An Explicit Discontinuous Galerkin Solver for GS2

P. J. Knight, C. M. Roach, W. Arter

EURATOM/CCFE Fusion Association, Culham Science Centre, Abingdon, OX14 3DB, UK

November 2012

Abstract

We present a fully-explicit solution method for the GS2 gyrokinetic code. Presently, GS2 uses an implicit algorithm for the evolution of linear terms, which is numerically stable for large time-steps in linear simulations. However, initialising the implicit response matrices is an expensive operation in computational terms, and in certain cases needs to be repeatedly performed throughout the course of the run. It was therefore considered that a move to create a fully explicit solution method would be beneficial.

The new method is a finite-element Discontinuous Galerkin (DG) scheme, using an adaptive explicit Runge-Kutta time-stepping solver to advance the solution accurately in an efficient manner.

1 Introduction and Motivation

GS2 is a time-dependent gyrokinetic code used widely for the simulation of plasma microinstabilities and turbulence. It uses the gyrokinetic equation to evolve the 5-D phase space perturbed density distribution function g_s:

$$\frac{\partial g_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v}_d).\nabla g_s + C_s\, g_s = -q_s \frac{\partial f_{0s}}{\partial E}\frac{\partial \chi}{\partial t} + \{\chi, f_{0s}\}$$

Presently, GS2 uses an implicit algorithm for the evolution of linear terms [1], which is numerically stable for large time-steps in linear simulations. However, initialising the implicit response matrices has a computational cost that scales strongly with grid size along the field line direction. Such a cost is expensive if the simulation is aimed at resolving disparate scales, such as

- microtearing modes
- multi-scales with both ion temperature gradient (ITG) and electron temperature gradient (ETG) modes present
- for equilibria with low finite magnetic shear.

The nonlinear term is handled pseudo-spectrally in space (perpendicular to the field lines) with an explicit second order Adams-Bashforth time stepping scheme; thus, in nonlinear simulations the time-step is limited by the CFL condition. Since changing the time-step requires the code to perform the expensive re-initialisations of the large response matrix, it was considered that a move to create a fully explicit solution method would be beneficial. Our new explicit algorithm complements the implicit algorithm, and should facilitate larger grid simulations by reducing the large initialisation overhead.

The implementation of the DG algorithm in GS2 described here is built on preparatory work performed by Hammer and Hatzky [2] of the EFDA High Level Support Team (EFDA workprogramme 2009 under project WP09-HPC-HLST). This report provides a full description of the method as implemented in GS2, including a detailed derivation of the chosen DG scheme and the necessary changes that had to be made to the form of the gyrokinetic equation. A summary of the adaptive Runge-Kutta time advancement method employed is given, together with an overview of the code modifications that were made.

We conclude with the presentation of results obtained by running GS2 with the new explicit scheme, and compare these with results obtained from equivalent runs with the original implicit scheme. We demonstrate that the DG implementation has been successful, with close agreement between the two methods being evident.

2 The GS2 Gyrokinetic Equation and DG

First we turn to the physics, and describe how we rearrange the gyrokinetic equation so that it is in a suitable form for the DG implementation.

2.1 Rearrangement of the form of the GS2 gyrokinetic equation

The non-adiabatic part of the perturbed distribution function g_s is obtained from the linearised electromagnetic gyrokinetic equation:

$$\frac{\partial g_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla g_s + C_s g_s = -q_s \frac{\partial f_{0s}}{\partial E}\frac{\partial \chi}{\partial t} + \{\chi, f_{0s}\} \tag{1}$$

$$\text{where} \quad \chi = \left(\Phi - v_\parallel A_\parallel\right) J_0(Z_s) + \frac{m_s v_\perp^2}{q_s}\frac{B_\parallel}{B}\frac{J_1(Z_s)}{Z_s} \tag{2}$$

$$Z_s = k_\perp \rho_s$$

$$\text{and} \quad \{\chi, f_{0s}\} = \frac{1}{B}\nabla_\perp \chi \times \mathbf{b}.\nabla f_{0s} \tag{3}$$

The perturbed electromagnetic fields $\Phi, A_\parallel, B_\parallel$ contributing to the gyrokinetic electromagnetic potential χ are determined self-consistently from Maxwell's equations. Full details about the derivation and implementation of the field equations are contained in the Appendix.

GS2 solves the gyrokinetic equation above for a modified form of the perturbed distribution function h_s:

$$h_s = g_s - \frac{q_s}{T_s}\Phi J_0(Z) f_{0s} - \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z} f_{0s} \tag{4}$$

It should be noted that GS2 confusingly uses the variable named `g` for g_s **and** h_s (though of course never both at the same time); conversion between the two is performed via subroutine `g_adjust` in `dist_fn.f90`.

Combining Equations 1, 2 and 4 gives the form of the gyrokinetic equation that is numerically solved in GS2:

$$\begin{aligned}\frac{\partial h_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla h_s + C_s g_s = {} & q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} \\ & - f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] + \{\chi, f_{0s}\}\end{aligned} \tag{5}$$

The right hand side of Equation 5 is evaluated by routine `set_source`, '`contain`'ed within routine `get_source_term` in `dist_fn.f90`.

In order to convert this equation into a form suitable for use with the DG scheme discussed later, it is necessary to subsume the $\partial A_\parallel/\partial t$ term into the left hand side, and this may be done by introducing another change of variables:

$$
\begin{aligned}
i_s &= h_s + \frac{q_s}{T_s} v_\parallel A_\parallel J_0(Z) f_{0s} && (6)\\
&= g_s - \frac{q_s}{T_s} \chi f_{0s} && (7)
\end{aligned}
$$

The conversion between h_s and i_s is performed via new subroutine `g_adjust_exp` in source file `dist_fn.f90` (see Section 6.2).

Equation 5 becomes

$$
\begin{aligned}
&\frac{\partial i_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla i_s - \frac{\partial}{\partial t}\left(\frac{q_s}{T_s} v_\parallel A_\parallel J_0(Z) f_{0s}\right) - (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left(\frac{q_s}{T_s} v_\parallel A_\parallel J_0(Z) f_{0s}\right) + C_s g_s \\
&\qquad = q_s v_\parallel J_0(Z) \frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} - f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] + \{\chi, f_{0s}\} \\
&\frac{\partial i_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla i_s + C_s g_s = q_s v_\parallel J_0(Z) \frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} + \frac{\partial}{\partial t}\left(\frac{q_s}{T_s} v_\parallel A_\parallel J_0(Z) f_{0s}\right) \\
&\qquad - f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] \\
&\qquad + f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left(\frac{q_s}{T_s} v_\parallel A_\parallel J_0(Z)\right) + \{\chi, f_{0s}\} \\
&\frac{\partial i_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla i_s + C_s g_s = q_s v_\parallel J_0(Z) \frac{\partial A_\parallel}{\partial t}\cancelto{0}{\left(\frac{\partial f_{0s}}{\partial E} + \frac{f_{0s}}{T_s}\right)} \\
&\qquad - \frac{q_s}{T_s} f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left(\Phi J_0(Z) + \frac{m_s v_\perp^2}{q_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z} - v_\parallel A_\parallel J_0(Z)\right) + \{\chi, f_{0s}\} \qquad (8)
\end{aligned}
$$

where the cancellation is due to the properties of the Maxwellian.

This is simplified using Equation 2 to give the following evolution equation for i_s:

$$
\begin{aligned}
&\frac{\partial i_s}{\partial t} + (v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla i_s + C_s g_s = -\frac{q_s}{T_s} f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\chi + \{\chi, f_{0s}\} \\
\implies &\frac{\partial i_s}{\partial t} + \mathbf{v_d}.\nabla i_s + v_\parallel \mathbf{b}.\nabla\left(i_s + \frac{q_s}{T_s}\chi f_{0s}\right) + C_s g_s = -\frac{q_s}{T_s} f_{0s}\mathbf{v_d}.\nabla\chi + \{\chi, f_{0s}\} \qquad (9)
\end{aligned}
$$

Finally, we note the following equivalences:

$$
\mathbf{v_d}.\nabla \equiv i\omega_d \quad ; \quad \mathbf{b}.\nabla \equiv \frac{\partial}{\partial z}
$$

and substituting these into Equation 9 we get

$$
\boxed{\frac{\partial i_s}{\partial t} + i\,\omega_d\, i_s + v_\parallel \frac{\partial}{\partial z}\left(i_s + \frac{q_s}{T_s}\chi f_{0s}\right) = -\frac{q_s}{T_s} i\,\omega_d\,\chi f_{0s} - C_s g_s + \{\chi, f_{0s}\}} \qquad (10)
$$

This equation is in a form suitable for solution via the DG method:

$$
\frac{\partial g}{\partial t} + i\,\omega_d g + v_\parallel \frac{\partial}{\partial z}(g + \mathcal{F}) = S \qquad (11)
$$

where $g \equiv i_s$.

3 The Discontinuous Galerkin Scheme

In this section we describe the mathematical formulation used to derive the algorithm coded into GS2.

To summarise with reference to Equation 10, we divide the z domain into a set of finite elements, within each of which the relevant quantities are fitted using a suitable polynomial basis. The Discontinuous Galerkin scheme enables us to convert the advection ($v_{\|}\partial/\partial z$) term into a simple matrix form, with information needing to be passed across the finite elements' boundaries from the upwind direction. On each element the resulting matrix equation is used to evaluate the polynomial coefficients for $\frac{\partial i_s}{\partial t}$, allowing the coefficients of i_s to be advanced in time using a Runge-Kutta technique.

3.1 Derivation for simple advection equation

As stated above, the gyrokinetic equation[1] evolved by GS2 is similar to the 1-D advection equation, and can be written in a suitable form (see Section 2) that allows us to use the DG scheme as an alternative to the existing implicit scheme in the code.

We start by considering the simple 1-D advection equation without sources, for clarity:

$$\frac{\partial g}{\partial t} + v\frac{\partial g}{\partial z} = 0 \tag{12}$$

In GS2 the z direction means the direction along the field lines, or equivalently the poloidal angle θ.

3.1.1 Discretisation of the spatial domain

We divide the domain into a set of N_e finite elements, centred at $z_1, z_2, \cdots, z_{N_e}$ and represent g on each element i as an approximation g_i obtained by fitting a polynomial of order $(p-1)$ to g. Each element straddles p grid points ('nodes'), which are equally-spaced in z (see Figure 1).

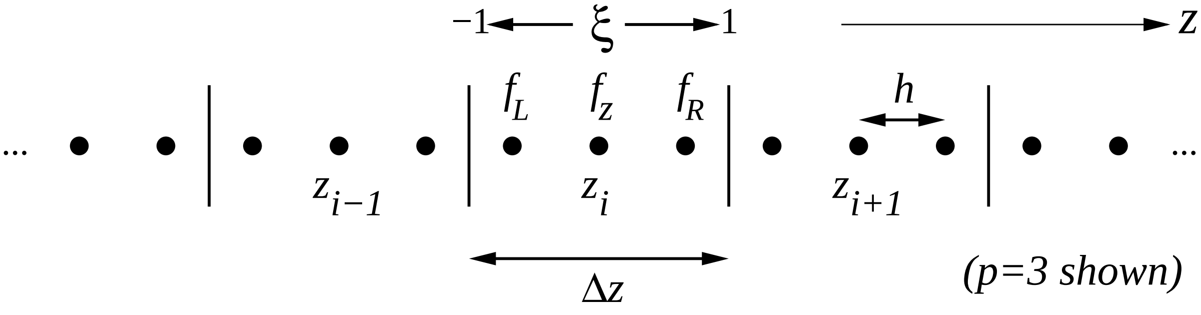

Figure 1: *Diagram showing relationship between finite elements and nodal grid points.*

At present the scheme implemented in the code requires that the N_e finite elements exactly cover the N_z (= `2*ntgrid + 1`) grid points, i.e. $N_z = p\,N_e$. Thus, the extreme ends of the z domain are located at grid point 1 of element $i = 1$ and grid point p of element $i = N_e$. The reason for the present implementation is to allow the arrays used for the nodal and modal representations of a given quantity to be identical in size and shape for convenience, as p modal coefficients describe the polynomial fit through p grid points.

[1] Although the distribution function g evolved by GS2 is a 5-D object, four of the dimensions are 'parallelised out', leaving the poloidal angle z as the only remaining co-ordinate to consider in this discussion.

It is also necessary for symmetry that $z = 0$ be a grid point, so this provides the constraint that p must be odd. Thus our DG implementation assumes that $p = 3$ is a suitable choice.

(Note that the above requirements severely limit the allowed values of GS2 grid dimensioning variables `nperiod` and `ntheta`, and therefore also `ntgrid`, but this requirement could be relaxed relatively straightforwardly by modifying the boundary condition treatment described later.)

We treat each element separately to begin with (hence ‘discontinuous’), and fit the polynomial over the local domain, $-1 \leq \xi \leq 1$:

$$g_i(\xi) = \sum_{m=0}^{p-1} \hat{g}_{i,m} P_m(\xi)$$

where $P(\xi)$ represents a suitable polynomial form. The objects $\hat{g}_{i,m}$ are the ‘modal’ coefficients of the polynomial terms used in the fit.

We solve Equation 12 by asserting that the residual R_i of the approximation must be orthogonal to any test function $\tau(z)$ in the domain $z_{i-\frac{1}{2}} \leq z \leq z_{i+\frac{1}{2}}$:

$$\begin{aligned} R_i(z,t) &= \frac{\partial g_i}{\partial t} + v\frac{\partial g_i}{\partial z} \\ \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} \tau(z)\, R_i(z,t)\, dz &= \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} \tau \left(\frac{\partial g_i}{\partial t} + v\frac{\partial g_i}{\partial z} \right) dz = 0 \end{aligned} \tag{13}$$

(Henceforth we will represent $z_{i-\frac{1}{2}}$ by z_-, and $z_{i+\frac{1}{2}}$ by z_+.)

We integrate the second term of Equation 13 by parts:

$$\int_{z_-}^{z_+} v\tau \frac{\partial g_i}{\partial z}\, dz = \left[v\tau g_i\right]_{z_-}^{z_+} - v\int_{z_-}^{z_+} \frac{\partial \tau}{\partial z} g_i\, dz$$

So (13) becomes

$$\begin{aligned} \int_{z_-}^{z_+} \tau(z)\frac{\partial g_i(z)}{\partial t}\, dz - v\int_{z_-}^{z_+} \frac{\partial \tau(z)}{\partial z} g_i(z)\, dz &= -\left[v\tau g_i\right]_{z_-}^{z_+} \\ &= \tau(z_-)\left\{v\, g\right\}(z_-) - \tau(z_+)\left\{v\, g\right\}(z_+) \end{aligned} \tag{14}$$

The terms in {} are the numerical fluxes required at the element boundaries; these fluxes provide the necessary linkage between adjacent elements. We will assume the simplest, upwind, case in which information only propagates in the direction of the flow v; thus:

$$\left\{vg\right\}(z_-) = \begin{cases} v\, g_{i-1}(z_-) & ; \quad v \geq 0 \\ v\, g_i(z_-) & ; \quad v < 0 \end{cases}$$

$$\left\{vg\right\}(z_+) = \begin{cases} v\, g_i(z_+) & ; \quad v \geq 0 \\ v\, g_{i+1}(z_+) & ; \quad v < 0 \end{cases}$$

3.1.2 Change of spatial co-ordinate variable

Let us now make the change of variable from z to ξ (see Figure 1):

$$\begin{aligned}
\xi = -1 \quad &\text{at} \quad z = z_{i-\frac{1}{2}} = z_- \\
\xi = +1 \quad &\text{at} \quad z = z_{i+\frac{1}{2}} = z_+ \\
\text{By inspection,} \quad \xi(z) &= \frac{2(z-z_i)}{z_+ - z_-} \\
&= \frac{2z - 2\left(\frac{z_+ + z_-}{2}\right)}{z_+ - z_-} \\
\Longrightarrow \quad \xi(z) &= \frac{2z - (z_+ + z_-)}{z_+ - z_-} \\
\frac{d\xi}{dz} &= \frac{2}{z_+ - z_-} = \frac{2}{\Delta z}
\end{aligned}$$

where Δz is the spatial extent covered by the finite element; $\Delta z = h\,p$, with h being the separation of the nodal points on the original grid.

Observe that, for $p = 3$, the three nodes within each element are at $\xi = -\frac{2}{3}$, $\xi = 0$, and $\xi = \frac{2}{3}$.

3.1.3 Legendre polynomials as base functions

We now choose a suitable basis for our fitting polynomial $P(\xi)$ and test function τ. The Legendre polynomials provide one such set of base functions, and have the following properties:

$$\begin{aligned}
P_0(x) = 1 \quad , \quad P_1(x) &= x \qquad (-1 \le x \le 1) \\
\text{Recurrence formula:} \quad P_n(x) &= \frac{2n-1}{n} x\, P_{n-1}(x) - \frac{n-1}{n} P_{n-2}(x) \\
\text{Orthogonality:} \quad \int_{-1}^{1} P_n(x)\, P_m(x)\, dx &= \frac{2}{2n+1}\, \delta_{nm}
\end{aligned}$$

where δ_{nm} is the Kronecker delta.

