The Gyrokinetic Tokamak Simulation Code: *Algorithms for Strongly-shaped Plasmas*

Stéphane Ethier
Weixing Wang

Princeton Plasma Physics Laboratory

NumKin 2013 workshop
Sep 2013, IPP-Garching

Work supported by the DOE contract DE-AC02-09CH11466

PPPL
PRINCETON
PLASMA PHYSICS
LABORATORY

Outline

- The usual intro to GTS
- How do we do it? Weight control
- How do we do it? Profile restoration
- New GTS feature: Neoclassical

Gyrokinetic Toroidal Simulation code (GTS)

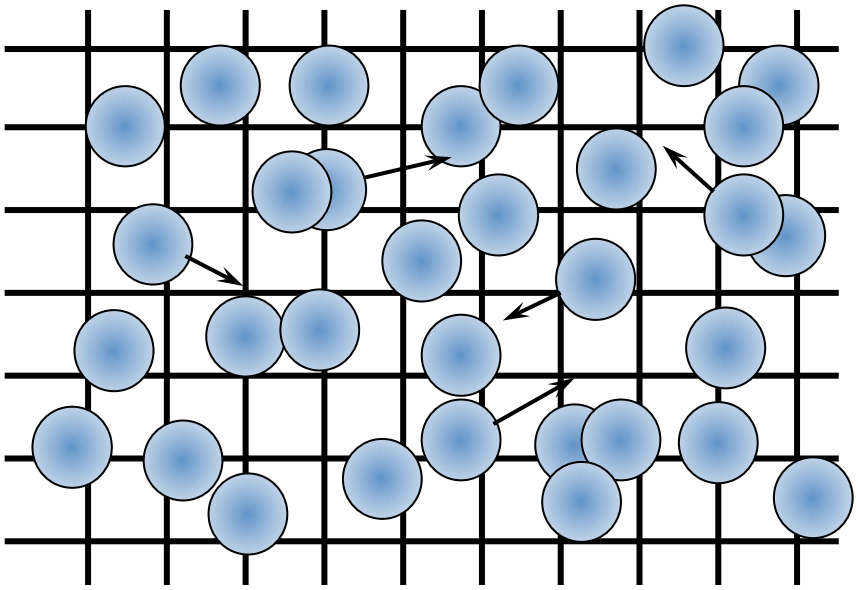
(Wang et al., PoP'06, PoP'10)

- Global 3D gyrokinetic particle-in-cell code
- Used for non-linear simulations of plasma microturbulence and transport
- Toroidal geometry
- Fully self-consistent
- Uses magnetic coordinates (ψ,θ,ζ) and field line following grid
- Energy and momentum conserving guiding center Lagrangian equations for particles time-stepping
- 4-point average method for charge deposition in real space (Lee87)
- Non-spectral Poisson solver (using PETSc)
- Low numerical noise algorithm ($\delta \boldsymbol{f}$ method – Parker & Lee)
- Kinetic ions and electrons (or adiabatic electrons)
- Robust implementation of general plasma geometry (can handle strongly-shaped plasmas such as NSTX)
- General geometry with input profiles from experiments (through the TRANSP database and equilibrium reconstruction with ESC code)
- Linearized Fokker-Planck collision operator

PPPL

The particle-in-cell method

- Particles sample distribution function
- Interactions via the grid, on which the potential is calculated (from deposited charges).
- Grid resolution dictated by Debye length or gyroradius



The PIC Steps

- **"SCATTER", or deposit, charges on the grid (nearest neighbors)**
- **Solve Poisson equation**
- **"GATHER" forces on each particle from potential**
- **Move particles (PUSH)**
- **Repeat…**

PPPL

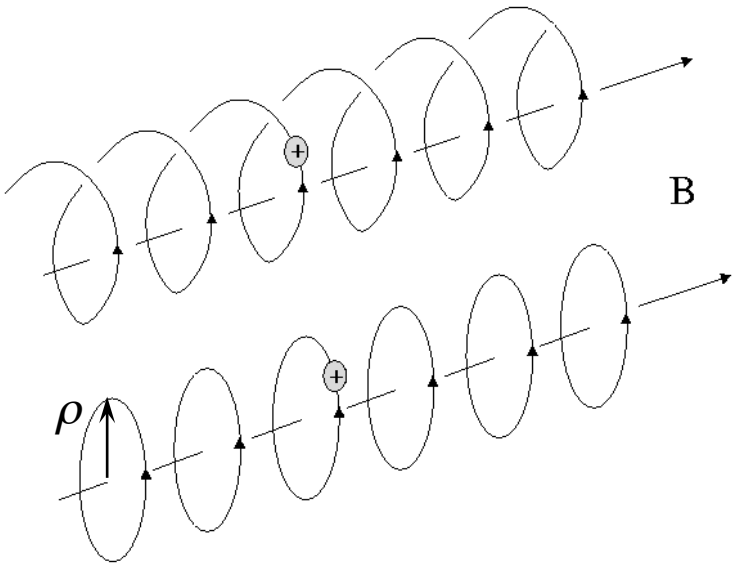
Gyrokinetic approximation for low frequency modes

