Title:	Benchmarks for basic turbulence cases - updated (Report on Test Cases and Proxy-App)
Authors:	Christopher Ridgers, Mike Kryjak, University of York
Date:	31st January 2024
Report code:	2067270-TN-06-01

1 BOUT++, Hermes-3 and the benchmark cases

In this section we will summarise the equations to be solved. Full details of the first system, denoted 'Blob2D', . The second is the 'Hasegawa-Wakatani' system described in Ref. [3, 4]. We will also summarise the methodology by which we will produce the benchmark solutions to these equations. In particular we will outline the code used, i.e. codes built on the BOUT++ framework , most notably the Hermes-3 fluid edge modelling code , as well as the setup of the benchmark problems. Hermes-3[12] is a multi-dimensional and multi-species code built on top of the BOUT++ framework [11]. In 1D, it allows fast simulations of SOL dynamics and detachment physics, extending the capabilities of the 1D BOUT++ code SD1D [13]. In 2D, its capabilities are similar to the code UEDGE [14] but with a more advanced fluid neutral model, while in 3D it is similar to the BOUT++ code STORM [15] but with several improvements: a finite volume instead of a finite difference scheme to improve conservation, the ability to solve the ion energy equation and an advanced fluid neutral model. In this report, Hermes-3 is used to simulate the first case, the toy 2D turbulence system 'Blob2D' as found in Ref. [1] and described extensively in the literature [2]. The second case is a 3D implementation of the Hasegawa-Wakatani system. Due to the simplicity of the equations and boundary conditions, an existing BOUT++ implementation is used and solved using the Hermes-3 default solver to ensure that the performance is equivalent. In the final section, the more complex LAPD system is given a brief description. LAPD simulations are subject to future work.

2 Blob2D

2.1 Equations

This system consists of the following equations.

$$\frac{\partial n_e}{\partial t} = -\nabla\cdot(n_e\mathbf{v}_{E\times B}) + \nabla\cdot\left(\frac{1}{e}\mathbf{j}_{sh}\right) \tag{1}$$

$$\frac{\partial \omega}{\partial t} = -\nabla\cdot(\omega\mathbf{v}_{E\times B}) + \nabla\left(p_e\nabla\times\frac{\mathbf{b}}{B}\right) + \nabla\cdot\mathbf{j}_{SH} \tag{2}$$

$$p_e = en_eT_e \quad \omega = \nabla\cdot\left(\frac{1}{B^2}\nabla_\perp\phi\right) \quad \nabla\cdot\mathbf{j}_{SH} = \frac{n_e\phi}{L_\parallel} \tag{3}$$

These equations describe the dynamics of a localised 'blob' in the 2D drift plane. We simplify to a local slab geometry with effective radial coordinate x, effective poloidal coordinate y and effective coordinate z following the magnetic field. For a full description of the derivation of these equations see the review in reference [2]. n_e, T_e and p_e are the electron number density, temperature and pressure respectively. ω is the vorticity and $\mathbf{v}_{ExB}$ is the $\mathbf{E}\times\mathbf{B}$ drift velocity. The electrostatic

potential ϕ is determined by the sheath current $\mathbf{j}_{SH}$, i.e. the system is sheath-limited. The magnetic field is $\mathbf{B} = B\mathbf{b}$ and the $L_{\parallel}$ is the connection length.

2.2 Solving in Hermes-3

The Blob2D problem is provided as an example case in the BOUT++ code Hermes-3 [1]. Hermes-3 simulation results can be read using the Python package xHermes [21] which automatically handles the normalisation of state variables and dimensions, providing the user with data in SI units.

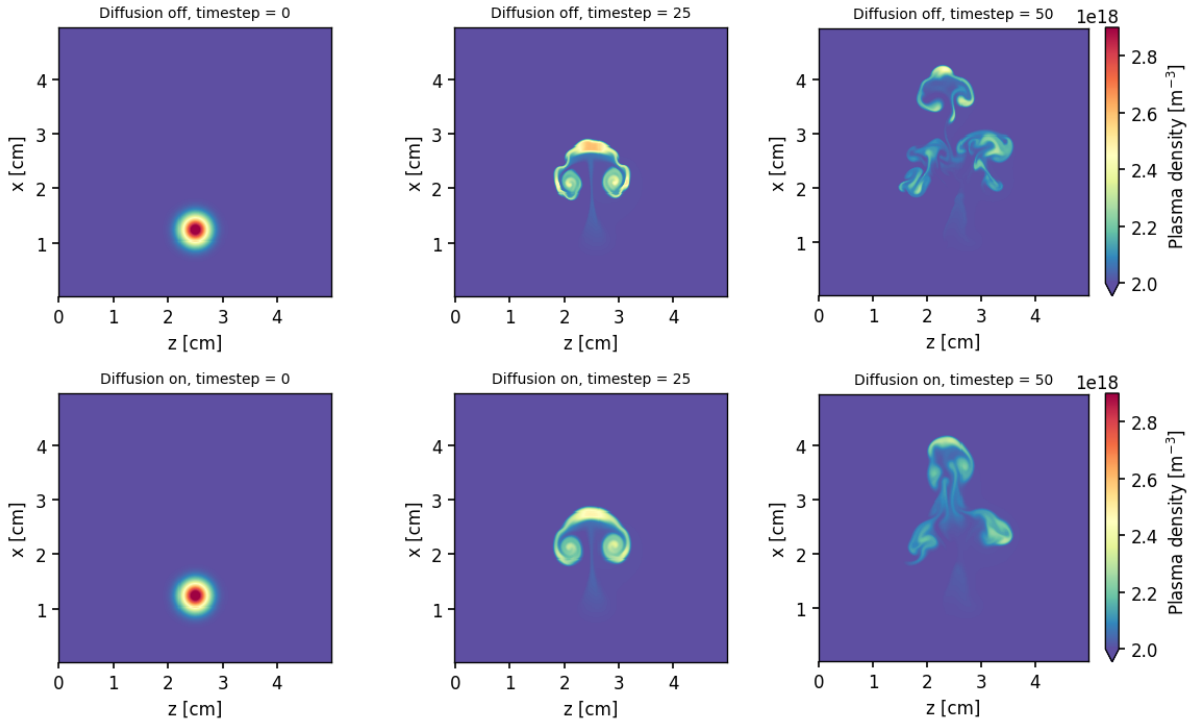

Figure 1: Hermes-3 simulation of the blob problem, i.e. equations 1-3. Density profiles at the first, middle and final timestep with and without additional dissipation.