A smooth function $g(z)$ can be expanded as a series of Legendre polynomials:

$$g(z) = \sum_{m=0}^{\infty} \hat{g}_m P_m(z)$$

and the coefficients $\hat{g}_m$ are obtained via the relation

$$\hat{g}_m = \frac{2m+1}{2} \int_{-1}^{1} g(z)\, P_m(z)\, dz$$

Thus the LHS of Equation 14 may be written as follows; we replace $\tau(z)$ by $P_n(\xi)$, and $g_i(z)$ by $\sum_{m=0}^{p-1} \hat{g}_{i,m} P_m(\xi)$, so

$$\text{LHS} = \int_{z_-}^{z_+} \tau(z) \frac{\partial g_i(z)}{\partial t}\, dz - v \int_{z_-}^{z_+} \frac{\partial \tau(z)}{\partial z} g_i(z)\, dz$$

becomes

$$\text{LHS} = \int_{-1}^{1} P_n(\xi) \sum_{m=0}^{p-1} P_m(\xi) \frac{d\hat{g}_{i,m}}{dt} \frac{dz}{d\xi}\, d\xi - v \int_{-1}^{1} \frac{dP_n(\xi)}{dz} \sum_{m=0}^{p-1} P_m(\xi)\, \hat{g}_{i,m} \frac{dz}{d\xi}\, d\xi$$

and the RHS terms involving the numerical fluxes become:

$$\tau(z_-)\,\{vg\}\,(z_-) = \begin{cases} v\,P_n(-1)\sum_{m=0}^{p-1}\,\hat{g}_{i-1,m}\,P_m(1) & ; \quad v \geq 0 \\ v\,P_n(-1)\sum_{m=0}^{p-1}\,\hat{g}_{i,m}\,P_m(-1) & ; \quad v < 0 \end{cases}$$

$$\tau(z_+)\,\{vg\}\,(z_+) = \begin{cases} v\,P_n(1)\sum_{m=0}^{p-1}\,\hat{g}_{i,m}\,P_m(1) & ; \quad v \geq 0 \\ v\,P_n(1)\sum_{m=0}^{p-1}\,\hat{g}_{i+1,m}\,P_m(-1) & ; \quad v < 0 \end{cases}$$

So, Equation 14 becomes

$$\sum_{m=0}^{p-1}\left\{\frac{dz}{d\xi}\left(\int_{-1}^{1}P_n(\xi)\,P_m(\xi)\,d\xi\right)\frac{d\hat{g}_{i,m}}{dt}\right\} - v\sum_{m=0}^{p-1}\left\{\frac{dz}{d\xi}\left(\int_{-1}^{1}\frac{dP_n(\xi)}{dz}P_m(\xi)\,d\xi\right)\hat{g}_{i,m}\right\}$$
$$= \begin{cases} v\,P_n(-1)\sum_{m=0}^{p-1}P_m(1)\,\hat{g}_{i-1,m} - v\,P_n(1)\sum_{m=0}^{p-1}P_m(1)\,\hat{g}_{i,m} & ; \quad v \geq 0 \\ v\,P_n(-1)\sum_{m=0}^{p-1}P_m(-1)\,\hat{g}_{i,m} - v\,P_n(1)\sum_{m=0}^{p-1}P_m(-1)\,\hat{g}_{i+1,m} & ; \quad v < 0 \end{cases} \tag{15}$$

3.1.4 Mass, stiffness and flux matrices

We can simplify the LHS of Equation 15 by introducing the 'mass matrix' M_{nm} and the 'stiffness matrix' D_{nm}, as follows:

$$M_{nm} \equiv \frac{dz}{d\xi}\int_{-1}^{1}P_n(\xi)\,P_m(\xi)\,d\xi$$
$$D_{nm} \equiv \frac{dz}{d\xi}\int_{-1}^{1}\frac{dP_n(\xi)}{dz}\,P_m(\xi)\,d\xi$$

The LHS of Equation 15 becomes simply

$$\text{LHS} = \sum_{m=0}^{p-1}M_{nm}\,\frac{d\hat{g}_{i,m}}{dt} - v\sum_{m=0}^{p-1}D_{nm}\,\hat{g}_{i,m}$$

We also introduce four so-called 'flux' matrices F_{nm} to simplify the RHS of Equation 15:

$$F_{nm}^{i-1,v_+} \equiv P_n(-1)\,P_m(1)$$
$$F_{nm}^{i,v_+} \equiv P_n(1)\,P_m(1)$$
$$F_{nm}^{i,v_-} \equiv P_n(-1)\,P_m(-1)$$
$$F_{nm}^{i+1,v_-} \equiv P_n(1)\,P_m(-1)$$

$$\Longrightarrow \quad v\,P_n(-1)\sum_{m=0}^{p-1}P_m(1)\,\hat{g}_{i-1,m} = v\,F^{i-1,v_+}\,\hat{\mathbf{g}}_{i-1}$$
$$v\,P_n(1)\sum_{m=0}^{p-1}P_m(1)\,\hat{g}_{i,m} = v\,F^{i,v_+}\,\hat{\mathbf{g}}_{i}$$
$$v\,P_n(-1)\sum_{m=0}^{p-1}P_m(-1)\,\hat{g}_{i,m} = v\,F^{i,v_-}\,\hat{\mathbf{g}}_{i}$$
$$v\,P_n(1)\sum_{m=0}^{p-1}P_m(-1)\,\hat{g}_{i+1,m} = v\,F^{i+1,v_-}\,\hat{\mathbf{g}}_{i+1}$$

where $\hat{\mathbf{g}}_j$ denotes the column vector of length p containing the values $\hat{g}_{j,0}, \hat{g}_{j,1}, \cdots, \hat{g}_{j,p-1}$ for element j.

So we have, for $v \geq 0$,

$$\begin{aligned}\sum_{m=0}^{p-1} M_{nm} \frac{d\hat{g}_{i,m}}{dt} &= v \left[\sum_{m=0}^{p-1} D_{nm}\, \hat{g}_{i,m} + \sum_{m=0}^{p-1} F_{nm}^{i-1,v_+}\, \hat{g}_{i-1,m} - \sum_{m=0}^{p-1} F_{nm}^{i,v_+}\, \hat{g}_{i,m} \right] \\ &= v \left[\sum_{m=0}^{p-1} F_{nm}^{i-1,v_+}\, \hat{g}_{i-1,m} + \sum_{m=0}^{p-1} \left(D_{nm} - F_{nm}^{i,v_+} \right) \hat{g}_{i,m} \right] \qquad (16)\end{aligned}$$

and for $v < 0$,

$$\begin{aligned}\sum_{m=0}^{p-1} M_{nm} \frac{d\hat{g}_{i,m}}{dt} &= v \left[\sum_{m=0}^{p-1} D_{nm}\, \hat{g}_{i,m} + \sum_{m=0}^{p-1} F_{nm}^{i,v_-}\, \hat{g}_{i,m} - \sum_{m=0}^{p-1} F_{nm}^{i+1,v_-}\, \hat{g}_{i+1,m} \right] \\ &= v \left[\sum_{m=0}^{p-1} \left(D_{nm} + F_{nm}^{i,v_-} \right) \hat{g}_{i,m} - \sum_{m=0}^{p-1} F_{nm}^{i+1,v_-}\, \hat{g}_{i+1,m} \right] \qquad (17)\end{aligned}$$

We now evaluate the matrices M_{nm}, D_{nm}, F_{nm} using the properties of Legendre polynomials.

The mass matrix is

$$\begin{aligned} M_{nm} &\equiv \frac{dz}{d\xi} \int_{-1}^{1} P_n(\xi)\, P_m(\xi)\, d\xi \\ &= \frac{\Delta z}{2} \frac{2}{2n+1} \delta_{nm} \\ \implies &\boxed{M_{nm} = \frac{\Delta z}{2n+1} \delta_{nm}} \end{aligned}$$

In matrix form[2], for $p = 3$

$$\begin{aligned} M &= \Delta z \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{5} \end{pmatrix} \\ \implies \quad M^{-1} &= \frac{1}{\Delta z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} \end{aligned}$$

To evaluate the stiffness matrix D_{nm} we use the following recurrence relation involving the derivatives of Legendre polynomials:

$$\begin{aligned} (2m+1) P_m(x) &= \frac{dP_{m+1}(x)}{dx} - \frac{dP_{m-1}(x)}{dx} \\ \implies \quad \frac{dP_{m+1}(x)}{dx} &= (2m+1) P_m(x) + \frac{dP_{m-1}(x)}{dx} \\ \text{Relabelling: } n = m+1 \implies \quad \frac{dP_n(x)}{dx} &= \left(2(n-1)+1\right) P_{n-1}(x) + \frac{dP_{n-2}(x)}{dx} \\ &= \left(2(n-1)+1\right) P_{n-1}(x) + \left(2(n-3)+1\right) P_{n-3}(x) + \cdots \end{aligned}$$

This can be written as the following series:

$$\frac{dP_n(x)}{dx} = \sum_{\substack{j\,=\,0 \\ j\,+\,n \text{ odd}}}^{n-1} (2j+1)\, P_j(x)$$

[2] Reminder: n denotes row number starting from zero, while m denotes column number starting from zero.

The criterion '$j+n \equiv$ odd' is required, since if n is odd, the only non-zero terms are for $j = n-1, n-3, \cdots, 0$, i.e. j is even. Similarly, if n is even, the only non-zero terms are for $j = n-1, n-3, \cdots, 1$, i.e. j is odd.

So, D is evaluated as follows:

$$\begin{aligned}
D_{nm} &\equiv \frac{dz}{d\xi} \int_{-1}^{1} \frac{dP_n(\xi)}{dz}\, P_m(\xi)\, d\xi \\
&= \frac{dz}{d\xi} \int_{-1}^{1} \frac{d\xi}{dz} \frac{dP_n(\xi)}{d\xi}\, P_m(\xi)\, d\xi \\
&= \int_{-1}^{1} \frac{dP_n(\xi)}{d\xi}\, P_m(\xi)\, d\xi \\
&= \int_{-1}^{1} \sum_{\substack{j=0 \\ j+n \text{ odd}}}^{n-1} (2j+1)\, P_j(\xi)\, P_m(\xi)\, d\xi \\
&= \sum_{\substack{j=0 \\ j+n \text{ odd}}}^{n-1} (2j+1) \int_{-1}^{1} P_j(\xi)\, P_m(\xi)\, d\xi \\
&= \sum_{\substack{j=0 \\ j+n \text{ odd}}}^{n-1} (2j+1) \frac{2}{2j+1} \delta_{jm}
\end{aligned}$$

$$\Longrightarrow \quad \boxed{D_{nm} = \sum_{\substack{j=0 \\ j+n \text{ odd}}}^{n-1} 2\, \delta_{jm}}$$

In matrix form, for $p = 3$

$$D = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

The first matrix encodes the logic $[\text{column } j] \leq ([\text{row } n] - 1)$ `AND` $(j+n)$ is odd. The second matrix is simply $2\delta_{jn}$.

$$\Longrightarrow \quad D = \begin{pmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 2 & 0 \end{pmatrix}$$

Finally, the flux matrices are easily calculated using the identities

$$\begin{aligned}
P_n(1) &= 1 \quad \text{for all } n \\
P_n(-1) &= (-1)^n \\
P_n(-1)\, P_m(-1) &= (-1)^n (-1)^m = (-1)^{n+m}
\end{aligned}$$

So, for $p = 3$

$$F^{i-1,v_+} = \begin{pmatrix} 1 & 1 & 1 \\ -1 & -1 & -1 \\ 1 & 1 & 1 \end{pmatrix}, \quad F^{i,v_+} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

$$F^{i,v_-} = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}, \quad F^{i+1,v_-} = \begin{pmatrix} 1 & -1 & 1 \\ 1 & -1 & 1 \\ 1 & -1 & 1 \end{pmatrix}$$

So, at last we can write down in matrix form the equations relevant to $v \geq 0$ and $v < 0$. Equation 16 for $v \geq 0$ is

$$\begin{aligned}\sum_{m=0}^{p-1} M_{nm} \frac{d\hat{g}_{i,m}}{dt} &= v \left[\sum_{m=0}^{p-1} F_{nm}^{i-1,v_+}\, \hat{g}_{i-1,m} + \sum_{m=0}^{p-1} \left(D_{nm} - F_{nm}^{i,v_+} \right) \hat{g}_{i,m} \right] \\ \implies \quad \frac{d\hat{\mathbf{g}}_i}{dt} &= v\, M^{-1} \left(F^{i-1,v_+} \hat{\mathbf{g}}_{i-1} + \left(D - F^{i,v_+} \right) \hat{\mathbf{g}}_i \right)\end{aligned}$$

or, explicitly,

$$\frac{d}{dt} \begin{pmatrix} \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \end{pmatrix} = \frac{v}{\Delta z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} \hat{g}_{i-1,0} \\ \hat{g}_{i-1,1} \\ \hat{g}_{i-1,2} \\ \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \end{pmatrix} \qquad v \geq 0 \qquad (18)$$

Similarly, Equation 17 for $v < 0$ is

$$\begin{aligned}\sum_{m=0}^{p-1} M_{nm} \frac{d\hat{g}_{i,m}}{dt} &= v \left[\sum_{m=0}^{p-1} \left(D_{nm} + F_{nm}^{i,v_-} \right) \hat{g}_{i,m} - \sum_{m=0}^{p-1} F_{nm}^{i+1,v_-}\, \hat{g}_{i+1,m} \right] \\ \implies \quad \frac{d\hat{\mathbf{g}}_i}{dt} &= v\, M^{-1} \left(\left(D + F^{i,v_-} \right) \hat{\mathbf{g}}_i - F^{i+1,v_-} \hat{\mathbf{g}}_{i+1} \right)\end{aligned}$$

or, explicitly,

$$\frac{d}{dt} \begin{pmatrix} \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \end{pmatrix} = \frac{v}{\Delta z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \\ \hat{g}_{i+1,0} \\ \hat{g}_{i+1,1} \\ \hat{g}_{i+1,2} \end{pmatrix} \qquad v < 0 \qquad (19)$$

3.2 Modifications for actual advection equation form in GS2

Recall that the above derivation was for the simplest form of the advection equation (12). However, the form of the equation in GS2 is more complicated (Equation 11):

$$\frac{\partial g}{\partial t} + i\, \omega_d\, g + v_\parallel \frac{\partial}{\partial z} (g + \mathcal{F}) = S$$

These variations are easily translated into the matrix equations, as follows:

$$\boxed{\frac{d}{dt} \begin{pmatrix} \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \end{pmatrix} = -i \begin{pmatrix} \widehat{\omega_d\, g}_{i,0} \\ \widehat{\omega_d\, g}_{i,1} \\ \widehat{\omega_d\, g}_{i,2} \end{pmatrix} + \frac{v_\parallel}{\Delta z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} \hat{g}_{i-1,0} + \hat{\mathcal{F}}_{i-1,0} \\ \hat{g}_{i-1,1} + \hat{\mathcal{F}}_{i-1,1} \\ \hat{g}_{i-1,2} + \hat{\mathcal{F}}_{i-1,2} \\ \hat{g}_{i,0} + \hat{\mathcal{F}}_{i,0} \\ \hat{g}_{i,1} + \hat{\mathcal{F}}_{i,1} \\ \hat{g}_{i,2} + \hat{\mathcal{F}}_{i,2} \end{pmatrix} + \begin{pmatrix} \hat{S}_{i,0} \\ \hat{S}_{i,1} \\ \hat{S}_{i,2} \end{pmatrix}} \qquad v_\parallel \geq 0 \qquad (20)$$

$$\boxed{\frac{d}{dt} \begin{pmatrix} \hat{g}_{i,0} \\ \hat{g}_{i,1} \\ \hat{g}_{i,2} \end{pmatrix} = -i \begin{pmatrix} \widehat{\omega_d\, g}_{i,0} \\ \widehat{\omega_d\, g}_{i,1} \\ \widehat{\omega_d\, g}_{i,2} \end{pmatrix} + \frac{v_\parallel}{\Delta z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} \hat{g}_{i,0} + \hat{\mathcal{F}}_{i,0} \\ \hat{g}_{i,1} + \hat{\mathcal{F}}_{i,1} \\ \hat{g}_{i,2} + \hat{\mathcal{F}}_{i,2} \\ \hat{g}_{i+1,0} + \hat{\mathcal{F}}_{i+1,0} \\ \hat{g}_{i+1,1} + \hat{\mathcal{F}}_{i+1,1} \\ \hat{g}_{i+1,2} + \hat{\mathcal{F}}_{i+1,2} \end{pmatrix} + \begin{pmatrix} \hat{S}_{i,0} \\ \hat{S}_{i,1} \\ \hat{S}_{i,2} \end{pmatrix}} \qquad v_\parallel < 0 \qquad (21)$$

(In the above, non-subscripts $i = \sqrt{-1}$, while subscripts containing i continue to refer to elements, as before.)

Note that there is the slight complication in that ω_d and $v_\parallel$ are not independent of z, so it is important to take the product of such terms with g etc. in 'nodal' space before converting the result to modal form. We have shown this above for the $i\,(\widehat{\omega_d\, g})_{i,m}$ terms, but the same approach must be used for the $v_\parallel\, \partial(g + \mathcal{F})/\partial z$ terms.