- Gyrokinetic ordering

$$\frac{\omega}{\Omega} \sim \frac{\rho}{L} \sim \frac{e\phi}{T} \sim k_{//}\rho << 1$$

$$k_{\perp}\rho \sim 1$$

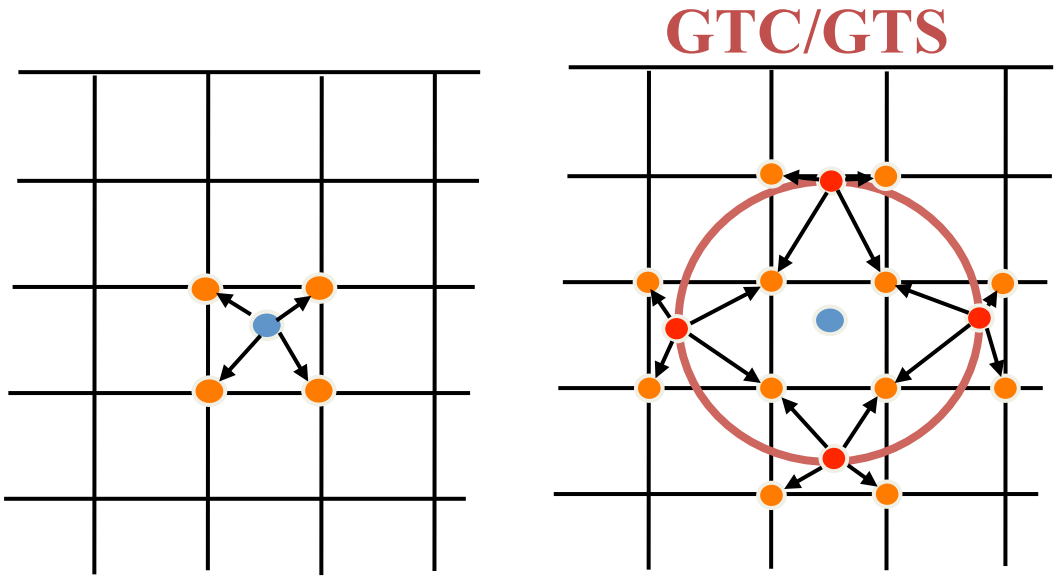
- Gyro-motion: guiding center drifts + charged ring
  - Parallel to B: mirror force, magnetically trapped
  - Perpendicular: E x B, polarization, gradient, and curvature drifts
- Gyrophase-averaged **5D** gyrokinetic equation
  - Suppress plasma oscillation and gyro-motion
  - Larger time step and grid size, smaller number of particles



PPPL

Charge deposition:
4-point average method

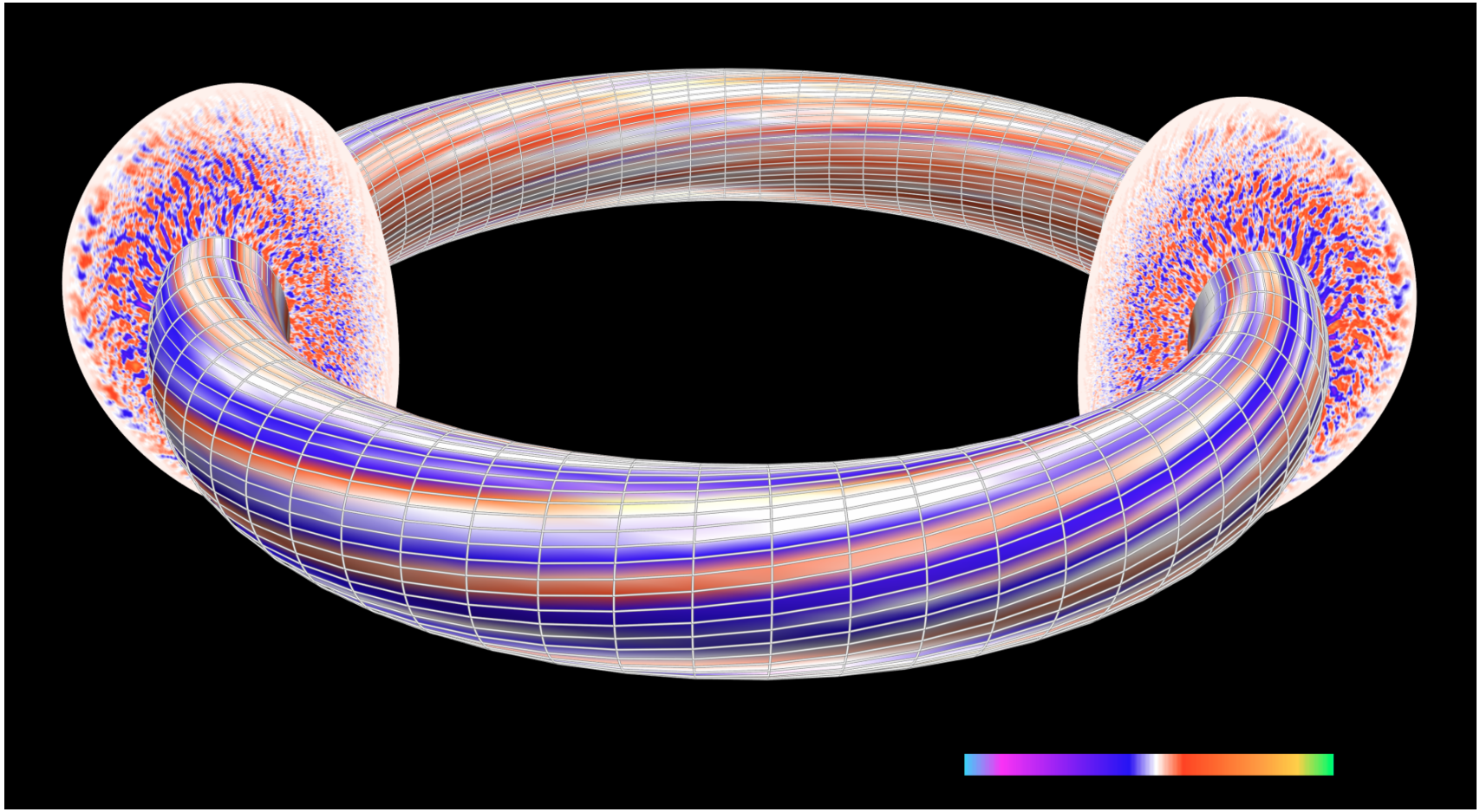
Charge Deposition Step (SCATTER operation)



