Full Flux Surface (FFS) δf-gyrokinetic code; `stella`

G. O. Acton[1,2], S. Newton[2] , H Thienpondt [5] ,F. Parra[3], W. Dorland [4], M. Barnes[1]

[1]Rudolf Peierls Centre For Theoretical Physics,
University of Oxford, OXford, OX1 3PU, UK

[2]Culham Centre for Fusion Energy,
United Kingdom Atomic Energy Authority,
Abingdon, OX14 3EB, UK

[3]Princeton University

[4]Maryland University [5]CIEMAT

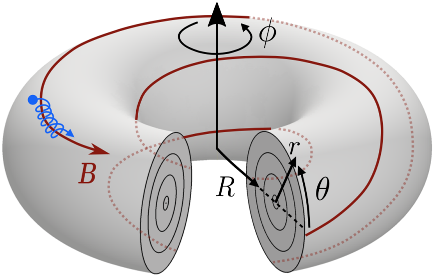
Vienna Workshop

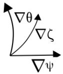
Table of Contents

To get fusion working

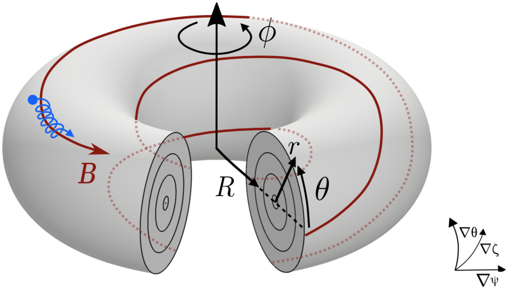
Table of Contents

Tokamak formalism



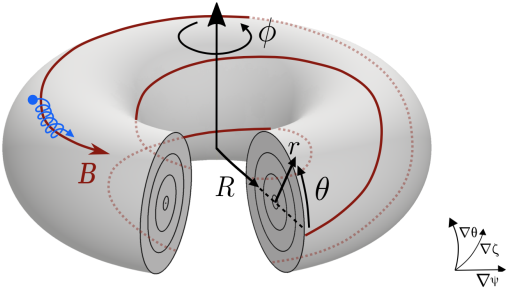


Tokamak formalism



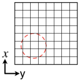
- Magnetic field geometry described by Clebsch representation: $\boldsymbol{B} = \boldsymbol{\nabla}\alpha \times \boldsymbol{\nabla}\psi$
- Cooridinate system: $(\psi, \theta, \zeta) \leftrightarrow (x, y, z)$
- $\psi \leftrightarrow$ flux surface label $\leftrightarrow x$
- $\theta \leftrightarrow$ poloidal angle $\leftrightarrow z$ (for tokamak)
- $\zeta \leftrightarrow$ toroidal angle $\leftrightarrow z$ (for stellarators)
- $\alpha = \theta - \iota\zeta \leftrightarrow y$

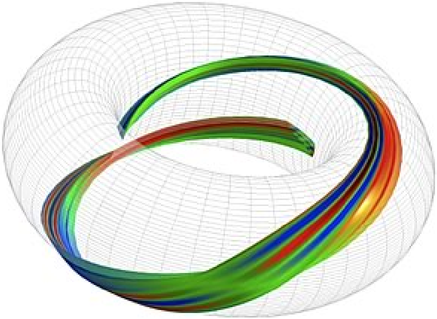
Tokamak formalism



- Despite fusion being only 10 years away, δf-gyrokinetic simulations for full devices are out of numerical reach
- Consider just a single field line and simulate turbulence along it

