Journal of Computational Physics 391 (2019) 365–380

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

stella: An operator-split, implicit–explicit δf-gyrokinetic code for general magnetic field configurations

M. Barnes [a,b,*], F.I. Parra [a,b], M. Landreman [c]

[a] Rudolf Peierls Centre for Theoretical Physics, University of Oxford, Oxford OX1 3PU, United Kingdom
[b] Culham Centre for Fusion Energy, Culham Science Centre, Abingdon OX14 3DB, United Kingdom
[c] Institute for Research in Electronics and Applied Physics, University of Maryland, College Park, MD 20742, USA

ARTICLE INFO

Article history:
Received 6 June 2018
Received in revised form 9 January 2019
Accepted 11 January 2019
Available online 7 February 2019

Keywords:
Gyrokinetics
Turbulence
Transport
Stellarator
Magnetic confinement fusion

ABSTRACT

Here we present details of an operator-split, implicit–explicit numerical scheme for the solution of the gyrokinetic-Poisson system of equations in the local limit. This scheme has been implemented in a new code called stella, which is capable of evolving electrostatic fluctuations with full kinetic electron effects and an arbitrary number of ion species in general magnetic geometry. We demonstrate the advantages of this mixed approach over a fully explicit treatment and provide linear and nonlinear benchmark comparisons for both axisymmetric and non-axisymmetric magnetic equilibria.

1. Introduction

The turbulent transport of particles, momentum and energy places a fundamental constraint on the confinement – and thus performance – of magnetic confinement fusion (MCF) plasmas. This turbulence is challenging to simulate for a number of reasons: the collisional mean free path in MCF plasmas is often larger than the system size, necessitating a kinetic treatment; the presence of a strong mean magnetic field makes the turbulence highly anisotropic; and the characteristic space–time scales of the turbulence are much smaller than the space–time scales associated with the mean density, flow and temperature. On the face of it, one must thus resolve six-dimensional phase space dynamics involving multiple space–time scales spanning several orders of magnitude.

However, by exploiting the anisotropy of the turbulence and scale separation in space and time, it is possible to reduce the complexity of the problem considerably. This is the approach taken by δf-gyrokinetics [1,2], which we describe in detail in Sec. 2. In brief, it eliminates the fast gyro-motion time scale and the gyro-angle phase space variable, and it separates the space–time scales of mean and fluctuating quantities. The development of gyrokinetics (and the gyrofluid models derived from it) and its subsequent numerical implementation in a wide range of codes [3–12] has facilitated a leap forward in our ability to accurately model, predict and understand turbulent transport in MCF plasmas. Given the proliferation of gyrokinetic codes and their growing success in describing experimental behavior, it is worth considering if there is a need for yet another gyrokinetic code.

Most of the existing δf-gyrokinetic codes have been developed to simulate tokamak plasmas, with only a handful [13–17] capable of simulating non-axisymmetric magnetic field configurations. Indeed, there is a relative paucity of sim-

* Corresponding author at: Rudolf Peierls Centre for Theoretical Physics, University of Oxford, Oxford OX1 3PU, United Kingdom.
E-mail address: michael.barnes@physics.ox.ac.uk (M. Barnes).

https://doi.org/10.1016/j.jcp.2019.01.025
0021-9991/

ulations for stellarators, and most of them are linear simulations in the local (or flux-tube) limit with Boltzmann electrons. As it is beneficial to have a diversity of numerical approaches to the same class of problems, we have developed a new code `stella` with the goal of enabling routine and efficient simulation of turbulence in stellarators. The numerical scheme employed in `stella` is distinguished from other gyrokinetic codes through its use of an operator-split, implicit–explicit time advance scheme with strong-stability-preserving methods to maximize the allowable time step size for both linear and nonlinear calculations. There are only a handful of kinetic/gyrokinetic codes that employ implicit–explicit time advance methods (cf. [4,8,18,19]); as such, `stella` is expected to be a useful tool for simulations of turbulence in both tokamaks and stellarators.

The flux tube approach is very efficient for simulating plasma turbulence in an axisymmetric magnetic field because a single flux tube spanning a 2π domain in poloidal angle effectively samples an entire flux surface. For a non-axisymmetric confining field, this is not the case: Because a flux tube is asymptotically small in the local limit, it in principle must extend infinitely far along a magnetic field line as it ergodically samples an entire flux surface. As this is not feasible, one must either be content with sampling a fraction of a flux surface or one must go beyond the flux tube approach to simulate a flux annulus encompassing a full flux surface. To our knowledge only a few codes [20,15,17] currently allow for the simulation of an entire flux surface, and results of this type are few and far between. While the version of `stella` documented here employs the flux tube approximation, the numerical scheme has been formulated with the aim of extending the code to treat the full flux annulus. This is discussed in more detail when the algorithm is introduced in Sec. 5.

The paper is organized as follows: In Sec. 2 we describe the gyrokinetic model and state the governing equations. We then give an overview of the coordinates used in `stella` and the options available for specifying magnetic geometry in Sec. 3. The normalized simulation equations are provided in Sec. 4 before detailing the numerical scheme employed by `stella` in Sec. 5. We then compare simulation results from `stella` with those from the widely-benchmarked gyrokinetic code `GS2` in Sec. 6 before summarizing and discussing possibilities for future work on `stella` in Sec. 7.

2. Model equations

Derivations of the δf gyrokinetic model employed by `stella` are abundant in the plasma physics literature (cf. [1,2,21–23]) and so we provide only a brief overview of its orderings and assumptions here. The essence of gyrokinetics is an assumption that all dynamics of interest are slow compared to particle gyration about a mean magnetic field. This allows one to usefully split particle motion into a rapid, approximately circular orbit about the magnetic field and the movement of this orbit's centre, called the guiding centre; i.e., particle position $\mathbf{r}$ is given by $\mathbf{r} = \mathbf{R} + \boldsymbol{\rho}$, with $\mathbf{R}$ the guiding centre position, and $\boldsymbol{\rho} = \hat{\mathbf{b}} \times \mathbf{v}/\Omega$ the gyroradius vector. Here, $\hat{\mathbf{b}}$ is the unit vector pointing along the mean magnetic field, $\mathbf{v}$ is the particle velocity, and $\Omega = ZeB/mc$ is the frequency of gyration about the mean field, with Z charge number, e proton charge, B magnetic field strength, m particle mass, and c speed of light. After making this split, one averages over the rapid gyration to eliminate the gyration angle as a phase space variable. Gyrokinetics thus describes the motion of charged rings as they stream along the mean magnetic field and slowly drift across it.

We restrict our attention to plasmas with sub-sonic mean flows and electrostatic fluctuations so that to lowest order the electric field $\mathbf{E} = -\nabla\varphi$, with φ the fluctuating electrostatic potential. These constraints are not required by the gyrokinetic model and can thus be relaxed in future work. The use of δf gyrokinetics does rely on a separation of space–time scales between the plasma equilibrium and the turbulent fluctuations. In particular, the particle distribution function for species s, denoted f_s, is decomposed into a mean component, F_s, and a fluctuating component, δf_s, i.e., $f_s = F_s + \delta f_s$, and the following orderings are imposed:

$$\frac{\delta f_s}{f_s} \sim \frac{\omega_s}{\Omega_s} \sim \frac{\rho_s}{L} \sim \frac{k_\parallel}{k_\perp} \sim k_\parallel \rho_s \sim \frac{e\varphi}{T_s} \sim \epsilon \ll 1, \tag{1}$$

where ϵ is the fundamental ordering parameter in gyrokinetics, ω is a characteristic fluctuation frequency, $\rho = v_{\mathrm{th}}/\Omega$ is thermal gyroradius, $v_{\mathrm{th}} = \sqrt{2T/m}$ is thermal speed, T is temperature, L is a characteristic length associated with the equilibrium, and $k_\parallel$ and $k_\perp$ are characteristic fluctuation wave numbers along and across the mean magnetic field.

Upon gyro-averaging the Vlasov equation and applying the gyrokinetic ordering given above, one obtains the lowest-order, electrostatic gyrokinetic equation for the distribution of guiding centres $g_s(\mathbf{R}, v_\parallel, \mu, t) \doteq \langle \delta f \rangle_{\mathbf{R}}$:

$$\begin{aligned} &\frac{\partial g_s}{\partial t} + v_\parallel \hat{\mathbf{b}} \cdot \nabla z \left(\frac{\partial g_s}{\partial z} + \frac{Z_s e}{T_s} \frac{\partial \langle \varphi \rangle_{\mathbf{R}}}{\partial z} F_s \right) - \frac{\mu_s}{m_s} \hat{\mathbf{b}} \cdot \nabla B \frac{\partial g_s}{\partial v_\parallel} + \mathbf{v}_{Ms} \cdot \left(\nabla_\perp g_s + \frac{Z_s e}{T_s} \nabla_\perp \langle \varphi \rangle_{\mathbf{R}} F_s \right) \\ &\quad + \langle \mathbf{v}_E \rangle_{\mathbf{R}} \cdot \nabla_\perp g_s + \langle \mathbf{v}_E \rangle_{\mathbf{R}} \cdot \nabla \big|_E F_s = 0, \end{aligned} \tag{2}$$

where, unless noted otherwise, derivatives are taken at fixed guiding centre position $\mathbf{R}$, parallel speed $v_\parallel$, and magnetic moment $\mu = m_s v_\perp^2/2B$. Here, $E = m_s v^2/2$ is kinetic energy, t is time, z is a coordinate that measures distance along the magnetic field, F is the mean distribution function (taken here to be a Maxwellian in E), $\langle . \rangle_{\mathbf{R}}$ denotes a gyro-average at fixed $\mathbf{R}$, $\mathbf{v}_{Ms}$ is the drift velocity due to the magnetic field gradient and curvature, given by

$$\mathbf{v}_{Ms} = \frac{\hat{\mathbf{b}}}{\Omega_s} \times \left(\frac{v_\perp^2}{2} \frac{\nabla B}{B} + v_\parallel^2 \boldsymbol{\kappa} \right), \tag{3}$$

$\boldsymbol{\kappa} = \hat{\mathbf{b}} \cdot \nabla \hat{\mathbf{b}}$ is the curvature vector, and $\mathbf{v}_E$ contains the lowest order, fluctuating $\mathbf{E} \times \mathbf{B}$ drift velocity, given by

$$\mathbf{v}_E = \frac{c}{B} \hat{\mathbf{b}} \times \nabla_\perp \varphi. \tag{4}$$

The system is closed by coupling to Poisson's equation. In the usual gyrokinetic ordering – in which the Debye length is taken to be much smaller than the electron gyroradius – this reduces to quasineutrality:

$$\sum_s Z_s \delta n_s = \sum_s Z_s \int d^3 v \left(g_s + \frac{Z_s e}{T_s} F_s \left(\langle \varphi \rangle_{\mathbf{R}} - \varphi \right) \right) = 0. \tag{5}$$

3. Coordinates and magnetic geometry

The coordinates used in `stella` are $(x, y, z, v_\parallel, \mu)$, with $v_\parallel = \hat{\mathbf{b}} \cdot \mathbf{v}$ the speed along the magnetic field, $\mu = m_s v_\perp^2 / 2B$ the magnetic moment, z a coordinate measuring distance along the magnetic field, and (x, y) are coordinates in the plane perpendicular to $\hat{\mathbf{b}}$. The magnetic field vector $\mathbf{B}$ is expressed

$$\mathbf{B} = \nabla \alpha \times \nabla \psi, \tag{6}$$

where α labels field lines and ψ labels flux surfaces. The coordinates (x, y) are related to ψ and α via

$$x = \frac{dx}{d\psi} (\psi - \psi_0) \tag{7}$$

and

$$y = \frac{dy}{d\alpha} (\alpha - \alpha_0), \tag{8}$$

with ψ_0 and α_0 the values of ψ and α at the centre of the simulation domain. Note that there is flexibility in defining x, y, and z within `stella`: details on the currently supported options are given in the subsection on magnetic geometry below.