Classic PIC

4-Point Average GK
(W.W. Lee)

Pretty picture showing quasi-2D structure of electrostatic potential



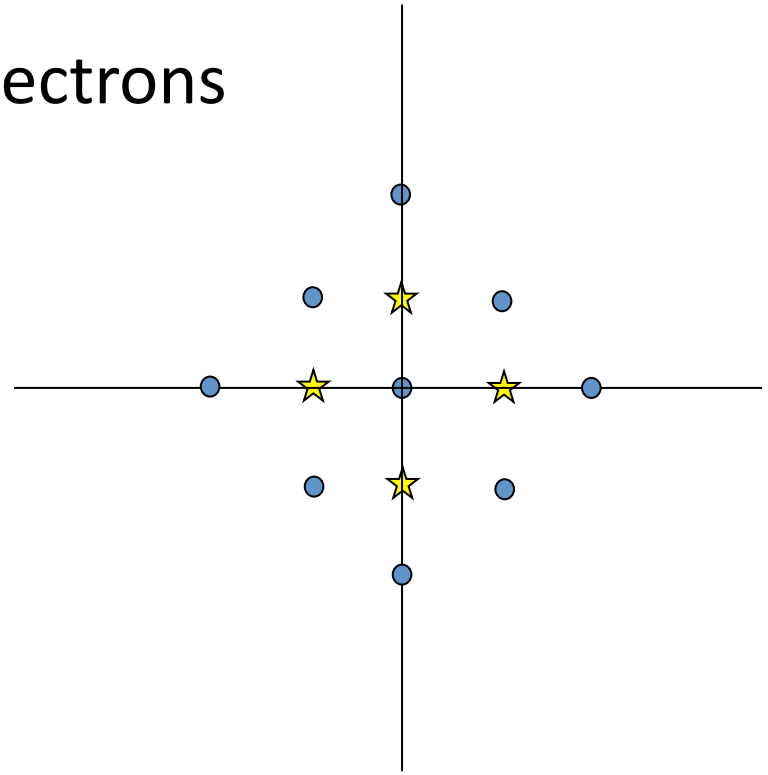
From Pr. Kwan-Liu Ma's students at UC Davis

Poisson Equation Solver

- Done in real space (iterative solver)
- 16-point stencil
- Hypre via PETSc for kinetic electrons

$$\frac{\tau}{\lambda_D^2}\left(\Phi - \tilde{\Phi}\right) = 4\pi e\left(\bar{n}_i - n_e\right)$$

