Princeton Plasma Physics Laboratory

PPPL-5212

A new hybrid-Lagrangian numerical scheme for gyrokinetic simulation of tokamak edge plasma

S. Ku[1], R. Hager[1], C.S. Chang[1], J.M. Kwon[2], and S.E. Parker[3]

1) Princeton Plasma Physics Laboratory, Princeton University, Princeton, New Jersey 08543, USA
2) National Fusion Research Institue, Korea
3) University of Colorado Boulder, USA

September 2015

Prepared for the U.S. Department of Energy under Contract DE-AC02-09CH11466.

Princeton Plasma Physics Laboratory
Report Disclaimers

Full Legal Disclaimer

Trademark Disclaimer

PPPL Report Availability

Princeton Plasma Physics Laboratory:

http://www.pppl.gov/techreports.cfm

Office of Scientific and Technical Information (OSTI):

http://www.osti.gov/scitech/

Related Links:

U.S. Department of Energy

U.S. Department of Energy Office of Science

U.S. Department of Energy Office of Fusion Energy Sciences

A new hybrid-Lagrangian numerical scheme for gyrokinetic simulation of tokamak edge plasma

S. Ku,[1, a)] R. Hager,[1] C.S. Chang,[1] J.M. Kwon,[2] and S.E. Parker[3]

[1)] *Princeton Plasma Physics Laboratory, Princeton University, Princeton, New Jersey 08543, USA*

[2)] *National Fusion Research Institue, Korea*

[3)] *University of Colorado Boulder, USA*

(Dated: September 29, 2015)

A new hybrid-Lagrangian δf scheme has been developed that utilizes the phase space grid in addition to the usual marker particles, in order to take advantage of the computational and physical strengths from both sides. The new scheme splits the particle distribution function of a kinetic equation into two parts. Marker particles contain the fast space-time varying, δf part of the distribution function and the coarse-grained phase-space grid contains the slow space-time varying part. The coarse-grained phase-space grid has a low memory-requirement while the marker particles provide scalable computing ability. Weights of the marker particles are determined by a direct weight evolution equation instead of the differential form weight evolution equations that the conventional delta-f schemes use. The particle weight is slowly transferred to the phase space grid, thereby reducing the growth of the particle weights. The non-Lagrangian part of the kinetic equation – e.g., collision operation, ionization, charge exchange, heat-source, radiative cooling, and others – can be operated directly on the phase space grid. Deviation of the particle distribution function on the velocity grid from a Maxwellian distribution function is allowed to be arbitrarily large. The numerical scheme is implemented in the gyrokinetic particle code XGC1, which specializes in simulating the tokamak edge plasma that crosses the magnetic separatrix and is in contact with the material wall.

[a)] Electronic mail: sku@pppl.gov

I. INTRODUCTION

In recent decades the δf scheme for Lagrangian particle-in-cell (PIC) simulation of the five dimensional (5D) gyrokinetic equations, of which some early works can be represented by Refs. 1–3, has rendered many successful interpretations of the turbulent plasma transport in toroidal magnetic confinement devices. An alternative method emerged later using the Eulerian phase-space grid that avoided the Monte-Carlo particle noise at the expense of the enhanced grid memory requirement[4]. The present work focuses on a further development of the Lagrangian PIC δf scheme from the "conventional" one as used by the existing gyrokinetic codes to a hybrid-Lagrangian scheme that also utilizes the Eulerian phase-space grid, in order to treat the irreversible physics phenomena on the right-hand side of the Boltzmann equation.

When the tokamak core plasma is near Maxwellian, the δf scheme reduces the statistical noise from the finite number of marker particles and allows computationally efficient turbulence simulations, since majority of the plasma is described by an analytic Maxwellian distribution function. Unlike the core plasma, however, the tokamak edge plasma can be significantly non-Maxwellian because it has a steep gradient whose scale length is similar to the banana orbit width, is in-contact with the material wall, and has a strong fluctuation level. Also, tokamak edge plasmas have strong sources and sinks from wall loss, neutral ionization and charge-exchange, radiatvie cooling and others. Non-maxwellian distributions and strong sources and sinks are difficult to be described by the conventional δf scheme since the conventional δf scheme prefers small perturbation of the particle distribution function from a known Maxwellian distribution function. The original total-f (also called full-f) method can include these effects. The total-f method yields the most straightforward particle-in-cell simulation in solving kinetic equations without changing of weight from particle motion. It was successfully used to simulate the tokamak edge plasmas with adiabatic electron response[5]. However, the total-f method has less freedom than the δf method in using various numerical techniques such as the split weight scheme[6] and fluid-kinetic hybrid schemes[7,8]. In this report, we introduce a new δf scheme, called "hybrid-Lagrangian δf" scheme here, which can handle the non-Maxwellian distribution function, non-linear collisions, and strong sources and sinks by utilizing the velocity space grids, which is mathematically identical to the total-f scheme.

In the sense that the new hybrid-Lagrangian scheme uses both the marker particles and the phase space grid together, it has some similarities with the semi-Lagrangian method[9,10]. The key differences between the new hybrid-Lagrangian scheme and the semi-Lagrangian method are that (1) the semi-Lagrangian method resets the particle positions on the phase-space grid at every time step, but the new hybrid-Lagrangian scheme keeps the particle positions, and that (2) the new hybrid-Lagrangian scheme stores the fast space-time varying distribution function in marker particles but the semi-Lagrangian scheme stores all the information on the phase space grid.