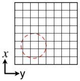
Current state of affairs

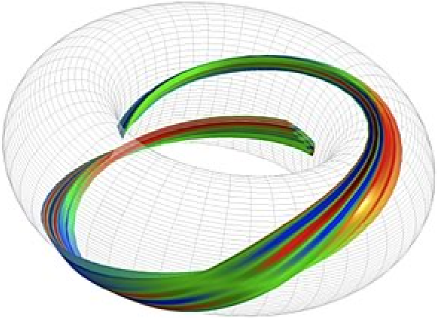




- Simulation coordinates: $(x, y, z) \to (\psi, \alpha, \zeta)$

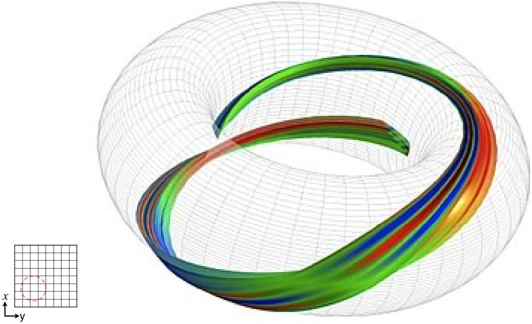
Current state of affairs





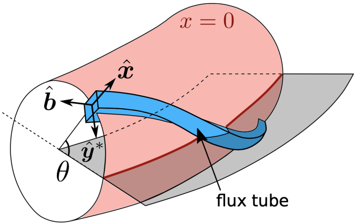
- Simulation coordinates: $(x, y, z) \rightarrow (\psi, \alpha, \zeta)$
- Initialise some δf and ϕ at $t = 0$ on simulation domain
- Evolve gyrokinetic equation pseudo-spectrally

Current state of affairs

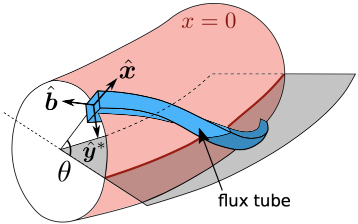


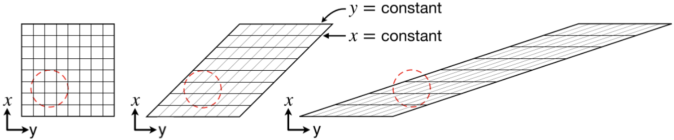
- Simulation coordinates: $(x, y, z) \to (\psi, \alpha, \zeta)$
- Initialise some δf and ϕ at $t = 0$ on simulation domain
- Evolve gyrokinetic equation pseudo-spectrally
- Decay in $v_\parallel$; $g(t, \boldsymbol{x}, v_\parallel \to \pm\infty, \mu) \to 0$
- Turbulence is taken as periodic in perpendicular directions, $k_x, k_y \gg 1/L$
- Use twist and shift boundary conditions in z to capture extended modes

Twisting and shifting



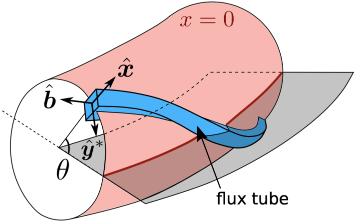
Twisting and shifting

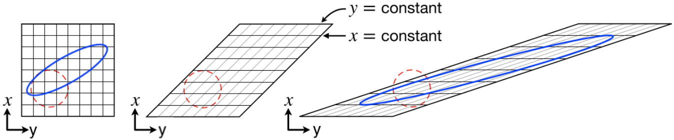




- If $\hat{s} \propto \mathrm{d}q/\mathrm{d}\psi \neq 0$ then domain gets sheared as it travels around device

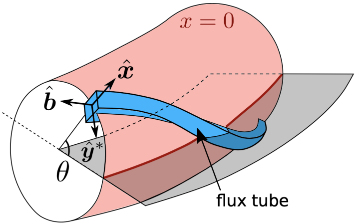
Twisting and shifting

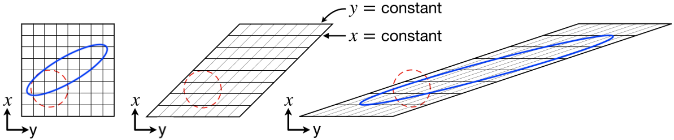




- If $\hat{s} \propto \mathrm{d}q/\mathrm{d}\psi \neq 0$ then domain gets sheared as it travels around device
- Eddies get sheared
- Pushed to higher perpendicular wavenumbers

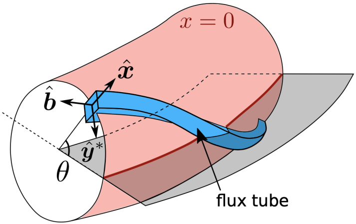
Twisting and shifting

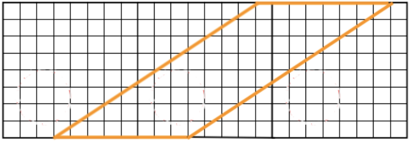




- Use "twist-and-shift" boundary conditions to map sheared domain back onto original one

