

**Vlasov model using kinetic phase point trajectories**F. Kazeminezhad,<sup>1,\*</sup> S. Kuhn,<sup>2</sup> and A. Tavakoli<sup>3</sup><sup>1</sup>*Independent Consultant for Plasma and Energy Physics Group, Institut für Theoretische Physik, Universität Innsbruck, Innsbruck, Austria**and Institute for Studies in Theoretical Physics and Mathematics (IPM), Tehran, Iran*<sup>2</sup>*Institut für Theoretische Physik, Universität Innsbruck, Innsbruck, Austria*<sup>3</sup>*Institute for Studies in Theoretical Physics and Mathematics, P.O. Box 19395-5531, Farmanieh Bldg., Tehran, Iran*

(Received 18 March 2001; revised manuscript received 10 September 2002; published 26 February 2003)

A method of solution of the collisionless Vlasov equation by following fixed collisionless phase point (“particle”) trajectories (characteristics) in phase space is presented. It solves the coupled Vlasov Maxwell system self-consistently and employs the Leapfrog-Trapezoidal scheme to solve for the characteristics explicitly. It then uses the bilinear finite element interpolation scheme in phase space and maps vital instantaneous phase point information (distribution function) to a fixed background phase space mesh while retaining it at the phase point. The scheme is an enhanced second order one in time and fourth order in space. The code is then used to model a thermal plasma as well as two stream instability using mobile electrons and fixed background ions: the scheme being a momentum conserving one by construction allows energy conservation without assignment of particle shape functions; Langmuir waves are obtained with very good agreement with the Bohm-Gross dispersion relation; the two stream results do not show any numerically induced oscillations attributed to the initial well-ordered velocity distributions. Retention of the characteristics also minimized diffusion. Extensive numerical stability analysis deriving Courant condition for the scheme as well as behavior of computational modes are done in Appendix A, as well as estimating the impact of numerical diffusion in Appendix B. Two to five dimensional versions in phase space exist.

DOI: 10.1103/PhysRevE.67.026704

PACS number(s): 83.85.Pt, 52.65.Ff, 82.20.Fd, 52.35.-g

**I. INTRODUCTION**

The knowledge of the temporal evolution of the distribution function has long been a desire of plasma physicists as well as many involved in many-body physics research. Here we cite a few examples where temporal evolution critically depends on the time dependent distribution function  $f$ : transport in plasmas, where transport terms possess functional dependence on  $f$ ; impact of the Tokamak scrape off layer impurities on the bulk; and impact of the chaotic particle distributions on the magneto-tail dynamics. The kinetic Vlasov models which follow the distribution function provide statistically acceptable  $f$  and therefore are valuable tools for the above mentioned problems.

Many researchers have taken great pains with some success in the numerical integration of the Vlasov equation [1–4,6,8,9]. Some sought solutions by assuming polynomial expansion of the distribution function in velocity [10]. One of the main obstacles in this endeavor has been the development of steep gradients in velocity space; i.e., a problem with no seemingly simple cure. Partial treatments such as increase in velocity resolution, have sharply limited the ability to extend the above work to higher dimensions and thus treat realistic problems.

Integration of the Vlasov equation along the collisionless phase point trajectories (characteristics) has been the most promising of these methods [4,11–13]. In most of these works, a splitting scheme initially proposed in Ref. [11] has been employed. In this scheme, they assumed that the Vlasov

equation “should be an approximation” to a couple of equations involving the “free-streaming” ( $\partial f/\partial x$ ) and “accelerating” [ $E(\partial f/\partial v)$ ] terms; the latter two involving lines of constant phase space coordinates. The new  $f$  was thus obtained as an algebraic expression in terms of the old  $f$  by a suitable interpolation method [14]. There are a couple of problems with this “approximation”: (i) it was not rigorously shown under which circumstances the coupled equations have solutions “approximately” consistent with the Vlasov equation; and (ii) following characteristics along the phase space coordinates departs one from the characteristics on which  $f$  truly remains invariant. As such the validity of the final solutions resulting from following these coordinates would be debatable.

In the present work though we directly follow the characteristics along which  $f$  is constant in the collisionless case, i.e., the collisionless phase point trajectories.<sup>1</sup> Initially  $f$  is prescribed on a fixed background mesh in phase space consisting of phase points. The phase points following the characteristics are advanced in time by the Leapfrog-Trapezoidal [15] method. Interpolation is performed between the phase points and the fixed background mesh in phase space using the method of bilinear rectangular finite elements [16]. From the interpolated  $f$  on the mesh, all the desired quantities such as space charge density, electron kinetic energy density, etc. are then computed; and used to further advance the phase points.

<sup>1</sup>Note that [4] also attempted a similar procedure; there they did not appear to retain the same characteristics throughout. We believe that was the main cause of their observed energy nonconservation; in section three we present results confirming this fact. Nunn [5] using a similar concept while retaining characteristics though, studied the driven impact of the VLF emissions on resonant particles in the magnetosphere.

\*Present address: 4723 Avenue De Las Flores, Yorba Linda, CA 92886

The organization of the paper is as follows. Section II describes the general model. Section III describes the numerical scheme used to update the characteristics and the interpolation scheme. Section IV includes tests of the model via energy conservation and the linear eigenmode analysis: the energy tests include comparison between cases with and without fixed characteristics, as well as Leapfrog versus Leapfrog-Trapezoidal numerical schemes for updating trajectories in the former case;<sup>2</sup> linear eigenmode tests include thermal as well as streaming plasmas for confirmation with the linear theory, as well as testing the extend of numerically induced streaming instability and diffusion in the model. Finally a brief conclusion with future directions is included.<sup>3</sup>

## II. THE MODEL

Here we model a system comprised of thermal electrons and immobile ions. At time  $t=0$ , Maxwellian electrons lie on top of the ions in a phase space mesh. For every point in configuration space, there exist  $nv_x$  points in velocity space which give rise to a discretized Maxwellian. This arrangement insures a Maxwellian distribution and practically to within discretization noise zero electric field within the configuration grid space initially. As such it corresponds to particle in cell codes which are initialized with “quiet starts” [6]. Generalization of this model to include Maxwellian ions at the same phase space points with the electrons would also insure “quiet starts” as it will insure zero electric field and related velocity distributions within that scale. This is in contrast to the PIC codes which are initialized with a random velocity distribution and therefore higher initial electric field fluctuations.

For a system composed of  $nx$  configuration space points, one thus has  $nx \times nv_x$  grid phase points. Each phase point is by definition characterized by its position  $x$  and its velocity  $v_x$ , and has associated with it a distribution function value  $f_p = (1/\sqrt{\pi v_{th}^2}) e^{-v_x^2/v_{th}^2}$  ( $v_{th}$  is the thermal speed); please see Fig. 1.

As the simulation begins, the representative electron phase points begin to move at their respective speeds under the influence of the resulting charge separation field  $E_x$ . Their trajectories are therefore advanced using the following set of equations:

$$\frac{dv_x}{dt} = -\frac{e}{m} E_x, \quad (1)$$

<sup>2</sup>We define fixed characteristics when they are retained throughout a simulation, in contrast to reconstructing them at each time step as in Ref. [4]. The energy tests are indented to distinguish the impact of characteristic retention from the computational modes (Appendix A) on the energy conservation.

<sup>3</sup>The presentations outlined are in two dimensional phase space. This is done to aid the reader better visualize the outlined interpolations and shorten equations and function indices without any loss of generality to higher dimensions. Where appropriate, it is explained how the extension to higher dimensions has been performed in the higher dimensional versions.

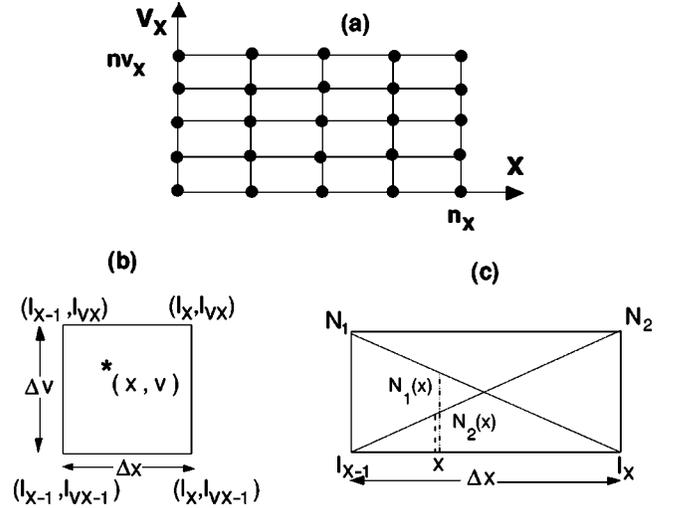


FIG. 1. (a) A typical phase space mesh. (b) A typical phase point  $(x, v_x)$  at its host cell;  $\Delta x$  and  $\Delta v$  correspond to the respective grid spacings. (c) One dimensional finite element shape factors (weights) of the two cell points at the phase point location.

$$\frac{dx}{dt} = v_x, \quad (2)$$

$$\frac{d^2 \phi}{dx^2} = -4 \pi e (n_0 - n_e), \quad (3)$$

where  $n_0$  is the density of the uniform background ion distribution and  $n_e$  corresponds to that of the electrons. As the representative phase points follow their collisionless trajectories, the  $f_p$  associated with them remains unchanged. That information is however mapped into a fixed background mesh; i.e., from the distribution of  $f_p$ 's,  $f_g$  associated with the phase space mesh is obtained by interpolation. From  $f_g$  any and all desired higher moments can be computed. Due to computational constraints, velocity cutoffs as in PIC models are imposed. For example,  $f_g$  at maximum (cutoff)  $v_x$  in our simulation was  $10^{-11}$  times maximum  $f_g$ . Due to the constancy of the phase density in the model, the fraction of phase space initially occupied by  $f_g$  should remain unchanged unless  $f_g$  begins to distort due to numerical errors.<sup>4</sup> That distortion will set limitations in the model which is discussed in Appendix B.