Hermes-3 simulations of the blob problem were run with the following parameters. The electron density was normalised to $n_0 = 2 \times 10^{18}\,\mathrm{m}^{-3}$ and the electron temperature was set to 5 eV.The domain was set to be $x_0 \times y_0 = 0.05\,\mathrm{m} \times 0.05\,\mathrm{m}$ and discretised using 260×256 grid cells. The magnetic field was set to 0.35 T, the major radius to $R = 1.5\,\mathrm{m}$ (gives the curvature as $1/R^2$) and the connection length to 10 m. A gaussian blob was then initialised with height $0.5n_0$ and width $0.05x_0$. Figure 1 shows a hermes-3 simulation results for the evolution of the blob (equations 1-3). The simulation was run for where $2500/\omega_{ci}$ in steps of $50/\omega_{ci}$.

2.3 Artificial dissipation

The fine scale features present in the simulation present a problem for a spectral code. Equations 1-3 do not include explicit dissipation and so finer and finer features will continue to evolve. We have therefore added an explicit diffusion term to equation 1.

$$\frac{\partial n_e}{\partial t} = -\nabla \cdot (n_e \mathbf{v}_{E\times B}) + \nabla \cdot \left(\frac{1}{e}\mathbf{j}_{sh}\right) + D\nabla^2 n_e \tag{4}$$

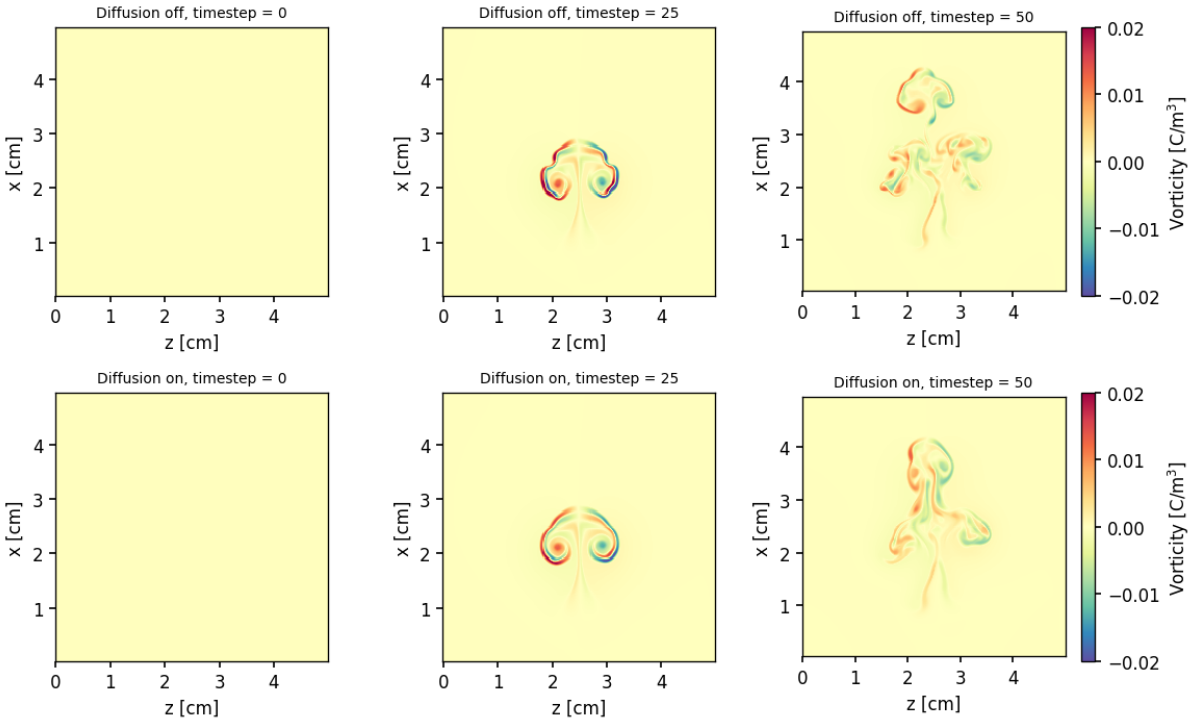

Figure 2: Hermes-3 simulation of the blob problem. Vorticity profiles at the first, middle and final timestep with and without additional dissipation.

The constant diffusion coefficient was set to $3 \times 10^{-4}\,\mathrm{m^2 s^{-1}}$. The bottom rows of figures 1 and 2 show the result of including this diffusion term. As expected, the development of fine scale features is prevented.

3 Hasegawa-Wakatani 3D

Due to the simplicity of the equations and boundary conditions, the Hasegawa-Wakatani problem wouldn't see significant benefit from being implemented into Hermes-3 apart from the finite volume operators. Instead, an existing finite difference BOUT++ implementation [22] is used and solved using the Hermes-3 default solver CVODE to ensure that the performance is as close as possible.