3.3 Conversion between nodal and modal forms

During the evaluation of Equations 20 and 21, and also elsewhere in the solution process, it is necessary to be able to convert back and forth between modal and nodal representations. Recall that in the $p = 3$ scheme, each finite element spans three equally-spaced grid points in the z direction. Within each element we fit Legendre polynomials $P_n(\xi)$, $n = 0, 1, 2$, over the internal range $\xi = (-1, 1)$ through the three grid points, which lie at $\xi = -\frac{2}{3}$, $\xi = 0$, $\xi = \frac{2}{3}$, respectively. For some quantity f, we can label its values at the three points as f_L, f_z and f_R respectively, and we can find the corresponding modal coefficients $\hat{f}_0, \hat{f}_1, \hat{f}_2$ by solving the three equations stemming from the following definition:

$$f(\xi) = \hat{f}_0\, P_0(\xi) + \hat{f}_1\, P_1(\xi) + \hat{f}_2\, P_2(\xi)$$

i.e.

$$\begin{aligned}
f_L &= \hat{f}_0\, P_0\left(-\frac{2}{3}\right) + \hat{f}_1\, P_1\left(-\frac{2}{3}\right) + \hat{f}_2\, P_2\left(-\frac{2}{3}\right)\\
f_z &= \hat{f}_0\, P_0\,(0) + \hat{f}_1\, P_1\,(0) + \hat{f}_2\, P_2\,(0)\\
f_R &= \hat{f}_0\, P_0\left(\frac{2}{3}\right) + \hat{f}_1\, P_1\left(\frac{2}{3}\right) + \hat{f}_2\, P_2\left(\frac{2}{3}\right)
\end{aligned}$$

where the Legendre polynomials are defined as

$$\begin{aligned}
P_0(\xi) &= 1\\
P_1(\xi) &= \xi\\
P_2(\xi) &= \frac{1}{2}(3\xi^2 - 1)
\end{aligned}$$

Thus, to convert from nodal values to modal coefficients, we have

$$\begin{aligned}
f_L &= \hat{f}_0 - \frac{2}{3}\,\hat{f}_1 + \frac{1}{6}\,\hat{f}_2\\
f_z &= \hat{f}_0 - \frac{1}{2}\,\hat{f}_2\\
f_R &= \hat{f}_0 + \frac{2}{3}\,\hat{f}_1 + \frac{1}{6}\,\hat{f}_2
\end{aligned}$$

$$\text{or} \quad \begin{pmatrix} f_L \\ f_z \\ f_R \end{pmatrix} = \begin{pmatrix} 1 & -\frac{2}{3} & \frac{1}{6} \\ 1 & 0 & -\frac{1}{2} \\ 1 & \frac{2}{3} & \frac{1}{6} \end{pmatrix} \begin{pmatrix} \hat{f}_0 \\ \hat{f}_1 \\ \hat{f}_2 \end{pmatrix}$$

and to make the reverse transformation we have the following relationship (by matrix inversion):

$$\begin{pmatrix} \hat{f}_0 \\ \hat{f}_1 \\ \hat{f}_2 \end{pmatrix} = \begin{pmatrix} \frac{3}{8} & \frac{1}{4} & \frac{3}{8} \\ -\frac{3}{4} & 0 & \frac{3}{4} \\ \frac{3}{4} & -\frac{3}{2} & \frac{3}{4} \end{pmatrix} \begin{pmatrix} f_L \\ f_z \\ f_R \end{pmatrix}$$

4 Solution procedure and boundary conditions

The procedure for evaluating $d\hat{g}/dt$ and dealing with the boundary conditions is performed in routine `dydt_dggs2`, and may be summarised as follows:

1. The input to the routine is the array of coefficients comprising $\hat{g}$ for all the finite elements, where $\hat{g}$ represents the modal coefficients for a *fully-consistent* (i.e. boundary condition satisfying) distribution function g; either the quantity from the previous time-step (g_{old}), or an intermediate g calculated at a particular step in the Runge-Kutta algorithm (see Section 5).

2. $\hat{g}$ is converted to its nodal representation g, and this is used to calculate the physical quantities Φ, $A_{\|}$, $B_{\|}$ by evaluating the Maxwell field equations; see Section 6.2. These are then used to calculate the source term S and the flux function $\mathcal{F}$ at each grid point. S is converted to its modal form $\hat{S}$.

3. The $-i\,\omega_d\,g$ term is calculated at each grid point, and converted to its modal representation.

4. The quantity $g + \mathcal{F}$ is calculated and converted to modal form, and the relevant matrix multiplication is done to perform the d/dz operation as described above (in particular, in Section 3.2). The result is converted back to nodal form to enable the multiplication by $v_{\|}/\Delta z$ to be done correctly (remembering the definition of Δz from Section 3.1.2), after which the product is returned to modal form.

   Note that Equation 20 for $v_{\|} \geq 0$ uses values from element $i-1$, i.e. the one to the left of the current element i, to calculate the d/dz terms, and Equation 21 for $v_{\|} < 0$ uses values from element $i+1$. At the extreme ends of the domain, this is dealt with by conceptually adding a 'ghost' element to the left or right as appropriate, containing zeroes for coefficients. This ensures that the d/dz terms are not contaminated with spurious contributions from outside the true domain.

5. At this point we now have all the terms on the RHS of Equations 20 and 21 in modal form, which we simply sum to obtain the $d\hat{g}/dt$ array.

6. We can now apply the necessary boundary conditions *a posteriori*. It must be remembered that this is an *explicit* scheme. As long as the *input* $\hat{g}$ array is consistent with the boundary conditions (which it should be!), the nodal dg/dt values corresponding to the just-calculated $d\hat{g}/dt$ coefficients are guaranteed to be correct at all grid points, except (possibly) at those grid points at which the boundary conditions must be re-applied. Unlike in the existing implicit algorithm, no integration is performed in the z direction—we are evaluating dg/dt, not g, and the d/dz terms are evaluated explicitly via the matrix equations. We may therefore safely apply the boundary conditions at the required points *a posteriori*, without affecting the dg/dt values elsewhere.

Passing particles and trapped particles (which occur under certain conditions at particular pitch angles) have different boundary conditions, and must be treated separately.

4.1 Passing particles

Boundary conditions are applied at the ends of the z domain. The required constraint is that $g = 0$ for incoming particles, i.e.

$$\begin{aligned} g(v_{\|} \geq 0) &= 0 \quad \text{at the left-most grid point} \\ g(v_{\|} < 0) &= 0 \quad \text{at the right-most grid point} \end{aligned}$$

This is straightforward to implement. We simply convert the $d\hat{g}/dt$ coefficients into nodal form, and set $dg/dt = 0$ at the left-most grid point for the particles with $v_{\|} \geq 0$, and similarly at the right-most grid point for the particles with $v_{\|} < 0$. Thus, we guarantee that $g_{\text{new}} = g_{\text{old}} = 0$ at these grid points as required.

4.2 Trapped particles

Trapped particles exist only within certain accessible portions of the poloidal angle range; so-called forbidden regions are therefore present for these particles, centred on the inboard side of the flux surface ($\theta = \pm\pi$).

- The pitch angle λ of a particle determines whether it will be trapped or not. The logical array `forbid(ig,il)` denotes whether a particular location `ig` on a field line is within a forbidden region for a particle with pitch angle `il`. The distribution function g is zero by definition in such regions, so we force $dg/dt = 0$ wherever `forbid(ig,il)` is true.
- The grid point adjacent to either end of a forbidden region is a so-called bounce point, where a trapped particle's parallel velocity $v_\parallel$ becomes zero (physically, the particle reverses direction). At these points, $g(v_\parallel > 0) = g(v_\parallel < 0)$, that is to say, there are as many "right-going" particles as "left-going" particles. Thus we must have $dg/dt(v_\parallel > 0) = dg/dt(v_\parallel < 0)$ at the bounce points.

  Therefore we apply boundary conditions at each bounce point, such that the value used is that calculated for those particles with $v_\parallel$ in the direction from the accessible region towards the bounce point in question:

  - Lower bounce points: $dg/dt(v_\parallel > 0) = dg/dt(v_\parallel < 0)$
  - Upper bounce points: $dg/dt(v_\parallel < 0) = dg/dt(v_\parallel > 0)$

There is a subtlety, though, in that the advection term in the matrix equation for the grid points up to $p + 1$ points away from each bounce point require values of $\mathcal{F}$ from the adjacent forbidden region to be used. However, there is some concern that these values may be unphysical/meaningless... To be resolved!

Totally trapped particles are also not treated properly yet.

4.3 Multiplication factors for certain quantities

Many of the variables within GS2 contain normalised quantities. Some of these normalisations needed to be modified for the DG scheme implementation, specifically those for the parallel velocity $v_\parallel$, drift frequency ω_d, and the source term S.

By studying the finite difference coding for the implicit scheme within `dist_fn.f90` (routines `init_invert_rhs` and `invert_rhs_1`), we can deduce how the GS2 variables relate to the physical quantities. Consider the following model equation depicting the form of the equation evolved implicitly in GS2:

$$\frac{\partial g}{\partial t} + i\,\omega_d g + v_\parallel \frac{\partial g}{\partial z} = S_0 \tag{22}$$

We derive the corresponding finite difference equation given the degree of upwinding s, $-1 \leq s \leq 1$, and the degree of explicitness f, $0 \leq f \leq 1$ (the values of s and f may be chosen by the user). The resulting FD equation is:

$$\begin{aligned}
&\left\{(1+s) + (1-f)\left(i\,\omega_d \Delta t(1+s) + 2v_\parallel \frac{\Delta t}{h}\right)\right\} g^{\text{new}}_{ig+1} + \\
&\quad \left\{(1-s) + (1-f)\left(i\,\omega_d \Delta t(1-s) - 2v_\parallel \frac{\Delta t}{h}\right)\right\} g^{\text{new}}_{ig} = 2S_0\Delta t \\
&\qquad\qquad + \left\{(1+s) - f\left(i\,\omega_d \Delta t(1+s) + 2v_\parallel \frac{\Delta t}{h}\right)\right\} g^{\text{old}}_{ig+1} \\
&\qquad\qquad + \left\{(1-s) - f\left(i\,\omega_d \Delta t(1-s) - 2v_\parallel \frac{\Delta t}{h}\right)\right\} g^{\text{old}}_{ig}
\end{aligned} \tag{23}$$

where h is the distance between spatial grid points (see Figure 1) and subscript ig labels the z grid point.

Now, using the definitions of arrays `ainv`, `a`, `b` and `r` in routine `init_invert_rhs`, and expanding the equation for the time advancement of the line labelled '$v > 0$ inhomogeneous part' in routine `invert_rhs_1`, we obtain the following finite difference equation:

$$\begin{aligned}
&\left\{(1+s) + (1-f)\frac{T}{q}\left(i\,\mathtt{w}_{\mathtt{ig}}(1+s) + 2\mathtt{v}_{\mathtt{ig}}\right)\right\}\mathtt{g}_{\mathtt{ig+1}}{}^{\mathrm{new}} + \\
&\quad\left\{(1-s) + (1-f)\frac{T}{q}\left(i\,\mathtt{w}_{\mathtt{ig}}(1-s) - 2\mathtt{v}_{\mathtt{ig}}\right)\right\}\mathtt{g}_{\mathtt{ig}}{}^{\mathrm{new}} = \mathtt{S}_0 \\
&\qquad\qquad + \left\{(1+s) - f\frac{T}{q}\left(i\,\mathtt{w}_{\mathtt{ig}}(1+s) + 2\mathtt{v}_{\mathtt{ig}}\right)\right\}\mathtt{g}_{\mathtt{ig+1}}{}^{\mathrm{old}} \\
&\qquad\qquad + \left\{(1-s) - f\frac{T}{q}\left(i\,\mathtt{w}_{\mathtt{ig}}(1-s) - 2\mathtt{v}_{\mathtt{ig}}\right)\right\}\mathtt{g}_{\mathtt{ig}}{}^{\mathrm{old}} \qquad (24)
\end{aligned}$$

where $\mathtt{g}_{\mathtt{ig}} = \mathtt{g(ig,1,iglo)}$, $\mathtt{v}_{\mathtt{ig}} = \mathtt{vpar(ig,1,iglo)}$, $\mathtt{w}_{\mathtt{ig}} = (\mathtt{wdrift(ig,iglo)} + \mathtt{wcoriolis(ig,iglo)})$, $\mathtt{S}_0 = \mathtt{source(ig,1)}$ (from routine `get_source_term`) and $T/q = \mathtt{spec(is)\%tz}$. Clearly, Equations 23 and 24 are identical in form and the GS2 quantities need to be multiplied by the following factors to get them into the required form for the DG implementation:

$$\begin{aligned}
S_0 &\equiv \frac{1}{2\Delta t}\,\mathtt{S}_0 \\
\omega_d &\equiv \frac{T}{q}\frac{1}{\Delta t}\,\mathtt{w} \\
v_\parallel &\equiv \frac{T}{q}\frac{h}{\Delta t}\,\mathtt{v} \\
\Longrightarrow \quad \frac{v_\parallel}{\Delta z} &= \frac{v_\parallel}{h\,p} = \frac{T}{q}\frac{1}{p\,\Delta t}\,\mathtt{v}
\end{aligned}$$

5 Runge-Kutta Time Advancement

Section 3 derived the matrix equations 20 and 21 used to evaluate the rate of change of the $\hat{g}$ coefficients within each finite element with respect to time. To evolve $\hat{g}$ as accurately as possible we use a Runge-Kutta (RK) scheme to perform the time-advancement from $\hat{g}_{\mathrm{old}}$ to $\hat{g}_{\mathrm{new}}$, and if we use a so-called adaptive RK scheme we can ensure that the time-step Δt used is the most efficient possible whilst retaining the desired level of accuracy in the solution.

5.1 Adaptive Runge-Kutta scheme

An explicit RK method may be summarised as follows. The derivative y' of quantity y with respect to (say) time t is evaluated at a number of test points in the (y,t) plane, and these y' values are combined in a way specific to the particular RK method in use, to form an estimate of the new value y_{n+1}:

$$\begin{gathered}
y' = f(t,y) \quad ; \quad y(t_0) = y_0 \\
y_{n+1} = y_n + \Delta t \sum_{i=1}^{s} b_i\,k_i
\end{gathered}$$

where

$$\begin{aligned}
k_1 &= f(t_n, y_n) \\
k_2 &= f(t_n + c_2\,\Delta t, y_n + a_{21}\,\Delta t\,k_1) \\
k_3 &= f(t_n + c_3\,\Delta t, y_n + a_{31}\,\Delta t\,k_1 + a_{32}\,\Delta t\,k_2) \\
&\;\vdots \\
k_s &= f(t_n + c_s\,\Delta t, y_n + a_{s1}\,\Delta t\,k_1 + a_{s2}\,\Delta t\,k_2 + \ldots + a_{s,s-1}\,\Delta t\,k_{s-1})
\end{aligned}$$

The number of steps s used in a particular method is related to (but not necessarily the same as) the order P of the scheme. The coefficients used in the above process may be summarised in a Butcher tableau [3]:

$$\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s1} & a_{s2} & \dots & a_{s,s-1} & 0 \\
\hline
 & b_1 & b_2 & \dots & b_{s-1} & b_s
\end{array}$$

An adaptive RK scheme uses two methods, typically one of order P and one of order $P - 1$. For efficiency many of the computation steps need not be replicated as the coefficients are reused in both methods; however, a second row of b coefficients is added to the Butcher tableau to incorporate the second method. By computing two estimates for y_{n+1}, we obtain an estimate of the error in the answer by taking the difference between the two, and this difference is of order $(\Delta t)^P$ (this is therefore described as a Pth order adaptive scheme). This allows us to adjust the time-step automatically, given a chosen error tolerance level, to maintain both sufficient accuracy and efficiency.

The third-order ($P = 3$) adaptive scheme we have chosen to use is described by Stoer and Bulirsch [4], and has the following Butcher tableau:

$$\begin{array}{c|cccc}
0 & & & & \\
0 & \frac{1}{4} & & & \\
0 & -\frac{189}{400} & \frac{729}{800} & & \\
0 & \frac{214}{891} & \frac{1}{33} & \frac{650}{891} & \\
\hline
 & \frac{214}{891} & \frac{1}{33} & \frac{650}{891} & 0 \\
 & \frac{533}{2106} & 0 & \frac{800}{1053} & -\frac{1}{78}
\end{array} \tag{25}$$

In source file `fields_explicit.f90` (see Section 6.1), we have written a new Fortran 90 module `rk_schemes` and a general RK solver routine `rk_adaptive_complex`, in which the above adaptive scheme is implemented in a suitable way for our purposes in GS2 (i.e. for complex y values). Routine `dydt_dggs2` (see Section 4) performs the necessary calculation of $y' = f(t, y)$ as necessary, but uses as inputs the modal coefficients $\hat{g}$ of the distribution function to evaluate two estimates for $d\hat{g}/dt$. Hence, `rk_adaptive_complex` advances the modal coefficients of g, rather than g itself.

On exit from `rk_adaptive_complex`, the modal to nodal conversion of the two estimates of $\hat{g}_{\mathrm{new}}$ is performed.

5.2 Adaptive time-step control

The time-step control for the above RK pair is based on that used in subroutine `rkf45` [5], a code for a Fehlberg 4th-5th order RK method. Time-step control is a potentially difficult area, because of the need not only to keep errors under control, but to do so efficiently, without excessively repeating steps. `rkf45` allows a user to set both a relative error tolerance and an absolute error level. The treatment for a general scheme of order P has been deduced from the original `rkf45` code (in which $P = 5$).

