Mid Year Progress Report on Upgrade to the GSP Gyrokinetic Code

George Wilkie (*gwilkie@umd.edu*)
Supervisor: William Dorland (*bdorland@umd.edu*)

December 18, 2011

Abstract

Simulations of turbulent plasma in a strong magnetic field can take advantage of the gyrokinetic approximation, the result of which is a closed set of equations that can be solved numerically. An existing code, GSP, uses a novel and highly efficient solution method to solve the nonlinear 5D gyrokinetic equation. In this project, we will seek to change the velocity-space representation in GSP. This transformation will simplify the inclusion of a new collision operator and make the algorithm more suitable for simulations of turbulence in Tokamak plasmas, while retaining the efficiency and accuracy of the original code.

1 Background

Plasma physics concerns the collective dynamics of collections of many charged particles interacting self-consistently with their respective electromagnetic fields. In a uniform magnetic field, the motion of a charged particle (we are usually referring to ions unless otherwise specified) in the plane perpendicular to the magnetic field is a circle whose radius (the gyroradius ρ) depends on its velocity in that plane:

$$\rho = \frac{v_\perp}{\Omega}$$

($\Omega \equiv \frac{eB}{mc}$ is the *gyrofrequency*). If this radius is much smaller than the characteristic size of the plasma, we can use that ratio as an expansion parameter. Gyrokinetics specifically is the regime in which turbulent fluctuations *can be* of a size on the order of this gyroradius. This is in contrast to drift-kinetic theory (and by extension, MHD), where the gyroradius needs to be much smaller than the size of the fluctuations. In order to get a closed theory in gyrokinetics, we demand that the drift velocity be small compared to the thermal velocity $v_t \equiv \sqrt{\frac{2k_BT}{m}}$. Such drifts come about when there is curvature or a gradient in the magnetic field or if there exists another force field (such as the electric field).

We can summarize the gyrokinetic assumptions thusly:

$$\frac{\rho}{L} \sim \frac{v_E}{v_t} \sim \frac{\omega}{\Omega} \sim \frac{\nu}{\Omega} \sim \epsilon \ll 1$$

Here, $v_E = \frac{cE}{B_0}$ is the $E \times B$ drift speed, ω is the frequency of turbulent fluctuations, and ν is the characteristic collision frequency. These assumptions are well justified in a wide range of plasmas found in magnetic fusion confinement devices and in astrophysical plasmas.

The gyrokinetic equation results from expanding the distribution function in the small parameter ϵ:

$$f(\mathbf{r}, \mathbf{v}, t) = F_0 + \delta f + \ldots$$

Where $\delta f \sim \epsilon F_0$ represents the fluctuations in the distribution function. If we transform coordinates to the center of the gyromotion $\mathbf{R}$, we can write

$$\delta f = -\frac{q\phi}{T} F_0 + h(\mathbf{R}, v_\parallel, v_\perp)$$

where ϕ is the electrostatic potential. The latter function h represents the part of δf that does *not* depend on the phase of the gyration. Note that this is not true for the first term because ϕ is a function of position, so it varies as the gyrophase changes. The background distribution function F_0 is given, usually a Maxwellian velocity distribution with temperature T and number density n, which of which can take gradients.

2 Approach

GSP is a Particle-In-Cell algorithm to solve the gyrokinetic equation coupled with Maxwell's equations for the fields generated by the particles' configuration. The gyrokinetic equation is:

$$\frac{\partial h}{\partial t} + v_\parallel \hat{\mathbf{z}} \cdot \frac{\partial h}{\partial z} - \frac{c}{B_0} \left(\hat{\mathbf{z}} \times \langle \mathbf{E} \rangle \right) \cdot \nabla h = \langle C \left[h \right] \rangle + \frac{c}{B_0} \left(\hat{\mathbf{z}} \times \langle \mathbf{E} \rangle \right) \cdot \nabla F_0 + \frac{q}{T} F_0 \frac{\partial \langle \phi \rangle}{\partial t}$$