3.1. Velocity space grids and integrals

In terms of our chosen $(v_\parallel, \mu, \sigma)$ coordinates, with σ the gyration angle, velocity space integrals are of the form

$$\int d^3 v \; f = \int_0^{2\pi} d\sigma \int_{-\infty}^{\infty} dv_\parallel \int_0^{\infty} d\mu \frac{B}{m} f \tag{9}$$

We truncate the $v_\parallel$ integral at the cutoff values $\pm v_{\parallel,c}$, with $v_{\parallel,c}$ chosen to ensure that the integrand is sufficiently small for $v_\parallel > v_{\parallel,c}$. The $v_\parallel$ grid points are then chosen to be equally spaced on the interval $[-v_{\parallel,c}, v_{\parallel,c}]$. The number of parallel velocities $N_{v_\parallel}$ is constrained in `stella` to be even so that $v_\parallel = 0$ is not included in the grid. This choice facilitates parallelization and avoids the need to apply a special treatment to phase space points where $v_\parallel = \hat{\mathbf{b}} \cdot \nabla B = 0$, which decouple from all other points for a collisionless plasma. The $v_\parallel$ integral is approximated numerically using an average of Simpson's 3/8 rule and composite Simpson's rule at the final four points at either end of the $v_\parallel$ domain and pure composite Simpson's rule at the interior points (cf. [24]).

The μ grid points are chosen according to Gauss–Laguerre quadrature (cf. [25]):

$$\begin{aligned} \int_0^{\infty} d\mu \frac{B}{m} f &= \int_0^{\infty} d \left(\frac{\mu B_0}{T} \right) \frac{TB}{mB_0} \exp(-\mu B_0 / T) f \exp(\mu B_0 / T) \\ &\approx \sum_{i=1}^{N_\mu} w_i f(\hat{\mu}_i) \exp(\hat{\mu}_i) \frac{TB}{mB_0}, \end{aligned} \tag{10}$$

with N_μ the number of μ grid points, $\hat{\mu} = \mu B_0 / T$, and B_0 is a free parameter. It is desirable to choose B_0 to be independent of z, as otherwise the physical μ grid would be z-dependent and would complicate computation of z derivatives at fixed μ; i.e.,

$$\left(\frac{\partial}{\partial z} \right)_\mu = \left(\frac{\partial}{\partial z} \right)_{\hat{\mu}} + \left(\frac{\partial \hat{\mu}}{\partial z} \right)_\mu \left(\frac{\partial}{\partial \hat{\mu}} \right)_z = \left(\frac{\partial}{\partial z} \right)_{\hat{\mu}} + \frac{\mu}{T} \frac{\partial B_0}{\partial z} \left(\frac{\partial}{\partial \hat{\mu}} \right)_z. \tag{11}$$

To satisfy the boundary condition $f(v_\perp \to \infty) \to 0$, we must choose our maximum $v_\perp$ at each z so that f evaluated there is approximately zero. Denoting $v_{\perp,c}$ as the smallest acceptable value satisfying $f(v_{\perp,c}) \approx 0$, we obtain the following constraint:

$$\hat{\mu}_{\max} \geq \frac{v_{\perp,c}^2}{v_{\mathrm{th},s}^2}\frac{B_0}{B_{\min}}, \tag{12}$$

where $B_{\min}$ is fixed by the magnetic geometry, $\hat{\mu}_{\max}$ is fixed by the choice of N_μ, and $v_{\perp,c}/v_{\mathrm{th},s}$ is an input parameter. To ameliorate the CFL constraint on the time step size, we want to minimize the largest value of $v_\perp$, and thus $\hat{\mu}_{\max}$, included in the simulation. Combined with the above inequality, we find

$$B_0 = B_{\min}\frac{v_{\mathrm{th},s}^2}{v_{\perp,c}^2}\hat{\mu}_{\max}. \tag{13}$$

3.2. Real space grids and boundary conditions

Periodic boundary conditions are enforced in x and y by expressing the guiding centre distribution g in terms of Fourier harmonics:

$$g(x,y,z,v_\parallel,\mu,t) \doteq \sum_{k_x,k_y} \hat{g}_{\mathbf{k}}(z,v_\parallel,\mu,t)\exp(ik_x x + ik_y y), \tag{14}$$

where $\hat{g}_{\mathbf{k}} = \hat{g}_{k_x,k_y}$. This is justified as long as $k_x \sim k_y \gg 1/L$. In this limit the turbulent fluctuations at separate ends of the (x,y) domain are decorrelated and can thus be treated as statistically periodic. This local, or flux tube, approximation is routinely used to model micro-instabilities and turbulence in axisymmetric magnetic field configurations and has been successfully validated across a range of experiments (cf. [26–31]). For non-axisymmetric field configurations, a single flux tube would in principle need to extend infinitely far along the magnetic field line as it ergodically samples a flux surface. A version of `stella` that simulates a flux annulus encompassing a full flux surface is currently in development to address this deficiency.

The grid in the parallel coordinate z is equally spaced, with the number of z points N_z forced to be odd in order to guarantee points at $z = \pm z_0$ and $z = 0$, with z_0 the boundary value for z. The boundary condition in z is a generalization of the 'twist and shift' boundary condition [32] in which different k_x values are coupled at the boundaries of the z-domain. For the sake of definiteness, we choose here $z = \zeta$ and $\alpha = \theta - \iota\zeta$, with θ and ζ straight-field-line poloidal and toroidal angles, respectively, and $\iota = \iota(\psi)$ the rotational transform. These choices for the z and α coordinates are used in `stella` only for stellarator simulations: The coordinates $z = \theta$ and $\alpha = \zeta - q\theta$ are used for all other magnetic field configurations, with $q = q(\psi)$ the safety factor. For an arbitrary fluctuating quantity A, physical periodicity dictates $A(x, y(\theta,z), z) = A(x, y(\theta, z + 2\pi p), z + 2\pi p)$, where $p = M/N_p$ with M any integer and N_p defined so that the magnetic geometry is periodic in ζ with period $2\pi/N_p$. In terms of the spectral representation, this parallel boundary condition becomes

$$\begin{aligned}&\sum_{k_x,k_y} \hat{A}_{k_x,k_y}(z)\exp\left(ik_y y(\theta,z) + ik_x x\right)\\ &\quad= \sum_{k_x,k_y} \hat{A}_{k_x,k_y}(z+2\pi p)\exp\left(ik_y y(\theta,z) + i\left(k_x - \delta k\right)x\right)\exp\left(-2\pi p i k_y \frac{dy}{d\alpha}\iota(\psi_0)\right),\end{aligned} \tag{15}$$

with ψ_0 the value of ψ on the flux surface of interest and $\delta k = 2\pi p k_y (dy/d\alpha)(d\psi/dx)(d\iota/d\psi)_{\psi_0}$. Orthogonality of the Fourier harmonics then implies that

$$\hat{A}_{k_x,k_y}(z) = \hat{A}_{k_x',k_y}(z+2\pi p)\exp\left(-2\pi p i k_y \frac{dy}{d\alpha}\iota(\psi_0)\right), \tag{16}$$

with $k_x' = k_x + \delta k$. An outgoing boundary condition is applied at the end of each set of connected 2π segments, with a zero incoming boundary condition on g.

For an axisymmetric confining field, the field period is 2π, making p an integer. The parallel boundary condition (16) (with $z \to \theta$ and $\iota \to q$) is then identical to the original 'twist and shift' boundary condition [32]. For a non-axisymmetric confining field, care must be taken when applying (16) to ensure that the turbulence is decorrelated at the ends of the domain in z – else the assumption of statistical periodicity in the (x,y) plane would be suspect. This should be achieved by choosing a sufficiently large value for M. Note that there are alternative and (likely) more efficient parallel boundary conditions that could be employed in the case of low magnetic shear stellarators [33]. These are not implemented in `stella` at present, but will be the subject of further `stella` development.

3.3. Magnetic geometry

With these coordinate choices, there are eight independent geometrical quantities appearing in the gyrokinetic equation: B, $\hat{\mathbf{b}}\cdot\nabla z$, $|\nabla x|^2$, $|\nabla y|^2$, $|\nabla x\cdot\nabla y|$, $(\hat{\mathbf{b}}\times\nabla B)\cdot\nabla y$, $(\hat{\mathbf{b}}\times\boldsymbol{\kappa})\cdot\nabla x$, and $(\hat{\mathbf{b}}\times\boldsymbol{\kappa})\cdot\nabla y$. There are currently two options

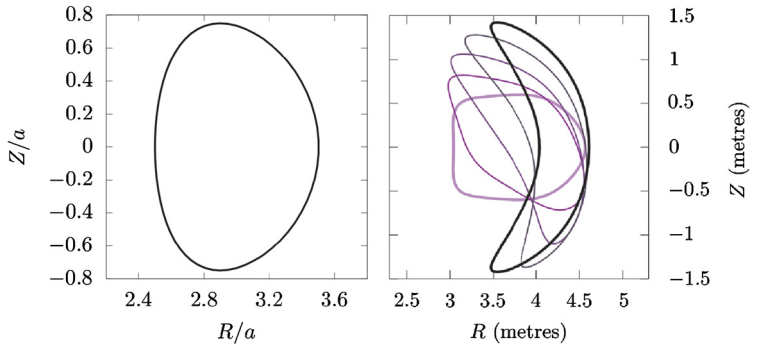

Fig. 1. (Left): Poloidal cut of a flux surface corresponding to a typical Miller local equilibrium, with inverse aspect ratio $R_0/a = 3$, normalized minor radius $r/a = 0.5$, elongation $\kappa = 1.5$ and triangularity $\delta = 0.2$. (Right): Poloidal cuts of a flux surface from NCSX design LI383. The flux surface corresponds to an enclosed normalized toroidal flux of 0.635, and the cuts are taken at five equally spaced locations in ζ, starting at $\zeta = 0$ (*thick black line*) and ending half a field period away at $\zeta = 1.047$ (*thin, light purple line*). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

in `stella` for obtaining these quantities. The first option is to use a magnetic equilibrium generated by `VMEC` [34,35]. A module in `stella` takes the `VMEC` output and computes all of the geometrical quantities needed to solve the gyrokinetic equation on a user-selected flux surface and field line. For this option, the x and y coordinates are chosen so that $dx/d\psi = -(\psi_t/|\psi_t|)\sqrt{\psi_{LCFS}/\psi_t}/aB_r$ and $dy/d\alpha = a\sqrt{\psi_t/\psi_{LCFS}}$, with a the effective minor radius computed by `VMEC`, $B_r = 2|\psi_{LCFS}|/a^2$, $\psi = -\psi_t$, ψ_t the enclosed toroidal flux divided by 2π, and ψ_{LCFS} its value at the outermost flux surface computed by `VMEC` (conventionally called the 'last closed flux surface'). As an illustrative example, we provide in Fig. 1 several poloidal cross sections of a single flux surface for NCSX design LI383. The equilibrium data was obtained from `VMEC` and used to benchmark `stella`, as discussed in Sec. 6.3.