The principle behind the algorithm may be summarised as follows:

- We obtain two arrays of values for the updated distribution function g_{new} using the adaptive RK method described above, namely $g_{\mathrm{new},1}$ from the $(P - 1)$th order RK advance, and $g_{\mathrm{new},2}$ from the Pth order RK advance.
- The user specifies maximum allowable absolute and relative error tolerances between $g_{\mathrm{new},1}$ and $g_{\mathrm{new},2}$, and from these the code computes the ratio R between the actual maximum difference $||g_{\mathrm{new},1} - g_{\mathrm{new},2}||$ and the

maximum allowed difference. This ensures that the time-step will respond to the most sensitive part of g as measured by the difference in the two estimates obtained.

- The time-step Δt to be used on the next time-advance is determined by a factor dependent on the value of R and the order of the scheme P:

$$\Delta t_{n+1} = \Delta t_n \frac{f}{R^{\frac{1}{P}}}$$

where $f = 0.9$ is typically used. So, if $R > f^P$ the time-step is reduced on the next advance, but if $R < f^P$ it is increased, as the algorithm detects that a larger time-step is tolerable — see Figure 2. Therefore, under approximately steady-state conditions the time-step should automatically converge on a value f^P times the maximum tolerable time-step, or $\Delta t \longrightarrow 0.729\, \Delta t_{\max}$ for $f = 0.9, \quad P = 3$.

(The method includes other tests to improve its robustness, not elaborated upon here; for instance if $R > 1$ the code rejects the time-advance that has just occurred, and does the advance again but with the new, lower Δt.)

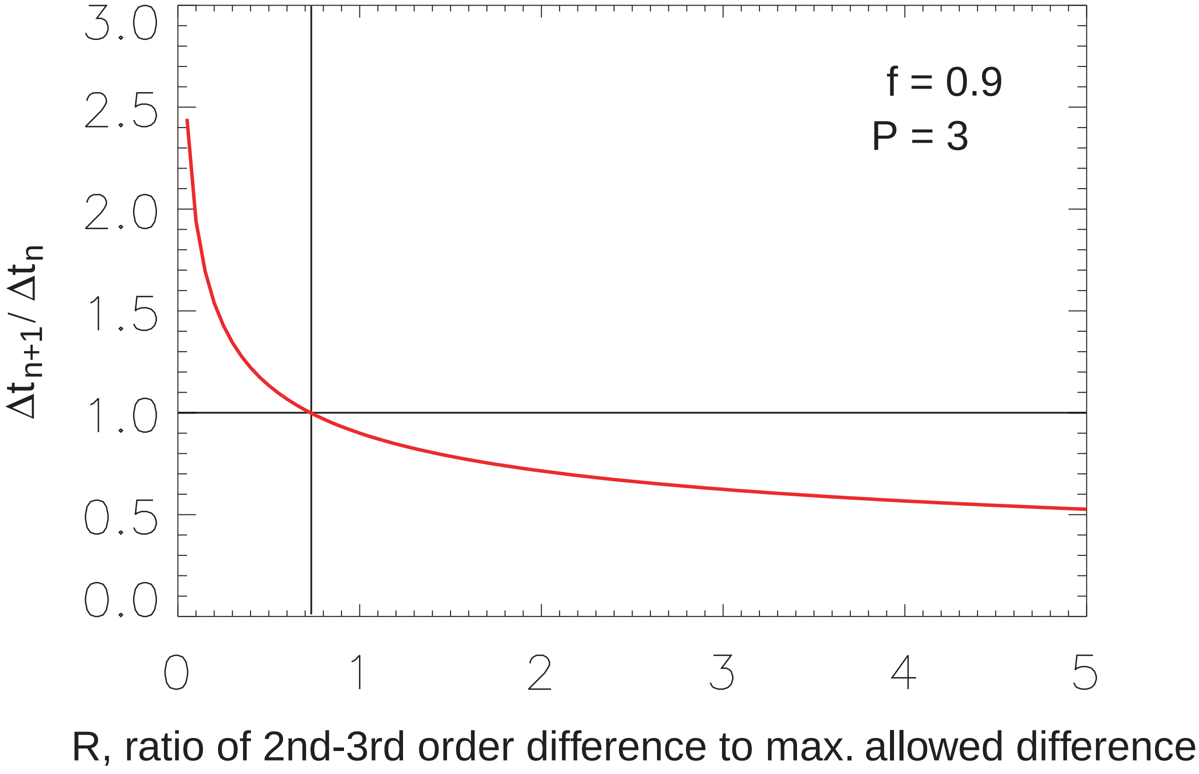

Figure 2: *Plot showing how the time-step varies according to the accuracy achieved on the previous iteration. In a well-behaved scenario, the time-step will settle down on a value such that* $R = f^P$*, indicated by the crossing-point shown.*

6 Implementation of the required changes to GS2

The following describes the changes made to the GS2 source files.

6.1 fields_explicit.f90

The pre-existing file fields_explicit.f90 was merely a placeholder for a previous prototype, unfinished explicit scheme. This file now contains the following subprograms that make up the bulk of the new DG scheme implementation.

- rk_schemes (module): Defines a number of generic Runge-Kutta schemes in terms of their Butcher tableaux.
- dg_scheme (module): Contains the DG scheme working variables, and the nodal to modal (and *vice versa*) conversion routines.
- fields_explicit (module): Contains the primary routines used to perform the explicit evolution of the distribution function, as follows:
  - init_fields_explicit (subroutine): Performs the necessary grid initialisations for the explicit scheme. Called by init_fields in fields.f90.
  - init_phi_explicit (subroutine): Calls getfieldexp to calculate the initial values for quantities Φ, $A_{\|}$ and $B_{\|}$. Called by init_fields in fields.f90.
  - init_dg_scheme (subroutine): Initialises the variables for the adaptive time-step algorithm, and allocates array space for local variables v and wd. Called by init_fields_explicit.
  - reset_init (subroutine): Resets the initialisation flag. Called by reset_time_step in gs2_reinit.f90.
  - advance_explicit (subroutine): This is the primary routine that performs the time-advancement of the distribution function via the DG scheme and the adaptive RK solver. Called by advance in fields.f90.
  - adaptive_dt0 (subroutine): Finds an appropriate value for the initial time-step, for use with the adaptive RK solver. Called by advance_explicit.
  - dydt_dggs2 (subroutine): Builds the array of $d\hat{g}/dt$ values as summarised in Section 4. Called by advance_explicit (through the argument list of rk_adaptive_complex) and adaptive_dt0.
  - rk_adaptive_complex (subroutine): Implements the Runge-Kutta algorithm to provide two estimates of the distribution function at the new time. Called by advance_explicit.

6.2 dist_fn.f90

The following new (or wholly-replaced) subroutines have been added to this file:

- getfieldexp: Calls getan and getfieldeq2 (see below) to evaluate the field equations for Φ, $B_{\|}$ and $A_{\|}$ given the distribution function h_s or i_s. Called by init_phi_explicit, advance_explicit, dydt_dggs2.
- getfieldeq2: Evaluates the field equations for Φ, $B_{\|}$ and $A_{\|}$ explicitly from the species summed velocity integrals derived from the perturbed distribution function, using Equations 47, 48 and 49. Called by getfieldexp.
- g_adjust_exp: Makes the conversion from h_s to i_s or *vice versa*, as described in Equation 6. Called by advance_explicit.

- get_source_term_exp (and its 'contain'ed routine set_source): This routine calculates the RHS source term of Equation 10 (S of Equation 11), with the exception of the collision term $-C_s g_s$ which is dealt with subsequently in existing routine solfp1. Called by dydt_dggs2.

  Details of how the GS2 variables map to the physical quantities shown in the following equations may be found in the Appendix.

  Let us compare the terms on the right-hand side of Equation 5 with those of the RHS of Equation 9:

$$q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} - f_{0s}(v_\parallel \mathbf{b} + \mathbf{v_d}).\nabla\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] + \{\chi, f_{0s}\} \quad \text{(RHS, 5)}$$

$$-f_{0s}\mathbf{v_d}.\nabla\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z} - \frac{q_s}{T_s}v_\parallel A_\parallel J_0(Z)\right] + \{\chi, f_{0s}\} \quad \text{(RHS, 9)}$$

  The differences are plain to see, and these dictate the changes required to create the new routine set_source from the original. In addition to the removal of various terms, it was also necessary to 'shift' some variables back onto the grid points, rather than use grid-centred values that were evaluated midway between grid points.

  The source term S calculated in set_source must be multiplied by the factor $1/(2\,\Delta t)$ to scale it correctly for the DG implementation (see Section 4.3).

  The flux function quantity $\mathcal{F}$ is also calculated in the new set_source routine using the relation:

$$\begin{aligned}\mathcal{F} &\equiv \frac{q_s}{T_s}\chi f_{0s} \\ &= \left(\frac{q_s}{T_s}(\Phi - v_\parallel\, A_\parallel)J_0(Z) + \frac{m_s\, v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right) f_{0s}\end{aligned}$$

  Unlike for S above, there is no need to apply a scale factor to $\mathcal{F}$.

The following routines within the file have been modified:

- allocate_arrays: Added allocation of arrays gwork and gold.
- getan: The evaluation of species summed velocity space integrals (see Table 4) needed to be modified slightly if i_s is the input quantity rather than h_s. The necessary changes have been made to this routine. The hardwired use of variable gnew as the input distribution function was modified so that if an optional argument g is present in the call, this was used in the calculations instead of gnew.
- finish_dist_fn: Added deallocation of arrays gwork and gold.

6.3 dist_fn_arrays.f90

- dist_fn_arrays (module): Declared new array gwork.

6.4 fields.f90

- fields (module): Added public attribute to variables fieldopt_switch, fieldopt_implicit, fieldopt_test and fieldopt_explicit.
- allocate_arrays (subroutine): Added allocation of three arrays phiold, aparold and bparold.
- finish_fields (subroutine): Added deallocation of three arrays phiold, aparold and bparold.

6.5 fields_arrays.f90

- fields_arrays (module): Declared new arrays phiold, aparold, bparold.

6.6 gs2_main.fpp

- run_gs2 (subroutine): Added the coding needed to reset the time-step if the new adaptive time-step algorithm requires it.

6.7 gs2_reinit.f90

- reset_time_step (subroutine): Added the coding needed to prevent problems if the new adaptive time-step algorithm is in use.

7 Results

We present here comparisons between the results for equivalent runs using the original implicit scheme and our newly-implemented explicit scheme ("*DG-RK*").

The important features of the test cases used for these comparisons are as follows:

- Fluctuations in $A_{\|}$ and $B_{\|}$ are suppressed, i.e. the runs are electrostatic.
- No trapped particles are assumed to be present — because of ongoing problems with the coding of the boundary conditions required for trapped particles.
- Electrons may be treated kinetically as a second fluid, or by evolving only the ion species' distribution function and treating the electrons adiabatically.

To make the comparison between the implicit and DG-RK schemes, in each case we show (1) a representative plot of the modulus of the distribution function, $|g(\theta)|$ for a typical pitch angle and particle energy, (2) the final, converged solution of the square of the electrostatic potential, $|\Phi(\theta)|^2$, and (3) a time-trace of the fastest growing part of $|\Phi|^2$ (plotted on a logarithmic scale); the imaginary growth rate γ is the slope of this curve, and its final (converged) value is the quantity of relevance to researchers.

7.1 Fixed time-step results

We performed a comprehensive set of runs to find the limiting time-step $\Delta t_{\max}$. A run with a particular time-step was deemed to be successful if (a) the run converged (using GS2's criterion that the real (Ω) and imaginary (γ) growth rates of the electrostatic potential remain constant over a given number of iterations), and (b) the growth rates are close to those for the corresponding reference implicit scheme case. The limiting time-step was found using a manual interval-halving method. The existence of $\Delta t_{\max}$ is due to the CFL condition that affects the explicit scheme only. The time-step in the implicit scheme affects the accuracy of the result but does not introduce a numerical instability; for the implicit runs we used $\Delta t = 0.01$, well below a value that might affect the accuracy of the result.

Figures 3 and 4 summarise the results gained, for runs treating the electrons adiabatically and kinetically, respectively. The plots show good agreement in all cases between the reference implicit runs and the DG-RK runs, demonstrating our primary result that the DG-RK scheme has been correctly and successfully implemented. Excellent agreement is evident for the adiabatic electrons case. The kinetic electrons case (being a more challenging situation to model because of the higher speed of the electrons) does show a less perfect match between the distribution functions and a slightly raised growth rate, although the results are still tolerably good.

Table 1 gives the growth rates for these runs. The agreement in the growth rates is clearly demonstrated, as is the higher $\Delta t_{\max}$ possible when the electrons are treated adiabatically, rather than being evolved kinetically as a second species. It was found from additional runs performed that, at least for the adiabatic case, the DG-RK scheme can use much larger time-steps (by a factor ~ 4) than the original implicit scheme running in explicit mode can.

(In the Table, runs with `field_option='implicit'` used the original GS2 routines `timeadv`, `invert_rhs`, `get_source_term` and `getfieldeq/getfieldeq1`. Runs with `field_option='explicit'` used the new routines `advance_explicit`, `rk_adaptive_complex`, `get_source_term_exp` and `getfieldexp/getfieldeq2`.)

7.2 Adaptive time-step results

We noted mixed results when the algorithm described in Section 5.2 was invoked. The solution with adiabatic electrons was excellent, with Δt settling quickly down to 0.160, which is very close to the value of $\Delta t_{\max} = 0.162$ shown in Table 1. The growth rates also agreed with the previous fixed time-step run. However, the kinetic electron case failed to converge; Figure 5 demonstrates the problem. The growth rate calculation in GS2 depends on the present value of Δt, and so if this jitters around, the growth rates also oscillate and the GS2 growth rate convergence criterion is never met. Something in the physics of the problem appears to cause the value of R to oscillate around f^3 in small but growing amplitude jumps, until R becomes above 1.0, when the oscillations reset back to small amplitude again, and this cycle repeats indefinitely.

The problem was alleviated by imposing a maximum allowed time-step that is reduced automatically every time $R > 1$ is encountered, thus damping the undesirable oscillations in Δt. The resulting $\Delta t = 0.00143$, close to the previous $\Delta t_{\max} = 0.00165$, and the growth rates agree with the corresponding case in the Table.

8 Concluding Remarks

We have produced a version of GS2 with a working explicit scheme based on a Discontinuous Galerkin finite element method combined with adaptive Runge-Kutta time-advancement. The results presented above have demonstrated, albeit for a reduced class of scenarios, that the new scheme reproduces the results from the existing implicit scheme sufficiently closely to be a useful addition to the code.

The DGGS2 Principal Investigator (Colin Roach) regards the project as having been successful in its aim of implementing the DG scheme within GS2. Notwithstanding the remaining work required to determine and correct the causes of the problems with the trapped particle boundary conditions, this project has resulted in a valuable complementary approach for GS2 users to use.

In the future (outside of the HLST remit), the authors intend to complete the trapped particle implementation, and to run additional tests to ensure that the new scheme is useful in electromagnetic and non-linear scenarios. At that point it will be possible to perform useful, large-scale tests of the algorithm's capability, to see if the predicted savings in computational costs are realised.

scheme	`field_option`	electrons	Δt_{max}	final Ω	final γ
Reference implicit	`implicit`	adiabatic	-	0.271	0.162
DG-RK	`explicit`	adiabatic	0.162	0.271	0.162
Reference implicit	`implicit`	kinetic	-	0.297	0.216
DG-RK	`explicit`	kinetic	0.00165	0.295	0.229

Table 1: *Real (Ω) and imaginary (γ) growth rates for the runs shown in Figures 3 and 4, together with the maximum tolerable time-step Δt_{max}.*

