UNIVERSITY *of* York

This is a repository copy of *Fluid simulations of plasma filaments in stellarator geometries with BSTING*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/136983/

Version: Published Version

Article:

White Rose
university consortium
Universities of Leeds, Sheffield & York

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

PAPER • OPEN ACCESS

Fluid simulations of plasma filaments in stellarator geometries with BSTING

To cite this article: B Shanahan *et al* 2019 *Plasma Phys. Control. Fusion* **61** 025007

View the article online for updates and enhancements.

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Fluid simulations of plasma filaments in stellarator geometries with BSTING

B Shanahan[1] , B Dudson[2] and P Hill[2]

[1] Max-Planck-Institut für Plasmaphysik, Wendelsteinstr. 1, 17491 Greifswald, Deutschland
[2] York Plasma Institute, Department of Physics, University of York, Heslington, York YO10 5DD, United Kingdom

E-mail: brendan.shanahan@ipp.mpg.de

Abstract

Here we present first results simulating plasma filaments in non-axisymmetric geometries, using a fluid turbulence extension of the BOUT++ framework. This is made possible by the implementation of the flux coordinate independent (FCI) scheme for parallel derivatives, an extension of the metric tensor components which allows them to vary in three dimensions, and development of grid generation. Tests have been performed to confirm that the extension to three dimensional metric tensors does not compromise the accuracy and stability of the associated numerical operators. Recent changes to the FCI grid generator in BOUT++, including a curvilinear grid system which allows for potentially more efficient computation, are also presented. Initial simulations of seeded plasma filaments in a non-axisymmetric geometry are reported. We characterize filaments propagating in the closed-field-line region of a low-field-period, rotating ellipse equilibrium as inertially-limited by examining the velocity scaling and currents associated with the filament propagation. Finally, it is shown that filaments in a non-axisymmetric rotating ellipse equilibrium propagate in a toroidally nonuniform fashion, and it is determined that the long connection lengths in the scrape-off-layer enable parallel gradients to establish, which has consequences for interpretation of experimental data.

Keywords: blob, filament, BOUT++, BSTING, stellarator, turbulence, fluid turbulence

(Some figures may appear in colour only in the online journal)

1. Introduction

Neoclassical transport is the dominant loss mechanism in sufficiently hot stellarator plasmas and can dominate in the plasma core [1]. In the outer, colder parts of the plasma, however, turbulence becomes more important and therefore dominates the plasma edge region [2]. Since the Wendelstein 7-X stellarator [3] has been optimized to have low neoclassical transport, turbulent transport could become comparable to neoclassical losses even in the center of the plasma. Wendelstein 7-X has already demonstrated novel edge physics; poloidally rotating filaments as measured by visible cameras [4], and a high-frequency variation of limiter heat fluxes [5] merit numerical investigation. Furthermore, the edge of Wendelstein 7-X in the island divertor configuration exhibits long connection lengths, such that cross field transport can become comparable to parallel transport. Predicting this cross-field transport in high density, collisional, detached plasmas without an ad hoc assumption for diffusion is a motivation of this work. It is becoming increasingly important to simulate turbulence in non-axisymmetric configurations.

In stellarator core plasmas, the most common method for simulating plasma turbulence is with gyrokinetic codes such as GENE [6], which is feasible due to the closed flux surfaces and the low collisionality. However, the simulations are computationally expensive for long (on the order of confinement time) temporal and global spatial scales. Additionally, GENE simulations are currently limited to flux-tube and flux-tube-ensemble geometries.

The high collisionality of tokamak and stellarator edge plasmas facilitates a fluid approach to turbulence simulations. While there are several fluid turbulence simulation codes for tokamak geometries [7–9], previous attempts to develop such a simulation framework for stellarators have been unsuccessful.

The recent implementation of the flux coordinate independent (FCI) [10] method for parallel derivatives in BOUT++ has allowed for simulations in non-axisymmetric geometries [11, 12]. Instead of aligning the computational grid to magnetic field lines, the FCI method uses interpolation of field line mapping on poloidal (or, in the case of linear geometries, azimuthal) planes to obtain values for finite-difference differentiation parallel to the magnetic field. In BOUT++, a cubic Hermite spline is utilized, although other methods have been implemented [12]. The FCI method removes the inherent singularities in flux- or field-aligned coordinates near magnetic null points. Additionally, since the computational grid is no longer aligned to the magnetic field, the simulation of complex geometries including X-points is possible. For a more complete discussion of the FCI method, see [10–12].

Here, we present the first results simulating plasma fluid turbulence in non-axisymmetric geometries, made possible by extensive modifications to the BOUT++ framework [13, 14]. Section 1.1 describes the recent modifications to the BOUT++ framework which are relevant for this work. Initial testing of the modified framework is described in section 2, where sections 2.1 and 2.2 test the accuracy FCI parallel gradient operators and their associated boundary conditions, and section 2.3 reports the modifications to the Laplacian inversion algorithms. Section 3 introduces a new curvilinear coordinate system for FCI simulations in BOUT++ which is used in section 4 to simulate plasma filaments in non-axisymmetric geometries; filaments in the closed-field-line region of a rotating ellipse geometry are determined to be inertially-limited and exhibit a toroidally non-uniform propagation, a result which has implications for interpretation of experimental data. Finally, section 5 describes how the curvilinear FCI grids can be used for simulation of realistic geometries, namely Wendelstein 7-X.

1.1. Modifications to the BOUT++ framework

The BOUT++ framework is a modular, object oriented and open source framework for fluid simulations with an international team of developers [13]. This paper presents recent progress in modifying BOUT++ to Simulate Turbulence In Non-axisymmetric Geometries under the 'BSTING' project.

Previous work in simulating non-axisymmetric geometries has focused on the conventional BOUT++ framework, which is a 3D code but was written with metric tensor components which vary in two dimensions due to an assumption of toroidal symmetry. For an accurate simulation of plasma dynamics in stellarators, BSTING must include metric components which are fully three dimensional. This extension to three dimensions is simple in principle (and was in fact mentioned in the introduction of the original BOUT++ paper [13]), but unfortunately the geometrical components

are integral to many different parts of the code, and the work presented here has required extensive modifications to the framework.

The majority of modifications are primarily focused on the numerical methods of spatial operators and do not affect file handling, parallelization, post processing, and many other functions in BOUT++. Development has focused on implementing operators relevant to edge transport and turbulence simulations: spatial derivatives of scalar fields which vary in three dimensions, and Laplacian inversion. Here we address the most relevant issues: the accuracy of spatial gradient operators, boundary condition implementation, and Laplacian inversion which allows plasma potential to be calculated from vorticity. The following section provides initial tests for the implementation of these methods.

2. Testing

The development of BSTING is an extensive modification to the BOUT++ framework, and therefore careful testing of numerical accuracy is required. In this section, we concentrate on ensuring the accuracy of spatial derivatives, boundary conditions, and Laplacian inversion. All tests in this section use a geometry where the poloidal planes are described by the radial x-coordinate and vertical z-coordinate while the y-coordinate describes the toroidal (or longitudinal in linear geometries) direction. The FCI operators therefore interpolate the relevant values based on field line mapping in the x–z planes. The FCI method relies only on the local magnetic field, and can handle both axisymmetric and non-axisymmetric geometries. In section 3 we will discuss an alternative poloidal coordinate system for complex geometries.

2.1. Flux surface mapping using heat diffusion

Preconditioning is often used in implicit time integration schemes, to step over fast dynamics [15]. In BOUT++ with structured grids this has been used effectively to precondition shear Alfvén waves [14], though not the more dispersive kinetic Alfvén waves which are present in most models of interest. When using the FCI technique the domain can not in general be divided into magnetic flux surfaces, making preconditioning of parallel dynamics in some cases more difficult. In many cases the preconditioner would involve solving a 3D problem, but preconditioning of kinetic Alfvén waves in any case couples perpendicular and parallel directions. Design and implementation of an effective preconditioner with the FCI technique is the subject of ongoing work, so here we use explicit or implicit timestepping with no preconditioner.

A potential issue with the implementation of the FCI scheme as discussed in section 1.1 is that since the poloidal planes are not orthogonal to the magnetic field lines, there could be a considerable pollution of perpendicular dynamics due to the projection of parallel effects [16]. A simple and common test to ensure the proper calculation of parallel dynamics using the FCI method in complex geometries is to implement a parallel diffusion model such as that shown in
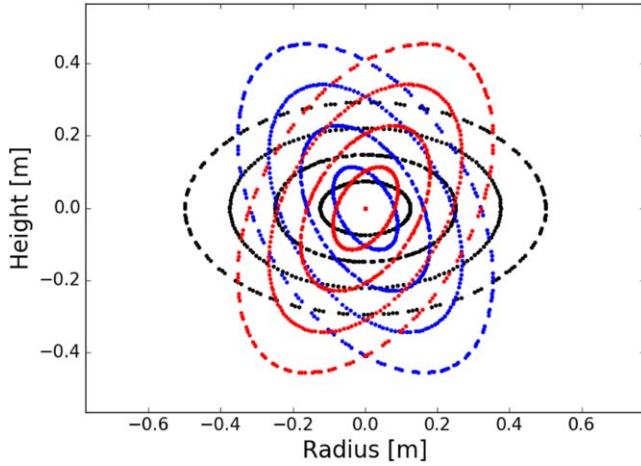
Figure 1. Poincare plot indicating the flux surfaces in the analytic straight rotating ellipse equilibrium as calculated by the Zoidberg grid generator.
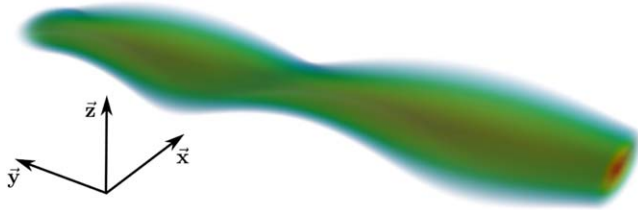

Figure 2. Flux surfaces for a straight rotating ellipse equilibrium as calculated using the flux coordinate independent operators in BSTING, reproducing to the test shown in figure 4 from [17].

equation (1).

$$\frac{\partial f}{\partial t} = \nabla \cdot (\mathbf{b}\mathbf{b} \cdot \nabla f) \equiv \nabla_\parallel^2 f, \tag{1}$$

where **b** is the magnetic field vector. Here the diffusion model in equation (1) is used to test the numerical diffusion in a rotating ellipse equilibrium as done in [11, 12]. Specifically, we will simulate this model on a rotating ellipse geometry. The input grid for BSTING is created by the Zoidberg grid generator, using an analytic description of the magnetic field based on coil currents and position. A Poincaré plot showing the flux surfaces calculated in Zoidberg is shown in figure 1.