Where $\hat{\mathbf{z}}$ is directed along the equilibrium magnetic field $\mathbf{B_0}$, F_0 is the equilibrium distribution function, and $\mathbf{E}$ is the electric field (with electrostatic potential ϕ). The angle brackets $\langle\rangle$ signify the operation of gyroaveraging at constant $\mathbf{R}$ - the gyrocenter coordinate. So that we don't have to keep track of the time derivative of ϕ, we recast the gyrokinetic equation in terms of $g \equiv \langle \delta f \rangle_{\mathbf{R}} \equiv -\frac{q\langle\phi\rangle}{T} F_0 + h$:

$$\frac{\partial g}{\partial t} + v_\parallel \hat{\mathbf{z}} \cdot \frac{\partial g}{\partial z} - \frac{c}{B_0} \left(\hat{\mathbf{z}} \times \langle \mathbf{E} \rangle \right) \cdot \nabla g = \langle C \left[g \right] \rangle + \frac{c}{B_0} \left(\hat{\mathbf{z}} \times \langle \mathbf{E} \rangle_{\mathbf{R}} \right) \cdot \nabla F_0 + v_\parallel \langle E_z \rangle F_0$$

The distribution function $g = g\left(\mathbf{R}, v_\parallel, v_\perp\right)$ is the gyroaverage of δf: the small perturbation to the full distribution function $f = F_0 + \delta f + \ldots$, of which h

is a part. We can obtain useful information about the flow (number density and temperature perturbations) by taking integrals over velocity space of this distribution. For example, the fluctuation in number density is:

$$\delta n_i(\mathbf{r}) = \int d^3\mathbf{v}\delta f(\mathbf{r}, \mathbf{v})$$

The gyrokinetic equation is solved by following Lagrangian markers (distrubted according to a monte carlo scheme, see [Aydemir, 1994]) along characteristic curves in 5D phase space. The characteristics for a uniform $\mathbf{B_0}$ are defined by:

$$\begin{aligned}
\frac{d\mathbf{R}}{dt} &= v_\parallel \hat{\mathbf{z}} + \frac{c}{B_0}\hat{\mathbf{z}} \times \nabla \langle\phi\rangle \\
\frac{dv_\parallel}{dt} &= 0 \\
\frac{dv_\perp}{dt} &= 0
\end{aligned}$$

So each marker is assigned a $v_\parallel$-$v_\perp$ pair and its trajectory in $\mathbf{R}$-space is advanced along its characteristic curve according to the drift velocity above. Then, the distribution function represented by each particle is updated along characteristics. We use the function $w \equiv \frac{g}{F_0}$ to represent this distribution function in GSP (the function F_0 is known, so this is just a change of representation). The equation along characteristics is:

$$\dot{w} = \frac{1}{F_0}\frac{dg}{dt} = \frac{1}{F_0}\langle C[g]\rangle + \frac{c}{B_0}(\hat{\mathbf{z}} \times \langle\mathbf{E}\rangle) \cdot \frac{\nabla F_0}{F_0} + v_\parallel \langle E_z\rangle$$

In order to calculate each the markers' drift velocities and the updated "weights" w, the gyroaveraged electric field $\langle\mathbf{E}\rangle \equiv -\nabla\langle\phi\rangle$ and collision operator $C[g]$ must be known. These calculations are described in more detail in later sections.

To summarize, each marker in GSP has the following values associated with each of them:

- *Gyrocenter position*, $\mathbf{R}$: Evolves in time along characteristics defined above.
- *Velocity Coordinates*, $v_\parallel$ *and* $v_\perp$ Fixed for a given particle. Together with position, it defines the part of phase space of which the marker is representative.
- *Distribution Weight*, $w \equiv \frac{g}{F_0}$: Evolves in time according to the equation along characteristics given above. It's the representation of the distrubion function being solved for.

