Numerical study of a reduced model coupling 2D+2V drift-kinetic ions and 2D+3V kinetic neutrals in a helical magnetic field with wall boundaries

M. R. Hardman[1], J. Omotani[3], M. Barnes[1], and F. I. Parra[3]

[1] Rudolf Peierls Centre for Theoretical Physics, University of Oxford, Clarendon Laboratory, Parks Road, Oxford OX1 3PU, United Kingdom

[2] Princeton Plasma Physics Laboratory, P.O. Box 451, Princeton, New Jersey 08540, United States

[3] Culham Centre for Fusion Energy, Culham Science Centre, Abingdon, Oxon, OX14 3DB, United Kingdom

E-mail: michael.hardman@physics.ox.ac.uk

1. Introduction

The aim of this report is to describe an initial investigation into the challenges associated with developing 2D edge plasma models with kinetic ion and neutral species. We focus on two fundamental aspects of the problem: we consider the implementation of model ion-neutral collision operators, and the implementation of simple wall boundary conditions. There are other challenges to developing numerical simulations in the plasma edge, for example, a comprehensive model must also deal with the magnetic flux coordinate singularity at the separatrix between the core plasma and the scrape off layer, as well as the inherently global nature of the problem in both the radial and poloidal directions. For a full review of the physics challenges, please see [1].

Previous reports in this series have examined the 1D problem of neutrals and ions interacting on a single magnetic field line, neglecting cross-field physics [2, 3, 4, 5]. Both the 'standard' Eulerian drift-kinetic approach and the novel 'moment kinetics' approach proposed in [6, 7, 8, 9] were implemented and tested [3, 4, 5]. It was demonstrated that the 'moment kinetics' approach is a viable scheme for solving the system numerically, although some problems were encountered with wall boundary conditions [5]. Analytical tests of the implemented code were developed and automated, and an initial physics study was carried out looking at the difference between the ion and neutral distribution functions in steady state in the case with wall boundary conditions.

The major differences between the 2D and 1D models are the following. Cross-field drifts now enter into the ion drift kinetic equation, and the difference between charged particle and neutral particle trajectories manifest themselves in qualitatively different velocity space structure for the two different types of particle species. Whilst

neutral particles are only accelerated by interparticle collisions, charged particles are also constrained to perform helical cyclotron orbits around magnetic field lines by the Lorentz force. This means that whilst neutral particles must be described with three velocity coordinates, we can describe the motion of charged particles with just two velocity coordinates, by averaging over the rapid cyclotron motion. This fact has consequences for the implementation of any kinetic model.

For simplicity in the initial development of the implementation of the 2D model, in this report we restrict our attention to the 'standard' drift-kinetic approach. We take as our starting point the 2D model with wall boundary conditions proposed in [7]. We present the 2D model in the next section. In section 3, we discuss the differing velocity-space grids of charged and neutral particles, and the consequent challenges for the simulation. In sections 4 and 5 we consider the wall boundary conditions for ions and neutrals, and electrons, respectively. In section 6, we write down the normalised system of equations that are implemented in the 2D code. In section 7, we briefly describe the numerical algorithms underpinning the implementation of the model. In section 8, we describe tests that verify the implementation of the 2D model using the method of manufactured solutions. Finally, in section 9, we provide suggestions for future development.

2. Drift-kinetic ions coupling to kinetic neutrals

Here we provide a brief overview of the 2D kinetic model that was derived in detail in [7]. The system that we consider consists of a single ion species of charge e and mass m_i, a single neutral species of mass $m_n = m_i$, and an electron species that is modelled as having a Boltzmann response. The magnetic field is taken to be helical, with the form

$$\mathbf{B} = B_z\hat{\mathbf{z}} + B_\zeta\hat{\boldsymbol{\zeta}}, \tag{1}$$

where (r, z, ζ) are cylindrical coordinates, and B_z and B_ζ are constant in z, r, and ζ. See figure 1. Helical field lines approximate the geometry of the open-field-line scrape-off layer region of plasma that is confined in an axisymmetric toroidal magnetic field. With this interpretation we can identify z as the vertical coordinate along the axis of symmetry, r as the major radial coordinate and ζ as the toroidal angle. We include a simple model charge exchange collision operator acting between ions and neutrals, and a simple model ionisation collision operator involving ions, electrons and neutrals. We do not account for charged-species collisions that couple ions to ions, or electrons to ions. To be consistent with the lack of radial variation in the magnetic field, we impose periodic boundary conditions in r, but we accommodate either wall boundary conditions or periodic boundary conditions in z. We assume that the plasma equilibrium is independent of ζ.

The model kinetic equations are as follows. We have a drift-kinetic equation for

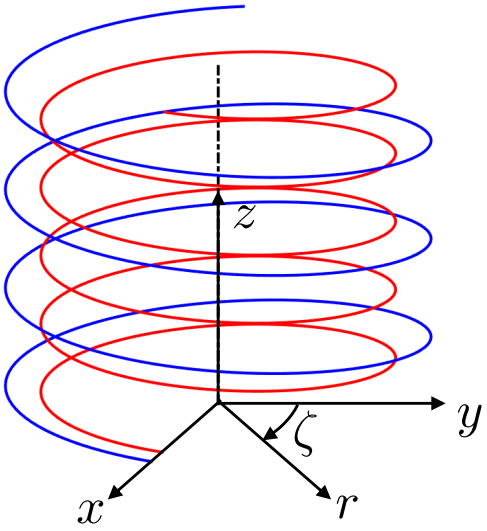


Figure 1: The form of the magnetic field and the left-handed coordinates (r, z, ζ) in terms of the right-handed cartesian (x, y, z). Taken from [7]. To make a connection to the geometry of the scrape-off layer – the open field lines surrounding the toroidal flux surfaces of the core plasma – we should think of r as the major radial coordinate, z as the vertical axial coordinate, and ζ as the toroidal angle.

ions

$$\frac{\partial F_i}{\partial t} + \left(b_z v_\parallel - \frac{E_r}{B}\right)\frac{\partial F_i}{\partial z} + \frac{E_z}{B}\frac{\partial F_i}{\partial r} + \frac{e b_z E_z}{m_i}\frac{\partial F_i}{\partial v_\parallel} = \\ - R_{\text{in}}\left(n_n F_i - n_i \langle F_n \rangle\right) + R_{\text{ion}} n_e \langle F_n \rangle + S_i, \tag{2}$$

where F_s is the particle distribution function for species s, n_s is the particle number density for species s, $v_\parallel = \mathbf{v}\cdot\mathbf{b}$ and $v_\perp = |\mathbf{v} - v_\parallel \mathbf{b}|$ are the components of the particle velocity $\mathbf{v}$ that are parallel and perpendicular to the magnetic field direction $\mathbf{b} = \mathbf{B}/B$, respectively, with $B = \sqrt{B_z^2 + B_\zeta^2}$. The operator $\langle\cdot\rangle = \int_{-\pi}^{\pi} d\vartheta/2\pi$ is the gyroaverage, ϑ is the gyro-angle, t is the time, $b_z = B_z/B$, ϕ is the electrostatic potential, R_{in} and R_{ion} are the charge exchange and ionization collision frequency factors, respectively, and S_i is a source function that may be used to inject particles, momentum, and heat, or to facilitate a test via the method of manufactured solutions. In addition to equation (2) we must solve the 2D-3V kinetic equation for neutrals:

$$\frac{\partial F_n}{\partial t} + v_z\frac{\partial F_n}{\partial z} + v_r\frac{\partial F_n}{\partial r} = -R_{\text{in}}\left(n_i F_n - n_n F_i\right) - R_{\text{ion}} n_e F_n + S_n, \tag{3}$$

where S_n is the neutral source function. We use velocity coordinates aligned with $\hat{\mathbf{z}}$, $\hat{\mathbf{r}}$, and $\hat{\boldsymbol{\zeta}}$ to describe the advection of the neutral species; here, we use the coordinates $v_z = \mathbf{v}\cdot\hat{\mathbf{z}}$, $v_r = \mathbf{v}\cdot\hat{\mathbf{r}}$, and $v_\zeta = \mathbf{v}\cdot\hat{\boldsymbol{\zeta}}$.

The ion density n_i is computed by integrating over the ion distribution function:

$$n_i(z,r,t) = 2\pi \int_{-\infty}^{\infty} dv_\parallel \int_0^{\infty} dv_\perp v_\perp F_i(z,r,v_\perp,v_\parallel,t). \tag{4}$$

The same is true for the neutral density n_n:

$$n_n(z,r,t) = \int_{-\infty}^{\infty} dv_\zeta \int_{-\infty}^{\infty} dv_r \int_{-\infty}^{\infty} dv_z F_n(z,r,v_\zeta,v_r,v_z,t). \tag{5}$$

The electrostatic potential is computed by enforcing quasineutrality $n_i = n_e$ and using a Boltzmann response for electrons, i.e.,

$$n_i = n_e = N_e \exp\left(\frac{e\phi}{T_e}\right), \tag{6}$$

where the constant N_e is either taken to be a reference density, or calculated through a simple electron sheath model. We obtain the electric fields by differentiation of ϕ:

$$E_z = -\frac{\partial \phi}{\partial z}, \text{ and } \quad E_r = -\frac{\partial \phi}{\partial r}. \tag{7}$$

3. Evaluating the collision operator terms

There are two major differences between the 2D and 1D models. The first is that the $\mathbf{E} \times \mathbf{B}$ drift now appears, causing spatial advection across field lines radially. The second difference is that the charged particles and neutral particles have very different velocity space dynamics. A large component of this project is the investigation of how to efficiently couple neutrals to charged particles in a kinetic code. Charged particles and neutral particles couple in two distinct ways: through the wall boundary condition, and through inter-particle collisions. In this section, we discuss a problem related to the fast implementation of the collision operators.