3.1 Equations

The 3D Hasegawa-Wakatani equation system solves for density n and vorticity $\nabla^2\phi$:

$$\frac{\partial n}{\partial t} = -\kappa\frac{\partial \phi}{\partial y} - \nabla\cdot\{\mathbf{b}\alpha(\partial_\parallel\phi - \partial_\parallel n)\} + [n,\phi] + D\nabla^2 n \tag{5}$$

$$\frac{\partial}{\partial t}\nabla^2\phi = -\nabla\cdot\{\mathbf{b}\alpha(\partial_\parallel\phi - \partial_\parallel n)\} + [\nabla^2\phi,\phi] + \mu\nabla^2(\nabla^2\phi) \tag{6}$$

Both the number density n and potential ϕ are fluctuations about a background. A slab geometry is used so that $\partial_\parallel$ is parallel to the direction of the magnetic field. The Poisson brackets $[A, B]$ are defined in consistency with the BOUT++ coordinate convention, i.e. $[A, B] = (\partial A/\partial x)(\partial B/\partial z) -$

$(\partial A/\partial z)(\partial B/\partial x)$ (see another BOUT++ implementation in [16]) but note that the code implements this in the equivalent form $-[B, A]$. Both equations feature diffusion terms to aid numerical stability with user-set particle diffusivity D and viscosity μ.

The dimensionless adiabaticity parameter α determines the strength of coupling between n and ϕ as well as the degree to which electrons can move rapidly and establish a Boltzmann response [4]. Increasing α makes the solution tend more towards zonal flows. These are coherent spatial structures which can arise in a turbulent system where the flow organises itself in a way to make it more efficient to absorb available system free energy, with many examples in nature such as the "great red spot" in the Jovian atmosphere. In tokamaks, these structures facilitate the L to H mode transition and suppress cross-field transport [17].

$$\alpha = \frac{T_0}{n_0 e \eta \omega_{ci}} \tag{7}$$

Where T_0 is the background electron temperature in eV, n_0 is the background electron density in m^{-3}, e the electron charge, η the electron resistivity and ω_{ci} the ion cyclotron frequency $\omega_{ci} = \frac{eB}{m_i}$. The electron resistivity can be estimated using the Spitzer formula [18] $\eta = 5.2 \times 10^{-5} Z ln\Lambda / T_0^{3/2}$ where the Coulomb logarithm $ln\Lambda$ can be calculated using the NRL formulary ([19], p.34).

The formulation for α equation is distinct from the one used in the 2D HW system which depends on an assumption of a parallel wavenumber k due to the domain not extending in the parallel direction:

$$\alpha = \frac{k^2 T_0}{n_0 e \eta \omega_{ci}} \tag{8}$$

In order to compare the 3D solution to 2D solutions in literature, a "2D-equivalent α" or α_{2D} can be computed by substituting the parallel wavenumber for the parallel domain length in metres:

$$\alpha = \frac{L_{par}^2 T_0}{n_0 e \eta \omega_{ci}} \tag{9}$$

The normalised parameter κ controls the radial density gradient scale length, which can be estimated from the scrape-off layer density width λ_n:

$$\kappa = \frac{\rho_{s0}}{n_0} \frac{\partial N}{\partial x} = \frac{\rho_{s0}}{\lambda_n} \tag{10}$$

Note that κ is a requirement of the Hasegawa-Wakatani system and is present in the BOUT++ HW3D example code as well as other 3D implementations [5].

The physical quantities are normalised as follows: $\hat{\phi} = \phi/T_e$, $\hat{n} = n/n_0$, $\hat{t} = \omega_{ci} t$ and $\hat{l} = l/\rho_{s0}$ where l is either x, y or z. ω_{ci} is the ion gyrofrequency and $\rho_{s0} = \sqrt{m_i T_e}/eB$ is the hybrid Larmor radius. Normalised vorticity is obtained by multiplying it by en_0 since it represents charge density. For more details on this system see references [3] and [4] as well as reference [16] for a HW2D implementation in BOUT++.

3.2 Normalisation

All of the state variables and dimensions will be in their normalised form and must be converted into SI units. The normalisation parameters are provided in the results file with the following variables:

$T_0 = Tnorm$, $n_0 = Nnorm$, $\rho_{s0} = rho\_s0$, $\omega_{ci} = Omega\_ci$. The simulations can be read in using the package xBOUT [23]. There are notebook examples of xBOUT post-processing workflows available including a Hasegawa-Wakatani case [24].

3.3 Initial conditions

The density equation is initialised to 0, while the vorticity equation is initialised with a fluctuation field given by:

$$vort_0 = 0.1(mixmode(2\pi x) * mixmode(z - y)) \tag{11}$$

Where x is the radial position between 0 and 1 while y and z are the parallel and binormal positions between 0 and 2π respectively. The function is a standard BOUT++ feature, see documentation in [25].