The second option, valid only for axisymmetric magnetic field configurations, is to specify a set of Miller parameters that are used to construct a local solution to the Grad–Shafranov equation [36]. In brief, the cylindrical coordinates (R, Z) of the desired flux surface are assumed to be of the form

$$R(r, \vartheta) = R_0(r) + r\cos(\vartheta + \sin\vartheta \arcsin\delta(r)), \tag{17}$$

$$Z(r, \vartheta) = \kappa(r) r \sin(\vartheta), \tag{18}$$

where the flux label r is the half-diameter of the flux surface at the height of the magnetic axis, R_0 is the average of the minimum and maximum values of the major radius at the height of the magnetic axis, and ϑ is a poloidal angle. Note that ϑ is not in general a straight-field-line angle. With this assumed form for the desired flux surface (and for neighboring surfaces), all of the required geometric quantities can be calculated by providing: the flux surface location r, the local major radius R_0 and its derivative dR_0/dr, the local elongation κ and its derivative $d\kappa/dr$, the local triangularity δ and its derivative $d\delta/dr$, the local safety factor q and its derivative dq/dr, and the MHD α-parameter $(4\pi p_{\rm tot}/B_r^2)(d\ln p_{\rm tot}/dr)$, where $p_{\rm tot}$ is the species-summed plasma pressure. For this option, the x and y coordinates are chosen so that $dx/d\psi = q/rB_r$ and $dy/d\alpha = (d\psi/dr)/B_r$, with reference length a the half-diameter of the plasma volume at the height of the magnetic axis, B_r is the user-specified reference magnetic field strength, and ψ is the enclosed poloidal flux divided by 2π. An illustrative example of the flux surface shape corresponding to this Miller local parameterization is given in Fig. 1.

4. Simulation equations

The gyrokinetic equation solved in `stella` is obtained by taking the discrete Fourier transform $\mathcal{F}_{\mathbf{k}}$ of the gyrokinetic equation (2) in x and y and multiplying by the normalizing factor $(a^2/\rho_r v_{{\rm th},r})\exp(-v^2/v_{{\rm th},s}^2)/F_s$, with $v_{{\rm th},r} \doteq \sqrt{2T_r/m_r}$, $\rho_r \doteq v_{{\rm th},r}/\Omega_r$, and $\Omega_r \doteq eB_r/m_r c$. The reference mass m_r, density n_r, and temperature T_r are user-specified, while the reference length a and magnetic field strength B_r (different from B_0) are determined by the choice of magnetic geometry model as detailed in Subsection 3.3. With this choice of normalization, the gyrokinetic equation solved by `stella` is

$$\begin{aligned}\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial \tilde{t}} + \frac{v_{{\rm th},s}}{v_{{\rm th},r}}\tilde{v}_\parallel \hat{\mathbf{b}}\cdot\tilde{\nabla}z\left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial z} + \frac{Z_s}{\tilde{T}_s}\frac{\partial J_0(a_{\mathbf{k},s})\tilde{\varphi}_{\mathbf{k}}}{\partial z}\exp\left(-\tilde{v}_s^2\right)\right) - \frac{v_{{\rm th},s}}{v_{{\rm th},r}}\tilde{\mu}_s\hat{\mathbf{b}}\cdot\tilde{\nabla}\tilde{B}\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial \tilde{v}_\parallel} \\ + i\omega_{d,\mathbf{k},s}\left(\tilde{g}_{\mathbf{k},s} + \frac{Z_s}{\tilde{T}_s}J_0(a_{\mathbf{k},s})\tilde{\varphi}_{\mathbf{k}}\exp\left(-\tilde{v}_s^2\right)\right) + i\omega_{*,\mathbf{k},s}J_0(a_{\mathbf{k},s})\tilde{\varphi}_{\mathbf{k}} + \mathcal{N}_{\mathbf{k},s} = 0,\end{aligned} \tag{19}$$

where $\tilde{t} \doteq ta/v_{{\rm th},r}$, $\tilde{T}_s \doteq T_s/T_r$, $\tilde{v}_\parallel = v_\parallel/v_{{\rm th},s}$, $\tilde{\nabla} \doteq a\nabla$, J_0 is a Bessel function of the first kind, $a_{\mathbf{k},s} \doteq k_\perp v_\perp/\Omega_s$, $k_\perp^2 = k_x^2|\nabla x|^2 + k_y^2|\nabla y|^2 + 2k_x k_y \nabla x\cdot\nabla y$, $\tilde{\mu}_s \doteq \mu_s B_r/2T_s$, $\tilde{B} \doteq B/B_r$,

$$\tilde{g}_{\mathbf{k},s} \doteq \frac{\hat{g}_{\mathbf{k},s}}{F_s} \exp\left(-\tilde{v}_s^2\right) \frac{a}{\rho_r}, \tag{20}$$

$$\tilde{\varphi}_{\mathbf{k}} \doteq \frac{e\hat{\varphi}_{\mathbf{k}}}{T_r} \frac{a}{\rho_r}, \tag{21}$$

$$\omega_{*,\mathbf{k},s} \doteq \frac{k_y \rho_r}{2} a B_r \frac{dy}{d\alpha} \exp\left(-\tilde{v}_s^2\right) \frac{d \ln F_s}{d\psi}, \tag{22}$$

$$\mathcal{N}_k \doteq \frac{B_r}{2} \frac{dy}{d\alpha} \frac{dx}{d\psi} \mathcal{F}_{\mathbf{k}} \left[\mathcal{F}_{\mathbf{k}}^{-1}\left[ik_y \rho_r J_0(a_{\mathbf{k},s})\tilde{\varphi}_{\mathbf{k}}\right] \mathcal{F}_{\mathbf{k}}^{-1}\left[ik_x \rho_r \tilde{g}_{\mathbf{k},s}\right] - \mathcal{F}_{\mathbf{k}}^{-1}\left[ik_x \rho_r J_0(a_{\mathbf{k},s})\tilde{\varphi}_{\mathbf{k}}\right] \mathcal{F}_{\mathbf{k}}^{-1}\left[ik_y \rho_r \tilde{g}_{\mathbf{k},s}\right]\right], \tag{23}$$

and

$$\omega_{d,\mathbf{k},s} \doteq \frac{\tilde{T}_s}{Z_s \tilde{B}} \left(\tilde{v}_\parallel^2 \mathbf{v}_\kappa + \tilde{\mu}_s \mathbf{v}_{\nabla B}\right) \cdot \left(k_y \rho_r \nabla y + k_x \rho_r \nabla x\right), \tag{24}$$

with $\mathbf{v}_\kappa = \hat{\mathbf{b}} \times \left(\hat{\mathbf{b}} \cdot \tilde{\nabla}\hat{\mathbf{b}}\right)$ and $\mathbf{v}_{\nabla B} = \hat{\mathbf{b}} \times \tilde{\nabla}\tilde{B}$.

The normalized form of quasineutrality for Fourier component $\mathbf{k}$ is

$$\sum_s Z_s \frac{\delta n_{\mathbf{k},s}}{n_r} \frac{a}{\rho_r} = \sum_s Z_s \tilde{n}_s \left(\frac{2\tilde{B}}{\pi^{1/2}} \int_{-\infty}^{\infty} d\tilde{v}_\parallel \int_0^\infty d\tilde{\mu}_s J_0(a_{\mathbf{k},s}) \tilde{g}_{\mathbf{k},s} + \frac{Z_s}{\tilde{T}_s} \left(\Gamma_0(b_{\mathbf{k},s}) - 1\right) \tilde{\varphi}_{\mathbf{k}} \right) = 0, \tag{25}$$

with $\tilde{n}_s \doteq n_s/n_r$, $b_{\mathbf{k},s} \doteq k_\perp^2 \rho_s^2/2$, $\Gamma_0(b) \doteq \exp(-b) I_0(b)$, and I_0 is a modified Bessel function of the first kind.

5. Algorithm

When electron dynamics are retained in the gyrokinetic equation (19), the parallel streaming and acceleration terms are scaled up compared to all other terms by a factor of $v_{\mathrm{th},e}/v_{\mathrm{th},i} \sim 60$. This places a significant restriction on the time step size for explicit time advance schemes. The severity of this restriction becomes prohibitive at long wavelengths [37,8], since the electrostatic potential obtained from quasineutrality (25) is derived from a polarization density that vanishes at infinite wavelength. In the absence of electromagnetic effects, this leads to a discretization-dependent CFL condition that scales as either $1/k_\perp$ or $1/k_\perp^2$, as we demonstrate numerically in the next section.

In light of these considerations, it is desirable to treat the parallel streaming and acceleration terms implicitly. We achieve this without undue computational expense by employing operator splitting to separate the faster time scales associated with streaming and acceleration from the rest of the dynamics. We start by splitting the gyrokinetic equation into three pieces:

$$\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t} = \left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_1 + \left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_2 + \left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_3, \tag{26}$$

with

$$\left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_1 + i\omega_{d,\mathbf{k},s} \left(\tilde{g}_{\mathbf{k},s} + \frac{Z_s}{\tilde{T}_s} J_0(a_{\mathbf{k},s}) \tilde{\varphi}_{\mathbf{k}} \exp\left(-\tilde{v}_s^2\right)\right) + i\omega_{*,\mathbf{k},s} J_0(a_{\mathbf{k},s}) \tilde{\varphi}_{\mathbf{k}} + \mathcal{N}_{\mathbf{k},s} = 0, \tag{27}$$

$$\left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_2 - \frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}} \tilde{\mu}_s \hat{\mathbf{b}} \cdot \tilde{\nabla}\tilde{B} \frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial \tilde{v}_\parallel} = 0, \tag{28}$$

$$\left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial t}\right)_3 + \frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}} \tilde{v}_\parallel \hat{\mathbf{b}} \cdot \tilde{\nabla} z \left(\frac{\partial \tilde{g}_{\mathbf{k},s}}{\partial z} + \frac{Z_s}{\tilde{T}_s} \frac{\partial J_0(a_{\mathbf{k},s}) \tilde{\varphi}_{\mathbf{k}}}{\partial z} \exp\left(-\tilde{v}_s^2\right)\right) = 0. \tag{29}$$

Symbolically we can write

$$\frac{\partial \mathbf{g}}{\partial t} = \mathcal{A}[\mathbf{g}] + (\mathcal{B} + \mathcal{C})\,\mathbf{g} \tag{30}$$

with $\mathbf{g}$ a vector whose components are the values of g evaluated for the various species and phase space locations, $\mathcal{B}$ and $\mathcal{C}$ are matrices corresponding to the linear operators defined by Eqs. (28) and (29), respectively, and $\mathcal{A}$ is the nonlinear operator defined by Eq. (27). There is no explicit mention of φ in Eq. (30), as φ itself can be expressed via quasineutrality (25) as an operator acting on $\mathbf{g}$. Discretizing in time and splitting the operators gives

$$\mathbf{g}^{\overline{n}} = \mathbf{g}^n + \Delta t \mathcal{A}[\mathbf{g}] \tag{31}$$

$$\mathbf{g}^{\overline{\overline{n}}} = \mathbf{g}^{\overline{n}} + \Delta t \mathcal{B}\mathbf{g} \tag{32}$$

$$\mathbf{g}^{n+1} = \mathbf{g}^{\overline{\overline{n}}} + \Delta t \mathcal{C}\mathbf{g}, \tag{33}$$

where n indicates the time index and $\Delta t = \tilde{t}_{n+1} - \tilde{t}_n$. We leave specification of the time discretization of the righthand sides of each of these equations to dedicated subsections below. The Lie splitting given above is accurate to first order in Δt. Reversing the order of operations – operating first with $\mathcal{C}$, then $\mathcal{B}$ and finally $\mathcal{A}$ – in the next time step makes the splitting accurate to second order in Δt when $\mathcal{A}$ is also a linear operator (cf. [38]), albeit with the effective time step doubled in size. When the nonlinearity is included in $\mathcal{A}$, only first order accuracy is guaranteed. This 'flip-flop' version of Lie splitting is what we employ in `stella`.

