Global full-f gyrokinetic simulation of linear ion temperature gradient modes in a stellarator plasma

S. Matsuoka[1,2], Y. Idomura[3], M. Nakata[1,2]

[1] *National Institute for Fusion Science, Toki, Japan*

[2] *The Graduate University for Advanced Studies, SOKENDAI, Toki, Japan*

[3] *Japan Atomic Energy Agency, Kashiwa, Japan*

Introduction

Global full-f gyrokinetic simulations, in which the total particle distribution function f is solved without a scale separation based on the first principles, are recognized as powerful tools to study the turbulent and neoclassical transport in magnetically confined plasmas. Rich physics such as avalanche-like non-local transport and the enhanced neoclassical transport of impurities has been revealed for axisymmetric tokamaks using the full-f gyrokinetic simulations. However, most of the existing global full-f gyrokinetic simulations have been applied to exclusively to tokamaks, and the extension to three-dimensional (3D) devices such as stellarators/heliotrons still needs much numerical effort due to their complicated magnetic field geometries. Owing to the relatively large neoclassical transport and the ambipolar radial electric field determined by the neoclassical particle flux, it is rather important to self-consistently study the neoclassical and turbulent transport in 3D plasmas based on the full-f model. In this work, we report the recent development of the 3D extension of our global full-f gyrokinetic Eulerian simulation code, GT5D [1].

Numerical model of GT5D

GT5D is based on the modern gyrokinetic theory, in which the gyrokinetic equation for the total particle distribution f is solved in the conservative form. In the full-f gyrokinetic simulation model, it is essential to numerically satisfy the conservation laws of the particle number and energy. For this purpose, GT5D uses the non-dissipative conservative finite difference scheme, called Morinishi scheme [2], in which the divergence of the magnetic field should be zero so as to guarantee the incompressible Hamiltonian flow. Thus, in extending GT5D to 3D devices, it is required to use the solenoidal magnetic field with 3D toroidal configurations. In order to meet the requirement, we employ an analytical 3D toroidal field model by Dommaschk [3], which is derived to satisfy $\nabla \cdot \boldsymbol{B} = 0$, since it is difficult for numerical equilibrium codes based on the ideal MHD equation to rigorously satisfy the condition. The use of the model 3D fields enables us to use Morinishi scheme in discretizing the gyrokinetic equation in GT5D.

The electrostatic potential is determined by solving the gyrokinetic Poisson equation, or the quasineutrality condition given as,

$$-\nabla_\perp \cdot \frac{\rho_{\mathrm{ti}}^2}{\lambda_{\mathrm{Di}}^2} \nabla_\perp \phi - \frac{1}{\lambda_{\mathrm{De}}^2} (\phi - \langle \phi \rangle) = 4\pi e \int d^6 Z \delta \left(\boldsymbol{R} + \boldsymbol{\rho} - \boldsymbol{x} \right) \delta f_i, \tag{1}$$

where ϕ is the electrostatic potential, $\delta f_i = f - f_0$ is the perturbed distribution with the initial distribution f_0, $\boldsymbol{R} + \boldsymbol{\rho}$ denotes the particle position, $\boldsymbol{\rho}$ is the Larmor radius, $d^6 Z$ is the phase space volume of the gyrocenter coordinates $(\boldsymbol{R}, v_\parallel, \mu, \alpha)$, ρ_{ti} is the Larmor radius evaluated with the thermal velocity $v_{\mathrm{ti}} = \sqrt{T_{\mathrm{i}}/m_{\mathrm{i}}}$, Λ_{Ds} $(s = \mathrm{e}, \mathrm{i})$ is the Debye length, and $\langle \cdot \rangle$ denotes the flux surface average. Equation (1) contains the flux surface average of the electrostatic potential $\langle \phi \rangle$ as a consequence of the adiabatic electron response, which requires the construction of the smooth magnetic coordinates for general 3D toroidal fields. However, the construction of the magnetic coordinates for 3D configurations often suffers from a severe numerical discontinuity near a low-order rational surface. The discontinuity can be avoided by a new numerical technique, called dense mapping, in which the field-line average computed for all field lines in the computational domain is used to label flux surfaces [4].

