DIRECT INTEGRATION OF THE COLLISIONLESS BOLTZMANN EQUATION IN SIX-DIMENSIONAL PHASE SPACE: SELF-GRAVITATING SYSTEMS

Kohji Yoshikawa[1], Naoki Yoshida[2,3], and Masayuki Umemura[1]

[1] Center for Computational Sciences, University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki 305–8577, Japan; kohji@ccs.tsukuba.ac.jp
[2] Department of Physics, The University of Tokyo, Tokyo 113-0033, Japan
[3] Kavli Institute for the Physics and Mathematics of the Universe, The University of Tokyo, Kashiwa, Chiba 277-8583, Japan

ABSTRACT

We present a scheme for numerical simulations of collisionless self-gravitating systems which directly integrates the Vlasov–Poisson equations in six-dimensional phase space. Using the results from a suite of large-scale numerical simulations, we demonstrate that the present scheme can simulate collisionless self-gravitating systems properly. The integration scheme is based on the positive flux conservation method recently developed in plasma physics. We test the accuracy of our code by performing several test calculations, including the stability of King spheres, the gravitational instability, and the Landau damping. We show that the mass and the energy are accurately conserved for all the test cases we study. The results are in good agreement with linear theory predictions and/or analytic solutions. The distribution function keeps the property of positivity and remains non-oscillatory. The largest simulations are run on 64^6 grids. The computation speed scales well with the number of processors, and thus our code performs efficiently on massively parallel supercomputers.

Key words: galaxies: kinematics and dynamics – methods: numerical

Online-only material: color figures

1. INTRODUCTION

Gravitational interaction is one of the most important physical processes in the dynamics and the formation of astrophysical objects, such as star clusters, galaxies, and the large-scale structure of the universe. Stars and dark matter in these self-gravitating systems are essentially collisionless, except for a few cases, such as globular clusters and stars around supermassive black holes. The dynamics of the collisionless systems is described by the collisionless Boltzmann equation or the Vlasov equation.

Conventionally, gravitational N-body simulations are used to follow the evolution of collisionless systems. In such simulations, particles represent sampled points of the distribution function in the phase space. The particles—point masses—interact gravitationally with other particles, through which their orbits are determined. They are actually superparticles of stars or dark matter particles. The gravitational potential field reproduced in an N-body simulation is therefore intrinsically grainy rather than what it should be in the real physical system. It is well known that two-body encounters can alter the distribution function in a way that violates the collisionless feature of the systems, and undesired artificial two-body relaxation is often seen in N-body simulations. There is another inherent problem in N-body simulations. Gravitational softening needs to be introduced to avoid artificial large-angle scattering of particles caused by close encounters. Physical quantities such as mass density and velocity field are subject to intrinsic random noise owing to the finite number of particles especially in low-density regions.

To overcome these shortcomings of the N-body simulations, several alternative approaches have been explored. For example, the self-consistent field (SCF) method (Hernquist & Ostriker 1992; Hozumi 1997) integrates orbits of particles under the gravitational field calculated by expanding the density and the gravitational potential into a set of basis functions. In the SCF method, the particles do not directly interact with one another but move on the smooth gravitational potential calculated from the overall distribution of the particles. Despite of these attractive features, the major disadvantage of the SCF method is its inflexibility that the basis set must be chosen so that the lowest order terms reproduce the global structure of the systems under investigation (Weinberg 1999). In other words, the SCF method can be applied only to the symmetric gravitational collapse or the secular evolution of the collisionless systems.

The ultimate approach for numerical simulations of the collisionless self-gravitating systems would be direct integration of the collisionless Boltzmann equation, or Vlasov equation, combined with the Poisson equation. The advantage of the Vlasov–Poisson simulations was previously shown by Janin (1971) and Cuperman et al. (1971), who studied one-dimensional violent relaxation problems using the waterbag method (Hohl & Feix 1967; Roberts & Berk 1967). Fujiwara (1981, 1983), for the first time, successfully solved the Vlasov–Poisson equations for one-dimensional and spherically symmetric systems using the finite volume method. Other grid-based approaches include the seminal splitting method of Cheng & Knorr (1976), more generally the semi-Lagrangean methods (Sonnendrücker 1998), a finite element method (Zaki et al. 1988), a finite volume method (Filbet et al. 2001), the spectral method (Klimas 1987; Klimas & Farrell 1994), and a more recent multi-moment method (Minoshima et al. 2011). A comparison study of some of these methods is presented in Filbet & Sonnendrücker (2003).

So far, such direct integration of the Vlasov equation has been applied only to problems in one or two spatial dimensions. Solving the Vlasov equation in six-dimensional phase space requires an extremely large memory and computational time. However, the rapid development of massively parallel supercomputers has made it possible to simulate collisionless self-gravitating systems in the full six-dimensional phase space by numerically integrating the Vlasov–Poisson equations with a scientifically meaningful resolution.

In this paper, we present the results from a suite of large simulations of collisionless self-gravitating systems. To this end, we develop a fully parallelized Vlasov–Poisson solver. We perform an array of test calculations to examine the accuracy of our simulation code. We compare the obtained results with analytic solutions as well as linear theory predictions. We discuss the advantage and disadvantage of the Vlasov–Poisson approach over the conventional *N*-body method.

The rest of the paper is organized as follows. Section 2 is devoted to describing the detailed implementation of our numerical code to directly integrate the Vlasov–Poisson equations. In Section 3, we present the results of several test runs and their comparison with those obtained with the *N*-body method. The CPU timing and the parallelization efficiency are presented in Section 4. Finally, in Section 5, we summarize our results.

2. NUMERICAL SCHEME

For a collisionless self-gravitating system, the distribution function of matter $f(\boldsymbol{x}, \boldsymbol{v}, t)$ obeys the Vlasov–Poisson equations

$$\frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \frac{\partial f}{\partial \boldsymbol{x}} - \frac{\partial \phi}{\partial \boldsymbol{x}} \cdot \frac{\partial f}{\partial \boldsymbol{v}} = 0, \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{v}$ are the spatial and velocity coordinates, and ϕ is the gravitational potential satisfying the Poisson equation

$$\nabla^2 \phi = 4\pi G \rho = 4\pi G \int f \, d^3 \boldsymbol{v}. \tag{2}$$

We normalize the distribution function so that its integration over the entire velocity space yields the mass density.

In order to numerically compute Equations (1) and (2) simultaneously using the finite volume method, we configure $N_x \times N_y \times N_z$ uniformly spaced Cartesian grids (the spatial grids) in a simulation volume defined in $-L_x/2 < x < L_x/2$, $-L_y/2 < y < L_y/2$, and $-L_z/2 < z < L_z/2$. We also configure $N_x^{\rm v} \times N_y^{\rm v} \times N_z^{\rm v}$ uniform Cartesian grids (the velocity grids) in the velocity space with $V_x^- < v_x < V_x^+$, $V_y^- < v_y < V_y^+$, and $V_z^- < v_z < V_z^+$ at each spatial grid. Thus, the grid spacings are given by

$$\Delta x = \frac{L_x}{N_x}, \quad \Delta y = \frac{L_y}{N_y}, \quad \Delta z = \frac{L_z}{N_z} \tag{3}$$

and

$$\Delta v_x = \frac{V_x^+ - V_x^-}{N_x^{\rm v}}, \quad \Delta v_y = \frac{V_y^+ - V_y^-}{N_y^{\rm v}}, \quad \Delta v_z = \frac{V_z^+ - V_z^-}{N_z^{\rm v}} \tag{4}$$

for the spatial and velocity grids, respectively.

2.1. Vlasov Solver

We adopt the time splitting (operator splitting) scheme proposed by Cheng & Knorr (1976), which is essentially a divide-and-conquer strategy for solving multi-dimensional partial-differential equations (PDEs) by decomposing PDEs into simpler subproblems. The Vlasov equation is split into one-dimensional advection equations for each dimension of the phase space. Practically, we solve the following six one-dimensional advection equations sequentially; three for the advection in the position space

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} = 0 \tag{5}$$

$$\frac{\partial f}{\partial t} + v_y \frac{\partial f}{\partial y} = 0 \tag{6}$$

$$\frac{\partial f}{\partial t} + v_z \frac{\partial f}{\partial z} = 0 \tag{7}$$

and the remaining three equations in the velocity space

$$\frac{\partial f}{\partial t} - \frac{\partial \phi}{\partial x} \frac{\partial f}{\partial v_x} = 0 \tag{8}$$

$$\frac{\partial f}{\partial t} - \frac{\partial \phi}{\partial y} \frac{\partial f}{\partial v_y} = 0 \tag{9}$$

$$\frac{\partial f}{\partial t} - \frac{\partial \phi}{\partial z} \frac{\partial f}{\partial v_z} = 0. \tag{10}$$

A number of schemes are available to solve the advection equations on regular grids, such as the semi-Lagrange scheme (Cheng & Knorr 1976; Sonnendrücker 1998) and the spectral method (Klimas 1987; Klimas & Farrell 1994). An important property of the Vlasov equation is the conservation of the phase space density of matter, which leads to the conservation of mass in the system. Therefore, it is quite natural to adopt a manifestly conservative scheme. The positivity of the phase space density also has to be ensured. In this paper, we adopt the positive flux conservation (PFC) scheme proposed by Filbet et al. (2001) for the time evolution of the advection equation. The PFC scheme, by construction, ensures the conservation of the mass, the preservation of the positivity, and the maximum principle.

Here, we describe the PFC scheme briefly. Let us consider discretizing the following one-dimensional advection equation:

$$\frac{\partial f(x, t)}{\partial t} + u \frac{\partial f(x, t)}{\partial x} = 0. \tag{11}$$

Let f_i^n be the averaged value of the distribution function at a spatial grid with the central value of x_i and the interval of Δx such that

$$f_i^n = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} f(x, t^n) \, dx, \tag{12}$$

where $x_{i\pm1/2} \equiv x_i \pm \Delta x/2$ are the boundaries of the grid. Suppose that the values of the distribution function f_i^n at a time of $t^n = n\Delta t$ are known for all the spatial grid points. The conservation of the phase space density leads to

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f(x, t^{n+1}) \, dx = \int_{X(t^n, t^{n+1}, x_{i-1/2})}^{X(t^n, t^{n+1}, x_{i+1/2})} f(x, t^n) \, dx, \tag{13}$$

where $t^{n+1} = t^n + \Delta t$ and $X(t_1, t_2, x)$ is the value of the x-coordinate of the characteristic curve at a time of $t = t_1$ originating from the phase space coordinate (x, t_2). By denoting

$$\Phi^+ = \frac{1}{\Delta x} \int_{X(t^n, t^{n+1}, x_{i+1/2})}^{x_{i+1/2}} f(x, t^n) \, dx, \tag{14}$$

and

$$\Phi^- = \frac{1}{\Delta x} \int_{X(t^n, t^{n+1}, x_{i-1/2})}^{x_{i-1/2}} f(x, t^n) \, dx, \tag{15}$$

Equation (13) can be rewritten as

$$f_i^{n+1} = f_i^n + \Phi^- - \Phi^+. \tag{16}$$

We compute Φ^+ and Φ^- by interpolating the values of the distribution function at the grid points. Specifically, we adopt the third-order approximation of $f(x, t^n)$ with a slope corrector to suppress artificial numerical oscillations (Filbet et al. 2001). A detailed description of Φ^+ and Φ^- is given in the Appendix. As for the boundary condition in solving the one-dimensional advection equations, the outflow boundary condition is implemented in the velocity space. Thus, when the matter is accelerated beyond the predefined velocity limit $V_{x,y,z}^{\pm}$, it is regarded as vanished. In the position space, both the periodic and outflow boundary conditions are available depending on problems.

Using the PFC scheme for the numerical integration of one-dimensional advection equations, we advance of the distribution function from $f(x, v, t^n)$ to $f(x, v, t^{n+1})$ by sequentially updating each one-dimensional advection equation as

$$\begin{aligned} f(x, v, t^{n+1}) = \ & T_{v_z}(\Delta t/2) T_{v_y}(\Delta t/2) T_{v_x}(\Delta t/2) \\ & \times T_x(\Delta t) T_y(\Delta t) T_z(\Delta t) T_{v_z}(\Delta t/2) T_{v_y}(\Delta t/2) \\ & \times T_{v_x}(\Delta t/2) f(x, v, t^n), \end{aligned} \quad (17)$$

where $T_l(\Delta t)$ denotes the numerical advection operator along the l-direction for a time step of Δt. Here, we solve the Poisson equation after operating the advection equations in the position space. This time integration scheme is equivalent to the second-order leapfrog scheme.

2.2. Poisson Solver

The gravitational potential ϕ is computed under the periodic boundary conditions or the isolated boundary conditions. For a given distribution function $f(x, v, t)$, the mass density ρ at a spatial grid point x is obtained simply by integrating the distribution function over the velocity space,

$$\rho(x) = \int f(x, v, t) \, d^3 v. \quad (18)$$

We adopt the convolution method with the Fourier transform (Hockney & Eastwood 1981) to numerically solve the Poisson equation.

For the periodic boundary conditions, we first compute the discrete Fourier transform (DFT) of the density $\hat{\rho}(k)$ using the fast Fourier transform (FFT), where $k = (k_x, k_y, k_z)$ is a wavenumber vector. Then, the Fourier-transformed gravitational potential is given by

$$\hat{\phi}(k) = \hat{G}(k) \hat{\rho}(k), \quad (19)$$

where $\hat{G}(k)$ is the DFT of the green function of the Poisson equation discretized with the second-order central difference scheme. For $\Delta x = \Delta y = \Delta z = \Delta$, it is given by

$$G(k) = -\frac{\pi G \Delta^2}{\sin^2(k_x \Delta/2) + \sin^2(k_y \Delta/2) + \sin^2(k_z \Delta/2)}. \quad (20)$$

Finally, the inverse FFT of $\hat{\phi}(k)$ yields the gravitational potential $\phi(x)$ in the real space.

As for the isolated boundary condition, we adopt the doubling up method (Hockney & Eastwood 1981), in which the number of the spatial grid points is doubled for all coordinate axes, and the mass densities in the extended grid points are set to zero. The Green function is constructed as follows. First, it is defined at $N_x \times N_y \times N_z$ grid points in real space as

$$G(x, y, z) = \frac{G}{(x^2 + y^2 + z^2)^{1/2}} \quad (21)$$

for $0 \leqslant x \leqslant L_x$, $0 \leqslant y \leqslant L_y$, $0 \leqslant z \leqslant L_z$. By duplicating and mirroring it in the extended grid points, we obtain the Green function periodic in the $2N_x \times 2N_y \times 2N_z$ grid points. After computing the Fourier transform of the Green function $\hat{G}(k)$, the gravitational potential in the real space $\phi(x)$ is obtained in the same manner as in the periodic boundary condition.