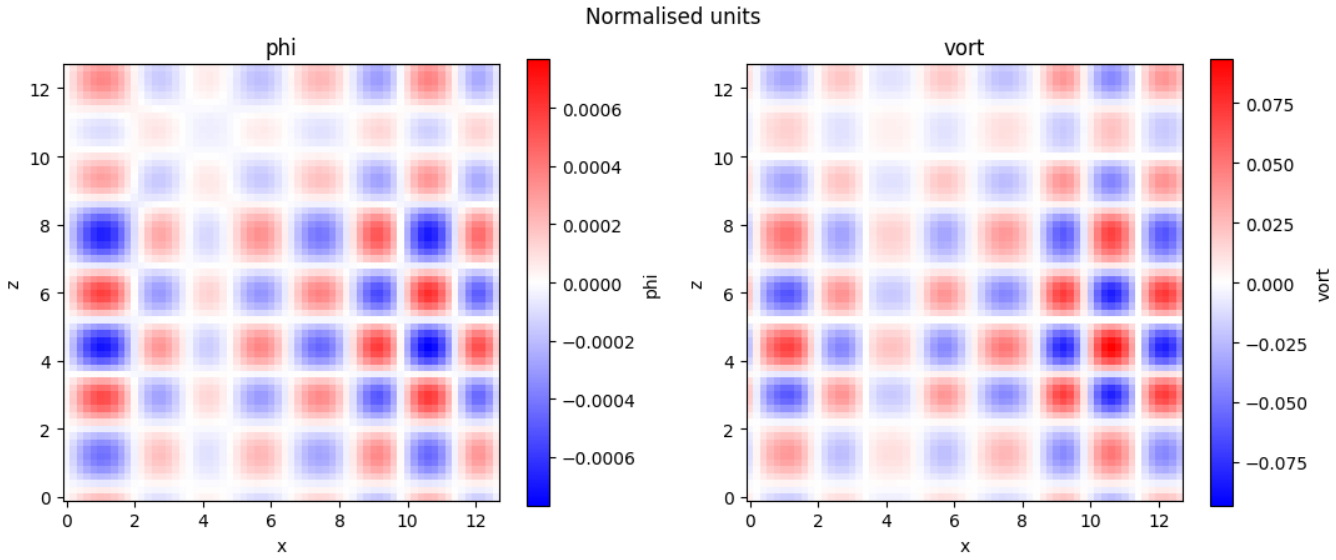

Figure 3: Initial phi and vorticity fields in normalised units for the baseline test case (excl. guard cells).

3.4 Boundary conditions

The y and z directions feature periodic boundary conditions, while the in the x direction the boundary conditions for density, potential and vorticity are set to a dirichlet (constant value) of zero. It is possible to set the x direction boundary periodic as well, but it requires a special treatment for the Laplacian inversion [26].

3.5 Domain and simulation time

The domain was discretised with 64 domain cells plus 4 guard cells in the x direction, making $nx \times ny \times nz = 68 \times 64 \times 64$. Note that if the z and y directions were not periodic, they would have the same ny and nz value: Following BOUT++ convention, the definition of nx includes guard (ghost) cells while ny and nz do not [20] due to historical reasons. Note that any non-periodic direction will always include guard cells, and that while there are two guard cells on each end of

the domain, the outer cell is used only for higher order boundary conditions not present in these models. The grid widths set to $dx \times dy \times dz = 0.2 \times 120.9 \times 0.2$, which makes the domain size 18.2 x 1000 x 18.2mm. The simulation was run for 500 output timesteps where each timestep represented $1 \times 10^{-6}\Omega_{ci}$ normalised time units, or 1μs of simulation time.

3.6 Input parameters

Case	T_0 [eV]	n_0 [m^{-3}]	B [T]	Z [-]	λ_n [m]	κ [-]	α_{2D} [-]	α [-]
1	40	1.59×10^{19}	0.5	1	0.00646	0.2	1×10^{-1}	2.54×10^{-1}
2	40	1.45×10^{18}	0.5	1	0.00646	0.2	$1 \times 10^{+0}$	2.54×10^{0}
3	40	1.33×10^{17}	0.5	1	0.00646	0.2	$1 \times 10^{+1}$	2.54×10^{1}

Table 1: Input parameters for the three HW3D cases.

The model inputs are shown in 1. All cases were ran with density and vorticity diffusivities set to 1×10^{-4}. The parameters in Case 1 were chosen to approximate typical conditions in the MAST-U scrape-off layer, with λ_n reduced to reach a κ value of 0.2 for comparison with other studies in literature [16]. Cases 2 and 3 had their α parameter increased to move the solution into more zonal driven regimes (see [6] for an in-depth study of the influence of α and κ parameters on the regimes).

3.7 Results

3.7.1 Performance

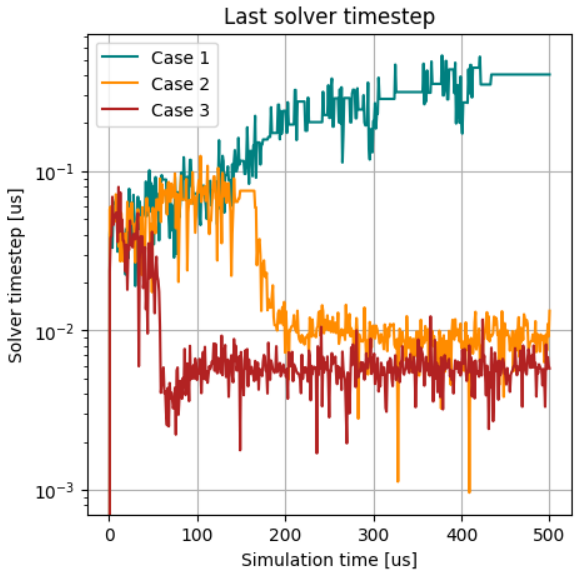

Figure 4: Last solver timestep.

The simulations were ran on the Viking2 cluster at the University of York using 32 cores on one node with a runtime of 57s, 4m 25s and 10m 41s for cases 1, 2 and 3 respectively using the CVODE solver with default settings. Figure 4 shows the final solver timesteps for each of the output timesteps for all cases.

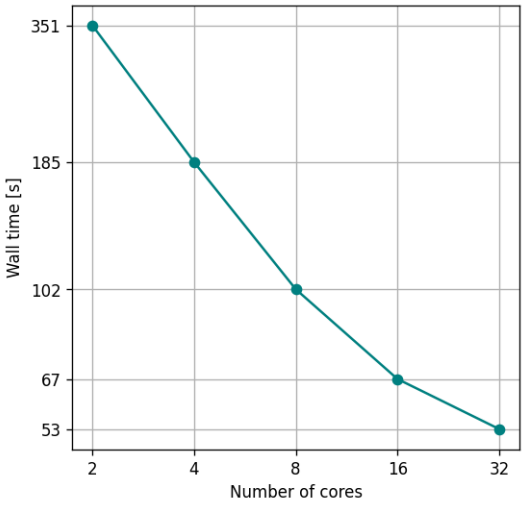

Figure 5: Simulation wall time for case 1 on different core counts on CSD3.

