Upgrade to the GSP Gyrokinetic Code

Final Presentation

George Wilkie (gwilkie@umd.edu)

May 10, 2012

Supervisor: William Dorland (bdorland@umd.edu)

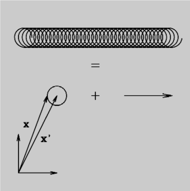
Outline

1. Background
   - Gyrokinetics
   - GSP Algorithm
2. Challenges
   - Collision Operator
   - Particle Trapping
   - Monte Carlo Integration
3. Validation and Testing
   - Phi Integral Validation
   - Z-pinch Entropy Mode
   - Parallel Scaling
4. Conclusion

Outline

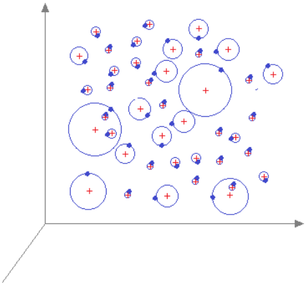
Particle Drift

- A charged particle in a magnetic field undergoes circular motion in the plane perpendicular to the magnetic field.
- A perpendicular force on a gyrating particle has the effect of causing a constant *drift velocity* perpendicular to *both* the magnetic field and the force in question.
- In general, $\mathbf{v}_{d,F} = \frac{c}{qB}\mathbf{F} \times \mathbf{B}$, but specifically:
  - "$\mathbf{E} \times \mathbf{B}$" drift: $\mathbf{v}_E = \frac{c}{B}\mathbf{E} \times \mathbf{B}$
  - Curvature drift: $\mathbf{v}_c = \frac{c}{qB}\frac{mv_\parallel^2}{R_c}\hat{\mathbf{r}} \times \mathbf{B}$
  - Grad-B drift: $\mathbf{v}_{\nabla B} = -\frac{c}{qB}\mu\nabla\mathbf{B} \times \mathbf{B}$



Gyrokinetics Review

- The dynamics of many charged particles are described by a probability distribution in phase space: $f(x, y, z, v_\perp, v_\parallel, \theta)$
- In the appropriate limit, the kinetic transport equation is expressed in "gyrocenter" coordinates **R**, averaging over gyroangle θ.



δf vs h

- For $f \approx F_0 + \delta f$, we can express the gyrokinetic equation in terms of either form of the perturbed distribution: $\langle \delta f \rangle_{\mathbf{R}}$ or $h = \langle \delta f \rangle_{\mathbf{R}} + \phi F_0$.
- Both versions of the code now exist.

Summary:

Function:	$\langle \delta f \rangle_{\mathbf{R}}$	h
RHS of GK equation contains:	$\langle E_z \rangle_{\mathbf{R}} = -\frac{\partial \langle \phi \rangle_{\mathbf{R}}}{\partial z}$	$\frac{\partial \langle \phi \rangle_{\mathbf{R}}}{\partial t}$
Complication:	Introduces a Courant stability condition due to z derivative.	Inaccurate due to delayed effect of time derivative.

The Gyrokinetic Equation - Form

$$\frac{D}{Dt}\langle \delta f \rangle_{\mathbf{R}} = \langle C[\delta f] \rangle_{\mathbf{R}} - \mathbf{v}_E \cdot \nabla F_0 - (\mathbf{v}_c + \mathbf{v}_{\nabla B}) \cdot \nabla \langle \phi \rangle_{\mathbf{R}} F_0 + v_{\|} \langle E_z \rangle_{\mathbf{R}} F_0$$

Where:

- F_0 is the equilibrium Gaussian velocity distribution, with possible gradients in n and T
- $C[\delta f]$ is the collision operator
- $\langle \rangle_{\mathbf{R}}$ signifies the *gyroaverage* at constant gyrocenter **R**:

$$\langle g(\mathbf{r}) \rangle_{\mathbf{R}} = \int d\theta \, g\left(\mathbf{R} + \boldsymbol{\rho}(\theta)\right)$$

The Gyrokinetic Equation - Method of Solution

$$\frac{D}{Dt}\left\langle \delta f \right\rangle_{\mathbf{R}} = \left\langle C[\delta f] \right\rangle_{\mathbf{R}} - \mathbf{v}_E \cdot \nabla F_0 - F_0 \left(\mathbf{v}_c + \mathbf{v}_{\nabla B} \right) \cdot \nabla \left\langle \phi \right\rangle_{\mathbf{R}} + v_{\parallel} \left\langle E_z \right\rangle_{\mathbf{R}} F_0$$

In the coordinates $E = \frac{1}{2}mv^2$ and $\mu = \frac{mv_\perp^2}{2B}$, we can solve via the method of characteristics with:

$$\frac{dz}{dt} = v_{\parallel} = \sqrt{\frac{2}{m}}\sqrt{E - \mu B_0}$$

$$\frac{d\mathbf{R}_\perp}{dt} = \mathbf{v}_E + \mathbf{v}_c + \mathbf{v}_{\nabla B}$$

$$\frac{dE}{dt} = \frac{d\mu}{dt} = 0$$

The objective of this project was to make this coordinate transformation in GSP and allow for a spatially-varying magnetic field $\mathbf{B_0}(z)$.

