LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-JRNL-409772

Status and Verification of Edge Plasma Turbulence Code BOUT

M. V. Umansky, X. Q. Xu, B. Dudson, L. L. LoDestro, J. R. Myra

January 12, 2009

Computer Physics Communications

Disclaimer

Status and Verification of Edge Plasma Turbulence Code BOUT

M.V. Umansky, X.Q. Xu, B. Dudson[1], L.L. LoDestro, and J.R. Myra[2]

Lawrence Livermore National Laboratory, CA 94550, USA

[1] *University of York, UK*

[2] *Lodestar Research Corporation, Boulder, CO 80301, USA*

8 January 2009

Abstract

The BOUT code is a detailed numerical model of tokamak edge turbulence based on collisional plasma fluid equations. BOUT solves for time evolution of plasma fluid variables: plasma density N_i, parallel ion velocity $V_{||i}$, electron temperature T_e, ion temperature T_i, electric potential ϕ, parallel current $j_{||}$, and parallel vector potential $A_{||}$, in realistic 3D divertor tokamak geometry. The current status of the code, physics model, algorithms, and implementation is described. Results of verification testing are presented along with illustrative applications to tokamak edge turbulence.

Key words:

edge plasma turbulence

PACS:

1 Introduction

In plasma confinement devices the area near the last closed flux surface (LCFS), encompassing the region of open field lines intersecting material walls, i.e. the scrape-off layer (SOL), and the outer part of the main plasma inside of LCFS, is referred to as the edge or boundary plasma region. The boundary region in fusion plasmas has been recognized as a critically important component dramatically affecting the overall performance of the device [1]. Due to the complex physics and geometry of edge plasma, an analytic treatment does not appear to be feasible in most cases. Therefore analysis of boundary plasmas heavily relies on numerical modeling. Numerical simulation of turbulence in fusion boundary plasmas has been developing since the early work [2–4]. Due to the relatively high collisionality of edge plasmas, most models are based on the collisional plasma fluid equations [5]. Applicability of the collisional plasma fluid model implies short collisional mean free path, $\lambda_{ei}/L_{||} \ll 1$, and small gyroradius, $k_\perp \rho \ll 1$. The collisionality condition is reasonably well satisfied in many present day fusion experiments but will be certainly violated in next generation devices. Furthermore, the small gyroradius condition becomes questionable in edge pedestal regions with steep gradients. Therefore more general edge models, gyrofluid [6], and gyrokinetic [7,8], are being developed. Still, the general phenomenology of edge plasmas seems to be rather similar for strongly collisional and weakly collisional edge plasmas; and that lends some assurance that some, perhaps much, of essential edge plasma physics can be captured by collisional fluid plasma models.

One of the fullest collisional fluid models for edge plasma is the BOUT code [9,10]. BOUT solves for the time evolution of a set of plasma fluid variables: plasma density N_i, parallel ion velocity $V_{||i}$, electron temperature T_e, ion temperature T_i, electric potential ϕ, potential vorticity ϖ, parallel current $j_{||}$, and the parallel component of the vector potential $A_{||}$, in realistic 3D divertor tokamak geometry. BOUT has a long history of development, starting from the late-1990s, and the code has significantly evolved in recent years. A review of BOUT development and applications was given in [11]. This paper attempts to provide a coherent description of the current status of the BOUT model, algorithms, and implementation, along with some results of verification testing

and illustrative applications to tokamak edge turbulence.

2 Physics model

The fluid plasma model in BOUT is described in [10] and given in the Appendix for reference. These equations are based on the Braginskii fluid equations for collisional plasma [5], with the additional assumptions of low plasma pressure, $\beta = 8\pi(p_e + p_i)/B^2 \ll 1$, and field aligned perturbations, $k_{||}/k_\perp \ll 1$.

One should note that the Braginskii equations assume an ordering in which the ion mean flow is on the order of the ion thermal speed. An alternative ordering scheme for weak flow, on the order of the ion diamagnetic velocity was proposed in [12,13] and based on it a set of collisional plasma fluid equations was derived in [14]. In the experiments the parallel component of the flow can sometimes be large, M$\sim$0.5, [15]. Fortunately, the difference between equations in [10] and [14] is relatively small, it boils down to additional terms in the viscous stress tensor [11], which so far have not been demonstrated to have any effect on the stability of low frequency dynamics [16].

The dynamic equations solved in the BOUT code are the equations for the plasma density N_i, parallel ion velocity $V_{||i}$, electron temperature T_e, ion temperature T_i, electric potential ϕ, vorticity ϖ. parallel current $j_{||}$, and parallel vector potential $A_{||}$.

The vorticity ϖ is defined as

$$\varpi \equiv \nabla \cdot (eN_i \nabla_\perp \phi + \nabla_\perp P_i) = eN_i \nabla_\perp^2 \phi + eN_i \nabla_\perp \phi \cdot \nabla_\perp \ln N_i + \nabla_\perp^2 P_i \quad (1)$$

Here $\nabla_\perp^2$ is the part of the Laplacian operator perpendicular to the magnetic field,

$$\nabla_\perp^2 = \nabla^2 - \partial_{||}^2, \quad (2)$$

and $\partial_{||}$ is the derivative in the direction parallel to the magnetic field,

$$\partial_{||} = \frac{1}{B}\vec{B} \cdot \vec{\nabla} \tag{3}$$

For given ϖ, N_i and P_i Eq.(1) can be solved for the electric potential ϕ which is commonly referred to as "vorticity inversion".

The canonical parallel electron velocity $A_{j||}$ combining the parallel components of the electron velocity and the magnetic vector potential is defined as

$$A_{j||} = V_{||e} - (e/m_e c)A_{||} \tag{4}$$

Then from the relation

$$\nabla^2 A_{||} = -\frac{4\pi}{c} j_{||} \tag{5}$$

one obtains the Helmholtz equation for $A_{||}$:

$$\nabla_\perp^2 A_{||} - \frac{{\omega_{pe}}^2}{c^2} A_{||} = \frac{4\pi}{c} e N_i (A_{j||} - V_{||i}), \tag{6}$$

Here the perpendicular Laplacian replaces the full Laplacian since $k_{||}/k_\perp$ is assumed very small. From given $A_{j||}$, $V_{||i}$, and N_i one can solve Eq.(6) for $A_{||}$, and then find $V_{||e}$ from Eq.(4).

The system of dynamic equations used in BOUT can be written as

$$\frac{\partial}{\partial t} \begin{pmatrix} N_i \\ V_{||i} \\ T_e \\ T_i \\ \varpi \\ A_{j||} \end{pmatrix} = \vec{F}(N_i, V_{||i}, T_e, T_i, \phi, j_{||}, \varpi, A_{||}) \tag{7}$$

The exact form of the right-hand side function $\vec{F}$ can be found in the Appendix. It is important to note that $\vec{F}$ contains ϕ, $j_{||}$ and $A_{||}$, so for every evaluation of $\vec{F}$ one needs to solve first Eq.(1) and Eq.(6) to find those. In terms of its general structure, the right-hand-side function $\vec{F}$ contains algebraic operations and spatial differential operators of the first and second order. These are: the parallel derivative operator $\partial_{||}$, the advection operator $v_E \cdot \nabla$, the perturbed parallel derivative operator $\tilde{b}\cdot\nabla$, the curvature operator $b\times\kappa\cdot\nabla$, and the perpendicular Laplacian operator $\nabla_\perp^2$. To evaluate the differential operators one needs to establish an appropriate coordinate system, as discussed further in the next session.

3 Magnetic geometry

In our notation, ζ is the geometric toroidal angle, θ is the poloidal angle coordinate defined from the poloidal flux, $\psi = (1/2\pi)\int B_{pol} ds$, with local orthogonality condition $\nabla\psi \cdot \nabla\theta$=0 (although the model can be extended to use a more general θ coordinate). Positive B_ζ is in the ζ direction, i.e. counter-clock-wise (looking from the top). For negative B_ζ, which is considered the "normal" case, the ion $\vec{\nabla}$B drift is down [17]. For the poloidal component, B_θ, the positive sign by convention corresponds to the direction from the inner plate to the outer one, see Fig.(1).

3.1 Flux-tube domain

The computational domain used in BOUT has the shape of a magnetic flux tube starting with a rectangular cross-section from a reference poloidal location. The toroidal extent of the domain is taken to be an integer fraction of the full circle, 2π/n, where n is chosen to provide optimal spatial resolution for the modes considered. The perpendicular size of the domain, $L_\perp$, is taken large compared to the turbulence perpendicular correlation scale $\lambda_\perp^{corr}$.

$$L_\perp \approx R\frac{2\pi}{n}\frac{B_p}{B_t} \gg \lambda_\perp^{corr} \tag{8}$$

For typical tokamak parameters, $R \sim$1 m, $\lambda_\perp^{corr} \sim$1 cm, and $B_p/B_t \sim$0.1 one would use n$\sim$10. Using a larger toroidal size, up to 2π, is certainly possible but that would require many more toroidal grid points to resolve the relevant toroidal modes. Figure (2) shows a flux-tube domain with toroidal size $2\pi/5$, and formation of the full toroidal surface by joining five such domains shifted toroidally. For a divertor tokamak edge domain with separatrices, the computational domain splits in two or more flux-tubes representing topologically distinct flux regions.

One should note that in spite of using a flux-tube domain BOUT is not a flux-tube code but a global code; it uses real magnetic geometry and the full radial profile information of the background plasma rather than just the profiles e-folding length, and does not use periodic radial boundary conditions as flux-tube codes do.