Charged particles in strong magnetic fields undergo a helical motion: particles are free to travel along the magnetic field line; but the $\mathbf{v} \times \mathbf{B}$ component of the Lorentz force causes particles to execute a rapid cyclotron (gyro-) motion in the plane perpendicular to $\mathbf{B}$, with a gyration radius ρ_i much smaller than either the radial or axial length scales L_r and L_z, respectively. Therefore, it is natural to use the field-aligned velocity coordinates $(v_\parallel, v_\perp, \vartheta)$, with which we can write

$$\mathbf{v} = v_\parallel \mathbf{b} + v_\perp(\cos\vartheta\ \hat{\mathbf{r}} - \sin\vartheta\ \mathbf{b} \times \hat{\mathbf{r}}). \tag{8}$$

This yields a great simplification in the description of the charged particle species because in strong magnetic fields

$$\frac{\mathrm{d}v_\parallel}{\mathrm{d}t} \sim \frac{\mathrm{d}v_\perp}{\mathrm{d}t} \sim \frac{c_s}{L_z} \ll \frac{\mathrm{d}\vartheta}{\mathrm{d}t} \sim \Omega_i, \tag{9}$$

where the sound speed is $c_s = \sqrt{2T_e/m_i}$ and the cyclotron frequency is $\Omega_i = eB/m_i$. Because we are not typically interested in motions on frequencies comparable to Ω_i, we can conveniently average over the ϑ motion, resulting in the drift-kinetic equation (2).

None of these simplifications apply to the neutral species, which experience no forces due to the electromagnetic fields. In the absence of collisions, neutral particles execute rectilinear motion, freely crossing magnetic field lines. It is therefore natural to consider a neutral velocity space grid that is aligned with the spatial coordinates, i.e.,

$$\mathbf{v} = v_z\hat{\mathbf{z}} + v_r\hat{\mathbf{r}} + v_\zeta\hat{\boldsymbol{\zeta}} \tag{10}$$

Such a grid may be useful if the kinetic neutral code must interface with legacy software that uses a particular spatially aligned coordinate system.

There are, however, major disadvantages to using different velocity space coordinates for the charged particle and neutral particle species. The simple reason for this is that there is a nontrivial transformation between (v_z, v_r, v_ζ) and $(v_\parallel, v_\perp, \vartheta)$. Combined with the collision operator terms in equations (2) and (3), this nontrivial transformation requires a 3D interpolation of distribution function data at every time integration step. Interpolation between 3D grids is necessarily slow and likely to be a source of error unless an advanced interpolation scheme is used (such as a spectral interpolation or cubic spline).

We now write the velocity coordinate transformation for later use. Using equations (8) and (10), one can show that

$$\begin{aligned} v_z(v_\parallel, v_\perp, \vartheta) &= v_\parallel b_z - v_\perp \sin\vartheta b_\zeta \\ v_r(v_\parallel, v_\perp, \vartheta) &= v_\perp \cos\vartheta \\ v_\zeta(v_\parallel, v_\perp, \vartheta) &= v_\parallel b_\zeta + v_\perp \sin\vartheta b_z \end{aligned}, \tag{11}$$

and

$$\begin{aligned} v_\parallel(v_z, v_r, v_\zeta) &= v_z b_z + v_\zeta b_\zeta \\ v_\perp(v_z, v_r, v_\zeta) &= \sqrt{v_r^2 + (v_\zeta b_z - v_z b_\zeta)^2}. \end{aligned} \tag{12}$$

To evaluate the cross-species collision operator terms in equation (2), we require the neutral distribution function $F_n(v_z, v_r, v_\zeta)$ at every (z, r) to be gyroaveraged and given on the $(v_\parallel, v_\perp)$ grid. This is achieved by writing

$$\langle F_n \rangle (v_\parallel, v_\perp) = \int_{-\pi}^{\pi} F_n(v_z(v_\parallel, v_\perp, \vartheta), v_r(v_\parallel, v_\perp, \vartheta), v_\zeta(v_\parallel, v_\perp, \vartheta)) \frac{d\vartheta}{2\pi}. \tag{13}$$

Similarly, to evaluate the charge exchange collision operator terms in equation (3) we require the charged particle distribution function $F_i(v_\parallel, v_\perp)$ at every (z, r) to be interpolated, i.e.,

$$F_i(v_z, v_r, v_\zeta) = F_i(v_\parallel(v_z, v_r, v_\zeta), v_\perp(v_z, v_r, v_\zeta)). \tag{14}$$

We carry out the necessary interpolation using a readily available `Julia` package: `Interpolations.jl` [10]. For simplicity and speed of evaluation we use a linear interpolation suitable for non-uniformly spaced data.

A simple strategy to avoid the cost of interpolation would be to choose to use the same velocity space grid for both charged and neutral particles. In the standard drift-kinetic approach where the moments are not evolved separately, this strategy would only introduce the need to evaluate v_z, v_r, and v_ζ via the transform (11). We have not attempted to implement this approach numerically due to time constraints, although we highlight this approach as a possible future optimization of the model.

4. Boundary conditions on ions and neutral particles

We must specify boundary conditions on the evolved particle distribution function in the coordinates where differential operators appear: z, r and $v_\parallel$. No boundary conditions are required on $v_\perp$, v_z, v_r, or v_ζ, as no advection occurs in these coordinates, and no Fokker-Planck collision operator currently appears in the model. The physical boundary condition in $v_\parallel$ is that

$$F_i(z, r, v_\parallel \to \pm\infty, v_\perp, t) \to 0. \tag{15}$$

This condition may easily be imposed by either forcing zero incoming particles in the $v_\parallel$ advection, or by enforcing that F_i is periodic in $v_\parallel$ at the ends of the $v_\parallel$ domain. This is equivalent to the zero boundary condition if the $v_\parallel$ domain is sufficiently large.

For the boundaries in r, one can imagine several physically interesting scenarios: a periodic boundary condition, in which case the model represents something like a local 'flux tube'; a Dirichlet boundary condition as in a 'global' model; and a limited configuration with conditions that model direct contact to the vacuum vessel. In this report we only consider the first possibility.

Finally, we consider two types of boundary condition in z; periodic boundary conditions, and 'wall' boundary conditions that model the presence of the 'sheaths' at the ends of open field lines on the vacuum vessel or divertor plates. The periodic boundary condition is implemented just as a testing exercise: we do not implement the vorticity equation that is necessary to calculate the potential in closed-field-line geometry [8]. Only the wall boundary condition option is physically consistent. To impose the wall boundary condition, we assume that the entrance to the sheath occurs at the ends of the domain in z. We also assume that ions that exit the simulation domain continue on to the wall, where they recombine. No ions enter the domain from the walls, giving a zero incoming boundary condition for the ions:

$$\begin{aligned} F_i(z = -L_z/2, v_\parallel > E_r/B_z, v_\perp, t) &= 0 \\ F_i(z = L_z/2, v_\parallel < E_r/B_z, v_\perp, t) &= 0. \end{aligned} \tag{16}$$

Note that in the case that $E_r = 0$ then the condition (16) reduces to the equivalent 1D condition. The boundaries of the domain (corresponding to the sheath entrances) are taken to be at $z = -L_z/2$ and $z = L_z/2$. Neutrals that leave the domain are assumed to hit the wall and thermalise at the temperature of the wall, T_w. Ions that recombine at the wall also re-enter as neutrals. The resulting boundary condition on the neutrals

is

$$\begin{aligned} F_n(z=-L_z/2, v_z>0, v_r, v_\zeta, t) &= \left(\Gamma_{i,-L_z/2}+\Gamma_{n,-L_z/2}\right) F_{Kw}\left(v_z, \sqrt{v_r^2+v_\zeta^2}\right), \\ F_n(z=L_z/2, v_z<0, v_r, v_\zeta, t) &= \left(\Gamma_{i,L_z/2}+\Gamma_{n,L_z/2}\right) F_{Kw}\left(v_z, \sqrt{v_r^2+v_\zeta^2}\right), \end{aligned} \tag{17}$$

where

$$F_{Kw}(v_z, v_t) \doteq \frac{3}{\pi}\left(\frac{m_i}{2T_w}\right)^2 \frac{|v_z|}{\sqrt{v_z^2+v_t^2}} \exp\left(-\frac{m_i\left(v_z^2+v_t^2\right)}{2T_w}\right) \tag{18}$$

is the Knudsen cosine distribution [11], and

$$\Gamma_{i,-L_z/2} \doteq 2\pi \int_{-\infty}^{E_r/B} dv_\parallel \int_0^\infty dv_\perp v_\perp \left| b_z v_\parallel - \frac{E_r}{B}\right| F_i(z=-L_z/2, r, v_\parallel, v_\perp, t) \tag{19}$$

$$\Gamma_{n,-L_z/2} \doteq \int_{-\infty}^0 dv_z \int_{-\infty}^\infty dv_r \int_{-\infty}^\infty dv_\zeta \, |v_z| \, F_n(z=-L_z/2, r, v_z, v_r, v_\zeta, t) \tag{20}$$

and

$$\Gamma_{i,L_z/2} \doteq 2\pi \int_{E_r/B}^{\infty} dv_\parallel \int_0^\infty dv_\perp v_\perp \left| b_z v_\parallel - \frac{E_r}{B}\right| F_i(z=L_z/2, v_\parallel, v_\perp, t) \tag{21}$$

$$\Gamma_{n,L_z/2} \doteq \int_0^{\infty} dv_z \int_{-\infty}^\infty dv_r \int_{-\infty}^\infty dv_\zeta \, |v_z| \, F_n(z=L_z/2, v_z, v_r, v_\zeta, t) \tag{22}$$

are the combined fluxes of neutrals and ions towards the walls at $z=-L_z/2$ and $z=L_z/2$, respectively. In the 1D model implementation [5], where $E_r=0$, we note that the wall boundary condition is imposed on the marginalised distributions functions $f_i = 2\pi\int dv_\perp v_\perp F_i$ and $f_n = \int dv_\zeta dv_r F_n$. Whilst the 1D and 2D implementations are able to share almost all subroutines related to the imposition of the wall boundary condition, we note that this slight difference between the 1D and 2D implementations means that the marginalised distribution of neutral particles f_n must be proportional to the marginalised Knudsen distribution $f_{Kw} = 2\pi\int dv_\perp v_\perp F_{Kw}$ at the wall. Please see sections 2 and 2.1 of [5] for further details.

5. A simple electron sheath model

In addition to the boundary conditions for the kinetic species described in the last section, we have implemented a simple electron sheath model that is compatible with a Boltzmann response in the bulk plasma [12]. This sheath model allows us to impose a wall potential ϕ_W, in place of a fixed-in-time constant density N_e in equation (6).