GSP Algorithm

1. **Initialize particles in phase space**
2. **Predictor step**
    - Calculate fields at step n
    - Calculate marker weights along characteristics for step $n + ^1/_2$
    - Advance marker positions for half timestep
    - Update weights with collision operator for step $n + ^1/_2$
3. **Corrector step**
    - Calculate fields at step $n + ^1/_2$
    - Calculate marker weights along characteristics for step $n + 1$
    - Advance marker positions for the full timestep
    - Update weights with collision operator for step $n + 1$
4. **Output results as necessary**

Repeat steps 2-4 N_t times

Outline

Collision Operator - Form

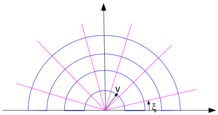
$$C\,[h] = L[h] + D[h] + U_{\|}[h] + U_{\perp}[h] + Q[h]$$

Where:

- L is a diffusion operator in *pitch angle* $\xi \equiv \cos^{-1}\frac{v_{\|}}{v}$
- D is a diffusion operator in speed v
- $U_{\|}$ and $U_{\perp}$ are integral operator designed to conserve momentum
- Q is an integral operator designed to conserve energy

There is a collision term for each species interacting with like-particles, and an additional term for electrons interacting with ions.

Collision Operator - Implementation



1. Using the updated positions and weights, distribute onto a 5D grid (x, y, z, E, ξ)
2. Fourier transform into k_x, k_y
3. Solve for diffusion in velocity space with a finite-difference implicit scheme
4. Calculate appropriate integrals over velocity grid to conserve momentum and energy
5. Inverse Fourier transform back
6. Reinterpolate weights back to particles' positions, appropriately weighted so that the effect of collisions is null if $\nu = 0$.

Collision Operator - Status

- All components are coded
- Diffusion operators appear well-behaved, but there is something wrong with the conservation terms.
- Conservation terms do not conserve momentum and energy, and instead introduce numerical instability!

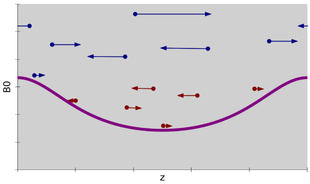
Outline

Particle Trapping

- A particle gyrating with magnetic moment $\mu \equiv \frac{mv_\perp^2}{2B_0}$ will experience a changing magnetic field as an effective potential with respect to its parallel motion:

$$U_{eff} = \mu B_0(z)$$

- If the particle does not have sufficient energy (i.e. if $E < U_{eff}$), then the particle is *forbidden* in a region with such a field.
  - The particle is said to be "trapped" in regions of lower magnetic field
- A gradient in the magnetic field results in an apparent force changing the particle's parallel velocity $v_\|$

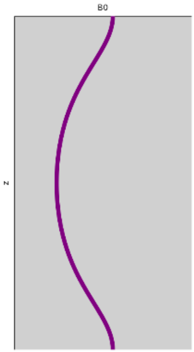


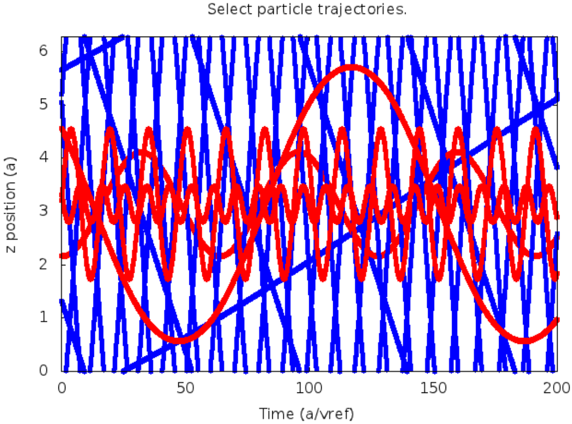
Bouncing Logic

- We can always find $v_{\|} = \pm\sqrt{E - \mu B(z)}$
- When advancing a particle's z position, check if it is predicted to travel beyond its turning point z^* and into the forbidden region. If so:

  1. Analytically solve for the particle's trajectory in a *linear* field $B(z) \approx B(z^*) + (z - z^*)B'(z^*)$
     - This relationship is exact for the current implementation in which the field is piecewise linear.
  2. Reverse the sign of $v_{\|}$

Particle Trajectories





Outline

Monte Carlo Integration

- In general, we can express any integral as an expectation value over some probability distribution p:

$$\int_a^b dx\, f(x) = \int_a^b dx \left(\frac{f(x)}{p(x)}\right) p(x) = \left\langle \frac{f(x)}{p(x)} \right\rangle_p$$

- Now, approximate the expectation value with a discrete set of markers *distributed according to* $p(x)$

$$\int_a^b dx\, f(x) \approx \frac{1}{N_p} \sum_i^{N_p} \frac{f(x_i)}{p(x_i)}$$

Monte Carlo Integration

- Source of error is stastistical. $\epsilon \sim \frac{\sigma}{\sqrt{N_p}}$ *indepdent of dimensionality*.
- Naive Monte Carlo with a uniform distribution function in one dimension is *worse than first order*!
- To improve accuracy, reduce σ by employing *importance sampling*: a strategic choice of $p(x)$
- Monte Carlo really shines in multiple dimensions, but unfortunately that's not what we're doing...

$$\phi \propto \int_0^{E_{max}/B} d\mu J_0\left(a\sqrt{\mu}\right) \int_\mu^{E_{max}} \frac{dEe^{-E}}{\sqrt{E-\mu B}} w(E,\mu)$$

$$\approx \sum_j^{N_\mu} (\Delta\mu)_j J_0(a\sqrt{\mu_j B}) I_j$$

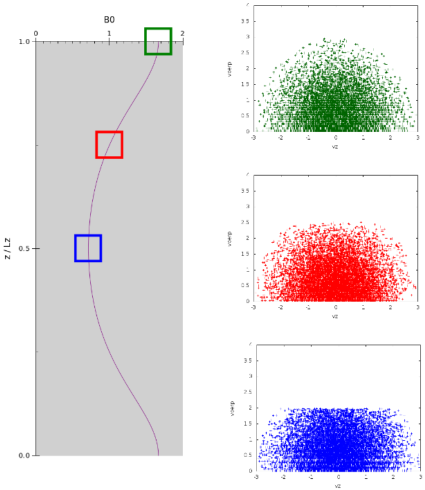
Importance Sampling

- At every grid point for every discrete μ_i, we evaluate the following 1D integral:

$$I_j = \int \frac{dE\, e^{-E}}{\sqrt{E - \mu B}} w(E, \mu_j) = \langle w \rangle_{p(E)} \approx \frac{1}{N_p} \sum_i^{N_p} w(E_i, \mu_j)$$

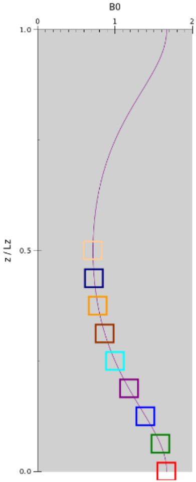
- Where the markers are distributed according to $p(E) = \frac{e^{-E}}{\sqrt{E - \mu_i B}}$
- **Important:** as the particles move around in space, this distrubtion must hold *everywhere and for all times*, else the method fails.
- There is hope. Despite its appearence, because the denominator is just the Jacobian for E,μ coordinates, it is indeed symmetric in E and thus is a Gaussian distribution in velocity space!

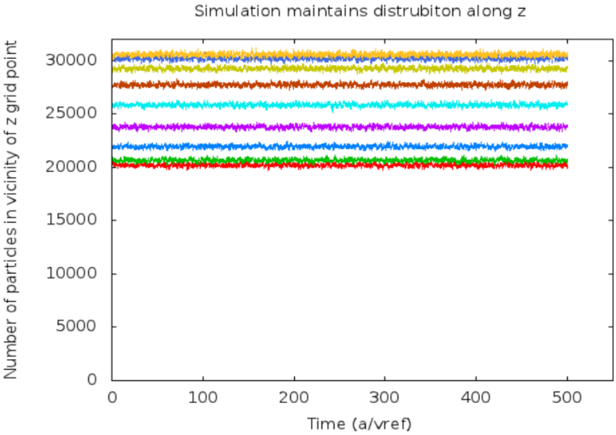
Velocity Space Distribution



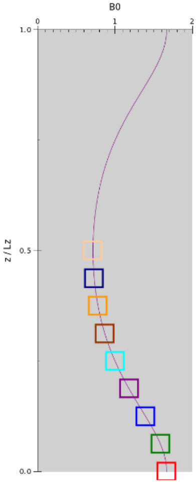
- Some regions of velocity space are forbidden depending on the local value of B_0
- Regions of lower magnetic field have *more* particles and vice versa
- Distribution is maintained throughout the simulation