Another point in extending GT5D is the development of the gyrokinetic Poisson equation solver for 3D fields with a reasonable numerical cost. In GT5D, the gyrokinetic Poisson equation (1) is discretized in the magnetic coordinates (s, θ, ζ) using the finite element method (FEM), in which the electrostatic potential ϕ is represented by a three-dimensional B-spline discretization: $\phi = \sum_\mu \hat{\phi}_\mu \Lambda_\mu (s, \theta, \zeta)$. Here μ denotes a multi index of 3D finite elements $\Lambda(s, \theta, \zeta)$. Equation (1) is solved with the Dirichlet ($\phi = 0$) and natural boundary conditions at the edge and the axis, respectively. Multiplying Eq. (1) by Λ_ν and integrating over the whole volume yield a matrix form of the quasineutrality condition:

$$\left(H_{\mu\nu} - M_{\mu\nu} \right) \hat{\phi}_\nu = \hat{g}_\mu \tag{2}$$

where $M_{\mu\nu}$ and $H_{\mu\nu}$ are the flux surface average operator from $\langle \phi \rangle$, and that from the other terms in the left hand side of Eq. (1), respectively, and $\hat{g}_\mu$ represents the ion perturbed density. The matrix $M_{\mu\nu}$ becomes dense with a large number of non-zeros in it due to the flux surface average, making it difficult to solve Eq. (2) efficiently. In order to reduce the computational cost of the 3D Poisson solver, we use the matrix decomposition (MD) technique developed by Borchardt *et al* [5]. Using the technique, only the $H_{\mu\nu}$ part needs to be solved at each time step, which gives the solution of $H_{\mu\nu} \tilde{\phi}_\nu = \hat{g}_\mu$; the solution to Eq. (2) is then reconstructed by applying additional matrix-by-vector products as $\hat{\phi}_\nu = \tilde{\phi}_\nu - \tilde{M}_{\nu\mu} \tilde{\phi}_\mu$, where $\tilde{M}_{\nu\mu}$ is a matrix precalculated at the initial step from $M_{\mu\nu}$. Since the matrix $H_{\mu\nu}$ is sparse, $H_{\mu\nu} \tilde{\phi}_\nu = \hat{g}_\mu$ is easily solved using an iterative method.

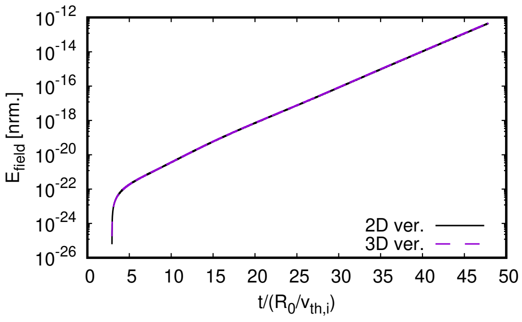

Figure 1: Time evolution of the field energy of the linear ITG turbulence simulations in a circular tokamak observed by the original 2D and 3D versions of GT5D.

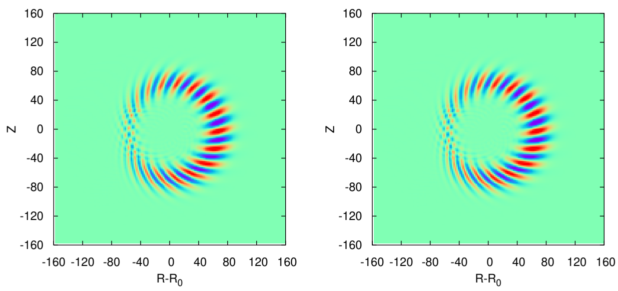

Figure 2: Linear eigenfunctions of the ITG mode turbulence with the toroidal mode number $n = 15$ observed by (left) 2D and (right) 3D versions of GT5D.