Note that higher order accuracy in time could be achieved, at the usual cost of more computation per time step, by employing an Additive Runge Kutta (ARK) scheme [39] instead of operator splitting. Indeed, such an approach has recently been applied to the full-f gyrokinetic equation and successfully used to treat turbulence in the edge of tokamaks [19]. The choice to use operator splitting in `stella` instead of an ARK variant is motivated primarily by its flexibility (ARK schemes do not allow for the separate implicit treatments detailed in Secs. 5.2 and 5.3) and simplicity, as well as by previous experience indicating that relatively low order accuracy in time is often sufficient for the accurate calculation of growth rates and turbulent fluxes. However, the inclusion of an ARK scheme in `stella` – possibly with the current operator splitting approach as a preconditioner – could potentially improve computational efficiency.

5.1. Explicit time advance for $\mathcal{A}$

The evolution of g due to magnetic drifts, background gradient drive, and $E \times B$ nonlinear advection, described by Eqs. (27) and (31), is treated explicitly in `stella`. Although not the default option, users may choose to treat the rest of the terms in the gyrokinetic equation, i.e., parallel streaming and acceleration, explicitly as well. For the explicit time advance algorithm, the user can choose between standard fourth order Runge–Kutta (RK4) and second or third order strong stability preserving (SSP) Runge–Kutta (RK2 and RK3) schemes [40]. While the overall time advance algorithm is limited to second order accuracy in time, the option to treat the explicit terms with a higher order scheme is provided in order to improve their stability properties. The SSP schemes are constructed so that they retain the stability properties of the forward Euler method and have been optimized so that they allow for the least restrictive CFL condition possible. We provide details here for the default scheme in `stella`, which is SSP RK3. Applying SSP RK3 to Eq. (31) gives

$$\mathbf{g}^{\overline{n}} = \frac{\mathbf{g}^n}{3} + \frac{\Delta t}{2}\mathbf{g}_1^n + \frac{\Delta t}{6}\left(\mathbf{g}_2^n + \mathbf{g}_3^n\right), \tag{34}$$

with $\mathbf{g}_1^n = \mathbf{g}^n + \mathcal{A}[\mathbf{g}^n]$, $\mathbf{g}_2^n = \mathbf{g}_1^n + \mathcal{A}[\mathbf{g}_1^n]$, and $\mathbf{g}_3^n = \mathcal{A}[\mathbf{g}_2^n]$.

Our Fourier spectral treatment in x and y eliminates all differential operators in the linear part of $\mathcal{A}$, making it algebraic. The nonlinear $E \times B$ advection is treated pseudo-spectrally, with de-aliasing achieved by padding the final third of the Fourier coefficients with zeros [41]. The use of Fast Fourier Transforms makes each of the $1D$ transforms in x and y computationally efficient ($\mathcal{O}(N \ln N)$ operations, with N the number of padded k_x or k_y coefficients retained). The explicit time advance is parallelized in `stella` so that each $v_\parallel$, μ and species can be solved independently, with the only communication occurring at the end of each fractional Runge–Kutta step when the potential φ must be updated.

5.2. Semi-Lagrange treatment of $\mathcal{B}$

The parallel acceleration described by Eqs. (28) and (32) is simply advection in $v_\parallel$. The semi-Lagrange approach employed in `stella` exploits the fact that this advection has the analytical solution

$$\tilde{g}_{\mathbf{k},s}(v_{\parallel,*}, \tilde{\mu}_s, \tilde{t} + \Delta t) = \tilde{g}_{\mathbf{k},s}(\tilde{v}_\parallel + \Delta t(v_{\mathrm{th},s}/v_{\mathrm{th},r})\tilde{\mu}_s \hat{\mathbf{b}} \cdot \tilde{\nabla}\tilde{B}, \tilde{\mu}_s, \tilde{t}). \tag{35}$$

The quantity $v_{\parallel,*} = \tilde{v}_\parallel + \Delta t(v_{\mathrm{th},s}/v_{\mathrm{th},r})\tilde{\mu}_s \hat{\mathbf{b}} \cdot \tilde{\nabla}\tilde{B}$, does not in general coincide with a grid location in $\tilde{v}_\parallel$. We thus approximate the value of g at $v_{\parallel,*}$ by interpolating the values from the four nearest-neighbor grid points, an approach that is accurate to fourth order in $v_\parallel$ grid spacing, $\Delta v_\parallel$. For $v_{\parallel,*}$ falling between grid points $\tilde{v}_{\parallel,j}$ and $\tilde{v}_{\parallel,j+1}$, the interpolation formula is

$$\tilde{g}(v_{\parallel,*}) = \frac{1}{6}\left(c_j c_{j+1} c_{j+2}\tilde{g}_{j-1} - 3c_{j-1}c_{j+1}c_{j+2}\tilde{g}_j + 3c_{j-1}c_j c_{j+2}\tilde{g}_{j+1} - c_{j-1}c_j c_{j+1}\tilde{g}_{j+2}\right) + \mathcal{O}(\Delta v_\parallel^4), \tag{36}$$

with $c_j \doteq (\tilde{v}_{\parallel,j} - v_{\parallel,*})/\Delta v_\parallel$. At the boundaries in $\tilde{v}_\parallel$, simple linear interpolation is used. Note that all phase space indices aside from the one corresponding to $v_\parallel$ have been suppressed for simplicity of notation. Combining this interpolation formula with the analytical solution (35) and applying it to the split equation (32) gives

$$\tilde{g}^{\overline{\overline{n}}}_{\mathbf{k},s,j} = \frac{1}{6}\left(c_j c_{j+1} c_{j+2}\tilde{g}^{\overline{n}}_{\mathbf{k},s,j-1} - 3c_{j-1}c_{j+1}c_{j+2}\tilde{g}^{\overline{n}}_{\mathbf{k},s,j} + 3c_{j-1}c_j c_{j+2}\tilde{g}^{\overline{n}}_{\mathbf{k},s,j+1} - c_{j-1}c_j c_{j+1}\tilde{g}^{\overline{n}}_{\mathbf{k},s,j+2}\right). \tag{37}$$

To facilitation this interpolation, a re-mapping is done so that $\tilde{g}(\tilde{v}_\parallel)$ is available on all processors; information about g at all other phase space locations can be spread over multiple processors and solved for simultaneously.

Note that the semi-Lagrange approached detailed here places neither an accuracy nor a stability restriction on the time step size: the only error comes from interpolation in $\tilde{v}_\parallel$, which can be carried out to high order with relatively little numerical expense.

5.3. Implicit treatment of $\mathcal{C}$

The dynamics of parallel streaming described by Eqs. (29) and (33) are treated implicitly in `stella` following a similar approach to that taken by the local, δf gyrokinetic code `GS2` [4]. We discretize Eq. (29) using variable centering in z and t. For z, we use a compact, two-point stencil to facilitate the use of tridiagonal matrix solution methods. Derivatives in z are given by

$$\left(\frac{\partial \tilde{g}}{\partial z}\right)_{i*} = \frac{\tilde{g}_{i+1} - \tilde{g}_i}{\Delta z}, \tag{38}$$

where the subscripts i and $i+1$ denote evaluation at grid locations z_i and z_{i+1}, respectively, and $\Delta z \doteq z_{i+1} - z_i$. In Eq. (38) and for the remainder of this subsection we suppress all phase space indices except those corresponding to z and t to simplify notation. The subscript $i*$ indicates evaluation at

$$z_{i*} \doteq \frac{1 \mp u_z}{2} z_i + \frac{1 \pm u_z}{2} z_{i+1}, \tag{39}$$

with the top (bottom) signs used when the parallel advection speed is positive (negative). This sign convention will be used for the remainder of this subsection. The user-specified parameter u_z controls spatial centering: at the extremes, $u_z = 0$ corresponds to a centered derivative that is accurate to second order in Δz, and $u_z = 1$ corresponds to a fully upwinded derivative that is accurate to first order in Δz. All other z-dependent quantities are evaluated at z_{i*} using the approximation

$$\tilde{g}_{i*} = \frac{1 \mp u_z}{2} \tilde{g}_i + \frac{1 \pm u_z}{2} \tilde{g}_{i+1}, \tag{40}$$

which is accurate to second order in Δz.

The time discretization is treated in a manner analogous to the z discretization, with implicitness taking the place of upwinding. The time derivative is given by

$$\left(\frac{\partial \tilde{g}}{\partial t}\right)^{n*} = \frac{\tilde{g}^{n+1} - \tilde{g}^{\overline{n}}}{\Delta t}, \tag{41}$$

with the superscript $n*$ indicating evaluation at

$$t_{n*} \doteq \frac{1 - u_t}{2} t_n + \frac{1 + u_t}{2} t_{n+1}. \tag{42}$$

The user-specified parameter u_t controls temporal centering: at the extremes, $u_t = 0$ corresponds to a centered derivative that is accurate to second order in Δt, and $u_t = 1$ corresponds to a fully implicit treatment accurate to first order in Δt. All other t-dependent quantities are evaluated at t_{n*} using the approximation

$$\tilde{g}^{n*} = \frac{1 - u_t}{2} \tilde{g}^{\overline{n}} + \frac{1 + u_t}{2} \tilde{g}^{n+1}, \tag{43}$$

which is accurate to second order in Δt.

Applying the above z and t discretizations to Eq. (29) yields

$$\begin{aligned}
&\left(\frac{1 \pm u_z}{2} \tilde{g}^{n+1}_{i+1} + \frac{1 \mp u_z}{2} \tilde{g}^{n+1}_{i}\right) + \frac{1 + u_t}{2} \frac{\Delta t}{\Delta z} \frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}} \tilde{v}_\parallel \left(\hat{\mathbf{b}} \cdot \tilde{\nabla} z\right)_{i*} \\
&\quad \times \left(\tilde{g}^{n+1}_{i+1} - \tilde{g}^{n+1}_{i} + \left(J_{0,i+1} \tilde{\varphi}^{n+1}_{i+1} - J_{0,i} \tilde{\varphi}^{n+1}_{i}\right) \frac{Z}{\tilde{T}} \exp\left(-\tilde{v}^2_{i*}\right)\right) \\
&= \left(\frac{1 \pm u_z}{2} \tilde{g}^{\overline{n}}_{i+1} + \frac{1 \mp u_z}{2} \tilde{g}^{\overline{n}}_{i}\right) - \frac{1 - u_t}{2} \frac{\Delta t}{\Delta z} \frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}} \tilde{v}_\parallel \left(\hat{\mathbf{b}} \cdot \tilde{\nabla} z\right)_{i*} \\
&\quad \times \left(\tilde{g}^{\overline{n}}_{i+1} - \tilde{g}^{\overline{n}}_{i} + \left(J_{0,i+1} \tilde{\varphi}^{\overline{n}}_{i+1} - J_{0,i} \tilde{\varphi}^{\overline{n}}_{i}\right) \frac{Z}{\tilde{T}} \exp\left(-\tilde{v}^2_{i*}\right)\right)
\end{aligned} \tag{44}$$

In principle, solving Eq. (44) involves the solution of a linear system that is bidiagonal in the z-component of the matrix and is dense in $(v_\parallel, \mu, s)$ due to the velocity space integral and species sum implicit in φ. The inversion of the dense matrix can be avoided by using a Green's function approach [4], leaving only a computationally inexpensive bidiagonal matrix solve.