An additional performance benchmark was undertaken on the CSD3 cluster where case 2 was ran on 2, 4, 8, 16 and 32 cores on a single node of the cclake partition. The results are shown in fig. 5.

3.7.2 System evolution

Contours of density fluctuations are shown in fig. 6. As the alpha parameter increases, the turbulence becomes more zonal, leading to finer scale structures. The 3D implementation of Hasegawa-Wakatani will always tend towards a zonal solution. Since the damping depends on the parallel potential gradient, extending the model to explicitly simulate the parallel/poloidal direction allows a scenario where there a flux surface has a high potential but one without a parallel gradient, which prevents damping. This undamped mode leads to the growth of zonal flows in contrast to the classical Hasegawa-Wakatani where an assumption is made on the parallel dynamics through a constant wavenumber, but in a similar way to that in modified Hasegawa-Wakatani [16] where the zonal component is taken out of the damping term explicitly, allowing zonal flows to grow.

Higher values of α increase total system damping, which results in a greater prominence of the undamped zonal modes. This progression is clearly visible in fig. 6 where Case 1 features large blobs and less defined zonal structures while Case 3 features stronger zonal structures which break up the eddies into smaller ones.

To capture the evolution of the turbulence in a more quantitative way, the total system energy and enstrophy were calculated according to the methodology in [7] but in SI units, resulting in equations 12, 13 and 14. Note that n is the density fluctuation about a background, n_0 the constant background, and T_0 is in eV.

$$E_K = \int \frac{1}{2}\left[\frac{E \times B}{B^2}\right]^2 n_0 m_i \, dv \tag{12}$$

$$E_P = \int \frac{3}{2}\frac{n^2}{n_0} T_0 e \, dv \tag{13}$$

$$W = \int \frac{1}{2}(ne - \nabla^2\phi)\ dv \tag{14}$$

As shown in fig. 9, each case has been run for long enough to get close to steady state. Increased values of α_{2D} result in lower energy and enstrophy as the damping increases. The change in the kinetic energy profile in Case 2 at around $200\mu s$ corresponds to a transition from a more turbulent to a more zonal regime.

Figure 10 shows the power spectrum of the density field fluctuations in the three simulations. There is a clear difference between the more turbulent case 1 which features larger blobs and cases 2 and 3 where the zonal flows lead to many more smaller scale features as the larger eddies are broken up. The RHS cutoff of the plot represents the smallest length scale fluctuation resolved by the grid.