The model is obtained as follows. First, we demand that there is zero net current to the wall plate at the boundaries in z, i.e., the current parallel to the magnetic field line vanishes:

$$J_\parallel = J_{\parallel,i} + J_{\parallel,e} = 0 \text{ at } z = \pm L_z/2 \tag{23}$$

where $J_{\parallel,e}$ and $J_{\parallel,i}$ are the electron and ion contributions to the current, respectively. Second, we assume that the electron distribution function is a Maxwellian close to the entrance of the sheath, i.e.,

$$F_e \approx \frac{n_e}{\pi^{3/2}} \left(\frac{m_e}{2T_e}\right)^{3/2} \exp\left(-\frac{m_e(v_\parallel^2 + v_\perp^2)}{2T_e}\right). \tag{24}$$

We assume that the deviation in F_e from the Maxwellian is due to the high velocity tail of electrons that are absorbed by the wall. The electron distribution function changes with z in the sheath according to the illustration in figure 2. In particular, we note that

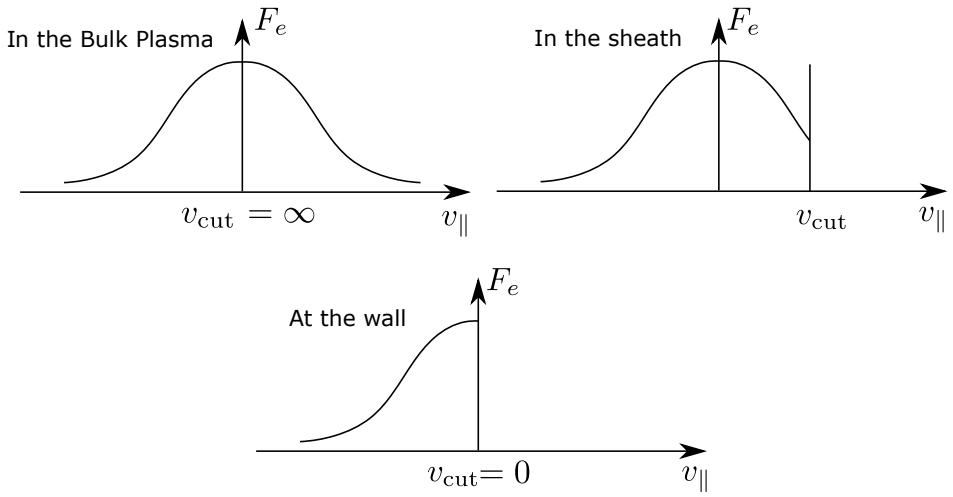


Figure 2: An illustration of the assumed form of the electron distribution function F_e as a function of $v_\parallel$ near the wall at $z = -L_z/2$. There is a high velocity tail of electrons that reach the wall despite the repelling wall potential, as as a result, are lost to the distribution. The cut off velocity v_{cut} drops to zero at the wall itself, yielding the half-Maxwellian that is used to compute $J_{\parallel,e}$ for (25). This argument is developed in full in [12].

any electrons that have reached the wall are unable to return. Third, we assume that $J_{\parallel,e}$, T_e and N_e are constant across the sheath, and that the Boltzmann response (6) continues to hold. Hence, by integrating the distribution function of electrons (24) at the wall at $z = -L_z/2$, we obtain a relationship between $J_{\parallel,e}$ and N_e, T_e and ϕ_W:

$$J_{\parallel,e} = en_e\sqrt{\frac{T_e}{2\pi m_e}} = eN_e \exp\left(\frac{e\phi_W}{T_e}\right)\sqrt{\frac{T_e}{2\pi m_e}} \tag{25}$$

Finally, we rearrange equation (25) for N_e, with the result

$$N_e = -\sqrt{\frac{4\pi m_e}{m_i}} \exp\left(-\frac{e\phi_W}{T_e}\right)\frac{J_{\parallel,i}}{ec_s}. \tag{26}$$

Equation (26) determines the constant N_e to be used in equation (6).

norm. variable	definition
$\tilde{t}$	$t(c_{\rm ref}/L_{\rm ref})$
$\tilde{z}$	$z/L_{\rm ref}$
$\tilde{r}$	$r/L_{\rm ref}$
$\tilde{v}_{\parallel}$	$v_{\parallel}/c_{\rm ref}$
$\tilde{v}_{\perp}$	$v_{\perp}/c_{\rm ref}$
$\tilde{v}_z$	$v_z/c_{\rm ref}$
$\tilde{v}_r$	$v_r/c_{\rm ref}$
$\tilde{v}_{\zeta}$	$v_{\zeta}/c_{\rm ref}$
$\widetilde{N}_e$	$N_e/n_{\rm ref}$
$\tilde{n}_s$	$n_s/n_{\rm ref}$
$\widetilde{\phi}$	$e\phi/T_{\rm ref}$
$\widetilde{E}_r$	$eL_{\rm ref}E_r/T_{\rm ref}$
$\widetilde{E}_z$	$eL_{\rm ref}E_z/T_{\rm ref}$
$\tilde{R}_{\rm in}$	$R_{\rm in}(n_{\rm ref}L_{\rm ref}/c_{\rm ref})$
$\tilde{R}_{\rm ion}$	$R_{\rm ion}(n_{\rm ref}L_{\rm ref}/c_{\rm ref})$
$\widetilde{F}_s$	$F_s(c_{\rm ref}^3\pi^{3/2}/n_{\rm ref})$
$\widetilde{S}_s$	$S_s(c_{\rm ref}^3\pi^{3/2}L_{\rm ref}/n_{\rm ref}c_{\rm ref})$
ρ_*	$c_{\rm ref}/L_{\rm ref}\Omega_{\rm ref}$
b_z	B_z/B

ref. quantity	definition
$L_{\rm ref}$	ref. length (m)
$T_{\rm ref}$	ref. temperature (KeV)
$n_{\rm ref}$	ref. density ($\rm m^{-3}$)
$c_{\rm ref}$	$\sqrt{2T_{\rm ref}/m_i}$ ($\rm ms^{-1}$)
m_i	ion mass (kg)
$B_{\rm ref}$	ref. B (T)
$\Omega_{\rm ref}$	$eB_{\rm ref}/m_i$ ($\rm s^{-1}$)

Table 1: Definitions for normalised and reference quantities used in the report.

6. Normalisation of the system of model equations

We now define the normalisations used when solving numerically the model equations (2)-(7). We define normalised and reference quantities in Table 1. In terms of the normalised variables, the model system equations takes the following form. Equation (2) takes the form

$$\frac{\partial\widetilde{F}_i}{\partial\tilde{t}}+\left(b_z\tilde{v}_{\parallel}-\frac{\rho_*}{2}\widetilde{E}_r\right)\frac{\partial\widetilde{F}_i}{\partial\tilde{z}}+\frac{\rho_*}{2}\widetilde{E}_z\frac{\partial\widetilde{F}_i}{\partial\tilde{r}}+\frac{b_z\widetilde{E}_z}{2}\frac{\partial\widetilde{F}_i}{\partial\tilde{v}_{\parallel}}= \\ -\tilde{R}_{\rm in}\left(\widetilde{n}_n\widetilde{F}_i-\widetilde{n}_i\left\langle\widetilde{F}_n\right\rangle\right)+\tilde{R}_{\rm ion}\widetilde{n}_e\left\langle\widetilde{F}_n\right\rangle+\widetilde{S}_i, \tag{27}$$

where $\rho_*=c_{\rm ref}/L_{\rm ref}\Omega_{\rm ref}$ is a small parameter that measures the size of the ion gyro-orbits compared to the macroscopic size of the system. Note that b_z is also a small

parameter in the formal ordering of the 2D model [7, 8]. Equation (3) takes the form

$$\frac{\partial \widetilde{F}_n}{\partial \tilde{t}} + \tilde{v}_z \frac{\partial \widetilde{F}_n}{\partial \tilde{z}} + \tilde{v}_r \frac{\partial \widetilde{F}_n}{\partial \tilde{r}} = -\tilde{R}_{\text{in}} \left(\widetilde{n}_i \widetilde{F}_n - \widetilde{n}_n \widetilde{F}_i \right) - \tilde{R}_{\text{ion}} \widetilde{n}_e \widetilde{F}_n + \widetilde{S}_n, \tag{28}$$

and quasineutrality, equation (6), takes the form

$$\widetilde{n}_i = \widetilde{n}_e = \widetilde{N}_e \exp\left(\frac{\widetilde{\phi}}{\widetilde{T}_e} \right). \tag{29}$$

The normalised electric fields are defined by

$$\widetilde{E}_z = -\frac{\partial \widetilde{\phi}}{\partial \tilde{z}}, \text{ and } \quad \widetilde{E}_r = -\frac{\partial \widetilde{\phi}}{\partial \tilde{r}}. \tag{30}$$

and the densities have the following definitions

$$\widetilde{n}_i = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} d\tilde{v}_{\parallel} \int_0^{\infty} 2 d\tilde{v}_{\perp} \tilde{v}_{\perp} \widetilde{F}_i. \tag{31}$$

and

$$n_n = \frac{1}{\pi^{3/2}} \int_{-\infty}^{\infty} d\tilde{v}_{\zeta} \int_{-\infty}^{\infty} d\tilde{v}_r \int_{-\infty}^{\infty} d\tilde{v}_z \widetilde{F}_n. \tag{32}$$