Due to the discretization noise, the physical system considered maintains small amplitude collective oscillations approximately consistent with the following linearized set of fluid equations:<sup>5</sup>

<sup>4</sup>This condition will no longer hold if micro scale fields (and the resulting transport correlations [7]) could no longer be neglected in comparison with the long range fields and result in collision integral terms in the Vlasov equation.

<sup>5</sup> $u$  here is the velocity moment (average velocity), while  $v_x$  above corresponds to the phase point velocities. Also for sufficiently low frequencies, the electrons remain practically isothermal and the eigenmode oscillations are well approximated by the noted set of fluid equations.

$$-i\omega n_e + n_0 i k v = 0, \quad (4)$$

$$-i\omega n_0 m u = -ik T n_e - n_e e E_x, \quad (5)$$

$$-ik E_x = 4\pi n_e e, \quad (6)$$

which give rise to the well known Bohm-Gross dispersion relation

$$\omega^2 = \omega_{pe}^2 + k^2 v_{th}^2. \quad (7)$$

This equation will be used for testing the model.

### III. NUMERICAL SCHEME

#### A. Interpolation procedure

As the representative phase points follow their characteristics, they continually exchange information with the fixed background mesh depicted in Fig. 1. In this exchange the method of bilinear weighting is employed [16], where at each instance a representative phase point contributes to the corners of its instantaneous host cell [Fig. 1(b)] its distribution function. The basis of this scheme is depicted in Fig. 1(c): i.e., for each degree of freedom (e.g.,  $x$ ) a representative phase point  $x$  located within the cell  $(l_x - 1, l_x)$  contributes its  $f_p$  with the weights

$$N_2(x) = \frac{x - (l_x - 1)\Delta x}{\Delta x}, \quad (8)$$

$$N_1(x) = 1 - N_2(x) \quad (9)$$

to the points  $l_x$  and  $(l_x - 1)$ , respectively, and  $N_{v_2}$  and  $N_{v_1}$  [with  $(l_x, x)$  replaced by  $(l_{vx}, v_x)$  in these equations] to the mesh points  $l_{vx}$  and  $(l_{vx} - 1)$ , respectively. In higher dimensions then, a phase point  $(x, v)$  [Fig. 1(b)], should make the following contributions to  $(l_x - 1, l_{vx} - 1)$ ,  $(l_x, l_{vx} - 1)$ ,  $(l_x - 1, l_{vx})$ , and  $(l_x, l_{vx})$ , respectively,

$$A_1 = N_1 N_{v_1},$$

$$A_2 = N_2 N_{v_1},$$

$$A_3 = N_1 N_{v_2},$$

$$A_4 = N_2 N_{v_2},$$

the sum of which gives

$$\begin{aligned} \sum_{i,j} N_i N_{vj}(x, v) &= N_1(N_{v_1} + N_{v_2}) + N_2(N_{v_1} + N_{v_2}) \\ &= N_1 + N_2 = 1, \end{aligned} \quad (10)$$

where the important property of these weight factors outlined in Eq. (9) has been used for both  $N$  as well as  $N_v$ . Therefore, the mesh point  $(l_x, l_{vx})$  should receive contributions from all its associated representative phase points as follows:

$$f_g(l_x, l_{vx}) = \frac{\sum_{\mathbf{v}(x,v)} N(x) N_v(v) f_p}{\sum_{\mathbf{v}(x,v)} N(x) N_v(v)}, \quad (11)$$

where the summation runs over all the associated phase points. Likewise, extension of this procedure to include more dimensions involves taking account of the extra dimensions' contributions. This scheme will thus allow usage of nonuniform meshes in phase space as long as rectangular geometry is maintained.

From  $f_g$  then, any moment of the distribution function such as electron density  $n_e$  can be computed; e.g.,

$$n_e(l_x) = \sum_{l_{vx}} \frac{f_g(l_x, l_{vx}) + f_g(l_x, l_{vx} + 1)}{2.0} \Delta v, \quad (12)$$

where  $n_e(l_x)$  corresponds to the electron density at the point  $l_x$ .<sup>6</sup> Using the density  $n_e$ , the potential  $\phi$  and, therefore, the electric field  $E_x$  are obtained on the mesh, and used to advance the phase points by the same interpolation procedure. This interpolation scheme, with the numerical schemes to be outlined, have ensured mass, momentum, and energy conservation as will be demonstrated in the density and energy plots.

#### B. Trajectory integration

The collisionless phase point trajectories and the electric potential are advanced by finite differencing Eqs. (1)–(3) above. The following normalizations are employed:<sup>7</sup>

$$\omega_{pe} t \rightarrow t, \quad \frac{x}{r_D} \rightarrow x, \quad \frac{v}{v_{th}} \rightarrow v, \quad \frac{e\phi}{T} \rightarrow \phi, \quad (13)$$

where  $\omega_{pe}$  and  $r_D$  are the electron plasma frequency and Debye lengths, respectively.

This results in the normalized  $\omega_{pe} = 1$  and reduces the said equations to the following set:

$$\frac{dv_x}{dt} = -E_x, \quad (14)$$

$$\frac{dx}{dt} = v_x, \quad (15)$$

$$\frac{d^2\phi}{dx^2} = \frac{n_e}{n_0} - 1. \quad (16)$$

For Eqs. (14) and (15), the Leapfrog-Trapezoidal scheme is employed. There, the values of  $v_x$  and  $x$  need to be known

<sup>6</sup>This sum corresponds to the Trapezoidal method of numerical integration of  $f_g$ . In higher velocity dimensions, the contributions of the other dimensions are multiplied by this contribution.

<sup>7</sup>Note this normalization is henceforth also applied to all the other equations in this article.

TABLE I. Numerical algorithm of the kinetic trajectory Vlasov model.

---



---

Initially we have  $v_x^{n-1}, x^{n-1/2}, f_p^{n-1/2} = f_p(x^{n-1/2}, v_x^{n-1})$

1. Interpolate  $f_p^{n-1/2}$  to obtain  $f_g^{n-1/2} = f_g(x^{n-1/2}, v_x^{n-1})$  and  $n_e^{n-1/2}$ .
2. Solve Poisson equation to obtain  $E_x^{n-1/2}$  from  $n_e^{n-1/2}$ .
3. Push the phase point velocities one time step,  $v_x^{*n} = v_x^{n-1} - E_x^{n-1/2} \Delta t$ .  
For  $n=0$  only compute  $x^{n-1} = x^{n-1/2} - [v_x^{n-1} + 0.5(v_x^{*n} + v_x^{n-1})] \Delta t/2$ .  
For  $n=0$  only determine  $f_g^{n-1}, n_e^{n-1}$ , and  $E_x^{n-1}$ .
4. Push the phase point positions one time step,  $x^{*n+1/2} = x^{n-1/2} + v_x^{*n} \Delta t$ .
5. Trapezoidal step one; determine  $v_x^{n-1/2}, x^n$  (middle step):  
(a)  $v_x^{n-1/2} = v_x^{n-1} - 0.5(E_x^{n-1/2} + E_x^{n-1}) \Delta t/2$ ,  
(b)  $x^n = x^{n-1/2} + 0.5(v_x^{*n} + v_x^{n-1/2}) \Delta t/2$ .
6. Obtain “improved”  $f_g^n = f_g(x^n, v_x^{n-1/2})$ ,  $n_e^n, E_x^n$  from  $x^n$  and  $v_x^{n-1/2}$ . Optional: Do the same calculation using  $v_x^{*n}$  instead of  $v_x^{n-1/2}$ .
7. Push the phase point velocities in the middle time step,  $v_x^{*n+1/2} = v_x^{n-1/2} - E_x^n \Delta t$ .
8. Trapezoidal step two (determine the “improved” updated velocities),  $v_x^n = v_x^{n-1/2} - \frac{1}{2}(E_x^n + E_x^{n-1/2}) \Delta t/2$ .
9. Re-advance the phase point positions using the “improved” velocities,  $x^{n+1/2} = x^n + \frac{1}{2}(v_x^{*n+1/2} + v_x^n) \Delta t/2$ .
10. Pass  $x^{n+1/2}, v_x^n, x^n$ , and  $E_x^n$  to the next time step.

---



---

at both time steps  $t - \Delta t$  and  $t - (\Delta t/2)$  in order to advance them to steps  $t$  and  $t + (\Delta t/2)$ , respectively. In the conventional Leapfrog scheme, though,  $v_x(t - \Delta t)$  and  $x[t - (\Delta t/2)]$  are sufficient to obtain  $v_x(t)$  and  $x[t + (\Delta t/2)]$ . The Leapfrog-Trapezoidal as the Leapfrog scheme itself has a total truncation error when integrating to a fixed time proportional to  $(\Delta t)^2$  in the limit as  $\Delta t$  goes to zero. The trapezoidal steps have two functions: (i) the primary function is to eliminate the “parasitic” solution associated with the Leapfrog scheme which results in a growing computational mode; (ii) as a result, the constant which multiplies the truncation error [ $\sim (\Delta t)^2$ ] becomes smaller; i.e., the Leapfrog-Trapezoidal scheme becomes an enhanced second order scheme in time [15].<sup>8</sup>

Equation (16) is solved numerically by the inversion of the Poisson operator matrix using the left-right (LR) theorem [17]. The potential  $\phi$  thus obtained is second order accurate in  $\Delta x$ . They are then made fourth order accurate by the following substitution [18]:

$$\phi_h \left( x + \frac{\Delta x}{2} \right) = \frac{7}{12} [\phi(x + \Delta x) + \phi(x)] - \frac{1}{12} [\phi(x + 2\Delta x) + \phi(x - \Delta x)], \quad (17)$$

where  $\phi_h$  is the fourth order potential. The interpolated electron density’s accuracy was likewise increased. These corrections helped enhance the  $\phi$  and  $E_x$  accuracy and consequently improve the results markedly.