The algorithm implemented in GSP is illustrated below. The time-marching scheme is a predictor-corrector method where the locations of the particles are used to calculate the fields on a grid, which in turn affect the evolution of the particles and their "weight functions" w_i.

1. Initialization
    - Place particles as sampled from the equilibrium distribution function
2. Simulation
    - For each particle i:
        - $\langle\phi\rangle^n$ calculated on a grid in Fourier space, using all the $\mathbf{R}_j^n$
        - $\mathbf{R}_i^{n+\frac{1}{2}} = \mathbf{R}_i^n + \frac{\Delta t}{2}\left(\frac{d\mathbf{R}}{dt}\right)_i^n$
        - Contribution from collision operator $\langle C[g]\rangle$ calculated for a half timestep
        - $w_i^{n+\frac{1}{2}} = w_i^n + \frac{\Delta t}{2}\dot{w}_i$
        - $\langle\phi\rangle^{n+\frac{1}{2}}$ calculated on a grid in Fourier space, using all the $\mathbf{R}_j^{n+\frac{1}{2}}$
        - $\mathbf{R}_i^{n+1} = \mathbf{R}_i^n + \Delta t\left(\frac{d\mathbf{R}}{dt}\right)_i^{n+\frac{1}{2}}$
        - Contribution from collision operator $\langle C[g]\rangle$ calculated for a full timestep
        - $w_i^{n+1} = w_i^n + \Delta t\dot{w}_i$
3. Repeat Step 2 as needed
4. Obtain moments of $\langle\delta f\rangle$ such as density, temperature, and flux

3 Implementation

The above algorithm is valid only when the macroscoping background magnetic field $\mathbf{B_0}$ is uniform. When it's permitted to vary, each particles' $v_\parallel$ and $v_\perp$ will change. We choose instead to represent velocity space using constants of motion as our coordinates. Suitable choices are energy $\mathcal{E} \equiv v^2$ and magnetic moment $\mu \equiv \frac{v_\perp^2}{B_0}$, each of which are constants of motion to the order at which we are concerned. Therefore, $\frac{d\mathcal{E}}{dt} = \frac{d\mu}{dt} = 0$, and the markers can each be assigned their own velocity coordaintes which do not change along characteristics. This project consists of making the aforementioned coordinate change while attempting to preserve the efficiency and parallelism of the original code.

The most challenging aspects of the project, both in the sense of computational intensity and difficulty of understanding are:

- The calculation of $\langle\phi\rangle$
- The calcualtion of the collision operator $\langle C\left[g\right]\rangle$

The philosphy and status of each of these aspects will be discussed in the following.

Calculating the Fields

We wish to find the gyroaveraged electric field $\langle \mathbf{E} \rangle \equiv -\nabla \langle \phi \rangle$. We can calculate ϕ using the quasineutrality condition: $\delta n_i = \delta n_e$ to leading order. These number density fluctations of the ions and electrons respectively are moments of the distribution function. The equation for $\phi(\mathbf{r})$ ignoring the normalization constant becomes:

$$\phi \propto \int_{-\infty}^{\infty} dv_{\parallel} \int_{0}^{\infty} dv_{\perp} v_{\perp} g\left(v_{\parallel}, v_{\perp}\right) J_0\left(\frac{k_{\perp} v_{\perp}}{\Omega}\right)$$

Note that the ϕ being calculated here is *not* gyro-averaged. This integral is performed in Fourier space, which simplifies the gyro-averaging operation. For instance, consider some function $A(\mathbf{r})$:

$$\left\langle A(\mathbf{r}) e^{i\mathbf{k}\cdot\mathbf{r}} \right\rangle = A(\mathbf{R}) J_0\left(\frac{k_{\perp} v_{\perp}}{\Omega}\right) e^{i\mathbf{k}\cdot\mathbf{R}}$$

That is, in Fourier space, the gyro-averaging operation consists of simply multiplying by a Bessel function. Furthermore, in keeping with Aydemir's monte carlo analysis, our perturbed ddistribution g is sampled from the background distribution F_0. So what is saved for each marker is a "weight" $w_i(v_{\parallel}, v_{\perp})$, which was defined above. This weight contains information about g for a marker with a particular $v_{\parallel}$ and $v_{\perp}$. The many particles in a vicinity when combined give our best monte carlo estimate for $g(\mathbf{R}, v_{\parallel}, v_{\perp})$.