Another Lagrangian particle scheme that utilizes the phase space grid is the coarse-graining reset procedure[11–13] for reduction of the growing weight problem. In the procedure, the δf particles are periodically interpolated at a large time interval to the grid and the particles are reset to their original phase-space coordinates with the new δf values. The key difference between the new hybrid scheme and the coarse-graining procedure is that (1) the new hybrid scheme keeps the coarse-graining information on the phase-space grid while the coarse graining procedure gives it back to the particles, and that (2) the new hybrid scheme evolves f_0 so that it can reduce both the mean and the variance of the δf particle weights.

In section II, the new hybrid-Lagrangian δf scheme is presented. In section III, we demonstrate the new hybrid-Lagrangian scheme using the XGC1 code. Section IV presents summary and discussions.

II. A NEW HYBRID-LAGRANGIAN δf SCHEME

A. Generalization of the δf scheme

In the conventional δf scheme that solves the collisionless Vlasov equation

$$\frac{Df}{Dt} \equiv \frac{\partial f}{\partial t} + \dot{\mathbf{z}}\frac{\partial f}{\partial \mathbf{z}} = 0, \tag{1}$$

where f is the particle distribution function, $\mathbf{z}$ is the vector of phase space variables (the configuration space and velocity coordinates), and $f = f(\mathbf{z})$ is the distribution function in phase space, the division of f into two parts $f = f_0 + \delta f$ yields

$$\frac{D\delta f}{Dt} = -\frac{Df_0}{Dt}.$$

δf is then evaluated from the weight evolution equation[1–3], and f_0 can be an arbitrary predetermined function. If f_0 is a known function and δf is a small perturbation from f_0, then the computational cost savings can be substantial.

In a realistic system, there is the Coulomb collision operator on the right-hand side (RHS) and the system is described by the Boltzmann equation,

$$\frac{Df}{Dt} \equiv \frac{\partial f}{\partial t} + \dot{\mathbf{z}}\frac{\partial f}{\partial \mathbf{z}} = C(f).$$

Utilizing $f = f_0 + \delta f$, we obtain

$$\frac{D\delta f}{Dt} - C(\delta f) = -\frac{Df_0}{Dt} + C(f_0)$$

In this system, the equilibrium thermodynamics yields $f_0 = f_M$ with $\delta f = 0$ according to the H-theorem, where f_M is the equilibrium Maxwellian distribution function in the velocity space, with $C(f_M) = 0$. In order to describe a small deviation of the system from the thermodynamic equilibrium, $f_0 = f_M$ is chosen and some δf is allowed to develop. If we choose $f_0 \neq f_M$, then the physics is dominated by the large collisional process on f_0: $C(f_0) \neq 0$, hence there is a fast relaxation of f_0. The small-δf scheme, then, loses its merit. For example, when δf is chosen to describe the turbulence physics, then the collisional physics from f_0 dominates the δf turbulence physics, and the choice $f_0 \neq f_M$ becomes poor. In a quiescent core region of a tokamak plasma, the non-Maxwellian driver is regarded to be small. Thus, the choice of a near-equilibrium $f_0 = f_M$ with a small perturbative δf is justified, and used by most of the tokamak microturbulence codes describing the core region (based on a reduced five dimensional gyrokinetic equation). Representative examples of such Lagrangian codes are GEM[14], GTS[15], and GTC[2]. Since f_0 is a Maxwellian, a linearized Coulomb collision can be used, which makes the simulation much easier. Furthermore, many such codes consider only the turbulence drivers in Df_M/Dt on the right-hand side of the Boltzmann equation by neglecting the non-Maxwellian-causing Grad-B and curvature drivers, under the assumption that the spatial gradient of f_M is negligibly small (in this system, the definition of the total differential operator, D is not the same between RHS and LHS, and the Liouville's phase-space conservation law is broken by that amount). If the spatial gradient is not small, the Grad-B and curvature drivers make the δf particle weights to grow to large negative values inhomogeneously in the velocity space, causing a large velocity-dependent deviation of δf from a Maxwellian distribution. In a particle

code, this also means the growth of the Monte Carlo noise. In a continuum code, the steep gradient in the configuration and velocity space could provoke a type of Courant instability. The issue of ∇B and curvature drivers could be resolved by using a canonical Maxwellian distribution for f_0 in a collisionless simulation[16]. However, the Coulomb collision operator sees the canonical Maxwellian distribution as a non-Maxwellian distribution function and the validity of the canonical Maxwellian f_0 is destroyed.

In the edge region of a tokamak, unfortunately, there are multiple non-Maxwellian drivers that can put the plasma to a non-equilibrium thermodynamic situation, opposing the Coulomb Maxwellian driver. They are the steep pressure gradient whose scale length is as short as the ion banana width (the non-local ion banana mixing does not allow a local thermodynamic equilibrium state), the neutral ionization and charge exchange, the particle loss to the material wall, the radiative energy loss, and others. All the simplifications used for the near-Maxwellian plasmas are not allowed any more. The Coulomb collisions, which is in the velocity space, needs to be fully non-linear while satisfying the conservation properties. f_0 may be far away from Maxwellian and evolving in time to self-organize with the sources and sinks, the Coulomb collisions, the Grad-B and curvature drifts, and the turbulence. The weight growth from these forces needs to be mitigated. The purpose of this paper is to report the construction of a new hybrid-Lagrangian PIC scheme that utilizes the phase space grid to resolve such a non-equilibrium thermodynamics issue in the kinetic Boltzmann simulation.