3.2 Field-aligned coordinates

Due to the high anisotropy of the transport, in magnetized plasma the physical phenomena usually have a field-aligned spatial structure. In theoretical and computational plasma physics, using coordinates aligned with the magnetic field is often a natural choice simplifying calculations. Similarly, BOUT uses a field-aligned coordinate system described below; see Fig. (3).

The poloidal flux ψ can have either maximum or minimum at the magnetic axis, so $\vec{\nabla}\psi$ can point from plasma to the wall, or the opposite way. In the former case, (ψ, θ, ζ) form a right-handed orthogonal coordinate system, in the latter it is left-handed.

The magnetic field can be represented in the standard way as

$$\vec{B} = F(\psi)\vec{\nabla}\zeta + \vec{\nabla}\zeta \times \vec{\nabla}\psi, \tag{9}$$

where $F = RB_\zeta$, the first term is the toroidal field, B_ζ, and the second one is the poloidal field, B_θ.

For the coordinates (ψ, θ, ζ) the Jacobian is given by

$$J^{-1} = [\vec{\nabla}\psi \times \vec{\nabla}\theta] \cdot \vec{\nabla}\zeta = [\vec{\nabla}\zeta \times \vec{\nabla}\psi] \cdot \vec{\nabla}\theta = \vec{B}_\theta \cdot \vec{\nabla}\theta = B_\theta / h_\theta; \tag{10}$$

so it can be either positive or negative, depending on the sign of B_θ. Here h_θ is the effective local minor radius of curvature

$$h_\theta = 1/|\vec{\nabla}\theta| = |\partial\vec{r}/\partial\theta|. \tag{11}$$

Note that orthogonality of $\vec{\nabla}\psi$ and $\vec{\nabla}\theta$ is used for the latter equality.

To define the radial coordinate x so that it grows outwards, from plasma to the wall, we introduce the sign of the poloidal field, $\sigma_\psi = B_\theta/|B_\theta|$. The field-aligned coordinates (x, y, z) are constructed from (ψ, θ, ζ) as follows:

$$\begin{aligned} x &= \sigma_\psi(\psi - \psi_s) \\ y &= \theta \\ z &= \sigma_\psi \left(\zeta - \int_{\theta_0}^{\theta} \nu(\psi, \theta) d\theta \right) \end{aligned} \tag{12}$$

where ψ_s is the separatrix flux, θ_0 is a reference poloidal location, and ν is the local pitch of the magnetic field

$$\nu = \frac{B_\zeta h_\theta}{B_\theta R} = \frac{(F/R)h_\theta}{B_\theta R} = FJ/R^2. \tag{13}$$

Note that in these coordinates constant x and z means staying on the same magnetic line.

Since

$$\vec{\nabla}x = \sigma_\psi \vec{\nabla}\psi = \sigma_\psi R B_\theta \hat{e}_x, \tag{14}$$

and

$$\vec{\nabla}z = \sigma_\psi \left(\vec{\nabla}\zeta - \vec{\nabla} \left[\int_{\theta_0}^{\theta} \nu(\psi, \theta) d\theta \right] \right) = \tag{15}$$

$$\sigma_\psi \left(\hat{e}_\zeta / R - \left[\int_{\theta_0}^{\theta} \frac{\partial \nu(\psi,\theta)}{\partial \psi} d\theta \right] \vec{\nabla}\psi - \nu(\psi,\theta)\vec{\nabla}\theta \right),$$

the magnetic field is simply represented in the field-aligned coordinates in the Clebsch form as

$$\vec{B} = \vec{\nabla} z \times \vec{\nabla} x. \tag{16}$$

The Jacobian of (x, y, z) is given by

$$J^{-1} = [\vec{\nabla} x \times \vec{\nabla} y] \cdot \vec{\nabla} z = B_\theta / h_\theta. \tag{17}$$

Note that (17) is equivalent to (10), and that has a simple geometric interpretation.

Denoting the integrated local shear as I,

$$I(\psi,\theta) = \int_{\theta_0}^{\theta} \frac{\partial \nu(\psi,\theta)}{\partial \psi} d\theta \tag{18}$$

one can derive the dot-products and cross-products of the gradients, ∇x, ∇y, ∇z, summarized in the tables below.

Table 1

Dot products (a, b)

$a \setminus b$	$\vec{\nabla} x$	$\vec{\nabla} y$	$\vec{\nabla} z$
$\vec{\nabla} x$	$(RB_\theta)^2$	0	$-I(RB_\theta)^2$
$\vec{\nabla} y$	0	$1/h_\theta^2$	$-\sigma_\psi \nu / h_\theta^2$
$\vec{\nabla} z$	$-I(RB_\theta)^2$	$-\sigma_\psi \nu / h_\theta^2$	$I^2(RB_\theta)^2 + B^2/(RB_\theta)^2$

Table 2

Cross products $[a, b]$

$a \setminus b$	$\vec{\nabla}x$	$\vec{\nabla}y$	$\vec{\nabla}z$
$\vec{\nabla}x$	0	$(R\|B_\theta\|/h_\theta)\hat{e}_\zeta$	$-\vec{B}$
$\vec{\nabla}y$	$-(R\|B_\theta\|/h_\theta)\hat{e}_\zeta$	0	$(\sigma_\psi/(Rh_\theta))\hat{e}_x + (IR\|B_\theta\|/h_\theta)\hat{e}_\zeta$
$\vec{\nabla}z$	$\vec{B}$	$-(\sigma_\psi/(Rh_\theta))\hat{e}_x - (IR\|B_\theta\|/h_\theta)\hat{e}_\zeta$	0

Then one has for the contravariant metric tensor $g^{ij} \equiv e^i \cdot e^j = \nabla u^i \cdot \nabla u^j$

$$g^{ij} = \begin{pmatrix} (RB_\theta)^2 & 0 & -I(RB_\theta)^2 \\ 0 & 1/h_\theta^2 & -\sigma_\psi \nu/h_\theta^2 \\ -I(RB_\theta)^2 & -\sigma_\psi \nu/h_\theta^2 & I^2(RB_\theta)^2 + B^2/(RB_\theta)^2 \end{pmatrix}; \tag{19}$$

and the covariant metric tensor, $g_{ij} = [g^{ij}]^{-1}$, is

$$g_{ij} = \begin{pmatrix} \frac{1}{(RB_\theta)^2} + I^2R^2 & \sigma_\psi \frac{B_\xi h_\theta IR}{B_\theta} & IR^2 \\ \sigma_\psi \frac{B_\xi h_\theta IR}{B_\theta} & \frac{B^2 h_\theta^2}{B_\theta^2} & \sigma_\psi \frac{B_\xi h_\theta R}{B_\theta} \\ IR^2 & \sigma_\psi \frac{B_\xi h_\theta R}{B_\theta} & R^2 \end{pmatrix} \tag{20}$$

Knowledge of the metric tensor components provides the full information needed for evaluating the spatial differential operators.

4 Differential operators

To represent the BOUT dynamic equations in the field-following (x, y, z) coordinates one needs to express several differential operators, which we address next.

4.1 Parallel derivative $b_0 \cdot \vec{\nabla}$

From the definition of the parallel derivative with respect to unperturbed magnetic field one finds

$$\frac{1}{B}\vec{B}_0 \cdot \vec{\nabla} = \frac{1}{B}(\vec{\nabla}z \times \vec{\nabla}x) \cdot (\vec{\nabla}y\frac{\partial}{\partial y}) = \frac{1}{B}\frac{1}{J}\frac{\partial}{\partial y} = \frac{B_{\theta 0}}{Bh_\theta}\frac{\partial}{\partial y} \tag{21}$$

4.2 Perpendicular Laplacian $\nabla^2_\perp$

The perpendicular Laplacian is defined as

$$\nabla^2_\perp = \nabla^2 - \frac{\partial^2}{\partial^2_{||}} \tag{22}$$

Using the general expression

$$\nabla^2 = \frac{1}{J}\sum_i \frac{\partial}{\partial x_i}\left[J(\sum_j \vec{\nabla}x_i \cdot \frac{\partial}{\partial x_j}\vec{\nabla}x_j)\right] \tag{23}$$

or, more explicitly,

$$\begin{aligned}\nabla^2 = |\vec{\nabla}x|^2\frac{\partial^2}{\partial x^2} + |\vec{\nabla}y|^2\frac{\partial^2}{\partial y^2} + |\vec{\nabla}z|^2\frac{\partial^2}{\partial z^2} + \\ +2(\vec{\nabla}x \cdot \vec{\nabla}y)\frac{\partial^2}{\partial x \partial y} + 2(\vec{\nabla}x \cdot \vec{\nabla}z)\frac{\partial^2}{\partial x \partial z} + 2(\vec{\nabla}y \cdot \vec{\nabla}z)\frac{\partial^2}{\partial y \partial z} + \\ +\nabla^2 x\frac{\partial}{\partial x} + \nabla^2 y\frac{\partial}{\partial y} + \nabla^2 z\frac{\partial}{\partial z}\end{aligned} \tag{24}$$

one first eliminates all terms with parallel (i.e. $\partial/\partial y$) derivatives to obtain the perpendicular Laplacian $\nabla^2_\perp$.