Figure 2 illustrates that simulating a parallel diffusion model qualitatively reveals the flux surfaces for a rotating ellipse equilibrium, recovering the results from [11, 12]—however this result differs in that it uses fully three dimensional metric tensor components, whereas the previous results utilized a metric tensor that varied in only two dimensions. It is perhaps worth mentioning that in initial testing the addition of variation in the third dimension has increased calculation time by about 5%–15%, so calculations do not require prohibitively more resources relative to conventional BOUT+ + simulations. This added flexibility allows for non-axisymmetric toroidal geometries. Figure 3 indicates the flux surfaces as calculated by BSTING in a toroidal rotating ellipse geometry. The red surfaces indicate the 2D projection for the core region on each poloidal plane, and the blue/green cloud is the interpolated function between the poloidal planes.
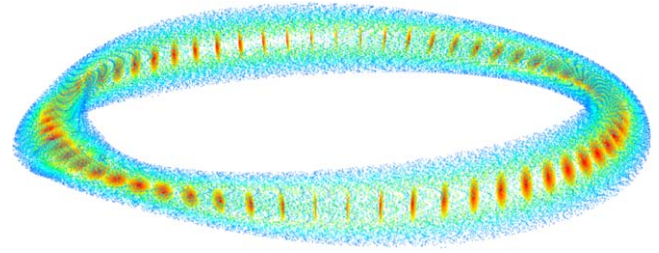

Figure 3. Non-axisymmetric flux surfaces for a toroidal rotating ellipse equilibrium as calculated in BSTING.

This heat flux mapping indicates that the FCI operators are capable of simulating non-axisymmetric geometries after the transition to three dimensional metric tensors in BSTING. The following section will use a more quantitative method to ensure the numerical operators and implementation of boundary conditions with three dimensional metric tensors have sufficiently small numerical error.

2.2. Method of manufacturing solutions for parallel derivatives

Imposing correct boundary conditions on plasma fluid turbulence simulations is complicated [18]—but the FCI method has particular issues at the boundaries, since the field lines can leave the domain before reaching the next toroidal plane, therefore leading to non-uniform grid point spacing for interpolation and complicating the correct calculation of derivatives. There have been a few recent advances in boundary condition calculation for FCI operators; BOUT+ + utilizes the Leg-Value-Fill (LVF) method detailed in [12], which employs a Taylor expansion about the boundary to extrapolate the fields onto the 'leg' of the field line which lies outside the boundary. In this section we extend previous testing [12] using the method of manufactured solutions [19, 20] to ensure that the extension to three dimensional metric tensors has not diminished the accuracy and stability of the framework. Two coupled differential equations were therefore simulated for a single time step:

$$\frac{\partial f}{\partial t} = \nabla_\parallel g + D\nabla_\parallel^2 f \tag{2}$$

$$\frac{\partial g}{\partial t} = \nabla_\parallel f + D\nabla_\parallel^2 g, \tag{3}$$

where parameters are identical to those in [12]; namely, $D = 10$, and the domain measures $0.1 \times 10 \times 1$ (x, y, z) meters. The magnetic geometry is a sheared slab, such that $(B_x, B_y, B_z) = (0, 1, 0.05 + (x\text{-}0.05)/10)$. The manufactured solutions are also those from [12]:

$$f = \sin(\bar{y} - \bar{z}) + \cos(t)\sin(\bar{y} - 2\bar{z}) \tag{4}$$

$$g = \cos(\bar{y} - \bar{z}) - \cos(t)\sin(\bar{y} - 2\bar{z}), \tag{5}$$

where $\bar{y}$ and $\bar{z}$ are normalized between 0 and 2π. The diffusion terms in equations (2) and (3) scale with y-spacing, and do not affect the convergence of $\nabla_\parallel$. Therefore the grid is scaled in y and z simultaneously. Figure 4 indicates the convergence of FCI operators in BSTING, including LVF boundary conditions.
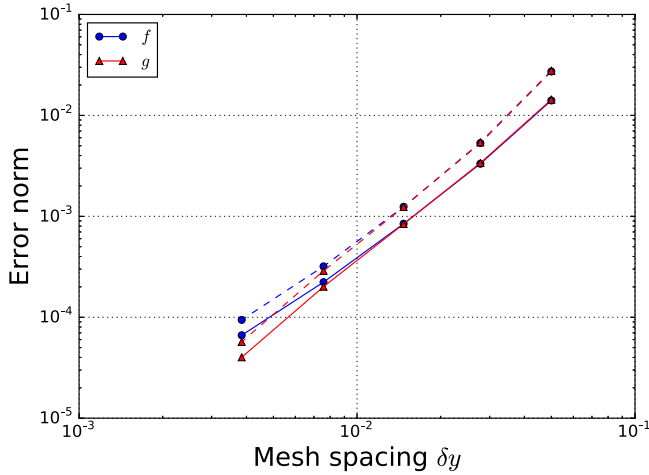
Figure 4. Second order convergence for FCI operators in BSTING: the slope of the fits are 2.06 and 2.26 for *f* and *g*, respectively. The dashed lines indicate the maximum error (described as l_∞ in [12]).

Figure 4 indicates a second order convergence of our operators. Explicitly, the convergence order is 2.08 for *f*, and 2.26 for *g*. A second order convergence is expected, as the FCI operators are second-order-central-differencing operators.

Having established the accuracy and stability of the FCI operators and the associated LVF boundary conditions in BSTING, the following section describes the implementation of Laplacian inversion routines which allow for the calculation of plasma potential from vorticity.

2.3. Laplacian inversion with complete poloidal metrics

One of the advantages of BOUT++ is its modular nature; numerical methods can be modified without compromising the stability or accuracy of the rest of the framework. For this reason, several different methods for Laplacian inversion have been implemented in BOUT++. Unfortunately for BSTING, many of these routines assume a periodicity in one direction (the *z* coordinate, usually the toroidal angle in tokamak simulations), since BOUT++ was originally designed to simulate turbulence in tokamak scrape-off-layers. Recent work on implementing the Hermes model [21] in BOUT++ has included several new numerical methods. One of these is the implementation of a Laplacian inversion routine in three dimensions, which inverts an inhomogeneous Helmholtz equation in the conservative form:

$$\nabla \cdot (A\nabla_\perp f) + Bf = b, \quad (6)$$

where *A* and *B* are coefficients set based on the equation to be solved, *b* is most often vorticity and *f* is the unknown quantity for which one solves (usually plasma potential). In most cases for fluid turbulence simulations, $B = 0$ so that this equation becomes a Laplacian equation. Here, the Laplacian is solved at each poloidal or azimuthal slice separately. The discretization of equation (6) is then described in terms of fluxes through cell faces in the poloidal plane:

$$\frac{1}{J}\frac{\partial}{\partial x}\left(JAg^{xx}\frac{\partial f}{\partial x}\right) + \frac{1}{J}\frac{\partial}{\partial z}\left(JAg^{zz}\frac{\partial f}{\partial z}\right)$$
$$+ \frac{1}{J}\frac{\partial}{\partial x}\left(JAg^{xz}\frac{\partial f}{\partial z}\right) + \frac{1}{J}\frac{\partial}{\partial z}\left(JAg^{xz}\frac{\partial f}{\partial x}\right) \quad (7)$$
$$+ Bf = b,$$

where *J* is the Jacobian, g^{ij} are the metric tensor components, and *A*, *B* and *b* are variables which are specific to each situation. The current implementation of this solver utilizes the PETSc suite of data routines [22], which is available with several features including preconditioners for efficient computation. This implementation differs from conventional BOUT++ since it includes the off-diagonal metric terms (g^{xz}). By setting the metric tensor components, g^{ij}, to non-zero values and comparing the implemented inversion routine using PETSc to explicit calculation of equation (7) indicated a difference of less than 10^{-15}. Testing with zero-value diagonal metric tensor components indicated similar errors relative to the implementation without off-diagonal metrics in BOUT++, suggesting proper convergence of the inversion routines.