To summarize: At the beginning of each time step, from the phase point distribution functions the charge density distribution is obtained. With this charge density, the Poisson equation is solved for the electric potential. Next comes trajectory advancement [i.e., Eqs. (14) and (15)]: here the Leap-

frog scheme steps advance  $v_x$  and  $x$  one time step and basically complete one push cycle. They, however, serve as auxiliary quantities for the trapezoidal steps which serve to improve them iteratively. In the Leapfrog scheme steps (steps 3 and 4 of Table I), the time centering is performed in explicit style. The advanced quantities (asterisk superscripted) serve as auxiliary first iterations which are “improved” in the trapezoidal steps (steps 8 and 9 in Table. I); the time centering here is achieved in implicit style using the auxiliary quantities. This improves the time advanced quantities in the advanced quantities and ends the push cycle. The outline of these procedures is given in Table I.

## IV. TESTS OF THE MODEL

### A. Conservation laws

A thermal plasma is modeled by simply setting the phase points in Fig. 1 into motion at their respective initial velocities. The round-off errors arising from finite differencing act as initial perturbations in the system, plus constant source of noise at any later time.

Conservation laws are the first order tests following the construction of a simulation model. The most basic of these are the continuity and mass conservation: Figures 2 and 3 show the plots of density as the first moment of the normalized distribution function  $f_g$  as obtained from the Eq. (12) at times 50 and  $100/\omega_{pe}$ , respectively, for a system 256-D lengths across (note: the time step in these simulations was  $0.1/\omega_{pe}$ ). The former corresponding to an earlier time clearly shows  $n_e \approx 1$  across the system. The latter though indicates enhanced oscillations corresponding to the Langmuir waves (following section) and possesses spatial average about the expected value of one. Next Fig. 4 also displays the electron density but at  $200/\omega_{pe}$ ; a comparison of this with Fig. 3 indicates the fluctuation levels must have saturated at roughly 25% of the background. Since we used 20 phase points per configuration mesh point, we would have expected fluctuation levels of  $1/\sqrt{20}$  analytically which is about 23%.

<sup>8</sup>When the Trapezoidal steps described below are eliminated, the total electron kinetic energy begins to rise at some point (linearly in time) and result in unphysical results.

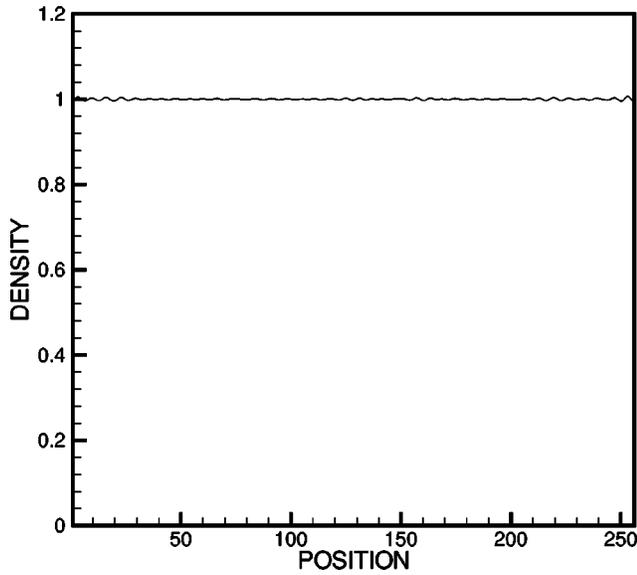


FIG. 2. The density versus position  $50/\omega_{pe}$  (plasma) periods after start of the simulation. Note the small amplitude Langmuir oscillations.

Here some comparison with PIC codes is in order. One of the most important functions of the representative phase points in the Vlasov code is that they maintain the distribution function information in addition to  $x(t)$  and  $v_x(t)$ . In a PIC code, the velocity moments in principle can be obtained directly from the discrete particle distributions in space, while in the Vlasov code moments are obtained from their distribution in phase space; i.e., particles all contribute equally to these moments regardless of their energy while phase points' contributions get weighted by their energy via the distribution function (see the equation for  $E_k$  below) (De-

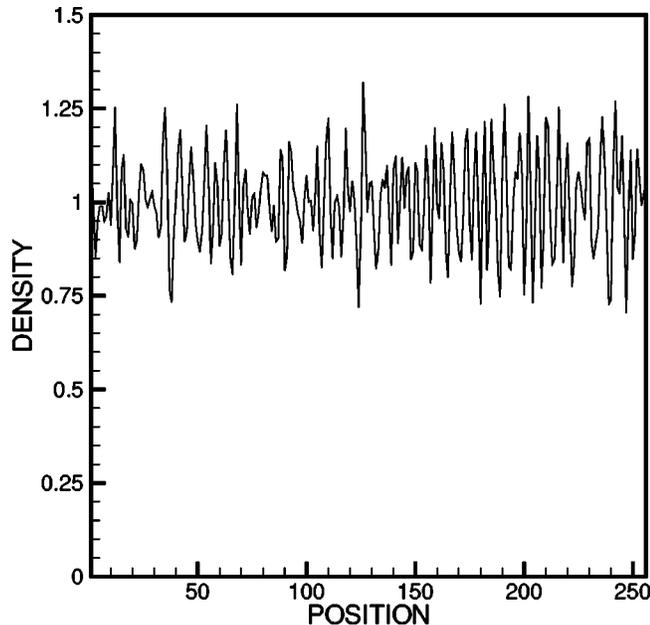


FIG. 3. The density versus position  $100/\omega_{pe}$  (plasma) periods after start of the simulation. Note the enhanced Langmuir oscillations.

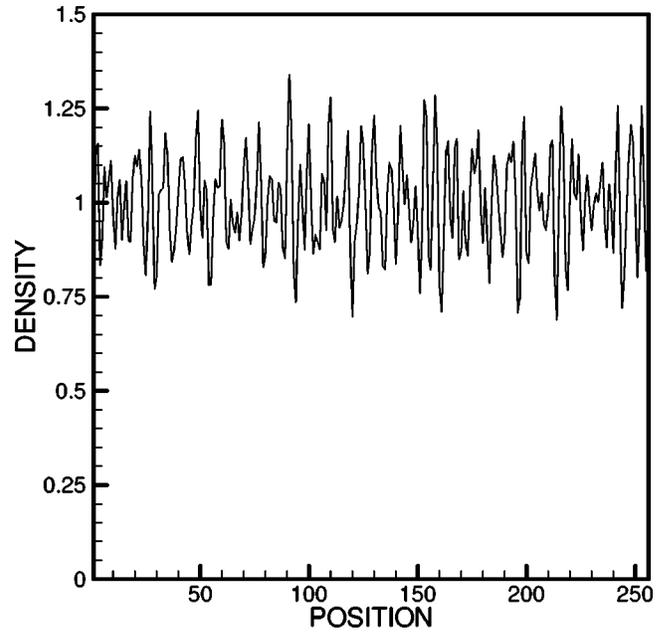


FIG. 4. The density versus position  $200/\omega_{pe}$  (plasma) periods after start of the simulation.

navit [25] demonstrated that Vlasov model's energy plots maintained less "spurious" oscillations). To see the difference, the electron density is also obtained here for the same system size and parameter range using a PIC code for comparison. Figure 5 shows the density 50-plasma periods into that simulation. Comparing this with Fig. 2 indicates a much smoother density in the former.

One of the most important properties of the Vlasov models lies in their ability to conserve the phase density  $f_g \Delta v \Delta x$ . This conservation is most fundamental since other moments conservation depends on it. The gradual distortion of  $f_g$  due to numerical errors will eventually break down a simulation. This issue is discussed at length in Appendix B.

Next comes momentum conservation. In this scheme, where there exists symmetry of interpolation between the representative phase points and the phase space mesh and centered spatial differencing is performed, there exist zero self-forces and momentum is therefore conserved to round off error [19]. Calculations of  $\langle v_x \rangle$ , as well as the internal energy serve as the best test of the momentum conservation which will be discussed below,  $\langle v_x \rangle = (1/n_e) \int f_g(v_x) v_x dv_x$  corresponds to the first velocity moment.

To determine the appropriate energy equations (kinetic, electrostatic, and total), we shall make use of the Vlasov equation corresponding to the electrons; i.e., in our normalizations [ $f = f(x, v_x, t)$ ],

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} - E_x \frac{\partial f}{\partial v_x} = 0. \quad (18)$$

The second moment of this equation after using the first two moments and Poisson's equation [Eq. (16)] gives rise to the following energy equation:

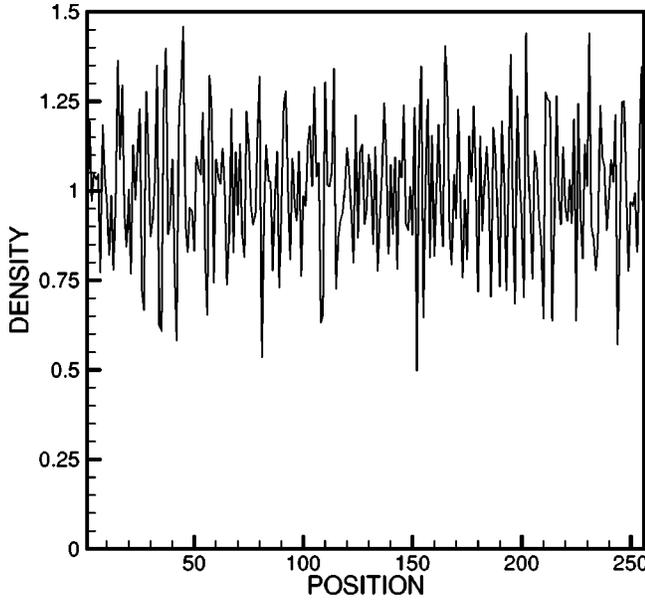


FIG. 5. The density versus position at  $50/\omega_{pe}$  (plasma) periods using a regular PIC code.

$$\frac{\partial}{\partial t} \left( \frac{1}{2} \int f v_x^2 dv_x + \frac{E_x^2}{2} \right) + \frac{\partial}{\partial x} \left( \frac{1}{2} \int f v_x^3 dv_x \right) = 0. \quad (19)$$

Integration of this equation over all space results in

$$\frac{\partial}{\partial t} \left[ \frac{1}{2} \int \int f v_x^2 dv_x dx + \frac{1}{2} \int E_x^2 dx \right] = 0. \quad (20)$$

The first term on the left is identified with the kinetic and the second with the electrostatic energy.

