Parallelization of a local gyrokinetic Vlasov simulation code for peta-scale computing

S. Maeyama [1], T.-H. Watanabe [2], Y. Idomura [1], M. Nakata [1], M. Nunami [2], and A. Ishizawa [2]

[1] Japan Atomic Energy Agency, 2-166 Omotedate, Obuchi, Rokkasho-mura, Kamikita-gun 039-3212, Japan
[2] National Institute for Fusion Science, 322-6 Oroshi-cho, Toki 509-5292, Japan

1 Introduction

Turbulent transport is one of the most important issues in magnetically-confined plasma researches. Pressure gradients of confined plasma destabilize micro-instabilities, and the destabilized micro-fluctuations drive plasma turbulence via nonlinear coupling between perturbations of electric potential and plasma distribution function. The plasma turbulence causes particle and heat transport perpendicular to magnetic fields and degrades confinement. Typical wavelengths of the plasma turbulence are of the order of gyroradii of charged particles, while its time scales are slower than gyrations.

To treat plasma turbulence, the gyrokinetic theory has been developed. The equations describe time evolution of gyrophase-averaged distribution functions and electric potential with retaining finite gyroradius effects. A lot of numerical simulations based on the gyrokinetic equations has been carried out and have contributed to understandings of plasma turbulence[1]. The development of numerical techniques for gyrokinetic simulations has continued to improve their applicability, accuracy and efficiency.

Gyrokinetic simulations require expensive computational resources, since they have to solve time evolution in five-dimensional phase space. We have investigated ion-scale plasma turbulence by using our gyrokinetic simulation code GKV[2] on tera-scale supercomputers. Plasma turbulence, however, is originally multi-scale physics, which includes temporal and spatial scales of electrons and ions. Problem sizes for resolving the multi-scale turbulence are 1836 (corresponding to the ion-to-electron mass ratio) times larger than that of ion-scale turbulence. To deal with this numerically challenging problem, we have to develop numerical methods for applying the GKV code to peta-scale computing.

2 The GKV code

The GKV code is originally developed to investigate ion-temperature-gradient-driven turbulence with the adiabatic electron approximation. Extensions of the code for treating both of kinetic ions and electrons are in progress. Since the employed numerical algorithms are principally the same, we treat gyrokinetic equations with the adiabatic electron approximation in the following manuscript.

2.1 Governing equations

The GKV code solves the so-called δf gyrokinetic equations, where the distribution function is split into the equilibrium part F_M and the perturbed part δf. Then the time evolution of the gyrophase-averaged perturbed ion distribution function $\delta\bar{f}_i(\boldsymbol{r}, v_{||}, \mu; t)$ is described by the gyrokinetic Vlasov equation,

$$\left[\frac{\partial}{\partial t} + \left(v_{||}\frac{\boldsymbol{B}}{B} + \boldsymbol{v}_d + \boldsymbol{v}_E\right)\cdot\nabla - \frac{\mu\nabla_{||}B}{m_i}\frac{\partial}{\partial v_{||}}\right]\delta\bar{f}_i = S + C, \quad (1)$$

where $v_{||}$, $\mathbf{v}_d$ and $\mathbf{v}_E$ are the velocity parallel to the confinement magnetic field, the perpendicular magnetic drift velocity and the perpendicular E×B drift velocity due to electric potential perturbations. The term with the magnetic moment μ and the ion mass m_i represents parallel acceleration by the mirror force. The linear term associated with the equilibrium distribution S contains contributions of the parallel electric field and equilibrium pressure gradients, which drive micro-instabilities and plasma turbulence. The model collision operator C is friction and diffusion operators in velocity space ($v_{||}$, μ). The perturbed electric potential ϕ is given by the gyrokinetic quasi-neutrality equation with the elementary electric charge e and the ion equilibrium temperature T_i,

$$\int\left[\delta\bar{f}_i - \frac{eF_M}{T_i}(\phi - \bar{\phi})\right]dv^3 = \delta n_e, \quad (2)$$

where $\bar{\phi}$ is the gyrophase-averaged potential. It should be noted that the velocity space integral must be taken holding particle (not gyrocenter) position fixed. The perturbed electron density n_e is assumed to be

$$\frac{\delta n_e}{n_0} = \frac{e(\phi - \langle\phi\rangle)}{T_e}, \quad (3)$$

where $\langle\cdots\rangle$ denotes the flux surface average.

2.2 Simulation domain and boundary conditions

In the δf framework, it is assumed that a steady equilibrium exists and satisfies the MHD equilibrium condition. Then, we can employ magnetic coordinates as configuration space coordinates $\mathbf{r}$ = (x, y, z), where the equilibrium magnetic field is described as

$$\boldsymbol{B} = B_0\nabla x\times\nabla y = \frac{B_0}{\sqrt{g}}\frac{\partial\boldsymbol{r}}{\partial z}. \quad (4)$$

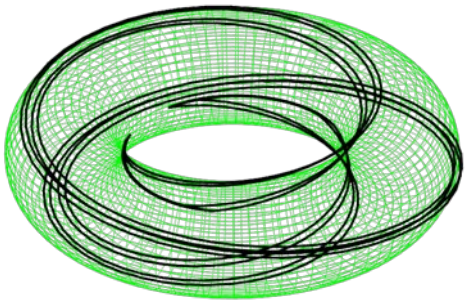

Fig. 1. An example of the flux-tube simulation domain is plotted by black lines. Green lines represent a circular toroidal flux surface.