Having implemented the FCI operators and Laplacian inversion with Cartesian poloidal grids, the BSTING project is now capable of simulating turbulence in non-axisymmetric geometries. A significant challenge for this method, however, is to handle the entire plasma cross section in a Cartesian poloidal grid while neglecting the plasma core and far edge. One solution to this issue is to use a penalization function to mask the areas where the variables should not be evolved. This method has been used previously in BOUT++ [17] to remove solid-density magnetic coils in the simulation domain and is currently used in with FCI operators in GRILLIX [9] to mask the plasma core and far scrape-off-layer. The disadvantage of this method is that it requires a large poloidal grid for a relatively small computational area. In the following section we present a new method for generating FCI grids in BOUT++ and BSTING which does not use a grid over the entire plasma cross section, potentially providing faster computation.

3. Elliptic FCI grid generation

3.1. Implementation of elliptic grids

While all previous simulations using the FCI method have used poloidal planes with Cartesian coordinates [9–12, 23], this is not required. The method is independent of the poloidal grid system as long as interpolation in these planes is correctly calculated and communicated. Here we present recent results using structured, non-Cartesian poloidal grids which are still logically rectangular [24, 25]. As an illustration of this method, figure 5 illustrates a sample grid with independent inner and outer surfaces.

These new grids have been added to the BOUT++ FCI grid generator, Zoidberg, and are included in a recent release
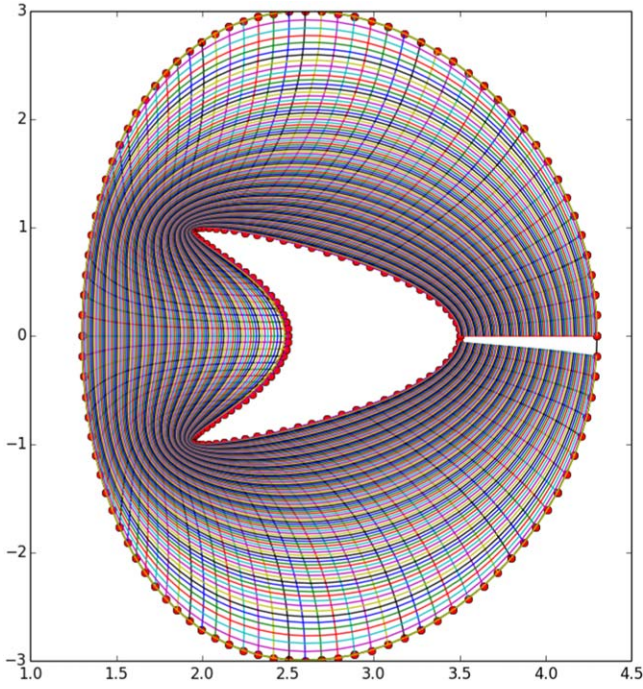
Figure 5. An example of a curvilinear grid generated by the Zoidberg grid generator, which can be found in the BOUT++ manual [26].

of BOUT++ (version 4.1). These grids are particularly advantageous as they include a periodic direction which could potentially increase computational efficiency. A grid is generated by prescribing an inner and outer surface, and then inverting an elliptic equation to connect the inner and outer points. This process is described in detail in [24, 25], but is repeated here.

If the inner and outer points are defined by functions of the major radius R and vertical location Z, the points between the surfaces defined by the coordinates $x(R, Z)$ and $z(R, Z)$ obey the elliptic relation:

$$\frac{d^2x}{dR^2} + \frac{d^2x}{dZ^2} = 0 \tag{8}$$

$$\frac{d^2z}{dR^2} + \frac{d^2z}{dZ^2} = 0 \tag{9}$$

which can be inverted, giving:

$$aR_xx - 2bR_xz + cR_zz = 0 \tag{10}$$
$$aZ_xx - 2bZ_xz + cZ_zz = 0, \tag{11}$$

where:

$$a = R_z^2 + Z_z^2 \tag{12}$$
$$b = R_xR_z + Z_xZ_z \tag{13}$$
$$c = R_x^2 + Z_x^2 \tag{14}$$

which gives a nonlinear set of equations which can be solved iteratively to determine the locations of the grid points [24, 25].

In Zoidberg, both the inner and outer surface shapes are independently prescribed, and can be described using various methods: Zoidberg includes an flux surface shape generator, which will describe a shape based on elongation, triangularity

and indentation. Alternatively, one can use the Zoidberg field line tracer to construct flux surfaces from a given magnetic field (i.e. from VMEC, a vacuum field solver, or an analytic magnetic field description), and generate a shape based on this flux surface mapping.

These grids provide an additional degree of flexibility and avoid some potential problems—primarily how to mask the core/outer edges: perpendicular (poloidal) boundaries are logically perpendicular to the grid cells, simplifying the imposition of boundary conditions—although parallel boundaries must still utilize a method such as the LVF method [12] discussed earlier. Some minor modifications to numerical operators are required for this poloidally-curvi-linear coordinate system, which are discussed in the appendix of this work.

The right side of figure 6 illustrates the curvilinear grid used in the following section for simulations of plasma filaments in a rotating ellipse geometry, and the left side is a lower-resolution example of the same geometry which indicates the field line map locations.

This two-field period, analytically-prescribed rotating ellipse geometry has a major radius of 2.5 m, The inner surface is described by a flux surface calculated by Zoidberg, but the rest of the grid is not aligned to flux surfaces; the outer surface is a circle centered around the magnetic axis with a radius of 50 cm. Therefore, this geometry incorporates both open and closed field lines. The left side of figure 6 indicates grid points as blue crosses. The intersection of field lines from the previous plane are indicated by circles: red circles indicate field lines which land within the computational domain, and the remaining circles indicate where the field lines intersect the boundary—either through the outer surface (blue) or inner (black). The grid has a resolution of $68 \times 128 \times 16$ (radial, poloidal, toroidal), which gives an average poloidal resolution of 0.5 cm (radial) by 1.5 cm (poloidal).

4. Nonlinear filament simulations

4.1. Isothermal reduced MHD model

The following section utilizes a finite-β electromagnetic iso-thermal reduced magnetohydrodynamic model similar to that used in the isothermal version of TOKAM3X [27] which evolves vorticity ω, electromagnetic potential $A_\parallel$, electron density n, and parallel momentum $\Gamma = m_i n v_\parallel$. Electron and ion temperatures T_e and T_i are assumed constant, though independently specified. The magnetic field is described by a constant equilibrium field B_0 and a time-evolving poloidal field such that:

$$B = B_0 + \nabla \times (A_\parallel e_\phi) \tag{15}$$

$$= B_0 + \nabla\psi \times \nabla\phi, \tag{16}$$

where $A_\parallel$ is the parallel component of the vector potential and a large-aspect ratio approximation has been utilized such that $\psi = RA_\parallel$.
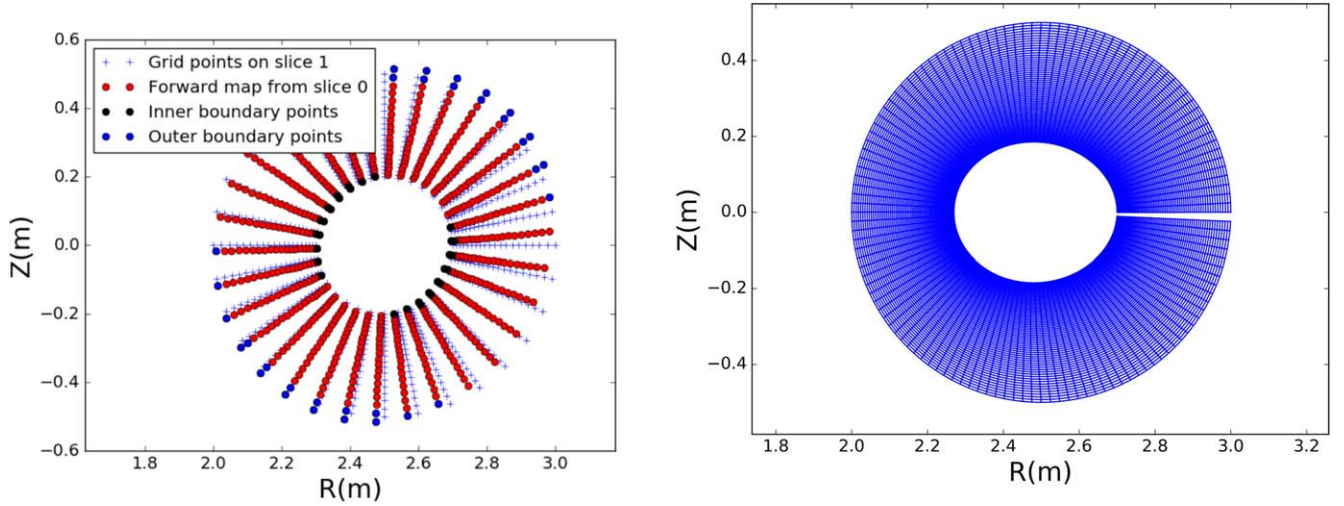
Figure 6. (Left): An example of a curvilinear grid for a rotating ellipse geometry, with an inner surface described by a flux surface, and a circular outer surface providing both open and closed field lines. Blue crosses indicate grid points, whereas circles indicate the locations of field line mapping from the previous plane for the FCI scheme—red circles indicate field lines which remain in the computational domain, black circles are field lines leaving the inner boundary, and blue circles leave the outer boundary. (Right): a curvilinear rotating ellipse grid used for simulations in section 4.