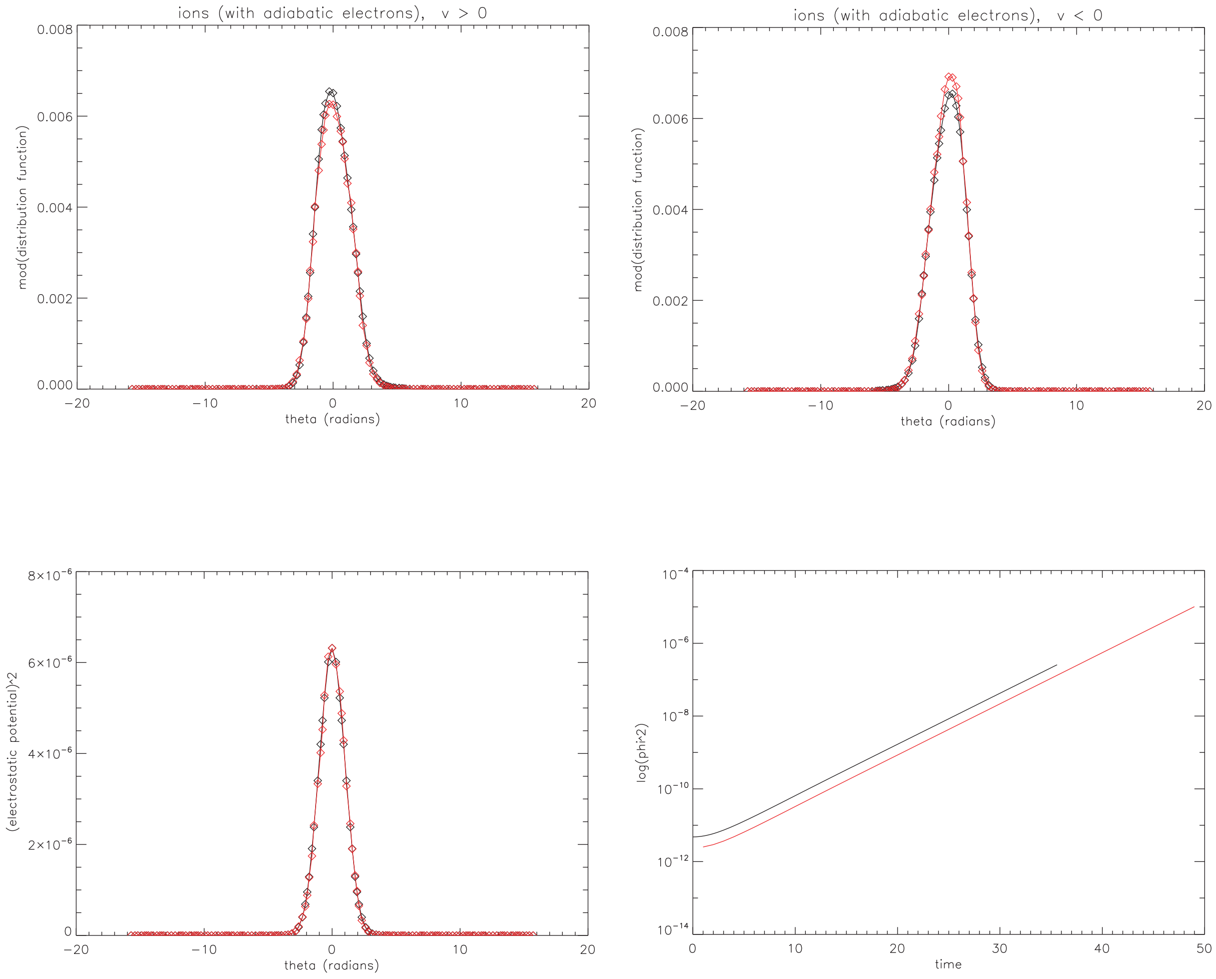

Figure 3: *Plots demonstrating the results for adiabatic electron runs, when using a fixed time-step. The top row shows the distribution functions vs. poloidal angle θ for the ions with positive and negative parallel velocities, respectively. The plot at bottom-left is $|\phi|^2$ vs. θ, and the bottom-right plot is ϕ^2 vs. time (with a logarithmic vertical scale). In all cases, the results using GS2's implicit algorithm are shown in black, while the corresponding results obtained using the new DG-RK algorithm are shown in red.*

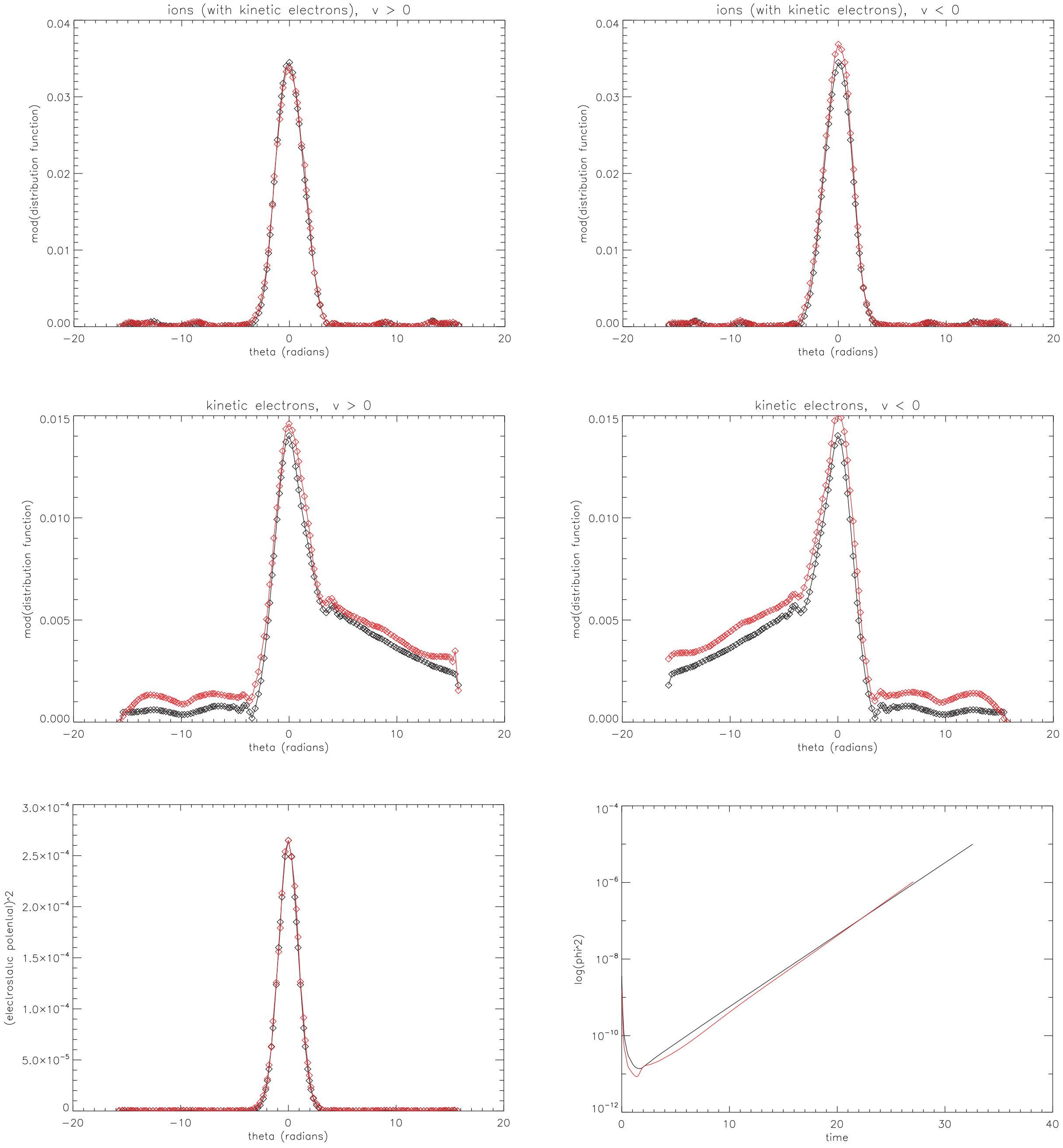

Figure 4: *Plots demonstrating the results for kinetic electron runs, when using a fixed time-step. The top row shows the distribution functions vs. poloidal angle θ for the ions with positive and negative parallel velocities, respectively, while the middle row shows the corresponding electron distribution functions. The plot at bottom-left is $|\phi|^2$ vs. θ, and the bottom-right plot is ϕ^2 vs. time (with a logarithmic vertical scale). In all cases, the results using GS2's implicit algorithm are shown in black, while the corresponding results obtained using the new DG-RK algorithm are shown in red.*

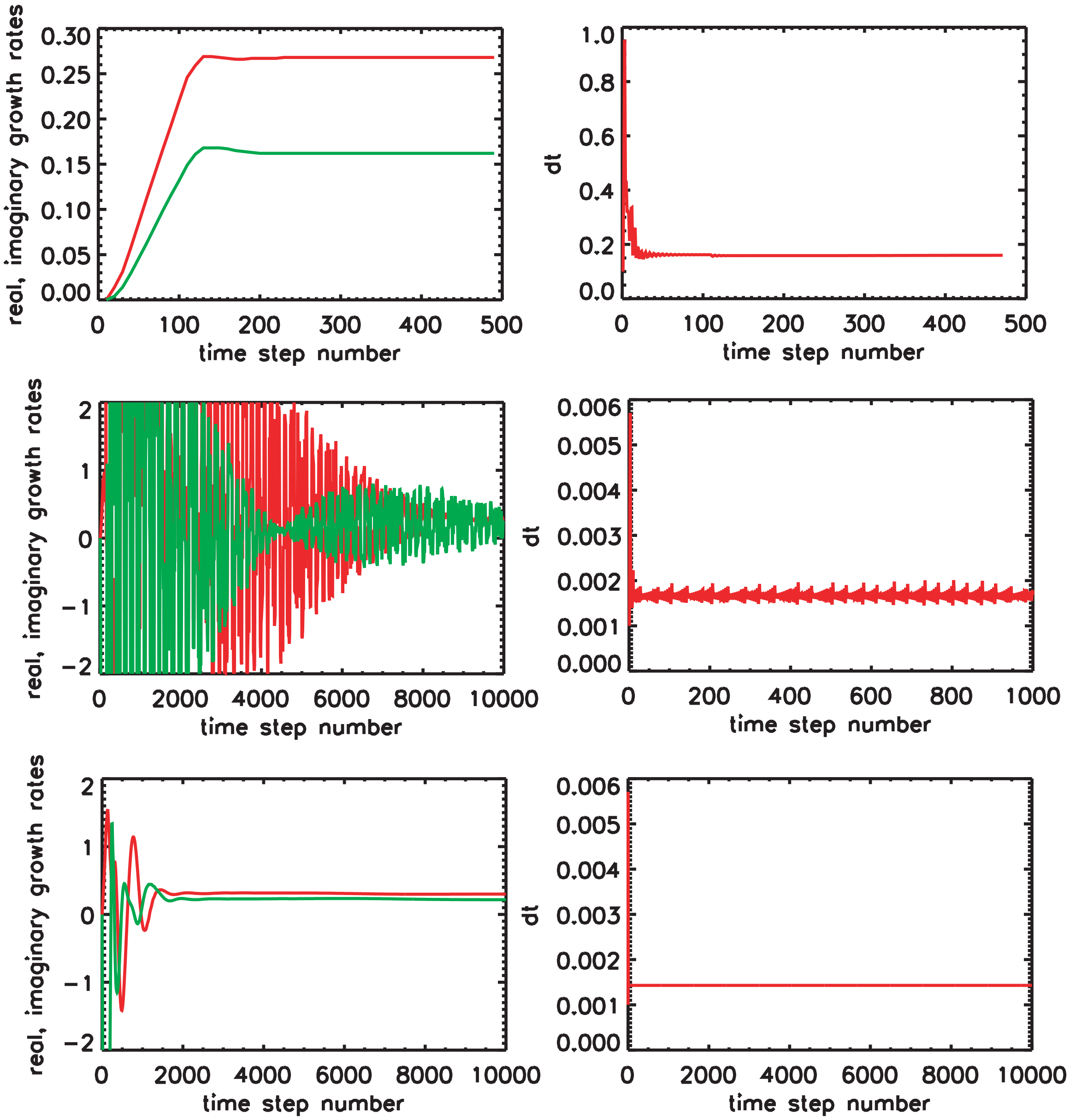

Figure 5: *Plots demonstrating the growth rate and time-step evolution for adiabatic (top) and kinetic (middle) electron cases, when using the adaptive time-step algorithm. The bottom plots show the kinetic electron case when the time-step is subject to additional damping as described in the text.*

Acknowledgements

This work was performed under the EFDA workprogramme 2009 under project WP09-HPC-HLST and its continuations. The UK researchers' work was supported by EPSRC grants EP/I501045 (as part of the RCUK Energy Programme), EP/H00212X/1 and EP/H002081/1, and the European Communities under the Contract of Association between EURATOM and CCFE. The views and opinions expressed herein do not necessarily reflect those of the European Commission. We would like to thank the other members of the High Level Support Team for fruitful discussions during the course of this and the preceding projects.

References

[1] M. Kotschenreuther, G. Rewoldt and W. M. Tang, Computer Physics Communications **88** (1995) 128

[2] Nicolay J. Hammer and Roman Hatsky, *Combining Runge-Kutta discontinuous Galerkin methods with various limiting methods*, IPP Report IPP 5/124, Max-Planck Society (2010), `edoc.mpg.de/446381`

[3] John Butcher, *Runga-Kutta methods*, Scholarpedia, 2(9):3147 (2007)

[4] J. Stoer and R. Bulirsch, §7.2.5*, Introduction to Numerical Analysis, 3rd Edition.* Springer, 2002.

[5] H. A. Watts and L. F. Shampine, Netlib library subroutine `http://www.netlib.org/ode/rkf45.f`

A Gyrokinetic and Maxwell Field Equations in GS2

A.1 Gyrokinetic Equation and Perturbed Distribution Function

In the absence of flows the linear electrostatic gyrokinetic equation is derived for the perturbed distribution function for an isotropic equilibrium:

$$f_{0s} = n_s(x)\left(\frac{m_s}{2\pi T_s(x)}\right)^{1.5} e^{-m_s v^2/2T_s(x)}$$

where $n_s(x)$ and $T_s(x)$ represent the equilibrium temperature and density on a given flux surface labelled by x. The perturbed distribution function at next order in $\delta = \rho/L$ is given by:

$$f_{1s} = \frac{q_s\Phi_1}{m_s}\frac{\partial f_{0s}}{\partial E} + g_s(\boldsymbol{r}, v_\parallel, v_\perp)e^{-i\boldsymbol{k.\rho_s}} \tag{26}$$

where the nonadiabatic part of the perturbed distribution function g_s is obtained from the linearised electromagnetic gyrokinetic equation:

$$\frac{\partial g_s}{\partial t} + \left(v_\parallel \boldsymbol{b} + \boldsymbol{v_d}\right).\boldsymbol{\nabla} g_s + C_s g_s = -q_s\frac{\partial f_{0s}}{\partial E}\frac{\partial \chi}{\partial t} + \{\chi, f_{0s}\} \tag{27}$$

where $\boldsymbol{b} = \boldsymbol{B_0}/B_0$, $Z_s = k_\perp \rho_s$, and:

$$\{\chi, f_{0s}\} = \frac{1}{B}\boldsymbol{\nabla}_\perp \chi \times \boldsymbol{b}.\boldsymbol{\nabla} f_{0s} \qquad\qquad \chi = \left(\Phi_1 - v_\parallel A_\parallel\right) J_0(Z) + \frac{m_s v_\perp^2}{q_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}$$

Φ_1 is perturbed electrostatic potential, $A_\parallel$ is perturbed $\parallel$ magnetic potential, and $B_\parallel$ is perturbed $\parallel$ magnetic field. $\boldsymbol{v_d}$ contains equilibrium magnetic drifts:

$$\boldsymbol{v_d} = \frac{1}{\Omega_s}\boldsymbol{b} \times \left(\frac{\mu}{m}\boldsymbol{\nabla} B + v_\parallel^2 \boldsymbol{b.\nabla} b\right)$$

The linear GKE in (27) is equivalent to equation (24) of [1], which is the formulation of gyrokinetics that is solved in GS2. g_s here corresponds to h that is defined in [1]'s equation (23), if we assume an isotropic equilibrium distribution function that satisfies $\frac{\partial f_{0s}}{\partial \mu} = 0$.

For numerical convenience, GS2 solves the gyrokinetic equation (27) for a modified form of the perturbed distribution function, h_s:

$$h_s(\boldsymbol{r}, E, \mu) = g_s(\boldsymbol{r}, E, \mu) - \frac{q_s}{T_s}\Phi J_0(Z) f_{0s} - \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z} f_{0s} \tag{28}$$

Translation between g_s and h_s is conputed in the GS2 subroutine g_adjust. Substituting (28) into (27) gives the gyrokinetic equation in the form that is most closely related to how the GKE is solved numerically in GS2:

$$\begin{aligned}\frac{\partial h_s}{\partial t} + \left(v_\parallel \boldsymbol{b} + \boldsymbol{v_d}\right).\boldsymbol{\nabla} h_s + C_s g_s = {} & q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} \\ & - f_{0s}\left(v_\parallel \boldsymbol{b} + \boldsymbol{v_d}\right).\boldsymbol{\nabla}\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] \\ & + \{\chi, f_{0s}\}\end{aligned} \tag{29}$$

The terms on the LHS that involve h_s are solved implicitly; the collision term $C_s g_s$ is computed as a separate step using operator splitting; and the source terms on the RHS correspond to the variable *source* that is computed in subroutine set_source in dist_fn.f90.

Velocity space coordinates are $\mu = mv_\perp^2/(2B)$, $E = mv^2/2$ and gyrophase angle γ. A volume element of velocity space $d^3v = v_\perp dv_\perp dv_\parallel d\gamma$, and when the integrand is independent of gyrophase this can be written in terms of two velocity coordinates as $d^3v = 2\pi v_\perp dv_\perp dv_\parallel = 2\pi\frac{Bd\mu dE}{\sqrt{2m_s(E-\mu B)}}$.

A.2 Maxwell's Field Equations in Gyrokinetics

Perturbed electric and magnetic fields obtained from electrostatic and magnetic potentials:

$$\boldsymbol{E_1} = -\boldsymbol{\nabla}\Phi - \epsilon_0 \frac{\partial A_\parallel \boldsymbol{b}}{\partial t} \tag{30}$$

$$\boldsymbol{B_1} = \boldsymbol{\nabla} \times \left(A_\parallel \boldsymbol{b}\right) + B_\parallel \boldsymbol{b} \tag{31}$$

Φ and $A_\parallel$ are determined self-consistently from Maxwell's equations, neglecting the displacement current which is small in the gyrokinetic orderings.