We may generalize the Boltzmann equation by adding the source/sink term S on RHS:

$$\frac{Df}{Dt} \equiv \frac{\partial f}{\partial t} + \dot{\mathbf{z}}\frac{\partial f}{\partial \mathbf{z}} = C(f) + S(f), \tag{2}$$

where $C(f)$ is the Fokker-Planck Coulomb collision operator, and $S(f)$ represents the sources and sinks of density, energy and momentum. The left-hand side (LHS) of this system, the Vlasov equation, conserves phase space volume.

In the Lagrangian δf scheme, the marker particles describe a perturbed distribution function ($\delta f = f_p$), from another function (f_0) that is expressed as function of the phase space variables. Hence, the distribution function can be written as

$$f = f_0 + f_p,$$

and the generalized Boltzmann equation becomes

$$\frac{Df_p}{Dt} = -\frac{Df_0}{Dt} + S^*(f), \tag{3}$$

where $S^*(f) = C(f)+S(f)$. We call this equation a generalized Boltzmann δf equation since it contains the source/sink term $S(f)$, the total differential operator D is identical between RHS and LHS, and the Coulomb collision operator C must be nonlinear. The generalized Boltzmann δf equation, Eq. (3) is mathematically identical to the original generalized total-f Boltzmann equation, Eq. (2). It has been shown by Ku et al.[17] that Eq. (3) yields the same self-organized quasi-equilibrium state as Eq. (2) does, even when the marker particle numbers are much smaller in the δf scheme.

The new hybrid-Lagrangian scheme presented in the following subsections solves the total-δf Boltzmann equation, Eq. (3). One issue in solving the total-δf Boltzmann equation is that f_p (or particle weights) can become large due to the strong ∇B and curvature drive, the wall loss, the sources/sinks, or a long time simulation. Growth of the weight in the long time simulation even in the absence of the non-Maxwellian drivers is known as the "growing weight problem"[18]. A method to mitigation the weight growth or f_p is discussed at the end of next subsections. The phase space grid is also used for calculating $S^*(f)$. In the next subsection, we construct a direct weight evolution method as a pre-requite for a successful numerical application of the new hybrid-Lagrangian scheme.

B. A direct weight evolution method for the δf scheme

Since we have $S^*(f)$ on RHS of Eq. (3), we use a two-weight scheme (two weight schemes already appeared in the literature[19–21]). For simplicity, we normalize the weight with w_0, and we choose the initial f to be f_0, i.e $f_p(t=0) = 0$. The normalization factor, w_0 is defined as $w_0 \equiv f_0(\mathbf{z}_{t=0}, t=0)/g$, where $\mathbf{z}_{t=0}$ is the initial positions of the marker particles and g is the distribution function of marker particles. f_p is the weighted marker particle distribution function with w_1,

$$f_p = w_1 w_0 g = w_1 f_0(\mathbf{z}_{t=0}, t=0).$$

For the weight evolution equation, we take the total derivative of the above equation, and obtain the usual differential form weight evolution equation,

$$\frac{dw_1}{dt} = \frac{1}{w_0 g}\left(-\frac{Df_0}{Dt} + S^*(f)\right). \tag{4}$$

Note that g is constant along the collisionless and sourceless/sinkless Vlasov particle trajectory since the phase space volume is conserved, and so is w_0.

Time integration of Eq. (4) gives simple difference form of weight change,

$$\begin{aligned}\Delta w_1 &= -\frac{\Delta f_0}{w_0 g} + \frac{1}{w_0 g}\int S^*(f)dt \\ &\simeq -\frac{\Delta f_0}{w_0 g} + \frac{\Delta t}{w_0 g}S^*(f),\end{aligned} \tag{5}$$

where Δ represents finite difference along the particle trajectory, and the Euler method can be applied to obtain the time integration of $S^*(f)$. We call this a direct weight evolution. Key advantage of using the difference form is that it can avoid numerical calculation of the derivative of f_0. This is important especially when we use the interpolation function on the phase space grid for f_0, to be described in the next section II C. Another advantage is that the error from time integration of Df_0/Dt can be avoided. This may allow a longer time simulation of weight evolution when $S^*(f)$ is small or slowly time varying. Note that if the ∇B and curvature drifts in Df_0/Dt on RHS of Eq. (4) is neglected, as many conventional δf codes do, Df_0/Dt is not an exact total derivative. Hence, these codes cannot utilize the direct weight evolution.

C. The phase space grid for f_0

In the new hybrid-Lagrangian scheme, we describe f_0 using the phase space grid. We decompose f_0 into f_a and f_g, where f_a is an analytic function and f_g is on the phase space grid. Hence,

$$f = f_0 + f_p = f_a + f_g + f_p.$$

If f has exponential decay in kinetic energy, we can chose f_a to be a Maxwellian. f_g is an interpolated function on the 5D phase space grid and has a slow time variation. To capture the slow time variation of f, the following operation is performed at every time step.

$$f_g(t + \Delta t) = f_g(t) + \alpha f_p, \tag{6}$$

where $\alpha \ll 1$ is an arbitrary coefficient which can vary in time and phase space. If we keep performing the above operation over many time steps, f_g can capture the slow time variation part of f_p. From the change of f_g and Eq. (3), Df_p/Dt has the time-decay term accordingly as

$$\frac{Df_p}{Dt} = -\frac{\alpha}{\Delta t} f_p + \left[\frac{Df_p}{Dt}\right]_{\alpha=0}, \tag{7}$$

where $[Df_p/Dt]_{\alpha=0}$ means Df_p/Dt without adjusting f_0. When f reaches a stationary solution without turbulence, f_p decays to zero and f_g contains the deviation of f from f_a.

