Automated Normal Mode Analysis in Reduced Models of Turbulence for Stellarator Optimization

B.J. Faber[1], I.J. McKinney[1], A. Bader[1], P.W. Terry[1], C.C. Hegna[1], and M.J. Pueschel[2]

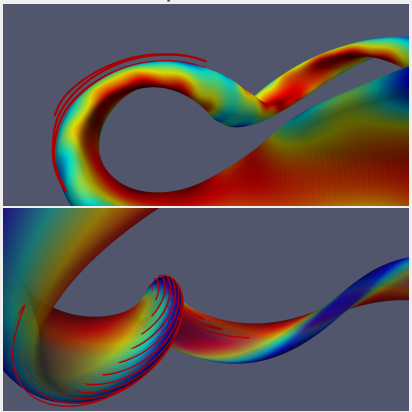
[1]University of Wisconsin-Madison, USA
[2]Dutch Institute for Fundamental Energy Research, The Netherlands

March 1, 2021

Stellarator Optimization Goals

- Design 3D magnetic configurations for improved plasma performance, buildability
  - Improved performance: reducing transport across flux surfaces
    - Neoclassical transport, *turbulent transport*, fast particle transport, …
    - Magnetic fields can be shaped to improve confinement
- Many optimization physics targets require complex and expensive numerical calculations
- Practical approach: use surrogate models capturing essential physics for efficient optimization

Unoptimized



Optimized

Modeling plasma turbulence

- Modeling electrostatic plasma turbulence in toroidal magnetic fields

$$\frac{\partial f_s(\mathbf{x},\mathbf{v})}{\partial t}+\left(\frac{\mathbf{B}\times\nabla\phi}{B^2}+\frac{m_s}{2q_sB}\left(2v_\perp+v_\parallel\right)\frac{\mathbf{B}\times\nabla B}{B^2}\right)\cdot\frac{\partial f_s(\mathbf{x},\mathbf{v})}{\partial\mathbf{x}}+\frac{q_s}{m_s}\left(-\nabla\phi+\mathbf{v}\times\mathbf{B}\right)\cdot\frac{\partial f_s(\mathbf{x},\mathbf{v})}{\partial\mathbf{v}}=0$$

$$-\nabla^2\phi=\sum_s\int\mathrm{d}\mathbf{v}f_s\left(\mathbf{x},\mathbf{v}\right)$$

Physical Complexity

- Range of disparate time, length scales
- Quadratic nonlinearity $\Rightarrow$ multiscale coupling

Computational Complexity

- 6 spatial, 1 time dimension: large grid/memory requirements
- Nonlinear simulation: $\sim 10^6$–10^8 CPU hours

- Reduced turbulence models *necessary* for practical optimization
  - Average over $\mathbf{v}$ (gyrokinetic/fluid model), model fluctuations (δf), small simulation volumes (flux tube)

- Cross-field transport driven by turbulent fluctuations:

$$Q = \int d\mathbf{x}\,(\tilde{\mathbf{v}} \cdot \nabla x)\,\tilde{T} \Rightarrow \int d\mathbf{x} \frac{\mathbf{B}(\mathbf{x}) \times \nabla\tilde{\phi}(\mathbf{x})}{B^2(\mathbf{x})}\tilde{T}(\mathbf{x})$$

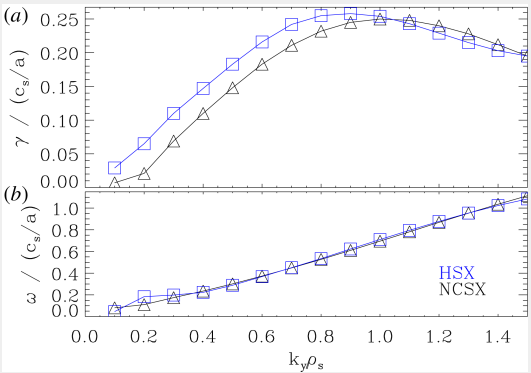
$$\text{local approx. (Fourier)} \Rightarrow -\int \frac{\sqrt{g(z)}dz}{B(z)} \sum_{k_x,k_y} \imath k_y \tilde{\phi}_{k_x,k_y}\tilde{T}_{k_x,k_y}$$