Numerical results

For the verification of the extended version of GT5D, linear ITG turbulence simulations for single toroidal mode $n = 15$ are performed for a circular concentric tokamak with Cyclone like parameters, $a/\rho_{\mathrm{ti}} = 150$, and $R_0/a = 2.79$ using the original 2D version of GT5D and its 3D extended version, where R_0 and a are the major and minor radii, respectively. It should be noted that the Fourier decomposition in the toroidal direction is used in the Poisson solver of the 2D version instead of B-splines of the 3D Poisson solver. Since B-splines in the 3D solver acts as a low-pass filter in Fourier space, we impose a numerical filter, which mimics the low-pass filter of B-splines, in the 2D Poisson solver in order to make a comparison between both versions. Figures 1 and 2 show the time evolution of the field energy and linear eigenfunctions of the ITG mode observed by the 2D and 3D versions of GT5D. From the figures, we can confirm that the 3D extended GT5D well reproduces almost the same results of the 2D version.

Then, the performance of the 3D Poisson solver is examined. In the performance test, we solve the quasineutrality equation (2) using $(N_s, N_\theta, N_\zeta) = (130, 256, 32)$ on JFRS-1. The total number of cores used in the test is 1600. Table 1 summarizes the elapsed time and the memory usage for three Poisson solvers in GT5D; 2D FEM with the 1D Fourier decomposition (2D solver), 3D FEM solver, and 3D FEM solver with the matrix decomposition (MD) technique. Here, we use the direct method based on LU decomposition in the 2D solver, whereas the iterative method (generalized minimum residual method (GMRES) with blcok Jacobi preconditioner) provided by PARCEL library [6] in the 3D solvers. From the table, the memory usage in the 3D FEM solver is significantly reduced by applying the MD technique. Furthermore, we can achieve ~ 40 times speed up in the 3D solver with MD compared with that without MD. This is due to the fact that the number of the matrix-by-vector products in the iterative method

Table 1: Summary of the elapsed time and the memory usage for the Poisson solvers of GT5D on JFRS-1. The problem size is $(N_s, N_\theta, N_\zeta) = (130, 256, 32)$.

Solver	Elapsed time / step	Memory Usage
2D-FEM + 1D-FFT	0.39 [s]	0.82 [GByte]
3D-FEM	246 [s]	345.7 [GByte]
3D-FEM (MD)	6.18 [s]	23.05 [GByte]

in $H_{\mu\nu}$ is smaller by two orders of magnitude than the total matrix $H_{\mu\nu} - M_{\mu\nu}$. Although the computational cost of the 3D solver with MD is still higher than the original 2D solver, the elapsed time is acceptable from the viewpoint of ITG simulations.

Summary

In this work, the recent development of the 3D extended version of a global full-f gyrokinetic simulation code, GT5D, has been reported. In order to numerically satisfy the conservation properties for complicated 3D fields, an analytical stellarator magnetic field of the Dommaschk model is used in the 3D version of GT5D. A new gyrokinetic Poisson solver for the 3D field has been implemented using the matrix decomposition technique by Borchardt *et al.* with the help of dense mapping to construct the smooth magnetic coordinates. The 3D extended GT5D has been successfully verified for the linear ITG simulation in an axisymmetric tokamak through the comparison with the original 2D version. The computational time and memory usage of the 3D Poisson solver has been confirmed to be significantly reduced by the matrix decomposition technique. Linear ITG simulations for the model stellarator fields will be performed in the next step using the 3D extended version of GT5D developed in this work.

References

[1] Y. Idomura *et al*., Comput. Phys. Commun. **179**, 391 (2008).

[2] Y. Morinishi *et al*., J. Comput. Phys. **143**, 90 (1998).

[3] W. Dommaschk, Comput. Phys. Commun. **40**, 203 (1986).

[4] S. Matsuoka, submitted to Plasma Fusion Res.

[5] M. Borcahrdt, R. Kleiber and W. Hackbusch, J. Comput. Phys. **231**, 6207 (2012).

[6] PARCEL; Parallel Computing Elements, available online at https://ccse.jaea.go.jp/software/PARCEL