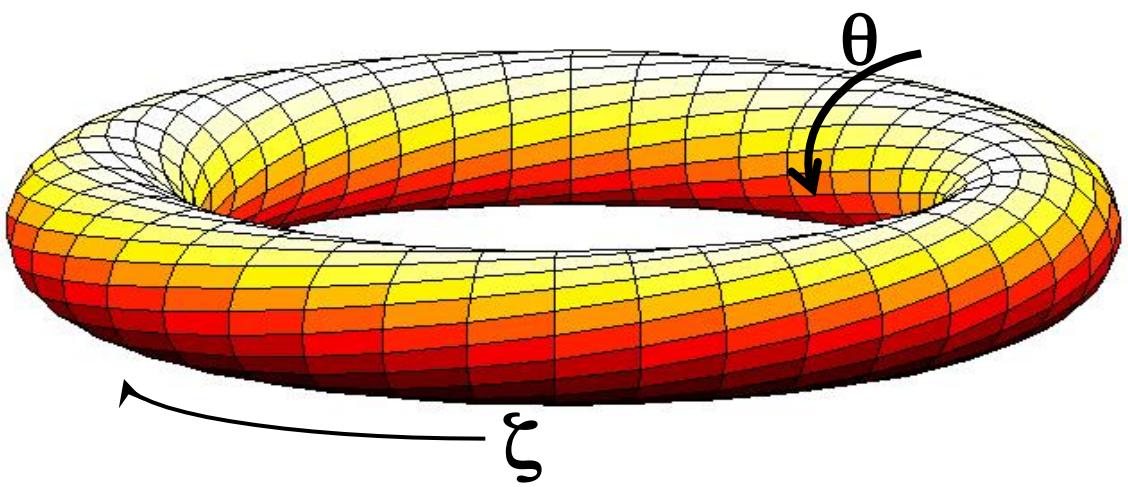
where $\tilde{\Phi}$ is the second gyrophase-averaged potential

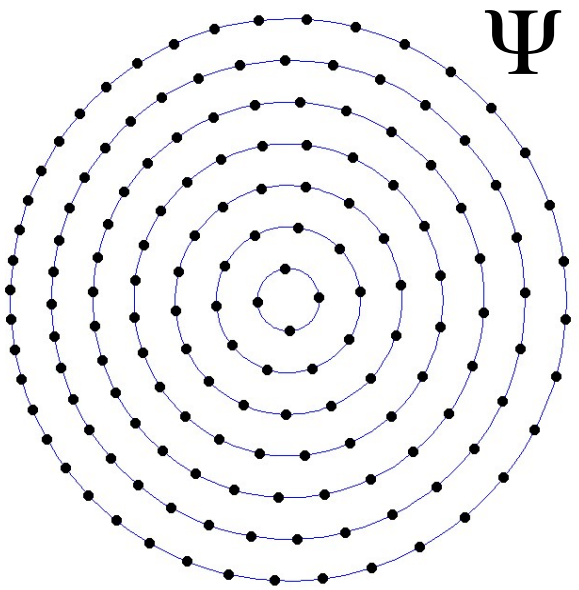


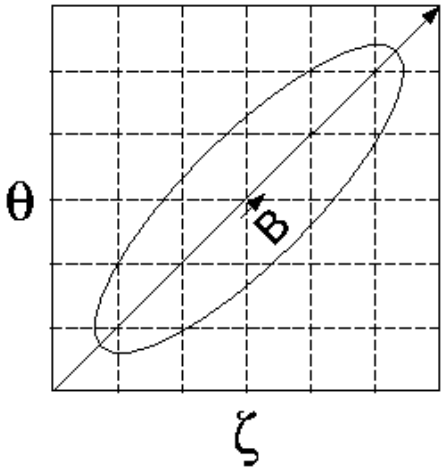
[Z. Lin and W. W. Lee, *Phys.Rev. E* **52**, 5646--5652 (1995)]