In order to calculate the gravitational force at each spatial grid point, we adopt the two-point finite-difference scheme, in which the gradient of the gravitational potential is calculated as

$$\left(\frac{\phi_{i+1,j,k} - \phi_{i-1,j,k}}{2\Delta x}, \frac{\phi_{i,j+1,k} - \phi_{i,j-1,k}}{2\Delta y}, \frac{\phi_{i,j,k+1} - \phi_{i,j,k-1}}{2\Delta z} \right), \quad (22)$$

where $\phi_{i,j,k}$ is the gravitational potential at a spatial grid point with indices of (i, j, k).

2.3. Parallelization

The computational cost for the time integration of the Vlasov equation and the required amount of memory to store the distribution function in the phase space roughly scale proportional to $N_x N_y N_z \times N_x^v N_y^v N_z^v$. Hence, efficient parallelization is indispensable for the numerical integration of the Vlasov–Poisson equations.

To parallelize our Vlasov–Poisson solver, we decompose the computational domain in the phase space as follows. The position space is divided along each spatial axes into subdomains, while the velocity space at a given spatial position is not decomposed. In this way, we can achieve an equal balance in memory on distributed memory computers. We use the Message Passing Interface (MPI) for the inter-node parallelization; each MPI process operates on a decomposed phase space. We also use the OpenMP implementation to utilize the multi-thread parallelization on multi CPU cores in individual computational nodes. To solve the advection equations along the spatial coordinate (x-, y-, and z-coordinates) on each MPI process, the values of the distribution function at the adjacent spatial grid points are exchanged between the computational nodes. In solving the Poisson equation, we do not parallelize the FFT because the required computational cost of the FFT is nearly negligible compared with other portions of the calculations and also because the size of FFT ($N_x N_y N_z$) is not large enough for sufficient speed-up of the calculations.

2.4. Time Step

In solving the one-dimensional advection equation (11) using the PFC method described in Section 2, the time step width Δt is not restricted by the Courant–Friedrichs–Levy condition. However, when integrating the multi-dimensional Vlasov–Poisson equations, we need to constrain the time step under the following considerations: (1) the accuracy of the characteristic lines is better for the smaller Δt. (2) To integrate the Vlasov equation on a distributed memory system with phase space domain decomposition, the exchange of the distribution function at the boundaries of subdomains is unavoidable. If we set too large a time step width, the trajectories of the characteristic lines become more distant from the boundaries and then the number of grid points whose data should be sent to the adjacent subdomains becomes also larger, resulting in the increase of data exchange among the MPI processes.

We constrain the time step for integrating the Vlasov equation as
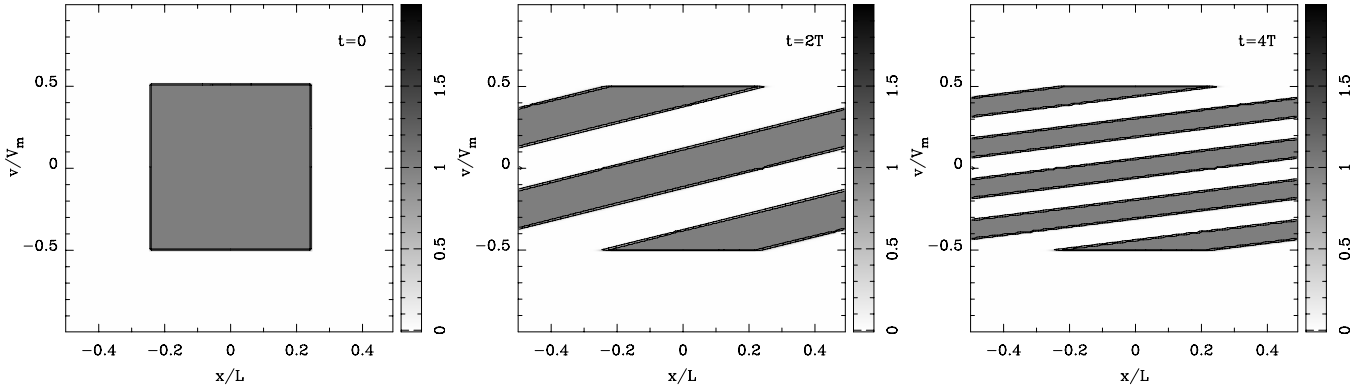
$$\Delta t = C \min(\Delta t_p, \Delta t_v), \quad (23)$$

Figure 1. Test 1: the phase space density of one-dimensional free-streaming matter at $t = 0$ (left), $2T$ (middle), and $4T$ (right). Black lines show the contours for $f(x, v, t) = 0.5$ and 1.0.

where C is a parameter to control the time step, and Δt_p and Δt_v is the time step constraints for the advection equations in position space (5)–(7) given by

$$\Delta t_p = \min\left(\frac{\Delta x}{V_x^{max}}, \frac{\Delta y}{V_y^{max}}, \frac{\Delta z}{V_z^{max}}\right) \qquad (24)$$

and for the ones in velocity space (8)–(10) given by

$$\Delta t_v = \min_i\left(\frac{\Delta v_x}{|a_{x,i}|}, \frac{\Delta v_y}{|a_{y,i}|}, \frac{\Delta v_z}{|a_{z,i}|}\right), \qquad (25)$$

where $a_{x,i}$, $a_{y,i}$, and $a_{z,i}$ are the x, y, and z-components of the gravitational acceleration, $\nabla\phi$, at the ith grid point, and the minimization is taken over all the spatial grids. In the following, we set $C = 0.5$ unless otherwise stated.

3. TEST CALCULATIONS

In this section, we present a series of Vlasov–Poisson simulations of self-gravitating systems using our newly developed parallel code.

3.1. Test 1: One-dimensional Advection

As a test of the PFC scheme to solve a one-dimensional advection equation, we perform simulations of one-dimensional freely streaming matter. This is the most trivial test, but it is indeed important to determine the positivity and non-oscillatory behavior of the distribution function. We solve the following Vlasov equation without the gravitational acceleration term:

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0. \qquad (26)$$

Here, we consider a two-dimensional phase space defined as

$$\begin{cases} -L/2 \leqslant x \leqslant L/2 \\ -V_m \leqslant v \leqslant V_m \end{cases}, \qquad (27)$$

where we impose the periodic boundary condition for the x-coordinate. The initial condition is given by

$$\begin{cases} f(x, v, t = 0) = 1 & -L/4 \leqslant x \leqslant L/4 \text{ and} \\ & -V_m/2 \leqslant v \leqslant V_m/2 \\ f(x, v, t = 0) = 0 & \text{otherwise}. \end{cases} \qquad (28)$$
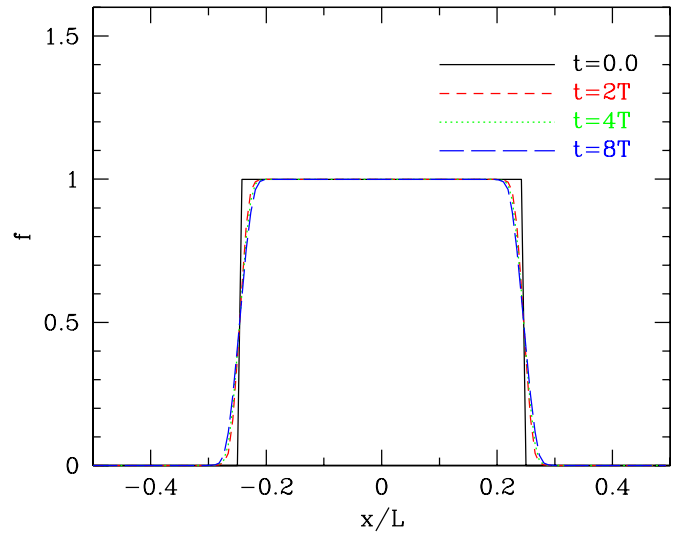

Figure 2. Test 1: profiles of the distribution function along $v = 0.5V_m$ at $t = 0.0$, $2T$, $4T$, and $8T$.

(A color version of this figure is available in the online journal.)

It is expected that for each velocity v the distribution function is translated with time at a speed of v and its shape with respect to x is preserved. The numbers of grids along x- and v-coordinates are both set to be 128.

Figure 1 shows the phase space density at $t = 0$, $2T$ and $4T$, where the system's unit T is defined as $T \equiv L/V$. The black lines show the contour for $f(x, v, t) = 0.5$ and 1.0. We clearly see that the sharp edge of the phase space density is well reproduced. We confirmed that there is no numerical oscillations around the sharp edge and also that the distribution function is always positive in the phase space.

Profiles of the distribution function with respect to x along $v = V_m/2$ at $t = 0, 2T, 4T$, and $8T$ are shown in Figure 2. Since we impose the periodic boundary condition in the x-direction, it is expected that the profile of the distribution function remains the same at $t = 0, 2T, 4T$, and $8T$ along $v = V_m/2$. Although the sharp edges around $x = \pm L/4$ are slightly smeared due to the phase error caused by numerical diffusion, the profiles at $t = 4T$ and $8T$ are almost the same. Numerical diffusion smears the distribution function only initially, but does not cause secular errors. Figure 3 shows the global and local errors of the numerically solved distribution function depicted in Figure 2 at
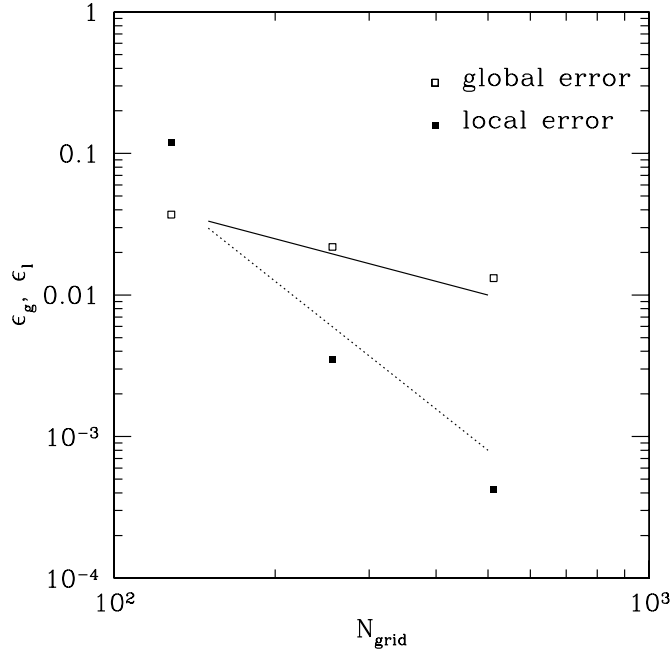
Figure 3. Test 1: the local and global errors as a function of grid size. Solid and dotted lines indicate the first- and third-order dependence, respectively.

$t = 8T$ estimated by

$$\epsilon_g = \frac{\sum_i |f_i - f(x_i, V_m/2, 8T)|\Delta x}{\int f(x, V_m/2, 8T)dx},$$ (29)

and

$$\epsilon_\ell = \frac{|f_i - f(x_i, V_m/2, 8T)|}{|f(x_i, V_m/2, 8T)|},$$ (30)

respectively, where f_i is the value of the distribution function numerically solved at $t = 8T$, and the local errors are estimated near the sharp edge at $x = -L/4$. The global error is proportional to the grid spacing Δx and is dominated by the error that originates from the adoption of the slope correctors as expected by Equation (A15). On the other hand, we observe that the local errors are roughly proportional to Δx^3, because the primitive of the distribution function for its interpolation is constructed using the third-order approximation. This is also consistent with the results shown in Minoshima et al. (2011).[4] The dependence of the global and local errors on the leading factor C of Equation (23) is shown in Figure 4, where it can be seen that both of the global and local errors are almost independent of C. This feature is understood by the fact that the PFC scheme traces the boundaries of a grid mesh along the characteristic lines to update the advection equation. (See Equation (13).)

Figure 5 shows the relative errors of the kinetic energy $K(t)$ given by

$$K(t) = \frac{1}{2} \int \int f(x, v, t)v^2 \, dv dx,$$ (31)

and the total mass $M(t)$

$$M(t) = \int \int f(x, v, t) \, dv dx$$ (32)
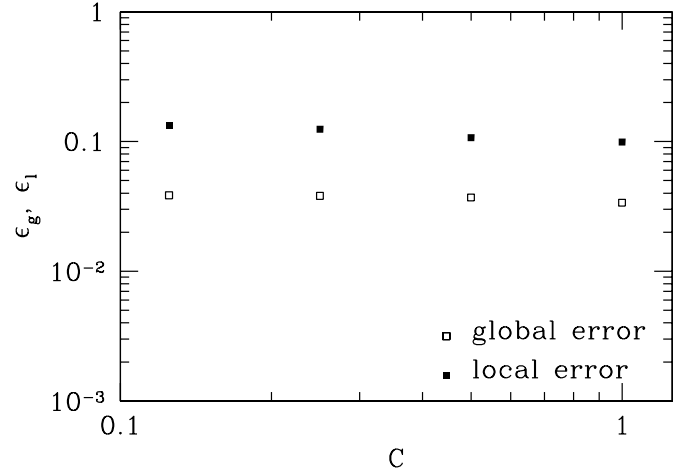
---

[4] See Figure 2 in their paper.

Figure 4. Test 1: the local and global errors at $t = 8T$ as a function of the leading-factor C in Equation (23).

during the calculation, manifesting that both of the kinetic energy and the mass are conserved within the accuracy of 10^{-5}. Note that the PFC scheme for the one-dimensional advection equation ensures the conservation of the mass, the zeroth-order velocity moment of the distribution function, but not the first- and second-order moment, and that the conservation of the latters mainly depends on the numerical resolution of the velocity space.