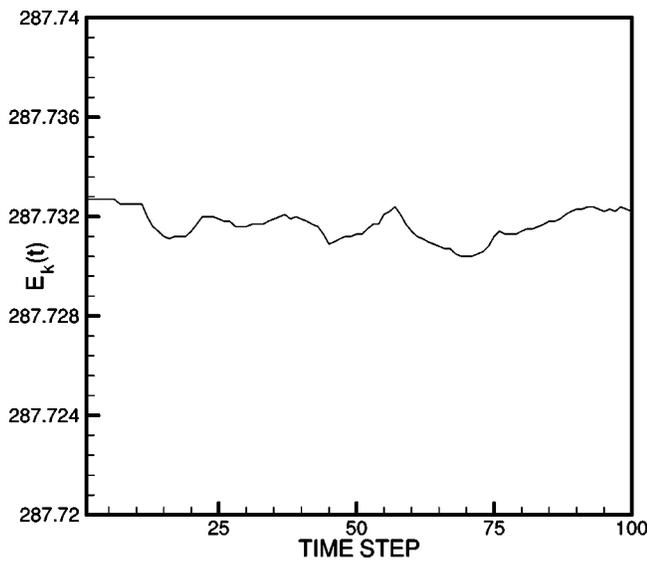


FIG. 6. The total electron kinetic energy versus normalized time during  $10/\omega_{pe}$  (plasma) periods, using fixed characteristics. The system size is 256-D lengths.

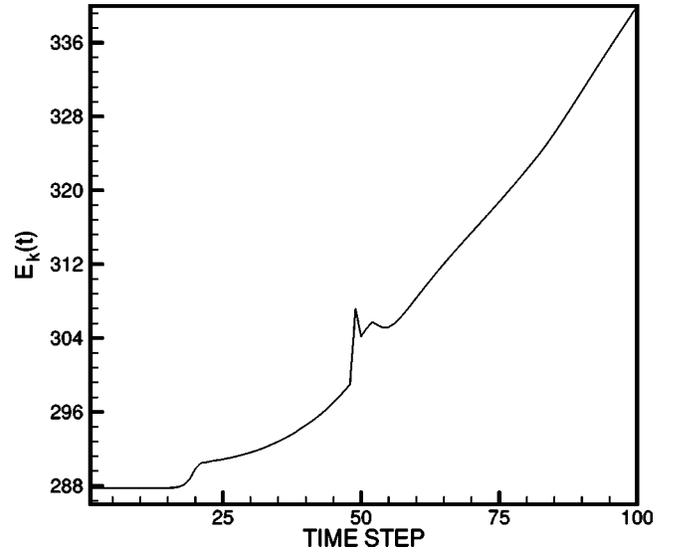


FIG. 7. The same as Fig. 6 but with reinitialized characteristics. Note the enhanced electron heating in this case.

With these then we can clearly define the necessary quantities for testing our model as follows; i.e., the following quantities are defined as the kinetic, electrostatic, total and internal energies:

$$E_k = \int \int \frac{1}{2} f_g(x, v_x) v_x^2 dv_x dx, \quad (21)$$

$$E_l = \int \frac{1}{2} E_x^2(x) dx, \quad (22)$$

$$E_t = E_k + E_l, \quad (23)$$

$$E_i = \int \int \frac{1}{2} f_g(x, v_x) (v_x - \langle v_x \rangle)^2 dv_x dx. \quad (24)$$

Here as in the density, the trapezoidal numerical integrations of these moments have been employed in the model. Extension to higher dimensions involves repetition of this procedure with other velocity components with similar results.

The very first energy tests involve the cases with the “fixed” and the “reinitialized” characteristics (last paragraph of the introduction); these tests are intended to: (i) justify the “fixed” versus the “reinitialized” characteristic approach due the intense artificial particle heating in the latter case; (ii) shed light on the impact of (i) versus parasitic (computational) modes on the energy conservation; and (iii) justify the Leapfrog-Trapezoidal as opposed to the simple Leapfrog scheme approach. The runs in these cases all correspond to systems 256-D lengths across. Figures 6 and 7 show  $E_k(t)$  for the cases with fixed and reinitialized characteristics, respectively, over a period of  $10/\omega_{pe}$ . Figures 8 and 9 show the electron kinetic energy [ $E_k(t)$ ] and the change in it [ $E_k(t) - E_k(0)$ ] versus time, respectively, when fixed characteristics were used. As Fig. 8 indicates, the rise in electron kinetic energy here over  $400/\omega_{pe}$  periods is comparable to that in Fig. 7 in only  $10/\omega_{pe}$  periods. Note that in both of

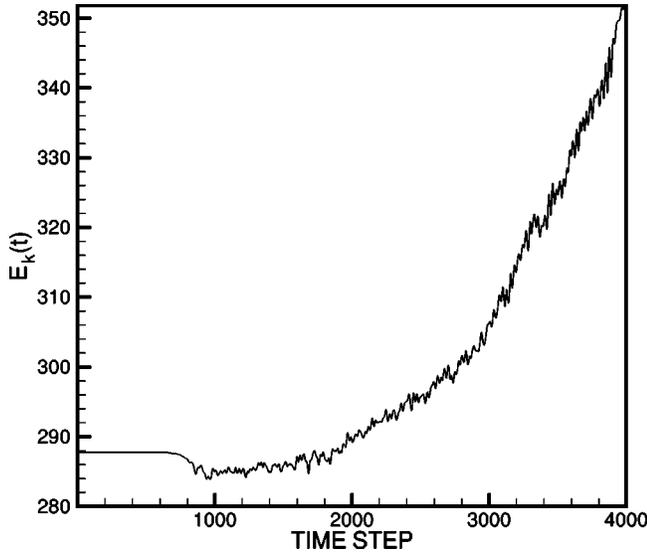


FIG. 8. The electron kinetic energy versus time  $400/\omega_{pe}$  (plasma) periods using fixed characteristics (the simple Leapfrog scheme was employed here).

these cases the simple Leapfrog and not the Leapfrog-Trapezoidal scheme was employed. Clearly, characteristic retention improved energy conservation and allowed longer time steps. We therefore believe the energy nonconservation due to characteristic reinitialization to be of truncation nature on the distribution function and to thus affect its moments including energy more severely than the interpolation procedures.<sup>9</sup> Indeed, in Ref. [25], weight functions of linear as well as quadratic orders in phase space when updating his distribution function information were attempted and it was concluded that it was not possible to derive weight functions for which all these moments were conserved for cases in which the distribution function was “reconstructed” at every time step.<sup>10</sup> In other words, the resulting diffusion from these “reconstructions” destroys too much information at the phase point distribution functions to allow even valid second moment (energy) conservation. Furthermore, a comparison of Figs. 9 and 13, where the Leapfrog scheme in the former and the Leapfrog-Trapezoidal scheme in the latter were used, indicate that in the former the parasitic (computational) modes begin to heat the particles after  $150/\omega_{pe}$  (note that Leapfrog-Trapezoidal scheme when trajectories were not fixed did not conserve energy). Using shorter time steps in the Leapfrog scheme did delay this heating though.<sup>11</sup> These results justify more noise free and longer time steps attributed to the Leapfrog-Trapezoidal scheme. The remaining fig-

<sup>9</sup>The energy nonconservation observed by [4] is therefore more associated with the characteristic reinitialization than the spatial charge deposition.

<sup>10</sup>Page 80 of the cited reference, paragraph one.

<sup>11</sup>The Appendix A includes extensive numerical analysis of both the Leapfrog versus Leapfrog-Trapezoidal schemes. There Courant conditions show much longer allowed time steps in the latter. The origin of the parasitic modes are shown. It is shown the latter scheme to damp these modes much better.

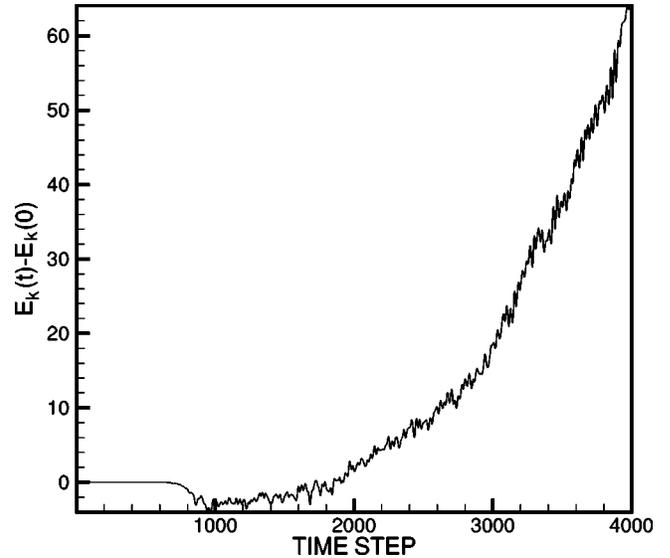


FIG. 9. Similar to Fig. 8 for the change in the electron kinetic energy versus normalized time. The system size is 256-D lengths in both cases.

ures in the paper all correspond to the “fixed characteristic” cases where Leapfrog-Trapezoidal scheme was employed.