The flux-surface label x, the field-line label y and the field-aligned coordinate z correspond to the toroidal coordinates (r, θ, ζ) as $x = r - r_0$, $y = r_0(q\theta - \zeta)/q_0$, $z = \theta$ in a large-aspect-ratio tokamak with concentric circular magnetic flux surfaces, where q is the so-called safety factor and the quantities with subscript 0 denotes the value at the center of the simulation domain. While plasma turbulence has short perpendicular wavelength, its structure elongates in the direction parallel to the magnetic field. Therefore, a long and thin simulation domain along magnetic field lines is suitable for capturing the nature of plasma turbulence with reducing computational costs. An example of this flux-tube simulation domain is shown in Fig. 1, which is written by the projection of a box with short lengths in x and y and a long length in z. The flux-tube model is widely used to analyse turbulent transport in the local approximation limit, where the equilibrium quantities are given by local values.

We assume statistically periodic boundary conditions in x and y and apply the Fourier decomposition as

$$\phi(x,y,z) = \sum_{k_x}\sum_{k_y} \phi_{k_x,k_y}(z)\exp\left[i(k_x x + k_y y)\right]. \quad (5)$$

Additionally, there is the physical periodicity in the poloidal angle θ as $\phi(r, \theta, \zeta) = \phi(r, \theta+2\pi, \zeta)$. This leads the modified periodic boundary condition along the field-aligned coordinate z,

$$\phi_{k_x,k_y}(z) = \Theta\phi_{k_x+\Delta,k_y}(z+2\pi), \quad (6)$$

where the connection phase Θ and connection wave number Δ depends on the poloidal wave number k_y[3].

Eqs. (1)-(3) are numerically solved in (k_x, k_y, z, $v_{\|}$, μ) space except the E×B advection. Since direct calculations of nonlinear convolutions in wave number space are computationally too expensive, the nonlinear E×B advection term is evaluated in the real space and transformed back to the wave number space by means of the fast Fourier transform (FFT) algorithms and the 3/2 de-aliasing rule.

3 Parallelization for peta-scale computing

To attain good performance on a distributed-memory system, domain decomposition by using message passing interface (MPI) library is necessary, as well as thread parallelization. The original GKV code decomposes five-dimensional phase space in three directions (z, $v_{\|}$, μ), which is not enough for peta-scale supercomputers. In order to advance to peta-scale computing, we additionally decompose wave-number space **k** = (k_x, k_y). By using the one-dimensional FFT, the E×B advection term is evaluated in the following way:

- 1D inverse FFT for k_y with the decomposition in k_x.
- Data distribution as the data in k_x becomes local.
- 1D inverse FFT for k_x with decomposition in y.
- Calculation of the E×B advection term in real space (x, y).
- 1D FFT for x with the decomposition in y.
- Data distribution as the data in y becomes local.
- 1D FFT for y with the decomposition in k_x.

Efficiency of this parallelization method is examined via a strong scaling test. Computations were carried out on the FX10 super computer in the University of Tokyo. In Fig. 2, relative speed of the computation, which is proportional to inverse of the elapsed time, is plotted as a function of the number of the wave-number-space decomposition. It demonstrates that the wave-number-space parallelization has good strong scaling in this range. We note that perpendicular grid number required for multi-scale turbulence simulations is larger than the case shown above, e.g., $N_x = N_y = 4096$. The

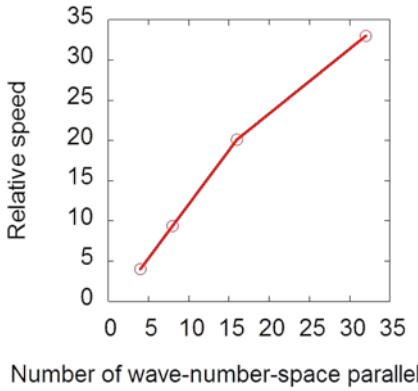

Fig. 2. Strong scaling of the wave-number-space decomposition (where $N_x = 512$, $N_y = 512$, $N_z = 32$, $N_{v_{\|}} = 32$, $N_{\mu} = 16$). (**k**, z, $v_{\|}$, μ) coordinates are decomposed into 4 (and 8, 16, 32) × 2 × 2 × 2 subdomains and 8 symmetric multi-processing threads are employed per subdomain, which corresponds to 256 - 2048 cores.

result shown in Fig. 2 suggests applicability of the wave-number-space parallelization on peta-scale supercomputers. When we employ hundreds of thousands of cores, however, MPI communication time will become crucial, since data distributions in wave-number space are collective communications. We have to check computation-to-communication ratio carefully, and may have to employ masking of MPI communication time by overlapping between computations and communications.

4 Conclusion

We have presented a local gyrokinetic Vlasov simulation code to treat turbulent transport in magnetically-confined plasmas. Calculations of time evolution of distribution functions in five-dimensional phase space are computationally expensive, even when a flux-tube simulation domain is employed to reduce numerical costs. The simulation code is massively parallelized by domain decomposition in five-dimensional phase space and thread parallelization. First scaling test of the wave-number-space decomposition demonstrates good strong scaling.

Acknowledgements

This work is performed with supports of HPCI Strategic Programs for Innovative Research and JAEA-NIFS collaboration program.

References

[1] X. Garbet, Y. Idomura, L. Villard, and T.-H. Watanabe, Gyrokinetic simulations of turbulent transport, Nucl. Fusion, 50, 043002, 2010.

[2] T.-H. Watanabe, and H. Sugama, Velocity–space structures of distribution function in toroidal ion temperature gradient turbulence, Nucl. Fusion, 46, 24, 2006.

[3] M. A. Beer, S. C. Cowley, and G. W. Hammett, Field-aligned coordinates for nonlinear simulations of tokamak turbulence, Phys. Plasmas, 2, 2687, 1995.