Poisson's equation reduces to the quasineutrality condition if we assume the limit where $k_\perp^2 \lambda_D^2 \ll 1$.

$$\sum_s \int d^3v\; q_s \left(q_s \Phi \frac{\partial f_{0s}}{\partial E} + g_s e^{-i\boldsymbol{k}.\boldsymbol{\rho_s}} \right) = 0$$

Gyrophase integration using $\frac{1}{2\pi}\int d\gamma e^{-i\boldsymbol{k}.\boldsymbol{\rho_s}} = J_0(Z_s)$ gives the GK Poisson equation:

$$-\Phi \sum_s \frac{n_s q_s^2}{T_s} + \sum_s q_s \int d^3v J_0(Z_s) g_s = 0 \tag{32}$$

Gyrokinetic orderings require that $B_\parallel$ satisfies perpendicular force balance, as GK does not capture high frequency fast Alfvèn waves. Some algebra gives the GK solution for the perpendicular component of Ampère's Law :

$$\frac{B_\parallel}{B} = -\frac{\mu_0}{B^2} \sum_s \int d^3v\; m_s v_\perp^2 g_s \frac{J_1(Z_s)}{Z_s} \tag{33}$$

More straightforwardly, the parallel component of Ampère's law gives:

$$k_\perp^2 A_\parallel = \mu_0 \sum_s \int d^3v\; v_\parallel q_s g_s J_0(Z_s) \tag{34}$$

It is clear that $B_1/B \propto \beta$, so that $\Rightarrow$ **$\boldsymbol{B_1}$ more important at high $\boldsymbol{\beta}$**

A.3 Field Equations in Terms of h_s

Substituting (28) for g_s into (32-34), and using summation convention over species, s, gives the field equations in terms of the distribution function h_s that is evolved in GS2.
Quasineutrality to obtain Φ:

$$-\Phi \frac{n_s q_s^2}{T_s} + q_s \int d^3v J_0(Z_s) \left\{ h_s + \frac{q_s}{T_s}\Phi J_0(Z_s) f_{0s} + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z_s)}{Z_s} f_{0s} \right\} = 0$$

$$\Phi \left(\frac{n_s q_s^2}{T_s} - \frac{q_s^2}{T_s} \int d^3v J_0^2(Z_s) f_{0s} \right) = q_s \int d^3v J_0(Z_s) h_s + \frac{B_\parallel}{B}\frac{q_s m_s}{T_s} \int d^3v v_\perp^2 \frac{J_0(Z_s) J_1(Z_s)}{Z_s} f_{0s} \tag{35}$$

Perpendicular force balance determines $B_\parallel$:

$$\frac{B_\parallel}{B} = -\frac{\mu_0}{B^2} \int d^3v\; m_s v_\perp^2 \frac{J_1(Z_s)}{Z_s} \left\{ h_s + \frac{q_s}{T_s}\Phi J_0(Z_s) f_{0s} + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z_s)}{Z_s} f_{0s} \right\}$$

$$\frac{B_\parallel}{B} \left(1 + \frac{\mu_0}{B^2}\frac{m_s^2}{T_s} \int d^3v\; v_\perp^4 \frac{J_1^2(Z_s)}{Z_s^2} f_{0s} \right) = -\frac{\mu_0 m_s}{B^2} \int d^3v\; v_\perp^2 \frac{J_1(Z_s)}{Z_s} h_s - \frac{\mu_0 m_s q_s}{B^2 T_s}\Phi \int d^3v\; v_\perp^2 \frac{J_0(Z_s) J_1(Z_s)}{Z_s} f_{0s} \tag{36}$$

The parallel component of Ampère's law gives $A_\parallel$:

$$k_\perp^2 A_\parallel = \mu_0 q_s \int d^3v\; v_\parallel J_0(Z_s) \left\{ h_s + \frac{q_s}{T_s}\Phi J_0(Z_s) f_{0s} + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z_s)}{Z_s} f_{0s} \right\}$$

$$k_\perp^2 A_\parallel = \mu_0 q_s \int d^3v\; v_\parallel J_0(Z_s) h_s \qquad \text{(integrals odd in } v_\parallel \text{ vanish)} \tag{37}$$

The above three field equations (35-37) are used to obtain Φ, $B_\parallel$ and $A_\parallel$ in GS2. Clearly equations (35) and 36) are coupled.

A.3.1 Adiabatic Species Approximations

In the adiabatic approximation, the perturbed distribution function for species a is given by:

$$f_{1a} = \frac{q_a\left(\Phi_1 - \mathcal{F}_m\right) f_{0a}}{T_a} \tag{38}$$

where $\mathcal{F}_m$ is a model velocity independent spatial surface average that depends on the nature of the perturbation.

Physics Model	$\mathcal{F}_m$ at $k_y = 0$	adiabatic_option
ion response to short wavelength ETG	0	‘iphi00=0’
electron response to long wavelength ITG	$\mathcal{F}_m = \langle\Phi_1\rangle$	‘iphi00=2’

Table 2: Physical adiabatic model options for $\mathcal{F}_m$, which is chosen in GS2 using the variable adiabatic_option. In all models $\mathcal{F}_m = 0$ for any finite k_y. $\langle.\rangle$ denotes the flux-surface average operator.

For Maxwellian equilibrium, f_{1a} is independent of gyrophase and the sign of $v_\parallel$: f_{1a}, therefore, contributes to the perturbed charge density, but not to the perturbed current density. Including adiabatic species modifies the field equations, and the quasineutrality condition (35) is modified as follows, to give:

$$\Phi\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right) - \frac{n_a q_a^2}{T_a}\mathcal{F}_m = q_s\int d^3v J_0(Z_s) h_s + \frac{B_\parallel}{B}\frac{q_s m_s}{T_s}\int d^3v v_\perp^2 \frac{J_0(Z_s)J_1(Z_s)}{Z_s} f_{0s} \tag{39}$$

When adiabatic species are included in the equations for quasineutrality (39), perpendicular force balance (36) and Ampère’s law (37), the implicit species sums over s exclude adiabatic species and are **only** over fully kinetic species.

In the adiabatic model where $\mathcal{F}_m = \langle\Phi_1\rangle$, we obtain the finite value of $\langle\Phi_1\rangle$ (for $k_y = 0$) from equation (39), as follows:

$$\Phi = \frac{q_s\int d^3v J_0(Z_s)h_s + \frac{B_\parallel}{B}\frac{q_s m_s}{T_s}\int d^3v v_\perp^2\frac{J_0(Z_s)J_1(Z_s)}{Z_s} f_{0s} + \frac{n_a q_a^2}{T_a}\langle\Phi\rangle}{\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right)}$$

$$\langle\Phi\rangle = \frac{\left\langle \frac{q_s\int d^3v J_0(Z_s)h_s + \frac{B_\parallel}{B}\frac{q_s m_s}{T_s}\int d^3v v_\perp^2\frac{J_0(Z_s)J_1(Z_s)}{Z_s} f_{0s}}{\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right)}\right\rangle}{1 - \left\langle \frac{\frac{n_a q_a^2}{T_a}}{\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right)}\right\rangle}$$

$$\Rightarrow \mathcal{F}_m = \langle\Phi\rangle = \frac{\left\langle \frac{q_s\int d^3v J_0(Z_s)h_s + \frac{B_\parallel}{B}\frac{q_s m_s}{T_s}\int d^3v v_\perp^2\frac{J_0(Z_s)J_1(Z_s)}{Z_s} f_{0s}}{\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right)}\right\rangle}{\left\langle \frac{\frac{n_s q_s^2}{T_s} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}}{\left(\frac{n_s q_s^2}{T_s} + \frac{n_a q_a^2}{T_a} - \frac{q_s^2}{T_s}\int d^3v J_0^2(Z_s) f_{0s}\right)}\right\rangle} \tag{40}$$

A.4 Implementation of Field Equations in GS2

A.4.1 GS2 Normalisations

All GS2 variables are normalised according to standard normalisations given in Table 3.

We now use Table 3 to normalise the magnetic drifts:

$$\begin{aligned}
\boldsymbol{v_d} &= \frac{1}{\Omega_s}\boldsymbol{b}\times\left(\frac{\mu}{m}\boldsymbol{\nabla}B + v_\parallel^2\boldsymbol{b}.\boldsymbol{\nabla}\boldsymbol{b}\right) = \frac{v_{ts}^2}{L_{\rm ref}\,\Omega_s}\boldsymbol{b}\times\left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}B + \mathsf{v}_\parallel{}^2\boldsymbol{b}.\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}\boldsymbol{b}\right)\\
\boldsymbol{v_d} &= \frac{\mathsf{T_s}\rho_{\rm ref}\,v_t}{\mathsf{q_s}L_{\rm ref}}\boldsymbol{b}\times\left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}B + \mathsf{v}_\parallel{}^2\boldsymbol{b}.\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}\boldsymbol{b}\right)
\end{aligned}$$

to obtain:

$$\boldsymbol{v_d}.\boldsymbol{\nabla}_\perp = \frac{\mathsf{T_s}v_t}{\mathsf{q_s}L_{\rm ref}}\boldsymbol{b}\times\left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}B + \mathsf{v}_\parallel{}^2\boldsymbol{b}.\boldsymbol{\nabla}_{\rm eq}^{\rm GS2}\boldsymbol{b}\right).\boldsymbol{\nabla}_\perp^{\rm GS2} \tag{41}$$

and the gyrokinetic electromagnetic potential:

$$\begin{aligned}
\chi &= \left(\Phi_1 - v_\parallel A_\parallel\right)J_0(Z) + \frac{m_s v_\perp^2}{q_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\\
\chi &= \frac{\rho_{\rm ref}}{L_{\rm ref}}\frac{T_{\rm ref}}{q_{\rm ref}}\left(\mathsf{\Phi} J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel\mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2\mathsf{B}_\parallel\frac{J_1(Z)}{Z}\right).
\end{aligned} \tag{42}$$

A.4.2 Normalised Field Equations

We now multiply all the field equations (39,36,37) by $L_{\rm ref}\,/\rho_{\rm ref}$ to remove the small ordering factor that scales these equations, and reexpress the equations in terms of the GS2 normalisations.

Quasineutrality

The quasineutrality equation (39) can be expressed:

$$\begin{aligned}
LHS &= n_{\rm ref}\,q_{\rm ref}\left[\mathsf{\Phi}\frac{\mathsf{n_s q_s}^2}{\mathsf{T_s}}\left(1 - \int d^3v\ f_0\ J_0^2(Z_s)\right) + \frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\left(\mathsf{\Phi} - \mathsf{F_m}\right)\right]\\
RHS &= n_{\rm ref}\,q_{\rm ref}\,\mathsf{n_s q_s}\left[\int d^3v\ f_0\ J_0(Z_s)\frac{L_{\rm ref}}{\rho_{\rm ref}}\frac{h_s}{f_{0s}} + \frac{B_\parallel}{B}\frac{L_{\rm ref}}{\rho_{\rm ref}}\int d^3v\ f_0\ 2\frac{v_\perp^2}{v_{ts}^2}\frac{J_0(Z_s)J_1(Z_s)}{Z_s}\right]
\end{aligned}$$

to give:

$$\mathsf{\Phi}\frac{\mathsf{n_s q_s}^2}{\mathsf{T_s}}\int d^3v\ f_0\ \left(1 - J_0^2(Z_s)\right) + \frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\left(\mathsf{\Phi} - \mathsf{F_m}\right) = \ \mathsf{n_s q_s}\int d^3v\ f_0\ J_0(Z_s)\mathsf{h_s} + \mathsf{B}_\parallel\mathsf{n_s q_s}\int d^3v\ f_0\ \frac{2\mathsf{v}_\perp{}^2 J_0(Z_s)J_1(Z_s)}{Z_s}. \tag{43}$$

The Φ average term of equation (40) used in one adiabatic response model is expressed in normalised variables as:

$$\langle\mathsf{\Phi}\rangle = \frac{\left\langle\dfrac{\mathsf{q_s n_s}\int d^3v f_0 J_0(Z_s)\mathsf{h_s} + \mathsf{B}_\parallel\mathsf{q_s n_s}\int d^3v f_0\frac{2\mathsf{v}_\perp{}^2 J_0(Z_s)J_1(Z_s)}{Z_s}}{\left(\frac{\mathsf{n_s q_s}^2}{\mathsf{T_s}}\left(1 - \int d^3v f_0 J_0^2(Z_s)\right) + \frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\right)}\right\rangle}{\left\langle\dfrac{\frac{\mathsf{n_s q_s}^2}{\mathsf{T_s}}\left(1 - \int d^3v f_0 J_0^2(Z_s)\right)}{\left(\frac{\mathsf{n_s q_s}^2}{\mathsf{T_s}}\left(1 - \int d^3v f_0 J_0^2(Z_s)\right) + \frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\right)}\right\rangle} \tag{44}$$

Perpendicular Force Balance

Following the same procedure for perpendicular force balance, equation (36), gives:

$$
\begin{aligned}
LHS &= \mathsf{B}_\parallel \left(1 + \frac{\mu_0 n_{\rm ref}\, \mathsf{n_s}}{B_{\rm ref}^2\, \mathsf{B}^2} \frac{m_s^2 v_{ts}^4}{T_s} \int d^3v\ f_0\ \frac{v_\perp^4}{v_{ts}^4}\ \frac{J_1^2(Z_s)}{Z_s^2}\right) = \mathsf{B}_\parallel \left(1 + \left(\frac{2\mu_0 n_{\rm ref}\, T_{\rm ref}}{B_{\rm ref}^2}\right) \frac{\mathsf{n_s T_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{2\mathsf{v}_\perp{}^4 J_1^2(Z_s)}{Z_s^2}\right) \\
&= \mathsf{B}_\parallel \left(1 + \beta \frac{\mathsf{n_s T_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{2\mathsf{v}_\perp{}^4 J_1^2(Z_s)}{Z_s^2}\right) \\
RHS &= -\frac{\mu_0 n_s m_s v_{ts}^2}{B_{\rm ref}^2\, \mathsf{B}^2} \int d^3v\ f_0 \frac{v_\perp^2}{v_{ts}^2}\ \frac{J_1(Z_s)}{Z_s} \mathsf{h_s} - \frac{\mu_0 m_s n_s v_{ts}^2}{B^2} \frac{\mathsf{q_s}}{\mathsf{T_s}} \left(\frac{q_{\rm ref}\ \Phi}{T_{\rm ref}}\right) \int d^3v\ f_0 \frac{v_\perp^2}{v_{ts}^2}\ \frac{J_0(Z_s) J_1(Z_s)}{Z_s} \\
&= -\beta \left(\frac{\mathsf{n_s T_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{\mathsf{v}_\perp^2 J_1(Z_s)}{Z_s} \mathsf{h_s} + \mathsf{\Phi} \frac{\mathsf{n_s q_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{\mathsf{v}_\perp^2 J_0(Z_s) J_1(Z_s)}{Z_s}\right)
\end{aligned}
$$

and GS2's dimensionless perpendicular force balance equation is:

$$
\mathsf{B}_\parallel \left(1 + \beta \frac{\mathsf{n_s T_s}}{\mathsf{B}^2} \int d^3v\ f_0\ 2 \frac{\mathsf{v}_\perp^4 J_1^2(Z_s)}{Z_s^2}\right) = -\beta \frac{\mathsf{n_s T_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{\mathsf{v}_\perp^2 J_1(Z_s)}{Z_s} \mathsf{h_s} - \mathsf{\Phi} \frac{\beta}{2} \frac{\mathsf{n_s q_s}}{\mathsf{B}^2} \int d^3v\ f_0 \frac{2\mathsf{v}_\perp^2 J_0(Z_s) J_1(Z_s)}{Z_s} \tag{45}
$$

Parallel Ampère's Law

In terms of normalised quantities the parallel Ampère's law, equation (37) we have:

$$
\begin{aligned}
LHS &= \frac{B_{\rm ref}}{2\rho_{\rm ref}} \left(k_\perp \rho_{\rm ref}\right)^2 \left(\frac{2A_\parallel}{B_{\rm ref}\ \rho_{\rm ref}}\right) \frac{L_{\rm ref}}{\rho_{\rm ref}} = \frac{B_{\rm ref}}{2\rho_{\rm ref}} \mathsf{k}_\perp{}^2 \mathsf{A}_\parallel = \frac{q_{\rm ref}\ B_{\rm ref}^2}{2m_{\rm ref}\ v_t} \mathsf{k}_\perp{}^2 \mathsf{A}_\parallel \\
RHS &= \mu_0 n_{\rm ref}\ q_{\rm ref}\ v_t \mathsf{n_s q_s} \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}} \int d^3v\ f_0 \frac{v_\parallel}{v_{ts}} J_0(Z_s) \left(\frac{h_s L_{\rm ref}}{f_{0s} \rho_{\rm ref}}\right).
\end{aligned}
$$

The normalised parallel Ampère Law is:

$$
\begin{aligned}
\mathsf{k}_\perp{}^2 \mathsf{A}_\parallel &= \frac{4\mu_0 n_{\rm ref}\ T_{\rm ref}}{B_{\rm ref}^2} \mathsf{n_s q_s} \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}} \int d^3v\ f_0 \mathsf{v}_\parallel J_0(Z_s) \mathsf{h_s} \\
\mathsf{k}_\perp{}^2 \mathsf{A}_\parallel &= 2\beta \mathsf{n_s q_s} \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}} \int d^3v\ f_0 \mathsf{v}_\parallel J_0(Z_s) \mathsf{h_s}
\end{aligned} \tag{46}
$$

$\mathsf{\Phi}$ and $\mathsf{B}_\parallel$ can be determined from equations (43) and (45), explicitly in terms of the perturbed distribution functions. Substituting for the species summed velocity integrals makes the algebra more compact, and here we will use the appropriate GS2 variables.

A.4.3 Species Summed Velocity Integrals in GS2

Various species summed velocity space integrals appear in equations (43-46), and the GS2 variables that contain them are given in Table 4.