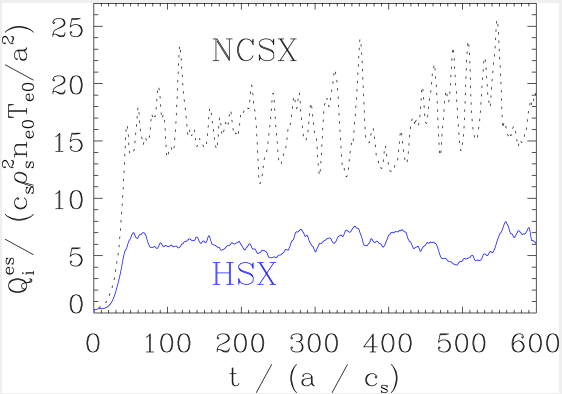
- Energy transport dependent on fluctuation *amplitude* and *phase*
- *Quasilinear* transport modeling:

$$Q \approx -n\chi\nabla T,\ \chi \propto \frac{\Delta x^2}{\Delta t} \sim \sum_k \frac{\gamma_k}{\langle k_\perp^2 \rangle},\ \langle k_\perp^2 \rangle = \frac{\int \sqrt{g(z)}dz|\phi_k(z)|^2 k_\perp^2(z)}{\int \sqrt{g(z)}dz|\phi_k(z)|^2}$$

- Instability growth rate γ_k, eigenfunction ϕ_k obtained from linear calculations

Turbulence in stellarators





From McKinney *et al.*, J. Plasma Phys. 2019

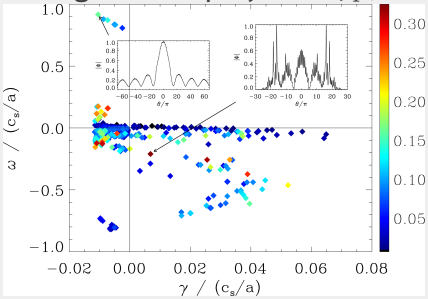
- Substantially different nonlinear transport between stellarator configurations despite similar growth rates
- Not captured by simple quasilinear transport models

Role of stable eigenmodes

- Many stable *eigenmodes* exist at same spatial scales as linear instability
- Provides energy sink, can reduce transport-driving fluctuation amplitudes through nonlinear coupling
- Goal: generate new surrogate models for turbulent transport incorporating stable mode coupling

$$\Rightarrow Q \propto \sum_{j,k} \tau_{jk} \frac{\gamma_k}{\left\langle k_\perp^2 \right\rangle}$$

Eigenmode projection, p_i



$$p_i = \langle f_i^* f_{NL} \rangle / \left\langle f_i^2 \right\rangle^{1/2} \left\langle f_{NL}^2 \right\rangle^{1/2}$$

From Faber *et al.*, J. Plasma Phys. 2018

- Full physics stellarator turbulence simulations show strong signatures of stable mode activity

Modeling stable mode effects

- Three-field fluid model for ion scale fluctuations[1] with eigenmodes $\beta_k = (\Phi_k(z), U_k(z), T_k(z))^T$, eigenvalues ω_k
- Close moment equations with EDQMN closure: $\langle \beta_p \beta_r \beta_s \beta_t \rangle = \langle \beta_p \beta_r \rangle \langle \beta_s \beta_t \rangle + \ldots$
- Express energy evolution as:

$$\frac{d}{dt} \left\langle |\beta_p|^2 \right\rangle = \mathrm{i} \left(\omega_p^* - \omega_p \right) \left\langle |\beta_p|^2 \right\rangle + \sum_{k',s,t} C_{pst}(k,k') \tau_{pst}(k,k') F(\beta_i, k, k') + C.C.$$

$$\tau_{pst} = 1/\mathrm{i} \left(\omega_t + \omega_p - \omega_s^* \right), \text{ three-wave correlation time}$$