3.2. Test 2: One-dimensional Homogeneous Self-gravitating System

In this test, we simulate a one-dimensional infinite self-gravitating system following the Vlasov equation:

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} - \frac{\partial \phi}{\partial x}\frac{\partial f}{\partial v} = 0,$$ (33)

coupled with the Poisson equation

$$\nabla^2 \phi = 4\pi G(\rho - \bar{\rho}) = 4\pi G \left(\int_{-\infty}^{\infty} f \, dv - \bar{\rho} \right),$$ (34)

under the periodic boundary conditions in the x-direction, where $\bar{\rho}$ is the mean mass density. We consider a Maxwellian system with a periodic density fluctuation. The initial distribution function is set to be

$$f(x, v, t = 0) = \frac{\bar{\rho}}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{v^2}{2\sigma^2} \right)(1 + A\cos kx),$$ (35)

where σ is the velocity dispersion and A is the amplitude of the density fluctuation. In this system, when the wavenumber of the density fluctuation k is smaller than the critical Jeans wavenumber k_J given by

$$k_J = \left(\frac{4\pi G\bar{\rho}}{\sigma^2} \right)^{1/2},$$ (36)

the density fluctuation grows through the Jeans instability. On the other hand, when $k > k_J$, the density fluctuation damps through the collisionless damping, or the Landau damping.
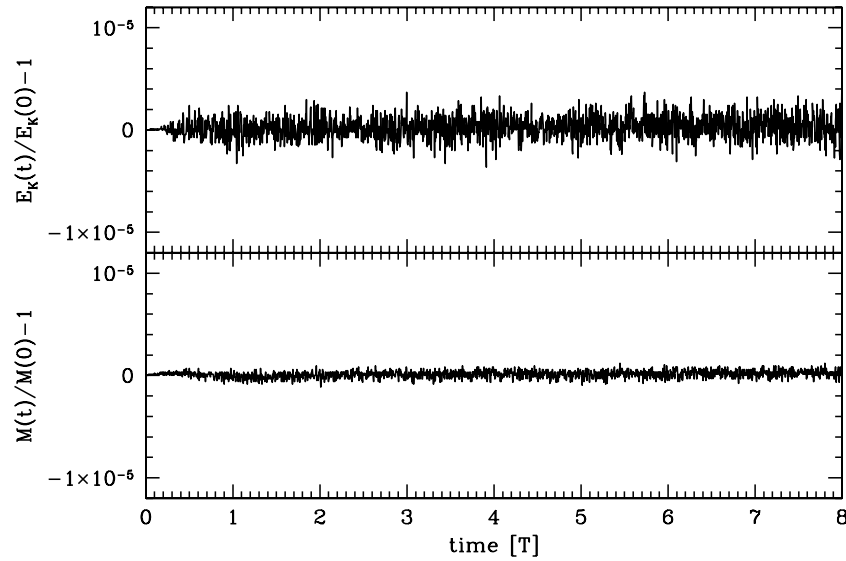
Figure 5. Test 1: the relative errors of the kinetic energy (upper panel) and the mass (lower panel).
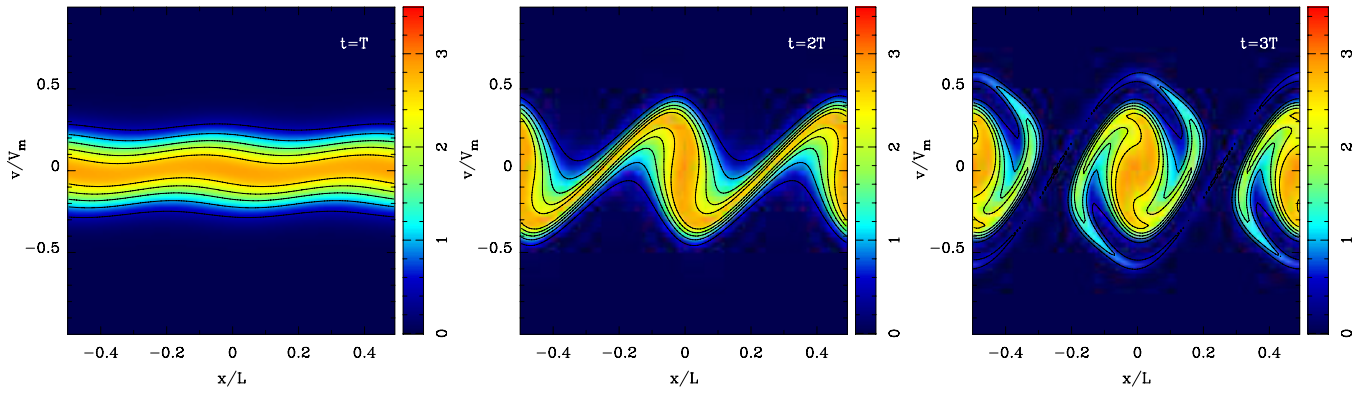

Figure 6. Test 2: phase space density in the run with $k/k_J = 0.5$ at $t = 1.0\,T$ (left), $2.0\,T$ (middle), and $3.0\,T$ (right).
(A color version of this figure is available in the online journal.)

The computational domain of the two-dimensional phase space is set to be

$$
\begin{cases}
-L/2 \leqslant x \leqslant L/2 \\
-V \leqslant v \leqslant V
\end{cases},
\tag{37}
$$

where V is defined as $V = L/T$ and T is the dynamical time defined by

$$
T = (G\bar{\rho})^{-1/2}.
\tag{38}
$$

The number of grid points is 128 in both x- and v-directions unless otherwise stated.

Since we impose the periodic boundary conditions, the wavenumber must be set to $k = nk_0$, where $k_0 = 2\pi/L$ and n is a positive integer, and the velocity dispersion σ is determined such that the ratio k/k_J is adjusted to have a specific value. In the following, the wavenumber is fixed to $k = 2k_0$ ($n = 2$) unless otherwise stated. We show the results for $k/k_J = 0.1, 0.5, 1.1$, and 2.0. The amplitude of the initial density perturbation A is set to $A = 0.1$ for $k/k_J > 1$ and $A = 0.01$ for $k/k_J < 1$. Figure 6 shows the phase space density for the case with $k/k_J = 0.5$ at $t = T, 2\,T$, and $3\,T$. In this case, as expected, the density fluctuation grows monotonically, and collapsed objects are formed through the gravitational instability. Contrastingly, the density fluctuation is damped through the Landau damping in the run with $k/k_J = 1.1$, as can be seen in Figure 7.

Figure 8 shows the time evolution of the amplitude of the density fluctuation $\delta \equiv (\rho - \bar{\rho})/\bar{\rho}$ for $k/k_J = 0.1, 0.5, 1.1$, and 2.0, where the amplitude is expressed in terms of the Fourier amplitude A_n which is given by

$$
\delta(x, t) = \sum_{m \geqslant 0} A_m(t) \exp(imk_0 x).
\tag{39}
$$

The time evolution of $|A_2(t)/A_2(0)|$ is plotted in Figure 8. We also check the convergence of the solution by doubling the resolution in the velocity space. The results from the runs with $N_x^v = 64$ and $N_x^v = 128$ are also compared in Figure 8.

The linear growth (or damping) rate γ can be computed using the dispersion relation

$$
\frac{k^2}{k_J^2} = 1 + wZ(w),
\tag{40}
$$

where $Z(w)$ is the plasma dispersion function

$$
Z(w) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} ds \, \frac{e^{-s^2}}{s - w}
\tag{41}
$$

and w is given by

$$
w = \frac{\pm i\gamma}{\sqrt{8\pi G\bar{\rho}}(k/k_J)}.
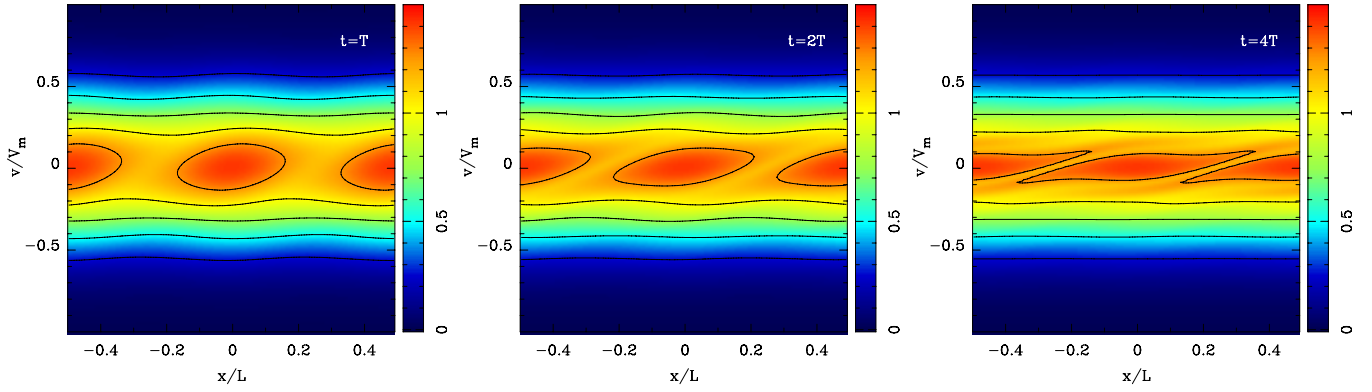\tag{42}
$$

Figure 7. Test 2: phase space density in the run with $k/k_J = 1.1$ at $t = 1.0\,T$ (left), $2.0\,T$ (middle), and $4.0\,T$ (right).
(A color version of this figure is available in the online journal.)
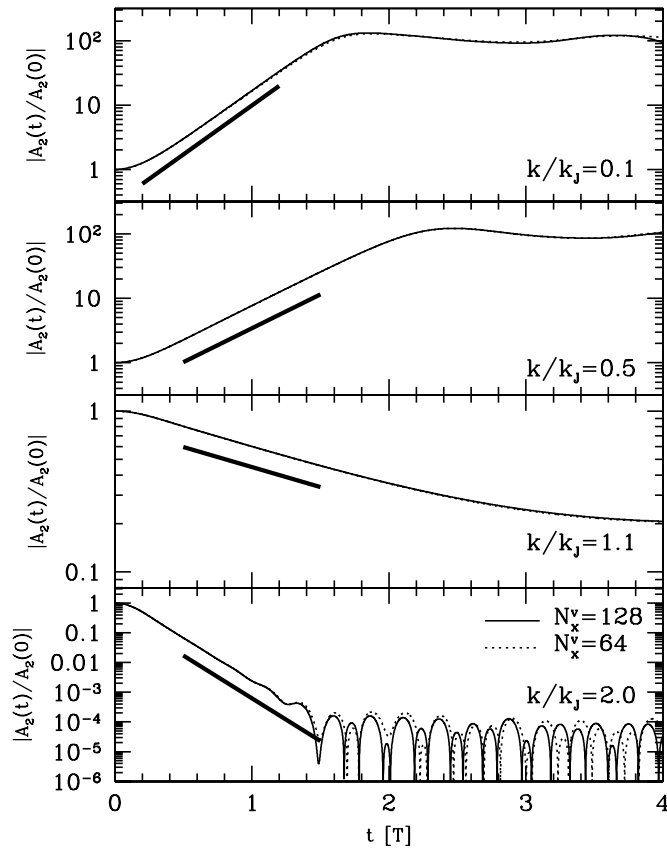

Figure 8. Test 2: time evolution of the density contrast at the density peak for the runs with $k/k_J = 0.1$, 0.5, 1.1, and 2.0 (from top to bottom). The solid and dotted lines indicate the results with $N_x^v = 128$ and 64, respectively. The bold lines show the linear damping rate (see the text).

A more detailed description on the growth rate and the dispersion relation can be found in Binney & Tremaine (2008). For a given value of k/k_J, the growth and damping rates can be computed by solving Equation (40). The bold line in each panel in Figure 8 indicates the theoretical linear growth or damping rate γ. Our numerical results are in good agreement with the linear theory prediction in the early phase. It is interesting that the growth of the perturbation saturates at $T > 1$ in the run with $k/k_J < 1$. For $k/k_J = 2.0$, the timescale of the damping is shorter than the dynamical timescale. A significant fraction of the mass is trapped in the trough of the gravitational potential, with the distribution function being peaked around $v = 0$. Since

such a distribution function with a small velocity dispersion cannot damp the density fluctuation efficiently via the Landau damping, density fluctuations begin oscillating after the early linear damping phase. Similarly, fluctuation damping saturates at $t \gtrsim 3T$ for $k/k_J = 1.1$. Figure 7 shows that the phase space density in the run with $k/k_J = 1.1$ departs from the initial Gaussian distribution. The distribution function is more concentrated around $v = 0$ at later times. These features are well known as the nonlinear Landau damping and also pointed out by Fujiwara (1981).

It should be also pointed out that there is no significant difference between the results with $N_x^v = 64$ and 128. Since Equation (36) can be modified as

$$\sigma = \frac{4\pi G \bar{\rho}}{k} \frac{k}{k_J},\qquad(43)$$

for a fixed wavenumber k or n, the initial velocity dispersion is proportional to k/k_J. In the case of $k/k_J = 0.1$, $\sigma/\Delta v_x = 1.8$ and 0.9 for $N_x^v = 128$ and 64, respectively. It can be said that calculations of the gravitational instability are not very sensitive to the resolution in the velocity space and can be solved fairly accurately even when the velocity resolution is not excellent.