The boundary conditions (15) and (16) carry over trivially. The wall boundary condition on the neutrals takes the form

$$\begin{aligned} \widetilde{F}_n(\tilde{z} = -L_z/2L_{\text{ref}}, \tilde{v}_z > 0, \tilde{v}_r, \tilde{v}_{\zeta}, \tilde{t}) &= \left(\widetilde{\Gamma}_{i,-L_z/2} + \widetilde{\Gamma}_{n,-L_z/2} \right) \widetilde{F}_{Kw} \left(\tilde{v}_z, \sqrt{\tilde{v}_r^2 + \tilde{v}_{\zeta}^2} \right), \\ \widetilde{F}_n(\tilde{z} = L_z/2L_{\text{ref}}, \tilde{v}_z < 0, \tilde{v}_r, \tilde{v}_{\zeta}, \tilde{t}) &= \left(\widetilde{\Gamma}_{i,-L_z/2} + \widetilde{\Gamma}_{n,-L_z/2} \right) \widetilde{F}_{Kw} \left(\tilde{v}_z, \sqrt{\tilde{v}_r^2 + \tilde{v}_{\zeta}^2} \right), \end{aligned} \tag{33}$$

where

$$\widetilde{F}_{Kw} = \frac{3\sqrt{\pi}}{\widetilde{T}_w^2} \frac{|\tilde{v}_z|}{\sqrt{\tilde{v}_z^2 + \tilde{v}_t^2}} \exp\left(-\frac{(\tilde{v}_z^2 + \tilde{v}_t^2)}{\widetilde{T}_w} \right) \tag{34}$$

is the normalised Knudsen cosine distribution [11], and the normalised particle fluxes are

$$\widetilde{\Gamma}_{i,-L_z/2} \doteq \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\rho_* \widetilde{E}_r/2} d\tilde{v}_{\parallel} \int_0^{\infty} 2 d\tilde{v}_{\perp} \tilde{v}_{\perp} \left| b_z \tilde{v}_{\parallel} - \frac{\rho_*}{2} \widetilde{E}_r \right| \widetilde{F}_i(\tilde{z} = -L_z/2L_{\text{ref}}, \tilde{r}, \tilde{v}_{\parallel}, \tilde{v}_{\perp}, \tilde{t}) \tag{35}$$

$$\widetilde{\Gamma}_{n,-L_z/2} \doteq \frac{1}{\pi^{3/2}} \int_{-\infty}^{0} d\tilde{v}_z \int_{-\infty}^{\infty} d\tilde{v}_r \int_{-\infty}^{\infty} d\tilde{v}_{\zeta} \left| \tilde{v}_z \right| \widetilde{F}_n(\tilde{z} = -L_z/2L_{\text{ref}}, \tilde{r}, \tilde{v}_z, \tilde{v}_r, \tilde{v}_{\zeta}, \tilde{t}) \tag{36}$$

and

$$\widetilde{\Gamma}_{i,L_z/2} \doteq \frac{1}{\sqrt{\pi}} \int_{\rho_* \widetilde{E}_r/2}^{\infty} d\tilde{v}_{\parallel} \int_0^{\infty} 2 d\tilde{v}_{\perp} \tilde{v}_{\perp} \left| b_z \tilde{v}_{\parallel} - \frac{\rho_*}{2} \widetilde{E}_r \right| \widetilde{F}_i(\tilde{z} = L_z/2L_{\text{ref}}, \tilde{r}, \tilde{v}_{\parallel}, \tilde{v}_{\perp}, \tilde{t}) \tag{37}$$

$$\widetilde{\Gamma}_{n,L_z/2} \doteq \frac{1}{\pi^{3/2}} \int_0^{\infty} d\tilde{v}_z \int_{-\infty}^{\infty} d\tilde{v}_r \int_{-\infty}^{\infty} d\tilde{v}_{\zeta} \left| \tilde{v}_z \right| \widetilde{F}_n(\tilde{z} = L_z/2L_{\text{ref}}, \tilde{r}, \tilde{v}_z, \tilde{v}_r, \tilde{v}_{\zeta}, \tilde{t}) \tag{38}$$

Finally, the electron sheath model for N_e, equation (26), takes the normalised form

$$\widetilde{N}_e = -\sqrt{\frac{4\pi m_e}{m_i}} \exp\left(-\frac{\widetilde{\phi}_w}{\widetilde{T}_e} \right) \frac{J_{\parallel,i}}{e n_{\text{ref}} c_s}. \tag{39}$$

7. Numerical implementation

The algorithms used to implement the 2D model are identical to those used to implement the 1D model described in previous reports. To avoid repetition, we simply list these algorithms here, with further details left to previous reports [2, 3, 4, 5].

We evolve the normalised systems of equations with a time-advance scheme. To maximise the efficiency of the algorithm with regards to numerical stability, we utilise a Strong Stability Preserving (SSP) Runge-Kutta (RK) scheme [13, 14, 15]. By default, we use the four-stage SSPRK3 method, although two and three stage SSPRK schemes are options in the code. The implementation of the time-stepping algorithm was tested in the 1D-1V code (figures 1 and 2 of [3]).

For the spatial and velocity discretisation both finite difference and Chebyshev spectral methods are implemented. For the finite difference option, a uniformly spaced grid is used. Derivatives are evaluated with the third order upwind scheme [16] and integration is carried out using the composite Simpson's rule [3]. For the Chebyshev spectral elements option, the coordinate grid is the Gauss-Chebyshev-Lobatto grid on each element [17]. Inclusion of the endpoints within each element facilitates enforcement of continuity at element boundaries, and the use of Chebyshev polynomials as a basis enables the use of Fast Fourier Transforms. In our code, these transforms are done using the widely-used FFTW library [18]. The associated integration weights used for integration are obtained using Clenshaw-Curtis quadrature rules [19]. Clenshaw-Curtis quadrature allows for the use of endpoints in the integration domain while still exactly integrating polynomials up to degree $N_{grid} - 1$, with N_{grid} the number of points within the element. The implementation of the spatial discretisation was tested in the 1D-1V code (figures 3, 4 and 5 of [3]).

The scale of the 2D-3V model, having 5 dimensions for the neutral species, and 4 for the ion species, requires a larger memory requirement than can be supported by a single core of a typical computer. To reach large enough resolutions to achieve the results presented in this report, it was necessary to implement shared memory OpenMPI support [20].

The previous implementation of the 1D-1V code [2, 3, 4, 5] allowed for the development of a suite of automatic functional tests. The tests come in two classes. In the first class, the automated 1D-1V tests check the numerical results for sound wave damping of small amplitude fluctuations against analytically obtained results [21, 4]. In the second class of test, results from converged 'nonlinear' simulations with finite fluctuation amplitudes are stored and tested against the output of the code after a new revision. The current 2D-3V version of the code passes all previous 1D-1V automated tests, suggesting that the time-stepping and spatial discretisation algorithms have been preserved in the generalisation of the code to the 2D-3V case.

All revisions of the 'moment kinetics' code are written in the Julia programming language, and are currently available on GitHub at `https://github.com/mabarnes/moment_kinetics`. The latest 2D-3V code is held in the branch `https://github.com/`

`mabarnes/moment_kinetics/tree/radial-vperp-standard-DKE-with-neutrals`.

8. Testing the numerical implementation with the method of manufactured solutions

The normalised system of equations (27)-(32) is difficult to solve analytically, especially with the wall boundary conditions described in sections 4-5. This presents a challenge for testing the numerical implementation of the 2D model. To overcome this challenge we use a standard technique from the fluid mechanics community: the method of manufactured solutions. The great power of the method of manufactured solutions is that tests derived from this method can be tuned to focus on individual operators, or the whole code, as desired. In the following, we present tests which can be used separately to cover the time advance algorithm, the spatial advection, the velocity advection, the collisions, periodic boundary conditions, and the wall boundary conditions. New tests can be developed as features are added to the model.

We use the following method to test the 2D model equations with manufactured solutions [22]. First, we propose manufactured, target, solutions $\widetilde{F}_i^{MS}$ and $\widetilde{F}_n^{MS}$ that we can write down in closed form with known functions. The manufactured solutions are only restricted to obey the boundary conditions that are imposed on the system of equations. Second, we compute manufactured sources $\widetilde{S}_i$ and $\widetilde{S}_n$ by inserting the manufactured solutions $\widetilde{F}_i^{MS}$ and $\widetilde{F}_n^{MS}$ into equations (27) and (28) and evaluating the necessary derivatives and velocity integrals analytically. Third, we perform a numerical simulation using the manufactured sources. We run the simulation for a fixed time $\tilde{t} = O(1)$ (to allow numerical errors to accrue) for varying resolutions. We check that the numerical solution is consistent with the target manufactured solution by confirming that the numerical error reduces towards zero as the resolution increases. To measure the error between the numerical solution and the target manufactured solution, we introduce the following measures. We define an error on the ion and neutral species density

$$\epsilon(\widetilde{n}_s) = \sqrt{\frac{\sum_{i,j} |\widetilde{n}_s(z_i, r_j) - \widetilde{n}_s^{MS}(z_i, r_j)|^2}{N_r N_z}}, \tag{40}$$

where $\widetilde{n}_s^{MS}$ is the target manufactured density of species s, and N_r and N_z are the total number of points in the r and z grids, respectively. We define an error on the ion distribution function

$$\epsilon(\widetilde{F}_i) = \sqrt{\frac{\sum_{i,j,k,l} |\widetilde{F}_i(v_{\parallel i}, v_{\perp j}, z_k, r_l) - \widetilde{F}_i^{MS}(v_{\parallel i}, v_{\perp j}, z_k, r_l)|^2}{N_{v_\parallel} N_{v_\perp} N_r N_z}}, \tag{41}$$

where $N_{v_\parallel}$ and $N_{v_\perp}$ are the total number of points in the $v_\parallel$ and $v_\perp$ grids, respectively. Finally, we define an error on the neutral distribution function

$$\epsilon(\widetilde{F}_n) = \sqrt{\frac{\sum_{i,j,k,l,m} |\widetilde{F}_n(v_{zi}, v_{rj}, v_{\zeta k}, z_l, r_m) - \widetilde{F}_n^{MS}(v_{zi}, v_{rj}, v_{\zeta k}, z_l, r_m)|^2}{N_{v_z} N_{v_r} N_{v_\zeta} N_r N_z}}, \tag{42}$$

where N_{v_z}, N_{v_r}, and N_{v_ζ} are the total number of points in the v_z, v_r and v_ζ grids, respectively. The errors $\epsilon(\widetilde{n}_s)$, $\epsilon(\widetilde{F}_i)$, and $\epsilon(\widetilde{F}_n)$ measure the average error per point in the array of the numerical solution.