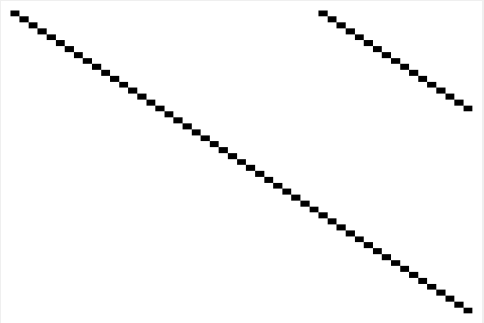
- Theory: increasing energy transfer (τ) to *stable modes* $\Rightarrow$ lowers transport
- Calculate ω_i from *linear* problem: $\mathcal{L}\beta_i = \omega_i \beta_i$
- Construct a "nonlinear" turbulence optimization metric from linear theory

[1] Hegna *et al.*, Phys. Plasmas, 2018

Sparse eigenvalue problem

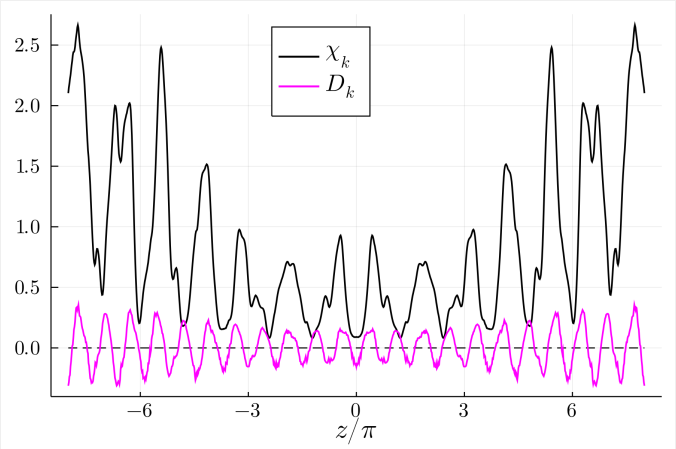
- Fluid eigenmodes solution to generalized eigenvalue problem along magnetic field line: $\mathbf{A}_k \beta_i^k = \lambda_i \mathbf{B}_k \beta_i^k$, with $\lambda = \omega_r + \mathrm{i}\gamma$
- Centered finite difference discretization yields sparse block matrices, recomputed for each (k_x, k_y) evaluated