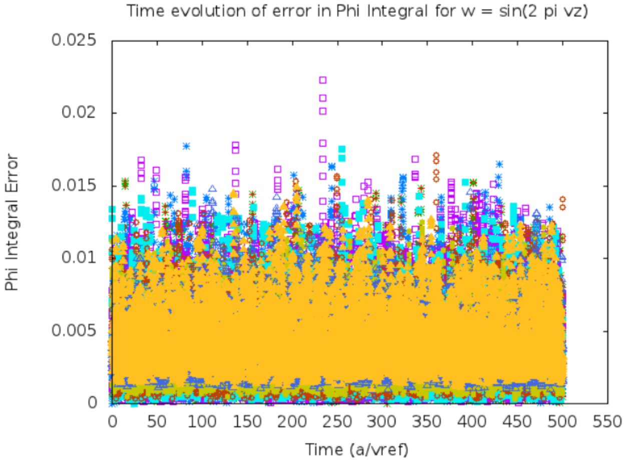
Velocity Space Distribution





Velocity Space Distribution





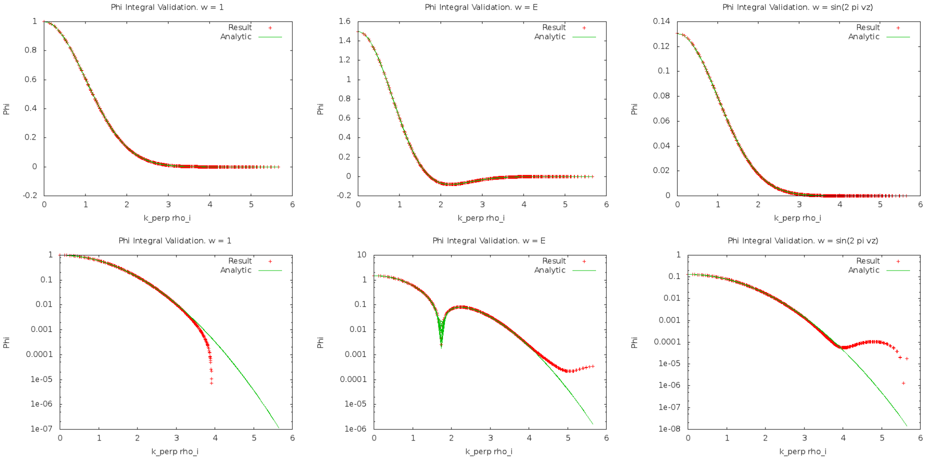
Outline

1. Background
   - Gyrokinetics
   - GSP Algorithm
2. Challenges
   - Collision Operator
   - Particle Trapping
   - Monte Carlo Integration
3. Validation and Testing
   - Phi Integral Validation
   - Z-pinch Entropy Mode
   - Parallel Scaling
4. Conclusion

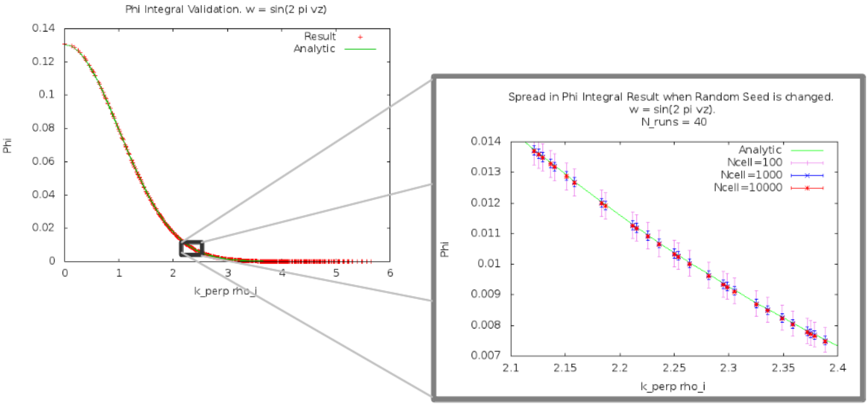
Phi Integral

- At the heart of the algorithm is the process of calculating the electrostatic potential ϕ.
- We wish to ensure this is being done properly with simple test cases.
- Substitute "dummy" functions for the weights such that we know the integral analytically:
  1. $w = 1 \rightarrow \phi \propto e^{-k_\perp^2/2}$
  2. $w = E \rightarrow \phi \propto \frac{1}{2}\left(3 - k_\perp^2\right) e^{-k_\perp^2/2}$
  3. $w = sin(2\pi\sqrt{E - \mu B}) \rightarrow \phi \propto D_+(\pi) e^{-k_\perp^2/2}$
- Perform this integration for each combination of k_x, k_y on the grid and compare against analytic results