To address the effect of velocity resolution in the Landau damping, we perform the simulations of the Landau damping with $k/k_J = 1.1$ for a fixed set of N_x and N_x^v and various wavenumbers k or n of the initial density fluctuation. For a given value of k/k_J, the initial velocity dispersion is proportional to k^{-1} or n^{-1} (see Equation (43)). Thus, the velocity dispersion is less resolved in the velocity space for a larger k or n. The results with $N_x = 128$ and $N_x^v = 64$ for various n are shown in the upper panel of Figure 9, in which the results significantly depend on $\sigma/\Delta v_x$ and the amplitudes of the density contrast for the runs with $n > 4$ initially damp more rapidly than those for $n = 2$ and $n = 4$ and, oscillate afterward. This suppression of the Landau damping takes place because the initial velocity dispersion is less resolved in the velocity space for a larger n, and because the velocity dispersion reduced through the Landau damping can no longer damp the density fluctuation when it gets comparable to or smaller than the velocity resolution Δv_x. Thus, these calculations indicates that the velocity dispersion is should be resolved with at least five velocity grids to properly simulate the Landau damping. We also perform the simulations of the Landau damping with $k/k_J = 1.1$ and $n = 2$, 4, and 8 for a fixed value of $\sigma/\Delta v_x = 9.6$. As can be seen in the lower panel of Figure 9, the damping rates with $n = 4$ and 8 are almost the
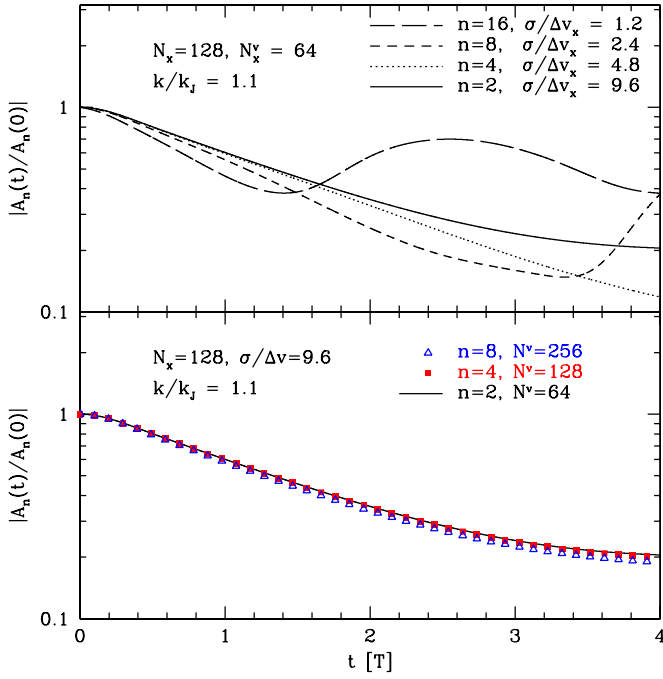
Figure 9. Test 2: time evolution of the amplitude of the density contrast for the Landau damping with $k/k_J = 1.1$. The results for various n with a fixed resolution of $N_x = 128$ and $N_x^v = 64$, and those with a fixed $\sigma/\Delta v_x$ are shown in the upper and lower panels, respectively.

(A color version of this figure is available in the online journal.)

same as that with $n = 2$ if $\sigma/\Delta v_x$ is kept constant, indicating that the suppression of the damping shown in the upper panel of Figure 9 is solely due to the relatively poor resolution in the velocity space.

The top panel of Figure 10 shows the time evolution of kinetic energy $K(t)$ given by Equation (31) and the gravitational potential energy $U(t)$ computed as

$$U(t) = \frac{1}{2} \int \rho(x)\phi(x)\,dx, \qquad (44)$$

as well as the total energy $E = K(t) + U(t)$. The middle and bottom panels indicate the relative errors in the total energy E and the total mass $M(t)$ given by Equation (32). The total energy and the mass are conserved within the relative errors of 10^{-3} and 10^{-6}, respectively.

3.3. Test 3: Galilean Invariance

It is well known that mesh-based hydrodynamical codes generally do not assure the Galilean invariance because approximate Riemann solvers employed in many of such codes are not manifestly Galilean invariant (Wadsley et al. 2008; Tasker et al. 2008). This is in good contrast with particle-based N-body simulations that are exactly Galilean invariant as long as a symmetric time integration scheme is used. In the light of this, it is interesting and important to examine the Galilean invariance of our mesh-based scheme for self-gravitating systems. We test our code by adding a constant translational velocity v_t to the initial conditions of the Test 2 problem. We compare the results with the original one (with $v_t = 0$) presented in the previous section.

Specifically, we set the initial distribution function as

$$f(x, v, t = 0) = \frac{\bar{\rho}}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{(v - v_t)^2}{2\sigma^2}\right)(1 + A\cos kx), \qquad (45)$$
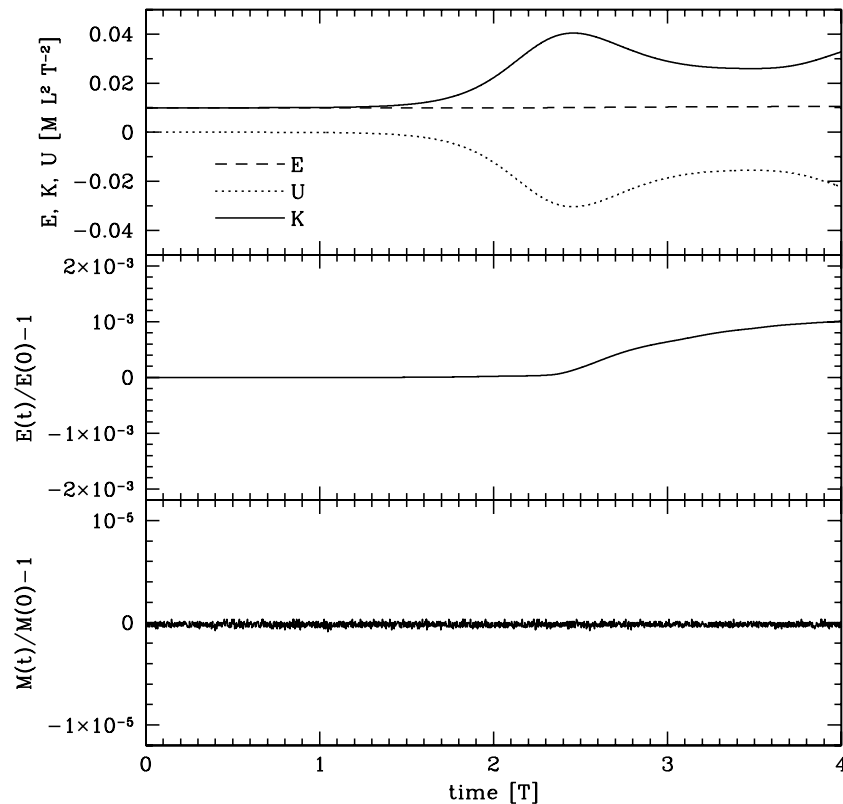

Figure 10. Test 2: the time evolutions of the kinetic, potential, and total energy in the run with $k/k_J = 0.5$ are shown in the top panel. The relative errors of the total energy and the mass conservation are depicted in the middle and bottom panels, respectively.
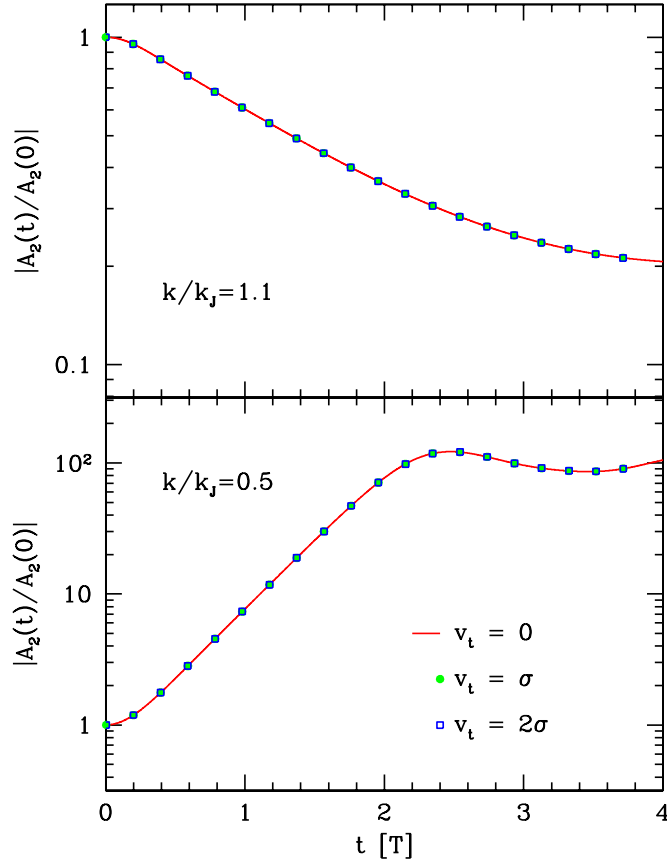
Figure 11. Test 3: Galilean invariance. We plot the time evolution of the amplitude of the density fluctuation of a mode of $n = 2$ with translational velocity of $v_t = 0$, σ, and 2σ. Upper and lower panels show the results for $k/k_J = 1.1$ and $k/k_J = 0.5$, respectively.

(A color version of this figure is available in the online journal.)

where the velocity dispersion is set such that $k/k_J = 0.5$ and $k/k_J = 1.1$ (see Equation (36)). We assign $v_t = \sigma$ and 2σ. The numbers of the grid points are set to $N_x = 128$ and $N_x^v = 128$. Figure 11 shows the comparison of the time evolution of the density fluctuation of a $n = 2$ mode with $v_t = \sigma$ and 2σ to

the original result with $v_t = 0$ presented in Test 2. Clearly, the results are independent of the translational velocity and hence our code is Galilean invariant to this accuracy.

3.4. Test 4: Three-dimensional Homogeneous Self-gravitating System

We study the gravitational instability and the Landau damping in a three-dimensional self-gravitating system. We solve the Vlasov equation coupled with the three-dimensional Poisson equation in six-dimensional phase space under the periodic boundary conditions for spatial coordinates. The run is configured as follows. At each spatial grid, the initial distribution function is given by

$$f(\boldsymbol{x}, \boldsymbol{v}, t = 0) = \frac{\bar{\rho}(1 + \delta_i(\boldsymbol{x}))}{(2\pi\sigma^2)^{3/2}} \exp\left(-\frac{|\boldsymbol{v}|^2}{2\sigma^2}\right), \quad (46)$$

where $\delta_i(\boldsymbol{x})$ is the initial density fluctuation at a spatial position $\boldsymbol{x}$. The density fluctuations are generated by assigning uniform random values between $-\delta_m/2$ and $\delta_m/2$ so that the resulting density field has a white noise power spectrum. We assign the velocity dispersion which is determined from a predefined Jeans wavenumber k_J.

The phase space volume with $-L/2 \leqslant x, y, z \leqslant L/2$ and $-V \leqslant v_x, v_y, v_z \leqslant V$ is discretized with $N_x = N_y = N_z = 64$ and $N_x^v = N_y^v = N_z^v = 64$, where V is again defined by $V = L/T$ and $T = (G\bar{\rho})^{-1/2}$. In the following, δ_m is set to 0.1, and we present the results for $k_J = 2\pi/(L/4) = 8\pi/L$ and $k_J = 2\pi/(L/8) = 16\pi/L$. Note that the velocity dispersions σ in the runs with $k_J = 8\pi/L$ and $k_J = 16\pi/L$ correspond to $9\Delta v_{x,y,z}$ and $4.5\Delta v_{x,y,z}$, respectively. To characterize the three-dimensional density fluctuations and their evolution, we compute the power spectrum $P(k) = \langle |\delta(\boldsymbol{k})|^2 \rangle$, where $\delta(\boldsymbol{k})$ is the DFT of $\delta(\boldsymbol{x})$ given by

$$\delta(\boldsymbol{x}) = \frac{1}{\bar{\rho}} \int f(\boldsymbol{x}, \boldsymbol{v}, t) \, d^3\boldsymbol{v} - 1. \quad (47)$$

Figure 12 shows the power spectra of the density field at $t = 0, 0.2T, 0.4T, 0.6T$, and $0.8T$ computed in the runs with
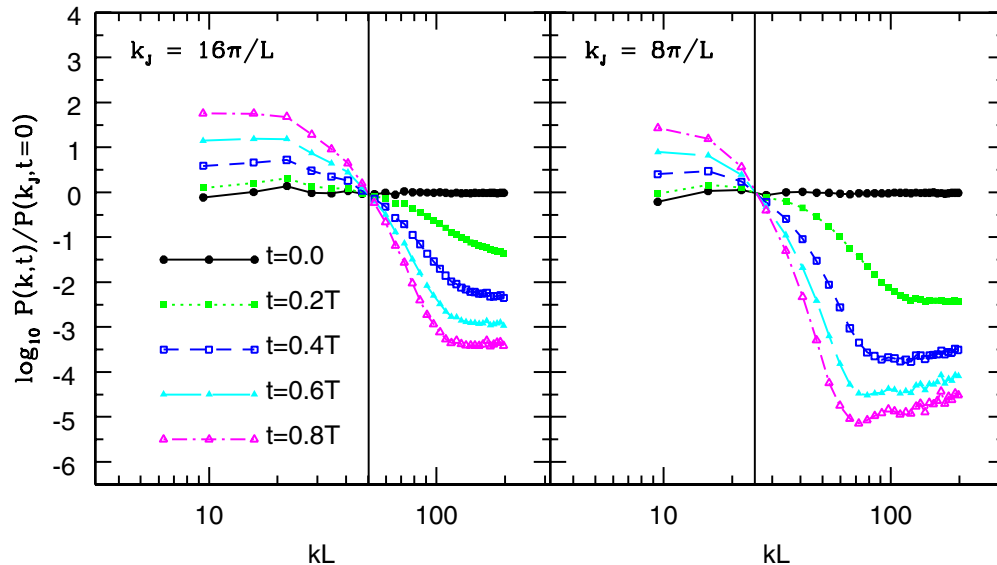

Figure 12. Test 4: power spectra of the density fluctuation at $t/T = 0.0, 0.2, 0.4, 0.6$, and 0.8 in the runs with $k_J = 8\pi/L$ (right) and $k_J = 16\pi/L$ (left). The vertical line in each panel indicates the location of the Jeans wavenumber.

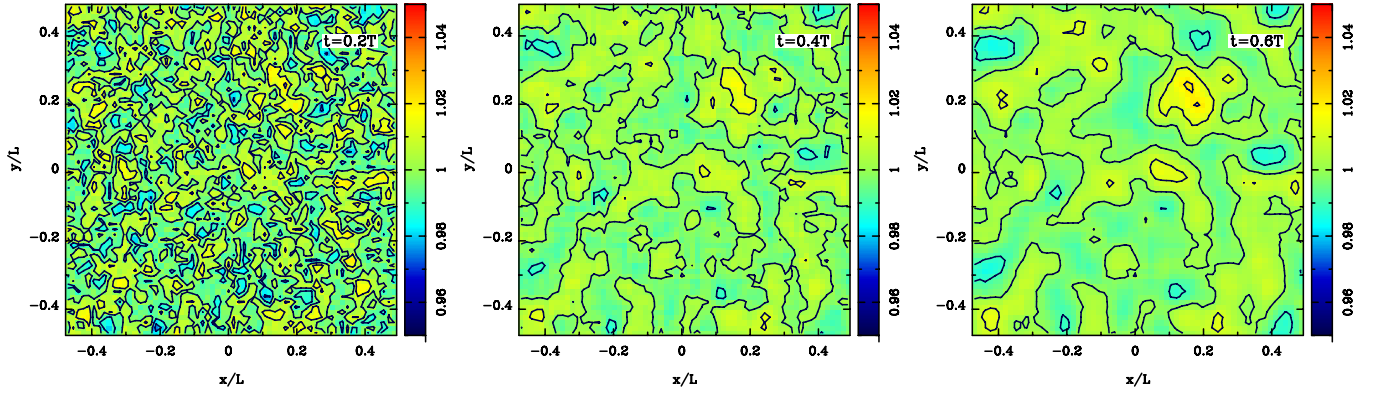(A color version of this figure is available in the online journal.)