Figures 10, 11, and 12 correspond to  $E_l(t)$ ,  $E_k(t) - E_k(0)$ , and  $E_t(t)$  versus time  $t$  for a system 16-D lengths across over  $400/\omega_{pe}$  periods. These clearly indicate that, in accordance with the expectations,  $E_k$  increases as  $E_l$  decreases, while  $E_t$  remains constant. The Figs. 13–15 likewise indicate similar patterns for a system 256-D lengths across; i.e., here also  $E_k$  decreases as  $E_l$  begins to rise. However, the total energy is not conserved as well as in the smaller system, primarily due to the usage of a coarser velocity grid; i.e., there exists an error of about 1.3% here, compared to less than 0.2% in the smaller system. However, in the larger system one observes the saturation of both the electron as well

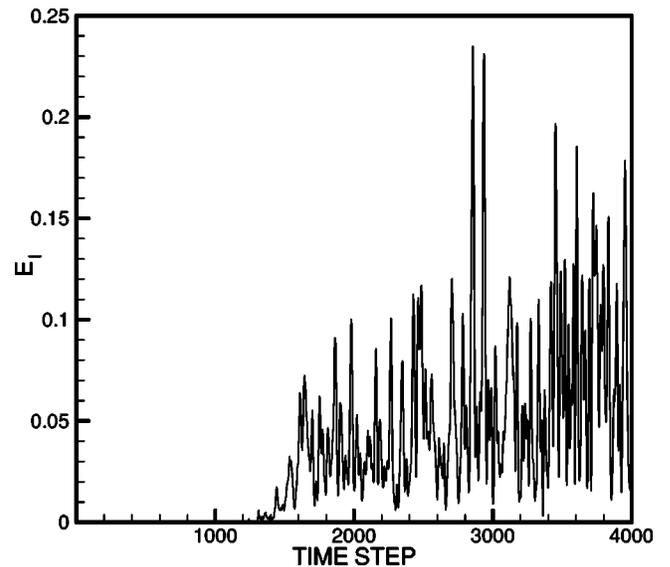


FIG. 10. The total electrostatic energy  $[E_l(t)]$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 16-D lengths.

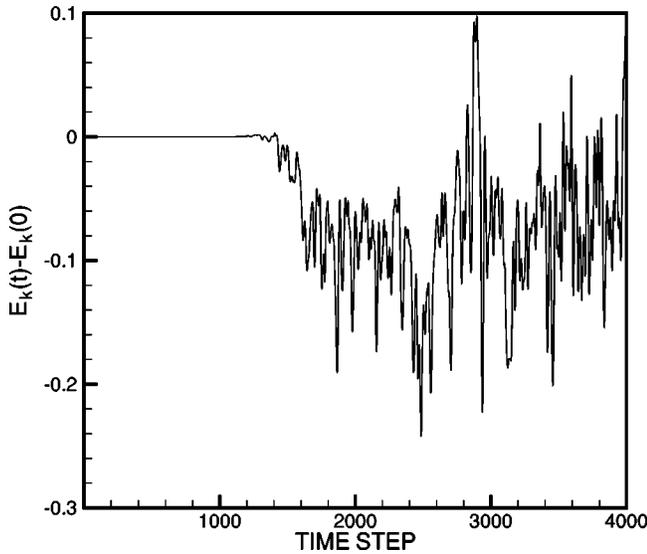


FIG. 11. The change in the total electron kinetic energy  $[E_k(t) - E_k(0)]$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 16-D lengths.

as the electric field energies much better. When the Leapfrog instead of the Leapfrog-Trapezoidal scheme was used, in both the small and the large system, after about  $150/\omega_{pe}$ ,  $E_k$  reversed its decrease and began to rise till the end of the simulations; i.e., it grew due to the computational mode discussed earlier. The rise in  $E_k$  was followed by a larger rise in  $E_l$ , and neither quantity indicated any saturation; a clearly unphysical behavior.

Figure 16 shows the time history of the internal energy  $E_i$  versus time for the same system size as Figs. 13–15. Here one observes an oscillatory behavior to appear about  $100/\omega_{pe}$  into the simulations and to persist with roughly the same amplitude throughout the simulation; i.e.,  $E_i(t)$  tends to os-

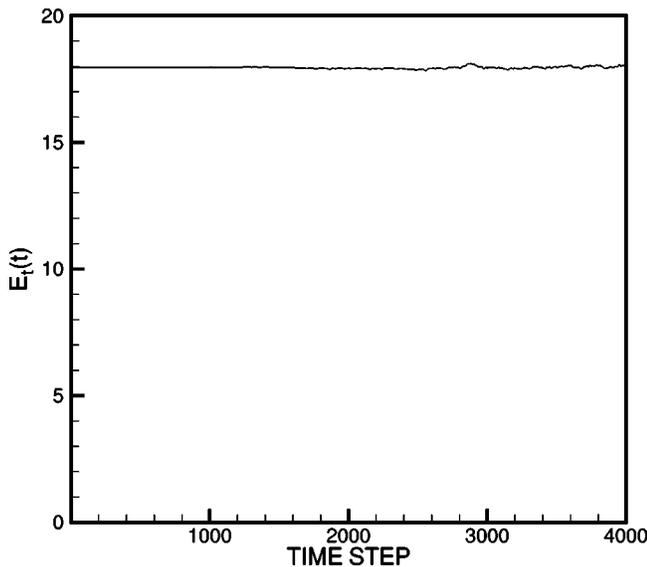


FIG. 12. The total energy  $E_i(t)$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 16-D lengths.

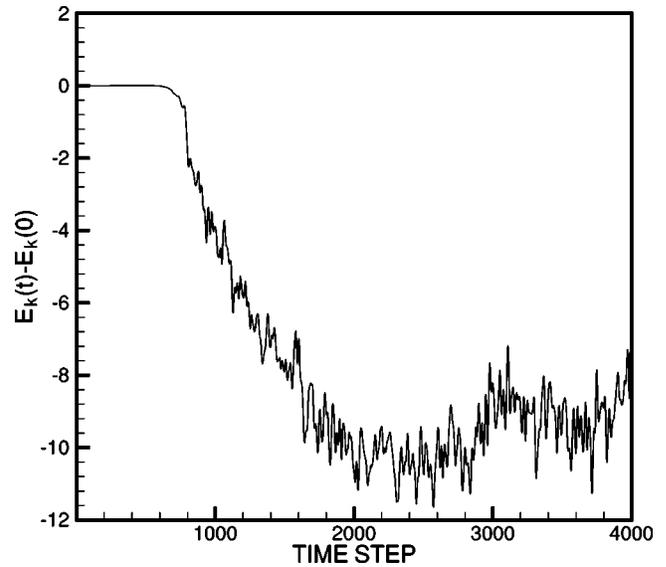


FIG. 13. The change in the total electron kinetic energy  $[E_k(t) - E_k(0)]$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 256-D lengths.

cillate about  $E_i(0)$ . This simply implies  $\langle v_x \rangle$  to average to zero within the simulation period. Indeed it was also observed that  $\langle v_x \rangle$  when computed for all the grid points, was quite negligible even at the end of the simulations. Negligible  $\langle v_x \rangle$  indicates momentum to be conserved within the roundoff errors. Momentum conservation implies no self-forces on the phase points, and further support the interpolation procedures employed. Also, when the Leapfrog instead of the Leapfrog-Trapezoidal scheme was used, it was observed that  $\langle v_x \rangle$  began to rise about the same time as when  $E_k$  reversed its decrease. Comparing the results of both schemes, we observed that the values of  $\langle v_x \rangle$  at the end of the simulations in the simple Leapfrog scheme were orders

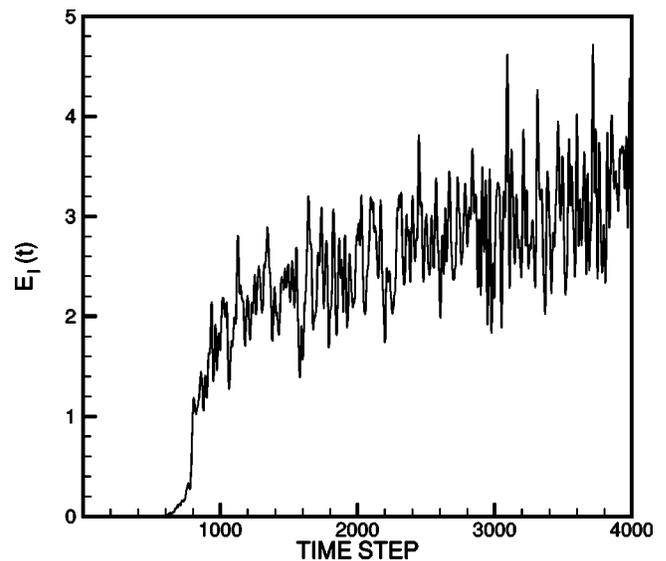


FIG. 14. The total electrostatic energy  $[E_l(t)]$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 256-D lengths.

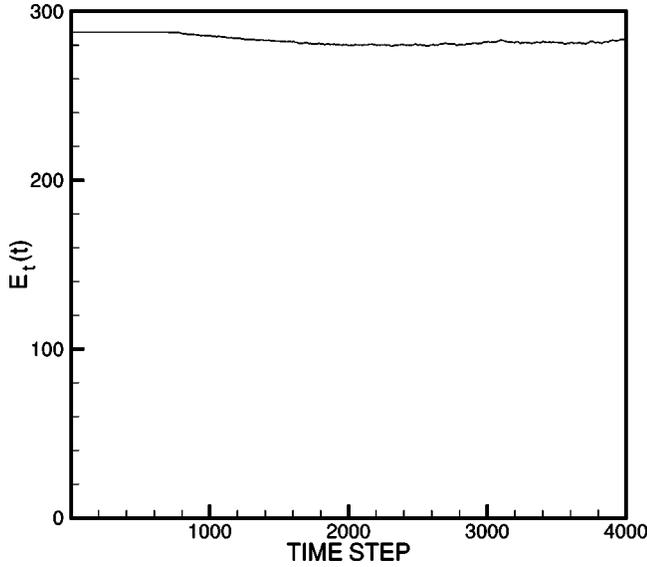


FIG. 15. The total energy  $E_t(t)$  versus normalized time in  $400/\omega_{pe}$  periods; the system size is 256-D lengths.

of magnitude larger than the corresponding ones in the Leapfrog-Trapezoidal scheme case; i.e., the latter scheme conserves momentum much better.

### B. Linear eigenmode analysis

The next test was checking for the dispersion of the Langmuir waves or Bohm-Gross relation [Eq. (7)] discussed earlier.

The fields and densities from the simulations were stored as the simulations of the thermal case<sup>12</sup> were proceeding; later their corresponding correlation functions as well as power spectra were determined by the techniques outlined in Refs. [20,21]. Figures 17 and 18 show two typical power spectra of the electric field corresponding to the modes one and seventeen, respectively {i.e.,  $k_x = 2\pi/L_x$  and  $k_x = [(2)(17)\pi]/L_x$ }. Here  $L_x$  represents the system size which in this case was 256-D lengths. Clear sharp peaks about the analytically expected values [Eq. (7)] correspond to the Langmuir waves. Figure 19 gives the simulation versus analytic results for twenty five modes. This plot shows very good agreement between theory and the simulations for  $k < 20$  (for  $k > 20$  the simulation results appear slightly nonlinear) and therefore demonstrates that the overall construction of the code (phase point interpolations, push schemes, Poisson solver) is working correctly and in good harmony. This figure does not show that much numerical dispersion; in a very dispersive code generally the simulation dispersion relation lies below the analytic one and for larger  $k$  the difference tends to get larger [22].