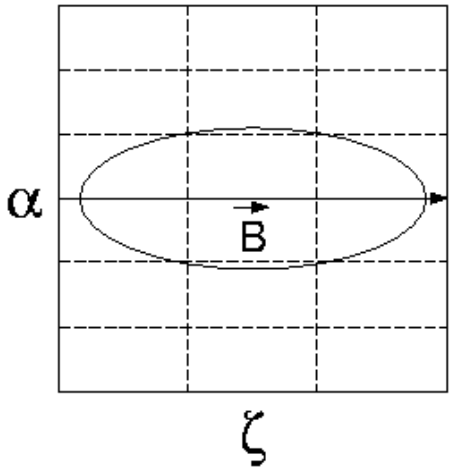
Global Field-aligned Mesh

$(\Psi,\alpha,\zeta) \Rightarrow \alpha = \theta - \zeta/q(\Psi)$





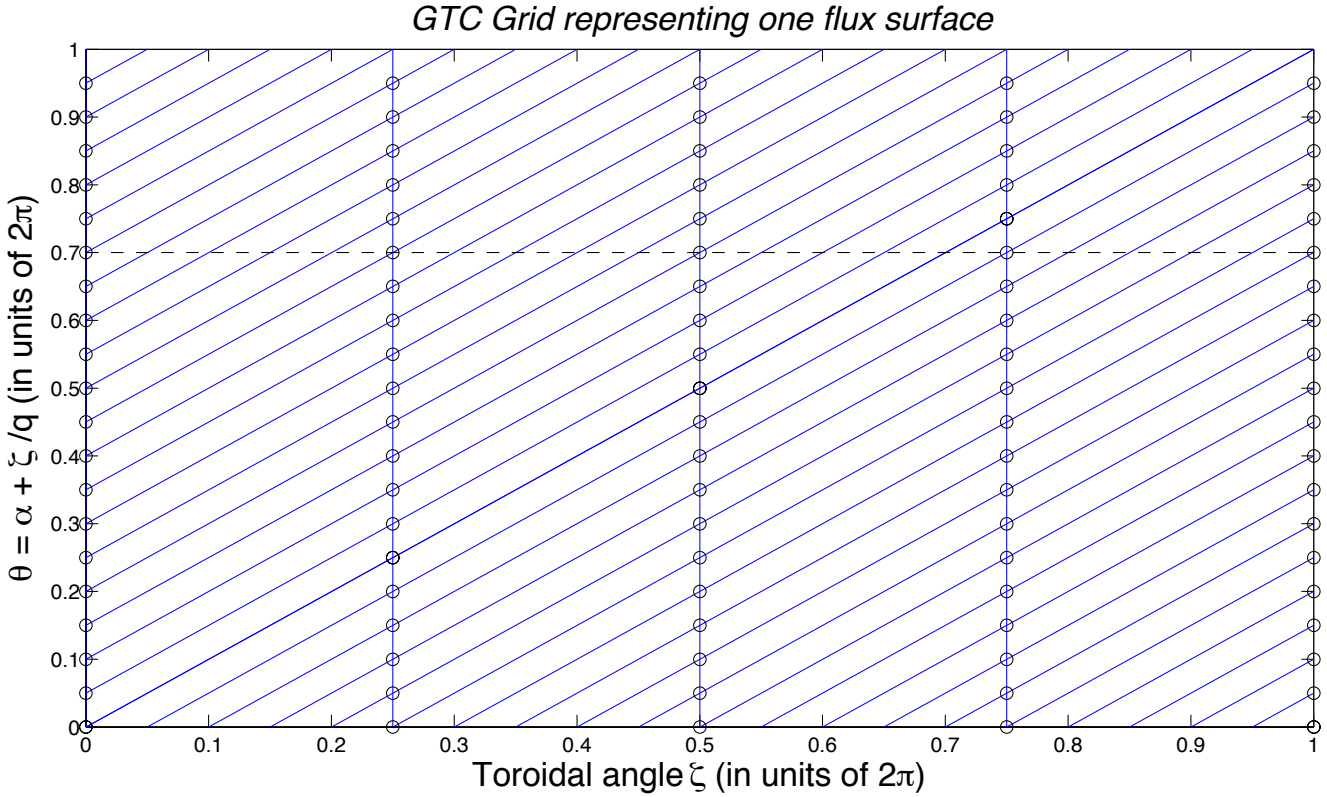




Saves a factor of about
100 in CPU time

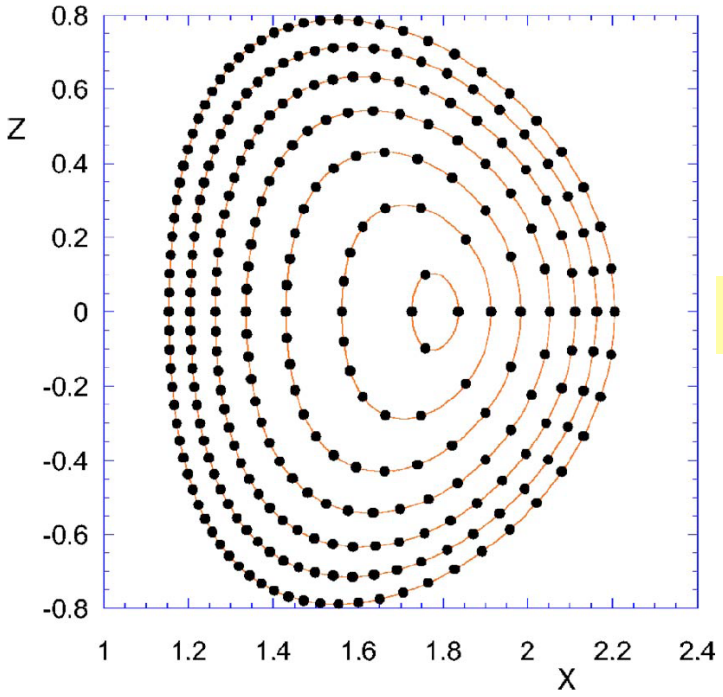
More toroidal modes than planes

- The field-line following “twisted” mesh naturally includes a large number of toroidal modes (linked to poloidal resolution)



Grid follows change in gyro-radius with temperature profile

- Local gyro-radius proportional to temperature $\rho \propto \sqrt{T}$
- Evenly spaced radial grid in new ρ coordinate where $\frac{d\rho}{dr} = \sqrt{T_c / T_i(r)}$



GTS grid follows T(r)

Using PEST1 magnetic coordinates for "flat" planes

Gyrokinetic equation and weight damping

Start from the "full-*f*" gyrokinetic equation

- Solving modern gyrokinetic equation in conservative form for $f(Z,t)$

$$\frac{\partial f_a}{\partial t} + \frac{1}{B^*}\nabla_Z \cdot (\dot{Z}B^* f_a) = \sum_b C\left[f_a, f_b\right]$$