The decay of w_1 can be evaluated in the following way. From Eq. (5), the change of w_1 becomes

$$\Delta w_1 = -\frac{f_g(t+\Delta t) - f_g(t)}{w_0 g} + [\Delta w_1]_{\alpha=0} \tag{8}$$

Since $[f_g(t+\Delta t) - f_g(t)]/[w_0 g]$ is originated from αf_p, according to Eq. (6), we get

$$\begin{aligned} \Delta w_1 &= -\alpha \frac{f_p}{w_0 g} + [\Delta w_1]_{\alpha=0} \\ &= -\alpha w_1 + [\Delta w_1]_{\alpha=0} \end{aligned} \tag{9}$$

If we use a stationary f_a, only f_g changes in time. In general, f_a can be updated from f_g, while keeping the relation $\Delta f_a = -\Delta f_g$.

D. Phase space grid for $S^*(f)$

Since we already evaluate αf_p at every time step on the phase space grid, the total f on the phase space grid can be obtained with minimal computational cost, and it can be used for calculation of $S^*(f)$. One example is the nonlinear Fokker-Planck-Landau collision on the velocity space grid[22,23]. In tokamak edge plasma, the distribution function is non-Maxwellian and the use of a nonlinear collision operator becomes significant. In this scheme, we calculate the nonlinear Fokker-Planck-Landau collision operator on the velocity space grid, and the $\partial f/\partial t$ on the grid is scattered back to particle weights. In the rare cases that a grid point does not have any corresponding particles, the collisional $\partial f/\partial t$ goes to f_g in order not to omit the physics. Other physics processes such as neutral ionization, neutral charge exchange, radiation cooling, heating, and momentum injection, are implemented in the same way.

III. DEMONSTRATION WITH ION TEMPERATURE GRADIENT TURBULENCE

To demonstrate the new hybrid-Lagrangian scheme, we performed the Ion Temperature Gradient (ITG) turbulence simulations. For simplicity, concentric circular geometry is used, and the adiabatic electron response is assumed, in a 5D gyrokinetic code, XGC1.

A. XGC1

The 5D gyrokinetic code, XGC1 includes the X-point geometry for tokamak edge plasmas. It normally uses the experimental magnetic field and tokamak boundary geometry of divertor and limiter. The simulation domain can be extended to the whole plasma including the magnetic axis and the scrap-off layer which is in-contact with the material wall. To handle the magnetic axis and the X-point, XGC1 uses a cylindrical coordinate system instead of a magnetic coordinate system in the particle equation of motion. Unstructured field-following triangular mesh is used for turbulent field solver. Kinetic equations for the main ions, impurities, and electrons are solved. Electro-static ITG turbulence is assumed for this study. A fully nonlinear Fokker-Plank-Landau collision operator is calculated on the 2D velocity space grid.

XGC1 is designed for high performance computers (HPC), and uses MPI, OpenMP, GPU computing, and vectorization. XGC1 shows good weak and strong scalings to the maximal capability of the leadership HPCs.

B. α factor

The α factor is an arbitrary small coefficient that determines the conversion rate of f_p to f_g according to Eq. (6). When a fraction of f_p from particles is converted into f_g on the grid, a numerical smoothing of δf_p occurs from the grid interpolation. This can dissipate turbulence energy. Fig. 1 shows the heat fluxes with various α values. The simulation size is relatively small, and the characteristic system size is $a/\rho_i = 100$, where a is the minor radius and ρ_i is the ion gyro radius. In the simulations, 3×10^5 real space grid points, each of which has 32 ($v_\perp$) $\times$ 31 ($v_\parallel$) velocity space grid points, are used. The total number of marker particles are 4×10^8, which corresponds to N_g =1300 particles per real space grid node or 1.5

particles per phase space grid node. A simple statistical error bar $1/\sqrt{N_g}$ is shown together in Fig. 1. Temperature profile is simple hyperbolic tangent functions with maximal gradient of $R/L_T = 8$, $R/L_n = 2$, where R is the major radius, L_T is the temperature gradient scale length, and L_n is the density gradient scale length. The adiabatic electron temperature is set to be the same as the ion temperature. The simulation shown in Fig. 1 is non-flux driven. Thus, the heat flux decays eventually as the free energy from the initial profile is exhausted. In Fig. 1, the heat flux grows as turbulence develops, and decays after it reaches the maximum with the non-linear effect as the plasma profile relaxation. The maximum heat flux and the time integrated heat flux depends on the α factor. When α is 0.004 or 0.01, a clear reduction of turbulence and heat flux is observed, and this is sign of damping from the numerical dissipation of the particle-grid interpolation. When $\alpha = 0.001$, the heat flux is acceptably close to the heat flux from the $\alpha = 0$ case.

Numerical dissipation also depends on the resolution of the phase space grid. In Fig. 2, the heat flux from a finer (62×61) velocity space grid is compared with the results from the original 32×31 velocity space grid. When $\alpha = 0.004$ with the original grid, it shows reduction of the turbulence and heat flux level from the $\alpha = 0$ result. Increasing the grid resolution gives reduction of the numerical dissipation, and it restores the heat flux to an acceptably close result to the $\alpha = 0$ result even at $\alpha = 0.004$.

In a rough estimation of the optimal α,

$$\alpha_{\rm opt} \sim C(\Delta \mathbf{z}) \frac{\Delta t}{\tau_{\rm phy}},$$

where Δt is the time step size of the particle-grid conversion operation, and $\tau_{\rm phy}$ is the time scale of the physics phenomena (turbulence correlation time), $\Delta \mathbf{z}$ is the size of phase space grid, and $C(\Delta \mathbf{z})$ is a function of order unity depending on the size of the phase space grid.