### C. Impact of the initial conditions on the beaming instability

The well-ordered discretized velocity distributions could give rise to beaming instability [23]; i.e., the velocity mesh

<sup>12</sup>When the phase point velocities were initialized to a Maxwellian distribution.

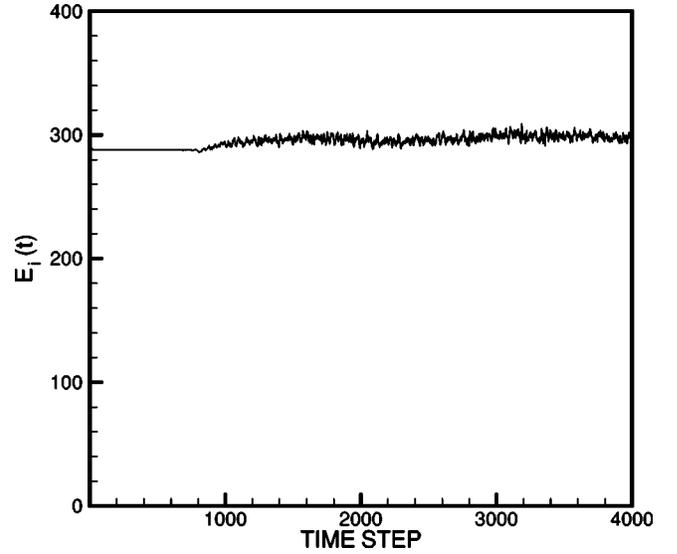


FIG. 16. The total electron internal energy  $E_i$  versus normalized time; the system size is 256-D lengths.

could resemble a set of cold “beams.” The results of the preceding section did not indicate any such instability to take place in the thermal case, as the energy as well as power spectra plots indicated.

Therefore, in order to trigger the instability, the initial distribution was changed to allow the so-called equal density “beams” with infinitesimal perturbations from the equilibrium as in Ref. [25]; i.e., the phase points distribution were therefore changed as follows:

$$f_p(x, v, t=0) = \frac{1}{v_{th}^3 \sqrt{2\pi}} v^2 e^{-v^2/2v_{th}^2} [1 + 2\epsilon \cos(2\pi x)], \quad (25)$$

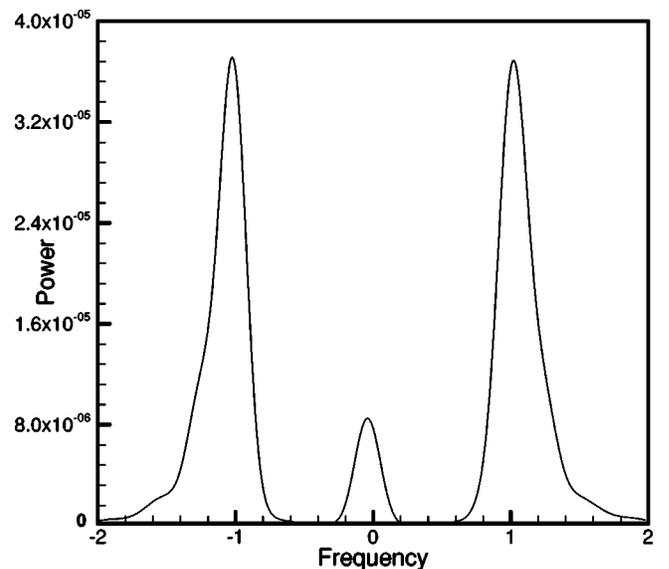


FIG. 17. The power spectrum for the  $E_x(k=1)$  mode; the system size is 256-D lengths. The zero frequency mode could indicate nonlinear behavior.

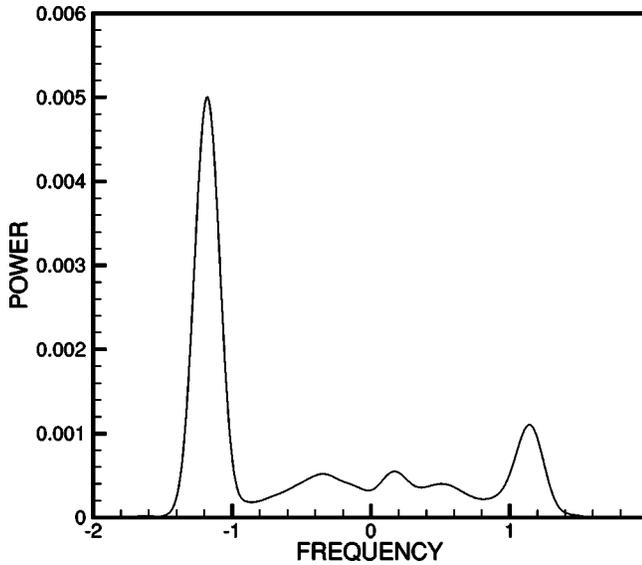


FIG. 18. The power spectrum for the  $E_x(k=17)$  mode; the system size is 256-D lengths. Different heights could arise from the numerical noise.

where,  $v_{th}=0.3/\pi$ ,  $\epsilon=0.0025$ , and the system size was 256-D lengths. The linear growth rates according to [24] show only the first mode to be unstable. So here plots of  $E_x^2(k_x, t)$  and the velocity of the central “beam” [i.e., where the phase points initially have velocity  $v_x \in (0, \Delta v)$ ] should reveal if such instability can take place.

Indeed, Denavit [25] shows that at the onset of the instability the electric field locally would grow gently first, then drop suddenly only to reach later a maximum 23 times its initial value. He also showed that the central “beam” velocity followed a similar pattern. Here, the results of the simulations are shown in Figs. 20–22. Neither one of these figures shows these patterns. The electrostatic mode energy (Fig. 20) simply indicates normal growth in the amplitudes of the Langmuir waves some 15-plasma periods into the simulations followed by saturation. The central “beam” velocity plot in Fig. 22 shows a similar pattern as the energy plot, and they both rule out the beaming instability of the nature predicted by Ref. [23]. Another interesting point is that the energy plot (Fig. 20) does not show any spurious

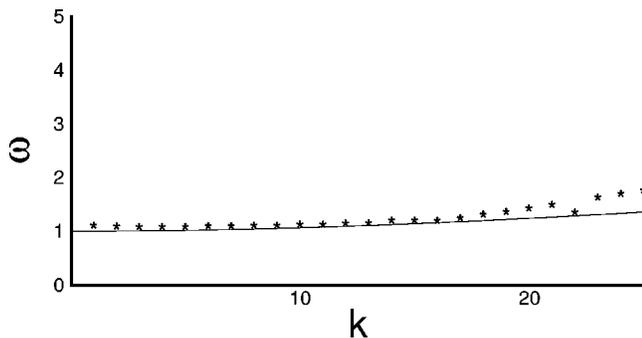


FIG. 19. The simulation (asterisks) versus analytic (solid curve) Bohm-Gross dispersion relation for the first twenty five harmonics. Nonlinear effects could be the cause of discrepancies for  $k > 20$ .

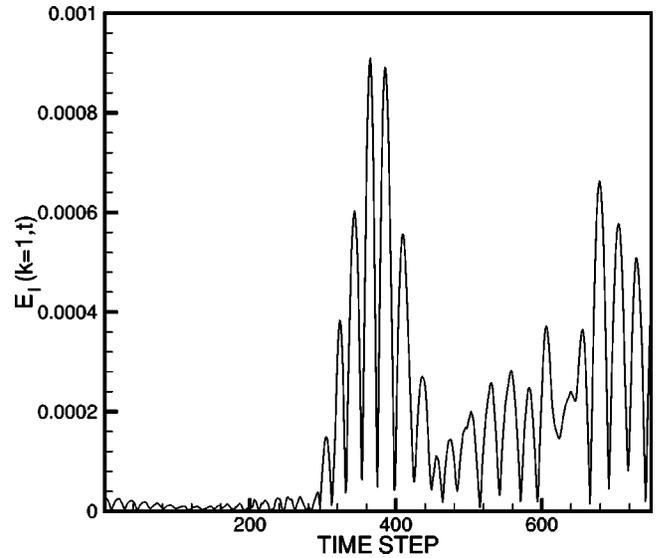


FIG. 20. The electrostatic energy in the first mode versus normalized time later in the simulation; i.e.,  $E_x^2(k_x=1, t)$  versus time step.

oscillations (i.e., oscillations with no physical origin and of purely numerical nature); i.e., peaks resulting from the presumed numerical instability. Indeed [25] while comparing the results of these simulations from PIC versus Vlasov models did show that the PIC simulations possess such oscillations.

There are reasons why the Vlasov solvers can prevent numerical beaming instability when quiet starts are implemented. Denavit [25] claims that the constancy of the phase density prevents the phase points from moving into regions where the “distribution function is zero initially” and that would in turn prevent spurious charge densities and numerical instabilities to arise. Another important factor could arise

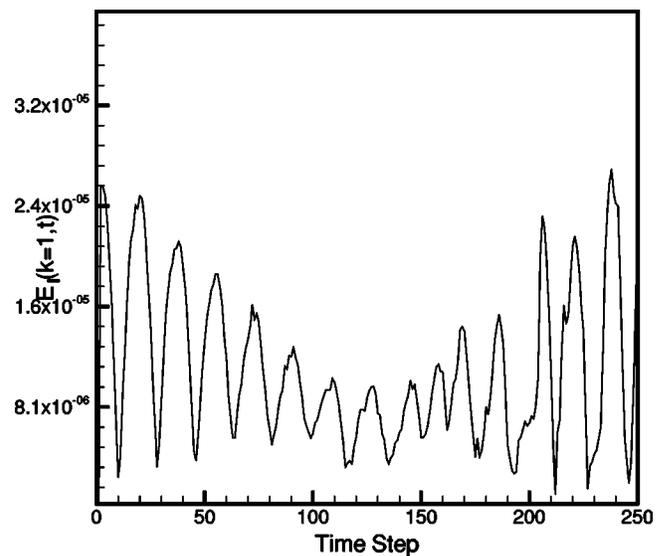


FIG. 21. The electrostatic energy in the first mode versus normalized time early in the simulation; i.e.,  $E_x^2(k_x=1, t)$  versus time step.