As noted previously [22], `Julia` has support for elements of symbolic algebra via the `Symbolics.jl` package [23, 24]. This allows us to partially automate the process of calculating the manufactured sources. Currently, the `Symbolics.jl` package supports symbolic differentiation, allowing for a great reduction in the effort of developing manufactured solutions tests. Unfortunately, symbolic integration is not yet supported by the `Symbolics.jl` package, so the target manufactured solutions must be chosen to be sufficiently simple to integrate by hand.

8.1. Tests with periodic boundary conditions in r and z

We first describe a manufactured solutions test that is compatible with periodic boundary conditions in r and z. We choose target manufactured solutions that depend on time and space in a nontrivial manner, allowing for tests of the timestepping algorithm and the advection in r, z, and $v_\parallel$. We choose the ion density to be of the form

$$\widetilde{n}_i = \frac{3}{2} + \frac{\sin(2\pi\tilde{t})}{10}\left(\sin\left(\frac{2\pi r}{L_r}\right) + \sin\left(\frac{2\pi z}{L_z}\right)\right). \tag{43}$$

We choose the neutral density to take the form

$$\widetilde{n}_n = \frac{3}{2} + \frac{\sin(2\pi\tilde{t})}{10}\left(\cos\left(\frac{2\pi r}{L_r}\right) + \cos\left(\frac{2\pi z}{L_z}\right)\right). \tag{44}$$

We specify the ion and neutral distributions as Maxwellians with constant temperatures and no flow, i.e.,

$$\widetilde{F}_i = \widetilde{n}_i \exp\left(-\tilde{v}_\parallel^2 - \tilde{v}_\perp^2\right), \tag{45}$$

and

$$\widetilde{F}_n = \widetilde{n}_n \exp\left(-\tilde{v}_z^2 - \tilde{v}_r^2 - \tilde{v}_\zeta^2\right). \tag{46}$$

The gyroaverage of F_n in equation (27) is computed analytically by noting that $\tilde{v}_\parallel^2 + \tilde{v}_\perp^2 = \tilde{v}_z^2 + \tilde{v}_r^2 + \tilde{v}_\zeta^2$. This test could easily be extended to include a specified temperature or mean velocity moment in addition to the density moment.

Having specified target solutions, we must calculate $\widetilde{S}_i$ and $\widetilde{S}_n$ for specific input physics parameters – the target solutions can be used to test cases with or without collisions, and with any specific choice of the helical magnetic field. We first focus on a case with $b_z = 0.5$, $\rho_* = 1.0$, $\widetilde{T}_e = \widetilde{N}_e = 1.0$ and $\tilde{R}_{\text{in}} = \tilde{R}_{\text{ion}} = 0$. We must also describe the numerical input parameters. We use spectral-element coordinate grids for all dimensions [3]. These grids are characterised by N_{grid}, the number of points in each spectral element, and $N_{element}$, the number of elements. The total number of grid points is $N_{grid}^{total} = (N_{grid}-1)N_{element}+1$. We identified that the biggest source of numerical error comes from the velocity grid discretization. In the following, we present tests where we

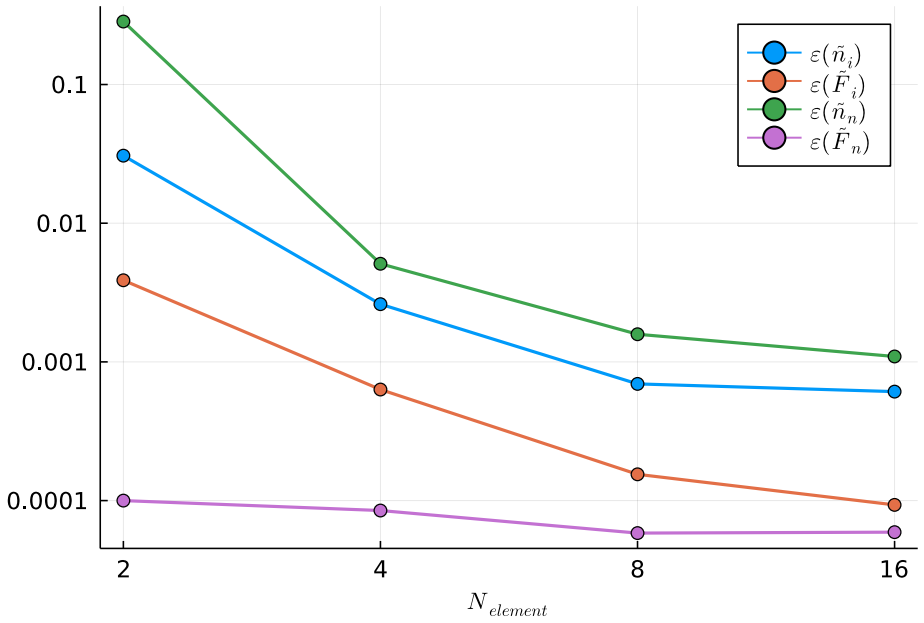


Figure 3: Plot of the measures of the error on the ion and neutral density, and the ion and neutral distribution function, defined in equations, (40), (41), and (42), respectively. The simulations use the manufactured solutions (43)-(46), and retain all but the collision terms in equations (27)-(32). The scan is in the number of elements $N_{element}$ in the spectral-element scheme in the velocity space coordinates $v_\parallel$, $v_\perp$, v_z, v_r, and v_ζ. The relatively small errors observed for $N_{element} = 16$ suggests that the implementation is performing well.

have increased the number of elements $N_{element}$ in the velocity space grids $v_\parallel$, $v_\perp$, v_z, v_r, and v_ζ, holding fixed $N_{grid} = 5$ for all coordinates, and holding fixed $N_{element} = 2$ in the coordinates r and z. We take the maximum velocity $v_{max}/c_{\text{ref}} = 6.0$ on all grids. We simulate for 160 time steps with $\Delta\tilde{t} = 0.002$. The results of these simulations are shown in figure 3. The numerical error in all fields is either small, or reduces for increasing $N_{element}$. At the largest $N_{element} = 16$, the error in the ion and electron densities appears to be converged. The residual error is likely due to the finite size of $\Delta\tilde{t}$, see fig 2 in [3]. The input files for these simulations are provided in Appendix A.1.

In a second test, we now introduce the charge-exchange and ionisation collision operators. We use the input parameters described above, with $\tilde{R}_{\text{in}} = \tilde{R}_{\text{ion}} = 1.0$, and we again scan in the velocity space resolution. The results of the simulations appear in figure 4. We again see convergence with increasing $N_{element}$, but at a slower rate than in the case without collisions. We can attribute the slower convergence to the presence of the linear interpolation in the collision operators, as discussed in section 3. The linear interpolation is likely to be more inaccurate for larger values of N_{grid}, where the spacing of the grid points in each element become more nonuniform. The input files for these simulations are provided in Appendix A.2.

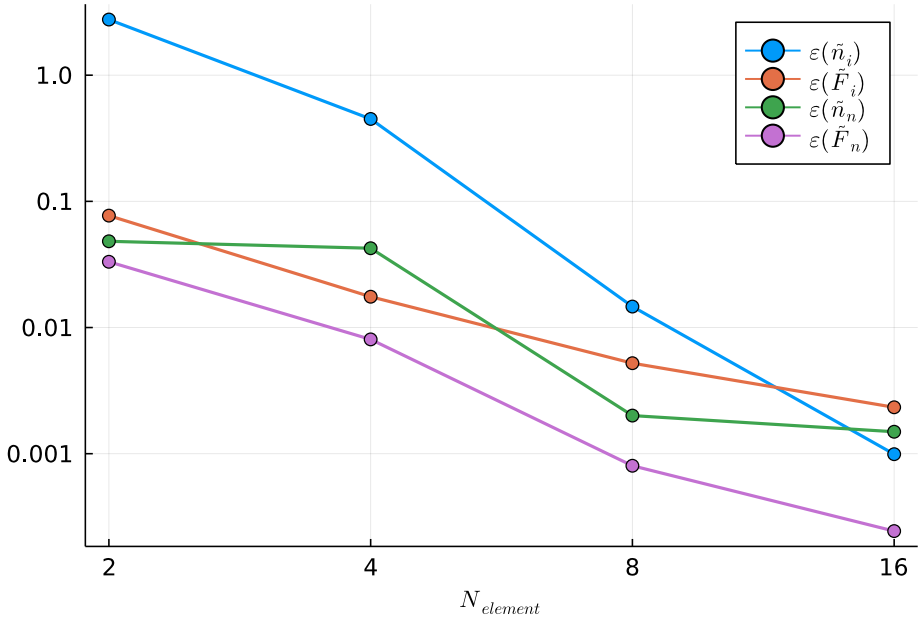


Figure 4: Plot of the measures of the error on the ion and neutral density, and the ion and neutral distribution function, defined in equations, (40), (41), and (42), respectively. The simulations use the manufactured solutions (43)-(46), and retain all terms in equations (27)-(32). The parameter $N_{element}$ is in the number of spectral elements in the velocity space coordinates $v_\parallel$, $v_\perp$, v_z, v_r, and v_ς. The size of the errors observed for $N_{element} = 16$ suggests that the implementation is performing well.