Next, assuming a thin annulus, $\Delta R/R \ll 1$, one can drop terms with first derivatives $\partial/\partial x$ and $\partial/\partial z$, which are small compared to those with second derivatives, $\partial^2/\partial x^2$ and $\partial^2/\partial z^2$. Then one obtains the expression

$$\nabla^2_\perp = |\vec{\nabla}z|^2\frac{\partial^2}{\partial z^2} + 2(\vec{\nabla}z \cdot \vec{\nabla}x)\frac{\partial^2}{\partial z \partial x} + |\vec{\nabla}x|^2\frac{\partial^2}{\partial x^2}. \tag{25}$$

Using Table (1) one finally arrives at

$$\nabla_\perp^2 = (RB_\theta)^2 \left(\frac{\partial^2}{\partial x^2} - 2I \frac{\partial^2}{\partial z \partial x} + \left[I^2 + \frac{B^2}{(RB_\theta)^4} \right] \frac{\partial^2}{\partial z^2} \right). \tag{26}$$

4.3 $E \times B$ advection $V_E \cdot \vec{\nabla}$

The $E \times B$ advection operator is

$$V_E \cdot \vec{\nabla} = \frac{c}{B^2} \vec{B} \times \vec{\nabla}\phi \cdot \vec{\nabla} = \frac{c}{B^2} \vec{B} \cdot \vec{\nabla}\phi \times \vec{\nabla}. \tag{27}$$

Then, using vector identities and Table (2), one finds

$$V_E \cdot \vec{\nabla} = \frac{c}{B^2} \vec{B} \times \vec{\nabla}\phi \cdot \vec{\nabla} = \tag{28}$$
$$c \left[(\frac{\partial \phi}{\partial x}\frac{\partial}{\partial y} - \frac{\partial \phi}{\partial y}\frac{\partial}{\partial x})(-\frac{RB_\zeta |B_\theta|}{h_\theta B^2}) + (\frac{\partial \phi}{\partial z}\frac{\partial}{\partial x} - \frac{\partial \phi}{\partial x}\frac{\partial}{\partial z}) - (\frac{\partial \phi}{\partial y}\frac{\partial}{\partial z} - \frac{\partial \phi}{\partial z}\frac{\partial}{\partial y})(I\frac{RB_\zeta |B_\theta|}{h_\theta B^2}) \right]$$

Note that the second term is dominant; it is larger the others by $O(k_\perp / k_{||})$.

4.4 Parallel derivative perturbation $\tilde{\vec{b}} \cdot \vec{\nabla}$

Since

$$\tilde{\vec{b}} = \tilde{\vec{B}}/B = (1/B)[\vec{\nabla} A_{||} \times \vec{b}_0], \tag{29}$$

one can write

$$\tilde{\vec{b}} \cdot \vec{\nabla} = (1/B^2)[\vec{\nabla} A_{||} \times \vec{B}] \cdot \vec{\nabla} = -(1/B^2)[\vec{B} \times \vec{\nabla} A_{||}] \cdot \vec{\nabla}. \tag{30}$$

Thus, similarly to (29),

$$\tilde{\vec{b}} \cdot \vec{\nabla} = (\frac{\partial A_{||}}{\partial x}\frac{\partial}{\partial y} - \frac{\partial A_{||}}{\partial y}\frac{\partial}{\partial x})(B_\zeta \frac{R|B_\theta|}{B^2 h_\theta}) + (\frac{\partial A_{||}}{\partial x}\frac{\partial}{\partial z} - \frac{\partial A_{||}}{\partial z}\frac{\partial}{\partial x}) + \tag{31}$$
$$+(\frac{\partial A_{||}}{\partial y}\frac{\partial}{\partial z} - \frac{\partial A_{||}}{\partial z}\frac{\partial}{\partial y}) \left(I B_\zeta \frac{R|B_\theta|}{B^2 h_\theta} \right).$$

4.5 Curvature advection $[\vec{B} \times \vec{\kappa}] \cdot \vec{\nabla}$

The magnetic-line curvature, $\vec{\kappa}$, is defined as

$$\vec{\kappa} = \vec{b} \cdot (\vec{\nabla}\vec{b}) = -\vec{b} \times (\vec{\nabla} \times \vec{b}) \tag{32}$$

In the BOUT ordering, the curvature is calculated from the unperturbed field, so here $\vec{b}_0$ is used. The operator $[\vec{b}_0 \times \vec{\kappa}] \cdot \vec{\nabla}$ has the form of an advection operator:

$$[\vec{b}_0 \times \vec{\kappa}] \cdot \vec{\nabla} = v_x \frac{\partial}{\partial x} + v_y \frac{\partial}{\partial y} + v_z \frac{\partial}{\partial z}, \tag{33}$$

with the velocity components v_x, v_y, v_z pre-calculated from the given unperturbed field $\vec{B}(r, z)$,

$$\begin{aligned} v_x &= [\vec{b}_0 \times \vec{\kappa}] \cdot \vec{\nabla} x \\ v_y &= [\vec{b}_0 \times \vec{\kappa}] \cdot \vec{\nabla} y \\ v_z &= [\vec{b}_0 \times \vec{\kappa}] \cdot \vec{\nabla} z. \end{aligned} \tag{34}$$

5 Inversion of the Laplacian operators

5.1 Inversion of vorticity

From the definition of the vorticity, Eq.(1), one can find the perturbation $\varpi = \varpi_{total} - \varpi_0$

$$\begin{aligned} \varpi = &(N_{i0} + N_i) q \nabla_\perp^2 (\phi_0 + \phi) + q \nabla_\perp (\phi_0 + \phi) \cdot \nabla_\perp (N_{i0} + N_i) \\ &+ \nabla_\perp^2 (P_{i0} + P_i) - N_{i0} q \nabla_\perp^2 \phi_0 - q \nabla_\perp \phi_0 \cdot \nabla_\perp N_{i0} - \nabla_\perp^2 P_{i0}. \end{aligned} \tag{35}$$

The terms involving $\nabla_\perp \phi_0$ and $\nabla_\perp N_{i0}$ are dropped assuming large gradient scale-length of the equilibrium profiles. Then what is left is

$$\varpi = N_{i0} q \nabla_\perp^2 \phi + N_i q \nabla_\perp^2 \phi + q \nabla_\perp \phi \cdot \nabla_\perp N_i + \nabla_\perp^2 P_i. \tag{36}$$

The nonlinear second and third terms are dropped as small, and what is left is

$$\varpi = N_{i0} q \nabla_\perp^2 \phi + \nabla_\perp^2 P_i. \tag{37}$$

Again, N_{i0} can be put under $\nabla_\perp$ by same argument; and therefore the equation solved is

$$\nabla_\perp^2 \left(q\phi + \frac{P_i}{N_{i0}} \right) = \frac{\overline{\omega}}{N_{i0}}. \tag{38}$$

Using Eq.(26) with $f = q\phi + P_i/N_{i0}$ and applying a Fourier transform in z leads to

$$(RB_\theta)^2 \left(\frac{\partial^2 f_k}{\partial x^2} - 2I(ik_z)\frac{\partial f_k}{\partial x} + \left[I^2 + \frac{B^2}{(RB_\theta)^4} \right] (-k_z^2) f_k \right) = \frac{\varpi_k}{N_{i0}}. \tag{39}$$

Solving the discretized ODE numerically using a linear solver yields f_k; then an inverse Fourier transform yields $f(x,z)$; and from the latter $q\phi = f - P_i/N_{i0}$. Note that in this procedure the radial boundary conditions for N_i, T_i, and ϕ become linked together. It is important to note that it is the dropping of a number of terms that leads to the simple form of vorticity (38) where the coefficients depend on x but not on z. That allows to use an efficient inversion procedure based on FFT in one dimension and solving a tri-diagonal linear system in the other.

5.2 Inversion of the parallel current

Using the Helmholtz equation for $\hat{A}_{||}$, Eq.(6), and applying a Fourier transform in z leads to

$$(RB_\theta)^2 \left(\frac{\partial^2 A_{||k}}{\partial x^2} - 2I(ik_z)\frac{\partial A_{||k}}{\partial x} + \left[I^2 + \frac{B^2}{(RB_\theta)^4} \right] (-k_z^2) A_{||k} \right) - \frac{\omega_{pe}^2}{c^2} A_{||k} = \frac{4\pi}{c} e[N_i(A_{j||} - V_{||i})]_k, \tag{40}$$

Equation (40) leads to a linear banded system of equations with complex coefficients. The solution procedure is same as that for the vorticity.

6 Taking radial derivatives on sheared grids

As the magnetic field in toroidal plasma confinement devices is generally sheared, the field-following grid is sheared too [22–24]. The issue arises then how to implement the finite-difference operator $\partial/\partial x$. Formally, the derivative $\partial/\partial x$ can be approximated by a finite-difference expression, e.g.,

$$\frac{\partial f}{\partial x} \approx 0.5\left[f(ix+1, iy, iz) - f(ix-1, iy, iz)\right]/\Delta x, \tag{41}$$

where ix, iy, iz are grid indices in the x, y, z directions, and Δx is the grid spacing in the x direction.

However, on strongly sheared grids the grid points $A = (ix-1, iy, iz)$ and $B = (ix+1, iy, iz)$ may be very far physically, and the values there are not correlated, so the direct finite-difference expression such as Eq.(41) would not be a good approximation to $\partial f/\partial x$. To circumvent this problem one can use the identity

$$(\frac{\partial f}{\partial \psi})_\zeta = (\frac{\partial f}{\partial x})_z (\frac{\partial x}{\partial \psi})_\zeta + (\frac{\partial f}{\partial z})_x (\frac{\partial z}{\partial \psi})_\zeta \tag{42}$$

where the third coordinate ($\theta = y$) is assumed fixed. Using the definition of the coordinate transformation, Eq.(13), one finds

$$\begin{aligned} (\frac{\partial x}{\partial \psi})_\zeta &= 1 \\ (\frac{\partial z}{\partial \psi})_\zeta &= -\sigma_\psi I(\psi, \theta), \end{aligned} \tag{43}$$

where I is the integrated local shear, Eq.(18). Then one finds for the $\partial f/\partial x$

$$(\frac{\partial f}{\partial x})_z = (\frac{\partial f}{\partial \psi})_\zeta + \sigma_\psi I (\frac{\partial f}{\partial z})_x. \tag{44}$$

On the right-hand side of (44) the second term can be treated by finite-difference directly, while to evaluate the first term one needs to interpolate f to the given ζ at both $ix \pm 1$ layers, and then apply the finite-difference; see Fig. (4). Our lowest-order interpolation is to use grid points at the $ix \pm 1$ layers with the z-index shifted by $\mp \sigma_\psi I \delta\psi/\delta z$, where $\delta\psi$ and δz are grid spacings in the ψ and z directions. For a higher order interpolation two or more toroidal grid points are used.