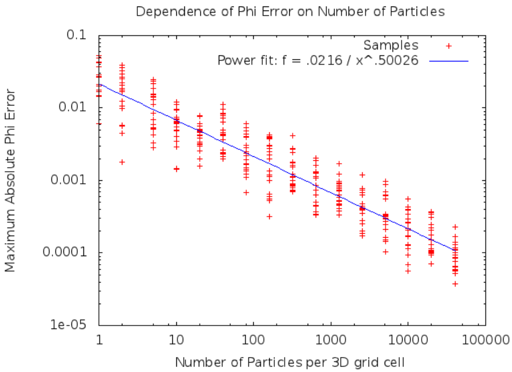
Analytic Phi Integral Results



Random Nature of Phi Error



Monte Carlo Error Analysis

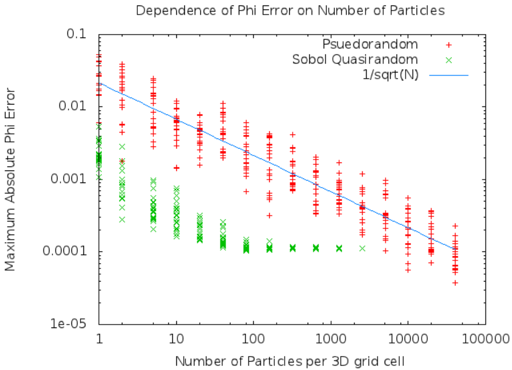


Error is as expected from Monte Carlo: $\propto \frac{1}{\sqrt{N}}$

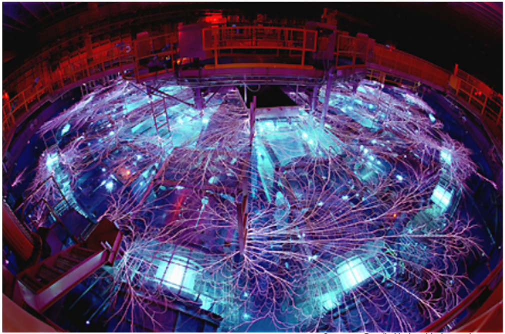
Pseudorandom vs Quasirandom

- But we can do better!
- Psuedorandom numbers and truly random numbers tend to over- or under-populate regions of the distribution unpredictably.
- If we use a *quasi-random sequence*, we can be guaranteed to fill out the distribution uniformly.

Monte Carlo Improvement



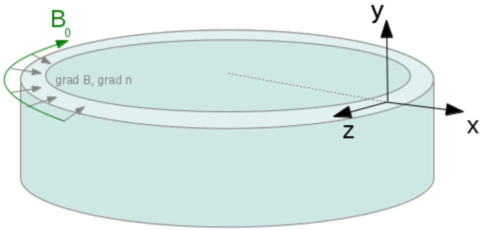
Z-pinch Entropy Mode



Courtesy, Sandia National Laboratories

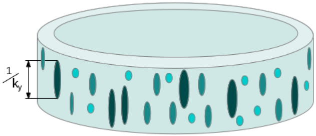
- The z-pinch confines a hot plasma in a collapsing ring of current.
- Limited by plasma instabilities due to radial density gradient.

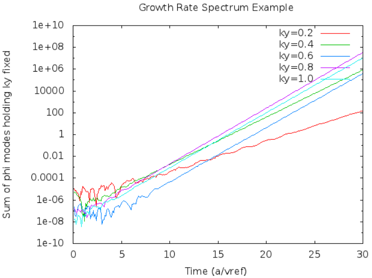
Z-pinch Geometry



- Gradients and curvature perpendicular to B_0 causes structure along the $\hat{\mathbf{y}}$-direction.
- Analyze the growth rate of the "Entropy Mode" instability for given radius R; gradient scale $\frac{1}{L_n} \equiv \frac{\nabla n}{n}$; temperature etc.