Figure 13. Test 4: maps of $\rho(\mathbf{x})/\bar{\rho}$ on $z = 0$ planes at $t/T = 0.2$, 0.4, and 0.6 in the run with $k_J = 16\pi/L$. Contours are drawn for $0.96 \leqslant \rho(\mathbf{x})/\bar{\rho} \leqslant 1.04$ with an interval of 0.1.

(A color version of this figure is available in the online journal.)

$k_J = 8\pi/L$ (left panel) and $k_J = 16\pi/L$ (right panel), where the vertical line in each panel indicates the Jeans wavenumber. As expected, the fluctuation modes with $k < k_J$ grow through the gravitational instability, whereas the modes with $k > k_J$ damp due to the Landau damping. These features can be directly observed in the time evolution of the density fields in $\mathbf{x}$-space shown in Figure 13, in which we set $k_J = 16\pi/L$. We can see that density fluctuations with smaller wavelength that is dominant at the early epoch ($t = 0.2T$) gradually vanish and only those with longer wavelength grow with time through the gravitational instability. Note that the Jeans length in this run ($k_J = 16\pi/L$) is $L/8$, and it can be seen, by visual inspection, that there are no density fluctuations with wavelength much smaller than the Jeans length $L/8$ at $t = 0.6T$.

We have also performed a convergence test for the three-dimensional case in order to examine the effect of the resolution in the velocity space. We run the runs with the same spatial resolution of $N_x = N_y = N_z = 64$ but with $N_x^v = N_y^v = N_z^v = 32$ and $k_J = 16\pi/L$. In Figure 14, the resulting power spectra are compared with those obtained in the run with $N_x^v = N_y^v = N_z^v = 64$ for the same Jeans wavenumber. Overall an excellent agreement is found except that the damping of modes of large wavenumbers $kL \gtrsim 100$ is somewhat suppressed at late times in the run with coarser velocity resolution. Note that the vertical axis is in logarithmic scale; the amplitudes of the small-scale modes damp over three orders of magnitude by $t = 0.8T$. This behavior can be ascribed to the numerical suppression of the Landau damping due to the low velocity resolution we observe in Figure 9.

In Figure 15, another convergence test is performed to address the effect of the spatial resolution, in which the power spectra with $N_x = N_y = N_z = 64$ and $N_x = N_y = N_z = 128$ are compared for a fixed velocity resolution $N_x^v = N_y^v = N_z^v = 32$. The time evolution of the gravitational instability at large scales $k < k_J$ for both spatial resolution is in excellent agreement. For smaller scales with $k > k_J$, the power spectra are in good agreement at $kL \lesssim 100$. The difference of the power spectra at $kL \gtrsim 100$ can be ascribed to the mode coupling or the forward and inverse cascade of the fluctuation power.

Figure 16 shows the time evolution of the ratio between the power spectra with respect to the initial power spectrum $t = 0$, $P(k, t)/P(k, t = 0)$, for various k/k_J. We plot both the results with $N_x^v = N_y^v = N_z^v = 64$ and 32 for comparison. In the linear regime, the ratios should be proportional to $\exp(2\gamma t)$, where γ is the growth or damping rate calculated from the linear theory (see the previous section). We show the growth/damping of
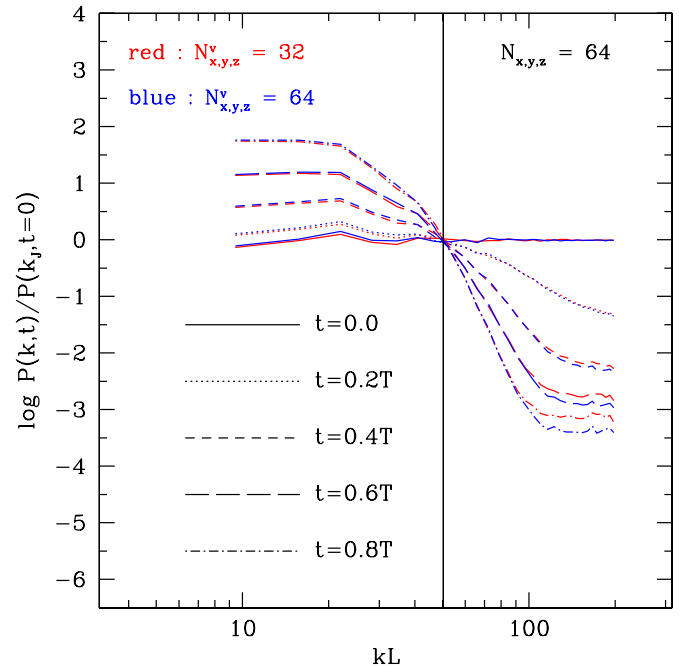

Figure 14. Test 4: the density power spectra at $t/T = 0.0$, 0.2, 0.4, 0.6, and 0.8 in the runs with $N_{x,y,z}^v = 32$ (red) and $N_{x,y,z}^v = 64$ (blue) and with $k_J = 16\pi/L$. The vertical line indicates the location of the Jeans wavenumber.

(A color version of this figure is available in the online journal.)

$\exp(2\gamma t)$ for the adopted value of k/k_J as solid lines. It can be seen that the results with $k/k_J = 0.19$, 0.81, and 1.18 agree with the linear theory well. For $k/k_J = 2.06$, however, the obtained damping rate is lower than the theoretical prediction especially at $t \gtrsim 0.3T$. This feature is also consistent with what we found in Test 2. It is likely owing to the same mechanism of the suppression of Landau damping for a "fluctuating" mode with large k/k_J as discussed in Section 3.2. There is no significant difference between the results with different velocity resolutions except for a very small deviation at late times for a strongly damping mode of $k/k_J = 2.06$. These differences in power spectra are again ascribed to the numerical suppression of the Landau damping due to the low resolution in the velocity space.

3.5. Test 5: King Sphere

We perform a Vlasov–Poisson simulation of the King sphere. The distribution function of the King sphere is a stable solution of the Vlasov–Poisson equations and has a finite extension in
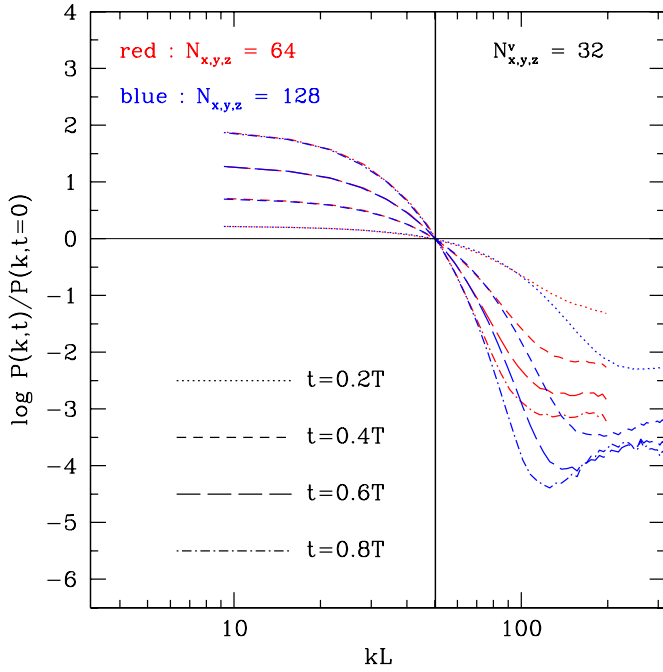
Figure 15. Test 4: the density power spectra at $t/T = 0.2$, 0.4, 0.6, and 0.8 in the runs with $N_{x,y,z} = 64$ (red) and $N_{x,y,z} = 128$ (blue) and with $k_J = 16\pi/L$. The vertical line indicates the location of the Jeans wavenumber.

(A color version of this figure is available in the online journal.)

Figure 16. Test 4: time evolution of $P(k)/P(k, t = 0)$ for various values of k/k_J. Results with $N_x^v = N_y^v = N_z^v = 64$ and 32 are plotted. Solid lines indicate the linear theory predictions, $P(k)/P(k, t = 0) \propto \exp(2\gamma t)$, where γ is the growth or damping rate of the density fluctuation.

(A color version of this figure is available in the online journal.)

the spatial coordinate unlike other analytic stable solutions such as the Plummer sphere and the Osipkov–Merritt model. The test is suitable for checking the accuracy of the time integration of the Vlasov equation.

Let us denote the relative potential $\Psi(r)$ and relative energy $\mathcal{E}$ by

$$\Psi(r) = -\Phi(r) \qquad (48)$$

and

$$\mathcal{E} = \Psi(r) - \frac{1}{2}\left(v_x^2 + v_y^2 + v_z^2\right), \qquad (49)$$

respectively, where $\Phi(r)$ is the gravitational potential with the boundary condition $\Phi(r) \to 0$ as $r \to \infty$. Then the distribution function of the King sphere is given by

$$f(\mathcal{E}) = \begin{cases} \rho_1 (2\pi\sigma^2)^{-3/2}(e^{\mathcal{E}/\sigma^2} - 1) & \mathcal{E} > 0 \\ 0 & \mathcal{E} < 0 \end{cases}, \qquad (50)$$

where ρ_1 and σ are the constants that determine the total mass M and the overall shape of the King sphere. The shape of the King sphere is characterized by the King parameter $W = \Psi(0)/\sigma^2$, which we set $W = 3$ in the followings. For $W = 3$, the tidal radius r_t, the outer boundary of the King sphere is $r_t = 5.37r_0$, where $r_0 \equiv 3\sigma/\sqrt{4\pi G\rho_0}$ and ρ_0 is the central mass density.

We consider the phase space volume with $-5.4r_0 \leqslant x, y, z \leqslant 5.4r_0$ and $-1.5V \leqslant v_x, v_y, v_z \leqslant 1.5V$ where $V \equiv r_0/T$ and $T = (GM/r_0^3)^{-1/2}$, and discretize it into grids with $N_x = N_y = N_z = 64$ and $N_x^v = N_y^v = N_z^v = 32$. In setting up the initial condition, after the phase space density in each phase space grid is calculated using the velocity and the relative potential at the grid center, we re-normalize the total mass of the King sphere so that the initial virial ratio $2K/|U|$ is unity, where K and U is the total kinetic and gravitational potential energy of the system.
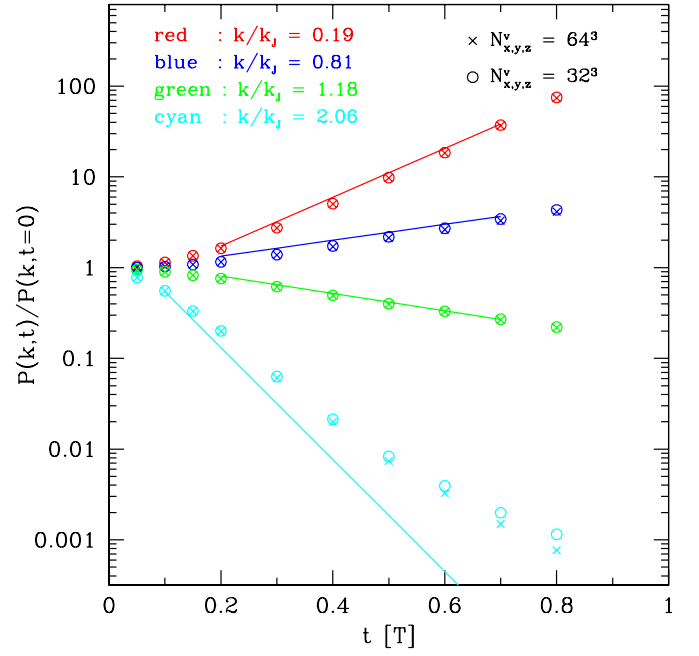
Figure 17 shows the mass distribution of the simulated King sphere as a function of radius r. Note that in this figure, mass within a shell with $r_1 < r < r_2$ is proportional to the area enclosed between the profiles $\rho(r)r^3$ in $r_1 < r < r_2$ and the horizontal dotted lines ($\rho(r)r^3 = 0$). It can be seen that the mass distribution does not change significantly over one dynamical timescale, irrespective of the numerical resolution of the velocity space. In Figure 17, we can see the slight mass transfer from the inner part ($r \simeq 0.4r_0$) to the outer ($r \simeq 1.5r_0$) regions of the King sphere. Since the grid spacing of the spatial grid is $\Delta x = \Delta y = \Delta z \simeq 0.0844r_0$ and the region with $r < r_0$ is resolved with only $\simeq 10$ grid points, the mass transfer can be ascribed to the numerical diffusion of the PFC scheme seen in Figure 2.

Figure 18 shows the time evolution of the kinetic, gravitational potential and total energy of the King sphere simulated with $N_x = N_y = N_z = N_x^v = N_y^v = N_z^v = 64$. Over one dynamical timescale, the total energy is kept constant with a relative error of $\lesssim 1\%$. The kinetic and gravitational potential energies are also kept constant with good numerical accuracy of 1% as long as $t = T$. The total mass is also conserved with a relative error of $\lesssim 10^{-4}$.

3.6. Test 6: Merging of Two King Spheres

As a final test, we perform a simulation of merging of two King spheres using our Vlasov–Poisson solver. We also run the same simulation using a conventional N-body method and compare the results.