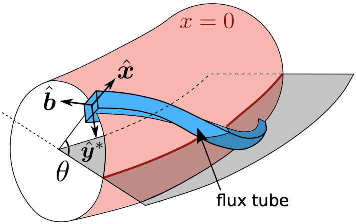
Twisting and shifting

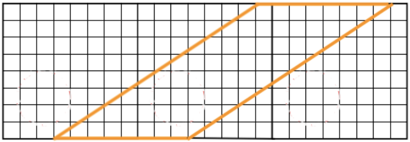




- Demand that any function, $A(t, x, y, z) = \sum_k \hat{A}_{k_x, k_y}(t, z) e^{ik_y(y-y_0)+ik_x(x-x_0)}$, be periodic at the same poloidal location, θ

Twisting and shifting

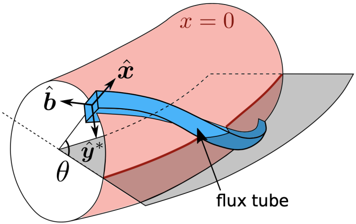


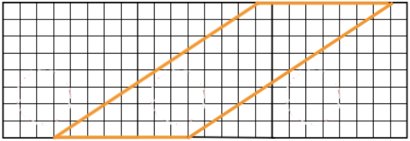


- Demand that any function, $A(t, x, y, z) = \sum_k \hat{A}_{k_x,k_y}(t, z)e^{ik_y(y-y_0)+ik_x(x-x_0)}$, be periodic at the same poloidal location, θ
- But $y = y(\psi(x), \theta, \zeta) = \zeta - q(\psi)\theta$

$$A(t, x, y(x, \theta, z), z) = A(t, x, y(x, \theta, z + 2p\pi), z + 2p\pi) \quad (1)$$

Twisting and shifting





- Demand that any function, $A(t,x,y,z) = \sum_k \hat{A}_{k_x,k_y}(t,z) e^{ik_y(y-y_0)+ik_x(x-x_0)}$, be periodic at the same poloidal location, θ
- But $y = y(\psi(x),\theta,\zeta) = \zeta - q(\psi)\theta$

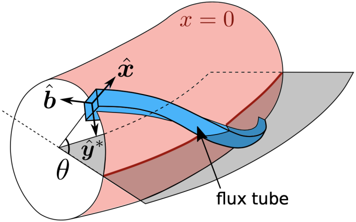
$$A(t,x,y(x,\theta,z),z) = A(t,x,y(x,\theta,z+2p\pi),z+2p\pi) \tag{1}$$

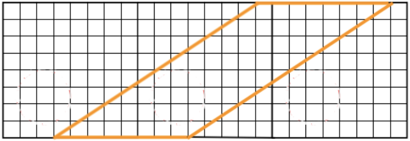
- Let $\boldsymbol{k}'$ be the wavenumber that satisfies the above:

$$\hat{A}_{\boldsymbol{k}}(z) = \hat{A}_{\boldsymbol{k}'}(z+2p\pi)(\text{phase factor}) \tag{2}$$

- $\{k'_x, k'_y\} = \{k_x + \delta k_x, k_y\}$
- $\delta k_x \propto \frac{\mathrm{d}y}{\mathrm{d}x}$, $\delta k_x = 2\pi\hat{s}k_y$
- phase factor $= -2\pi p i k_y \frac{\mathrm{d}y}{\mathrm{d}\alpha}$

Twisting and shifting





- Demand that any function, $A(t, x, y, z) = \sum_k \hat{A}_{k_x,k_y}(t, z)e^{ik_y(y-y_0)+ik_x(x-x_0)}$, be periodic at the same poloidal location, θ
- But $y = y(\psi(x), \theta, \zeta) = \zeta - q(\psi)\theta$