Rewriting the integral in terms of w and using the Maxwell distribution for F_0:

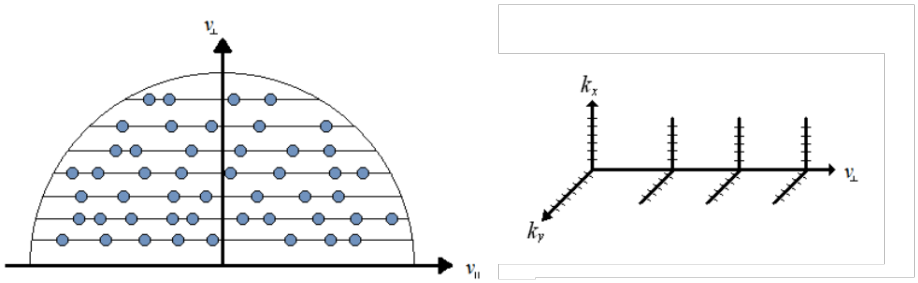
$$\phi \propto \int_{-\infty}^{\infty} dv_{\parallel} e^{-\frac{v_{\parallel}^2}{v_t^2}} \int_{0}^{\infty} dv_{\perp} v_{\perp} e^{-\frac{v_{\perp}^2}{v_t^2}} w\left(v_{\parallel}, v_{\perp}\right) J_0\left(\frac{k_{\perp} v_{\perp}}{\Omega}\right)$$

Since w is just a sum of delta functions (each of which is represented by a Lagrangian marker), to perform this integral we simply add each markers' contribution $we^{-\frac{v_{\parallel}^2 + v_{\perp}^2}{T}} v_{\perp} J_0\left(\frac{k_{\perp} v_{\perp}}{\Omega}\right)$ as deposited onto a grid.

After transforming to $\mathcal{E}$-μ coordinates, and using the definition of Ω, this integral becomes:

$$\phi \propto \int_{0}^{\infty} d\mathcal{E} e^{-\frac{\mathcal{E}}{v_t^2}} \int_{0}^{\frac{\mathcal{E}}{B_0}} \frac{1}{\sqrt{\mathcal{E} - \mu B_0}} J_0\left(\frac{mck_{\perp}}{e}\sqrt{\frac{\mu}{B_0}}\right) w\left(\mathcal{E}, \mu\right)$$

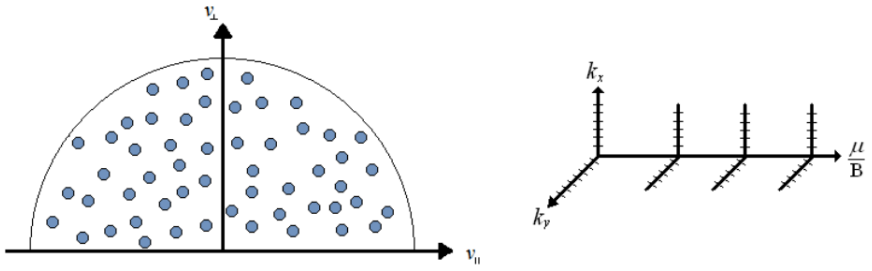
The challenge of this task is the repeated calculation of J_0, which involves a *cos* and a *sqrt* function for each call (using the approximation found in Abromowitz and Stegun). This would be an enormous computational burden. Instead, the current incarnation of GSP forces the particles to take only one of several discrete values of $v_{\perp}$, as illustrated in the figure below:



Therefore, after the markers' charges have been interpolated onto a grid and Fourier transformed, our phase space consists of a grid in four dimensions (three for $\mathbf{k}$ and one for $v_\perp$), with $v_\parallel$ allowed to take any random value for a given $\mathbf{k}$ and $v_\perp$.