The initial conditions are set up as follows. Two King spheres with the same physical parameters as in Test 5 are initially located at $(x, y, z) = (r_0, r_0, 0)$ and $(-r_0, -r_0, 0)$. The spheres are then given bulk velocities of $-0.2V$ and $0.2V$ along the x-axis, respectively. The phase space volume we consider has an dimension of $-6.4r_0 \leqslant x, y, z \leqslant 6.4r_0$ and
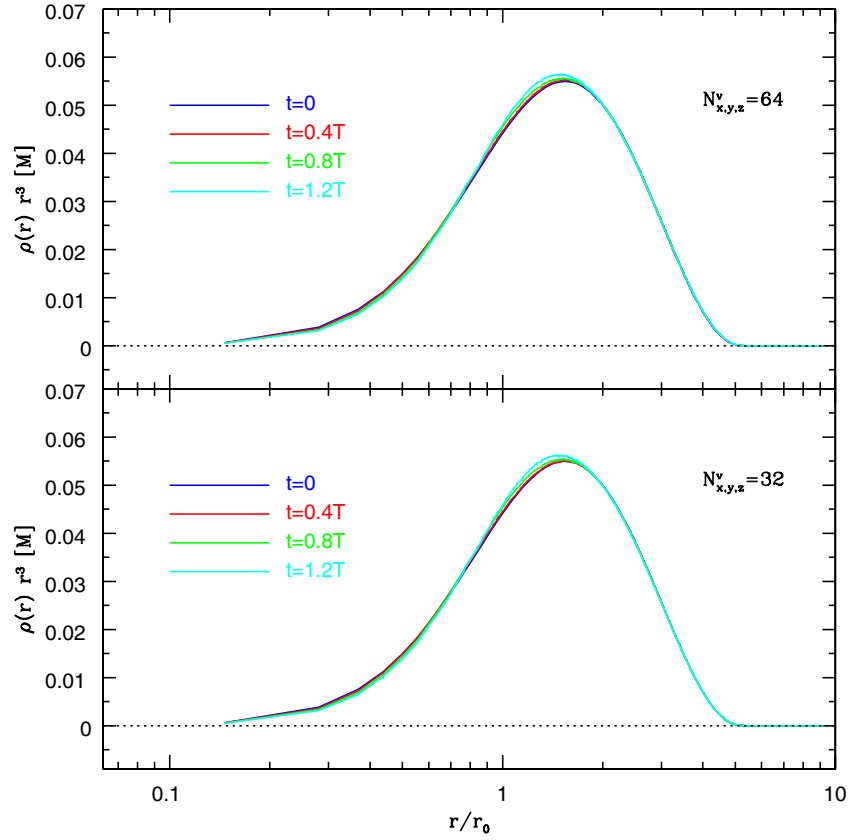
Figure 17. Test 5: mass distribution of the simulated King sphere at $t = 0, 0.4T, 0.8T$, and $1.2T$. The numbers of the spatial grid points are set to $N_x = N_y = N_z = 64$, and those of the velocity grid are $N_x^v = N_y^v = N_z^v = 64$ (upper panel) and $N_x^v = N_y^v = N_z^v = 32$ (lower panel).

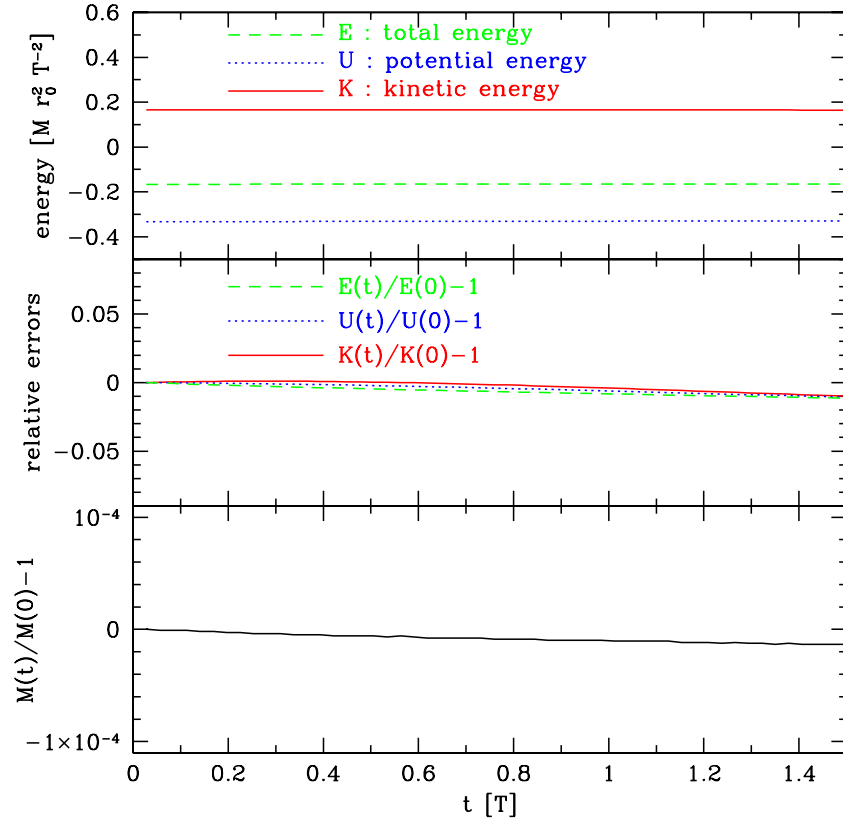(A color version of this figure is available in the online journal.)

Figure 18. Test 5: time evolution of the kinetic, gravitational potential, and total energy in the run with $N_x = N_y = N_z = N_x^v = N_y^v = N_z^v = 64$ and their relative difference are shown in the top and middle panels, respectively. The bottom panel depicts the relative error of the total mass.

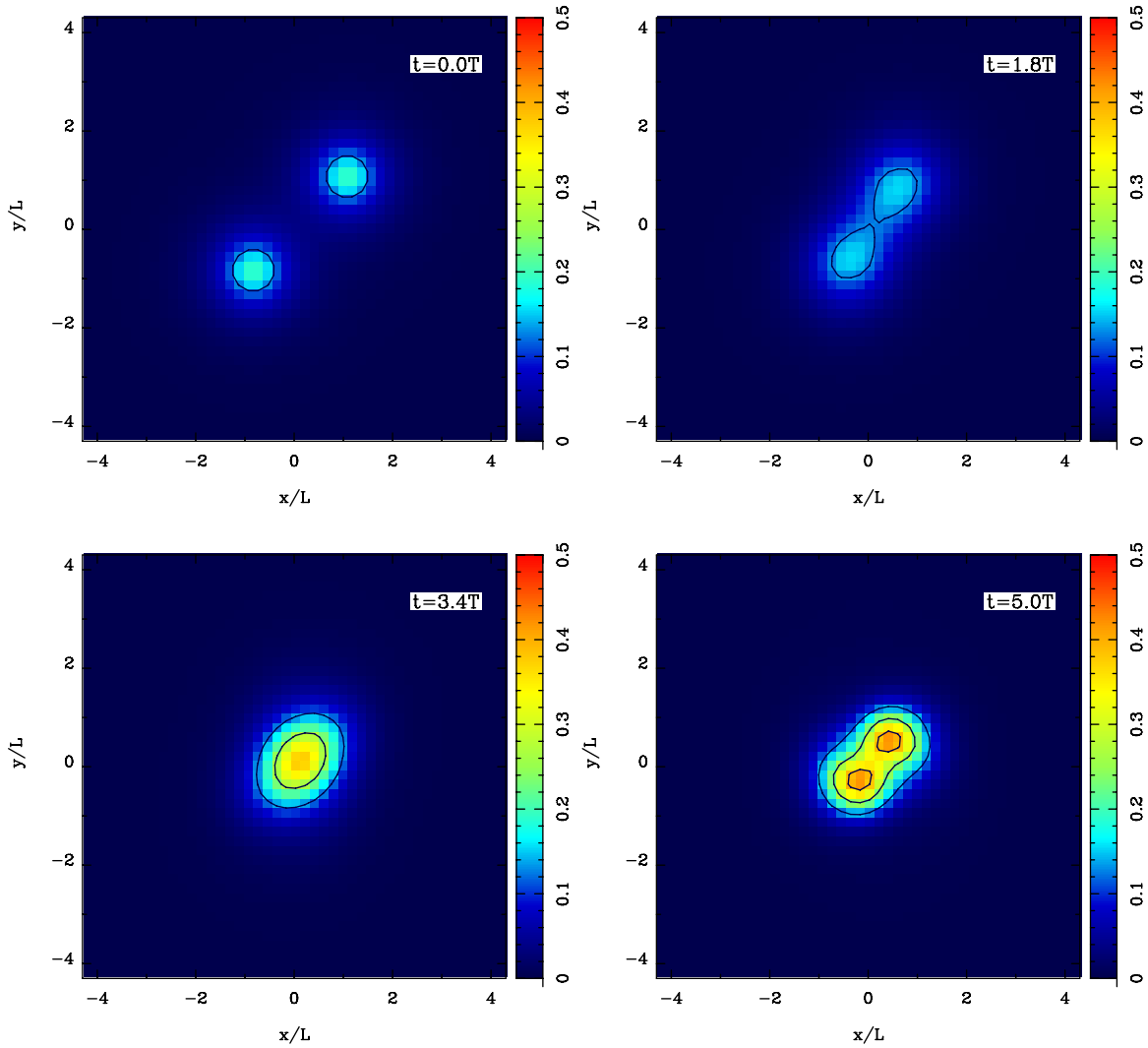(A color version of this figure is available in the online journal.)

Figure 19. Test 6: maps of the mass density $\rho(\boldsymbol{x})$ on the $z = 0$ plane at $t = 0.0$, $1.8T$, $3.4T$, and $5.0T$. The color scales indicate the mass density in units of M/r_0^3.
(A color version of this figure is available in the online journal.)

$-2.0V \leqslant v_x, v_y, v_z \leqslant 2.0V$, which is discretized onto grids with $N_x = N_y = N_z = 64$ and $N_x^{\mathrm{v}} = N_y^{\mathrm{v}} = N_z^{\mathrm{v}} = 64$ or 32. Note that the extension of the velocity space is larger than in the previous test, because some portion of the matter can have large velocities during the merging of the two spheres.

For comparison, we perform an N-body simulation of the same system. The initial conditions are set up in the same manner except that each King sphere is represented by 10^6 particles. In this N-body simulation, we adopt the particle-mesh method as a Poisson solver, in which the triangular-shaped cloud mass assignment scheme is used in computing mass density field from the particle distribution. The gravitational force is calculated using the four-point finite-difference scheme. The number of grid points to compute the gravitational potential is set to 64 for each of x-, y- and z-dimensions, which gives effectively the same spatial resolution as that of the Vlasov–Poisson solver.

Figure 19 depicts the time evolution of mass density map at $z = 0$ plane in the run with $N_x^{\mathrm{v}} = N_y^{\mathrm{v}} = N_z^{\mathrm{v}} = 64$. The cores of the two King sphere first encounter at $t = 3.4T$ and then go through each other in a collisionless manner. The density distribution is smooth at all the time. We compare the mass density distribution at $t = 5.0T$ between the Vlasov–Poisson simulation and the N-body simulation in Figures 19 and 20. Both

the simulations produce fairly consistent results, although the density distribution in the N-body simulations appears slightly asymmetric between the two spheres.

For a more quantitative comparison between the Vlasov–Poisson and N-body simulations, the time evolution of the kinetic, the gravitational potential, and the total energies in both the simulations are shown in the top panel of Figure 21. One can see clearly the consistent behaviors of the kinetic and gravitational potential energies in the Vlasov–Poisson and N-body simulations. The slight differences between the two runs are primarily due to the different discretization of the system. Relative errors of the total mass and energy conservation in the runs with $N_x^{\mathrm{v}} = N_y^{\mathrm{v}} = N_z^{\mathrm{v}} = 64$ and $N_x^{\mathrm{v}} = N_y^{\mathrm{v}} = N_z^{\mathrm{v}} = 32$ are shown in Figure 21. The total mass is well conserved with a sufficiently small relative error of $\ll 10^{-4}$ in both resolutions in the velocity space (bottom panel). We find a slight decrease in the total mass at $t \gtrsim 4.5T$ in the run with the lower velocity resolution. This is because the extent of the matter distribution in the velocity space exceeds the predefined velocity ranges during the merging of the cores of the two king spheres. In the run with the higher velocity resolution, while it is also the case that the extent of the matter distribution in the velocity space is not fully enclosed, the deviation from the
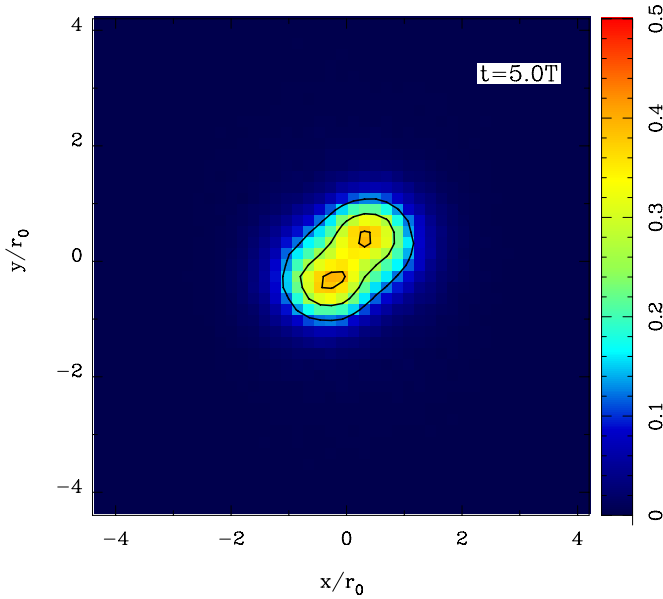
Figure 20. Test 6: a density map of the result at a $z = 0$ plane from the N-body simulation at $t = 5.0T$.

(A color version of this figure is available in the online journal.)

total mass conservation is kept relatively small because the better velocity resolution enables better reconstruction of the distribution function at the high velocity tails.

Total energy conservation is assured to be better in the run with the higher velocity resolution. Even with the higher reso-

lution, however, the relative error in the total energy is $\simeq 3\%$ at $t = 5T$, while it is $\lesssim 1.5\%$ at $t < 4T$. Again, this can be understood by the fact that the extent of the matter distribution in the velocity space is beyond the predefined velocity ranges during $4T \lesssim t \lesssim 5T$. Such velocity "overflow" gives a stronger impact to the total energy budget rather than the total mass conservation because matter with a large velocity has naturally a large kinetic energy. It would be desirable to develop a scheme that adaptively rescales the velocity space with proper reconstruction or re-mapping of the velocity distribution function.