8.2. Tests with wall and sheath boundary conditions in z

We now describe tests for the implementation of the wall and sheath boundary conditions described in sections 4-5. For simplicity, we consider periodic boundary conditions in r. We must construct a target distribution function that satisfies the necessary boundary conditions. The ion distribution function $\widetilde{F}_i$ must satisfy the condition (16). We can achieve this by constructing a distribution function from the velocity coordinate $\overline{v}_\parallel = \tilde{v}_\parallel - \rho_* \widetilde{E}_r / 2b_z$. We choose

$$\widetilde{F}_i = \left[H\left(\overline{v}_\parallel\right) \overline{v}_\parallel^2 \left(\frac{1}{2} + \frac{z}{L_z}\right) n_+(r) + H\left(-\overline{v}_\parallel\right) \overline{v}_\parallel^2 \left(\frac{1}{2} - \frac{z}{L_z}\right) n_-(r) + \left(\frac{1}{2} - \frac{z}{L_z}\right)\left(\frac{1}{2} + \frac{z}{L_z}\right) n_0(r) \right] \exp\left(-\overline{v}_\parallel^2 - \tilde{v}_\perp^2\right), \tag{47}$$

with $H\left(x\right)$ the Heaviside function, taking the values 1 for $x > 0$ and 0 for $x < 0$. In the Julia implementation $H\left(0\right) = 1/2$. Note that the forward going part of the distribution vanishes at $z = -L_z/2$ and the backward going part vanishes at $z = L_z/2$ – ensuring that condition (16) holds. In principle, we could specify any functions for $n_+(r)$, $n_-(r)$

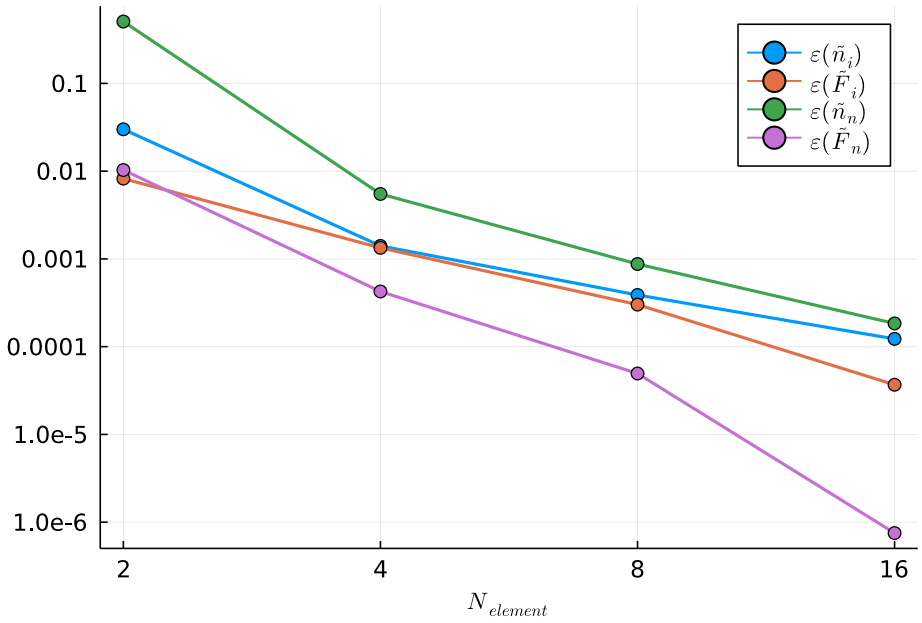


Figure 5: Plot showing the error measures on the ion and neutral densities, and the ion and neutral distribution functions for a test case with wall boundary conditions in z, no radial coordinate (meaning $\widetilde{E}_r = v_r = 0$), and the full ion and neutral velocity spaces. The target manufactured solutions (47)-(53) are used. Electrons are taken to be Boltzmann with $\widetilde{N}_e = 1.0$, and no collisions are included in the simulation. Convergence of the errors is observed for increasing number $N_{element}$ of spectral elements in the velocity space coordinates.

and $n_0(r)$ that satisfy the radial boundary conditions. We take $n_+ = n_- = n_0$, with

$$n_0 = 1 + \frac{1}{20}\sin\left(\frac{2\pi r}{L_r}\right). \tag{48}$$

We choose the radially varying component of n_0 to be relatively small to ensure that any $\mathbf{E}\times\mathbf{B}$ drifts are easy to resolve with $v_{max}/c_{\text{ref}} = 6.0$ – note that radial variation in n_0 induces radial variation in n_i, and hence, in ϕ, thus generating an $E_r \neq 0$. This choice is necessary, because the manufactured solution is proportional to a shifted Gaussian in $v_\parallel$, with the shift proportional to E_r/B. If E_r/B is set too large, then the bulk of the Gaussian is off grid.

The Julia `Symbolics.jl` package does not currently support symbolic or numerical integration. To specify the ion density, needed to compute the potential via quasineutrality, we compute the velocity integral of equation (47) analytically, with the result

$$\widetilde{n}_i = \frac{n_+(r)}{4}\left(\frac{1}{2}+\frac{z}{L_z}\right) + \frac{n_-(r)}{4}\left(\frac{1}{2}-\frac{z}{L_z}\right) + n_0(r)\left(\frac{1}{2}+\frac{z}{L_z}\right)\left(\frac{1}{2}-\frac{z}{L_z}\right). \tag{49}$$

To write down the neutral target distribution function for the MMS test, we also require

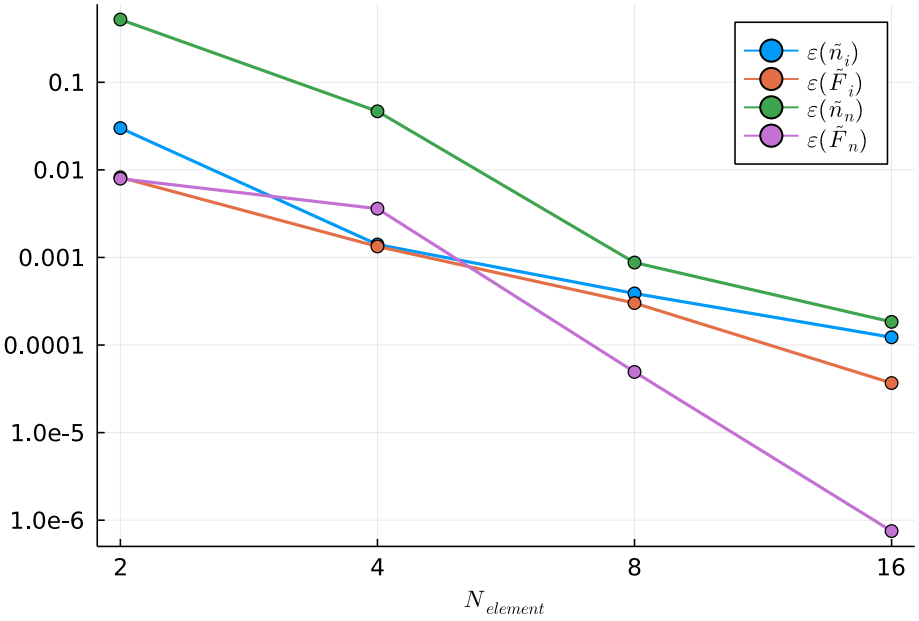


Figure 6: Plot showing the error measures on the ion and neutral densities, and the ion and neutral distribution functions for a test case with wall boundary conditions in z, no radial coordinate (meaning $\widetilde{E}_r = v_r = 0$), and the full ion and neutral velocity spaces. The target manufactured solutions (47)-(53) are used. Electrons are taken to be Boltzmann with $\widetilde{N}_e$ determined by the sheath through equation (39), and no collisions are included in the simulation. Convergence of the errors is observed for increasing number $N_{element}$ of spectral elements in the velocity space coordinates.

the particle fluxes $\widetilde{\Gamma}_{i,-L_z/2}$ and $\widetilde{\Gamma}_{i,L_z/2}$. We find that

$$\widetilde{\Gamma}_{-L_z/2} = \frac{b_z n_-(r)}{2\sqrt{\pi}}, \quad \text{and } \widetilde{\Gamma}_{L_z/2} = \frac{b_z n_+(r)}{2\sqrt{\pi}}. \tag{50}$$

For the neutral particles we can specify any target distribution function that satisfies the condition (33). We choose

$$\begin{aligned}\widetilde{F}_n =& H\left(\tilde{v}_z\right)\left[\widetilde{\Gamma}_{i,-L_z/2}\left(\frac{1}{2}-\frac{z}{L_z}\right)^2+\widetilde{\Gamma}_n\right]\widetilde{F}_{Kw}\left(\tilde{v}_z,\sqrt{\tilde{v}_r^2+\tilde{v}_\zeta^2}\right)\\ &+H\left(-\tilde{v}_z\right)\left[\widetilde{\Gamma}_{i,L_z/2}\left(\frac{1}{2}+\frac{z}{L_z}\right)^2+\widetilde{\Gamma}_n\right]\widetilde{F}_{Kw}\left(\tilde{v}_z,\sqrt{\tilde{v}_r^2+\tilde{v}_\zeta^2}\right)\end{aligned}. \tag{51}$$

The neutral particle flux $\widetilde{\Gamma}_n$ is a constant. That we can identify this constant with the particle flux is demonstrated by noting that

$$\frac{1}{\pi^{3/2}}\int_{-\infty}^{0} d\tilde{v}_z \int_{-\infty}^{\infty} d\tilde{v}_r \int_{-\infty}^{\infty} d\tilde{v}_\zeta \left|\tilde{v}_z\right| \widetilde{F}_{Kw}\left(\tilde{v}_z,\sqrt{\tilde{v}_r^2+\tilde{v}_\zeta^2}\right) = 1. \tag{52}$$

We must also evaluate the neutral density. We find that

$$\widetilde{n}_n = \frac{3\sqrt{\pi}}{4\widetilde{T}_w^{1/2}} \left(\widetilde{\Gamma}_{i,-L_z/2} \left(\frac{1}{2} - \frac{z}{L_z} \right)^2 + \widetilde{\Gamma}_{i,L_z/2} \left(\frac{1}{2} + \frac{z}{L_z} \right)^2 + 2\widetilde{\Gamma}_n \right). \tag{53}$$

To perform these integrals, we used the identities

$$\begin{aligned} &\int_0^\infty \frac{u}{\sqrt{x^2+u^2}} \exp\left(-\frac{u^2}{v^2}\right)\, du = \frac{\sqrt{\pi}}{2} v \operatorname{erfc}\left(\frac{x}{v}\right) \exp\left(\frac{x^2}{v^2}\right), \\ &\int_0^\infty x \operatorname{erfc}(x)\, dx = \frac{1}{4}, \quad \text{and} \quad \int_0^\infty x^2 \operatorname{erfc}(x)\, dx = \frac{1}{3\sqrt{\pi}}. \end{aligned} \tag{54}$$

The above target manufactured solutions can be used to test the implementation of the wall and sheath boundary conditions. We first describe a test for the wall boundary condition in z in isolation from the radial coordinate and sheath boundary condition. For our physics parameters we take $\widetilde{N}_e = 1.0$ (obviating the need for an electron sheath), $b_z = 1.0$, $\widetilde{T}_e = \widetilde{T}_w = 1.0$ and $\tilde{R}_{\text{in}} = \tilde{R}_{\text{ion}} = 0$. For the numerical input we take $\Delta\tilde{t} = 0.001$, 200 time steps, $N_{grid} = 1$ and $N_{element} = 1$ for r (so that no radial advection is carried out), and $N_{grid} = 6$ for all other dimensions. We take $N_{element} = 2$ for z, and we vary $N_{element}$ for the velocity space coordinates. The resulting error measures from these simulations are provided in figure 5. We note the rapid convergence of the four error measures as $N_{element}$ increases. We also note that it is important for the integration scheme that an even number of $N_{element}$ is used. This is to make sure that the discontinuity in the neutral distribution $\widetilde{F}_{Kw}$ appears at an element boundary. If an odd number of elements were utilised, no convergence would be observed.