Ahead of time, we can populate an array that contains all the discrete values of $J_0\left(\frac{k_\perp v_\perp}{\Omega}\right) = J_0\left(k_x, k_y, v_\perp\right)$ that are necessary. The code then just refers to the correct index of this array as needed when we sum over markers. Since in this case, B_0 is constant, the particles will not drift off of their assigned discrete value of $v_\perp$.

The method needs revision in the upgraded version when B_0, and by consequence $v_\perp$, are allowed to change. Each are arguments to J_0, so a more creative solution is necessary. Note that when the coordinate transformation to μ is performed, the argument of J_0 becomes proportional to $k_\perp\sqrt{\frac{\mu}{B_0}}$. Then, if the J_0 array is re-mapped so that it is defined instead for discrete values of $\sqrt{\frac{\mu}{B_0}}$, then we can perform the integral similarly (see figure below)



Since this parameter $\sqrt{\frac{\mu}{B_0}}$ itself changes for a given marker throughout the simulation, we will have to perform an interpolation onto the nearest discrete value. This method should have no impact on the accuracy of ϕ, although there is an additional variable on which to interpolate (in addition to r_x, r_y, r_z) which may have an impact on performance. Furthermore, we retain the freedom to choose whatever grid we like in velocity space (we could, for example discretize $\mathbf{E}$) should the situation call for it.

The difference that arises is that, while previously the particles were bound to a particular few values of $v_\perp$, in the update they will be free to wander on

this "grid", but will need to be interpolated onto the nearest value of the Bessel function argument $\frac{\mu}{B_0}$ instead of permanently residing at that value.

Once ϕ is calculated through the above integration, it is a simple matter to find $\langle \mathbf{E} \rangle = -i\mathbf{k}J_0\left(\frac{k_\perp v_\perp}{\Omega}\right)\phi$, which is the ultimate goal. This is reverse-Fourier transformed back onto **r**-space and re-interpolated from the grid back onto the actual positions of the markers.

The changes mentioned here have already been coded. Soon rudimentary validation will be performed to ensure that the results obtained are consistent with the original GSP code.

Collision Operator

The collision operator is in general an integro-differential operator on the distribution function that accounts for immediate interactions between the particles. On one extreme is the Fokker-Planck/Landau operator that is rigorously describes the physics of a Coulomb interaction, but is computationally expensive. On the other is the Krook collision operator, which is a simple decay based on a parameter ν. This operator has no rigorous physical basis, nor does is conserve bulk physical quantities as desired. As a compromise, several different collision operators are proposed that are not rigorously derived, but faithfully maintains various moments of the distribution function. That is: particle number (the zeroth moment), momentum (the first moment), and energy (the second moment) are all conserved, and the Boltzmann H-theorem (a statement of the Second Law of Thermodynamics) is obeyed. The two relevant operators are:

- **Pitch-Angle Scattering**: A diffusion operator that maintains a particle's energy $\mathcal{E}$, diffusing markers along "arcs" in velocity space of constants energy. This requires an integral correction term in order to conserve momentum.

- **Abel**: As described in [Abel et al, 2008], this operator (and its discrete form) have all the important physical properties, including the elusive H-theorem.

To calculate the collision term in the gyrokinetic equation, we essentially recreate the function g in 5D phase space, operate on it, then return its value. Interpolating onto a phase space grid involves "coarse-graining", which is a way to use artificial diffusion to simulate the effect of a collision operator. This is not the approach we take, so we must be careful with how the function is returned. That is, even if the collision frequency (the parameter that sets the strength of collisions) is set to zero, the simple act of scattering and gathering the function between the particles and the grid will result in unwanted diffusion (diffision which should not exist if there are no collisions). Therefore, the original function at a marker position is updated *by the same ratio* that the functions values at the neighboring grids are changed.