See, e.g., Brizard & Hahm, Rev. Mod. Phys. (2007)

where $\vec{Z}$ represents the phase space coordinates $\vec{R},\vec{V}$

PPPL

Move to a δf representation

- Using the δf method (importance sampling) $\delta f = f - f_0$

$$\frac{\partial \delta f_a}{\partial t} + \frac{1}{B^*}\nabla_Z \cdot (\dot{Z} B^* \, \delta f_a) = -\frac{1}{B^*}\nabla_Z \cdot (\dot{Z}_1 B^* \, f_{a0}) + \sum_b C^l(\delta f_a)$$

Where f_0 satisfies the neoclassical equilibrium:

$$\frac{\partial f_{a0}}{\partial t} + \frac{1}{B^*}\nabla_Z \cdot (\dot{Z} B^* \, f_{a0}) = \sum_b C\left[f_{a0}, f_{b0}\right]$$

$f_0 = f_{SM}$ for ions, $f_0 = f_{SM}$ or $(1 + e\delta\Phi/T_e) f_{SM}$ for kinetic electrons

(δf electrons) (δh split-weight)

$\dot{Z} \equiv \dot{Z}_0 + \dot{Z}_1 \Rightarrow$ full drift, equilibrium + fluctuations

where $\dot{Z}_1$ is the drift motion associated with fluctuations $e\delta\Phi$, $\delta\vec{A}_{||}$

Applying δf to GK PIC method

- PIC approach – solving marker particle distribution $F(Z)$

$$\frac{\partial F}{\partial t} + \frac{1}{B^*}\nabla_Z \cdot (\dot{Z} B^* F) = 0 \quad \text{and} \quad \delta f = \int w F dw$$

$$(1/B^*)\nabla_Z \cdot (\dot{Z} B^* F) \Rightarrow \dot{Z} \cdot \nabla_Z F \text{ with } Z = \{\psi, \theta, \phi, \mathrm{v}_{\|}, \mu\}$$

- Lagrangian equations in general flux coordinates for guiding center motion:

$$\frac{d}{dt}\left(\frac{\partial}{\partial \dot{x}_i} L\right) - \frac{\partial}{\partial x_i} L = 0$$

$$L(\mathrm{x}, \dot{\mathrm{x}}; t) = (A + \rho_{\|} B) \cdot \mathrm{v} - H$$

$$H = \rho_{\|}^2 B^2 / 2 + \mu B + \Phi \quad \text{(Littlejohn PF'81)}$$

- Weight equation:

$$\frac{dw}{dt} = \frac{1 - w}{f_0}\left[-\frac{1}{B^*}\nabla_Z \cdot (\dot{Z}_1 B^* f_{a0})\right]$$

Weight evolution

PPPL

Growing weight issue

- As the system evolves and turbulence develops, δf increases and so does the average weight of the particles
- A particle gaining a large weight will have a proportionally large contribution to physical quantities (density fluctuations, fields, temperature fluctuations, etc.)
- Eventually leads to what is called “noise” in PIC method
- It means that some regions of phase space would need to be represented by a larger number of marker particles
- Collisions take care of redistributing the weights between particles
- Many algorithms have been implemented in GK codes to reduce the growth further (Krook operator, remapping, etc.)

PPPL

Extended phase space

- Original idea of Chen and White (PoP'97) → extended phase space
  - Originally introduced to treat collisions in δf algorithm

$$F(\mathbf{x},\mathbf{v},t) \rightarrow F(\mathbf{x},\mathbf{v},t,w)$$

- Weight spreading reduction scheme by Brunner, Valeo, Krommes (PoP'99)

$$\frac{d}{dt}w(t) = -p_i(t)\frac{C[\delta f, f_M]}{f_M} - \eta[w_i - W(\vec{v}_i;t)]$$

- "η" arbitrary… In practice, periodic reassignment of the weights to their (local) average values

PPPL

Weight damping in GTS

- Use extended phase space idea while keeping formal phase space conservation
- Solve marker distribution in extended phase space $F(Z,w)$

$$\frac{\partial F}{\partial t}+\frac{1}{B^*}\nabla_Z\cdot(\dot{Z}B^*F)+\frac{\partial}{\partial w}(\dot{w}F)=0$$

- To ensure phase space incompressibility:

$$\frac{\partial}{\partial w}\dot{w}=0 \quad !!! \quad \text{leading to the following weight equation}$$

$$\frac{dw}{dt}=\frac{1-w}{f_0}\left[-\frac{1}{B^*}\nabla_Z\cdot(\dot{Z}_1B^*f_{a0})\right]+\frac{w-\langle w\rangle}{f_0}\left[-\frac{1}{B^*}\nabla_Z\cdot(\dot{Z}_1B^*f_{a0})\right]$$

- Calculate <w> by conserving particles, momentum and energy

PPPL

Anti-relaxation algorithm in GTS

Maintaining constant drive in global simulations

- Issue: by construction, "flux tube" (local) algorithm has a constant drive (no change in the temperature or density gradients)
- "Total" ($f_0+\delta f$) profiles in global codes change as δf increases and reaches steady state
- To allow for comparisons between flux tube results and global code results, most codes implement a so-called "profile restoration" method to maintain the temperature and density gradients constant during the simulation
- A lot of them are *ad hoc*, changing the weights of the particles to restore the original temperature profile
- GTS implements a more rigorous method…