Numerical dissipation at α factor is also observed in the flux-driven simulations. Fig. 3 shows the heat fluxes and the temperature gradient scale length by turbulence in the flux-driven simulations with various α. Heat and cooling are applied to near axis and edge region, respectively. In the flux-driven plasma, simulations reach a quasi-steady state, so the final heat flux is almost saturated. If we examine the temperature gradient scale length, $\alpha = 0$ and $\alpha = 0.001$ converges to a similar value. However, $\alpha = 0.01$ gives higher gradient (in other words, less time integrated heat flux) compared to the $\alpha = 0$ case. This tendency is the same as that observed in the non-flux driven simulations of Fig. 1.

Figure 4 shows the mean of $(w_1 w_0 g)^2$ in the velocity space cells with $\alpha = 0$ and $\alpha = 0.0001$ after 1500 time steps. When $\alpha = 0.001$, the particle noise variance is reduced by factor of 4 compared to $\alpha = 0$ case. Since $\alpha \times$ [No. of time steps] $= 1.5$, the result is roughly what we expected from the conversion of weights from f_p to f_g.

IV. SUMMARY AND DISCUSSION

A new hybrid-Lagrangian scheme for gyrokinetic simulation of tokamak edge plasma is developed and implemented in XGC1. The new scheme uses the combination of particle and continuum method. Lagrangian particle push is used for the LHS Vlasov dynamics of the Boltzmann equation. RHS physics of the Boltzmann equation is handled using the continuum gird method. The scheme slowly converts particle weights into the coarse-grained phase space grid in order to mitigate the growing weight issue. Hence, the slow time-varying physics is mostly resides on the phase space grid, while the fast time varying function remains in particles. The optimal conversion rate, α depends on the physics to be resolved and the size of phase space grid. The scheme reduces the statistical noise of a non-Maxwellian plasma simulation, and relaxes growing weight problem.

The new scheme takes advantage of the advantageous features from both particle and continuum schemes. Since the rapidly time varying fine structure is handled by the particles, it is less sensitive to the Courant condition and posses a higher velocity resolution. Since majority of the non-Maxwellian plasma resides on the continuum grid, the Monte-Carlo noise is reduced. Since the particles scale well on HPCs and the coarse-grained continuum grid does not require much memory, the new scheme satisfies both the HPC scalability and the low memory requirement.

ACKNOWLEDGEMENT

This work was supported by the U.S. Department of Energy under Contract No. DE-AC02-09CH11466 and Grant No. DE-SC000801. This work used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

REFERENCES

[1]W. Lee, J. Comput. Phys. **72**, 243 (1987).

[2]Z. Lin, T. Hahm, W. Lee, W. Tang, and R. White, Science **281**, 556 (1998).

[3]S. Parker, C. Kim, and Y. Chen, Phys. Plasmas **6**, 1709 (1999).

[4]F. Jenko, W. Dorland, M. Kotschenreuther, and B. Rogers, Phys. Plasmas **7**, 1904 (2000).

[5]C. Chang, S. Ku, P. Diamond, Z. Lin, S. Parker, T. Hahm, and N. Samatova, Phys. Plasmas **16**, 056108 (2009).

[6]Y. Chen and S. Parker, J. of Computational Phys. **220**, 839 (2007).

[7]Z. Lin and L. Chen, Phys. Plasmas **8**, 1447 (2001).

[8]Y. Chen, S. Parker, J. Lang, and G.-Y. Fu, Phys. Plasmas **17**, 102504 (2010).

[9]E. Sonnendrckera, J. Rochea, P. Bertrandb, and A. Ghizzob, J. Comput. Phys. **149**, 201 (1999).

[10]X. Garbet, Y. Sarazin, V. Grandgirard, G. Dif-Pradalier, and et. al., Nucl. Fusion **47**, 1206 (2007).

[11]J. Denavit, J. Comput. Phys. **9**, 75 (1972).

[12]S. Vadlamani, S. E. Parker, Y. Chen, and C. Kim, Computer Physics Communications **164**, 209 (2004).

[13]Y. Chen and S. Parker, Phys. Plasmas **14**, 082301 (2007).

[14]Y. Chen and S. Parker, Phys. Plasmas **16**, 052305 (2009).

[15]W. Wang, T. Hahm, W. Lee, G. Rewoldt, J. Manickam, and W. Tang, Phys. Plasmas **14**, 072306 (2007).

[16]Y. Idomura, S. Tokuda, and Y. Kishimoto, Nuclear Fusion **43**, 234 (2003).

[17]S. Ku, C. Chang, and P. Diamond, Nucl. Fusion **49**, 115021 (2009).

[18]J. Krommes and G. Hu, Phys. Plasmas **1**, 3211 (1994).

[19]G. Hu and J. Krommes, Phys. Plasmas **1**, 863 (1994).

[20]Y. Chen and R. White, Phys. Plasmas **10**, 3591 (1997).

[21]M. O. W X Wang, N Nakajima and S. Murakami, (1999).

[22]E. Yoon and C. Chang, Phys. Plasmas **21**, 032503 (2014).

[23]R. Hager, E. Yoon, S. Ku, and et al., J. Comput. Physics , submitted (2015).