A.4.4 GS2 Field Equations

Equations (43), (45) and (46) gives $\mathsf{\Phi}$, $\mathsf{B}_\parallel$ and $\mathsf{A}_\parallel$ explicitly in terms of the perturbed distribution functions.

$$\mathsf{\Phi} = \frac{\left(1 + \frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right)\left(\mathrm{ANTOT} + \frac{\mathsf{n_a q_a}^2\mathsf{F_m}}{\mathsf{T_a}}\right) - \frac{\beta}{\mathsf{B}^2}\mathrm{ANTOTP.GAMTOT1}}{\left(1 + \frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right).\mathrm{GAMTOT} + \frac{\beta}{2\mathsf{B}^2}\mathrm{GAMTOT1}^2} \tag{47}$$

$$\mathsf{B}_\parallel = -\beta\frac{\left(\mathrm{GAMTOT.ANTOTP} + \mathrm{GAMTOT1.}\left(\mathrm{ANTOT} + \frac{\mathsf{n_a q_a}^2\mathsf{F_m}}{\mathsf{T_a}}\right)/2\right)}{\left(\mathsf{B}^2 + \beta\mathrm{GAMTOT2}\right).\mathrm{GAMTOT} + \beta\mathrm{GAMTOT1}^2/2} \tag{48}$$

$$\mathsf{A}_\parallel = \frac{\mathrm{ANTOTA}}{\mathsf{k}_\perp{}^2} \tag{49}$$

The $\mathsf{F_m} = \langle\Phi\rangle$ term used in one adiabatic model, can be expressed in terms of GS2 variables using (44):

$$\langle\mathsf{\Phi}\rangle = \frac{\left\langle\frac{\mathrm{ANTOT}+\mathsf{B}_\parallel.\mathrm{GAMTOT1}}{\mathrm{GAMTOT}}\right\rangle}{\left\langle\frac{\mathrm{GAMTOT}-\frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}}{\mathrm{GAMTOT}}\right\rangle}. \tag{50}$$

A more convenient but cumbersome expression, independent of $\mathsf{B}_\parallel$, can be obtained from equation (47):

$$\langle\mathsf{\Phi}\rangle = \frac{\left\langle\frac{\left(1+\frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right)\mathrm{ANTOT}-\frac{\beta}{\mathsf{B}^2}\mathrm{ANTOTP.GAMTOT1}}{\left(1+\frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right).\mathrm{GAMTOT}+\frac{\beta}{2\mathsf{B}^2}\mathrm{GAMTOT1}^2}\right\rangle}{\left\langle 1 - \frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\left\langle\frac{\left(1+\frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right)}{\left(1+\frac{\beta}{\mathsf{B}^2}\mathrm{GAMTOT2}\right).\mathrm{GAMTOT}+\frac{\beta}{2\mathsf{B}^2}\mathrm{GAMTOT1}^2}\right\rangle\right\rangle} \tag{51}$$

In the $\beta = 0$ limit $\mathsf{B}_\parallel$ must vanish, and (50) and (51) both simplify to give:

$$\langle\mathsf{\Phi}\rangle = \frac{\left\langle\frac{\mathrm{ANTOT}}{\mathrm{GAMTOT}}\right\rangle}{\left\langle\frac{\left(\mathrm{GAMTOT}-\frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}\right)}{\mathrm{GAMTOT}}\right\rangle} = \frac{\left\langle\frac{\mathrm{ANTOT}}{\mathrm{GAMTOT}}\right\rangle}{\langle\mathrm{GAMTOT3}\rangle}. \tag{52}$$

Equation (52) is the model implemented in GS2 by setting adiabatic_option='iphi00=2', and is the appropriate electron adiabatic response to low k_y modes. In GS2 TITE=$\frac{\mathsf{n_a q_a}^2}{\mathsf{T_a}}$, and FL_AVG2=$\frac{\mathsf{n_a q_a}^2\mathsf{F_m}}{\mathsf{T_a}}$ is used in adiabatic models where $\mathsf{F_m}$ is finite, but GS2 does NOT include this term in its equation (48) for $\mathsf{B}_\parallel$.

A.4.5 GS2 Gyrokinetic Equation

Source Terms on RHS

The RHS of the gyrokinetic equation (29) solved in GS2 contains the source terms:

$$RHS = q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} - f_{0s}\left(v_\parallel\boldsymbol{b} + \boldsymbol{v_d}\right).\boldsymbol{\nabla}\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right] + \{\chi, f_{0s}\}$$

$$= \overset{[I]}{q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t}} - \overset{[II]}{f_{0s}v_\parallel\boldsymbol{b}.\boldsymbol{\nabla}\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right]} - \overset{[III]}{f_{0s}\boldsymbol{v_d}.\boldsymbol{\nabla}\left[\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z}\right]} + \overset{[IV]}{\{\chi, f_{0s}\}}$$

We now express each term in its normalised from:

$$[I] = q_s v_\parallel J_0(Z)\frac{\partial f_{0s}}{\partial E}\frac{\partial A_\parallel}{\partial t} = -\mathsf{q_s}\cancel{q_{\rm ref}}\,\cancel{v_t}\sqrt{\frac{T_s}{m_s}}\mathsf{v}_\parallel J_0(Z)\frac{\mathsf{n_s}n_{\rm ref}\ f_0}{T_s\cancel{T_{\rm ref}}}\frac{v_t}{L_{\rm ref}}\frac{\rho_{\rm ref}}{L_{\rm ref}}\frac{\cancel{T_{\rm ref}}}{\cancel{q_{\rm ref}}\,\cancel{v_t}}\frac{\partial\mathsf{A}_\parallel}{\partial\mathsf{t}}$$

$$[I] = -\frac{\mathsf{n_s}n_{\rm ref}\ f_0\rho_{\rm ref}\ v_t}{L_{\rm ref}^2}\frac{\mathsf{q_s}}{\sqrt{T_s m_s}}\mathsf{v}_\parallel J_0(Z)\frac{\partial\mathsf{A}_\parallel}{\partial\mathsf{t}}$$

In evaluating $[II]$ and $[III]$ we use equation (41) and:

$$\frac{q_s}{T_s}\Phi J_0(Z) + \frac{m_s v_\perp^2}{T_s}\frac{B_\parallel}{B}\frac{J_1(Z)}{Z} = \frac{\rho_{\rm ref}}{L_{\rm ref}}\left[\frac{\mathsf{q_s}}{\mathsf{T_s}}\mathsf{\Phi}J_0(Z) + 2\mathsf{v}_\perp^2\mathsf{B}_\parallel\frac{J_1(Z)}{Z}\right]$$

to give:

$$
\begin{aligned}
[II] &= -\frac{\mathsf{n_s} n_{\mathrm{ref}}\; f_0 \rho_{\mathrm{ref}}\; v_t}{L_{\mathrm{ref}}^2}\sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \nabla_\parallel^{\mathrm{GS2}}\left[\frac{\mathsf{q_s}}{\mathsf{T_s}}\Phi J_0(Z) + 2\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right] \\
[III] &= -\frac{\mathsf{n_s} n_{\mathrm{ref}}\; f_0 \rho_{\mathrm{ref}}\; v_t}{L_{\mathrm{ref}}^2}\frac{\mathsf{T_s}}{\mathsf{q_s}}\boldsymbol{b}\times\left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}} B + \mathsf{v}_\parallel{}^2 \boldsymbol{b}.\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}}\boldsymbol{b}\right).\boldsymbol{\nabla}_\perp^{\mathrm{GS2}}\left[\frac{\mathsf{q_s}}{\mathsf{T_s}}\Phi J_0(Z) + 2\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right]
\end{aligned}
$$

Now we evaluate the linear drive term using equation (42):

$$
\begin{aligned}
[IV] &= \{\chi, f_{0s}\} \\
&= \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{T_{\mathrm{ref}}}{q_{\mathrm{ref}}}\boldsymbol{\nabla}_\perp\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\frac{\times\boldsymbol{b}.\boldsymbol{\nabla} f_{0s}}{B} \\
&= in_0\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{T_{\mathrm{ref}}}{q_{\mathrm{ref}}}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\frac{\boldsymbol{\nabla}\alpha\times\boldsymbol{b}.\boldsymbol{\nabla}\psi}{B}\partial f_{0s}\psi \;\; \text{using } \Phi = \Phi(\theta)e^{in_0(\alpha+q\theta_0)} \\
&= -in_0\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{T_{\mathrm{ref}}}{q_{\mathrm{ref}}}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\partial f_{0s}\psi \;\; \text{since } \boldsymbol{\nabla}\alpha\times\boldsymbol{\nabla}\psi.\boldsymbol{b} = B \\
&= -i\left(\frac{n_0\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\partial\rho_n\psi_n\right)\frac{T_{\mathrm{ref}}}{q_{\mathrm{ref}}\; B_{\mathrm{ref}}\; L_{\mathrm{ref}}^2}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\partial f_{0s}\rho_n \quad \text{using } \partial f_{0s}\psi = \frac{1}{B_{\mathrm{ref}}\; L_{\mathrm{ref}}^2}\partial\rho_n\psi_n\partial f_{0s}\rho_n \\
&= -i\frac{1}{2}\left(\frac{n_0\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\partial\rho_n\psi_n\right)\frac{\rho_{\mathrm{ref}}\; v_t}{L_{\mathrm{ref}}^2}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\partial f_{0s}\rho_n \quad \text{using } \frac{T_{\mathrm{ref}}}{q_{\mathrm{ref}}\; B_{\mathrm{ref}}} = \frac{1}{2}\rho_{\mathrm{ref}}\; v_t \\
&= -i\frac{1}{2}\mathsf{k_y}\frac{\rho_{\mathrm{ref}}\; v_t}{L_{\mathrm{ref}}^2}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\partial f_{0s}\rho_n \quad \text{using } \mathsf{k_y} = \frac{n_0\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\partial\rho_n\psi_n.
\end{aligned}
$$

It is helpful to note that:

$$
\partial f_{0s}\rho_n = \left(\frac{\mathsf{n_s}'}{\mathsf{n_s}} + \frac{\mathsf{T_s}'}{\mathsf{T_s}}\left(\mathsf{v}^2 - \frac{3}{2}\right)\right)\mathsf{n_s} n_{\mathrm{ref}}\; f_0
$$

so we have:

$$
[IV] = i\mathsf{k_y}\frac{\mathsf{n_s} n_{\mathrm{ref}}\; f_0 \rho_{\mathrm{ref}}\; v_t}{2L_{\mathrm{ref}}^2}\left(\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right)\left(-\frac{\mathsf{n_s}'}{\mathsf{n_s}} - \frac{\mathsf{T_s}'}{\mathsf{T_s}}\left(\mathsf{v}^2 - \frac{3}{2}\right)\right)
$$

GS2 solves the gyrokinetic equation (29), normalised by the common factor $\frac{\mathsf{n_s} n_{\mathrm{ref}}\; f_0 \rho_{\mathrm{ref}}\; v_t}{L_{\mathrm{ref}}^2}$. To summarise, we can now express all the source terms on the RHS of the normalised GKE in terms of GS2 normalisations:

$$
[I] = -\frac{\mathsf{q_s}}{\sqrt{T_s m_s}}\mathsf{v}_\parallel J_0(Z)\frac{\partial \mathsf{A}_\parallel}{\partial \mathsf{t}} \tag{53}
$$

$$
[II] = -\frac{\mathsf{q_s}}{\sqrt{\mathsf{m_s}\mathsf{T_s}}}\mathsf{v}_\parallel \nabla_\parallel^{\mathrm{GS2}}\left[\Phi J_0(Z) + 2\frac{\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right] \tag{54}
$$

$$
[III] = -\boldsymbol{b}\times\left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}} B + \mathsf{v}_\parallel{}^2 \boldsymbol{b}.\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}}\boldsymbol{b}\right).\boldsymbol{\nabla}_\perp^{\mathrm{GS2}}\left[\Phi J_0(Z) + 2\frac{\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right] \tag{55}
$$

$$
[IV] = i\frac{\mathsf{k_y}}{2}\left(-\frac{\mathsf{n_s}'}{\mathsf{n_s}} - \frac{\mathsf{T_s}'}{\mathsf{T_s}}\left(\mathsf{v}^2 - \frac{3}{2}\right)\right)\left(\Phi J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z} - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z)\right) \tag{56}
$$

The sum of these terms is computed in GS2 subroutine set_source, and it only remains to express these quantities in terms of the variables actually used in GS2.

Here are some comments from the gs2 source code to document some of the key variables.

Listing 1: Comments documenting some key GS2 variables that set source.

```
! phigavg  = phi J0 + 2 T_s/q_s . vperp^2 bpar/bmag J1/Z
! apargavg = apar J0                          (decentered in t)
! NB apargavg and phigavg combine to give the GK EM potential chi
! phigavg - apargavg*vpa(:,isgn,iglo)*spec(is)%stm = chi
! phi_p = 2 phigavg                       .... (roughly!)
! phi_m = d/dtheta (phigavg)*DTHETA
! apar_p = 2 apargavg
! apar_m = 2 vpa d/dt (J0(Z) apar)*DELT
! => phi_p - apar_p*vpa(:,isgn,iglo)*spec(is)%stm = 2 chi   .... (roughly!)
! vparterm = -2.0*vpar (IN ABSENCE OF LOWFLOW TERMS)
! wdfac = wdrift + wcoriolis/spec(is)%stm (IN ABSENCE OF LOWFLOW TERMS)
! wstarfac = wstar  (IN ABSENCE OF LOWFLOW TERMS)
! vpar = q_s/sqrt{T_s m_s} (v_||^GS2). \gradpar(theta)/DTHETA . DELT (centred)
! wdrift =    q_s/T_s  v_d.\grad_perp . DELT
! wcoriolis = q_s/T_s  v_C.\grad_perp . DELT
! source      appears to contain following physical terms
!   -2q_s/T_s v||.grad(J0 phi + 2 vperp^2 bpar/bmag J1/Z T_s/q_s).delt
!   -2d/dt(q v|| J0 apar / T).delt
!   +hyperviscosity
!   -2 v_d.\grad_perp (q J0 phi/T + 2 vperp^2 bpar/bmag J1/Z).delt
!   -coriolis terms
!   2{\chi,f_{0s}}  (allowing for sheared flow)
```

In GS2 the source term on the RHS of the GKE is multiplied by the time-step and stored in variable **source**, which gets set as follows:

```
 source(ig) = anon(ie)*(vparterm(ig,isgn,iglo)*phi_m &
                          -spec(is)%zstm*vpac(ig,isgn,iglo) &
                   *((aj0(ig+1,iglo) + aj0(ig,iglo))*0.5*apar_m  &
                          + D_res(it,ik)*apar_p) &
                          -zi*wdfac(ig,isgn,iglo)*phi_p) &
              + zi*(wstarfac(ig,isgn,iglo) &
              +vpac(ig,isgn,iglo)*code_dt*wunits(ik)*ufac(ie,is) &
              -2.0*omprimfac*vpac(ig,isgn,iglo)*code_dt*wunits(ik)*g_exb*itor_over_B(ig)/spec(is)%stm) &
              *(phi_p - apar_p*spec(is)%stm*vpac(ig,isgn,iglo))
```

Greyed out terms do not appear in equation (29), but can be added as auxiliary terms to include: point of inflection in sheared equilibrium flow, hyperviscosity, and parallel sheared flows arising from toroidal equilibrium flows.