7 Twist-shift boundary condition

At the branch cut $\theta = 0$ and $\theta = 2\pi$ a special boundary condition is needed to take into account the rotational transform [22–24].

Recalling that local safety factor ν is defined in Eq.(13) as

$$\nu = \frac{h_\theta B_\zeta}{R B_\theta}, \tag{45}$$

where $h_\theta = dl_\theta/d\theta$, and

$$\frac{dl_\theta}{R d\zeta} = \frac{B_\theta}{B_\zeta}, \tag{46}$$

the increment of the toroidal angle ζ corresponding to one poloidal turn is

$$\Delta\zeta = \oint \nu d\theta. \tag{47}$$

Here the integral covers a θ range equal to 2π and does not depend on the starting point. As $\Delta\zeta$ can be anything, depending on the safety factor, the grid lines on $\theta = 0$ don't match those on the $\theta = 2\pi$ side. However, a matching condition can be found using the grid periodicity in the toroidal angle.

Consider a bundle of magnetic lines that has toroidal coordinates with a range $[0, \zeta_{max}]$ at $\theta = 0$, as illustrated in Fig.(5) (a), (b). A magnetic line starts at the branch cut location at point A at the toroidal location ζ_1 and poloidal location $\theta = 0$. The end point B at the other side of the branch cut location, $\theta = 2\pi$, corresponds to the toroidal angle $\zeta'_1 = \zeta_1 + \Delta\zeta$. As the bundle

of magnetic lines is periodically continued in the toroidal angle with period ζ_{max}, one can map the ζ'_1 to the range $[0, \zeta_{max}]$ by the *modulo* operation: $\zeta'_2 = \zeta'_1 mod\ \zeta_{max}$, as shown in Fig. (5) (b). The corresponding toroidal grid index is $iz_2 = integer(\zeta'_2/(\zeta_{max}/nz_{max}))$. In the lowest order interpolation this is the index used for "quasi-ballooning" boundary condition across the branch cut. Similarly one can calculate the matching toroidal index for the magnetic line starting at the outer branch cut and going in the negative θ direction.

8 BOUT implementation

8.1 Implicit time integration

The differential equations in BOUT are discretized in space and then integrated in time as a system of ODEs, i.e., using the method of lines [25]. The solution of the ODEs is done by the package PVODE [26] that performs adaptive implicit time integration. Implicitness gives a great advantage since it removes the stability limitations on the time step; and using the reliable software package for time integration simplifies the code implementation.

8.2 Separating physics and numerics

To make BOUT a flexible modeling framework, the differential operators are represented by a set of separate subroutines. For example, the term in the equations $N_i \nabla_{||} V_{||i}$ in the code appears as `Ni*Div_par(&Vi)`. The building blocks here are functions representing the differential operators, such as `Div_par()` for $\nabla_{||}$ etc. At a lower level, spatial derivatives of fluid fields are calculated by a set of functions implementing various options for finite-differencing. This hierarchical code structure allows one to easily alter the physics model, and turn on and off individual terms, and makes the code more readable, and greatly simplifies the book-keeping and debugging, admittedly somewhat sacrificing the performance.

8.3 Spatial discretization

The spatial discretization is done by the finite difference method, with a number of options available in the code. Spatial derivatives are approximated by finite difference up to the 4th order. Options for discretization of advective terms include linear central and upwind-biased finite-difference schemes up to the 4th order, and nonlinear limiter schemes such as Van Leer [27] and WENO [28]. The nonlinear schemes tend to be most robust, although they are most computationally expensive.

8.4 Parallelization

The parallel implementation of BOUT is described in [29]. The parallel algorithm is based on decomposing the spatial domain into subdomains, where each subdomain belongs to an individual process. The subdomains have guard cells that contain data from neighbor subdomains. For time integration one needs evaluation of the right-hand-side function of the system of ODEs representing the spatially discretized dynamic equations. The first step for evaluation of the right-hand-side function consists of exchanging data between processes to set the guard cell values. Next, the right-hand-side function is evaluated on each subdomain by finite-difference.

9 Code structure

The principal flow chart of BOUT is

```
bout{}
{
  bout_init(); /*import grid data, set run options, allocate memory*/

  for (it=1; it <= itmax; it++) {
    bout_step(); /*perform one time step*/
  }
```

```
  bout_fin(); /*close files, free memory*/
}
```

The function `bout_step()` makes a call to the PVODE package that time-advances the system of ODEs with the right-hand-side function `f()` that can be represented symbolically by the flow chart

```
f()
{
   pvode2bout(); /*copy data from PVODE state vector to local BOUT variables*/

   fields();     /*invert the vorticity to calculate electric potential */

   emfields();   /*invert Ajpar to calculate vector-potential Apar*/

   fucomm();     /*pass data between processors*/

   flocal();     /*calculate right-hand-side values for each grid point*/

   bout2pvode(); /*copy data from local BOUT variables to PVODE state vector*/
}
```

The function `fields()` solves the Poisson equation for the vorticity, Eq.(38), using the method of cyclic reduction [30]. It can be symbolically represented as follows

```
fields()
{

  for (jy=jstart; jy<=jend; jy++)    /*solve for one y-index at a time*/
  {
       rhok=fft(rho,-1);             /*do forward FFT for the right-hand side*/

       phik=linear_solve(rhok);      /*solve complex tridiagonal system for phi_k*/

       phi=fft(phik,1);              /*do backward fft for phi*/
```

```
  }

}
```

And similarly the function `emfields` solving the Helmholtz equation for $\hat{A}_{||}$, Eq.(6), can be schematically represented by

```
emfields()
{

  for (jy=jstart; jy<=jend; jy++)     /*solve for one y-index at a time*/
  {
       Ajpark=fft(Ni0*(Ajpar-Vi),-1); /*do forward FFT for the right-hand side*/

       Apark=linear_solve(Ajpark);    /*solve complex tridiagonal system for Apar_k*/

       Apar=fft(Apark,1);             /*do backward fft for Apar*/
  }

}
```

10 Pre-processing and post-processing

10.1 Grid generation

BOUT uses grids generated externally. Grids are generated from a poloidal flux function $\psi(R, Z)$ by constructing a set of lines ψ=const and orthogonal to them lines θ=const such that $\nabla\psi \cdot \nabla\theta$=0. This forms a projection of the actual grid to a constant-toroidal-angle plane. In the toroidal direction the grid is uniform in the toroidal angle ζ and the toroidal domain covers an integer fraction of 2π. Figure (6) shows the projection to the (R,Z) plane for three different configurations: cylindrical slab, circular toroidal geometry, and divertor tokamak geometry. The flux function $\psi(R, Z)$ for the cylindrical slab with uniform poloidal field B_0 is taken as $\psi = B_0 R^2$; for the shifted-circle

toroidal geometry the poloidal field is based on the standard analytic model [31]; for actual tokamak geometry the flux function is taken from experimental magnetic reconstruction, e.g., with codes EFIT [32] or TEQ [33]. A 3D view of grid lines for an actual tokamak geometry is shown in Fig.(3).

10.2 Pre-processing

Pre-processing consists of the calculation of quantities related to the magnetic geometry: metric coefficients, integrated shear etc. These grid data are saved in a grid file that is subsequently imported into BOUT.

10.3 Setting options and boundary conditions

In addition to the grid file, BOUT uses an input file in which various run options are set. The main options are: switches for individual terms in the equations that are turned on and off, physics options like electrostatic or electromagnetic model, zero or finite electron mass, toroidal grid size and others.

Periodic boundary conditions are used for the toroidal-angle direction z. On the radial boundaries one can set either Neumann or Dirichlet boundary conditions. On the poloidal boundaries one can set periodic boundary conditions (for closed flux surfaces), or Neumann or Dirichlet conditions, or material wall (sheath) conditions as described in [9,10].

10.4 Turbulent flux of particles and energy

One of primary quantities of interest from turbulence simulations is the flux of particles and energy induced by the turbulence. The radial outward particle flux is

$$\Gamma_x = \langle \tilde{n}\tilde{V}_x \rangle \tag{48}$$

Here the average can be over a time or space domain large compared to the characteristic scale of fluctuations. For tokamak simulations, due to the axial

symmetry, it is convenient to use an average over the toroidal coordinate z. The positive sign for the x direction is from the plasma to the wall, for either sign of the poloidal field.