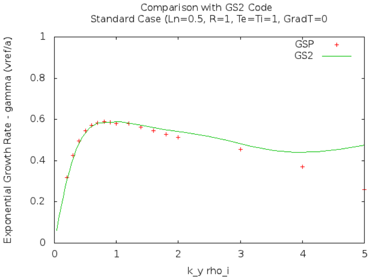
Entropy Mode Growth Spectrum

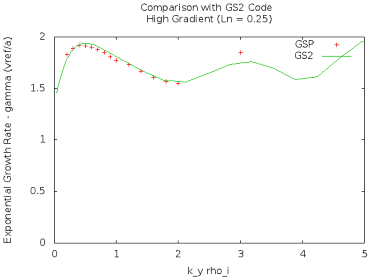




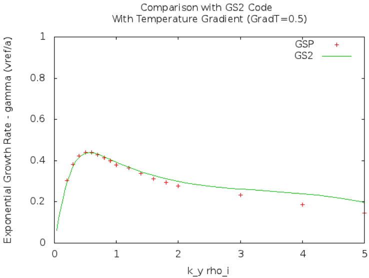
- Find the exponential growth rates of different Fourier modes
- Compare against a well-established Eulerian gyrokinetic code: GS2

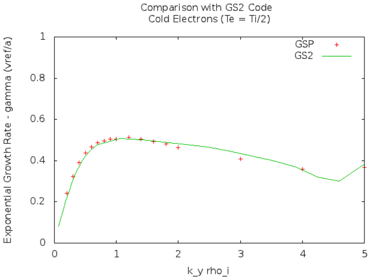
GS2 Growth Rate Spectrum Comparisons





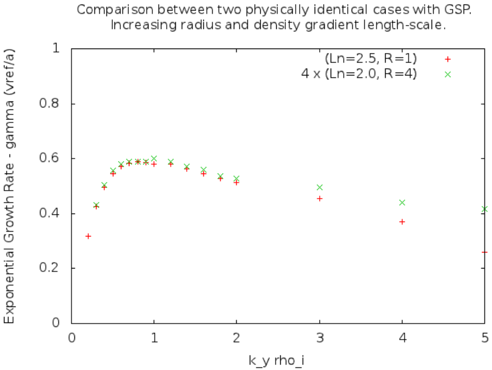
C





Equivalent Physics

- In the analytic dispersion relation, L_n and R only appear together as $\frac{R}{L_n}$, except for a single factor of $\frac{1}{L_n}$ overall.
- Increasing radius and length scale together should give the same physics, up to this factor.

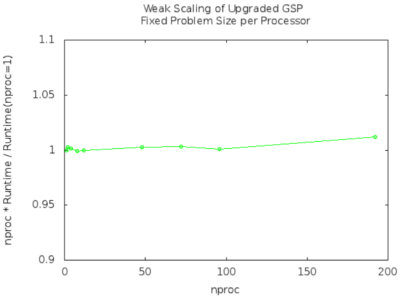


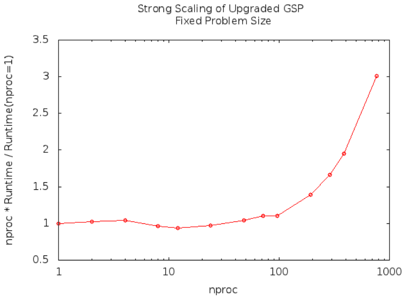
Parallelization Scheme

- Each processor as its own set of particles that it manipulates independently of others
- Each processor has a copy of the grid (which takes up much less memory than the particles), and independently computes grid-related quantities on the grid
  - Except for some operations in the collision operator (FFTs and tridiagonal matrix inversions): those are split among processors appropriately
- Processors only communicate when their respective contributions to the Monte Carlo integral are summed.
- As long as $N_{per\ cell}$ is sufficiently greater than N_{proc}, this should be advantageous.

Parallelization Results

$$N_{per\ cell} = 1600, N_{tot} \sim 400k$$





Outline

Progress

Goals

- Change coordinates used in GSP - **Complete**
- Code collision operator in new coordinates - **Complete**
- Validate accuracy of updated code - **Complete**
- Test parallel performance - **Complete**

Deliverables

A tarball to be sent via email by May 15, containing:

- GSP source code
- Makefiles with instructions for compiling and running
- Sample input file
- Data used in this presentation
- Final written report and presentation

Thank you!

Special thanks to:

- Prof. Bill Dorland for insight and encouragement
- Anjor Kanekar for the thoughtful discussions
- Dr. Ingmar Broemstrup for the carefully-designed base code
- Profs. Ide and Balan and the rest of AMSC 663 for the helpful questions and guidance.