Quantity	Normalised Physical Quantities in GS2	Comments
Normalisation Quantities*	n_{ref}, T_{ref}, m_{ref}, q_{ref}	reference n, T, m and charge q
	B_{ref}, L_{ref}	reference equ'm B and length L
Thermal Velocity[1]	$v_t = \sqrt{\frac{2T_{\mathrm{ref}}}{m_{\mathrm{ref}}}}$	
Species Thermal Velocity	$v_{ts} = \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}v_t$	
Equilibrium Dist Fn	$f_{0s} = n_s f_0$, Maxwellian f_0, $\int d^3v f_0 = 1$	
Equilibrium Quantities	$\mathsf{n_s} = \frac{n_s}{n_{\mathrm{ref}}}$, $\mathsf{T_s} = \frac{T_s}{T_{\mathrm{ref}}}$, $\mathsf{m_s} = \frac{m_s}{m_{\mathrm{ref}}}$	
Equilibrium B	$\mathsf{B} = \frac{B}{B_{\mathrm{ref}}}$	
Equilibrium Gradients	$\boldsymbol{\nabla}_{\mathsf{eq}}^{\mathrm{GS2}} = L_{\mathrm{ref}}\ \boldsymbol{\nabla}$	
Velocity Variable	$\mathsf{v} = \frac{v}{v_{ts}}$	velocity variable for species s
Energy	$\mathsf{E} = \frac{v^2}{v_{ts}^2} = \mathsf{v}^2$	
Pitch Angle	$\lambda = \frac{\mathsf{v_\perp}^2}{\mathsf{Bv}^2}$	$\lambda = \frac{\mu}{E}$
Larmor Radius Quantities	$\rho_{\mathrm{ref}}\ = \frac{m_{\mathrm{ref}}\ v_t}{q_{\mathrm{ref}}\ B_{\mathrm{ref}}}$	normalises $\perp$ lengths
Beta	$\beta = \frac{2\mu_0 n_{\mathrm{ref}}\ T_{\mathrm{ref}}}{B_{\mathrm{ref}}^2}$	in GS2 field equations
Time	$\mathsf{t} = \frac{tv_t}{L_{\mathrm{ref}}}$	
Frequency	$\omega = \frac{\omega L_{\mathrm{ref}}}{v_t}$	
Parallel Gradients	$\nabla_{\parallel}^{\mathrm{GS2}} = \nabla_{\parallel} L_{\mathrm{ref}}$	
Perpendicular Gradients	$\boldsymbol{\nabla}_{\perp}^{\mathrm{GS2}} = \rho_{\mathrm{ref}}\ \boldsymbol{\nabla}_{\perp}$, or $\mathsf{k}_\perp = k_\perp \rho_{\mathrm{ref}}$	
k_y	$\mathsf{k_y} = \frac{n_0 \rho_{\mathrm{ref}}}{L_{\mathrm{ref}}} \partial \rho_n \psi_n$	n_0 is toroidal mode number, ρ_n is radial flux label for equ'm derivatives $\psi_n = \frac{\psi}{B_{\mathrm{ref}}\ L_{\mathrm{ref}}^2}$ is normalised poloidal flux
Perturbed Φ	$\mathsf{\Phi} = \frac{q_{\mathrm{ref}}\ \Phi}{T_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}$	
Adiabatic $\mathcal{F}_m$	$\mathsf{F_m} = \left\vert \begin{array}{ll} \frac{q_{\mathrm{ref}}\ \langle\Phi\rangle}{T_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}, \text{ or } 0 & \text{for } k_y = 0 \\ 0 & \text{for } k_y \neq 0 \end{array} \right.$	NB $\mathcal{F}_m$ is depends on adiabatic model
Perturbed $B_{\parallel}$	$\mathsf{B}_{\parallel} = \frac{B_{\parallel}}{B} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}} \neq \frac{B_{\parallel}}{B_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}$	
Perturbed $A_{\parallel}$	$\mathsf{A}_{\parallel} = \frac{q_{\mathrm{ref}}\ v_t A_{\parallel}}{T_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}} = \frac{2A_{\parallel}}{\rho_{\mathrm{ref}}\ B_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}$	follows from $\mathsf{vA}_{\parallel} = \sqrt{\frac{T_s}{m_s}} \frac{v}{v_{ts}} \frac{q_{\mathrm{ref}}\ v_t A_{\parallel}}{T_{\mathrm{ref}}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}$
Perturbed Dist Fn	$\mathsf{h_s} = \frac{h_s}{f_{0s}} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}} = \frac{h_s}{\mathsf{n_s} n_{\mathrm{ref}}\ f_0} \frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}$	$\mathsf{h_s}$ stored in GS2 variable g

Table 3: All GS2 variables are normalised dimensionless quantities, and the general normalisation rules are given in the table. * The dimensional normalisation parameters do not appear in the code. [1] Internally GS2 always uses $v_t = \sqrt{2T_{\mathrm{ref}}/m_{\mathrm{ref}}}$, but there is a (potentially confusing) option to allow output times and frequencies to be normalised using $v_t = \sqrt{T_{\mathrm{ref}}/m_{\mathrm{ref}}}$.

Species Summed Velocity Integrals	Variable Name
$\frac{\mathsf{n_s}\mathsf{q_s}^2}{\mathsf{T_s}} \int d^3v\ f_0\ \left(1 - J_0^2(Z_s)\right) + \frac{\mathsf{n_a}\mathsf{q_a}^2}{\mathsf{T_a}}$	GAMTOT
$\mathsf{n_s}\mathsf{q_s} \int d^3v\ f_0 \frac{2\mathsf{v}_\perp{}^2 J_0(Z_s) J_1(Z_s)}{Z_s}$	GAMTOT1
$\mathsf{n_s}\mathsf{T_s} \int d^3v\ f_0 \frac{2\mathsf{v}_\perp{}^4 J_1^2(Z_s)}{Z_s^2}$	GAMTOT2
$\frac{\mathrm{GAMTOT} - \frac{\mathsf{n_a}\mathsf{q_a}^2}{\mathsf{T_a}}}{\mathrm{GAMTOT}}$	GAMTOT3
$\mathsf{n_s}\mathsf{q_s} \int d^3v\ f_0\ J_0(Z_s)\mathsf{h_s}$	ANTOT
$2\beta\mathsf{n_s}\mathsf{q_s}\sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}} \int d^3v\ f_0\ \mathsf{v}_\parallel J_0(Z_s)\mathsf{h_s}$	ANTOTA
$\mathsf{n_s}\mathsf{T_s} \int d^3v\ f_0 \frac{\mathsf{v}_\perp{}^2 J_1(Z_s)}{Z_s}\mathsf{h_s}$	ANTOTP

Table 4: These velocity space integrals are calculated in GS2 routines in dist_fn.f90. GAMTOT, GAMTOT1, GAMTOT2 (and GAMTOT3 if there are adiabatic species) are computed at initialisation in routine init_fieldeq. ANTOT, ANTOTA, and ANTOTP involve integrals over the evolving perturbed distribution function, and are computed in routine getan. Note that the GS2 variable AJ1 $= J_1(Z_s)/Z_s$.

Subroutine	Comment
init_fieldeq	computes equilibrium velocity space integrals GAMTOT, GAMTOT1, GAMTOT2
getan	computes velocity space integrals over $\mathsf{h_s}$: ANTOT, ANTOTA, ANTOTP
get_init_field	computes the fields, corrected by CMR for nonuniform B August 2011
getfieldeq2	another routine to compute fields, inconsistent with get_init_field and with (47-49) resolved by CMR/PJK

Table 5: Key subroutines in dist_fn.f90 for computing the fields in GS2. The closest to being fully consistent is get_init_field. Note that getfieldeq2 is never used, unless explicit fields are selected, and this is NOT recommended. Maybe this is why!

RHS of GKE (29)	Corresponding Term in GS2 variable source
$[I] = -\frac{\mathsf{q_s}}{\sqrt{T_s m_s}}\mathsf{v}_\parallel J_0(Z)\frac{\partial \mathsf{A}_\parallel}{\partial \mathsf{t}}$	-spec(is)%zstm*vpac(ig,isgn,iglo) *((aj0(ig+1,iglo) + aj0(ig,iglo))*0.5*apar_m
$[II] = -\frac{\mathsf{q_s}}{\sqrt{\mathsf{m_s}\mathsf{T_s}}}\mathsf{v}_\parallel \nabla_\parallel^{\mathrm{GS2}}\left[\Phi J_0(Z) + 2\frac{\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right]$	vparterm(ig,isgn,iglo)*phi_m
$[III] = -\boldsymbol{b} \times \left(\frac{\mathsf{v}_\perp^2}{2B}\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}} B + \mathsf{v}_\parallel{}^2 \boldsymbol{b}.\boldsymbol{\nabla}_{\mathrm{eq}}^{\mathrm{GS2}}\boldsymbol{b}\right).\boldsymbol{\nabla}_\perp^{\mathrm{GS2}}$ $\left[\Phi J_0(Z) + 2\frac{\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right]$	-zi*wdfac(ig,isgn,iglo)*phi_p
$[IV] = i\frac{\mathsf{k_y}}{2}\left(-\frac{\mathsf{n_s}'}{\mathsf{n_s}} - \frac{\mathsf{T_s}'}{\mathsf{T_s}}\left(\mathsf{v}^2 - \frac{3}{2}\right)\right)$ $\left[\Phi J_0(Z) - \sqrt{\frac{\mathsf{T_s}}{\mathsf{m_s}}}\mathsf{v}_\parallel \mathsf{A}_\parallel J_0(Z) + \frac{2\mathsf{T_s}}{\mathsf{q_s}}\mathsf{v}_\perp^2 \mathsf{B}_\parallel \frac{J_1(Z)}{Z}\right]$	zi*wstarfac(ig,isgn,iglo)* (phi_p - apar_p*spec(is)%stm*vpac(ig,isgn,iglo))

Table 6: Specification of GKE source terms in GS2 subroutine set_source. The GS2 variable source is defined as: source = 2.RHS.code_dt.

A.5 Gyrokinetic Nonlinear Transport Fluxes

Gyrokinetic transport fluxes appear at second order in ρ_*, and can be determined by taking appropriate velocity space moments of f_{1s} multiplied by the perturbed velocity. As an example the particle flux is given by:

$$\mathbf{\Gamma_s} = \int d^3v \frac{\boldsymbol{b} \times \boldsymbol{\nabla}_\perp \chi}{B} f_{1s} \tag{57}$$

Only the non-adiabatic part of f_{1s} contributes, and integration over gyrophase angle gives:

$$\mathbf{\Gamma_s} = \int d^2v \frac{\boldsymbol{b} \times \boldsymbol{\nabla}_\perp \chi}{B} g_s J_0(Z).$$

After performing the appropriate velocity space, a contribution to a GK flux $\boldsymbol{F}_s$ has the general form:

$$\boldsymbol{F}_s = \frac{\boldsymbol{b} \times \boldsymbol{\nabla}_\perp \mathcal{F}}{B} \mathcal{M}_s \tag{58}$$

where $\mathcal{F}$ is a field quantity from χ and $\mathcal{M}_s$ is a velocity moment of g_s. Now we compute the radial component of a flux, with perturbations varying as $\tilde{A} = \tilde{A}(\theta) e^{in_0(\alpha + q\theta_0)}$:

$$\begin{aligned}
\boldsymbol{F}_s.\boldsymbol{\nabla}\psi &= \frac{\boldsymbol{b} \times \boldsymbol{\nabla}_\perp \mathcal{F}.\boldsymbol{\nabla}\psi}{B} \mathcal{M}_s \\
&= in_0 \frac{\boldsymbol{b} \times \boldsymbol{\nabla}\alpha.\boldsymbol{\nabla}\psi}{B} \mathcal{F}\mathcal{M}_s \\
&= in_0 \mathcal{F}\mathcal{M}_s \quad \text{since } \boldsymbol{\nabla}\alpha \times \boldsymbol{\nabla}\psi.\boldsymbol{b} = B \\
\Rightarrow \boldsymbol{F}_s.\boldsymbol{\nabla}\rho_n. &= in_0 \frac{\mathcal{F}\mathcal{M}_s}{\partial\psi\rho_n} \\
&= in_0 \frac{\mathcal{F}\mathcal{M}_s}{B_{\text{ref}}\ L_{\text{ref}}^2\ \partial\psi_n\rho_n} \quad \text{using } \psi_n = B_{\text{ref}}\ L_{\text{ref}}^2 \\
&= i \frac{n_0\rho_{\text{ref}}}{L_{\text{ref}}\ \partial\psi_n\rho_n} \frac{\mathcal{F}\mathcal{M}_s}{B_{\text{ref}}\ \rho_{\text{ref}}\ L_{\text{ref}}} \\
&= i\mathsf{k_y} \frac{\mathcal{F}\mathcal{M}_s}{B_{\text{ref}}\ \rho_{\text{ref}}\ L_{\text{ref}}} \quad \text{using } \mathsf{k_y} = \frac{n_0\rho_{\text{ref}}}{L_{\text{ref}}\ \partial\psi_n\rho_n} \\
&= i\mathsf{k_y} \frac{\frac{\rho_{\text{ref}}^2}{L_{\text{ref}}^2} \frac{T_{\text{ref}}\ \mathcal{M}_{\text{ref}}}{q_{\text{ref}}} \mathcal{F}^{\text{gs2}} \mathcal{M}_s^{\text{gs2}}}{B_{\text{ref}}\ \rho_{\text{ref}}\ L_{\text{ref}}} \quad \text{using } \mathcal{F} = \frac{\rho_{\text{ref}}}{L_{\text{ref}}} \frac{T_{\text{ref}}}{q_{\text{ref}}} \mathcal{F}^{\text{gs2}} \text{ and } \mathcal{M}_s = \frac{\rho_{\text{ref}}}{L_{\text{ref}}} \mathcal{M}_{\text{ref}}\ \mathcal{M}_s^{\text{gs2}} \\
&= \frac{\rho_{\text{ref}}^2\ v_{t\text{ref}}\ \mathcal{M}_{\text{ref}}}{L_{\text{ref}}^3} i\mathsf{k_y} \frac{\mathcal{F}^{\text{gs2}} \mathcal{M}_s^{\text{gs2}}}{2} \quad \text{using } T_{\text{ref}} = \frac{1}{2} m_{\text{ref}}\ v_{t\text{ref}}^2 \text{ and } \rho_{\text{ref}} = \frac{m_{\text{ref}}\ v_{t\text{ref}}}{q_{\text{ref}}\ B_{\text{ref}}}
\end{aligned}$$

1D transport equations require the integrated flux across an equilibrium surface labelled by ρ_n:

$$\begin{aligned}
F_s(\psi) &= \int_{S(\rho_n)} \boldsymbol{F}_s.\boldsymbol{dS} = \int_{S(\rho_n)} \boldsymbol{F}_s.\boldsymbol{\nabla}\rho_n \frac{dS}{|\boldsymbol{\nabla}\rho_n|} = \int_{\Delta V(\rho_n)} \boldsymbol{F}_s.\boldsymbol{\nabla}\rho_n d^3x \\
\Rightarrow F_s(\psi) &= V'(\rho_n)\ \langle \boldsymbol{F}_s.\boldsymbol{\nabla}\rho_n \rangle
\end{aligned}$$

where the angle brackets denote the flux surface average operator, defined:

$$\langle A \rangle = \frac{\int_{\Delta V(\rho_n)} A d^3x}{\int_{\Delta V(\rho_n)} d^3x}$$

and $V'(\rho_n) = \int_{\Delta V(\rho_n)} d^3x$. In ballooning coordinates the volume element is given by:

$$d^3x = \frac{d\theta d\alpha d\psi}{\boldsymbol{\nabla}\alpha \times \boldsymbol{\nabla}\psi.\boldsymbol{\nabla}\theta} = \frac{d\theta d\alpha d\psi}{\boldsymbol{\nabla}\alpha \times \boldsymbol{\nabla}\psi.\boldsymbol{b}|\nabla_\parallel\theta|} = \frac{d\theta d\alpha d\psi}{B|\nabla_\parallel\theta|}$$

so that:

$$\langle \boldsymbol{F}_s.\boldsymbol{\nabla}\rho_n \rangle = \frac{\rho_{\text{ref}}^2\ v_{t\text{ref}}\ \mathcal{M}_{\text{ref}}}{L_{\text{ref}}^3} i\mathsf{k_y} \frac{\int_{\Delta V(\rho_n)} \frac{d\theta d\alpha d\psi}{B|\nabla_\parallel\theta|} \frac{\mathcal{F}^{\text{gs2}} \mathcal{M}_s^{\text{gs2}}}{2}}{\int_{\Delta V(\rho_n)} \frac{d\theta d\alpha d\psi}{B|\nabla_\parallel\theta|}}$$

Perturbations are periodic in α and ψ, so only the term constant in α and ψ contributes to the numerator. Integrations over α and ψ cancel. We are interested only in the real part, and have:

$$
\begin{aligned}
\langle \boldsymbol{F}_s . \boldsymbol{\nabla} \rho_n \rangle &= \frac{\rho_{\mathrm{ref}}^2 \, v_{t\mathrm{ref}} \, \mathcal{M}_{\mathrm{ref}}}{L_{\mathrm{ref}}^3} \frac{\int \frac{d\theta}{B|\nabla_\parallel \theta|} \frac{\mathrm{Re}\left(i\mathsf{k_y} \mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}}_{s,-ky} \right)}{2}}{\int \frac{d\theta}{B|\nabla_\parallel \theta|}} \\
&= \frac{\rho_{\mathrm{ref}}^2 \, v_{t\mathrm{ref}} \, \mathcal{M}_{\mathrm{ref}}}{L_{\mathrm{ref}}^3} \frac{\int \frac{d\theta}{B|\nabla_\parallel \theta|} \frac{\mathrm{Re}\left(i\mathsf{k_y} \mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}\,*}_{s,ky} \right)}{2}}{\int \frac{d\theta}{B|\nabla_\parallel \theta|}} \quad \text{using } A_{-ky} = A^*_{ky} \text{ for real } A
\end{aligned}
$$

Now:

$$
\begin{aligned}
\mathrm{Re}\left(i\mathsf{k_y} \mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}\,*}_{s,ky} \right) &= \mathrm{Re}\left(i\mathsf{k_y} \left(\mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}\,*}_{s,ky} - \mathcal{F}^{\mathrm{gs2}\,*}_{ky} \mathcal{M}^{\mathrm{gs2}}_{s,ky} \right) \right) \quad \text{including contribution from } -k_y \\
&= 2\mathrm{Re}\left(i\mathsf{k_y} \mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}\,*}_{s,ky} \right) \quad \text{CMR : possible sign ambiguity}
\end{aligned}
$$

Therefore we obtain:

$$
\langle \boldsymbol{F}_s . \boldsymbol{\nabla} \rho_n \rangle = \frac{\rho_{\mathrm{ref}}^2 \, v_{t\mathrm{ref}} \, \mathcal{M}_{\mathrm{ref}}}{L_{\mathrm{ref}}^3} \frac{\int \frac{d\theta}{B|\nabla_\parallel \theta|} \mathrm{Re}\left(i\mathsf{k_y} \mathcal{F}^{\mathrm{gs2}}_{ky} \mathcal{M}^{\mathrm{gs2}\,*}_{s,ky} \right)}{\int \frac{d\theta}{B|\nabla_\parallel \theta|}}
$$

References

[1] T. M. Antonsen and B. Lane, Phys. Fluids **23**, 1205 (1980)