We have seen in Figures 19 and 20 that the density distributions are quite similar between the Vlasov run and the N-body run. It is interesting to compare the matter distribution in the velocity space between the two simulations. The left panel of Figure 22 shows the phase space density in the velocity space at a single spatial grid point near the mass center of the two King spheres. For this plot, we use the output at $t = 4.2T$, when the two peaks of the phase space density match the bulk velocities of the two King spheres. The velocity distribution of the particles in the same spatial volume of the N-body simulation are depicted in the right panel of Figure 22. Although there are two broad clumps at roughly the same locations as those in the Vlasov–Poisson run, one can clearly see severe contaminations by the shot noise. The velocity structure is not well sampled even in the $N = 10^6$ run. Figure 23 is the same as Figure 22 but shows the phase space density in the velocity space in a spatial grid point at the outskirts of the system, $(x, y, z) = (1.8L, 1.8L, 0)$. In the Vlasov–Poisson simulations, we can see the two peaks in the velocity distribution, each of which corresponds to the bulk velocity of the two King spheres. On the other hand, since
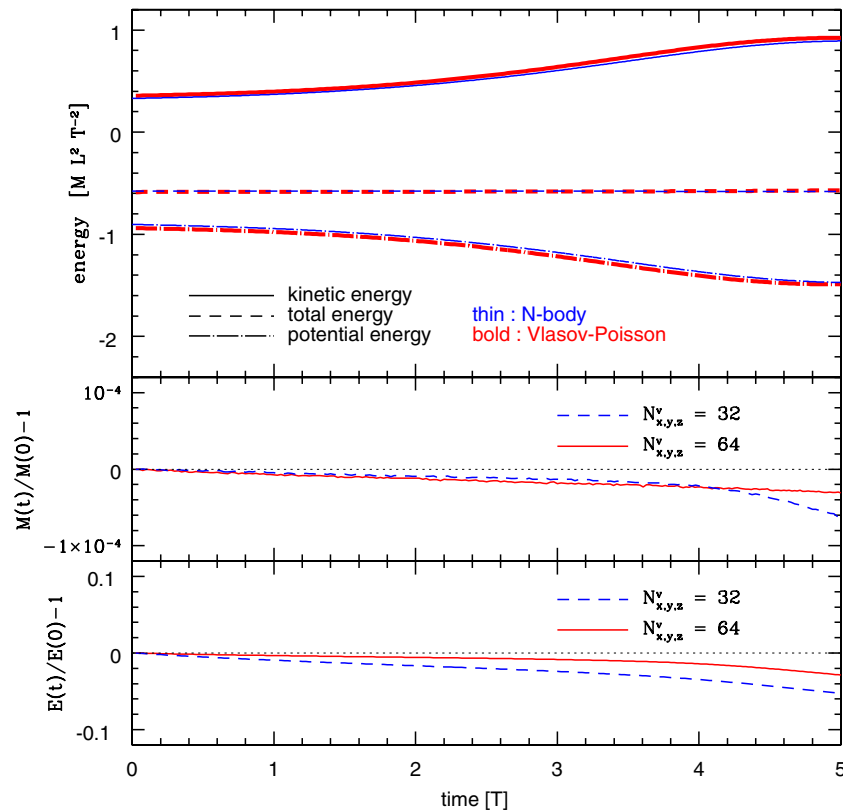

Figure 21. Test 6: time evolution of the kinetic, gravitational potential, and total energy in the run with $N_x = N_y = N_z = N_x^v = N_y^v = N_z^v = 64$ are shown by thick lines. Those in the N-body simulation are also shown by thin lines. The relative differences of the total energy and the total mass in the runs with $N_x^v = N_y^v = N_z^v = 64$ (solid lines) and $N_x^v = N_y^v = N_z^v = 32$ (dashed lines) are shown in the middle and bottom panels, respectively.

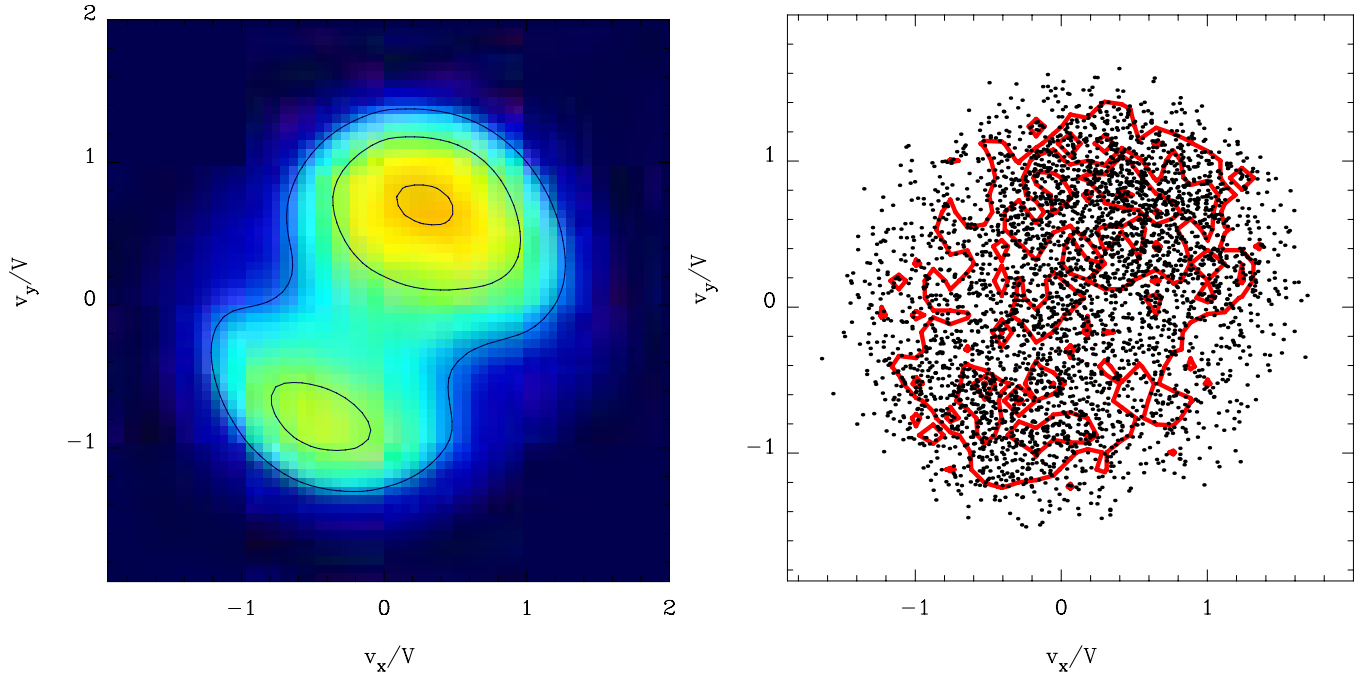(A color version of this figure is available in the online journal.)

14

Figure 22. Test 6: we compare the phase space density in the velocity space at a single spatial grid point near the center of the system at $t = 4.2T$ in the Vlasov–Poisson simulation (left panel) and the distribution of the particles in the same spatial volume in the N-body simulation (right panel). In the right panel, contours of the particle distribution are also drawn with the same binning as the velocity grid in the Vlasov–Poisson simulation.

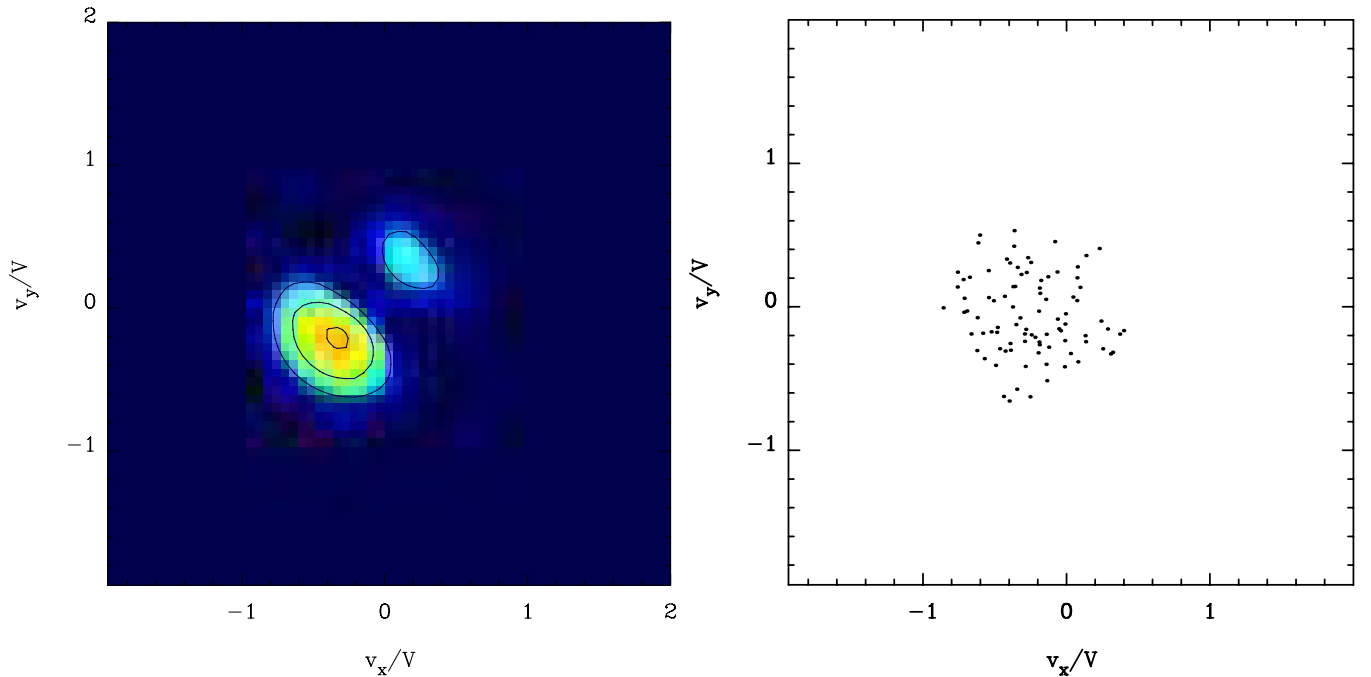(A color version of this figure is available in the online journal.)

Figure 23. Test 6: same as Figure 22 except that the phase space density in the velocity space at the outskirts of the system, $(x, y, z) = (1.8L, 1.8L, 0)$.

(A color version of this figure is available in the online journal.)

the number density of N-body particles in this region is small compared with that in the central regions, we cannot see such two-peak structure in the velocity distribution in the N-body simulation.

In order to quantify the shot noise level in the velocity distribution of the N-body simulation, we compute the power spectra of the velocity distribution function

$$P_v(k_v) = \langle |\hat{F}(k_v)|^2 \rangle, \quad (51)$$

where $\hat{F}(k_v)$ is a DFT of the distribution function in the velocity space and k_v is a wavenumber vector corresponding to a certain velocity vector. The velocity power spectra thus calculated at the same spatial position as in Figure 22 are shown in Figure 24. The two power spectra are in good agreement with each other at large velocity scales, $k_v V \lesssim 10$. At small velocity scales ($k_v V \gtrsim 10$), however, the N-body simulation exhibits a flat spectrum, showing good contrast with the nearly
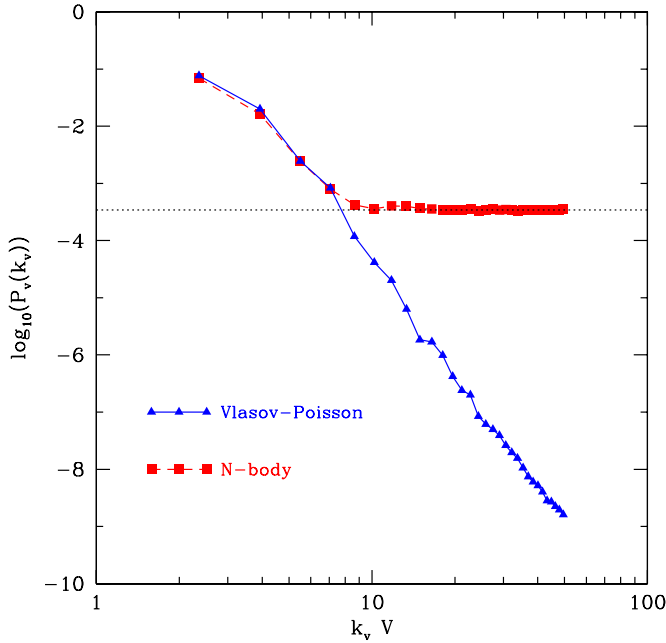
Figure 24. Test 6: power spectra of phase space density in the velocity space in the Vlasov–Poisson and N-body simulations.
(A color version of this figure is available in the online journal.)

power-law spectrum in the Vlasov–Poisson simulation. The velocity power for the N-body run is significantly contaminated by the shot noise. On the assumption that the velocity power of the Vlasov–Poisson simulation is accurate to $k_v V \sim 50$, we argue that the same result would be obtained if we could employ nearly a five orders of magnitude larger number of particles ($\simeq 10^{11}$) in the N-body simulation.

4. MEMORY CONSUMPTION, CPU TIMING, AND PARALLELIZATION EFFICIENCY

All of the simulations presented in the present paper were performed with a large-scale massively parallel supercomputer, T2K-Tsukuba system installed at Center for Computational Sciences, University of Tsukuba.[5] Each computational node of the T2K-Tsukuba system consists of four sockets of 2.3 GHz quad-core AMD Opteron and 32 GB of DDR2 SDRAM memory. All the nodes are connected through quad-rail of DDR Infiniband interconnection network.

The required memory M is approximately computed as

$$M = 256 \left(\frac{N^p}{64^3} \right) \left(\frac{N^v}{64^3} \right) \text{ GB}, \qquad (52)$$

where $N^p = N_x N_y N_z$ and $N^v = N_x^v N_y^v N_z^v$ are the numbers of grids in the spatial and velocity spaces, respectively. Our Vlasov code uses single-precision floating point numbers for storing the value of the distribution function. Since each node of the T2K-Tsukuba system can store data up to 24 GB on its memory, for the runs with $N_x N_y N_z = 64^3$ and $N_x^v N_y^v N_z^v = 64^3$, we typically use 16–64 nodes.