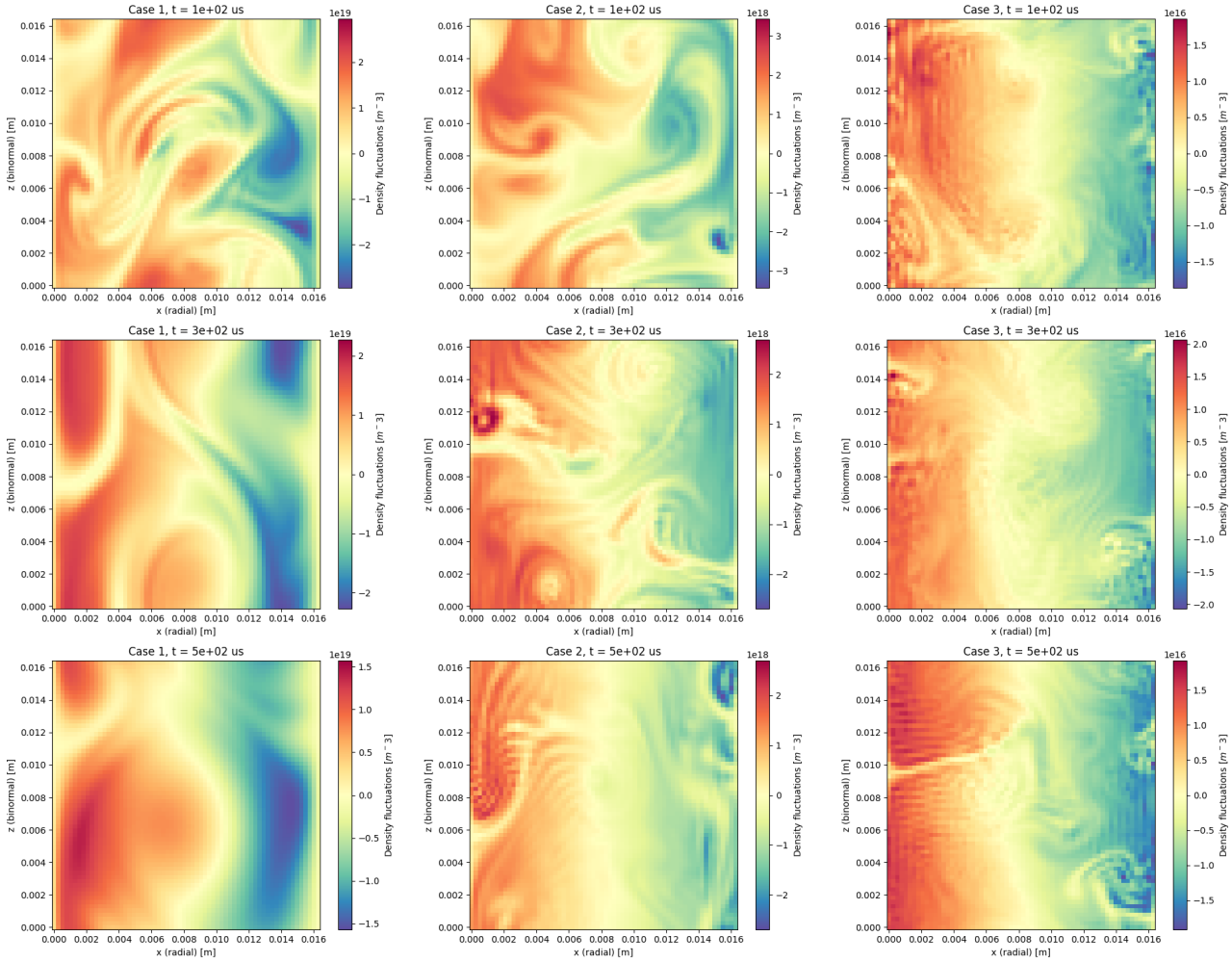

Figure 6: Density fluctuation contours for the three cases across three time slices halfway along the Y direction in the XZ plane.

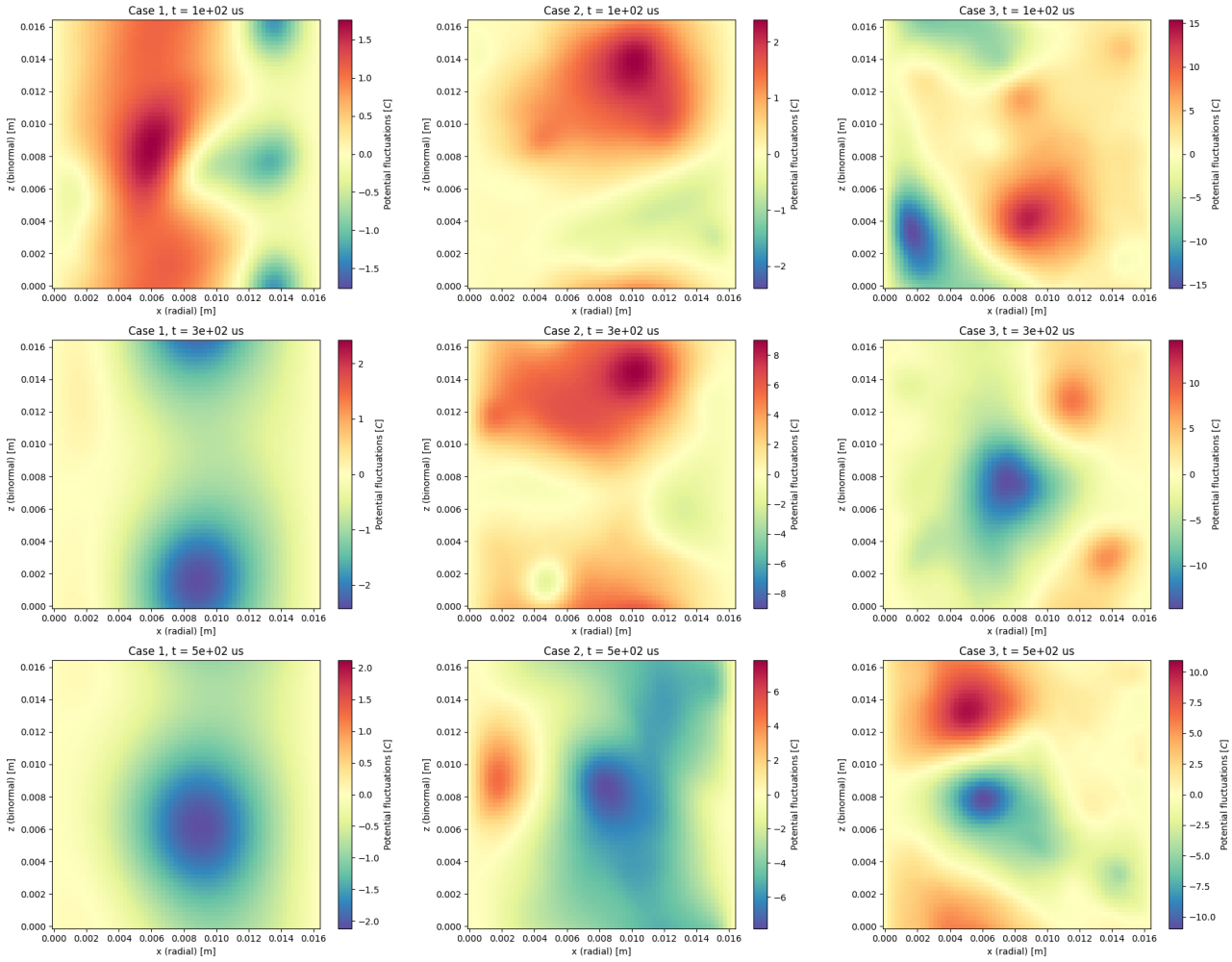

Figure 7: Potential fluctuation contours for the three cases across three time slices halfway along the Y direction in the XZ plane.