To formulate the Green's function approach, we start by noting that Eq. (44) is linear in $\tilde{g}^{n+1}$. We can thus express $\tilde{g}^{n+1}$ as the linear combination $\tilde{g}^{n+1} = \tilde{g}^{n+1}_h + \tilde{g}^{n+1}_{inh}$, with

$$\left(\frac{1\pm u_z}{2}\tilde{g}^{n+1}_{inh,i+1}+\frac{1\mp u_z}{2}\tilde{g}^{n+1}_{inh,i}\right)+\frac{1+u_t}{2}\frac{\Delta t}{\Delta z}\frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}}\tilde{v}_\parallel\left(\hat{\mathbf{b}}\cdot\tilde{\nabla}z\right)_{i*}\left(\tilde{g}^{n+1}_{inh,i+1}-\tilde{g}^{n+1}_{inh,i}\right)$$
$$=\left(\frac{1\pm u_z}{2}\tilde{g}^{\overline{\overline{n}}}_{i+1}+\frac{1\mp u_z}{2}\tilde{g}^{\overline{\overline{n}}}_{i}\right)-\frac{1-u_t}{2}\frac{\Delta t}{\Delta z}\frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}}\tilde{v}_\parallel\left(\hat{\mathbf{b}}\cdot\tilde{\nabla}z\right)_{i*} \tag{45}$$
$$\times\left(\tilde{g}^{\overline{\overline{n}}}_{i+1}-\tilde{g}^{\overline{\overline{n}}}_{i}+\left(J_{0,i+1}\tilde{\varphi}^{\overline{\overline{n}}}_{i+1}-J_{0,i}\tilde{\varphi}^{\overline{\overline{n}}}_{i}\right)\frac{Z}{\tilde{T}}\exp\left(-\tilde{v}^2_{i*}\right)\right)$$

and

$$\left(\frac{1\pm u_z}{2}\tilde{g}^{n+1}_{h,i+1}+\frac{1\mp u_z}{2}\tilde{g}^{n+1}_{h,i}\right)+\frac{1+u_t}{2}\frac{\Delta t}{\Delta z}\frac{v_{\mathrm{th},s}}{v_{\mathrm{th},r}}\tilde{v}_\parallel\left(\hat{\mathbf{b}}\cdot\tilde{\nabla}z\right)_{i*}$$
$$\times\left(\tilde{g}^{n+1}_{h,i+1}-\tilde{g}^{n+1}_{h,i}+\left(J_{0,i+1}\tilde{\varphi}^{n+1}_{i+1}-J_{0,i}\tilde{\varphi}^{n+1}_{i}\right)\frac{Z}{\tilde{T}}\exp\left(-\tilde{v}^2_{i*}\right)\right)=0. \tag{46}$$

The 'twist-and-shift' boundary condition described in Sec. 3.2 is applied at the end of each 2π segment in z. This boundary condition couples multiple 2π segments in z with different k_x values, leading to an extended z domain with N_{z+} points, where $N_{z+}=N_z\times N_{\mathrm{seg}}$ and N_{seg} is the number of connected segments.

From quasineutrality, we have

$$\tilde{\varphi}^{n+1}_i=\left(\sum_s\left(1-\Gamma_{0,i,s}\right)\frac{Z_s^2}{\tilde{T}_s}\tilde{n}_s\right)^{-1}\sum_s Z_s\tilde{n}_s\frac{2\tilde{B}}{\pi^{1/2}}\int_{-\infty}^{\infty}d\tilde{v}_\parallel\int_0^{\infty}d\tilde{\mu}_s J_{0,i,s}\tilde{g}^{n+1}_{i,s}. \tag{47}$$

We get the Green's function for $\tilde{g}^{n+1}_h$ by supplying a unit impulse to $\tilde{\varphi}$ for each z location in the extended z domain and solving Eq. (46) for the response $\tilde{g}^{n+1}_h$. Following this approach we have

$$\tilde{g}^{n+1}_i=\sum_{p=1}^{N_{z+}}\frac{\delta\tilde{g}_{h,i}}{\delta\tilde{\varphi}_p}\tilde{\varphi}^{n+1}_p+\tilde{g}^{n+1}_{inh,i}, \tag{48}$$

where N_{z+} is the number of grid points in the extended z domain, and $\delta\tilde{g}_{h,i}/\delta\tilde{\varphi}_p$ is the response of $\tilde{g}^{n+1}_h$ at grid location z_i to a unit perturbation in $\tilde{\varphi}^{n+1}$ at grid location z_p. Substituting this form for $\tilde{g}^{n+1}_i$ into Eq. (47) yields an implicit equation for the vector $\boldsymbol{\varphi}^{n+1}$ whose i^{th} component is $\tilde{\varphi}^{n+1}_i$:

$$\left(I-Q\sum_{p=1}^{N_{z+}}\frac{\delta\boldsymbol{g}^{n+1}}{\delta\boldsymbol{\varphi}^{n+1}}\right)\boldsymbol{\varphi}^{n+1}=\boldsymbol{\varphi}^{n+1}_{inh}, \tag{49}$$

where $(\delta\boldsymbol{g}/\delta\boldsymbol{\varphi})_{ip}=\delta\tilde{g}_{h,i}/\delta\tilde{\varphi}_p$, I is the $N_{z+}\times N_{z+}$ identity matrix,

$$Q=\left(\sum_s\left(1-\Gamma_{0s}\right)\frac{Z_s^2e}{T_s}n_s\right)^{-1}\sum_s Z_s\int d^3v\,J_{0,s} \tag{50}$$

is the velocity-space operator appearing in quasineutrality and $(\boldsymbol{\varphi}_{inh})_i=Q\,\tilde{g}_{inh,i}$.

Thus `stella` first solves Eq. (45) for $\tilde{g}^{n+1}_{inh}$ and uses it in Eq. (49) to obtain $\tilde{\varphi}^{n+1}$ via LU decomposition and back-substitution. Finally, the updated distribution function $\tilde{g}^{n+1}$ is calculated via Eq. (44). The layout of the data for this implicit solve is the same as for the explicit advance described in Subsection 5.1: information for each k_x, k_y, and z are available to all processors, while g evaluated at each $v_\parallel$, μ, and species can be solved for simultaneously.

5.3.1. Zonal modes

The $k_y=0$ modes, often referred to as zonal modes, must be treated specially, as they are periodic in z. To enforce periodicity, we solve the gyrokinetic equation twice each time it is required: once with a zero incoming boundary condition in z; and once with a unity incoming boundary condition in z, but with no terms involving $\tilde{g}^{\overline{\overline{n}}}$ or $\tilde{\varphi}^{\overline{\overline{n}}}$. We denote the former solution as $\tilde{g}_{PI}$ and the latter as $\tilde{g}_{CF}$. Noting that any linear combination of $\tilde{g}_{PI}$ and $\tilde{g}_{CF}$ is also a solution for $\tilde{g}$ and enforcing periodicity, we have

$$\tilde{g}^{n+1}_1=\tilde{g}_{PI,1}+d\tilde{g}_{CF,1}=d=\tilde{g}_{PI,N_z}+d\tilde{g}_{CF,N_z}. \tag{51}$$

Solving for d and substituting into the above linear combination gives the solution for zonal modes:

$$\tilde{g}^{n+1}_i=\tilde{g}_{PI,i}+\frac{\tilde{g}_{PI,N_z}}{1-\tilde{g}_{CF,N_z}}\tilde{g}_{CF,i}. \tag{52}$$

Table 1
List of `stella` input parameters for the CBC simulations

CBC input parameters	
Input variable	Input value
$\tilde{r} = r/a$	0.5
$\tilde{R} = R_0/a$	2.77778
$d\tilde{R}/d\tilde{r}$	0
q	1.4
$\hat{s} = d\ln q/d\ln r$	0.796
κ	1.0
$d\kappa/d\tilde{r}$	0.0
δ	0.0
$d\delta/d\tilde{r}$	0.0
B_r	$B_\zeta(R_0)$
n_i/n_e	1.0
T_i/T_e	1.0
m_i/m_e	3672
$d\ln n_i/d\tilde{r}$	−0.8
$d\ln n_e/d\tilde{r}$	−0.8
$d\ln T_i/d\tilde{r}$	−2.49
$d\ln T_e/d\tilde{r}$	−2.49

Table 2
List of `stella` input parameters for the NCSX simulations

NCSX design LI383 input parameters	
Input variable	Input value
$\tilde{\psi} = \psi_t/\psi_{t,LCFS}$	0.635
α	0
# of field periods	3
n_i/n_e	1.0
T_i/T_e	1.0
m_i/m_e	3672
$d\ln n_i/dx$	0.0
$d\ln T_i/dx$	4.0

6. Numerical tests

In this section we provide simulation data to illustrate the accuracy and efficiency of `stella`. Throughout, we verify the `stella` simulations by comparing with the widely-benchmarked, δf-gyrokinetic code `GS2` [4,42]. We focus on two magnetic field configurations for our comparisons: the first is the so-called 'Cyclone Base Case' (CBC), a widely-used benchmark case [43] in the magnetic confinement fusion community that has an axisymmetric magnetic field with concentric circular flux surfaces; the second is design LI383 for the National Compact Stellarator Experiment (NCSX), a case which has also been used for benchmarking within the stellarator community [13].

All simulations for the CBC used the Miller local equilibrium [36] option to specify geometric coefficients, while the NCSX simulations used data from the `VMEC`-generated equilibrium for LI383. Tables 1 and 2 provide the relevant `stella` input parameters for these cases.

6.1. Linear simulation results for the CBC

We start by comparing linear growth rates, frequencies, and mode structures obtained from `stella` and `GS2` simulations for the CBC. Unless stated otherwise, all linear CBC simulation results shown here were obtained with the following resolution: Both `stella` and `GS2` used $N_z = 25$ and three 2π segments in an extended ballooning domain. Additionally, `stella` used $N_{v_\parallel} = 48$, $N_\mu = 12$, and $\tilde{\nu}_{\parallel,c} = \tilde{\nu}_{\perp,c} = 3$, while `GS2` used 33 pitch angles (20 in the untrapped region of phase space and 13 in the trapped region) and 16–32 energy grid points. The `stella` cell-centering parameters in z and t were set to $u_z = u_t = 0.02$. A description of the `GS2` velocity space treatment can be found in Ref. [44].

First, we compare the two codes for the case of a modified Boltzmann response for the electrons; i.e., $\delta n_e = e n_e(\varphi - \overline{\varphi})/T_e$, with the overline denoting an average along the magnetic field. The results are given in Fig. 2. All quantities agree to within a few percent across the entire range of unstable k_y values. We compare the same quantities with kinetic electrons at both ion and electron scales in Figs. 3 and 4. In both cases, there is again excellent agreement between the data from `stella` and `GS2`.