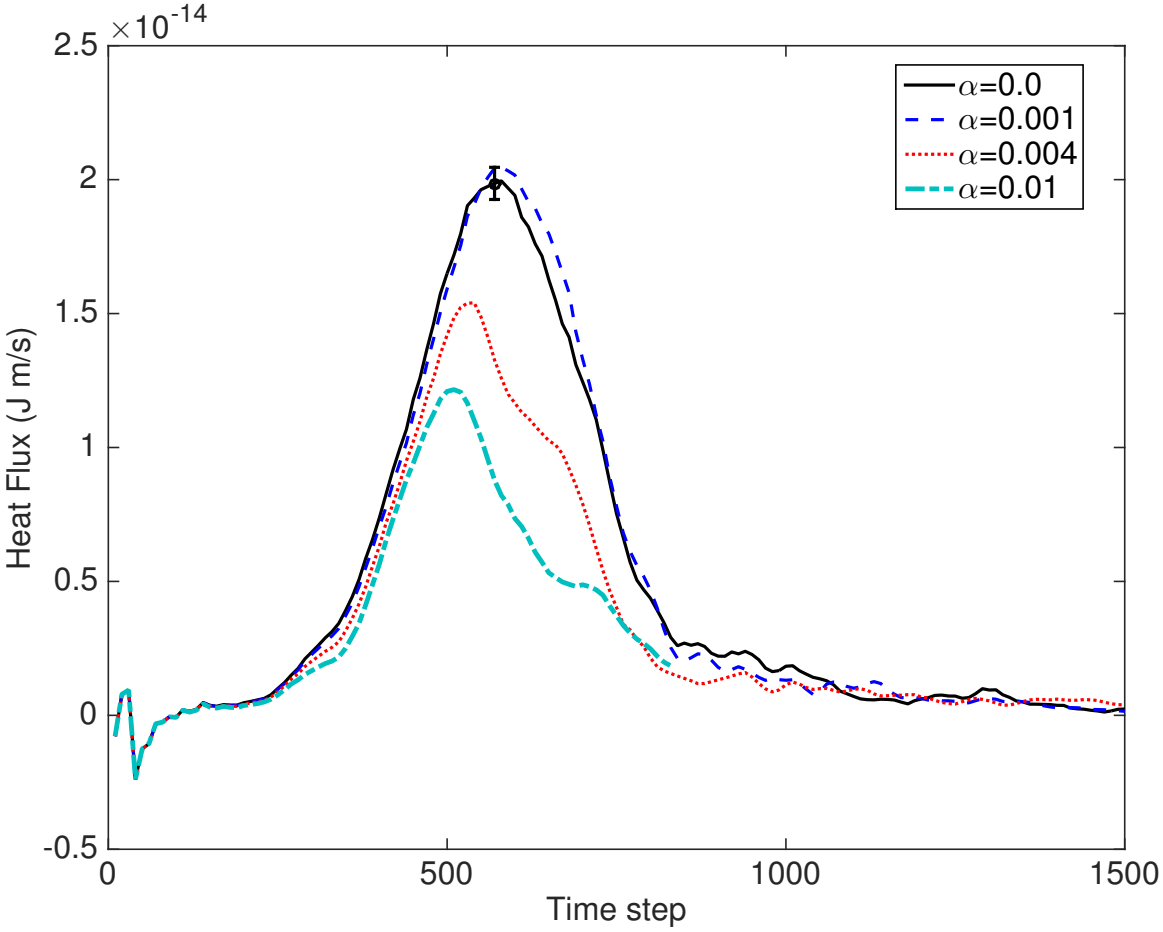

Figure 1. Heat flux per particle from the XGC1 ITG turbulence simulations with $\alpha = 0$, 0.001, 0.004 and 0.01. Standard $1/\sqrt{N_g}$ error bar is shown together. When $\alpha = 0.004$ and $\alpha = 0.01$, the reduction of the heat flux and the turbulence is observed.