PPPL

Anti-relaxation scheme in GTS

- Assume a relaxed Maxwellian distribution

$$f_0(T_0 + \langle \delta T \rangle) = f_0(T_0) + \delta f_0$$

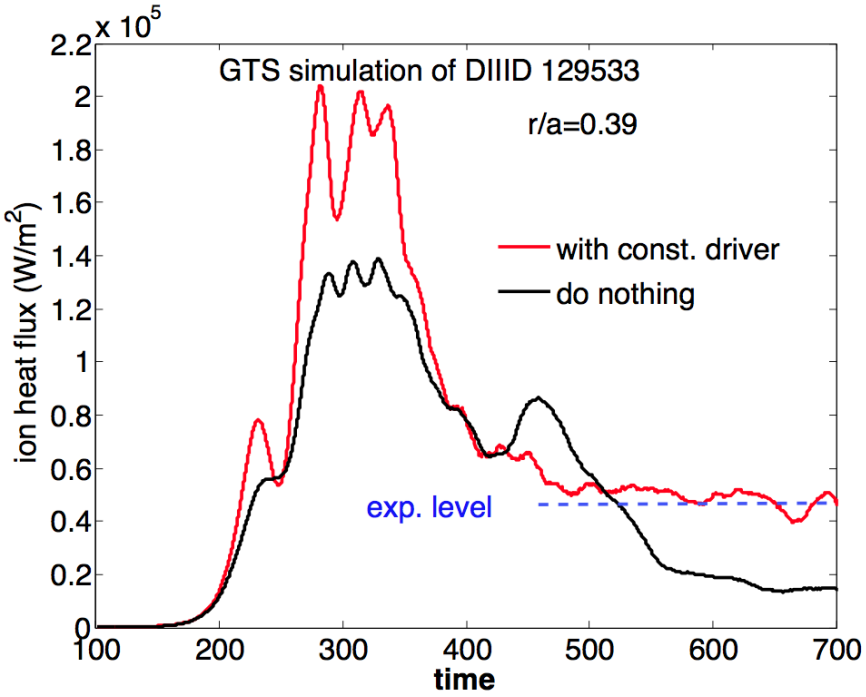
$$\delta f_0 \approx \left(\frac{m\mathrm{v}^2}{2T_0} - \frac{3}{2} \right) \frac{\langle \delta T \rangle}{T_0} f_0 \ , \quad \text{with} \quad \delta \mathrm{T} \equiv \frac{1}{3\mathrm{n}_0} \int d\mathrm{v} m\mathrm{v}^2 \delta f - \frac{T_0}{n_0} \int \delta f d\mathrm{v}$$

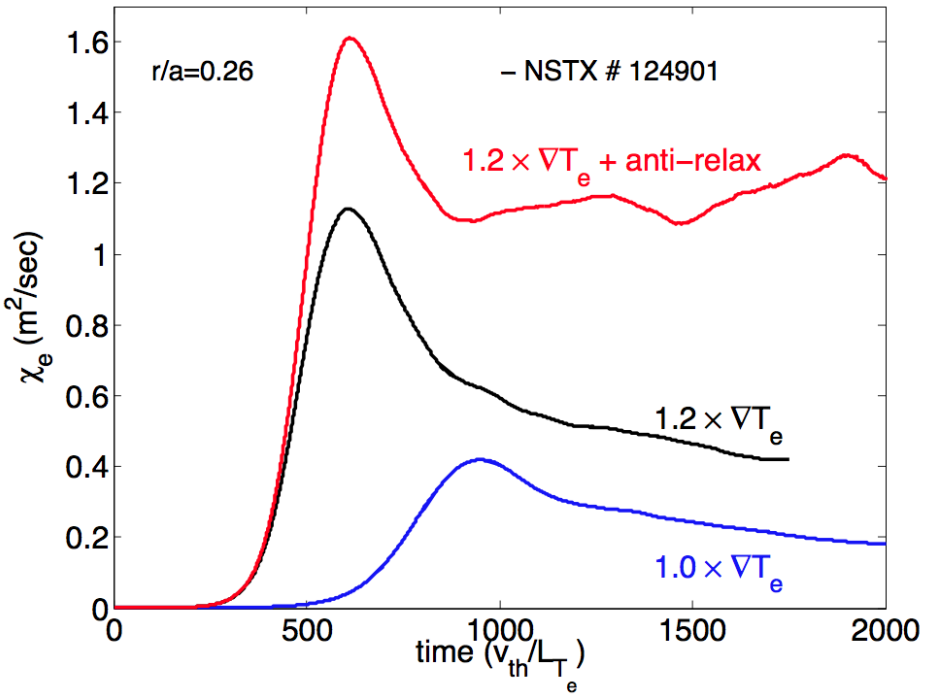
$$\text{which comes from} \quad \frac{3}{2}\delta p = \int d\mathrm{v} \frac{1}{2} m\mathrm{v}^2 \delta f = \frac{3}{2}\left(n_0 \delta T + T_0 \delta n \right)$$