Backup: Normalization and sampling from the $p(E)$ distribution

$$p(E) = A\frac{e^{-E}}{\sqrt{E-\mu B}}$$

Since $\int_{\mu B}^{E_{max}} p(E)dE = 1$ necessarily, normalize:

$$\frac{1}{A} = e^{-\mu B}\mathrm{Erf}\left(\sqrt{E_{max}-\mu B}\right)$$

Backup: Normalization and sampling from the $p(E)$ distribution

In order to obtain this distribution from a standard [0,1] uniform random distribution, we need to invert the cumulative distribution:

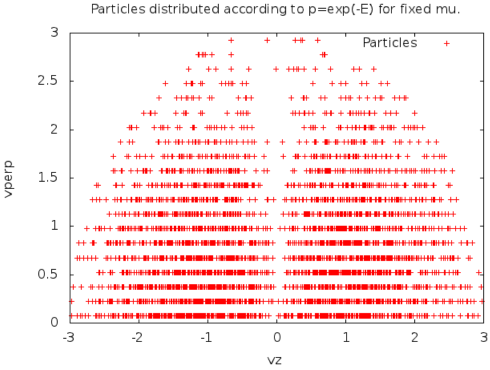
$$y = P(x) = A \int_{\mu B}^{x} \frac{dE e^{-E}}{\sqrt{E - \mu B}}$$

To obtain:

$$x = \mu B + \left(\mathrm{Erf}^{-1} \left[y \mathrm{Erf} \left(\sqrt{E_{max} - \mu B} \right) \right] \right)^2$$

Example of a Bad Distribution

$$p(E) = e^{-E}$$



Calculating the Fields

- In Fourier space, the gyroaveraging operation is multiplication by a Bessel function J_0
- The equation for the potential becomes:

$$\tilde{\phi}(\mathbf{k}) \propto \int d^3\mathbf{v} \left\langle \langle \delta f \rangle_{\mathbf{R}} \right\rangle_{\mathbf{r}} = \int d^3\mathbf{v} J_0 \left(\frac{k_\perp v_\perp}{\Omega} \right) \langle \delta f \rangle_{\mathbf{R}}$$

Change of Coordinates

- When the background field $\mathbf{B_0}$ is allowed to change, it makes sense to solve the gyrokinetic equation in E-μ coordinates so that one does not need to follow characteristics in velocity space
- The integral for the potential in these coordinates becomes:

$$\phi \propto \sum_i \int \int \frac{dEd\mu}{\sqrt{E - \mu B_0}} J_0 \left(k_\perp \sqrt{\frac{\mu}{B_0}} \right) e^{-E/2} w_i \delta(E - E_i) \delta(\mu - \mu_i)$$

- Particles are interpolated onto nearest $\frac{\mu}{B}$ so that after charge distribution is Fourier-transformed, the Bessel function can be computed
- Problem: Due to this Jacobian, the integration of E and μ cannot be performed separately

Collision Operator

- Accounts for interactions between particles
- Two parts:
  - **Pitch-angle scattering**: Diffusive. Calculated implicitly on a 5D grid
  - **Energy and momentum correction**: Integral operators.

Collision Operator

$$C[h] = \tfrac{\nu}{2}\frac{\partial}{\partial \xi}\left(1-\xi^2\right)\frac{\partial h}{\partial \xi} - \tfrac{\nu}{2} v^2 \frac{k_\perp^2}{\Omega^2} h + \nu F_0 \times$$

$$\left(2 v_\perp J_1\left(\tfrac{k_\perp v_\perp}{\Omega}\right) U_\perp[h] + 2 v_\| J_0\left(\tfrac{k_\perp v_\perp}{\Omega}\right) U_\|[h] \ldots + v^2 J_0\left(\tfrac{k_\perp v_\perp}{\Omega}\right) Q[h]\right)$$

- $h = \langle \delta f \rangle_{\mathbf{R}} + \frac{q\langle \phi \rangle_{\mathbf{R}}}{T} F_0$
- $\xi = \frac{v_\|}{v}$
- $\nu =$ Collision Frequency
- $U_\perp$, $U_\|$, and Q are moments of the distribution function; very similar integrals to the ϕ integral