$\mathbf{A}_k$



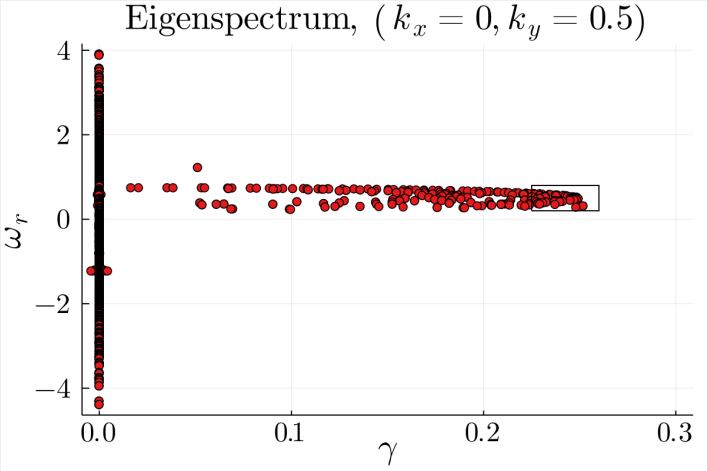
$\mathbf{B}_k$

Geometry dependence



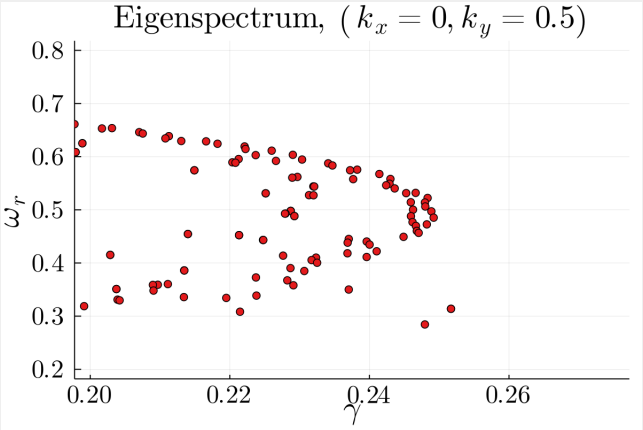
- Complex interplay of model coefficients along field line $\Rightarrow$ complicated geometry dependence of fluctuations

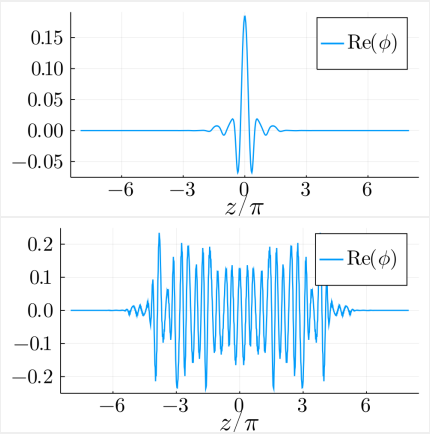
Eigenspectrum properties



- Desire solution only to most physically important eigenmodes

Eigenspectrum properties



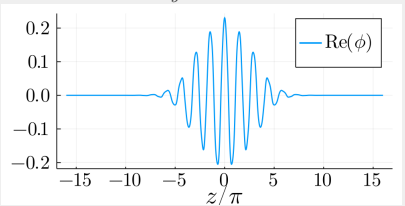


- Different eigenmode branches coexist in small region of complex plane
- Target physically important eigenmodes with sparse iterative solvers

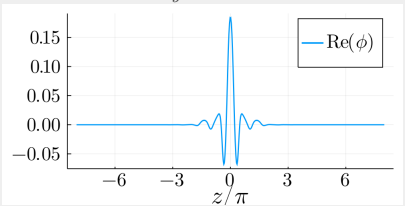
Eigenmode properties

- Multiple, competing eigenmode branches, have different character
  - Classify modes by $\left\langle k_\parallel^2 \right\rangle = \int \sqrt{g} B dz |\partial \mathbf{v}/\partial z|^2 / \int \sqrt{g} B dz |\mathbf{v}_j|^2$
  - Smaller $\left\langle k_\parallel^2 \right\rangle$ modes more transport relevant, balance against γ
- Problem dimensions change with (k_x, k_y)
  - Solution interval grows with decreasing k_y
  - Accurate eigenmode calculation at small k_y essential to model performance
- Solution convergence sensitive to initial guess/target region

$k_y = 0.2$



$k_y = 0.5$



Algorithmic requirements

- Necessary requirements for stellarator optimization:
  - Fast calculation ($\lesssim 60$ seconds) of exterior eigenmodes per (k_x, k_y)
  - Robustness: reliably compute/identify the physically important eigenvalues
- Use methods from the SLEPc[1] library
  - Parallelized for fast calculation on large domains (e.g. 12288×12288)
  - Highly tunable methods to maximize performance
    - Large library of possible methods
    - Support for preconditioners, initializing solver subspace
  - Method used: `EPSKyrlovSchur` with shift-and-invert preconditioning