We can use an identical test setup to check implementation of the electron sheath boundary condition. To evaluate the ion current at the wall we use the fluxes evaluated in equation (50). Instead of specifying $\widetilde{N}_e = 1.0$ as a constant, we calculate $\widetilde{N}_e$ via equation (39). The resulting numerical errors from this series of simulations is shown in figure 6. We see almost (but not quite) identical results as in figure 5. As expected from the reports on the 1D wall boundary condition in the 1V case [5], the implementation handles well the wall boundary condition in the 3V case, providing that the velocity space resolution is high enough. The input files for the simulations in the 1D-3V wall tests are provided in Appendix A.3 and Appendix A.4.

Finally, we describe the results of testing the wall boundary condition in a case with both z and r spatial dimensions. To avoid complication in evaluating the electrostatic potential, in this test we revert to a simple Boltzmann response for electrons with $\widetilde{N}_e = 1.0$. We take $N_{grid} = 6$ and $N_{element} = 2$ for the r dimension, and we take $\rho_* = 1.0$. Otherwise, we use the physics and numerical input parameters described above. We again vary the number of velocity space spectral elements. We plot the resulting error measures from the simulations in figure 7. In contrast to the other tests presented in this report, figure 7 does not show good convergence as $N_{element}$ increases. In fact, more detailed investigations reveal that the error results from deviations between the numerical and symbolic solutions at the wall boundary, see figures 8 and 9. This

suggests that the numerical integration scheme is not handling well the discontinuities in the ion distribution function that now depend on $\widetilde{E}_r$. It is worth noting that the electric fields show larger errors than the ion density, indicating that problems with the spatial differentiation may also contribute to the issue highlighted here. Further investigations are required to determine an adequate solution to this problem. The input files for the simulations in the 2D-3V wall tests are provided in Appendix A.5.

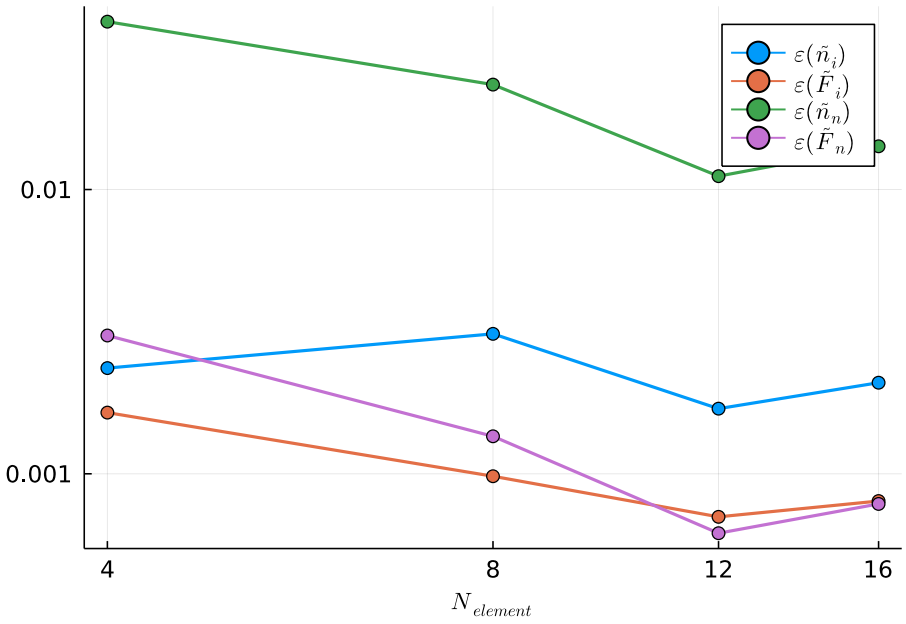
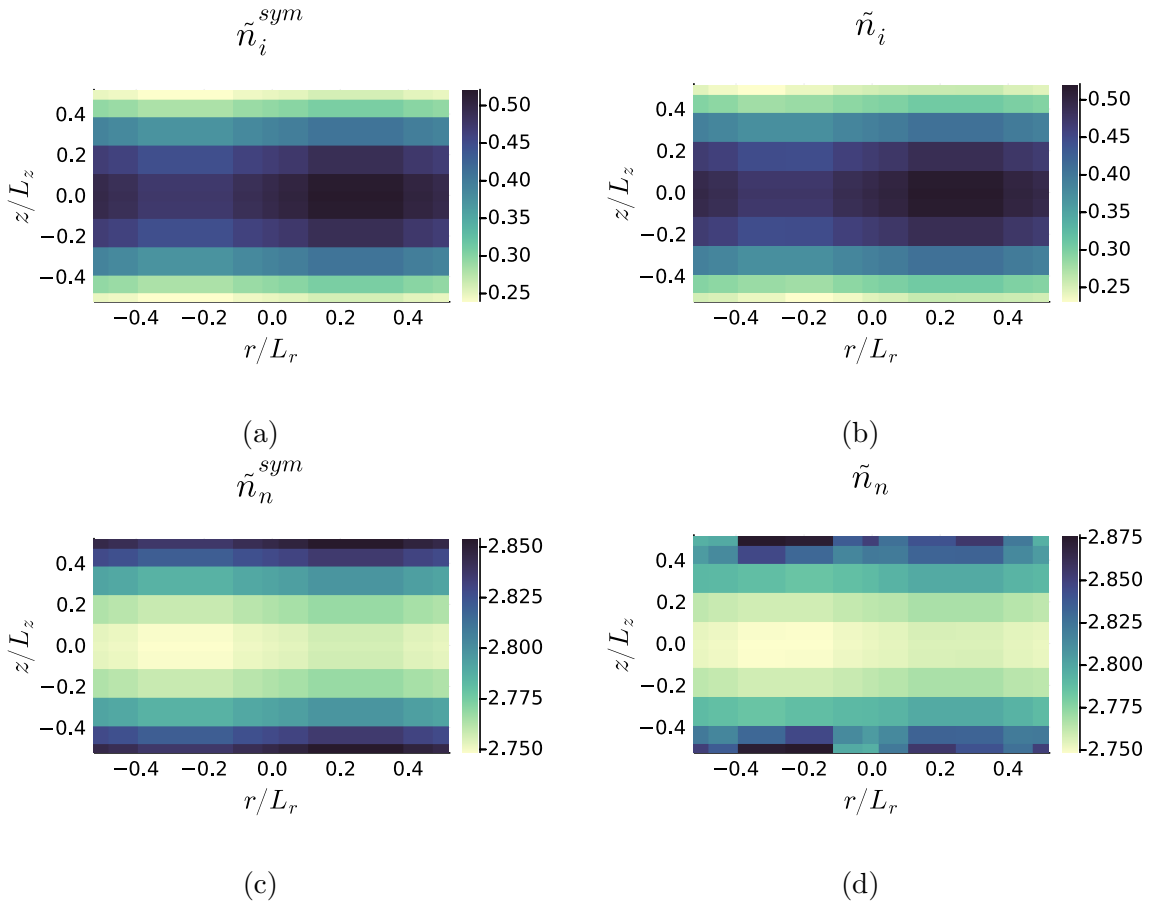


Figure 7: Plot showing the error measures on the ion and neutral densities, and the ion and neutral distribution functions for a test case with wall boundary conditions in z, periodic boundary conditions in r, and the full ion and neutral velocity spaces. Equations (27)-(32) are solved with a Boltzmann electron response and $\widetilde{N}_e = 1.0$. Collisions are removed by setting $\tilde{R}_{\text{in}} = \tilde{R}_{\text{ion}} = 0.0$. The target manufactured solutions (47)-(53) are used. Convergence of the errors is not observed for increasing number $N_{element}$ of spectral elements in the velocity space coordinates, suggesting an underlying problem with the implementation of the numerical scheme.

9. Conclusions and future plans

The work presented in this report consists of two parts. First, the implementation of, and description of, a 2D (drift-) kinetic model that includes both ion and neutral species and wall boundary conditions. Second, the development of manufactured solutions tests that can be used to check the accuracy of the implementation.

To our knowledge, the model presented here is unique in the plasma physics community in supporting drift-kinetic ions, fully kinetic neutrals, and wall boundary conditions for both species. The code has the potential to be developed into a leading



(a) (b) (c) (d)

Figure 8: Plots showing moments at the last timestep of the simulation with $N_{element} = 16$ in figure 7. We plot (a) the symbolic, target ion density n_i, (b) the numerically computed ion density n_i, (c) the symbolic, target neutral density n_n, (d) the numerically computed neutral density n_n, Note that the scales on each plot differ, and that the error in concentrated near the boundaries in z.

physics simulation software for modelling plasmas in the tokamak edge. Some of the features that would be desirable in generalising the model are available in other codes, such as GKEYLL [25] and COGENT [26], making it plausible that our model could be developed in a relatively short timeframe. Important features to be developed in a generalisation of the model could include the following: a fluid or kinetic electron model; a vorticity equation for computing the electric field in the closed field line regions; nontrivial tokamak geometry, including a separatrix; more complicated collision models including ion-ion collisions; more accurate sheath models; and electromagnetic fluctuations.