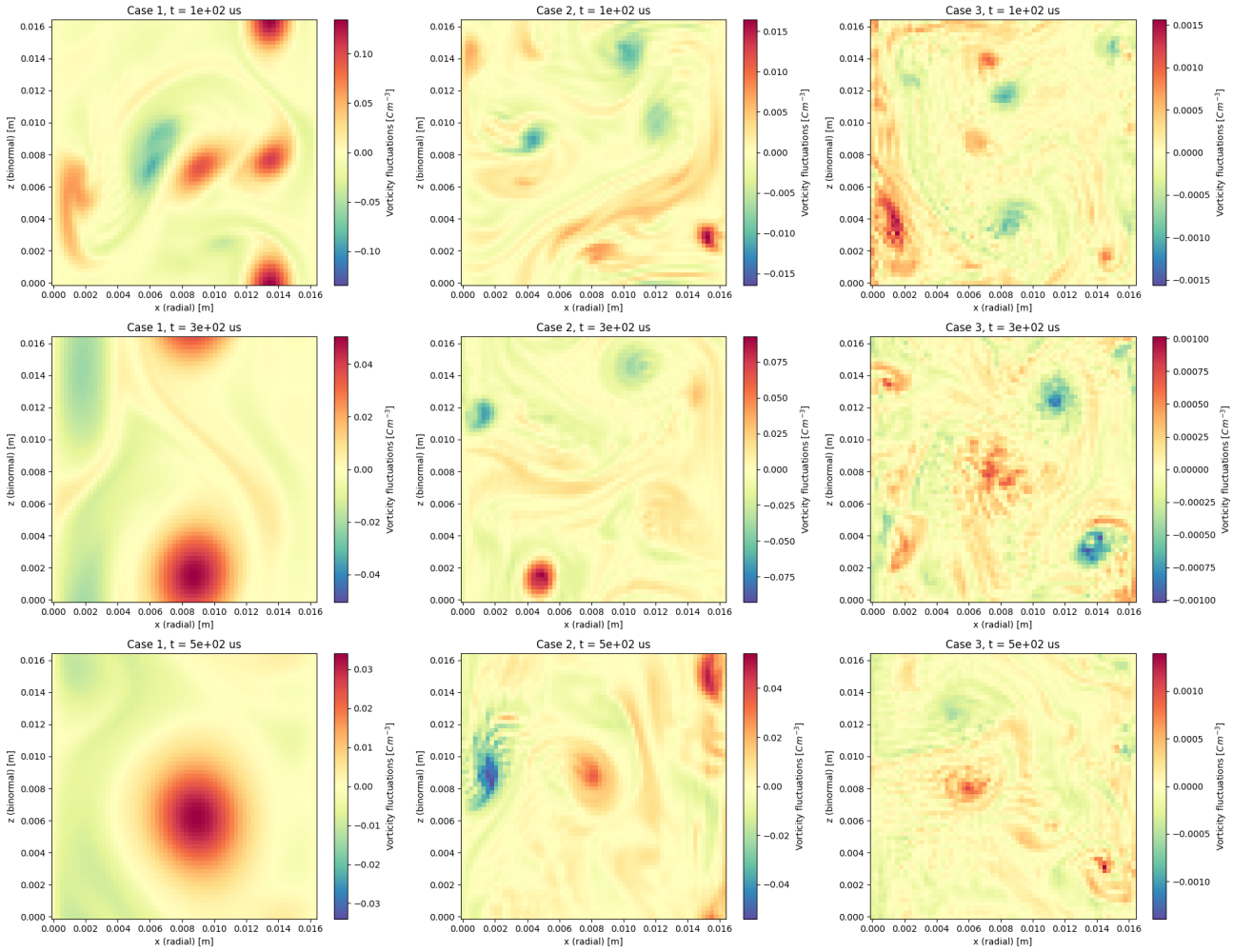

Figure 8: Vorticity fluctuation contours for the three cases across three time slices halfway along the Y direction in the XZ plane.

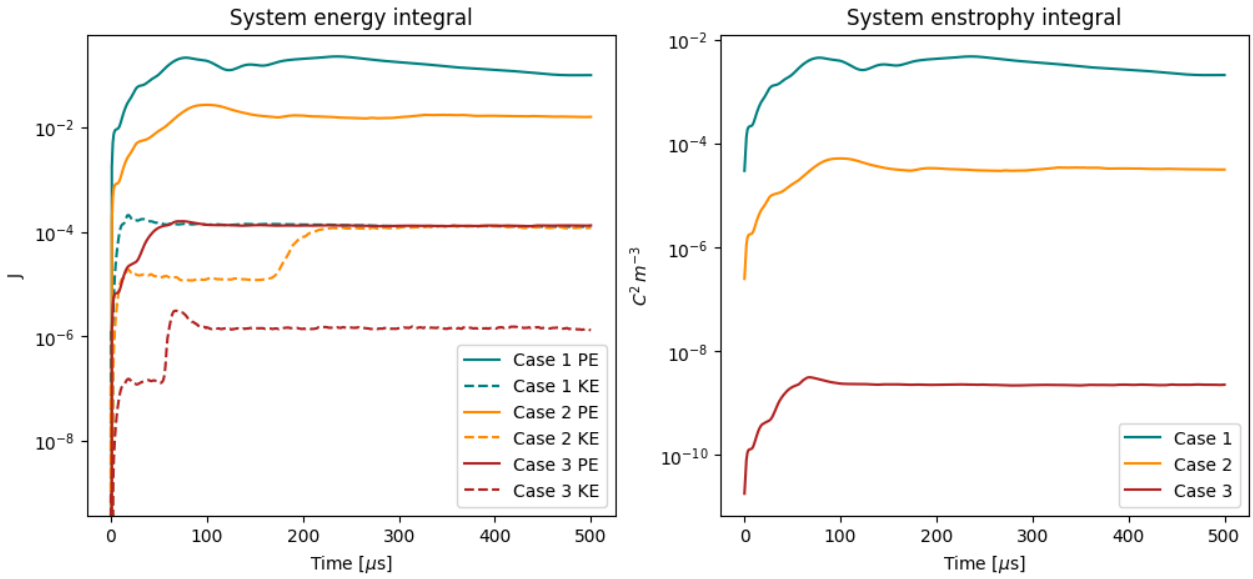

Figure 9: Energy and enstrophy system integrals over time for each simulation.