The equations are described as follows in SI units:

$$\frac{\partial \omega}{\partial t} + (\mathbf{v_E} + \mathbf{v_{\|i}}) \cdot \nabla \omega = \nabla_\| J_\| + \nabla \cdot \left(p \nabla \times \frac{\mathbf{b}}{B} \right) + \nu \nabla_\perp^2 \omega \tag{17}$$

$$\frac{\partial}{\partial t} \left[A_\| - \frac{m_e}{e} v_{\|e} \right] = -\partial_\| \phi + \frac{1}{n} \partial_\| p_e - \frac{1}{en} \eta J_\| \tag{18}$$

$$\frac{\partial n}{\partial t} + \mathbf{v_E} \cdot \nabla n = -\nabla_\|(v_{\|e} n) + \nabla \cdot \left(p_e \nabla \times \frac{\mathbf{b}}{B} \right) \tag{19}$$

$$\frac{\partial \Gamma}{\partial t} + \mathbf{v_E} \cdot \nabla \Gamma = -\nabla_\|(v_{\|e} \Gamma) + \nabla \cdot \left(\Gamma e T_i \nabla \times \frac{\mathbf{b}}{B} \right) - \partial_\| p \tag{20}$$

$$\omega = \nabla \cdot \left[\frac{m_i n}{B_0^2} \left(\nabla_\perp \phi + \frac{\nabla_\perp p_i}{en} \right) \right] \tag{21}$$

$$J_\| = -\frac{1}{\mu_0} \nabla_\perp^2 A_\|. \tag{22}$$

Here $\partial_\| \equiv \mathbf{b} \cdot \nabla$ and $\nabla_\| f \equiv \nabla \cdot (\mathbf{b} f) = B \partial_\| \left(\frac{f}{B} \right)$. The pressure is $p = p_e + p_i = n(T_e + T_i)$. The vector $\mathbf{b_0} = \mathbf{e}_\phi$ is the 'toroidal' magnetic field unit vector, and $\mathbf{b} = \mathbf{B}/B_0$ is the unit vector along the total magnetic field, assuming that the poloidal magnetic field is small relative to the toroidal field. Gradients in the poloidal plane, which is not necessarily perpendicular to the magnetic field (in the case using FCI derivatives, as is used here), are defined by $\nabla_\perp = \nabla - \mathbf{b_0}\mathbf{b_0} \cdot \nabla$. Dissipation terms are determined by the kinematic viscosity ν and the resistivity η, in units of $m^2\,s^{-1}$ and Ωm, respectively.

In this model, the magnetic drift term is treated generally (in comparison to, for instance, equation (A13)) and is written

as:

$$\nabla \cdot \left[p \nabla \times \frac{\mathbf{b}}{B} \right] = \nabla \times \frac{\mathbf{b}}{B} \cdot \nabla p \tag{23}$$

$$= \left(\nabla \frac{1}{B^2} \times \mathbf{B} + \frac{1}{B^2} \nabla \times \mathbf{B} \right) \cdot \nabla p \tag{24}$$

$$= -\frac{2}{B^3} \nabla B \times \mathbf{B} \cdot \nabla p \tag{25}$$

$$= \frac{2}{B} \mathbf{b} \times \nabla \log B \cdot \nabla p \tag{26}$$

which uses $\nabla \times \mathbf{B} \cdot \nabla p = 0$ which is valid in equilibrium since $\mathbf{J} \cdot \nabla p = 0$. The curvature operator is then defined as:

$$C(f) = \frac{2}{B} \mathbf{b} \times \nabla \log B \cdot \nabla f \tag{27}$$

which has a similar form as that derived in the appendix (equation (A8)), meaning that we can use the bracket coefficient to calculate the curvature effects in curvilinear grids. This is especially convenient as the magnetic field does not, in general, vary solely with the major radius in stellarators—an approximation which is often used in fluid turbulence simulations [17, 28, 29]. In the simulations presented here, all cross-field drifts are implemented with the 2nd order Arakawa brackets [30].

4.2. A weakly-non-axisymmetric, rotating ellipse geometry

As an initial investigation of turbulence in non-axisymmetric geometries, a seeded plasma filament in a rotating ellipse geometry was considered. While there have been experimental investigations of turbulent filaments in stellarators [31], this study will serve as the first example of fluid
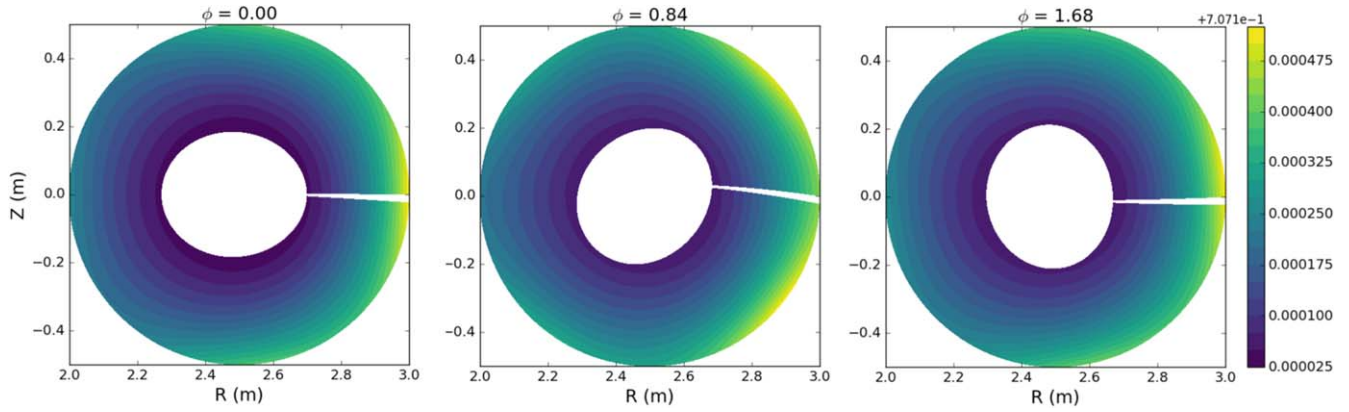
Figure 7. Variation of the non-toroidal magnetic field at three different toroidal locations—obtained by multiplying the total field by the major radius *R*, and calculating the difference with respect to the mean value.

turbulence simulations in non-axisymmetric geometries. A seeded filament test offers a somewhat straightforward approach to studying important phenomena in plasma transport. Previous studies in BOUT++ have investigated filaments in slab [32], toroidal pinch [28], and X-point geometries [17, 33].

For the studies presented here, an analytically calculated, low-field-period rotating ellipse geometry was chosen due to the relatively straightforward implementation and analysis. These analytic equilibria are a necessary step before geometries like W7-X. Wendelstein 7-X grids for use in BSTING are described in section 5, but turbulence studies in these more complex geometries will be a subject of further study. Furthermore, low-field-period rotating ellipse geometries exhibit a magnetic field which generally varies as $1/R$ (see figure 1 from [34]), allowing for a more straightforward analysis since this configuration is most similar to axisymmetric configurations. Figure 7 illustrates the degree of non-axisymmetry by plotting the variation of the magnetic field multiplied by the major radius, since a plot of the magnetic field strength would be dominated by the predominantly $1/R$ variation.

From figure 7 it can be deduced that the magnitude magnetic field which does not vary like $1/R$ only changes toroidally by less than a percent, indicating a small degree of non-axisymmetry in the magnetic field strength, which can affect the drive term for filament propagation (equation (26)).

4.3. Filament characterization

To characterize filament propagation in this non-axisymmetric geometry, a field-aligned plasma filament is first initialized; an approximately circular density perturbation at $(R, Z, \phi) = (2.5, -0.3\,\text{m}, 0.0)$ is prescribed and a simple parallel diffusion model as in equation (1) is first simulated to achieve an initial condition of a field-aligned filament. As this is a low-shear geometry, the filament approximately becomes field aligned once the initial distribution diffuses once toroidally. The initial field-alignment is determined when the maximum value of the density on a plane varies by less than 5% in a timestep ($100/\omega_{Ci}$, where ω_{Ci} is the ion cyclotron frequency). This condition is satisfied after 100 timesteps, or

ten thousand ion cyclotron times. This field-aligned density distribution, where the peak density perturbation is $n = 1.05 \times 10^{19}\,\text{m}^{-3}$, is then used as an initial condition for the seeded filament simulation using the model described in section 4.1. All other plasma fields are not initialized and, once the field-aligned filament is achieved, are allowed to develop independently. The ion and electron temperature is set to 100 eV and the background density is $n_0 = 1 \times 10^{19}\,\text{m}^{-3}$.

Plasma filaments (or blobs) are often characterized by the method by which the charge separation is resolved; if charge is carried via parallel currents through the sheath, filaments are considered 'sheath limited'. If the connection length to the sheath is large, however, this charge separation can be short-circuited via perpendicular currents and the filaments propagate in a so-called 'inertially-limited' regime [35]. Filament propagation can also characterized by the scaling of the propagation speed as a function of its poloidal cross section, $\delta_\perp$; inertially limited filaments scale proportional to $\delta_\perp^{1/2}$, whereas sheath-limited filaments scale as $\delta_\perp^{-2}$. A derivation of the inertial scaling is given in the appendix B, but for a more complete discussion of filaments, see [35, 36].

One can therefore determine the filament propagation regime by plotting the scaling of the maximum speed as a function of filament diameter $\delta_\perp$. The edge and scrape-off-layer of stellarators such as Wendelstein 7-X can exhibit large connection lengths [5]. As an initial insight into filament behavior in a non-axisymmetric field with long connection lengths, filaments were seeded in the closed-field-line region in the weakly non-axisymmetric geometry discussed in the previous sections. The scaling of these filaments is shown in figure 8, where $\delta_\perp = 1$ is normalized to 7 cm, the initial filament diameter for the filaments in the following section (4.4).