Table 1 shows a breakdown of the wall clock time consumed by several parts in our code over a single time step integration. All of the runs (A, B, and C) are performed with 64 MPI processes and each MPI process is also parallelized in a

Table 1
Breakdown of the Wall Clock Time for a Single Time Step

ID	N^p	N^v	N^{node}	$T^{p\,a}$ (s)	$T^{v\,a}$ (s)	$T^{grav\,b}$ (s)	$T^{comm\,c}$ (s)	$T^{total\,d}$ (s)
A	64^3	32^3	16	9.1	6.4	1.6	1.95	72.1
B	64^3	64^3	16	60.1	56.3	4.7	11.1	550.2
C	64^3	64^3	64	21.2	14.4	5.1	10.5	181.4

Notes. Since we perform three and six advection operations in a single time step in the spatial and velocity grids (see Equation (17)), their contributions to T^{total} are $3T^p$ and $6T^v$, respectively.
[a] Time for solving advection equations along a single dimension of the position and velocity spaces.
[b] Time for solving the Poisson equation including the calculation of the density field and the communication among nodes.
[c] Overhead for communicating the data in the adjacent computational subdomains.
[d] Total wall clock time to advance the system by a single time step.

multi-thread manner using the OpenMP implementation. For example, Run A adopts 16 nodes (equivalently 256 CPU cores), and each MPI process invokes 4 threads. Although the explored parameter space is limited, we confirm that our code performs well on up to 1024 cores (64 nodes). Comparing Run A and Run B, we see a good weak scaling in which the wall clock time T^{total} almost precisely scales with the simulation size ($N^p N^v$) with the same number of nodes. On the other hand, Run C, using four times more nodes, is only three times faster than Run B for the same number of grids. This is because the PFC scheme needs global maximum values of the phase space density over the entire one-dimensional computational regions to ensure the maximum principle, and thus solving the advection equations in the position space requires data transfer among the nodes associated with the adjacent computational regions through the inter-node network. As a result, as we see in the difference between T^p and T^v of Table 1, the advection operations in the position space take longer than in the velocity space. The necessary data transfer hampers the strong scaling in solving the advection equation in the position space.

5. SUMMARY AND DISCUSSION

In this paper, we have developed a fully parallelized Vlasov–Poisson solver in six-dimensional phase space for collisionless self-gravitating systems. The Vlasov solver is based on the recently proposed PFC scheme, whereas the Poisson solver utilizes the conventional convolution method based on the DFT. We have conducted large simulations of collisionless self-gravitating systems on the phase space discretized onto 64^6 grids. We have performed a suite of test calculations to examine the accuracy and performance of our simulation code.

The results of the test suite are summarized as follows. In Test 1, we examine the overall accuracy of the PFC scheme to solve a one-dimensional advection equation that is adopted in all the simulations presented in this paper. The mass and the energy conservations are confirmed to an accuracy of 10^{-5} for the one-dimensional advection problem. The initial distribution function is well preserved, without significant smearing due to numerical diffusion. In one-dimensional and three-dimensional tests for the time evolution of the density perturbation through gravitational interactions (Tests 2 and 4, respectively), the growth and damping rates of the density perturbations are consistent with the linear theory prediction at early phases. The Galilean invariance is also explicitly shown (Test 3). In Test 5, a stable spherical solution of the Vlasov–Poisson equations, the

King sphere, is also reproduced in full six-dimensional phase space. The results manifest that our time-integration scheme is accurate. Finally, our code works efficiently on massively parallel computers. It runs well on up to 1024 cores and scales well with the problem size and with the number of processors.

We summarize the advantages of the simulations of collisionless self-gravitating systems based on the Vlasov–Poisson equations over the conventional N-body simulations as follows. The resolution in the velocity space in the Vlasov–Poisson simulations is shown to be significantly better than that of N-body simulations in which the particle distribution in the velocity space is intrinsically rather noisy. Since the matter distribution in the velocity space is explicitly represented in the form of a continuum distribution function, physical processes that are sensitive to the velocity distribution such as the Landau damping can be treated accurately as seen in Tests 2 and 4. In that sense, the effect of free streaming of massive neutrinos on the large-scale structure formation in the universe might be better simulated with the Vlasov–Poisson simulations rather than the N-body simulations, since it is essentially the same mechanism as the Landau damping. Another important advantage of the Vlasov–Poisson simulations is that the collisionless feature is assured in the Vlasov–Poisson simulations, while artificial two-body relaxation could compromise the results of N-body simulations. Thus, for example, it might be more suitable to simulations of long-term or secular evolution of collisionless self-gravitating systems. In addition, the existence of two independent methods for simulating collisionless self-gravitating systems are essentially of great importance, since any simulation methods cannot be perfect and have some drawbacks. So far, we do not have any simulation methods for collisionless self-gravitating systems other than the N-body method. The Vlasov–Poisson simulation presented in this work is complementary to the N-body simulation and could be such an independent method of the N-body method. Thus, we could objectively evaluate the validity of simulations of collisionless self-gravitating systems by cross-checking the results obtained in both the methods.

Currently the spatial resolution of the Vlasov–Poisson simulations is not as impressive as those of the state-of-the-art N-body simulations, since the computational costs (computational capability and amount of memory) to conduct the Vlasov–Poisson simulations is relatively large compared with that of the N-body simulations. This drawback can be remedied by further improvements of the Vlasov–Poisson solver, which includes an adaptive mesh approach to improve the spatial and velocity resolutions without significantly increasing the required amount of the memory, and an adoption of more sophisticated higher-order schemes to solve one-dimensional advection equations to reduce numerical errors caused by the coarse-grained discretization of the phase space. In addition, the performance of our grid-based Vlasov solver scales well with the number of processors. Thus we expect the simulation size can be steadily increased as the available computing power increases in the near future. We foresee direct integration of the collisionless Boltzmann equation will be a promising method in the era of exa-flops computing.

APPENDIX

INTEGRAL FUNCTIONS IN THE PFC SCHEME

Here, we explicitly describe the scheme to compute Φ^+ and Φ^- in Equations (14) and (15), respectively. For a given set of f_i^n defined by equation (12), the values of Φ^+ and Φ^- can be obtained by interpolating the distribution function $f(x, t^n)$.[6]

In this work, we adopt the third-order interpolation for the primitive of the distribution function. On the interval $x_{i-1/2} \leqslant x \leqslant x_{i+1/2}$, the distribution function is interpolated as

$$\bar{f}(x, t^n) = f_i^n + \frac{a(x)}{6}(f_{i+1} - f_i) + \frac{b(x)}{6}(f_i - f_{i-1}), \quad \text{(A1)}$$

where $a(x)$ and $b(x)$ are given by

$$a(x) = \frac{1}{\Delta x^2}[2(x - x_i)(x - x_{i-3/2}) + (x - x_{i-1/2})(x - x_{i+1/2})], \quad \text{(A2)}$$

$$b(x) = \frac{1}{\Delta x^2}[2(x - x_i)(x - x_{i+3/2}) + (x - x_{i-1/2})(x - x_{i+1/2})]. \quad \text{(A3)}$$

It is straightforward to prove that

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \bar{f}(x, t^n)\, dx = f_i^n \Delta x, \quad \text{(A4)}$$

since one can obviously see that

$$\int_{x_{i-1/2}}^{x_{i+1/2}} a(x)\, dx = \int_{x_{i-1/2}}^{x_{i+1/2}} b(x)\, dx = 0. \quad \text{(A5)}$$

Here, we introduce slope correctors to ensure the maximum principle and avoid spurious oscillation:

$$\tilde{f}(x, t^n) = f_i^n + \frac{\epsilon_i^+}{6}a(x)(f_{i+1} - f_i) + \frac{\epsilon_i^-}{6}b(x)(f_i - f_{i-1}), \quad \text{(A6)}$$

where ϵ^+ and ϵ^- given by

$$\epsilon_i^+ = \begin{cases} \min(1, 2f_i/(f_{i+1} - f_i)) & \text{if } f_{i+1} > f_i, \\ \min(1, -2(f^{\max} - f_i)/(f_{i+1} - f_i)) & \text{if } f_{i+1} < f_i, \end{cases} \quad \text{(A7)}$$

$$\epsilon_i^- = \begin{cases} \min(1, 2(f^{\max} - f_i)/(f_i - f_{i-1})) & \text{if } f_i > f_{i-1}, \\ \min(1, -2f_i/(f_i - f_{i-1})) & \text{if } f_i < f_{i-1}, \end{cases} \quad \text{(A8)}$$

---

[6] Note that the expressions of Φ^+ and Φ^- in Filbet et al. (2001) contains several typos.

where $f^{\max}$ is the maximum value of the distribution function in the entire computational domain under consideration.

Using this interpolation, we can obtain the expression of Φ^+ and Φ^- by substituting the interpolation (A6) into the distribution function in Equations (14) and (15). When the propagating velocity (u in Equation (11)) is positive, we have

$$\Phi^+ = \sum_{\ell=j+1}^{i} f_\ell^n \Delta x + \alpha_i \left[f_j^n + \frac{\epsilon_j^+}{6} \left(1 - \frac{\alpha_i}{\Delta x} \right) \left(2 - \frac{\alpha_i}{\Delta x} \right) \right.$$
$$\left. \times \left(f_{j+1}^n - f_j^n \right) + \frac{\epsilon_j^-}{6} \left(1 - \frac{\alpha_i}{\Delta x} \right) \left(1 + \frac{\alpha_i}{\Delta x} \right) \left(f_j^n - f_{j-1}^n \right) \right],$$
(A9)

and

$$\Phi^- = \sum_{\ell=k+1}^{i-1} f_\ell^n \Delta x + \alpha_{i-1} \left[f_k^n + \frac{\epsilon_k^+}{6} \left(1 - \frac{\alpha_{i-1}}{\Delta x} \right) \left(2 - \frac{\alpha_{i-1}}{\Delta x} \right) \right.$$
$$\left. \times \left(f_{k+1}^n - f_k^n \right) + \frac{\epsilon_k^-}{6} \left(1 - \frac{\alpha_{i-1}}{\Delta x} \right) \left(1 + \frac{\alpha_{i-1}}{\Delta x} \right) \left(f_k^n - f_{k-1}^n \right) \right],$$
(A10)

where we set $\alpha_i = x_{j+1/2} - X(t^n, t^{n+1}, x_{i+1/2})$, and the indices, j and k, are defined such that

$$x_{j-1/2} \leqslant X(t^n, t^{n+1}, x_{i+1/2}) \leqslant x_{j+1/2}, \qquad (A11)$$

$$x_{k-1/2} \leqslant X(t^n, t^{n+1}, x_{i-1/2}) \leqslant x_{k+1/2}. \qquad (A12)$$

For a negative propagating velocity, Φ^+ and Φ^- are given as

$$\Phi^+ = - \sum_{\ell=j+1}^{i} f_\ell^n \Delta x + \beta_i \left[f_j^n - \frac{\epsilon_j^+}{6} \left(1 - \frac{\beta_i}{\Delta x} \right) \left(1 + \frac{\beta_i}{\Delta x} \right) \right.$$
$$\left. \times \left(f_{j+1}^n - f_j^n \right) - \frac{\epsilon_j^-}{6} \left(2 + \frac{\beta_i}{\Delta x} \right) \left(1 + \frac{\beta_i}{\Delta x} \right) \left(f_j^n - f_{j-1}^n \right) \right],$$
(A13)

and

$$\Phi^- = - \sum_{\ell=k+1}^{i-1} f_\ell^n \Delta x + \beta_{i-1} \left[f_k^n - \frac{\epsilon_k^+}{6} \left(1 - \frac{\beta_{i-1}}{\Delta x} \right) \left(1 - \frac{\beta_{i-1}}{\Delta x} \right) \right.$$
$$\left. \times \left(f_{k+1}^n - f_k^n \right) - \frac{\epsilon_k^-}{6} \left(2 + \frac{\beta_{i-1}}{\Delta x} \right) \left(1 + \frac{\beta_{i-1}}{\Delta x} \right) \left(f_k^n - f_{k-1}^n \right) \right],$$
(A14)

where we set $\beta_i = x_{j-1/2} - X(t^n, t^{n+1}, x_{i+1/2})$.

It should be noted that the global difference between $\bar{f}(x, t)$ and $\tilde{f}(x, t)$ is constrained by the relation

$$\int |\tilde{f}(x, t) - \bar{f}(x, t)| dx \leqslant 4TV(f)\Delta x, \qquad (A15)$$

where $TV(f)$ is the total variation of the distribution function given by

$$TV(f) = \sum_i |f_{i+1} - f_i|. \qquad (A16)$$

REFERENCES

Binney, J., & Tremaine, S. 2008, Galactic Dynamics (2nd ed.; Princeton, NJ: Princeton Univ. Press)
Cheng, G., & Knorr, J. 1976, JCoPh, 22, 330
Cuperman, S., Harten, A., & Lecar, M. 1971, Ap&SS, 13, 411
Filbet, F., & Sonnendrücker, E. 2003, CoPhC, 150, 247
Filbet, F., Sonnendrücker, E., & Bertrand, P. 2001, JCoPh, 172, 166
Fujiwara, T. 1981, PASJ, 33, 531
Fujiwara, T. 1983, PASJ, 35, 547
Hernquist, L., & Ostriker, J. P. 1992, ApJ, 386, 375
Hockney, R. W., & Eastwood, J. W. 1981, Computer Simulation Using Particles (New York: McGraw-Hill)
Hohl, F., & Feix, M. R. 1967, ApJ, 147, 1164
Hozumi, S. 1997, ApJ, 487, 617
Janin, G. 1971, A&A, 11, 188
Klimas, A. J. 1987, JCoPh, 68, 202
Klimas, A. J., & Farrell, W. M. 1994, JCoPh, 110, 150
Minoshima, T., Matsumoto, Y., & Amano, T. 2011, JCoPh, 230, 6800
Roberts, K. V., & Berk, H. L. 1967, PhRvL, 19, 297
Sonnendrücker, E., Roche, J., Bertrand, P., & Ghizzo, A. 1998, JCoPh, 149, 201
Tasker, E. J., Brunino, R., Mitchell, N. L., et al. 2008, MNRAS, 390, 1267
Wadsley, J. W., Veeravalli, G., & Couchman, H. M. P. 2008, MNRAS, 387, 427
Weinberg, M. D. 1999, AJ, 117, 629
Zaki, S. I., Gardener, L. R. T, & Boyd, T. J. M. 1988, JCoPh, 79, 184