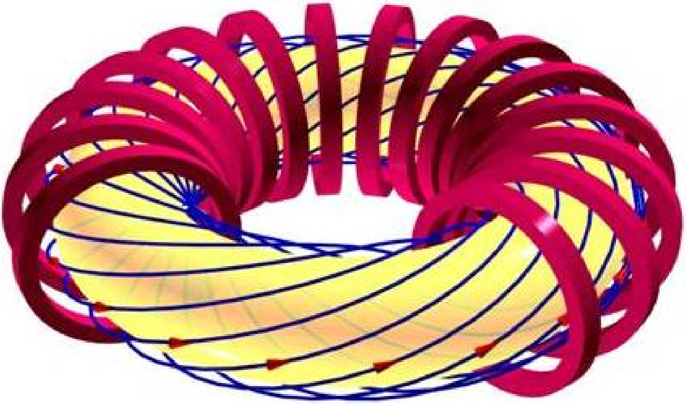
$$A(t, x, y(x, \theta, z), z) = A(t, x, y(x, \theta, z + 2p\pi), z + 2p\pi) \quad (1)$$

- Let $\boldsymbol{k}'$ be the wavenumber that satisfies the above:

$$\hat{A}_{\boldsymbol{k}}(z) = \hat{A}_{\boldsymbol{k}'}(z + 2p\pi)(\text{phase factor}) \quad (2)$$

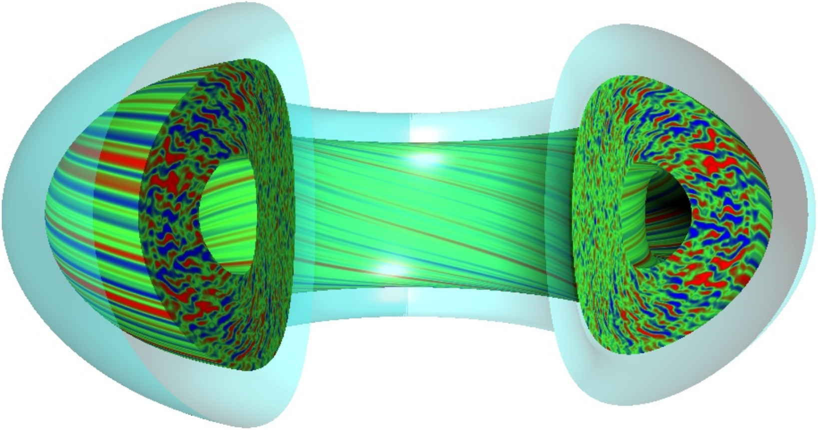
- $\delta k_x = 2\pi\hat{s}k_y$
- $\delta k_x = N\Delta k_x$
- $2\pi\hat{s}k_y = N\Delta k_x$
- $j_{\text{twist}} = \frac{2\pi\hat{s}L_x}{L_y} \in \mathbb{Z}$

Tokamaks are trivial



- Fluxtube simulations are sufficient for Tokamak because we can stitch our fluxtubes together

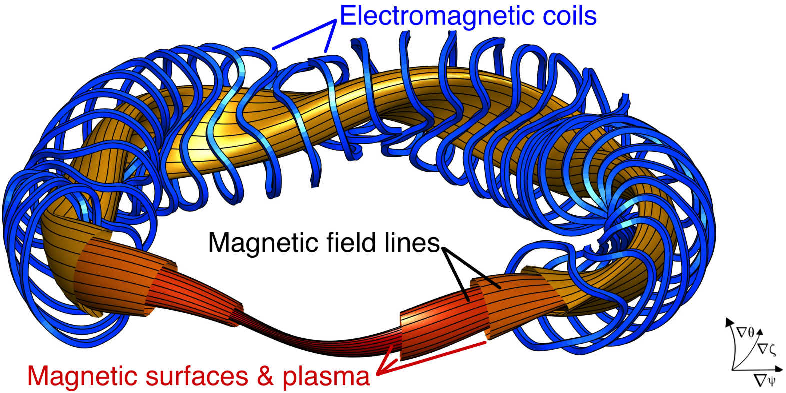
Tokamaks are trivial



- Fluxtube simulations are sufficient for Tokamak because we can stitch our fluxtubes together