Similar to the tokamak (axisymmetric) case, the scaling of filaments initialized in the closed-field-line region propagate in an inertially-limited regime, as indicated by the $\delta_\perp^{1/2}$ scaling in figure 8. As a confirmation of the inertially-limited propagation, figure 9 illustrates the currents which dictate the propagation of the filament at $t \approx 4\ \mu\text{s}$.
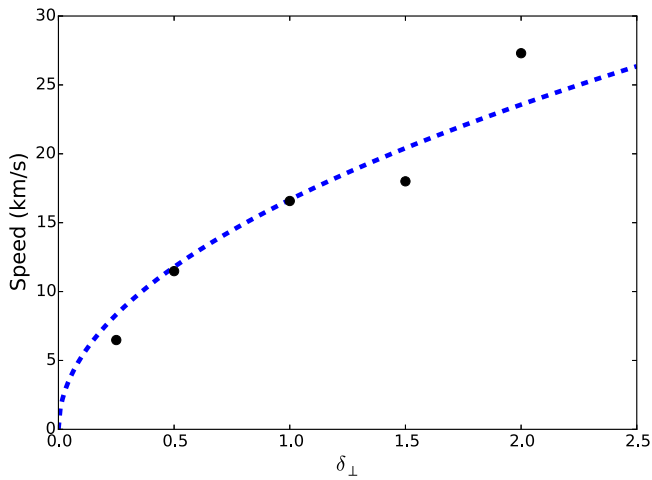
Figure 8. Inertial filament scaling; filament velocity (circles) and tend to follow a $\delta_\perp^{1/2}$ scaling, indicating propagation in the inertial regime.

Since the divergence of the parallel current is much smaller than the perpendicular currents, the potential difference is resolved via short-circuiting perpendicular currents, instead of traveling along field lines to the sheath. This again supports the characterization an inertially-limited regime. As this is only a weakly-non-axisymmetric field, it is reasonable to find similarities to filaments in an axisymmetric field, for instance in [33], where inertially-limited filaments were characterized in a MAST (tokamak) geometry. For a more strongly-non-axisymmetric geometry such as Wendelstein 7-X, the filament propagation may exhibit different behavior, since the filament drive changes directions relative to the major radius within a field period. While filament simulations in Wendelstein 7-X await a future publication, the following section discusses how even a weakly-non-axisymmetric field can alter the toroidal uniformity of the filament propagation.

4.4. The effects of nonaxisymmetry

If the magnetic geometry is not axisymmetric, the filament drive due to the magnetic field curvature can vary along the length of a filament. If the drive is toroidally non-uniform, one would expect the propagation to also vary toroidally. It is often assumed, however, that filaments propagate uniformly along field lines, for instance in [31]. To test the effects of a non-axisymmetric magnetic field, we can investigate the propagation of a filament at different toroidal locations. Figure 10 illustrates the radial filament velocity (solid) and displacement (dotted) of a 100 eV plasma filament at various toroidal angles.

Figure 10 exhibits typical behavior of plasma filaments seen in other magnetic configurations [17, 28]; an initial radial acceleration and expansion before being decelerated as a result of viscosity and dissipation of the charge separation. Figure 10 also indicates that even a modestly non-axisymmetric field, as simulated here, can visibly affect the propagation of filaments. This effect is a direct consequence of the non-axisymmetric filament drive, as shown in figure 11 which

indicates how the magnetic drive term (black, also fitted), and the resulting maximum filament velocity vary as a function of toroidal angle. Here, the filament velocity is normalized to the toroidally-averaged radial velocity at 100 timesteps. The blue squares in figure 11 indicate the normalized velocity at each toroidal position, averaged over the 100 timesteps. The fill cloud indicates the standard deviation of the toroidally-normalized velocity for these sample timesteps.

The non-axisymmetric propagation of filaments can be clarified by considering the timescales associated with filament propagation. First, we approximate the timescale for parallel propagation along a filament to follow the relation $t_\parallel \sim \frac{l}{c_s}$ where l is the length along the filament and c_s is the ion sound speed. In the simulations presented here, $c_s \approx 6.9 \times 10^4 \text{ m s}^{-1}$, which indicates that information takes about 14 μs to propagate one meter. Therefore, if the filament is driven non-uniformly, the time which the filament needs to restore the symmetry is longer than the propagation timescale $t_\perp$, which can be approximated by assuming $L_\perp \approx \delta_\perp \approx 7$ cm and $v_\perp \approx 13 \text{ km s}^{-1}$—indicating therefore that $t_\perp \approx 5$ μs.

This assertion can be tested by increasing the speed at which this restoration is performed, for instance by increasing the sound speed. When simulations were performed with hotter (1 keV), smaller filaments—thus keeping the pressure constant—the standard deviation of the position of the filaments averaged 79% of that for the colder simulation, indicating that a hotter filament propagates more uniformly. This can also be seen in the resulting speed of the hotter filament, shown as red triangles in figure 11, which does not vary as strongly with toroidal location.

It is also possible, however, that the filament is restored to uniform propagation toroidally at the Alfvén velocity. This would also explain the more uniform propagation for a hotter filament, since the density perturbation was reduced to provide an equal drive (from pressure), and the Alfvén velocity is a function of the plasma β. To determine the extent to which this non-axisymmetric nature is affected by the Alfvénic effects, one can simulate a filament in an electrostatic case. In an electrostatic case, all terms in the model described in section 4.1 which are dependent on the plasma β are neglected, which in essence provides an infinite Alfvén speed. Figure 12 illustrates how the propagation of a filament in an electrostatic and electromagnetic filament compare as a function of toroidal angle.

Figure 12 indicates that the non-uniform propagation is not an electromagnetic effect and thus cannot be adequately mitigated by parallel transport at infinite Alfvénic speeds, since the electrostatic and electromagnetic case exhibit very similar characteristics.

5. Wendelstein 7-X curvilinear grids

BSTING is designed to provide numerical support for experimental measurements. The curvilinear grid system presented in section 3 has therefore been applied to Wendelstein 7-X geometries using various descriptions of the
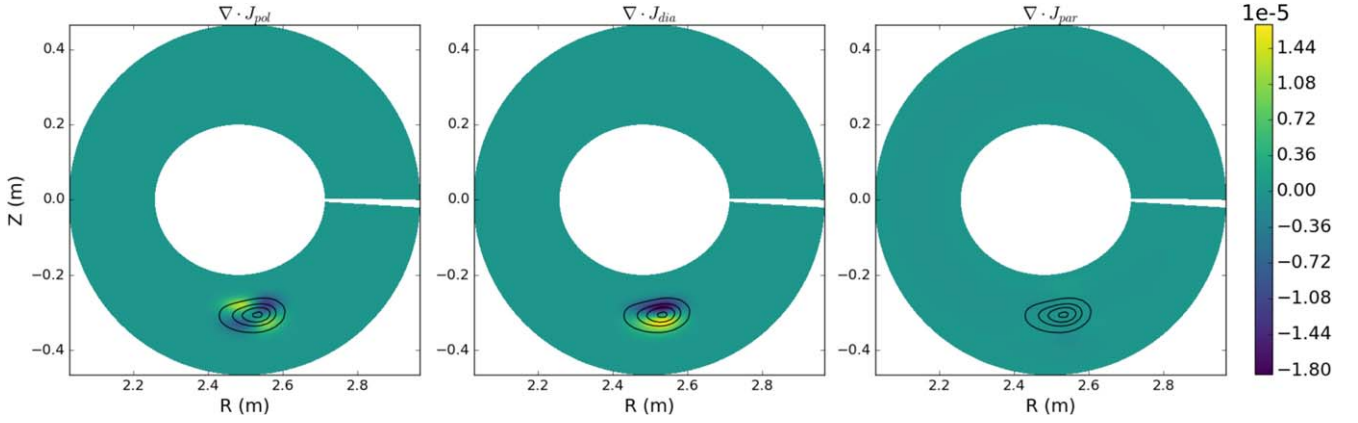
Figure 9. An illustration of the divergences for parallel and perpendicular currents (color contours) which dictate the propagation of a filament (black contours, overlaid); parallel currents are negligible, indicating inertially-limited propagation.
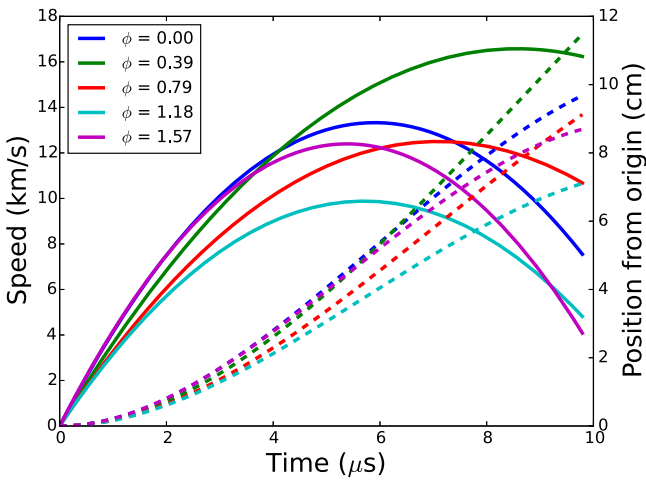