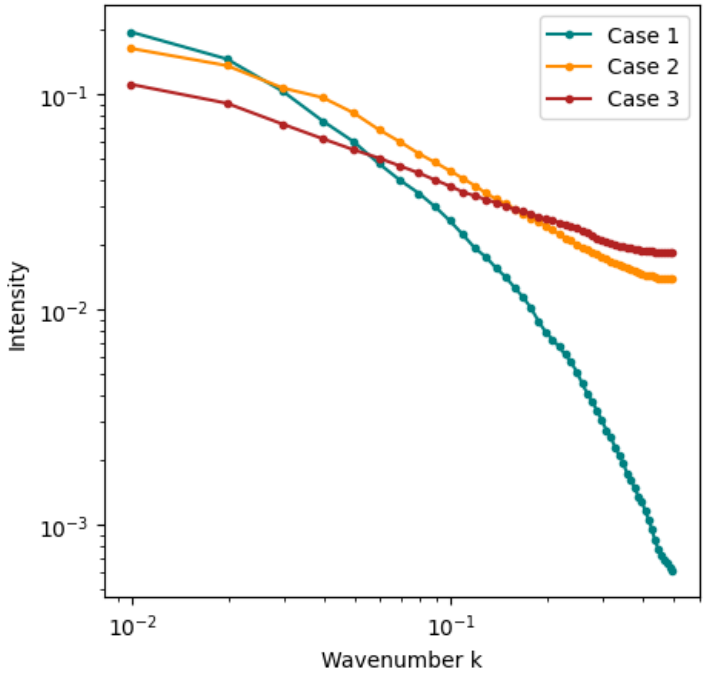

Figure 10: Power density spectrum of density fluctuations.

4 LAPD

The LAPD system is not a major focus of the present work and no benchmark simulations have been done so far. Instead, a brief description of the system is provided herein. The Large Plasma Device (LAPD) is a linear plasma device at UCLA [8]. With a plasma length of 17m, a vacuum vessel radius of 50cm and typical plasma radius of 30cm, LAPD primarily features helium plasmas [9]. The plasma itself is created by electrodes at each end of the device creating a stream of electrons in the centre of the plasma which also acts as an energy source of the system. This leads to high electron temperatures and lower temperatures of the ions which must obtain their energy through electron-ion collisions. Due to the system featuring relatively low densities, collisionalities are low and lead to a large ion-electron temperature disparity as well as weak interactions with the neutrals. A simple Hermes-3 implementation features evolved electron energy, constant temperature or constant profile ion temperatures along with a optional addition of an isotropic neutral background along with a charge exchange reaction rate. The domain is annular with zero-gradient radial boundary conditions and sheath boundaries at each longitudinal end [10].

5 Conclusions

1. BOUT++ codes can be used to provide an effective benchmark for blob transport and simple edge turbulence cases
2. In the blob case artificial diffusion must be added to facilitate comparison with spectral codes.
3. The Hasegawa-Wakatani 3D turbulence system runs well for MAST-U type parameters in BOUT++ and can be used to investigate different turbulence regimes. Runtimes are on the order of minutes on 32 cores.

6 Supplementary material

Please see the GitHub repo in [27] for additional content. Simulation files are not provided due to their size but are available to send upon request. The repo contains:

1. All of the plots in the report along with all of the code used to generate them in a Jupyter Notebook (note: requires xBOUT and xHermes)
2. Code to unnormalise the simulations and calculate all of the post-processing and input quantities
3. Code to generate animations from the simulations as well as gif animations of density, potential and vorticity for all three cases
4. Input files for all three cases

References

[1] Example 'Blob2d' taken from hermes-3 manual (https://hermes3.readthedocs.io/en/latest/examples.html)

[2] S.I. Krasheninnikov, D.A. D'Ippolito & J.R. Myra, J. Plasma Physics, 74. 679 (2008).

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

[4] J. Anderson & B. Hnat, Phys. Plasmas, 24, 062301 (2017).

[5] B. Friedman, T. A. Carter, Phys. Plasmas, 22, 012307 (2015)

[6] R. Numata, R. Ball, R. Dewar Phys. Plasmas 14, 102312 (2007)

[7] S. B. Korsholm, P. K. Michelsen, V. Naulin Phys. Plasmas 6, 2401–2408 (1999)

[8] W. Gekelman et al. Rev. Sci. Instrum. 62, 2875–2883 (1991)

[9] P. Popovich et al. Phys. Plasmas 17, 122312 (2010)

[10] B. Dudson, personal communication.

[11] B.D. Dudson et al., Computer Physics Communications, 180, 1467 (2009)

[12] B.D. Dudson et al., arXiv:2303.12131

[13] B D Dudson et al Plasma Phys. Control. Fusion 61 065008 (2009)

[14] T.D Rognlien et al J. Nucl. Mater. 196–198 347–51 (1992)

[15] L. Easy et al Phys. Plasmas 21, 122515 (2014)

[16] T. Gheorghiu, F. Militello, J. Juul Rasmussen, Phys. Plasmas 31, 013901 (2024)

[17] Ö D Gürcan and P H Diamond, J. Phys. A: Math. Theor. 48 293001 (2015)

[18] F. Chen, Introduction to Plasma Physics and Controlled Fusion, ISBN 978-3-319-22308-7 (2016)

[19] https://library.psfc.mit.edu/catalog/online_pubs/NRL_FORMULARY_19.pdf

[20] https://bout-dev.readthedocs.io/en/latest/user_docs/bout_options.html

[21] https://github.com/boutproject/xhermes

[22] https://github.com/boutproject/BOUT-dev/tree/master/examples/hasegawa-wakatani-3d

[23] https://github.com/boutproject/xBOUT

[24] https://github.com/boutproject/xBOUT-examples/tree/master

[25] https://bout-dev.readthedocs.io/en/stable/user_docs/variable_init.html#expressions

[26] https://bout-dev.readthedocs.io/en/stable/user$_d$ocs/input$_g$rids.htmlperiodic $-$ x $-$ $domainshttps://github.com/mikekryjak/neptune-docs$