As noted in Section 5, a guiding principle for the `stella` algorithm was the desirability of an implicit treatment for parallel streaming and acceleration when including kinetic electrons. To demonstrate the utility of the operator-split,

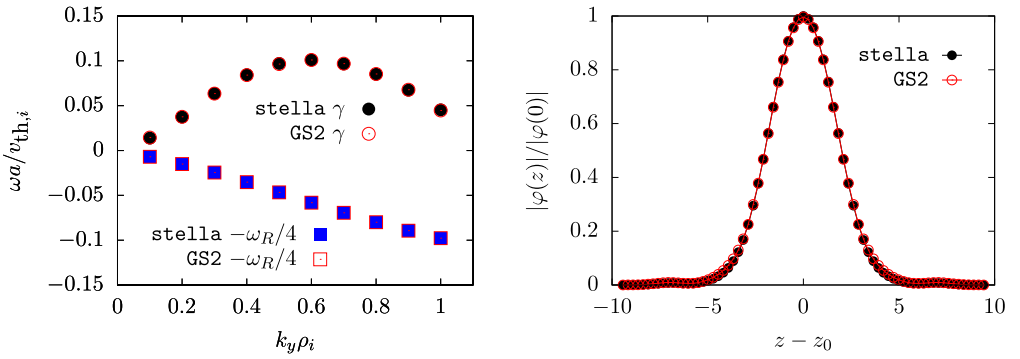

Fig. 2. (Left): Normalized real (*squares*) and imaginary (*circles*) components of mode frequency ω as a function of the normalized binormal wavenumber k_y for `stella` (*filled blue/black*) and `GS2` (*open red*). (Right): Electrostatic potential for the fastest growing mode ($k_y\rho_i = 0.6$) as a function of the extended parallel coordinate $z - z_0 = \theta - \theta_0$ for `GS2` (*open red circles*) and `stella` (*filled black circles*). These data correspond to CBC parameters with a modified Boltzmann response for electrons.

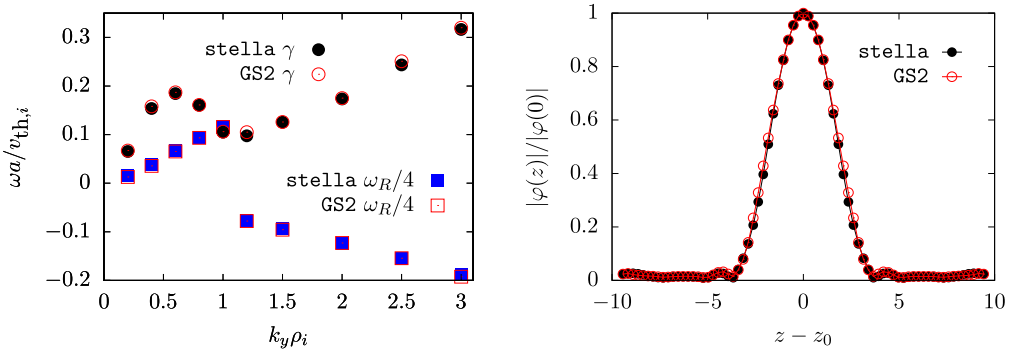

Fig. 3. (Left): Normalized real (*squares*) and imaginary (*circles*) components of mode frequency ω as a function of the normalized binormal wavenumber k_y for `stella` (*filled blue/black*) and `GS2` (*open red*). (Right): Electrostatic potential for the fastest growing mode ($k_y\rho_i = 0.6$) as a function of the extended parallel coordinate $z - z_0 = \theta - \theta_0$ for `GS2` (*open red circles*) and `stella` (*filled black circles*). These data correspond to CBC parameters with kinetic electrons.

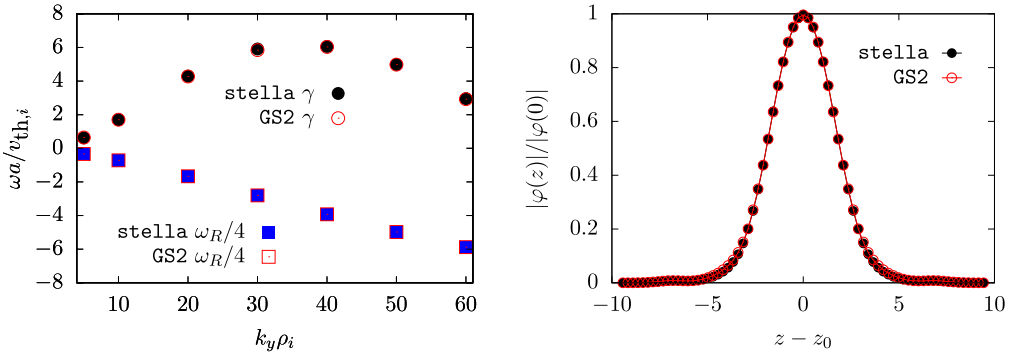

Fig. 4. (Left): Normalized real (*squares*) and imaginary (*circles*) components of mode frequency ω as a function of the normalized binormal wavenumber k_y for `stella` (*filled blue/black*) and `GS2` (*open red*). (Right): Electrostatic potential for the fastest growing mode at electron scales ($k_y\rho_i = 40$) as a function of the extended parallel coordinate $z - z_0 = \theta - \theta_0$ for `GS2` (*open red circles*) and `stella` (*filled black circles*). These data correspond to CBC parameters with kinetic electrons.

implicit–explicit treatment detailed in Section 5, we determined the maximum stable time step for both the fully explicit and the implicit–explicit versions of `stella`. For convenience of notation, we will henceforth refer to the implicit–explicit method as the IMEX method. The results are shown in Fig. 5. As mentioned in Section 5, the fully explicit approach has a much more restrictive CFL condition than the IMEX approach: For $k_y\rho_i \sim 1$, the maximum stable time step for the explicit scheme is ~ 100 times smaller than for the IMEX scheme, and this gap widens at longer wavelengths.

The severe time step constraint for $k_y\rho_i \ll 1$ is due to the rapid response of the electric field to small charge imbalances at long wavelength. This response leads to a high frequency mode in the plasma known as the shear-Alfvén wave [37], which has a frequency proportional to $1/k_\perp$. However, an even more restrictive constraint on the time step that scales as

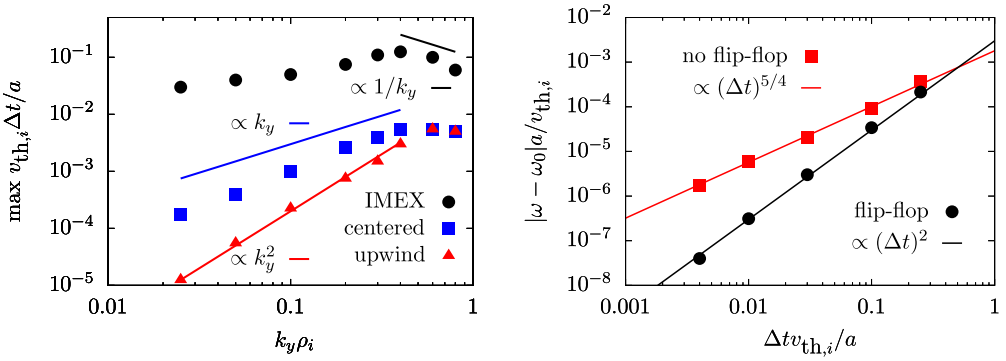

Fig. 5. (Left): Maximum stable time step as a function of $k_y\rho_i$ for the fully explicit scheme with upwind (*red triangles*) and centered (*blue squares*) differencing of $\partial\varphi/\partial z$ and for the mixed implicit–explicit (*black circles*) time advance schemes. Also shown are the expected scalings of the maximum stable time step with $k_y\rho_i$ (see main text for an explanation of these scalings). These correspond to CBC parameters with kinetic electrons. (Right): Modulus of the error in the complex frequency ω as a function of time step size Δt for the time advance scheme with Lie operator splitting (red squares) and with the 'flip-flop' variant of Lie splitting (black circles). The reference complex frequency ω_0 is obtained from a simulation with $\Delta t v_{\mathrm{th},i}/a = 0.0015$. All simulations used CBC parameters with a modified Boltzmann response for electrons.

$1/k_\perp^2$ appears if one does not use centered differences when discretizing the z derivative of φ [8]. A brief calculation deriving these time step constraints for the simplified case of an un-sheared, homogeneous plasma slab is given in Appendix A.

At short wavelengths, the explicit treatment of advection by magnetic drifts – with an advection speed proportional to the wavenumber – provides a CFL time step that scales as $1/k_y$. This is evident for $k_y\rho_i \gtrsim 0.4$ in Fig. 5. The explicit treatment of the magnetic drifts is also responsible for the CFL time step at long wavelengths in the IMEX approach. A calculation similar to that given in Appendix A shows that the terms containing a product of the magnetic drifts and the electrostatic potential scale inversely with k_y, leading to a CFL time step at long wavelengths that scales as k_y. This scaling is evident for $k_y\rho_i \lesssim 0.3$ in Fig. 5.

Finally, to demonstrate the utility of the 'flip-flop' version of Lie operator spitting discussed in Sec. 5, we show in Fig. 5 the convergence of the computed complex frequency ω with decreasing time step Δt for a case with Boltzmann electrons. Here, the numerical resolution used is $N_{v_\parallel} = 64$, $N_\mu = 6$ and $N_z = 9$, with one 2π segment along z. Denoting ω_0 as the value for ω at very small Δt (a factor of 3 below those shown), we see that the error $|\omega - \omega_0|$ scales as $(\Delta t)^2$ for the 'flip-flop' scheme and as approximately $(\Delta t)^{5/4}$ for regular Lie splitting.

6.2. Nonlinear simulation results for the CBC

We next compare turbulent heat fluxes from nonlinear simulations with CBC parameters for `stella` and `GS2`. Both `stella` and `GS2` simulations used $N_z = 32$, $N_{k_y} = 22$, and $N_{k_x} = 128$, with a box size in x and y of approximately $126\rho_i$. Note that the effective value for N_{k_y} should be doubled, as both `stella` and `GS2` use the reality condition to limit the simulated $\mathbf{k}$-domain so that $k_y \geq 0$. The velocity space resolution was the same for both codes as the linear case, with the exception that `stella` used $N_\mu = 16$. A small amount of hyper-viscosity was employed in all simulations to avoid spectral pile-up and was treated implicitly using the same 'flip-flop' operator splitting employed for the parallel acceleration and streaming. The form for hyper-viscosity currently used in `stella` is

$$\frac{\partial g}{\partial t} = -D\frac{k_\perp^4}{k_{\perp,max}^4}g, \tag{53}$$

with $D = 0.05$ for the simulations reported here.

The turbulent heat fluxes for simulations with kinetic electrons and with a modified Boltzmann response for electrons are given in Fig. 6. There is remarkable agreement between the `stella` and `GS2` heat fluxes for ions in both cases and for electrons in the case where they are treated kinetically.

6.3. Linear simulation results for NCSX design LI383

For the NCSX linear benchmarks, a modified Boltzmann response was enforced for the electrons. The resolution required to achieve converged results at all k_x and k_y values was higher than for the CBC simulations in both `stella` and `GS2`. Both codes included all three NCSX field periods (corresponding to a ζ domain of $[-15.653, 15.653]$). The `stella` simulations used $N_z = 1025$, $N_{v_\parallel} = 128$, and $N_\mu = 24$, while the `GS2` simulations used $N_z = 455$, 24 energy grid points, and 50 (60) pitch-angles for the k_y (k_x) scan. These extreme resolutions were not necessary in most cases, but were used to ensure agreement in the most challenging ones.

