, \lambda) * \frac{1}{\bar{k}(x, x)}.$$

Approximation of the noncentral χ^2

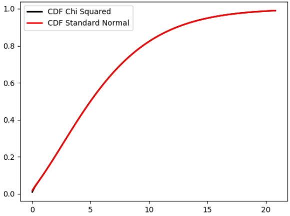
Assume $X \sim NC\chi^2(1, \lambda^2)$, with 1 DOF, λ^2 noncentrality parameter,

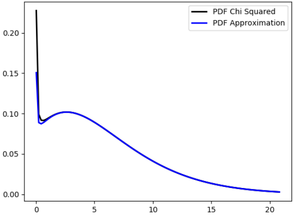
- CDF of X is approximated by standard normal CDF [2]:

$$\Phi\left(\sqrt{X} - \lambda\right)$$

- This leads to the PDF

$$\phi\left(\sqrt{X} - \lambda\right)\left(\frac{1}{2\sqrt{X}}\right)$$





[2] D.A.S Fraser, A.C.M. Wong, J. Wu, *An approximation for the noncentral chi-squared distribution*, Communications in Statistics - Simulation and Computation, 1998.

Computation of second part

Seek: distribution of the second part

$$\hat{f}(x) := \mathbb{E}_U[f^2(x - U)] = \int_{\mathbb{R}} f^*(x - u)^2 w(u)\, du$$

- follow third part for squared function

$$\rightsquigarrow (f^*(x))^2 \sim NC\chi^2(1, (\lambda^*)^2) * \frac{1}{k^*(x, x)}$$

with $\lambda := \frac{m^*(x)}{\sqrt{k^*(x,x)}}$

Computation of second part

Seek: distribution of the second part

$$\hat{f}(x) := \mathbb{E}_U[f^2(x-U)] = \int_{\mathbb{R}} f^*(x-u)^2 w(u)\, du$$

- follow third part for squared function

$$\rightsquigarrow (f^*(x))^2 \sim NC\chi^2(1, (\lambda^*)^2) * \frac{1}{k^*(x,x)}$$

with $\lambda := \frac{m^*(x)}{\sqrt{k^*(x,x)}}$

Status Quo:

- First term is gaussian
- Third term is noncentral χ^2 times constant
- In second term noncentral χ^2 is in integral

Conclusion

Conclusion

Conclusion:

- Optimizing stellarators is hard!
- Risk-neutral approach implemented
- Progress on risk-averse approaches
  - Representation of mean-variance
  - Closed-form approximation for VaR
  - Representation of CVaR

Conclusion

Conclusion:

- Optimizing stellarators is hard!
- Risk-neutral approach implemented
- Progress on risk-averse approaches
  - Representation of mean-variance
  - Closed-form approximation for VaR
  - Representation of CVaR

Further work:

- Implementing/Evolving measures
- Perform risk-neutral approach for stellarator problem
- Starting doing multi-objective optimization