The radial outward component of the fluctuating drift velocity is

$$\tilde{V}_x = \frac{\vec{V}_E \cdot \vec{\nabla} x}{|\vec{\nabla} x|}. \tag{49}$$

Using

$$\vec{V}_E = \frac{c}{B^2}[B \times \nabla\phi] = \frac{c}{B^2}[(\vec{\nabla} z \times \vec{\nabla} x) \times (\frac{\partial\phi}{\partial x}\vec{\nabla} x + \frac{\partial\phi}{\partial y}\vec{\nabla} y + \frac{\partial\phi}{\partial z}\vec{\nabla} z)], \tag{50}$$

one finds

$$\tilde{V}_x = -\frac{\sigma_\psi \nu R|B_\theta|}{B^2 h_\theta^2}\frac{\partial\phi}{\partial y} + \frac{1}{RB_\theta}\frac{\partial\phi}{\partial z}, \tag{51}$$

which can be used to calculate Γ_x. Similarly one calculates the turbulent radial heat flux. It splits into the convective and conductive parts

$$q_{e,i,x} = \langle\frac{3}{2}(n_0 + \tilde{n})(T_{e,i0} + \tilde{T_{e,i}})\tilde{V}_x\rangle = \frac{3}{2}T_{e,i0}\Gamma_x + \frac{3}{2}n_0\langle\tilde{T_{e,i}}\tilde{V}_x\rangle. \tag{52}$$

The radial turbulent fluxes of particles and energy associated with the magnetic perturbation are $\Gamma_x = \langle\Gamma_{||}\tilde{b}_x\rangle$ and $q_{e,i,x} = \langle q_{e,i,||}\tilde{b}_x\rangle$. Due to the smallness of the magnetic perturbation in the tokamak edge the flutter-induced flux is usually found in both experiments and modeling to be 1-2 orders of magnitude below the $E \times B$ flux.

11 Verification testing

11.1 Linear plasma instabilities

BOUT has been verified on a series of linear test problems reproducing several standard dispersion relations for basic linear plasma modes: interchange,

drift, shear-Alfven, acoustic, ideal and resistive ballooning, and conducting-wall modes. Some of these tests, presented earlier in [19], allowed comparison with exact analytic solutions. A series of more complex benchmarks for resistive-ballooning and conducting-wall (sheath-driven) modes was recently conducted with a linear edge-plasma code 2DX [20].

Here we describe in some detail the conducting-wall mode problem. Here the equations being solved are

$$(\frac{\partial}{\partial t} + i\omega_E)\nabla_\perp^2\phi = -D\nabla_{||}^2\phi; \tag{53}$$

$$(\frac{\partial}{\partial t} + i\omega_E)T_e = -\vec{V}_E \cdot \nabla T_{e0}; \tag{54}$$

and the linearized sheath boundary condition at the end plates takes the form

$$D\nabla_{||}\phi = \pm(\Lambda_1\phi - \Lambda_2 T), \tag{55}$$

where $D = \sigma_{||}T_0^2/N_0$, $\omega_E = -k_z\partial\Phi_0/\partial x$, $\Lambda_{1,2}$ are constants.

From Eqs. (54) one can derive the dispersion relation

$$\tilde{\omega}^2 \tan\frac{\eta}{2} = -i\frac{\eta}{2}\omega_s\left(\Lambda_1\tilde{\omega} - \Lambda_2\omega_{*eT}\right), \tag{56}$$

where $\tilde{\omega} = \omega - \omega_E$, $\eta = k_{||}L_{||}$, $k_{||} = (i\omega/\sigma_{||})^{1/2}k_z$, $\omega_s = 2/k_z^2L_{||}$, $\omega_{*eT} = k_z dT_0/dx$.

In the limit of small k_z, $\tan(\eta/2) \approx \eta/2$, and the dispersion relation for even (most unstable) modes becomes

$$\tilde{\omega}^2 + i\omega_s\left(\Lambda_1\tilde{\omega} - \Lambda_2\omega_{*eT}\right) = 0. \tag{57}$$

In the large k_z limit, $\tan(\eta/2) \approx i$, and the dispersion relation becomes

$$\tilde{\omega}^2 + (\eta/2)\omega_s\left(\Lambda_1\tilde{\omega} - \Lambda_2\omega_{*eT}\right) = 0. \tag{58}$$

Comparison of the results from BOUT and 2DX with the analytic solution is shown in Fig.(7) for the test case with $\phi_0 = 0$, Λ_1=0, and Λ_2=1.

11.2 Axisymmetric plasma transport

Axisymmetric benchmarks against the UEDGE code were conducted to verify the correctness of the treatment of the actual tokamak geometry. These test problems are also interesting since they are strongly nonlinear. Using for simplicity a single plasma temperature, one can write in the BOUT (x, y, z) coordinates the axisymmetric equations expressing conservation of plasma energy, mass, and the parallel momentum as

$$\frac{\partial}{\partial t}\left(3N_iT\right) = \frac{B_\theta}{h_\theta}\left[\frac{\partial}{\partial x}(R^2B_\theta h_\theta\ N_i\chi_\perp\frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}\left(\frac{B_\theta}{B^2h_\theta}\ \kappa_{||}\frac{\partial T}{\partial y}\right)\right], \tag{59}$$

$$\frac{\partial N_i}{\partial t} = -\frac{B_\theta}{h_\theta}\frac{\partial}{\partial y}\left(\frac{N_iV_{||i}}{B}\right) + \frac{B_\theta}{h_\theta}\frac{\partial}{\partial x}(R^2B_\theta h_\theta\ D_\perp\frac{\partial N_i}{\partial x}), \tag{60}$$

$$\frac{\partial V_{||i}}{\partial t} = -V_{||i}\frac{B_\theta}{h_\theta}\frac{\partial}{\partial y}(V_{||i}/B) - \frac{1}{N_i}\frac{B_\theta}{Bh_\theta}\frac{\partial P}{\partial y} + \frac{1}{N_i}\frac{B_\theta}{h_\theta}\frac{\partial}{\partial x}(R^2B_\theta h_\theta\ N_i\mu_\perp\frac{\partial V_{||i}}{\partial x}). \tag{61}$$

Exactly same equations with the same transport coefficients $\chi_\perp$, $D_\perp$, and $\mu_\perp$, in the same unbalanced double null divertor tokamak geometry were solved by BOUT and UEDGE. The results for steady-state distribution of plasma temperature are shown in Fig.(8). The level of agreement between the two independent codes verifies the correctness of the treatment of the metric coefficients. Also, since the BOUT solution in this case is carried out on several processors, this test verifies the correctness of parallel domain decomposition in BOUT.

11.3 Neutral fluid hydrodynamics

Implicit time integration is not typical for advection-dominated problems. In a sense implicit treatment contradicts the nature of advection since implicitness couples together the whole spatial domain while in advection information propagates only along the characteristics. To verify the performance of implicit time integration in BOUT for a nontrivial advection-dominated system, an ideal neutral fluid hydrodynamic problem was solved with BOUT and benchmarked against a well-known ICF hydro code LCPFCT [21]. In this test the system of equations describing 2D dynamics of the ideal fluid is solved:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0; \tag{62}$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial(\rho u^2 + p)}{\partial x} + \frac{\partial(\rho u v)}{\partial y} = 0; \tag{63}$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial(\rho u v)}{\partial x} + \frac{\partial(\rho v^2 + p)}{\partial y} = 0; \tag{64}$$

$$\frac{\partial E}{\partial t} + \frac{\partial(u(E+p))}{\partial x} + \frac{\partial(v(E+p))}{\partial y} = 0. \tag{65}$$

where ρ is the mass density, p is the pressure, E is the total energy per unit volume, u and v are the two components of the velocity.

The equation of state is that for the ideal gas:

$$E = \rho u^2 + \rho v^2 + \frac{p}{\gamma - 1}, \tag{66}$$

where γ is the adiabat index.

The initial condition is a Gaussian pulse of density at the center:

$$\rho(t = 0) = \rho_0(1 + \alpha \exp(-(x^2 + y^2)/\Delta^2)); \tag{67}$$

and the pressure is set according to the adiabatic equation of state, p/ρ^γ=const.

We solve this system of equations in a 2D square box with ideal reflecting-wall boundary conditions. In the subsequent time evolution shock waves emerge and are reflected off the walls and rejoin at the center, showing complex periodic patterns, Fig. (9). Comparing the time history of fluid fields between BOUT and LCPFCT, one can observe that they are in a good agreement, Fig. (9).

12 Application to tokamak edge turbulence

Starting from initial plasma profiles, BOUT can be used in full time-evolution mode, when the toroidally average plasma profiles are evolved. However, BOUT does not have all the physics that, aside from the turbulent transport, is in-

volved in forming the steady state plasma profiles, i.e., the atomic physics, neutral transport, impurity transport and radiation. Therefore, full time-evolution with BOUT would result in steady-state plasma profiles that, in general, would not match the experimental ones; and it would not be clear in what sense the code models the experiment.

To circumvent this problem, BOUT can be run in a mode with the toroidally average plasma profiles frozen. The latter can be interpreted as adding to the equations fake source terms that are functions of space but not of plasma variables, and the axisymmetric equilibrium profiles are maintained constant in time.

Here we illustrate the application of BOUT to the modeling of edge plasma in the Alcator C-Mod tokamak [34]. The physics model for this calculation is simplified from the full-blown BOUT: the temperature equations are not included and only main physics terms are kept. The computational grid is constructed from the actual magnetic geometry of the experiment; and the profiles of plasma density and temperature are set according to actually measured experimental edge profiles.

The run starts from a small, random, seed perturbation, and first the code goes through a linear growth phase, after which turbulence reaches saturation; see Fig. (10). In the saturated stage, the spatial structure of density fluctuations is similar to the experiment; see Fig. (11). Generally, in BOUT applications to tokamak edge plasmas, many characteristics of simulated turbulence are found to be in the ballpark of experimental data [35].