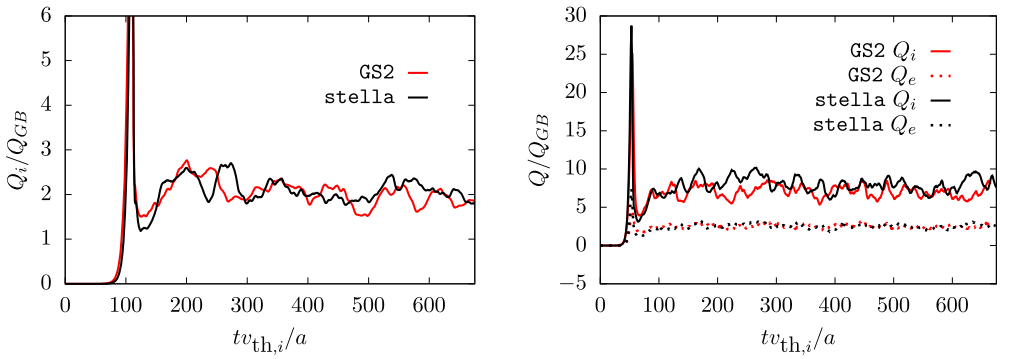

Fig. 6. Time traces of the gyro-Bohm-normalized heat flux ($Q_{GB} = n_i T_i v_{\mathrm{th},i} \rho_i^2/a^2$) for `stella` (*black line*) and `GS2` (*red line*). Simulations obtained using CBC parameters. (Left): Electrons have a modified Boltzmann response. (Right): Electrons are treated kinetically. Solid lines denote ion fluxes, and dashed line denote electron fluxes.

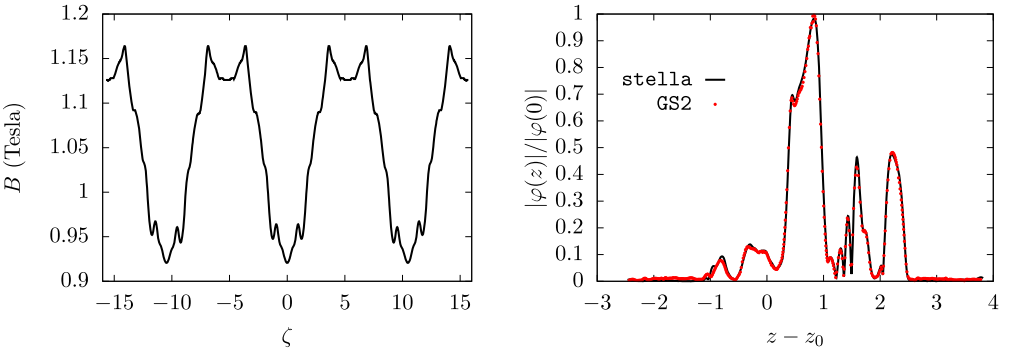

Fig. 7. (Left): Magnetic field strength vs ζ for the simulated NCSX equilibrium (LI383). (Right): Normalized modulus of φ vs $z - z_0$ for ($k_x\rho_i = 0.64$, $k_y\rho_i = 1.414$) obtained by `stella` (*black line*) and `GS2` (*red circles*).

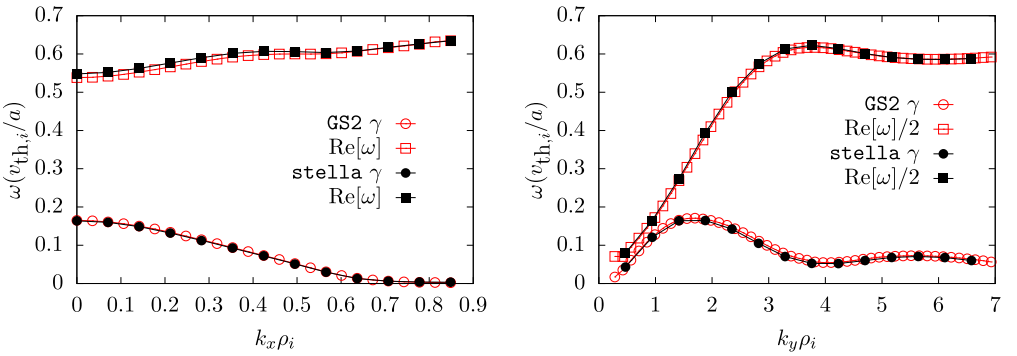

Fig. 8. Real frequency and growth rate spectra for scaled NCSX equilibrium LI383 for $k_y\rho_i = 1.414$ (left) and $k_x\rho_i = 0$ (right).

The variation in magnetic field strength with toroidal angle ζ is given in Fig. 7, along with an example of the mode structure along z for ($k_x = 0.636, k_y = 1.414$). One can see that there is a significant amount of structure, which is what necessitates the higher resolution in z. Plots of the linear growth rates and real frequencies for both k_x and k_y scans are given in Fig. 8.

7. Summary and future work

The algorithm for `stella` presented here enables fast, accurate evaluation of the gyrokinetic equation (2) subject to the quasineutrality constraint (5). Its use of a mixed implicit–explicit (IMEX) algorithm greatly reduces the CFL time step constraint, especially at long wavelengths. The implicit part of the solve is facilitated by operator splitting, which allows for a flexible treatment of the various different physics effects appearing in the gyrokinetic equation. The code has been benchmarked both linearly (Sec. 6.1) and nonlinearly (Sec. 6.2) – with and without retention of kinetic electron dynamics –

for axisymmetric magnetic field configurations. It has also been benchmarked linearly with a modified Boltzmann electron response in a non-axisymmetric magnetic field configuration (Sec. 6.3). As such, there are a wide range of problems to which it can be immediately and usefully applied.

However, there are a few obvious ways in which `stella` could be improved that are under development. The code does not currently include the effect of Coulomb collisions or of magnetic fluctuations, both of which should be straightforward extensions to `stella`. Also, as pointed out in Sec. 1, one of the motivations for developing `stella` was to study turbulence that is non-local in the bi-normal α coordinate. This can be achieved by abandoning the flux tube approach in favor of an annulus that encompasses the full flux surface of interest. Although this has not been done in the current version of `stella`, the IMEX algorithm with operator splitting was devised with full flux surface simulations in mind. Implementation of this full flux surface option is in progress and will be addressed in detail in a future publication.

Acknowledgements

The authors would like to thank W. Dorland for useful discussions. The authors acknowledge the use of ARCHER through the Plasma HEC Consortium EPSRC grant number EP/L000237/1 under project e281-gs2 and the use of the EUROfusion High Performance Computer (Marconi-Fusion) under project MULTEI. This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No. 633053 for the project WP17-ENR-CEA-02. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Appendix A. Long wavelength numerical instability

Here we address the issue of numerical stability of the gyrokinetic-Poisson system of equations (19) and (25) at long wavelengths. It has been shown previously that a high frequency mode termed the electrostatic shear-Alfvén wave is supported by the plasma within the electrostatic approximation and that this leads to a CFL time step that scales as $k_\parallel/k_\perp$ [37]. It has also been noted that one must take care when discretizing the z derivative appearing in the gyrokinetic equation (19) in order to avoid a numerical instability that scales inversely with $k_\perp^2$ [8]. Here we provide a brief calculation illustrating the origin of this numerical instability and reversion to the CFL constraint of the electrostatic Alfvén wave in the appropriate limit.

To simplify our analysis, we consider an un-sheared, homogeneous plasma slab. For such a system the perpendicular speed (or, equivalently, the magnetic moment) appears only as a parameter in the gyrokinetic equation and can thus be averaged away. The resulting system of equations is

$$\frac{\partial G_{\mathbf{k},s}}{\partial t} + v_\parallel \left(\frac{\partial G_{\mathbf{k},s}}{\partial z} + \frac{Z_s e n_s}{T_s} \Gamma_0(b_{\mathbf{k},s}) \frac{\partial \varphi_{\mathbf{k}}}{\partial z} \frac{e^{-v_\parallel^2/v_{\mathrm{th},s}^2}}{\sqrt{\pi} v_{\mathrm{th},s}} \right) = 0, \tag{A.1}$$

$$\sum_s Z_s e \left(\int_{-\infty}^{\infty} dv_\parallel \, G_{\mathbf{k},s} + \frac{Z_s e n_s}{T_s} \left(\Gamma_0(b_{\mathbf{k},s}) - 1 \right) \varphi_{\mathbf{k}} \right) = 0, \tag{A.2}$$

with $G_{\mathbf{k},s}(v_\parallel, z, t) \doteq 2\pi \int_0^\infty dv_\perp v_\perp J_0(a_{\mathbf{k},s}) g_{\mathbf{k},s}$. We discretize (A.1) and (A.2) using a simple, first-order upwind scheme for G and a more general scheme that combines upwind and centered differences for φ; i.e.,

$$\frac{\partial G_j}{\partial t} + \frac{v_\parallel}{\Delta z} \left(\sigma \left(G_j - G_{j-\sigma} \right) + \frac{Zen}{T} \Gamma_0(b) \left(\alpha\sigma \left(\varphi_j - \varphi_{j-\sigma} \right) + \frac{(1-\alpha)}{2} \left(\varphi_{j+1} - \varphi_{j-1} \right) \right) \frac{e^{-v_\parallel^2/v_{\mathrm{th}}^2}}{\sqrt{\pi} v_{\mathrm{th}}} \right) = 0, \tag{A.3}$$

where $\sigma \doteq v_\parallel/|v_\parallel|$, $\alpha \in [0,1]$ controls the balance of upwind and centered differences in the evaluation of $\partial\varphi/\partial z$, and the subscript j denotes evaluation at z_j. Note that we have suppressed species and wavenumber subscripts to simplify notation.