Figure 10. The effects of nonaxisymmetry; filament velocity (solid) and position (dotted) at various toroidal angles (color) in a rotating ellipse equilibrium.
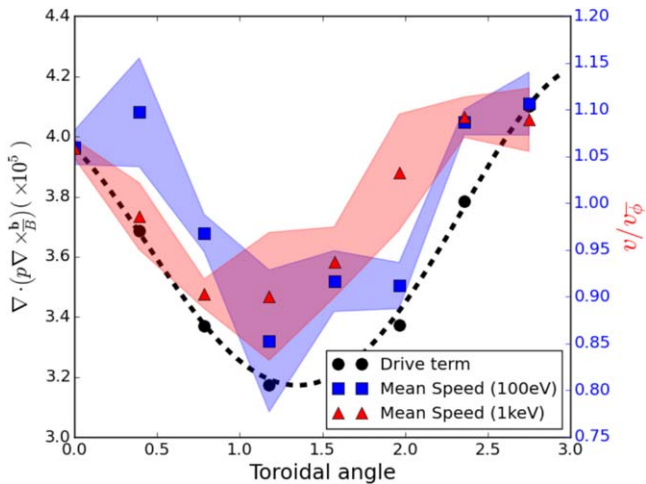

Figure 11. Filament speed and standard deviation normalized to the average toroidal speed, averaged over 100 timesteps, at each toroidal position for a 100 eV filament (blue squares) and a 1 keV filament (red triangles). The more uniform propagation of a hot filament indicates that the sound speed determines the timescale at which non-uniform propagation is mitigated.
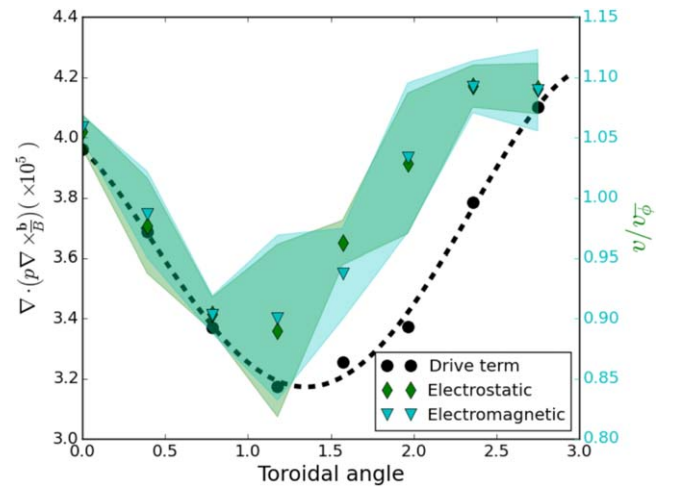

Figure 12. The non-uniform propagation of an electrostatic (green diamonds) and electromagnetic (cyan triangles) filament as a function of toroidal angle. Similar propagation indicates that filaments are not restored to uniformity at the Alfvén timescale.

magnetic field. As this geometry is considerably more complicated than the analytically-prescribed rotating ellipse equilibria presented earlier, the following sections extend the flux surface mapping tests to the W7-X grids.

5.1. Inherent perpendicular diffusion in W7-X curvilinear grids

Here we present the development of curvilinear poloidal grids for Wendelstein 7-X geometries using outputs from the VMEC code [37]. To test the implementation and limitations of grids in this complicated geometry, the parallel diffusion model in section 2.1, equation (1) was modified to include a perpendicular diffusion, as shown in equation (28).

$$\frac{\partial f}{\partial t} = \nabla \cdot (\mathbf{bb} \cdot \nabla f) + D \nabla \cdot (\nabla f - \mathbf{b_0}\mathbf{b_0} \cdot \nabla f) \qquad (28)$$

$$\equiv \nabla_\parallel^2 f + D \nabla_\perp^2 f. \qquad (29)$$
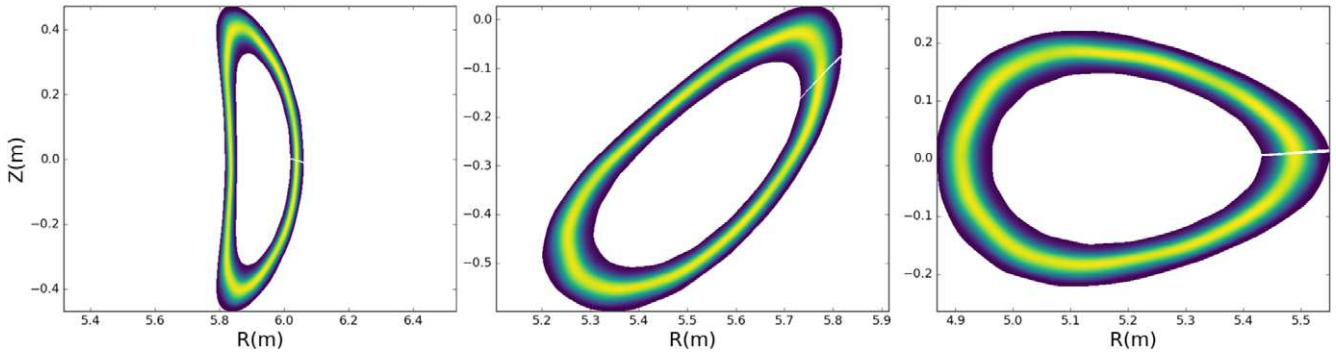
Figure 13. Three cross sections of the Wendelstein 7-X stellarator indicating flux surfaces as traced by a parallel heat diffusion equation in BSTING.
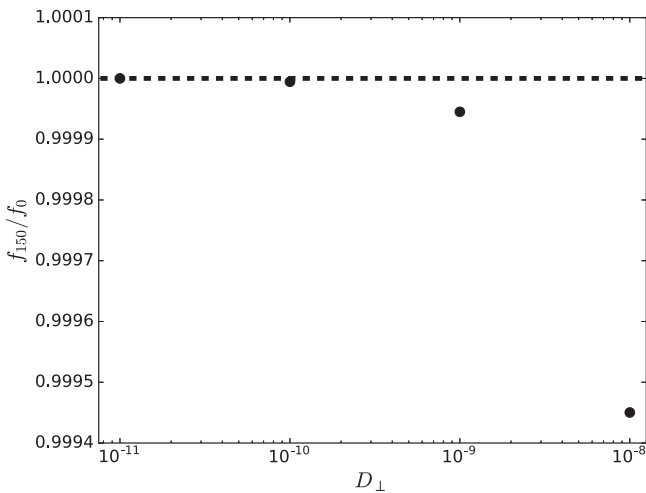

Figure 14. Proportion of the total test function f at the 150th timestep normalized to the zero-perpendicular-diffusion case, f_0, for several perpendicular diffusion coefficients in a Wendelstein 7-X grid.

By setting the diffusion coefficient D to zero and simulating equation (28), we can again recover flux surfaces, similar to the results described in section 2.1. The results of this simulation are shown in figure 13.

Varying the perpendicular diffusion coefficient D allows us to estimate the inherent perpendicular diffusion in Wendelstein 7-X curvilinear grids. Figure 14 illustrates how the proportion of the test function f at the 150th timestep compares to the total test function with zero perpendicular diffusion, f_0, for various values of D in a Wendelstein 7-X grid with a resolution of $132 \times 16 \times 256$ (radial, toroidal, poloidal). This corresponds to a resolution of approximately 0.3 mm—although this obviously is not uniform—which is a relatively coarse resolution for a Wendelstein 7-X turbulence study ($\rho_s \approx 0.1$ mm).

Figure 14 indicates that the inherent numerical perpendicular diffusion caused by pollution from parallel dynamics is less than a factor of 10^{-9} times smaller than the parallel diffusion, as this is where the points begin to diverge significantly from the zero-diffusion case (as indicated by the dashed line at $f_{150}/f_0 = 1.0$). This inherent perpendicular diffusion is sufficiently less than transport due to plasma drifts and turbulence [16]. This is encouraging as this result is for a

moderate-resolution grid, and higher-resolution grids will most likely be necessary for future turbulence simulations in Wendelstein 7-X.

5.2. W7-X curvilinear poloidal grid for the edge and scrape-off-layer

The grids described in the previous section are generated from VMEC [37, 38] equilibria, which assume closed flux surfaces. The edge of Wendelstein 7-X is much more complex as it includes magnetic islands and stochastic magnetic field lines. As such, another tool must be developed to trace field lines for grids which can accurately describe this region. To this end, development is ongoing to generate grids based on vacuum field solvers. Figures 15(a) and (c) illustrate two different toroidal positions of one such grid, which uses the Wendelstein 7-X web services vacuum field solver and components database [39] to provide a magnetic field. The inner surface is generated by tracing flux surfaces using the vacuum field solver, which simplifies core boundary conditions and potential coupling to core profiles and sources, and the outer surface is generated based on a description of the Wendelstein 7-X divertor and first wall developed by Michael Drevlak for fast particle calculations, and is also available on the Wendelstein 7-X webservices.