ϕ Integral Checkout

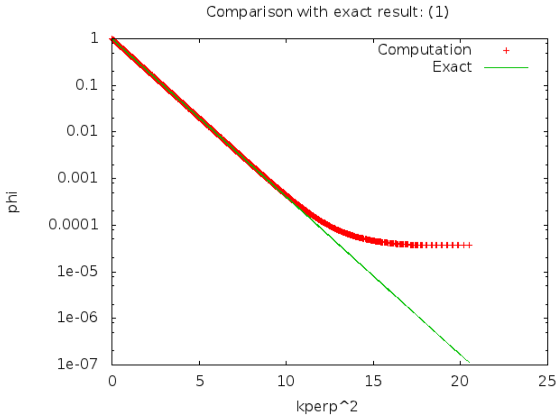
Polynomial	Error (RMS)
L_0	2.53×10^{-5}
L_1	2.85×10^{-5}
L_2	6.52×10^{-5}
L_3	1.45×10^{-4}
L_3	1.56×10^{-4}

Conditions:

- Particles per cell: 100
- Total particles: ~ 1.6 M
- Number of energy grid arcs: 160
- Maxmimum velocity: $6v_t$
- Normalized magnetic field: $0.8B_0$

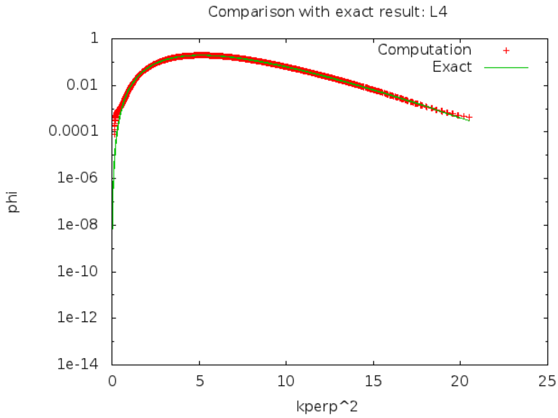
ϕ Integral Checkout

Estimation of: $\int d^3\mathbf{v} J_0\left(\frac{k_\perp v_\perp}{\Omega}\right) e^{-\frac{v_\perp^2}{2}}$ (RMS Error $= 2.53 \times 10^{-5}$)



ϕ Integral Checkout

Estimation of: $\int d^3\mathbf{v} J_0\left(\frac{k_\perp v_\perp}{\Omega}\right) e^{-\frac{v_\perp^2}{2}} L_4\left(\frac{v_\perp^2}{2}\right)$ (RMS Error $= 1.56 \times 10^{-4}$)



Backup: Gyroaveraging

Gyroaverage at constant gyrocenter $\mathbf{R}$

- $\langle \delta f \rangle_{\mathbf{R}}$
  - The gyroaveraged perturbation of the probability distribution function
  - The function that the gyrokinetic equation solves for
- $\langle \mathbf{E} \rangle_{\mathbf{R}}$
  - The average electric field that a marker with gyrocenter $\mathbf{R}$ "sees"
  - Determines the characteristic curves of a marker

Backup: Gyroaveraging

Gyroaverage at constant position $\mathbf{r}$

- $\langle\langle \delta f \rangle_{\mathbf{R}}\rangle_{\mathbf{r}}$
  - The charge deposited onto a position $\mathbf{r}$ from markers that have gyrocenter $\mathbf{R}$
  - Tricky concept
  - Used in Poisson's equation for the electrostatic potential

References

- Abel, et al. "Linearized model Fokker-Planck collision operators for gyrokinetic simulations. 1. Theory." *Physics of Plasmas*, **15**:122509 (2008)
- Antonsen and Lane, "Kinetic equations for low frequency instabilities in inhomogeneous plasmas." *Physics of Fluids*, **23**:1205 (1980)
- Aydemir, "A unified Monte Carlo interpretation of particle simulatinos and applications to non-neutral plasmas." *Physics of Plasmas*, **1**:822 (1994)
- Barnes, "Trinity: A Unified Treatment of Turbulence, Transport, and Heating in Magnetized Plasmas." PhD Thesis, University of Maryland Department of Physics (2009)
- Broemstrup, "Advanced Lagrangian Simulations Algorithms for Magnetized Plasma Turbulence." PhD Thesis, University of Maryland Department of Physics (2008)

- Dimits, et al, "Comparisons and physics basis of tokamak transport models and turbulence simulations." *Physics of Plasmas*, **7**: 969 (2000)
- Frieman and Chen, "Nonlinear gyrokinetic equations for low-frequency electromagnetic waves in general plasma equilibria." *Physics of Fluids*, **25**:502 (1982)
- Howes, et al. "Astrophysical gyrokinetics: basic equations and linear theory." *The Astrophysical Journal*, **651**: 590 (2006)
- Ricci, et al, "Gyrokinetic linear theory of the entropy mode in a Z pinch." *Physics of Plasmas*, **13**: 062102 (2006)