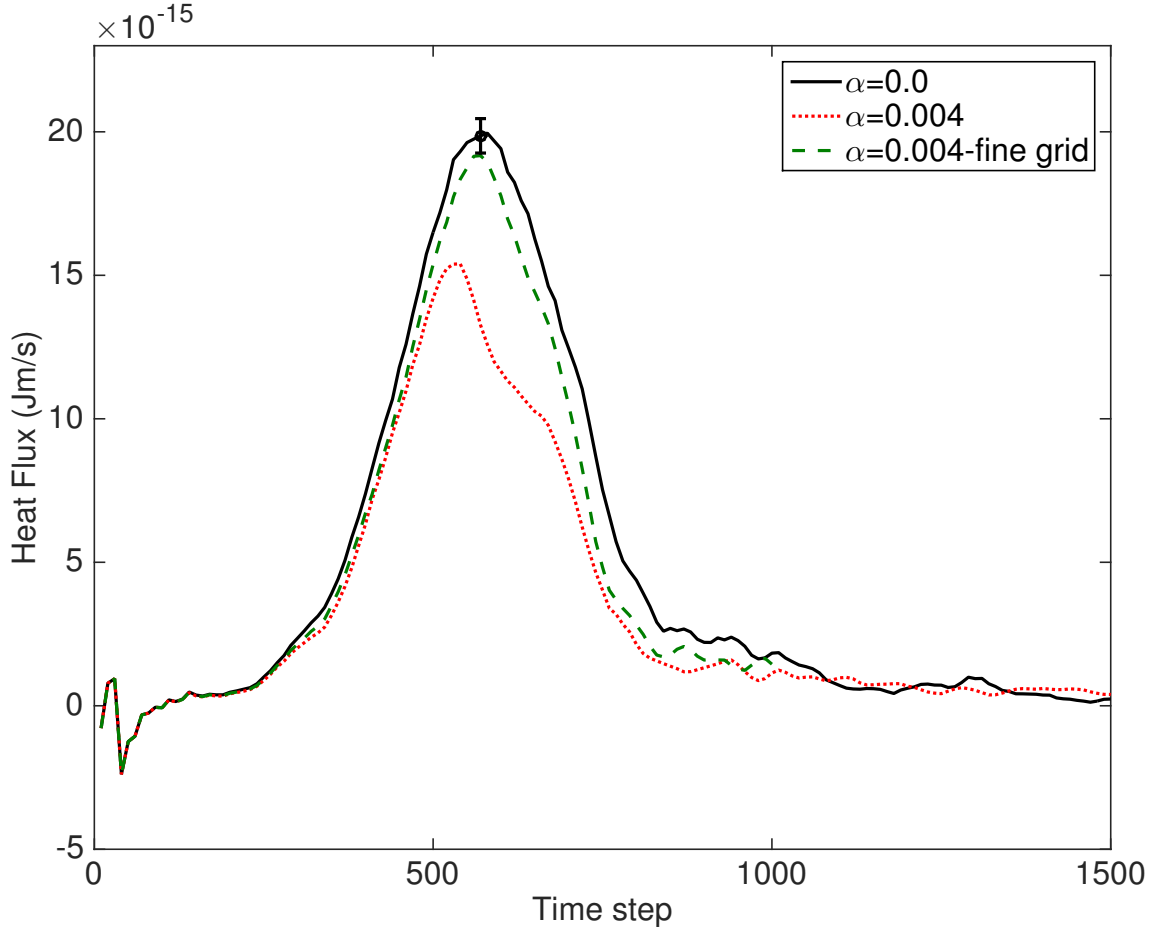

Figure 2. Heat flux per particle from the XGC1 ITG turbulence simulations with $\alpha = 0$, $\alpha = 0.004$ on the coarse (32×31), and $\alpha = 0.004$ on the finer grid (62×61). Standard $1/\sqrt{N_g}$ error bar is shown together. Finer grid restores the heat flux close to the $\alpha = 0$ case.