Figures 15(b) and (d) display the resulting flux surfaces calculated by simulating a parallel diffusion equation on the vacuum curvilinear grid, and overplot an example of a Poincaré plot for a nearby flux surface. This grid serves as a promising first step towards full edge Wendelstein 7-X simulations, which await a further publication following an effort to further parallelize FCI calculations and the implementation of a radial electric field profile. In addition to vacuum field solvers, Zoidberg has also been modified to use EXTENDER [40], allowing both plasma-generated magnetic fields and a smooth vacuum solution outside of the last closed flux surface.
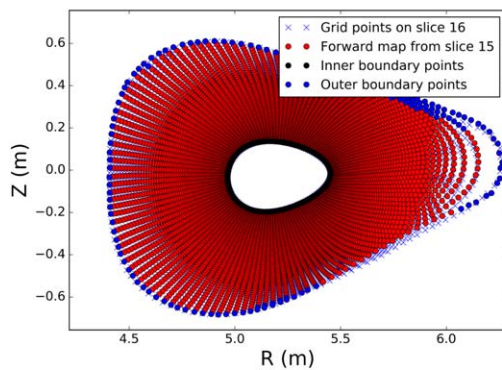
6. Conclusions

The first fluid turbulence simulations in a non-axisymmetric geometry have been performed, following the extension of
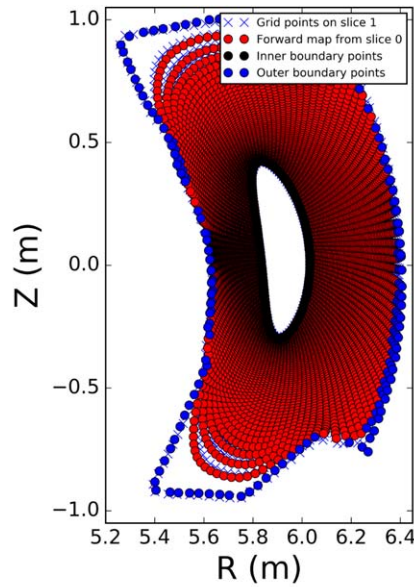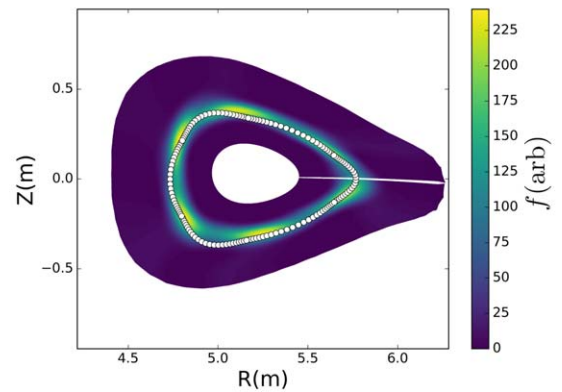
(a) Curvilinear grid for the bean-shaped-cross section of W7-X
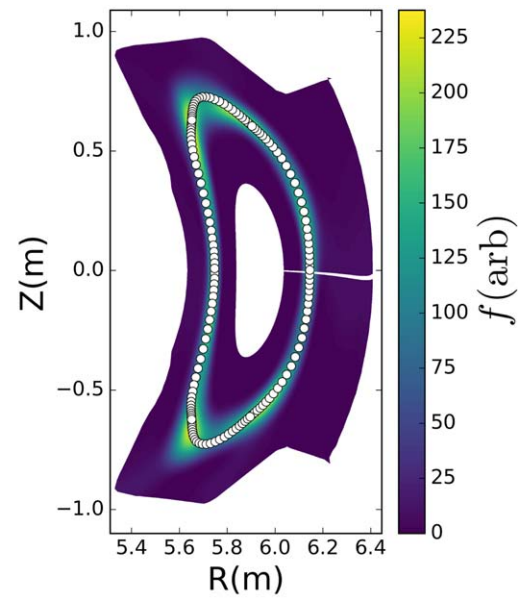
(b) Simulated surfaces for the bean-shaped-cross section

(c) Curvilinear grid for the triangular-cross section of W7-X

(d) Simulated flux surfaces for the triangular-cross section

Figure 15. (a), (c) Curvilinear grid as generated using by the Zoidberg grid generator indicating grid points (blue crosses), and field line maps (circles) for the FCI operators which land inside the domain (red), or leave through the inner (black) or outer (blue) surface. (b), (d): the calculated flux surfaces by parallel heat diffusion simulations for two toroidal locations in W7-X, where the contour is the quantity f in equation (28).

the `BOUT++` framework to allow the metric tensors to vary in three dimensions. This project has therefore yielded the first fluid simulation framework capable of providing edge and scrape-off-layer turbulence simulations in three dimensional geometries, which is necessary for numerical support of experiments on modern tokamaks and stellarators. One major advancement is the implementation of curvilinear grids for use with the FCI method. Initial simulations of filament propagation in non-axisymmetric geometry have been performed, and the filaments have been characterized to propagate in the inertially-limited regime. Furthermore, simulations indicate that even a weakly-non-axisymmetric field can significantly alter the propagation of filaments. The long connection lengths of the scrape-off-layer in non-axisymmetric geometries facilitates the establishment of parallel nonuniformity, an effect which must be considered when interpreting experimental data. For instance, in a system with strongly varying drive, a measurement of a turbulent structure at one toroidal location cannot simply be extrapolated uniformly along the field line. Instead, one must account for the non-uniform drive to accurately map the transport at other toroidal locations. Since three dimensional effects are becoming increasingly important—for instance the application of edge magnetic perturbations—the results presented here are applicable to both tokamak and stellarator configurations.

Future work will include simulations of filaments in the Wendelstein 7-X stellarator, where the non-uniform drive of a filament can be more pronounced. The curvature drive in Wendelstein 7-X reverses direction relative to the major radius within a single field period, which could lead to highly non-uniform propagation of filaments.

Acknowledgments

Appendix A. Modifications to numerical operators

The poloidally curvilinear coordinate system used in this work dictates that numerical operators in the perpendicular (*x*–*z*) plane must carefully incorporate the geometry into the calculation. Here we will concentrate on two operators in particular—Poisson brackets and an example of a curvature operator.

The operator $\frac{1}{B}\mathbf{b} \times \nabla g \cdot \nabla f$ appears often in plasma models and represents phenomena such as $\mathbf{E} \times \mathbf{B}$ advection. It often appears in equations in the form of Poisson brackets, and is what is referred to here as the bracket operator. To determine the modifications for the bracket operator in BOUT ++, we start by defining real space coordinates $R(x, z)$ and $Z(x, z)$ which depend on the radial coordinate *x* and the poloidal coordinate *z*. In the current formulation, *x* ranges from 0 to 1, and *z* from 0 to 2π. From here, we determine the covariant coordinate vectors by taking derivatives along the real-space coordinates:

$$e_{\mathbf{i}} = \frac{\partial}{\partial \mathbf{x_i}}\begin{pmatrix} R \\ Z \end{pmatrix}, \qquad (A1)$$

where $\mathbf{x_i}$ is either the *x* or *z* coordinate. We can now define the metric components as:

$$g_{xx} = \mathbf{e_x} \cdot \mathbf{e_x} \quad g_{xz} = \mathbf{e_x} \cdot \mathbf{e_z} \quad g_{zz} = \mathbf{e_z} \cdot \mathbf{e_z} \qquad (A2)$$

The *y*-direction is considered to be orthogonal to the $x - z$ plane, and is defined as the toroidal angle spanning 0 to 2π. The nonzero metric components are therefore simply:

$$g_{yy} = R^2 \quad g^{yy} = \frac{1}{R^2}, \qquad (A3)$$

where R is the major radius. The unit vector **b** is considered to be perpendicular to the *x*–*z* plane and is defined as:

$$\mathbf{b} = \frac{\mathbf{e_y}}{\sqrt{\mathbf{e_y} \cdot \mathbf{e_y}}} = \frac{\mathbf{e_y}}{\sqrt{g_{yy}}} = \nabla y \sqrt{g_{yy}}. \qquad (A4)$$

We can then begin to construct the bracket operator by taking:

$$\mathbf{b} \times \nabla g = \sqrt{g_{yy}}\left[\frac{\partial g}{\partial x}\nabla y \times \nabla x + \frac{\partial g}{\partial z}\nabla y \times \nabla z\right] \qquad (A5)$$

$$= \sqrt{g_{yy}}\left[-\frac{\partial g}{\partial x}\frac{1}{J}e_{\mathbf{z}} + \frac{\partial g}{\partial z}\frac{1}{J}e_{\mathbf{x}}\right], \qquad (A6)$$

where we have used that $J^2 = det(g_{ij})$. Finally, by taking the dot product with ∇f, we get:

$$\mathbf{b} \times \nabla g \cdot \nabla f = \frac{\sqrt{g_{yy}}}{J}\left[-\frac{\partial g}{\partial x}\frac{\partial f}{\partial z} + \frac{\partial g}{\partial z}\frac{\partial f}{\partial x}\right]. \qquad (A7)$$

The terms in the square brackets is defined as the Poisson bracket, which is what is conventionally described in BOUT++ by the bracket operator. Noting this, we arrive finally at:

$$\frac{1}{B}\mathbf{b} \times \nabla g \cdot \nabla f = \frac{\sqrt{g_{yy}}}{JB}[g, f], \qquad (A8)$$

where we see that a coefficient of $\frac{\sqrt{g_{yy}}}{JB}$ is required for proper calculation of $\mathbf{E} \times \mathbf{B}$ advection in curvilinear grids. In Clebsch coordinates, however, it is worth noting that $\nabla z \times \nabla x = \frac{1}{J}\mathbf{e_y} = \mathbf{B}$ and therefore $\sqrt{g_{yy}}/J = B$ and this coefficient becomes 1.