For this project, the numerical implementation of the collision operator will be editted so that it is performed in the new coordinates.

4 Databases

- GSP Source code:
  http://gyrokinetics.svn.sourceforge.net/viewvc/gyrokinetics/gsp/
- Article with analytic result:
  - Ricci, et al, "Gyrokinetic linear theory of the entropy mode in a Z pinch." *Physics of Plasmas*, **13**: 062102
- Article with computational benchmark:
  - Dimits, et al, "Comparisons and physics basis of tokamak transport models and turbulence simulations." *Physics of Plasmas*, **7**: 969

5 Validation

To ensure we have built a code that gives physical results, we will compare results obtained from both the old version of GSP and the update. Both should agree remarkably well in regimes in which the original GSP is accurate since a coordinate change shall not change the physics being simulated.

A standard computational benchmark for gyrokinetics is given in [Dimits et al, 2000]. We will use this reference to compare results obtained from GSP.

Furthermore, an analytical condition necessary for stability in a gyrokinetic plasma has been obtained in Appendix B of [Ricci et al, 2006]. Comparing this to the conditions at which the code predicts an unstable plasma would provide a good test of the physical fidelity of the code. Application of this test would require a translation in position coordinates as well, which is beyond the immediate scope of this project, although it is hoped to be accomplished.

6 Testing

Even if we have built a code with robust physics, it is possible that such an effort may be in vain. This would occur if the "updated" version of GSP is not an update at all and instead is considerably less efficient than the original code. Therefore, we shall benchmark the performance of the update against the original code and ensure that it scales well from a parallelization standpoint.

7 Schedule and Milestones

Phase I: Analytics
September - October 2011

- ✓ Understand rigorous derivation of the gyrokinetic equation
- ✓ Become familiar with analytical aspects of the gyrokinetic equation
- ✓ Make velocity coordinate change
- ✓ Understand algorithm used by GSP to solve GK equation
- ✓ Understand Abel collision operator and how it is to be implemented in GSP
- ✓ **Milestone:** Derive GK equation in velocity coordinates E and μ.
- ✓ **Milestone:** Develop an efficient method to handle the integration for ϕ

Phase II: Numerics
November 2011 - January 2012

- ✓ Become familiar with GSP code
- Make proposed changes to the code *(in progress)*
- Ensure code still runs
- **Milestone:** GSP Code updated with new velocity coordinates.

Phase III: Testing and Validation
February - April 2012

- Debug updated GSP code
- Validate against previous version of code iteratively
- Run test cases, organize results
- **Milestone:** GSP code in new velocity coordinates validated and tested

Phase IV: Communication
April - May 2012

- Prepare final presentation
- **Milestone:** Final presentation given

The schedule dates have been retroactively changed to reflect past scheduling realities. Phase II began behind schedule: beginning in November instead of October as initially planned. Another way the schedule (and indeed my vision of the project overall) has changed is the significance of handling the integration necessary to calculate ϕ as explained in section 3. This has also been retroactively added this as a Phase I milestone to reflect its importance.

Phase III is expected to begin in earnest in mid-February. Some Phase III activities, espcially validation against original code, is expected to be performed well before then.

I feel I have gained sufficient understanding of the algorithm and a promising method has been chosen to calculate ϕ efficiently. These were identified as the largest risks in the intitial proposal, and they have been significantly mitigated. The worst case scenario anticipated is J_0 will need to be calculated "on the fly " as needed. This will result in a code that runs and performs the relevant calculations, but is very inefficient.

8 Deliverables

- Updated GSP source code *(in progress)*
- Sample input files of test cases
- Test case comparison data
- ✓ Mid-year progress report
- Final presentation

9 Bibliography

- 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)
- Catto, "Linearized gyro-kinetics." *Plasma Physics*, **20**:719 (1978)
- 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)