13 Summary

The BOUT code is one of fullest models of tokamak edge turbulence based on collisional plasma fluid equations. BOUT solves for time evolution of a set of plasma fluid variables: plasma density N_i, parallel ion velocity $V_{||i}$, electron temperature T_e, ion temperature T_i, electric potential ϕ, parallel current $j_{||}$, and parallel vector potential $A_{||}$, in real 3D divertor tokamak geometry.

BOUT has undergone verification testing on a range of test problems. Successful passing of these tests supplies strong evidence that BOUT correctly solves the dynamic equations, as long as the spatial resolution is sufficient. Applications of BOUT to tokamak edge plasmas generally demonstrate similarity to many characteristics of experimentally measured turbulence.

14 Appendix: Summary of BOUT equations

14.1 Electron parallel momentum

$$\frac{\partial V_{||e}}{\partial t} + (\vec{V}_E + V_{||e}\vec{b}_0)\cdot\nabla V_{||e} = -\frac{e}{m_e}E_{||} - \frac{1}{N_i m_e}(T_e\partial_{||}N_i + 1.71N_i\partial_{||}T_e) + 0.51\nu_{ei}(V_{||i} - V_{||e}) - \frac{1}{N_i m_e}\frac{2}{3}B^{3/2}\partial_{||}(B^{-3/2}(P_{||e} - P_{\perp e})) + \frac{S^m_{||e}}{N_i m_e} - \frac{S^p_e}{N_i} \quad (68)$$

14.2 Vorticity

$$\frac{\partial \varpi}{\partial t} + (\vec{V}_E + V_{||i}\vec{b}_0)\cdot\nabla\varpi = (2\omega_{ci})\vec{b}_0\times\vec{\kappa}\cdot(\nabla P + \frac{1}{6}\nabla(P_{||i} - P_{\perp i})) + N_i Z_i e\frac{4\pi V_A^2}{c^2}\nabla_{||}j_{||} + \mu_{ii}\nabla_\perp^2\varpi - (B\omega_{ci})\nabla\cdot(\frac{\vec{b}_0}{B}\times(S^m_e + S^m_i)) - (\frac{S^p_i}{N_i})\varpi - (\omega_{ci}B)\nabla(\frac{S^p_i}{N_i\omega_{ci}B})\cdot(N_i Z_i e\nabla\phi + \nabla P_i) - \frac{1}{2}[N_i q V_{Pi}\cdot\nabla(\nabla_\perp^2\phi) - M_i\omega_{ci}\vec{b}\times\nabla N_i\cdot\nabla\vec{V}_E^2] + \frac{1}{2}[\vec{V}_E\cdot\nabla(\nabla_\perp^2 P_i) - \nabla_\perp^2(\vec{V}_E\cdot\nabla P_i)] \quad (69)$$

14.3 Density

$$\frac{\partial N_i}{\partial t} + (\vec{V}_E + V_{||i}\vec{b}_0)\cdot\nabla N_i = (\frac{2c}{eB})\vec{b}_0\times\vec{\kappa}\cdot(\nabla P_e - N_i e\nabla\phi) + \nabla_{||}(\frac{j_{||}}{e}) - N_i\nabla_{||}V_{||i} + S^p_e \quad (70)$$

14.4 Ion temperature

$$\frac{\partial T_i}{\partial t} + (\vec{V}_E + V_{||i}\vec{b}_0)\cdot\nabla T_i = \frac{4}{3}(\frac{cT_i}{N_i eB})\vec{b}_0 \times \vec{\kappa}\cdot(\nabla P_e - N_i e\nabla\phi - \frac{5}{2}N_i\nabla T_i) + \frac{2}{3N_i}\nabla_{||}(\kappa^c_{||i}\partial_{||}T_i) + \frac{2S^E_i}{3N_i} - \frac{2T_i}{3N_i}(N_i\nabla_{||}V_{||i} - \frac{1}{e}\nabla_{||}j_{||}) + + \frac{2m_e}{m_i}\nu_{ei}(T_e - T_i) + + \frac{2}{3}(\frac{20}{3}\mu_{ii})\nabla_\perp{}^2 T_i - \nu_i T_i \quad (71)$$

14.5 Electron temperature

$$\frac{\partial T_e}{\partial t} + (\vec{V}_E + V_{||e}\vec{b}_0)\cdot\nabla T_e = \frac{4}{3}(\frac{cT_e}{N_i eB})\vec{b}_0 \times \vec{\kappa}\cdot(\nabla P_e - N_i e\nabla\phi + \frac{5}{2}N_i\nabla T_e) + \frac{2}{3N_i}\nabla_{||}(\kappa^c_{||e}\partial_{||}T_e) + \frac{2\eta_{||}}{3N_i}j^2_{||} - \frac{2T_e}{3}\nabla_{||}V_{||e} - \frac{2m_e}{m_i}\nu_{ei}(T_e - T_i) + + 0.71\frac{2T_e}{3N_i e}\nabla_{||}j_{||} + \frac{2S^E_e}{3N_i} - \nu_i T_e \quad (72)$$

14.6 Ion parallel momentum

$$\frac{\partial V_{||i}}{\partial t} + (\vec{V}_E + V_{||i}\vec{b}_0)\cdot\nabla V_{||i} = -\frac{1}{N_i M_i}\partial_{||}P - \frac{1}{N_i M_i}P_i\nabla\times(\vec{b}/\omega_{ci})\cdot\nabla V_{||i} - \frac{1}{N_i M_i}\frac{2}{3}B^{3/2}\partial_{||}(B^{-3/2}(P_{||i} - P_{\perp i})) + \frac{S^m_{||i}}{N_i M_i} - \frac{S^{cx}_i + S^p_i}{N_i}V_{||i} \quad (73)$$

14.7 Auxiliary relations and definitions

$$\vec{V}_E = c\vec{b}_0 \times \nabla_\perp\phi/B$$
$$\vec{V}_{Pi,e} = c\vec{b}_0 \times \nabla_\perp P_{i,e}/nqB$$
$$E_{||} = -\partial_{||}\phi - (1/c)\frac{\partial A_{||}}{\partial t}$$
$$\varpi = N_i q\nabla^2_\perp\phi + N_i q\nabla_\perp\phi\cdot\nabla_\perp\ln N_i + \nabla^2_\perp P_i$$
$$(P_{||} - P_\perp)_{i,e} = \eta\sigma_{i,e}((\vec{V}_E + V_{Pi,e})\cdot\vec{\kappa} - (2/\sqrt{B})\partial_{||}(\sqrt{B}V_{||i,e}))$$
$$\nabla^2_\perp A_{||} = -(4\pi/c)j_{||}$$
$$\tilde{\vec{B}} = \nabla A_{||}\times\vec{b}_0$$

$$
\begin{aligned}
\mu_{ii} &= \frac{3}{10}\nu_{ii}\rho_i^2 \\
\eta_0 &= 0.96 P_i \tau_i \\
\nabla_{||} F &= B\partial_{||}(F/B) \\
\partial_{||} &= \partial_{||}^0 + \tilde{\vec{b}} \cdot \nabla \\
\tilde{\vec{b}} &= \tilde{\vec{B}}/B \\
\partial_{||}^0 &= \vec{b}_0 \cdot \nabla \\
\vec{\kappa} &= \vec{b}_0 \cdot \nabla \vec{b}_0
\end{aligned}
\tag{74}
$$

where $\mu_{ii}, \mu_{||}, \chi_{||}^c, \nu_{ii}$ are the classical transport parameters; with the source terms $S_{||e}^m$, $S_{||i}^m$ for parallel momentum, S_e^p, S_i^p for particle density, S_i^E, S_e^E for the energy.

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Security, LLC, Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

[1] J.W. Connor and H.R. Wilson, Plasma Phys. Control. Fusion 42 (2000), R1

[2] A. Hasegawa and M. Wakatani Phys. Rev. Lett. 50, 682 (1983)

[3] R.D. Sydora *et al.*, Phys. Fluids 29, 2871 (1986).

[4] Carreras, B.A. *et al.*, Phys. Fluids 30, 1388 (1987).

[5] S.I. Braginskii, *Transport Properties in a Plasma*, in Review of Plasma Physics, M.A. Leontovish (ed.), Consultants Bureau, vol. 1, pp. 205-311 (1965).

[6] B. Scott, Plasma Phys. Control. Fusion 45 (2003) A385

[7] C.S. Chang, S. Ku, Contrib. Plasma Physics, 46, 496 (2006)

[8] X.Q. Xu *et al.*, Nucl. Fusion 47, 809 (2007)

[9] X. Xu, and R.H. Cohen, Contributions to Plasma Physics, 38, (1998) no. 1-2, 158.

[10] X.Q. Xu *et al*, Phys. Plasmas **7** (2000) 1951.

[11] X.Q. Xu *et al*, Commun. Comput. Phys. **5** no. 5, pp. 949-979 (2008).

[12] A.B. Mikhailovskii and V.S. Tsypin, Beitr. Plasmaphys. 24, 335 (1984).

[13] P.J. Catto and A.N. Simakov, Phys. Plasmas 11, 2326 (2004)

[14] A.N. Simakov and P.J. Catto, Phys. Plasmas 10, 4744 (2003)

[15] B. LaBombard *et al.*, Nucl. Fusion 44,1047 (2004).

[16] D.D. Schnack *et al.* Phys. Plasmas 13, 058103 (2006)

[17] P.C. Stangeby *The plasma boundary of magnetic fusion devices* Institute of Physics Publishing, Bristol and Philadelphia, 2000.

[18] W.D. D'haeseleer, W.N.G. Hitchon, J.D. Callen, and J.L. Shohet, *Flux Coordinates and Magnetic Field Structure* (Springer Verlag, Berlin, Heidelberg, New York, 1991.