Curvature effects are one of the most important aspects of turbulence simulations, as this can drive drifts and ballooning behavior which contributes to radial transport. The introduction of curvilinear poloidal grids has necessitated careful implementation of curvature operators. To determine the effects of curvature on a quantity *f*, we must determine how to calculate $(\mathbf{b} \times \kappa) \cdot \nabla f$. As a simple example to illustrate this, we begin by assuming that the curvature vector is of the form:

$$\kappa = \mathbf{b} \cdot \nabla\mathbf{b} \approx -\frac{1}{R}\nabla R \qquad (A9)$$

$$= -\frac{1}{R}\left(\nabla x\frac{\partial}{\partial x}R + \nabla y\frac{\partial}{\partial y}R + \nabla z\frac{\partial}{\partial z}R\right) \qquad (A10)$$

we can then determine:

$$(\mathbf{b} \times \kappa) = \nabla y \sqrt{g_{yy}} \times \left(\nabla x\frac{\partial}{\partial x}R + \nabla y\frac{\partial}{\partial y}R + \nabla z\frac{\partial}{\partial z}R\right) \qquad (A11)$$

$$= -\frac{\sqrt{g_{yy}}}{RJ}e_{\mathbf{x}}\frac{\partial R}{\partial z} + \frac{\sqrt{g_{yy}}}{RJ}e_{\mathbf{z}}\frac{\partial R}{\partial x} \qquad (A12)$$

$$= -\frac{e_{\mathbf{x}}}{J}\frac{\partial R}{\partial z} + \frac{e_{\mathbf{z}}}{J}\frac{\partial R}{\partial x} \qquad (A13)$$

which, when dotted with ∇f, then allows the inclusion of curvature effects in curvilinear poloidal grids. This form of the curvature operator can then be used for large-aspect ratio simulations where the magnetic field varies inversely with major radius, an approximation which is often used in plasma fluid turbulence simulations [17, 33]. A more general curvature operator is derived in section 4.1.

Appendix B. Derivation of inertial scaling

The filaments presented in this work are seeded in the closed field line region, and should be inertially limited. Here we repeat a derivation of the inertial blob scaling which can be achieved by examining a current balance, starting by defining

the diamagnetic current:

$$\nabla \cdot \mathbf{J}_{\mathrm{d}ia} = \nabla \cdot \left[\frac{enT_e}{B} \frac{\mathbf{B} \times \nabla B}{B^2} \right] \tag{B1}$$

$$\sim \frac{1}{\delta_\perp} \frac{eT\delta n}{BR_c}, \tag{B2}$$

where we have assumed:

$$\frac{\mathbf{B} \times \nabla B}{B^2} \sim \frac{1}{R_c}.$$

When considering inertially limited filaments, the diamagnetic current is balanced through the polarization current:

$$\nabla \cdot \mathbf{J_{pol}} = \nabla \cdot \left[\frac{nm_i}{B^2} \frac{\mathrm{d}\mathbf{E}_\perp}{\mathrm{d}t} \right]. \tag{B3}$$

We assume the perpendicular electric field varies as:

$$\frac{\mathrm{d}\mathbf{E}_\perp}{\mathrm{d}t} \sim \frac{v_r}{\delta_\perp} \frac{\phi}{\delta_\perp}. \tag{B4}$$

We also assume that the radial velocity v_r is dominated by the $\mathbf{E} \times \mathbf{B}$ velocity, which allows us to write:

$$v_r \sim \frac{E}{B} \sim \frac{\phi}{\delta_\perp B}. \tag{B5}$$

Therefore the polarization current can be approximated as:

$$\nabla \cdot \mathbf{J_{pol}} \approx \frac{1}{\delta_\perp} \frac{nm_i}{B^2} \frac{v_r^2 B}{\delta_\perp} \tag{B6}$$

which, when equating with equation (B2), gives:

$$v_r = c_s \left[\frac{\delta_\perp}{R_c} \frac{\delta n}{n_0 + \delta n} \right]^{1/2}, \tag{B7}$$

where we have used that $c_s = \sqrt{T/m_i}$, and assumed that the density consists of a background and a perturbation, $n = n_0 + \delta_n$. Therefore, in inertially limited systems where the diamagnetic current is balanced by the polarization current, the blob velocity scales as $\delta_\perp^2$.

ORCID iDs

B Shanahan https://orcid.org/0000-0002-8766-8542
B Dudson https://orcid.org/0000-0002-0094-4867
P Hill https://orcid.org/0000-0003-3092-1858

References

[1] Ho D D-M and Kulsrud R M 1987 *Phys. Fluids (1958-1988)* **30** 442
[2] Helander P 2014 *Rep. Prog. Phys.* **77** 087001
[3] Beidler C *et al* 1990 *Fusion Sci. Technol.* **17** 148
[4] Kocsis G *et al* 2017 *44th EPS Conf. on Plasma Physics*
[5] Wurden G A *et al* 2017 *Nucl. Fusion* **57** 056036
[6] Görler T, Lapillonne X, Brunner S, Dannert T, Jenko F, Merz F and Told D 2011 *J. Comput. Phys.* **230** 7053
[7] Ricci P, Halpern F D, Jolliet S, Loizu J, Mosetto A, Fasoli A, Furno I and Theiler C 2012 *Plasma Phys. Control. Fusion* **54** 124047
[8] Tamain P, Bufferand H, Ciraolo G, Colin C, Ghendrih P, Schwander F and Serre E 2014 *Contrib. Plasma Phys.* **54** 555
[9] Stegmeir A, Coster D, Maj O, Hallatschek K and Lackner K 2016 *Comput. Phys. Commun.* **198** 139
[10] Hariri F and Ottaviani M 2013 *Comput. Phys. Commun.* **184** 2419
[11] Shanahan B W, Hill P and Dudson B D 2016 *J. Phys:.Conf. Series* **775** 012012
[12] Hill P, Shanahan B and Dudson B 2017 *Comput. Phys. Commun.* **213** 9
[13] Dudson B D, Umansky M V, Xu X Q, Snyder P B and Wilson H R 2009 *Comput. Phys. Commun.* **180** 1467
[14] Dudson B *et al* 2015 *J. Plasma Phys.* **81** 365810104
[15] Dudson B, Farley S and McInnes L C 2012 arXiv:1209.2054
[16] Günter S, Yu Q, Krüger J and Lackner K 2005 *J. Comput. Phys.* **209** 354
[17] Shanahan B W and Dudson B D 2016 *Plasma Phys. Control. Fusion* **58** 125003
[18] Loizu J, Ricci P, Halpern F D and Jolliet S 2012 *Phys. Plasmas* **19** 122307
[19] Salari K and Knupp P 2000 Code verification by the method of manufactured solutions *Technical Report* SAND2000-1444 Sandia National Labs
[20] Roache P J 2002 Transactions-American society of mechanical engineers *J. Fluids Eng.* **124** 4
[21] Dudson B D, Madsen J, Omotani J, Hill P, Easy L and Løiten M 2016 *Phys. Plasmas* **23** 062303
[22] Balay S, Gropp W D, McInnes L C and Smith B F 1997 *Modern Software Tools for Scientific Computing* (Berlin: Springer) pp 163–202
[23] Hariri F, Hill P, Ottaviani M and Sarazin Y 2014 *Phys. Plasmas* **21** 082509
[24] Thompson J F 1982 *Appl. Math. Comput.* **10** 79
[25] Thompson J F, Warsi Z U and Mastin C W 1985 *Numerical Grid Generation: Foundations and Applications* vol 45 (Amsterdam: North-holland)
[26] 'Bout++ manual' http://bout-dev.readthedocs.io (Accessed: 01 June 2018) (https://doi.org/10.5281/zenodo.1464133)
[27] Tamain P, Bufferand H, Ciraolo G, Colin C, Galassi D, Ghendrih P, Schwander F and Serre E 2016 *J. Comput. Phys.* **321** 606
[28] Riva F *et al* 2016 *Plasma Phys. Control. Fusion* **58** 044005
[29] Walkden N, Easy L, Militello F and Omotani J 2016 *Plasma Phys. Control. Fusion* **58** 115010
[30] Arakawa A and Lamb V R 1977 *General Circulation Models of the Atmosphere* (*Methods in Computational Physics: Advances in Research and Applications* vol 17) ed J Chang (Amsterdam: Elsevier) pp 173–265
[31] Fuchert G, Birkenmeier G, Ramisch M and Stroth U 2016 *Plasma Phys. Control. Fusion* **58** 054005
[32] Easy L, Militello F, Omotani J, Dudson B, Havlíčková E, Tamain P, Naulin V and Nielsen A H 2014 *Phys. Plasmas* **21** 122515
[33] Walkden N, Dudson B, Easy L, Fishpool G and Omotani J 2015 *Nucl. Fusion* **55** 113022
[34] Loizu J, Hudson S, Nührenberg C, Geiger J and Helander P 2017 *J. Plasma Phys.* **83** 715830601
[35] D'Ippolito D A, Myra J R and Zweben S J 2011 *Phys. Plasmas* **18** 060501
[36] Krasheninnikov S I, D'Ippolito D A and Myra J R 2008 *J. Plasma Phys.* **74** 679
[37] Hirshman S and Betancourt O 1991 *J. Comput. Phys.* **96** 99
[38] Hirshman S P and Whitson J 1983 *Phys. Fluids* **26** 3553
[39] Bozhenkov S, Geiger J, Grahl M, Kißlinger J, Werner A and Wolf R 2013 *Fusion Eng. Des.* **88** 2997
[40] Drevlak M, Monticello D and Reiman A 2005 *Nucl. Fusion* **45** 731