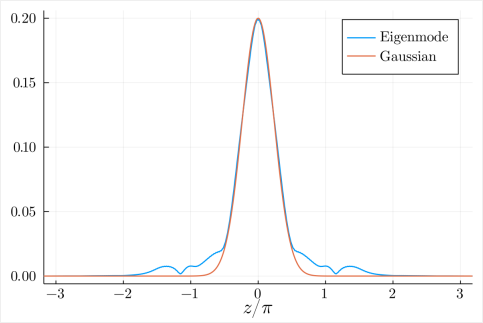
[1] `https://slepc.upv.es`

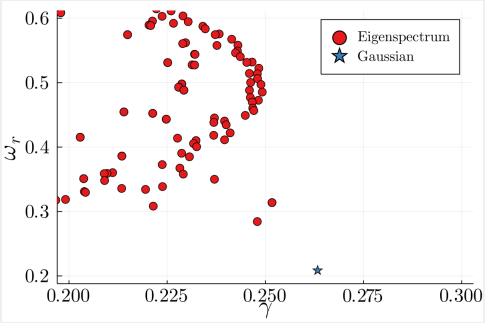
Result: Linear Eigenmode Spectrum
Input: $\left(k_x^0 = 0, k_y^0\right)$, k_x^{max}, k_y^{max}, α, β, ρ, σ
Estimate $\omega_0 = \omega\left(k_x^0, k_y^0\right)$ with Gaussian eigenmode ansatz;
while $k_x^i \leq k_x^{max}$ && $k_y^i \leq k_y^{max}$ **do**
  if $\mathrm{Im}\left(\omega_{i-1}\right) \leq 0$ **then**
    $\omega_i = 0$, $\mathbf{v}_i = \mathbf{0}$;
  else
    Set shift-and-invert target $\lambda_i = \rho\omega_{r,i-1} + \sigma\gamma_{i-1}$, initial subspace $\mathbf{w} = \mathbf{v}_{i-1}$;
    Compute $A_{k,i}\tilde{\mathbf{v}}_i \approx \tilde{\omega}_i B_{k,i}\tilde{\mathbf{v}}_i$ preconditioned by λ_i, $\mathbf{w}$;
    for $(\omega_j, \mathbf{v}_j) \in (\tilde{\omega}_i, \tilde{\mathbf{v}}_i)$ **do**
      Compute $\left\langle k_\parallel^2 \right\rangle_j$;
    Select $(\omega_i, \mathbf{v}_i)$ as solution to $\min_j \left(\left|\tilde{\omega}_j - \lambda_i\right|^\alpha \left\langle k_\parallel^2 \right\rangle_j^\beta \right)$;

Algorithm 1: Linear spectrum with preconditioning

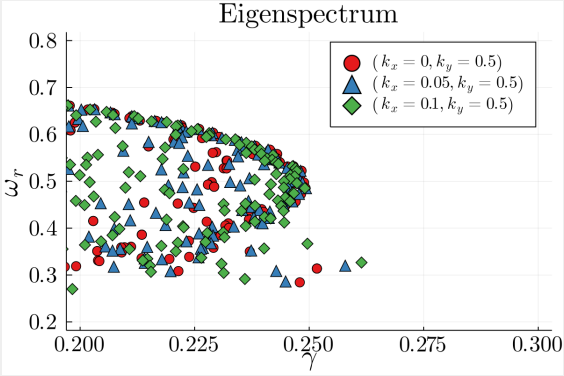
Seeding the initial state

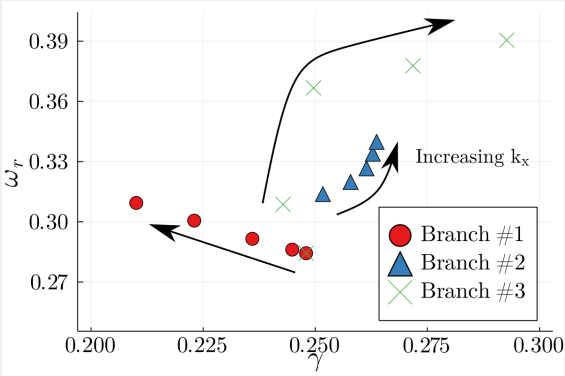
- Approximate initial eigenvector with Gaussian approximation:
  $\phi_k(z) = \exp\left(-\frac{(z-z_0)^2}{2\Delta^2}\right)$
- Approximate eigenvalue is root of 3rd order polynomial for $\phi_k(z)$, use as shift-and-invert preconditioner
- Iterative solver sensitive to initial guess