[19] M.V. Umansky, R.H. Cohen, L.L. LoDestro, and X.Q. Xu Contr. Plas. Phys. 48 (1-3) 27-31 (2008).

[20] J.R. Myra and M.V. Umansky, Lodestar Report LRC-08-120, April,2008 http://www.lodestar.com/LRCreports/LinearAnalysisLRCReport.pdf

[21] Boris, J.P.; A.M. Landsberg; E.S. Oran; and J.H. Gardner. 1993. "LCPFCT - A Flux-Corrected Transport Algorithm for Solving Generalized Continuity Equations." NRL Memorandum Report 93-7192.

[22] A.M. Dimits, *Phys. Rev. E*, 48, p. 4070, 1993.

[23] Beer, M.A. *et al* Phys. Plasmas v. 2 (7) pp. 2687-2700 (1995)

[24] Scott, B. Phys. Plasmas v. 8 (2) pp. 447-458 (2001)

[25] Schiesser, W.E. *The Numerical Method of Lines Integration of Partial Differential Equations* Academic Press, San Diego (1991).

[26] Byrne, G.D., and A. C. Hindmarsh, *User Documentation for PVODE, An ODE Solver for Parallel Computers* Lawrence Livermore National Laboratory Technical Report UCRL-ID-130884, May 1998

[27] Van Leer, B. J. Comp. Phys., 32, 101-136 (1979).

[28] C.-W. Shu, International Journal of Computational Fluid Dynamics, v17 (2003), pp.107-118.

[29] T.D. Rognlien, X.Q. Xu, and A.C. Hindmarsh, Journal of Computational Physics, Vol. 175, 249-268 (2002)

[30] R.W. Hockney, "The potential calculation and some applications", in: B. Adler, S. Fernback, M. Rotenberg (Eds.), Methods of Computational Physics, Vol. 9, Academic Press, New York and London, 1969.

[31] J.P. Freidberg, "Ideal Magnetohydrodynamics", Plenum Press (1987).

[32] L.L.Lao et al., Nucl. Fusion **25** 1611 (1985).

[33] J.A. Crotinger et al., "Corsica:A Comprehensive Simulation of Toroidal Magnetic-Fusion Devices" (UCRL-ID-126284) (1997).

[34] Hutchinson *et al.*, Phys. Plasmas 1, 1511 (1994)

[35] R.H. Cohen *et al.*, Nuclear Fusion 47 (2007) 612-625.

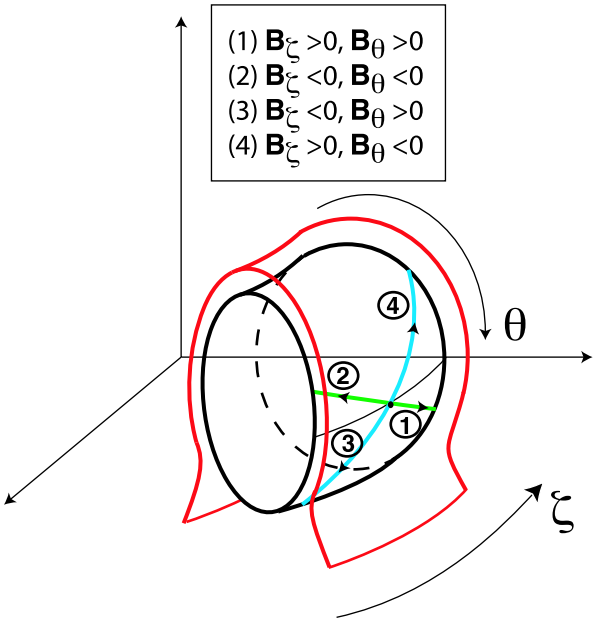

Fig. 1. The sign convention for the toroidal field. For the poloidal field B_θ the positive direction is from inner divertor to outer divertor. Positive toroidal field B_ζ is in the ζ direction, i.e. counter-clock-wise (looking from the top). Negative B_ζ, which is considered the "normal" case, corresponds to downward ion ∇B drift.

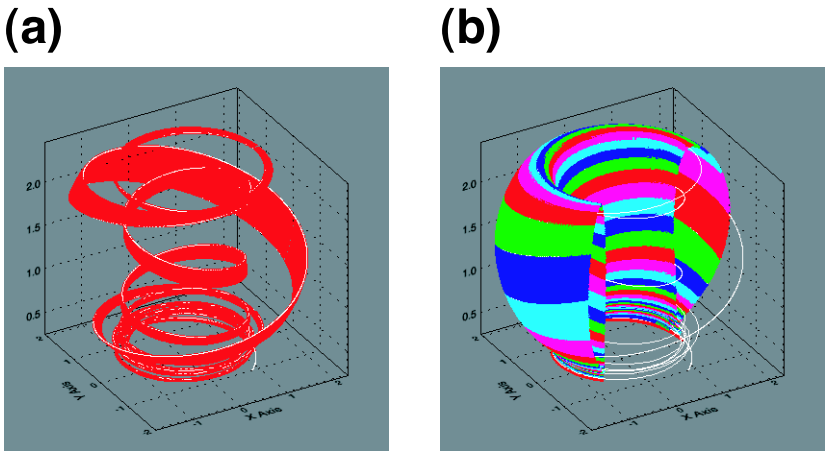

Fig. 2. In the physical coordinates the computational domain corresponds to a bundle of magnetic lines winding around the tokamak (a). The range of the toroidal angle in the computational domain is an integer fraction of full circle, $2\pi/\mathrm{n}$, so n such domains cover the whole torus (b). In the shown case (based on actual tokamak magnetic field) n=5.

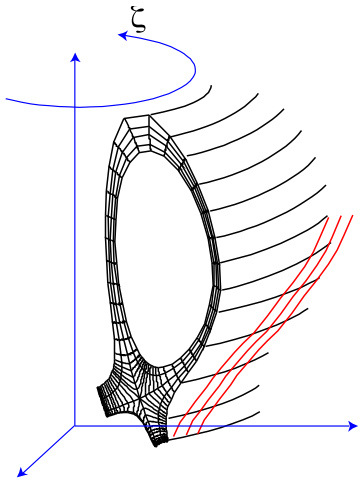

Fig. 3. In the field aligned coordinates x is the radial-like coordianate, and x=const corresponds to a fixed flux surface; y is the poloidal-like coordinate, and y=const corresponds to the lines going in the toroidal direction; z is the toroidal-like coordinate labeling magnetic lines within a flux surface, and z=const, x=const corresponds to an individual magnetic line (shown in red).

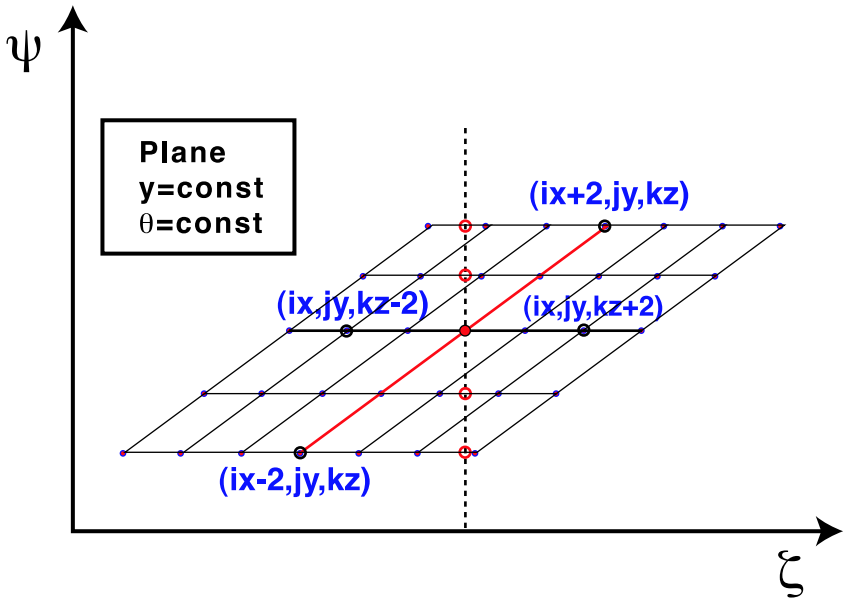

Fig. 4. The skewness of the grid is caused by shearing of the magnetic field since the grid is field aligned. Evaluating partial derivatives $\partial/\partial x$ by direct finite-difference (using grid points along the red line) involves physically separated locations and is highly inaccuarate for strong shear. Therefore the derivatives $\partial/\partial x$ are calculated by combining $\partial/\partial z$ (using grid points along the thick black line) and derivatives $\partial/\partial\psi$ using projected locations shown by red circles.

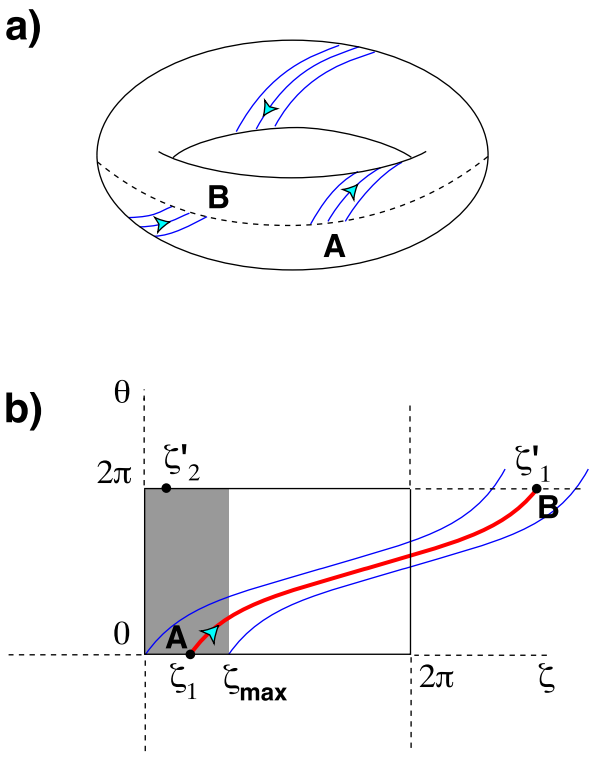

Fig. 5. The magnetic line winding around the torus does not close on itself unless the safety factor q is rational. To account for that one uses a twist-shift boundary condition where a magnetic line is closed on a different line accounting for the the safety factor q and the toroidal size of the computational domain.