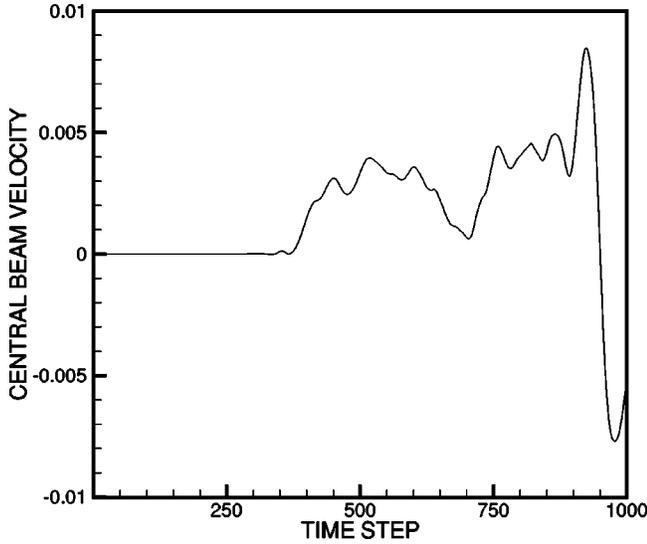


FIG. 22. The central beam velocity versus normalized time.

from the fact that the density here results from moments of the distribution function [Eq. (12)] and not just interpolations in configuration space as in the PIC codes. The contributions of the phase points are therefore weighted by the factor  $e^{-v^2/v_{th}^2}$  which substantially reduces the impact of the phase points with  $v \gg v_{th}$  on charge density.

The results here as well as Sec. III B above also do not indicate any significant diffusion. As Fig. 21 shows, the mode energy versus time early in these simulations do not show a systematic decrease and flattening; i.e., signatures of diffusion in the scheme and in particular in the distribution function. Denavit [25] shows examples of diffusive schemes (i.e., when the distribution function is remapped in every time step), where the electrostatic mode energy flattens very quickly after a few Langmuir oscillations.

## V. CONCLUSION

It was shown that, by extending interpolation in configuration space to one in phase space, one can enhance PIC simulation results to include the knowledge of distribution function. Retaining characteristics at the phase points minimized diffusion, while reconstructing the distribution function at a fixed phase space mesh eliminated spurious beaming instability which could arise from the quiet starts. Excellent energy conservation resulted without employing particle shape functions.

The applications of this model are countless. Here we cite a few: initialization of each species to any distribution within any given scale; imposition of quiet starts with relative ease as a result of the preceding item; the temporal knowledge of the distribution function with applications to transport and when chaotic and or ambient particle fluxes impact plasmas in equilibrium; and the impact of particle and or heat fluxes from boundaries into bounded plasmas. The PIC simulation results can, therefore, be considerably enriched and one can also reduce the fluid from the kinetic results as well as provide accurate descriptions of the macroscopic parameters

with relative ease by the moments of the distribution function.

## ACKNOWLEDGMENTS

This work was partly supported by the Austrian Research Fund (FWF) Projects Nos. P12477-TPH and p15013 and partly by the Iranian National Council For Research Grant No. 3663. The first author wishes to thank Dr. H. Hakimi-pazhoo of IPM for useful comments and great help in preparing most figures.

## APPENDIX A: NUMERICAL STABILITY ANALYSIS

In this appendix we shall derive the Courant condition and physical as well as computational modes of Leapfrog as well as Leapfrog-Trapezoidal schemes for our Vlasov-Maxwell system. Here we followed the methods in Ref. [19].

### 1. Leapfrog scheme

The basic equations of the Vlasov-Maxwell system are

$$\frac{dv_x}{dt} = -\frac{e}{m}E_x, \quad (\text{A1})$$

$$\frac{dx}{dt} = v_x, \quad (\text{A2})$$

$$\frac{dE_x}{dx} = 4\pi e(n_0 - n_e). \quad (\text{A3})$$

The difference versions of the Eqs. (A1) and (A2) when Leapfrog scheme is used (steps 3 and 4 of Table I) are

$$\frac{v_x^n - v_x^{n-1}}{\Delta t} = -\frac{e}{m}E_x^{n-1/2}, \quad (\text{A4})$$

$$\frac{x^{n+1/2} - x^{n-1/2}}{\Delta t} = v_x^n. \quad (\text{A5})$$

Let  $(x^{n+1/2}, v_x^n)$  denote the numerical and  $(X^{n+1/2}, V_x^n)$  the exact solution of the phase space coordinates at time level  $t^n$  and  $(\epsilon_x^{n+1/2}, \epsilon_v^{n+1/2})$  the corresponding errors between them, i.e.,

$$\epsilon_v^n = v_x^n - V_x^n, \quad (\text{A6})$$

$$\epsilon_x^{n+1/2} = x^{n+1/2} - X^{n+1/2}. \quad (\text{A7})$$

Using Eqs. (A6) and (A7) in Eqs. (A4) and (A5) will result in the following pair, respectively:

$$\frac{\epsilon_v^n - \epsilon_v^{n-1}}{\Delta t} = -\frac{e}{m} \frac{\partial E_x^{n-1/2}}{\partial x} \epsilon_x^{n-1/2} = -\omega_{pe}^2 \epsilon_x^{n-1/2}, \quad (\text{A8})$$

$$\frac{\epsilon_x^{n+1/2} - \epsilon_x^{n-1/2}}{\Delta t} = \epsilon_v^n. \quad (\text{A9})$$

Combining these equations gives rise to the following equation:

$$\frac{\epsilon_x^{n+1/2} - 2\epsilon_x^{n-1/2} + \epsilon_x^{n-3/2}}{(\Delta t)^2} = -\omega_{pe}^2 \epsilon_x^{n-1/2}. \quad (\text{A10})$$

This equation simply corresponds to the phase points undergoing oscillations at  $\omega_{pe}$ . In order to carry out the stability analysis, the amplification factor  $\lambda = e^{-i\omega\Delta t}$  is assumed in advancing  $\epsilon_x$ ; i.e.,

$$\epsilon_x^{n+1/2} = \lambda \epsilon_x^{n-1/2} \quad (\text{A11})$$

with the constraint

$$\left| \frac{\epsilon_x^{n+1/2}}{\epsilon_x^{n-1/2}} \right| \leq 1. \quad (\text{A12})$$

Using Eq. (A11) in (A10) gives

$$\frac{e^{-i\omega\Delta t} - 2 + e^{i\omega\Delta t}}{(\Delta t)^2} = -\omega_{pe}^2. \quad (\text{A13})$$

This equation is identical to

$$\frac{e^{-i\omega\Delta t/2} - e^{i\omega\Delta t/2}}{\Delta t} = \pm i\omega_{pe}, \quad (\text{A14})$$

which simply gives rise to a first order difference equation in  $\epsilon_x$  as follows:

$$\frac{\epsilon_x^{n+1/2} - \epsilon_x^{n-1/2}}{\Delta t} = \pm i\omega_{pe} \epsilon_x^n; \quad (\text{A15})$$

in principle this equation should therefore possess only one physically admissible solution.

To determine the numerical solutions from  $\lambda$ , using  $\Delta t_1 = \Delta t/2$  in Eq. (A14) results in the following quadratic equation:

$$\lambda^2 + 2i\omega_{pe}\Delta t_1\lambda - 1 = 0, \quad (\text{A16})$$

whose solutions are

$$\lambda_r = \pm \sqrt{1 - \omega_{pe}^2 \Delta t_1^2}, \quad (\text{A17})$$

$$\lambda_i = -\omega_{pe} \Delta t_1, \quad (\text{A18})$$

for  $\omega_{pe} \Delta t_1 \leq 1$ , and

$$\lambda_r = 0, \quad (\text{A19})$$

$$\lambda_i = -\omega_{pe} \Delta t_1 \pm \sqrt{\omega_{pe}^2 \Delta t_1^2 - 1}, \quad (\text{A20})$$

for  $\omega_{pe} \Delta t_1 \geq 1$ . In each case then we have two solutions  $\lambda_-$  and  $\lambda_+$ . Courant condition demands both  $\lambda_+$  and  $\lambda_-$  to obey

$$|\lambda_{\pm}| \leq 1. \quad (\text{A21})$$

This then simply rules out Eq. (A20) or  $\omega_{pe} \Delta t_1 \geq 1$  and the Courant stability criterion for the Leapfrog scheme case is therefore

$$\omega_{pe} \Delta t_1 \leq 1. \quad (\text{A22})$$

However, even here we have two solutions  $\lambda_-$  and  $\lambda_+$ . It can be seen though that  $\lambda_+$  corresponds to the physical solution and  $\lambda_-$  is nowhere close to the behavior of the analytic solution. Such an unphysical solution is termed ‘‘computational’’ or ‘‘parasitic.’’ The cause of the ‘‘parasitic’’ mode is due to the usage of a two time level scheme for a first order differential equation, [15]. (This reference also refers to others emphasizing this point.)

The computational mode causes eventual deviation of the numerical solution from the analytic one. The main remedy to counter this effect is to devise schemes which damp these modes. In this example we can clearly see that

$$|\lambda_+| = |\lambda_-| = 1, \quad (\text{A23})$$

i.e., in the Leapfrog scheme, computational modes have the same unitary amplification factors as the physical modes. The general solution though is determined from a linear combination of  $\lambda_+$  and  $\lambda_-$ , and the unphysical part will contribute to errors in each time step which will accumulate in time.