- Remove relaxation from gradient driver

$$\frac{D\delta f}{Dt} = -\{ \mathrm{v}_E \cdot \nabla f_0 - \mathrm{v}_E \cdot \nabla \delta f_0 + \ldots \}$$

Testing the anti-relaxation scheme…

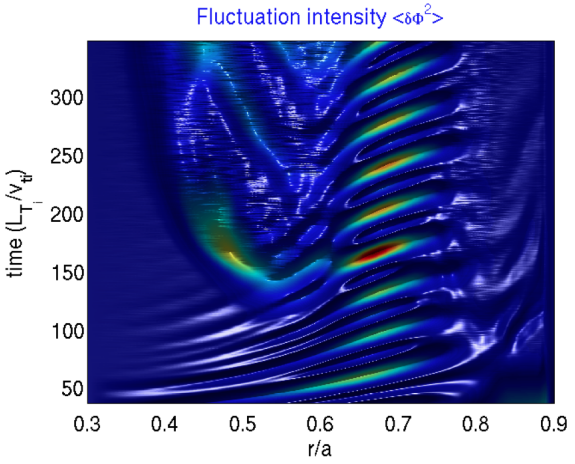


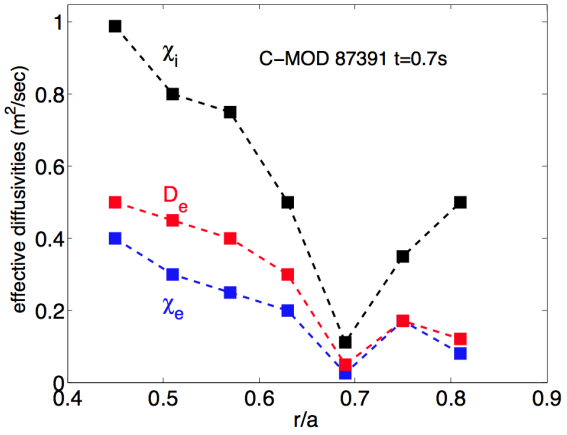


Seems to work…

Effect of including both turbulent and neoclassical physics self-consistently

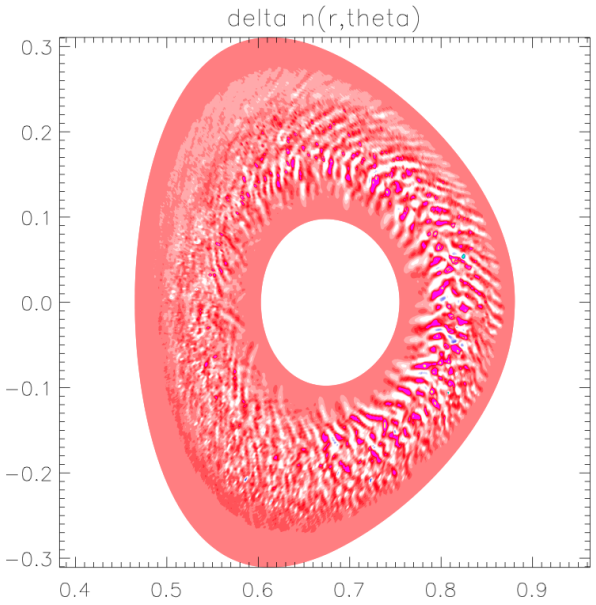
Dominant GAMs and impact in C-MOD L-mode phase

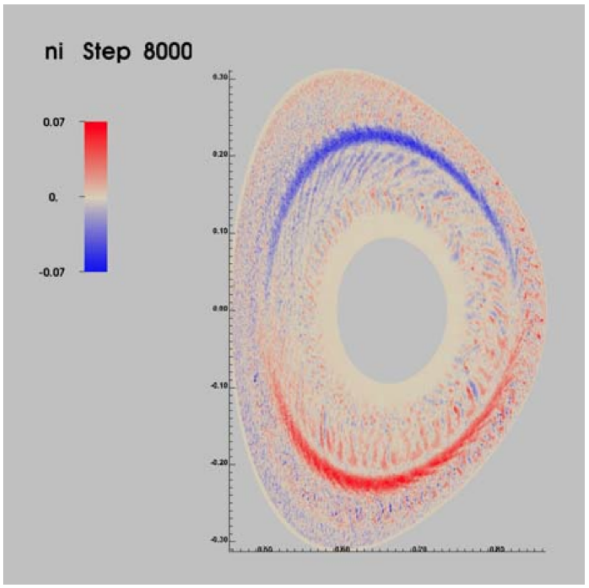




- GAMs may play important role over ZFs in edge as experiments suggest
- GAM layer decouples plasma inside and outside q=3/2 surface

W/O NEO TERM





WITH NEO TERM

Movie of CMOD simulation…