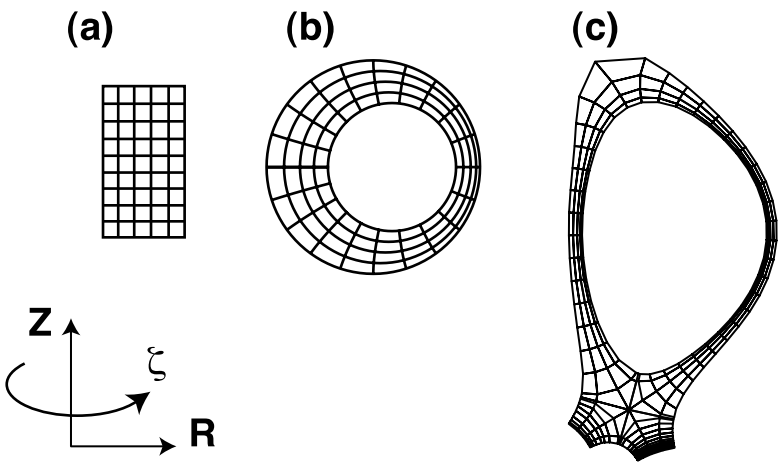

Fig. 6. Grids are generated from a given flux function $\psi(r, z)$ by constructing a set of lines ψ=const and orthogonal to them lines θ=const such that $\nabla\psi \cdot \nabla\theta$=0. This forms a projection of the actual grid to a constant toroidal angle plane. In the toroidal dimension the grid is uniform in the toroidal angle ζ and the toroidal domain covers an integer fraction of the 2π.

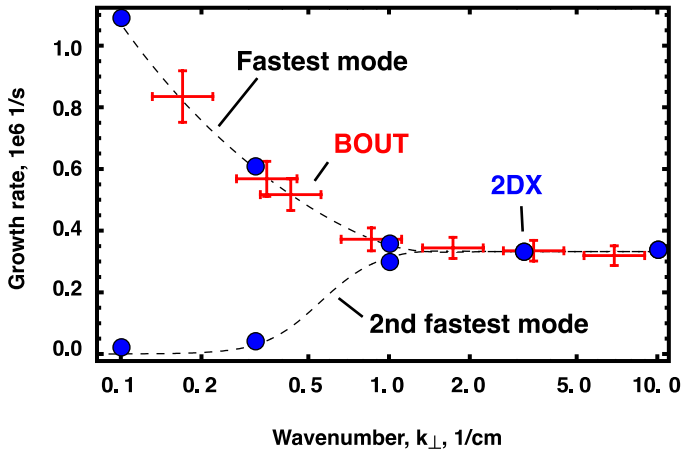

Fig. 7. Test of sheath boundary conditions for the conducting-wall mode problem. Growth rates vs. k_z for the fastest and second fastest growing mode are shown. Large dots are 2DX code results, dashed lines are the analytical solution, and crosses indicating error bars are the BOUT results for the fastest growing mode.

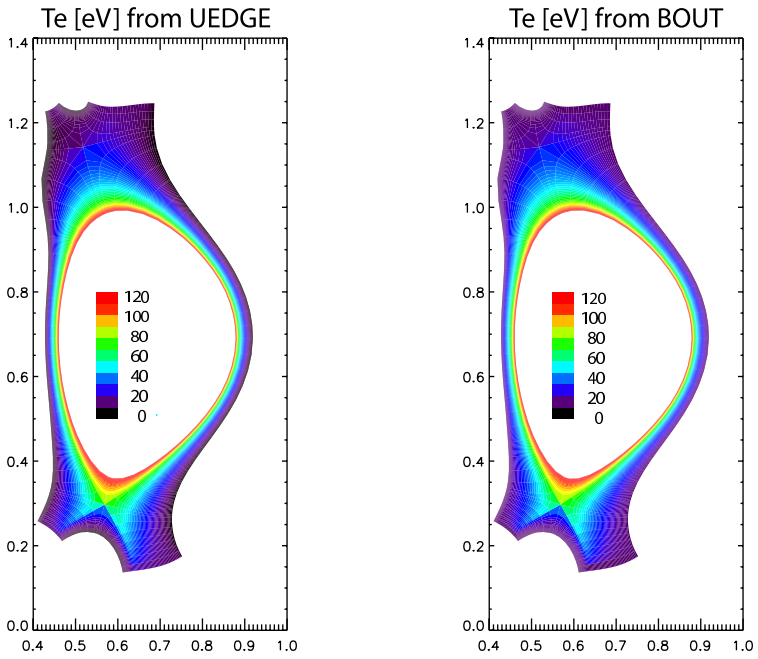

Fig. 8. Steady state distribution of plasma temperature in the axisymmetric benchmark test.

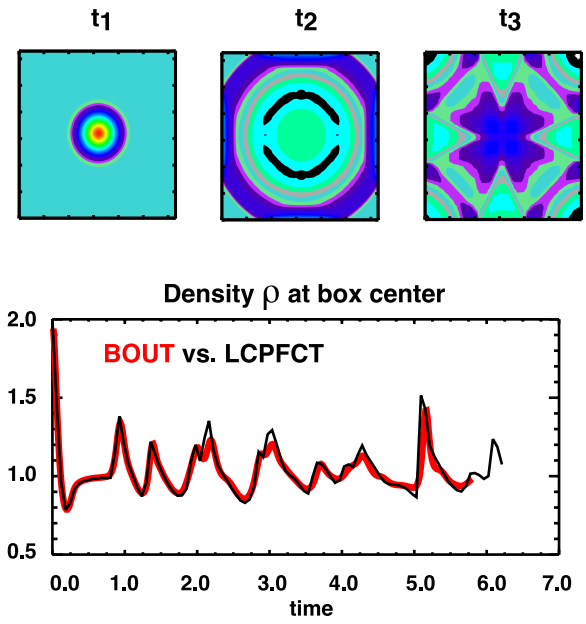

Fig. 9. Snapshots of fluid density at succesive times and time history of density at the central point for BOUT and LCPFCT.

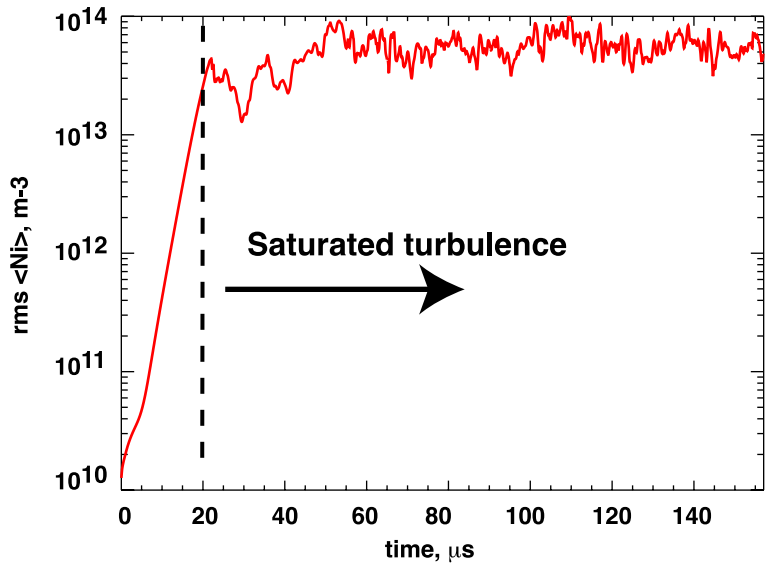

Fig. 10. Growth of RMS amplitude of turbulent fluctuations of plasma density in the tokamak edge plasma simulation. First fluctuations go through the phase of linear growth, then, at about 20 μs the turbulence reaches saturation.

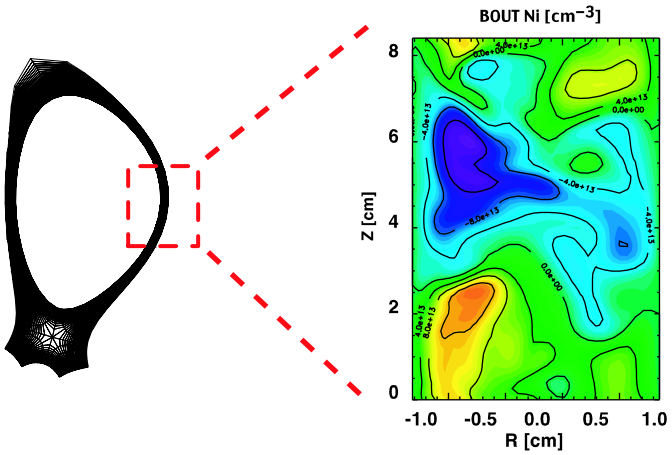

Fig. 11. Snapshot of plasma density perturbation at the outer midplane of the tokamak in the saturated turbulence phase.