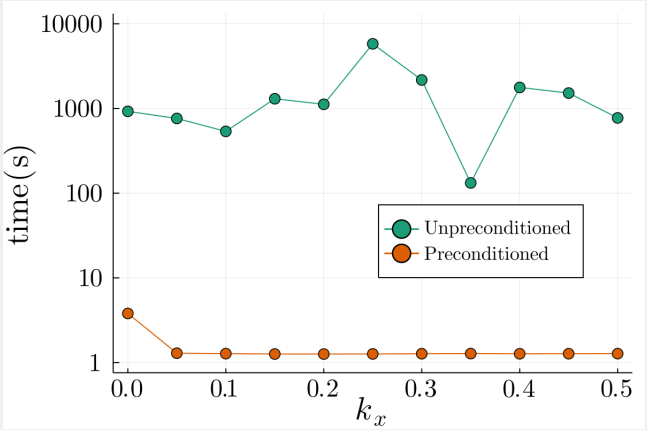
Deforming the eigenspectrum





- Eigenspectrum at nearby (k_x, k_y) shows smooth deformation in $k_x \Rightarrow$ previous eigenvectors provide suitable initial subspace for subsequent iterations
- Different instability branches compete, complicating choice of shift-and-invert parameter

Performance



- Preconditiong improves performance by factor of 10^3

Conclusions + future directions

- Parallel, sparse iterative solvers provide a suitable means for efficient automated eigenmode calculations in stellarators
- Crucial to performance is using previous calculation results to precondition subsequent iterations
  - Shift-and-invert preconditioner isolates the region of physical interest
  - Seeding the subspace with previously computed eigenvectors acclerates convergence
- Future work: incorporate into stellarator optimization frameworks

Backup slides

Fluid model for ITG turbulence saturation in stellarators

- Generalized eigenvalue equation $\mathbf{A}_k \beta_i^k = \lambda_i \mathbf{B}_k \beta_i^k$, with $\lambda = \omega_r + \mathrm{i}\gamma$

$$-\mathrm{i}D_k\left(\left(1+\frac{5}{3}\xi\right)\Phi_k + \xi T_k\right) + \mathrm{i}k_y\epsilon_n\Phi_k + \frac{1}{\sqrt{g}B}\frac{\partial U_k}{\partial z} - \left(\frac{\varepsilon}{\sqrt{g}B}\right)^2\frac{\partial^2\Phi_k}{\partial z^2} = \mathrm{i}\lambda\left[\left(1+\chi_k\left(1+\frac{5}{3}\xi\right)\right)\Phi_k + \chi_k\xi T_k\right],$$

$$\frac{1}{\sqrt{g}B}\frac{\partial}{\partial z}\left(\left(1+\frac{5}{3}\xi\right)\Phi_k + \xi T_k\right) - \left(\frac{\varepsilon}{\sqrt{g}B}\right)^2\frac{\partial^2 U_k}{\partial z^2} = \mathrm{i}\lambda U_k,$$

$$\mathrm{i}k_\alpha\left(\epsilon_T - \frac{2}{3}\epsilon_n\right)\Phi_k - \left(\frac{\varepsilon}{\sqrt{g}B}\right)^2\frac{\partial^2 T_k}{\partial z^2} = \mathrm{i}\lambda T_k$$

- Stellarator geometry encoded by χ_k, D_k for $\mathbf{B}(\psi,\alpha,z) = \nabla\psi(z)\times\nabla\alpha(z)$:

$$D_k = (k_\psi\nabla\psi + k_\alpha\nabla\alpha)\cdot\frac{\mathbf{B}\times\nabla B}{B^2},\ \chi_k = k_\psi^2\nabla\psi\cdot\nabla\psi + 2k_\psi k_\alpha\nabla\psi\cdot\nabla\alpha + k_\alpha^2\nabla\alpha\cdot\nabla\alpha$$

- Numerical dissipation included with hyperdiffusion term $\varepsilon^2\frac{\partial^2}{\partial z^2}$
- Model parameters $\epsilon_T = -(L/T)\nabla T$, $\epsilon_n = -(L/n)\nabla n$, $\xi = T_\mathrm{e}/T_\mathrm{i}$