Table of Contents

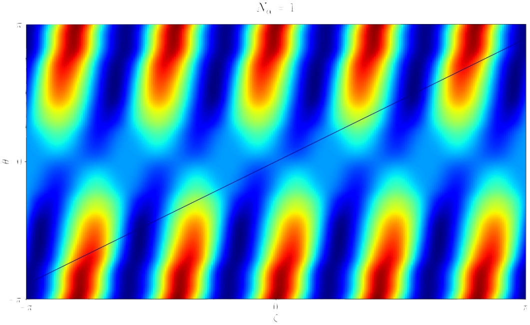
Stepping outside the axisymmetric box



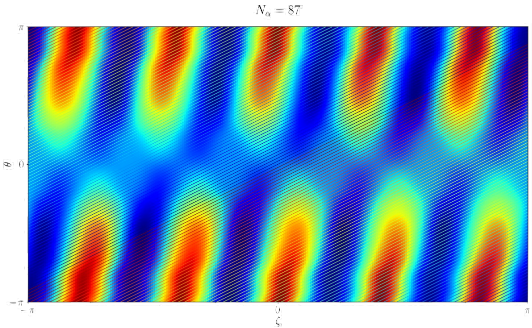
- Geometry sampled varies with field line
- Method of stitching together fluxtubes no longer holds

- Currently no δf poloidally-global code that can correctly deal with kinetic electrons

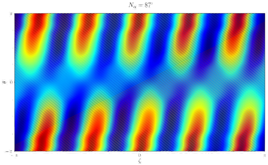
What do we expect to be different?

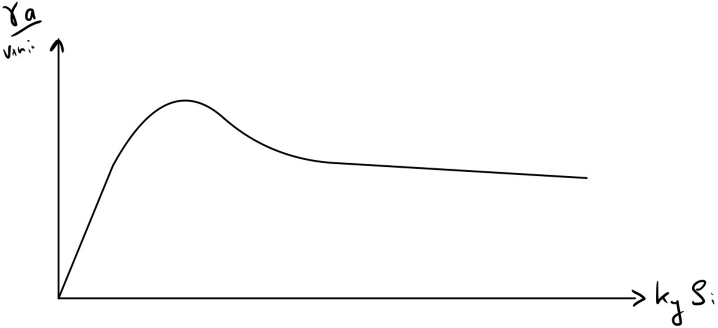


What do we expect to be different?



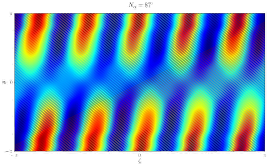
What do we expect to be different?

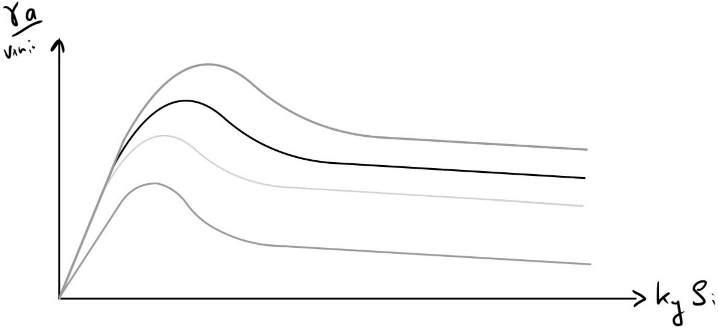




- We have a k_y spectrum for a given fieldlines

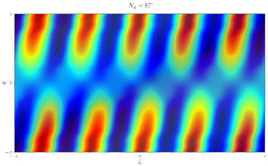
What do we expect to be different?

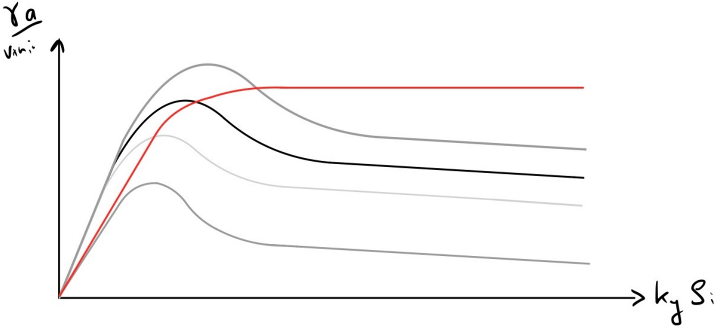




- Different fieldlines have different spectra

What do we expect to be different?