The manufactured solution tests that we have developed here were critical in the rapid implementation of the code, allowing us to quickly check the correct implementation of new features as they were added. For example, in the results shown in this report, the tests have revealed the difficulties that the numerical integration

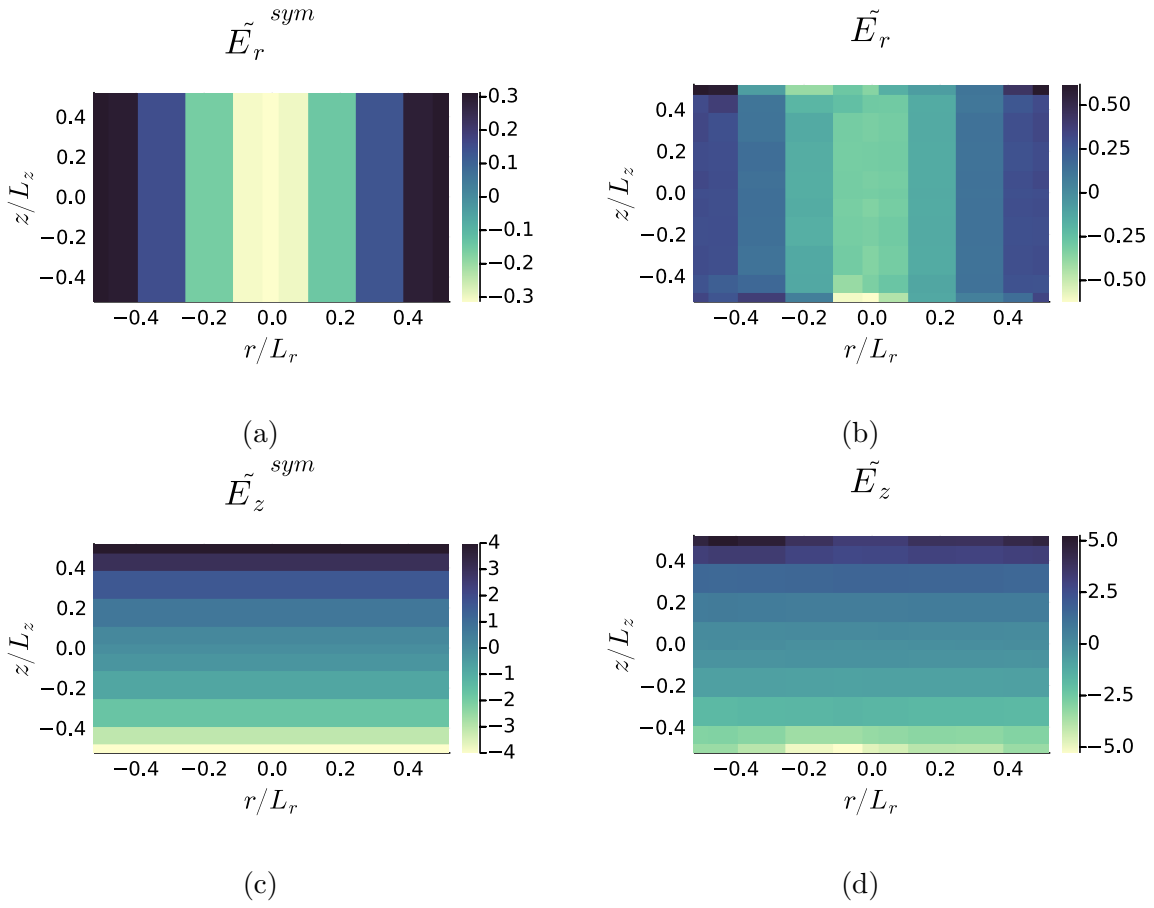


Figure 9: Plots showing the fields at the last timestep of the simulation with $N_{element} = 16$ in figure 7. We plot (a) the symbolic, target E_r, (b) the numerically computed E_r, and finally (c) the symbolic, target E_z and (d) the numerically computed E_z. Note that the scales on each plot differ, and that the error in concentrated near the boundaries in z, and are larger in the electric fields than the moments shown in figure 9.

algorithm has with dealing with the wall boundary condition – the wide coverage of the implemented tests means that we should be able to easily pinpoint the area of the algorithm that must be addressed. The results from the manufactured solutions tests are not only useful in determining if there is a problem with the algorithm: the tests also give information about the resolution needed to resolve different types of position space or velocity space features.

The main priority for future development of the 2D implementation should be to track down and fix the numerical issue that leads to poor convergence with velocity space resolution when wall boundary conditions are imposed, see figure 7 and the corresponding discussion. Preliminary evidence suggests that the error is due to the treatment of the discontinuity of the ion distribution function at $z = \pm L_z/2$ and $v_\parallel - E_r/B_z = 0$ that is introduced by the wall boundary condition. However, the dominant errors are observed in the electric fields and the ion and neutral densities,

rather than in the distribution function itself. Determining the source of these numerical errors is the object of future work.

The presence of the automated functional tests of the 1D-1V code and the manufactured solutions tests in the 2D-3V code will make this software attractive to scientists in the plasma physics community that desire a robust simulation tool with automated self-testing. With this in mind, we have made a start in automating the manufactured solutions tests. In the ideal case, a manufactured solution test would be run every time a revision was made to the code, and the result would be returned to the developer. This is already possible with the 1D-1V functional tests. However, in the case of the manufactured solutions tests on the 2D-3V model, significant computing resources are required across multiple cores to perform a 2D-3V model test. In the future, computing frameworks will be needed in order to facilitate rapid automated testing.

Appendix A. Supporting documentation for the manufactured solutions tests

The simulations used to create the data presented in this report were generated by the branch `https://github.com/mabarnes/moment_kinetics/tree/radial-vperp-standard-DKE-with-neutrals`, with the latest commit at the time of writing being `19ba012d11c5a0fb2c55d45547fa070ec27f30e7`.

In this appendix we give URL links to the input files used to generate the simulation data. To run a simulation use the following command

```
$ julia -O3 --project run_moment_kinetics.jl input.toml
```

To post process the simulation data and to generate the plots in this report run the following command:

```
$ julia -O3 --project run_MMS_test.jl
```

Appendix A.1. 2D-3V periodic boundary condition tests without collisions

`https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_nel_r_2_z_2_vpa_16_vperp_16.toml`
`https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_nel_r_2_z_2_vpa_8_vperp_8.toml`
`https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_nel_r_2_z_2_vpa_4_vperp_4.toml`
`https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_nel_r_2_z_2_vpa_2_vperp_2.toml`

Appendix A.2. 2D-3V periodic boundary condition tests with collisions

https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_cxiz_nel_r_2_z_2_vpa_16_vperp_16.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_cxiz_nel_r_2_z_2_vpa_8_vperp_8.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_cxiz_nel_r_2_z_2_vpa_4_vperp_4.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-sound-wave_cheb_cxiz_nel_r_2_z_2_vpa_2_vperp_2.toml

Appendix A.3. 1D-3V Wall boundary condition test

https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_1_z_2_vpa_16_vperp_16.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_1_z_2_vpa_8_vperp_8.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_1_z_2_vpa_4_vperp_4.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_1_z_2_vpa_2_vperp_2.toml

Appendix A.4. 1D-3V Wall boundary condition and electron sheath test

https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals-with-sheath_nel_r_1_z_2_vpa_16_vperp_16.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals-with-sheath_nel_r_1_z_2_vpa_8_vperp_8.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals-with-sheath_nel_r_1_z_2_vpa_4_vperp_4.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals-with-sheath_nel_r_1_z_2_vpa_2_vperp_2.toml

Appendix A.5. 2D-3V Wall boundary condition test

https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_2_z_2_vpa_16_vperp_16.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_2_z_2_vpa_12_vperp_12.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/runs/2D-wall_cheb-with-neutrals_nel_r_2_z_2_vpa_8_vperp_8.toml
https://github.com/mabarnes/moment_kinetics/blob/radial-vperp-standard-DKE-with-neutrals/

`runs/2D-wall_cheb-with-neutrals_nel_r_2_z_2_vpa_4_vperp_4.toml`

[1] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–03–01
[2] Barnes M, Parra F I and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–02–02
[3] Barnes M, Parra F I, Hardman M R and Omotani J 2021 *Excalibur/Neptune Report* 2047357–TN–04
[4] Barnes M, Parra F I, Hardman M R and Omotani J 2021 *Excalibur/Neptune Report* 2047357–TN–06
[5] Barnes M, Parra F I, Hardman M R and Omotani J 2021 *Excalibur/Neptune Report* 2047357–TN–08
[6] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–05–01
[7] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–07–01
[8] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–09–01
[9] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–11–01
[10] URL `https://github.com/JuliaMath/Interpolations.jl`
[11] Knudsen M 1916 *Annal. Phys.* **353** 1113
[12] Stangeby P C 2000 *The Plasma Boundary of Magnetic Fusion Devices* (Bristol and Philadelphia: Institute of Physics Publishing)
[13] Shu C W and Osher S 1988 *J. Comp. Phys.* 77:439–471
[14] Gottlieb S and Shu C W 1998 *Mathematics of Computation* 67:73–85
[15] Gottlieb S, Shu C W and Tadmor E 2001 *SIAM Rev.* 43:89
[16] Durran D R 1999 *Numerical methods for wave equations in geophysical fluid dynamics* (Springer)
[17] Abramowitz M and Stegun I 1972 *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables* (New York: Dover)
[18] Frigo M and Johnson S G 2005 Special issue on Program Generation, Optimization, and Platform Adaptation *Proceedings of the IEEE* 93(2):216–231
[19] Clenshaw C W and Curtis A R 1960 *Numerische Mathematik* 2:197
[20] Byrne S, Wilcox L C and Churavy V 2021 *JuliaCon Proceedings* 1(1), 68
[21] Parra F I, Barnes M and Hardman M R 2021 *Excalibur/Neptune Report* 2047357–TN–01–02
[22] Barnes M 2022 *Excalibur/Neptune Report* 2047357–TN–12
[23] Gowda S, Ma Y, Cheli A, Gwozdz M, Shah V B, Edelman A and Rackauckas C 2021 *arXiv preprint arXiv:2105.03949*
[24] URL `https://symbolics.juliasymbolics.org/dev/`
[25] Hakim A, Mandell N, Bernard T, Francisquez M, Hammett G and Shi E L 2020 *Phys. Plasmas* **27** 042304
[26] Dorf M, Dorr M, Hittinger J, Cohen R and Rognlien T 2016 *Phys. Plasmas* **23** 056102