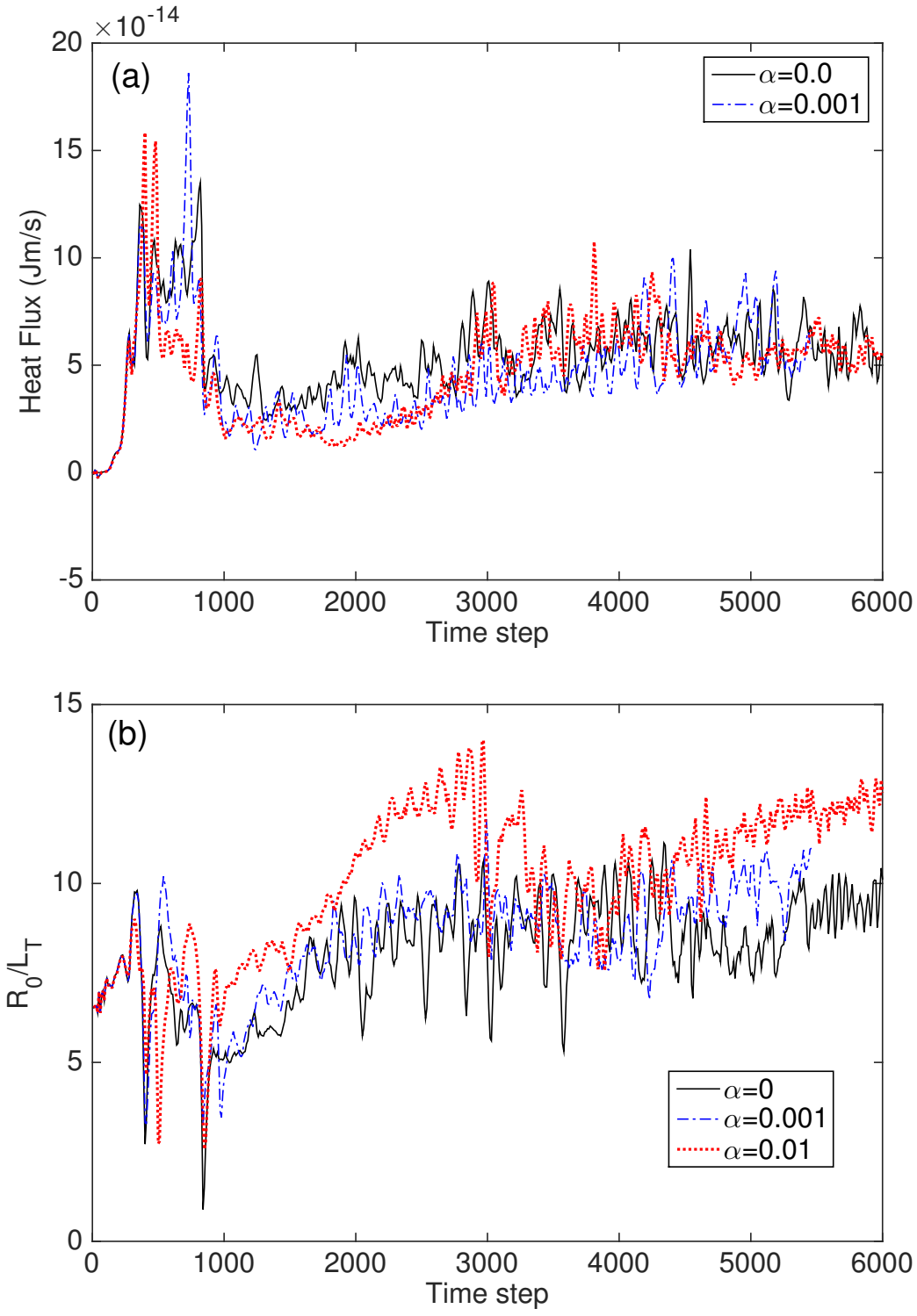

Figure 3. The heat flux per particle (above, a) and the temperature gradient scale length(below, b) from the XGC1 flux-driven ITG turbulence simulations with $\alpha = 0$, 0.001, and 0.01. When $\alpha = 0.001$ the gradient scale length converges to that of $\alpha = 0$. $\alpha = 0.01$ gives higher R_0/L_T than that of $\alpha = 0$ due to numerical reduction of turbulence. We note here that the heat flux always self-organize with the heating source in a flux-driven simulation, as shown in (a), by adjusting the temperature gradient.

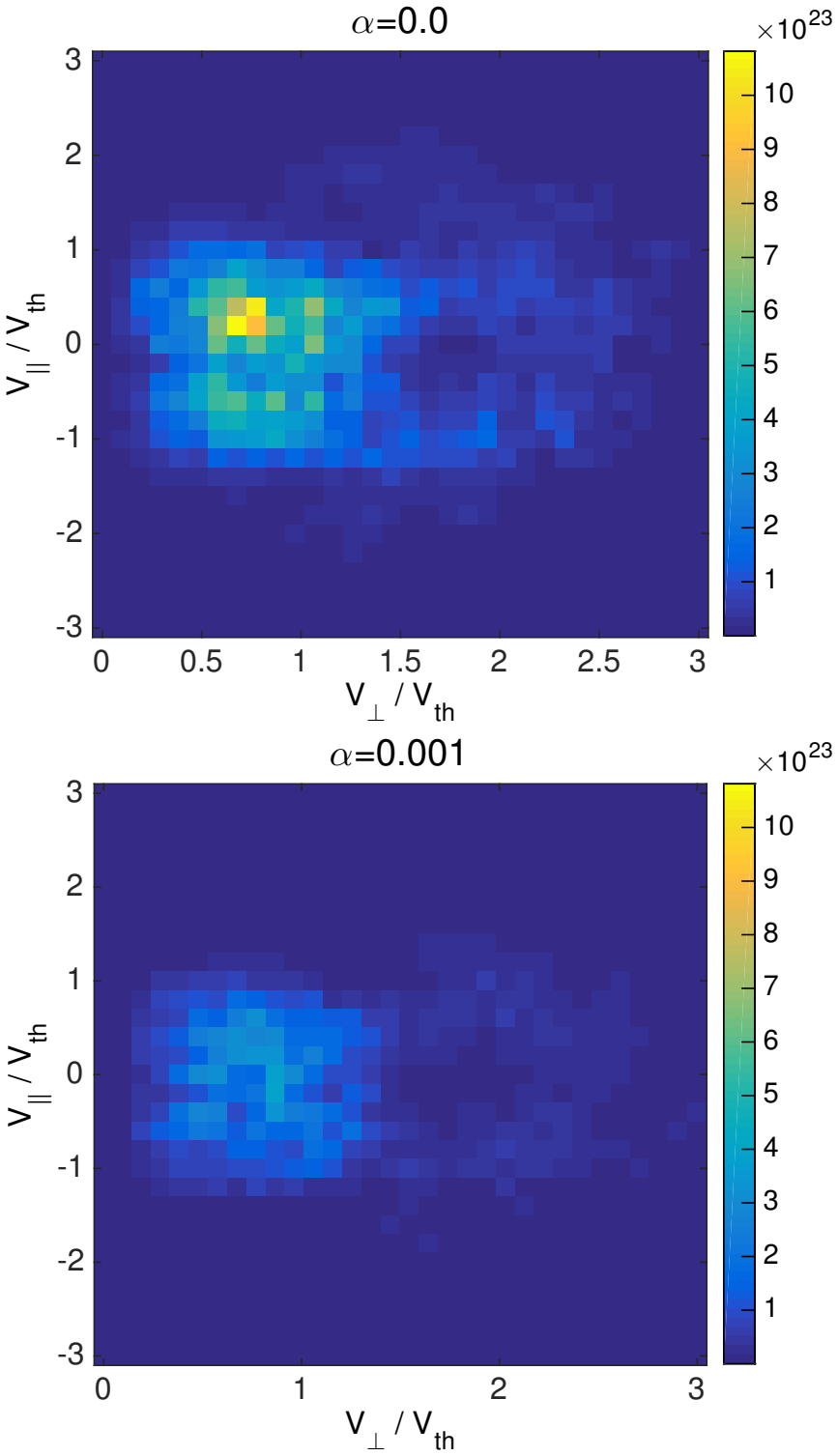

Figure 4. mean of $(w_1 w_0 g)^2$ of particles obtained on the velocity space grid with $\alpha = 0$ (above) and $\alpha = 0.001$ after 1500 time steps (below). Factor of approximately 4 reduction is observed with $\alpha = 0.001$ compared to $\alpha = 0$.

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Princeton Plasma Physics Laboratory
Office of Reports and Publications

Managed by
Princeton University

under contract with the
U.S. Department of Energy
(DE-AC02-09CH11466)

P.O. Box 451, Princeton, NJ 08543
Phone: 609-243-2245
Fax: 609-243-2751

E-mail: publications@pppl.gov

Website: http://www.pppl.gov