- Full flux effects should sample all fieldlines

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Simplified notation GK:

$$\frac{\partial g}{\partial t} = (\text{geometric factors}) \cdot \left(\boldsymbol{\nabla} g + \boldsymbol{\nabla} \langle \phi \rangle_{\boldsymbol{R}} \right) \tag{3}$$

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Simplified notation GK:

$$\frac{\partial g}{\partial t} = \underbrace{(\text{geometric factors})}_{\text{e.g } \hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} z} \cdot \left(\boldsymbol{\nabla} g + \boldsymbol{\nabla} \underbrace{\langle \phi \rangle_{\boldsymbol{R}}}_{J_{0,\boldsymbol{k}} \hat{\phi}_{\boldsymbol{k}}} \right) \tag{3}$$

- Bessel functions $J_0(a_{\boldsymbol{k}})$ with $a_{\boldsymbol{k}} = \frac{k_\perp v_\perp}{\Omega_s}$
- Geometric factors are α-dependent

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Simplified notation GK:

$$\frac{\partial g}{\partial t} = \underbrace{(\text{geometric factors})}_{\text{e.g } \hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} z} \cdot (\boldsymbol{\nabla} g + \boldsymbol{\nabla} \underbrace{\langle \phi \rangle_{\boldsymbol{R}}}_{J_{0,\boldsymbol{k}} \hat{\phi}_{\boldsymbol{k}}}) \tag{3}$$

- Bessel functions $J_0(a_{\boldsymbol{k}})$ with $a_{\boldsymbol{k}} = \frac{k_\perp v_\perp}{\Omega_s}$
- Geometric factors are α-dependent
- Gyro-averaging introduces coupling between different k_y-modes $\rightarrow$ no longer a local operation. But want to retain spectral accuracy

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Simplified notation GK:

$$\frac{\partial g}{\partial t} = \underbrace{(\text{geometric factors})}_{\text{e.g } \hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} z} \cdot (\boldsymbol{\nabla} g + \boldsymbol{\nabla} \underbrace{\langle \phi \rangle_{\boldsymbol{R}}}_{J_{0,\boldsymbol{k}} \hat{\phi}_{\boldsymbol{k}}}) \tag{3}$$

- Bessel functions $J_0(a_{\boldsymbol{k}})$ with $a_{\boldsymbol{k}} = \frac{k_\perp v_\perp}{\Omega_s}$
- Geometric factors are α-dependent
- Gyro-averaging introduces coupling between different k_y-modes $\rightarrow$ no longer a local operation. But want to retain spectral accuracy
- Geometric factors are introducing non-linear terms in k_y

Where there is geometry there are problems

Geometry is at the root of all evil...

- Geometry is no longer trivial
- But how does geometry enter our code?

.

Simplified notation GK:

$$\frac{\partial g}{\partial t} = (\text{geometric factors}) \cdot (\boldsymbol{\nabla} g + \boldsymbol{\nabla} \langle \phi \rangle_{\boldsymbol{R}}) \tag{3}$$

- Bessel functions $J_0(a_{\boldsymbol{k}})$ with $a_{\boldsymbol{k}} = \frac{k_\perp v_\perp}{\Omega_s}$
- Geometric factors are α-dependent
- Gyro-averaging introduces coupling between different k_y-modes $\rightarrow$ no longer a local operation
- Geometric factors are introducing non-linear terms in k_y
- Domain changes for each fieldline

Table of Contents

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's
- Solution: Michael Barnes

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's
- Solution: Michael Barnes

- Bessel functions, QN
- Geometric factors
- Non-constant domain

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's
- Solution: Michael Barnes