Assuming solutions for G of the form $G_j = \hat{G}(v_\parallel) \exp\left(ik_\parallel z_j - i\omega t\right)$ then leads to the dispersion relation

$$\sum_s \frac{Z_s^2 n_s}{T_s} \left(1 - \Gamma_0(b_{\mathbf{k},s}) \right) = \sum_s \frac{Z_s^2 n_s}{T_s} \frac{\Gamma_0(b_{\mathbf{k},s})}{\sqrt{\pi}} \int_{-\infty}^{\infty} du \left(\frac{\alpha S_1 + (1-\alpha) S_2}{\zeta_s - u S_1} \right) u e^{-u^2}, \tag{A.4}$$

where $u \doteq v_\parallel/v_{\mathrm{th},s}$, $\zeta_s \doteq \omega/k_\parallel v_{\mathrm{th},s}$, $S_1 \doteq \mathrm{sinc}(k_\parallel \Delta z/2) \exp(-i\sigma k_\parallel \Delta z/2)$, and $S_2 \doteq \mathrm{sinc}(k_\parallel \Delta z)$. Anticipating the existence of high frequency modes at long wavelength, we expand (A.4) for $\zeta_s \gg 1$ and $k_\perp \rho_s \ll 1$ to obtain

$$\begin{aligned} \sum_s \frac{Z_s^2 n_s}{T_s} \frac{k_\perp^2 \rho_s^2}{2} \approx \sum_s \frac{Z_s^2 n_s}{T_s} \frac{v_{\mathrm{th},s}}{\omega \Delta z} \Bigg(& -\frac{2i\alpha}{\sqrt{\pi}} \sin^2\left(\frac{k_\parallel \Delta z}{2} \right) \\ & + \frac{v_{\mathrm{th},s}}{\omega \Delta z} \left(2\alpha \cos(k_\parallel \Delta z) \sin^2\left(\frac{k_\parallel \Delta z}{2} \right) + \frac{1-\alpha}{2} \sin^2(k_\parallel \Delta z) \right) \Bigg), \end{aligned} \tag{A.5}$$

which has been truncated at second order in $1/\zeta_s$. From (A.5), we see that if there is any upwinding of $\partial\varphi/\partial z$ (i.e., $\alpha > 0$), the term proportional to $1/\omega^2$ can be neglected. The result is a spurious damped mode with maximum damping rate γ at the grid scale in z given by

$$\gamma \approx -\frac{4\alpha}{\sqrt{\pi}}\frac{v_{\mathrm{th},e}}{\Delta z}\left(\sum_s Z_s^2\frac{n_s}{n_e}\frac{T_e}{T_s}k_\perp^2\rho_s^2\right)^{-1}. \tag{A.6}$$

If $\partial\varphi/\partial z$ is discretized with centered differences (i.e., $\alpha = 0$), one must retain the term proportional to $1/\omega^2$. In this case, one recovers a discretized form of the electrostatic Alfvén wave:

$$\omega^2 \approx \sin^2(k_\parallel \Delta z)\frac{v_{\mathrm{th},e}^2}{\Delta z^2}\left(\sum_s Z_s^2\frac{n_s}{n_e}\frac{T_e}{T_s}k_\perp^2\rho_s^2\right)^{-1}. \tag{A.7}$$

The requirement that these frequencies be resolved by an explicit time advance method leads to a severe constraint on the allowed time step size.

References

[1] P.J. Catto, Linearized gyro-kinetics, Plasma Phys. 20 (1978) 719.
[2] E.A. Frieman, L. Chen, Nonlinear gyrokinetic equations for low-frequency electromagnetic waves in general plasma equilibria, Phys. Fluids 25 (1982) 502.
[3] S.E. Parker, W.W. Lee, R.A. Santoro, Gyrokinetic simulation of ion temperature gradient driven turbulence in 3D toroidal geometry, Phys. Rev. Lett. 71 (1993) 2042.
[4] M. Kotschenreuther, G. Rewoldt, W.M. Tang, Comparison of initial value and eigenvalue codes for kinetic toroidal plasma instabilities, Comput. Phys. Commun. 88 (1995) 128–140.
[5] Z. Lin, T.S. Hahm, W.W. Lee, W.M. Tang, R.B. White, Turbulent transport reduction by zonal flows: massively parallel simulations, Science 281 (1998) 1835.
[6] F. Jenko, Massively parallel Vlasov simulation of electromagnetic drift-wave turbulence, Comput. Phys. Commun. 125 (2000) 196.
[7] G. Jost, T.M. Tran, W. Cooper, K. Appert, Global linear gyrokinetic simulations in quasi-symmetric configurations, Phys. Plasmas 8 (2001) 3321.
[8] J. Candy, R.E. Waltz, An Eulerian gyrokinetic-Maxwell solver, J. Comput. Phys. 186 (2003) 545.
[9] Y. Idomura, S. Tokuda, Y. Kishimoto, Global gyrokinetic simulation of ion temperature gradient driven turbulence in plasmas using a canonical Maxwellian distribution, Nucl. Fusion 43 (2003) 234.
[10] T.-H. Watanabe, H. Sugama, Velocity-space structures of distribution function in toroidal ion temperature gradient turbulence, Nucl. Fusion 46 (2006) 24.
[11] S. Jolliet, A. Bottino, P. Angelino, R. Hatzky, T.M. Tran, B.F. McMillan, O. Sauter, K. Appert, Y. Idomura, L. Villard, A global collisionless pic code in magnetic coordinates, Comput. Phys. Commun. 177 (2007) 409.
[12] A.G. Peeters, Y. Camenen, F.J. Casson, W.A. Hornsby, A.P. Snodin, D. Strintzi, G. Szepesi, The nonlinear gyro-kinetic flux tube code GKW, Comput. Phys. Commun. 180 (2009) 2650.
[13] J.A. Baumgaertel, E.A. Belli, W. Dorland, W. Guttenfelder, G.W. Hammett, D.R. Mikkelsen, G. Rewoldt, W.M. Tang, P. Xanthopoulos, Simulating gyrokinetic microinstabilities in stellarator geometry with GS2, Phys. Plasmas 18 (2011) 122301.
[14] P. Xanthopoulos, G.G. Plunk, A. Zocco, P. Helander, Intrinsic turbulence stabilization in a stellarator, Phys. Rev. X 6 (2016) 021033.
[15] E. Sánchez, R. Kleiber, R. Hatzky, M. Borchardt, P. Monreal, F. Castejón, A. López-Fraguas, X. Sáez, J.L. Velasco, I. Calvo, A. Alonso, D. López-Bruna, Collisionless damping of flows in the TJ-II stellarator, Plasma Phys. Control. Fusion 55 (2013) 014015.
[16] M. Nunami, T.-H. Watanabe, H. Sugama, Gyrokinetic Vlasov code including full three-dimensional geometry of experiments, J. Plasma Fusion Res. 5 (2010) 016.
[17] D.A. Spong, I. Holod, Y. Todo, M. Osakabe, Global linear gyrokinetic simulation of energetic particle-driven instabilities in the LHD stellarator, Nucl. Fusion 57 (2017) 086018.
[18] R. Numata, G.G. Howes, T. Tatsuno, M. Barnes, W. Dorland, AstroGK: astrophysical gyrokinetics code, J. Comput. Phys. 229 (2010) 9347, arXiv:1004.0279.
[19] D. Ghosh, M.A. Dorf, M.R. Dorr, J.A.F. Hittinger, Kinetic simulation of collisional magnetized plasmas with semi-implicit time integration, J. Sci. Comput. 77 (2018) 819–849.
[20] P. Xanthopoulos, H.E. Mynick, P. Helander, Y. Turkin, G.G. Plunk, F. Jenko, T. Görler, D. Told, T. Bird, J.H.E. Proll, Controlling turbulence in present and future stellarators, Phys. Rev. Lett. 113 (2014) 155001.
[21] A.J. Brizard, T.S. Hahm, Foundations of nonlinear gyrokinetic theory, Rev. Mod. Phys. 79 (2007) 421.
[22] F.I. Parra, P.J. Catto, Limitations of gyrokinetics on transport time scales, Plasma Phys. Control. Fusion 50 (2008) 065014.
[23] I.G. Abel, G.G. Plunk, E. Wang, M. Barnes, S.C. Cowley, W. Dorland, A.A. Schekochihin, Multiscale gyrokinetics for rotating tokamak plasmas: fluctuations, transport, and energy flows, Rep. Prog. Phys. (2013) 116201.
[24] R.L. Burden, J.D. Faires, Numerical Analysis, Brooks/Cole, Boston, 2011.
[25] M. Abramowitz, I.A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, Dover, New York, 1972.
[26] A.E. White, N.T. Howard, M.J. Greenwald, M.L. Reinke, C. Sung, S. Baek, M. Barnes, J. Candy, A. Dominguez, D. Ernst, C. Gao, A.E. Hubbard, J.W. Hughes, Y. Lin, D. Mikkelsen, F.I. Parra, M. Porkolab, J. Rice, J. Walk, S.J. Wukitch, the Alcator C-Mod Team, Multi-channel transport experiments at Alcator C-Mod and comparison with gyrokinetic simulations, Phys. Plasmas 20 (2013) 056106.
[27] J. Citrin, F. Jenko, P. Mantica, D. Told, C. Bourdelle, R. Dumont, J. Garcia, J.W. Haverkort, Ion temperature profile stiffness: non-linear gyrokinetic simulations and comparison with experiment, Nucl. Fusion 54 (2014) 023008.
[28] T. Görler, A.E. White, D. Told, F. Jenko, C. Holland, T.L. Rhodes, A flux-matched gyrokinetic analysis of DIII-D L-mode turbulence, Phys. Plasmas 21 (2014) 122307.
[29] N.T. Howard, C. Holland, A.E. White, M. Greenwald, J. Candy, A.J. Creely, Multi-scale gyrokinetic simulations: comparison with experiment and implications for predicting turbulence and transport, Phys. Plasmas 23 (2016) 056109.
[30] M. Nakata, M. Honda, M. Yoshida, H. Urano, M. Nunami, S. Maeyama, T.-H. Watanabe, H. Sugama, Validation studies of gyrokinetic ITG and TEM turbulence simulations in a JT-60U tokamak using multiple flux matching, Nucl. Fusion 56 (2016) 086010.
[31] F. van Wyk, E.G. Highcock, A.R. Field, C.M. Roach, A.A. Schekochihin, F.I. Parra, W. Dorland, Ion-scale turbulence in mast: anomalous transport, subcritical transitions, and comparison to bes measurements, Plasma Phys. Control. Fusion 59 (2017) 114003.

[32] M.A. Beer, S.C. Cowley, G.W. Hammett, Field-aligned coordinates for nonlinear simulations of tokamak turbulence, Phys. Plasmas 2 (1995) 7.
[33] M.F. Martin, M. Landreman, P. Xanthopoulos, N.R. Mandell, W. Dorland, The parallel boundary condition for turbulence simulations in low magnetic shear devices, Plasma Phys. Control. Fusion 60 (2018) 095008.
[34] S.P. Hirshman, J.C. Whitson, Steepest-descent moment method for three-dimensional magnetohydrodynamic equilibria, Phys. Fluids 26 (1983) 3553.
[35] S.P. Hirshman, W.I. van Rij, P. Merkel, Three-dimensional free boundary calculations using a spectral green's function method, Comput. Phys. Commun. 43 (1986) 143.
[36] R.L. Miller, M.S. Chu, J.M. Greene, Y.R. Lin-Liu, R.E. Waltz, Noncircular, finite aspect ratio, local equilibrium model, Phys. Plasmas 5 (1998) 973.
[37] W.W. Lee, Gyrokinetic particle simulation model, J. Comput. Phys. 72 (1987) 243.
[38] J.P. Boyd, Chebyshev and Fourier Spectral Methods, Dover Publications, Inc., 2000.
[39] G.J. Cooper, A. Sayfy, Additive methods for the numerical solution of ordinary differential equations, Math. Comput. 35 (1980) 1159–1172.
[40] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (2001) 89.
[41] S.A. Orszag, On the elimination of aliasing in finite-difference schemes by filtering high-wavenumber components, J. Atmos. Sci. 28 (1971) 1074.
[42] W. Dorland, F. Jenko, M. Kotschenreuther, B.N. Rogers, Electron temperature gradient turbulence, Phys. Rev. Lett. 85 (2000) 5579.
[43] A.M. Dimits, G. Bateman, M.A. Beer, B.I. Cohen, W. Dorland, G.W. Hammett, C. Kim, J.E. Kinsey, M. Kotschenreuther, A.H. Kritz, L.L. Lao, J. Mandrekas, W.M. Nevins, S.E. Parker, A.J. Redd, D.E. Shumaker, R. Sydora, J. Weiland, Comparisons and physics basis of tokamak transport models and turbulence simulations, Phys. Plasmas 7 (2000) 969.
[44] M. Barnes, W. Dorland, T. Tatsuno, Velocity space resolution in gyrokinetic simulations, Phys. Plasmas 17 (2010) 032106.