- Bessel functions, QN
- Geometric factors
- Non-constant domain

- Fourier Decompose functions once at the beginning of code

$$\langle \phi \rangle_{\boldsymbol{R}} = \sum_{\mathbf{k}''} e^{i\mathbf{k}''\cdot\mathbf{R}} J_0(a_{\mathbf{k}'',s}) \hat{\phi}_{\mathbf{k}''}, \tag{4}$$

$$(\langle \phi \rangle_{\boldsymbol{R}})_{\boldsymbol{k}} = \int d^2\mathbf{R} \sum_{\mathbf{k}'',k'_\alpha} e^{i\left(k''_\psi - k_\psi\right)\psi} e^{i\left(k'_\alpha + k''_\alpha - k_\alpha\right)\alpha} \hat{J}_{\mathbf{k}'',k'_\alpha,s} \hat{\phi}_{\mathbf{k}''}, \tag{5}$$

where we have used

$$J_0(a_{\mathbf{k}'',s}) = \sum_{k'_\alpha} \hat{J}_{\mathbf{k}'',k'_\alpha,s}(z,\mu) e^{ik'_\alpha \alpha}. \tag{6}$$

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's
- Solution: Michael Barnes

- Bessel functions, QN
- Geometric factors
- Non-constant domain

- Work with $\bar{g} = \frac{g}{F_0}$
- Multiply coefficients in real space
- Modify implicit terms

Overcoming adversity

- Problem: A full flux δf-gyrokinetic code couples together all k_y's
- Solution: Michael Barnes

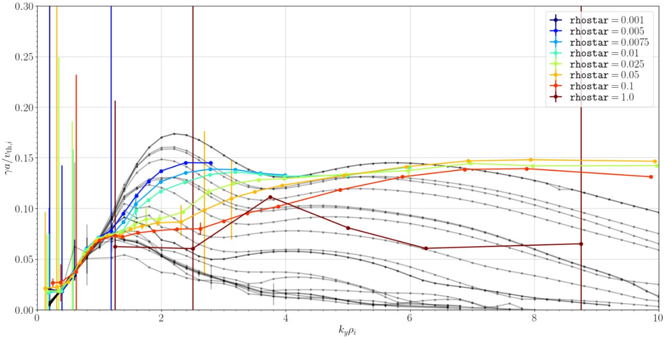
- Bessel functions, QN
- Geometric factors
- Non-constant domain

- Fixed 2π poloidal domain.in θ, now set Δk_y using ρ_*
- Fieldlines will vary in path length around device
- Modify coordinates accordingly, and make "large aspect ratio approximation" for terms treated implicitly

$$\hat{b} \cdot \boldsymbol{\nabla} z = \langle \hat{b} \cdot \boldsymbol{\nabla} z \rangle_\alpha + \left[\hat{b} \cdot \boldsymbol{\nabla} z - \langle \hat{b} \cdot \boldsymbol{\nabla} z \rangle_\alpha \right] \tag{4}$$

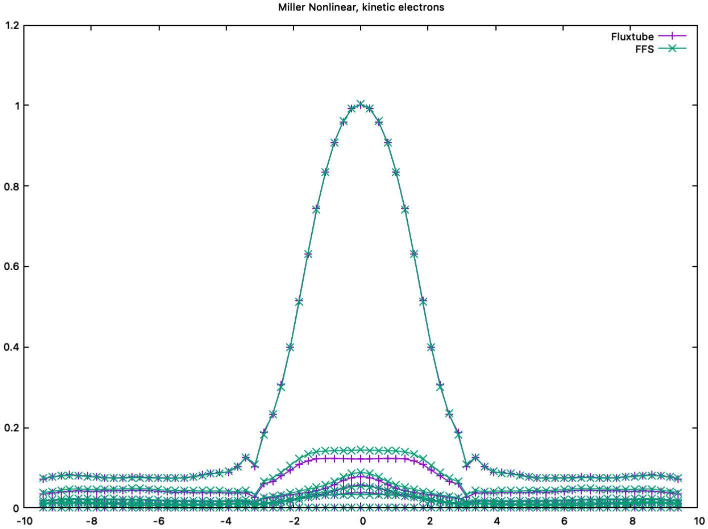
Table of Contents

Rhostar scans

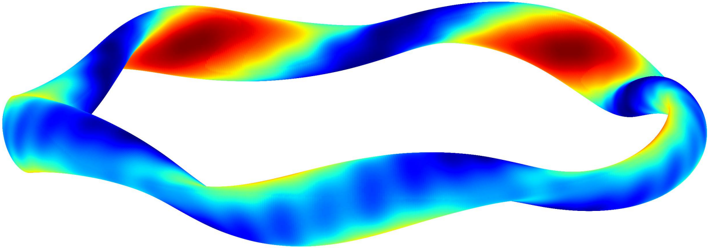


Sanity Check

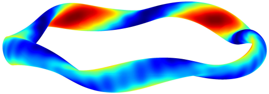
- Miller Geometry:



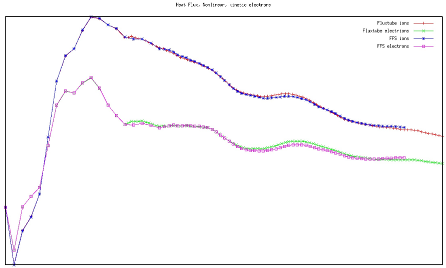
Sprinkle in some geometry

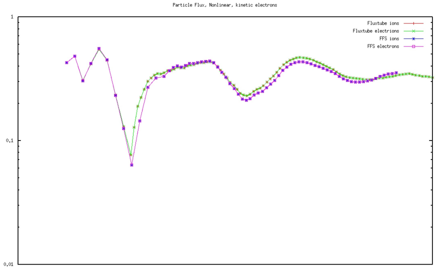


Sprinkle in some geometry



- W7-X, nonlinear with kinetic electrons

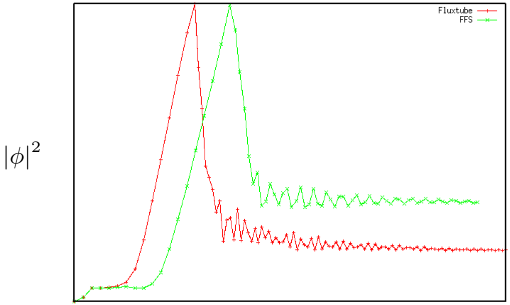




WARNING

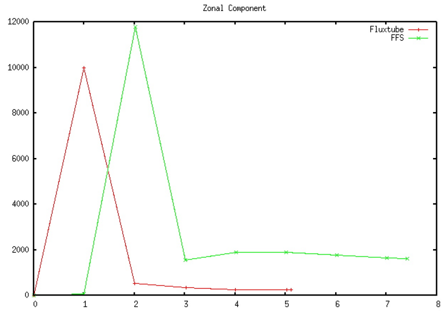
WARNING

- Zonal modes may not be being treated correctly



WARNING

- Zonal modes may not be being treated correctly



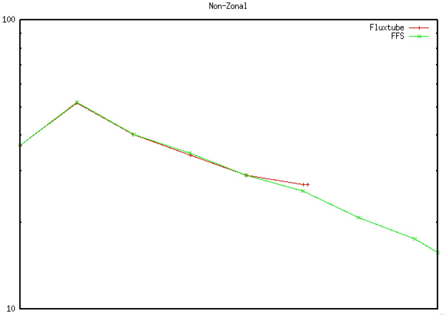


Table of Contents

Why do we care?

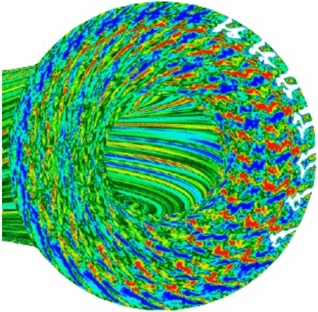
Want to answer some global questions

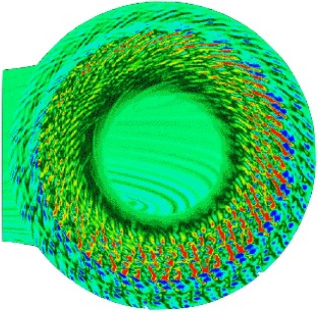
- Look at effects on transport
- Look at global modes, e.g. TEM

Why do we care?

Want to answer some global questions

- Look at effects on transport
- Look at global modes, e.g. TEM
- Can stellarators support zonal flows





Summary

- Aim for poloidally-global δf-gyrokinetic code
- Making progress but zonal flows are being dealt with incorrectly

Thank You