We shall next show how computational modes can cause deviations from correct ones even when they are initialized at a small value and assumed not to grow relatively to the physical modes with time. Consider  $\lambda = e^{-i\omega\Delta t}$  and Eqs. (A17) and (A18); i.e., we obtain ( $\omega = \omega_R + i\omega_I$ ),

$$\omega_R \Delta t_1 = \pm \tan^{-1} \left( \frac{\omega_{pe} \Delta t_1}{\sqrt{1 - \omega_{pe}^2 \Delta t_1^2}} \right), \quad (\text{A24})$$

$$\omega_I \Delta t_1 = 0, \quad (\text{A25})$$

with the plus and minus corresponding to the physical and computational solutions, respectively. Equation (A25) simply indicates that neither mode grows as expected since otherwise numerical instabilities will result; they are not damped either. So if  $\omega_R \Delta t_1$  represents the physical solution,  $-\omega_R \Delta t_1$  will correspond to the computational one and the general solution can be written as follows:

$$\begin{aligned} \lambda &= p_1 \lambda_+ + p_2 \lambda_- = p_1 e^{-i\omega_R \Delta t_1} + p_2 e^{i\omega_R \Delta t_1} \\ &= p_1 \left( e^{-i\omega_R \Delta t_1} + \frac{p_2}{p_1} e^{i\omega_R \Delta t_1} \right), \end{aligned} \quad (\text{A26})$$

assuming  $p_2 \ll p_1$  with  $p_1 + p_2 = 1$ . Now using Eq. (A11), it follows by induction that if  $p_1$  and  $p_2$  do not change with time, we have

$$\begin{aligned} \epsilon_x^{n+1/2} &= \lambda^n \epsilon_x^{n-1/2} = p_1^n e^{-i\omega_R n \Delta t_1} \left( 1 + \frac{p_2}{p_1} e^{2i\omega_R \Delta t_1} \right)^n \epsilon_x^{n-1/2} \\ &\approx p_1^n e^{-i\omega_R n \Delta t_1} \left( 1 + \frac{n p_2}{p_1} e^{2i\omega_R \Delta t_1} \right) \epsilon_x^{n-1/2}. \end{aligned} \quad (\text{A27})$$

In that case since the correct solution is

$$\epsilon_{xc}^{n+1/2} = e^{-i\omega_R n \Delta t_1} \epsilon_x^{-1/2}, \quad (\text{A28})$$

the percentage error can be determined to be roughly

$$\frac{\epsilon_x^{n+1/2}}{\epsilon_{xc}^{n+1/2}} = p_1^n \left( 1 + \frac{np_2}{p_1} \right). \quad (\text{A29})$$

For example if  $p_1 = 0.9999$  and  $p_2 = 0.0001$ , then from this equation one can see that one will have 5.3% error in 10 000 time steps. This error will certainly rise if  $p_2$  rises relative to  $p_1$  in time; i.e., the computational mode grows relative to the physical mode. These results then demonstrate how the computational mode can cause substantial deviations of the solutions after some time even under the most ideal circumstances. Therefore, any scheme which damps  $\lambda_-$  would improve the outcome by lessening those errors.

Another remedy is the usage of the backward biasing in the Leapfrog scheme. Using the same methods as in this section and the following, one can show that the corresponding Courant condition in this case is more restrictive ( $\omega_{pe} \Delta t_1 \leq 0.8$ ) than both this case as well as Leapfrog-Trapezoidal scheme to be discussed next.

## 2. Leapfrog-Trapezoidal scheme

From steps 8 and 9 in Table I, the following pair of equations are obtained after following the procedures of the last section (note that asterisks used for clarity are dropped here):

$$\epsilon_v^n - \epsilon_v^{n-1/2} = -\frac{\omega_{pe}^2 \Delta t}{2} (\epsilon_x^n + \epsilon_x^{n-1/2}), \quad (\text{A30})$$

$$\epsilon_x^{n+1/2} - \epsilon_x^n = \frac{1}{2} (\epsilon_v^{n+1/2} + \epsilon_v^n) \frac{\Delta t}{2}. \quad (\text{A31})$$

From Eq. (A31) we obtain

$$\epsilon_v^n = \frac{4}{\Delta t} (\epsilon_x^{n+1/2} - \epsilon_x^n) - \epsilon_v^{n+1/2}. \quad (\text{A32})$$

Using Eq. (A32) in Eq. (A30) results in

$$\begin{aligned} & \frac{4}{\Delta t} (\epsilon_x^{n+1/2} - 2\epsilon_x^n + \epsilon_x^{n-1/2}) - (\epsilon_v^{n+1/2} - \epsilon_v^n) \\ &= -\frac{\omega_{pe}^2 \Delta t}{2} (\epsilon_x^n + \epsilon_x^{n-1/2}). \end{aligned} \quad (\text{A33})$$

Now using Eq. (A30) in Eq. (A33) results in one equation involving  $\epsilon_x$  only, i.e.,

$$\frac{4}{\Delta t} (\epsilon_x^{n+1/2} - 2\epsilon_x^n + \epsilon_x^{n-1/2}) = -\omega_{pe}^2 \frac{\Delta t}{4} (\epsilon_x^{n+1/2} + 2\epsilon_x^n + \epsilon_x^{n-1/2}). \quad (\text{A34})$$

Using  $\epsilon_x^{n \pm 1/2} = e^{\pm(-i\omega \Delta t/2)} \epsilon_x^n$  in Eq. (A34) results in

$$\begin{aligned} & \frac{e^{-i\omega(\Delta t/2)} - 2 + e^{i\omega(\Delta t/2)}}{(\Delta t^2/4)} \epsilon_x^n \\ &= -\frac{\omega_{pe}^2}{4} [e^{-i\omega(\Delta t/2)} + 2 + e^{i\omega(\Delta t/2)}] \epsilon_x^n \end{aligned} \quad (\text{A35})$$

from which the following dispersion relation is obtained:

$$\frac{e^{-i(\omega \Delta t/4)} - e^{i(\omega \Delta t/4)}}{\Delta t/2} = \pm \frac{i\omega_{pe}}{2} [e^{-i(\omega \Delta t/4)} + e^{i(\omega \Delta t/4)}]. \quad (\text{A36})$$

Now if we multiply both sides of this equation by  $e^{(-i\omega \Delta t/4)} \epsilon_x^n$  we obtain the following difference equation in  $\epsilon_x$ :

$$\frac{\epsilon_x^{n+1/2} - \epsilon_x^n}{\Delta t/2} = \pm \frac{i\omega_{pe}}{2} (\epsilon_x^{n+1/2} + \epsilon_x^n). \quad (\text{A37})$$

So here also the main trapezoidal steps reduce to an ordinary difference equation too. So the combined Leapfrog scheme [Eq. (A15)] and its associated Trapezoidal step [Eq. (A37)] will give rise to the following pair:

$$\frac{\epsilon_x^{n+1/2} - \epsilon_x^{n-1/2}}{\Delta t} = \pm i\omega_{pe} \epsilon_x^n, \quad (\text{A38})$$

$$\frac{\epsilon_x^{n+1/2} - \epsilon_x^n}{\Delta t/2} = \pm \frac{i\omega_{pe}}{2} (\epsilon_x^{n+1/2} + \epsilon_x^n), \quad (\text{A39})$$

with the understanding that the  $\epsilon_x^{n+1/2}$  obtained from the first equation is to be used on the right-hand side of the second equation.

Here again using the same procedures as in the Leapfrog scheme we arrive at the following equation for the amplification factor  $\lambda$ :

$$\lambda - 1 = \frac{-i\omega_{pe} \Delta t_1}{2\lambda} - \omega_{pe}^2 \Delta t_1^2 - \frac{i\omega_{pe} \Delta t_1}{2}. \quad (\text{A40})$$

Direct numerical solution of this equation with the constraint  $|\lambda| \leq 1$  also gives the following Courant condition:

$$\omega_{pe} \Delta t_1 \leq \sqrt{2}. \quad (\text{A41})$$

Table II clearly illustrates this among other things. Here as in the Leapfrog scheme case,  $|\lambda_+|$  corresponds to the physical while  $|\lambda_-|$  to the computational modes.

As we see from the table, up to  $\omega_{pe} \Delta t_1 \leq 1.4$  we get admissible solutions; for  $\omega_{pe} \Delta t_1 = 1.5$ ,  $|\lambda_-| > 1$  and  $|\omega_+ \Delta t|$  is incorrect. For almost all the modes also  $|\lambda_-| < |\lambda_+|$ ; i.e., the computational mode is damped and this is more the case for the smaller time steps where damping is substantial. This is to be contrasted with the Leapfrog scheme case where these amplification factors were equal for all allowed modes. Therefore, Leapfrog-Trapezoidal scheme allows longer time steps and considerably smaller computational modes compared to the Leapfrog scheme. These should justify the added computational effort.

TABLE II. The physical versus computational modes eigenvalues for different time steps.

$\omega_p \Delta t_1$	$ \lambda_+ $	$\omega_+ \Delta t_1$	$ \lambda_- $	$\omega_- \Delta t_1$	$P.E.^a$
0.1	1.0000	-0.0999	0.0500	-1.4710	1.00
0.2	0.9996	-0.1995	0.1000	-1.3720	0.25
0.3	0.9983	-0.2985	0.1503	-1.2722	0.50
0.4	0.9953	-0.3974	0.2009	-1.1733	0.65
0.5	0.9901	-0.4969	0.2525	-1.0740	0.62
0.6	0.9821	-0.5975	0.3055	-0.9730	0.42
0.7	0.9708	-0.7003	0.3605	-0.8710	0.04
0.8	0.9559	-0.8059	0.4184	-0.7650	0.73
0.9	0.9368	-0.9154	0.4804	-0.6552	1.71
1.0	0.9124	-1.0299	0.5480	-0.5410	2.29
1.1	0.8814	-1.1504	0.6240	-0.4204	4.58
1.2	0.8415	-1.2782	0.7130	-0.2926	6.52
1.3	0.7889	-1.4133	0.8239	-0.1575	8.72
1.4	0.7187	-1.5520	0.9739	-0.0191	10.85
1.5	0.7726	-0.7648	1.8202	+0.1184	48.87

<sup>a</sup>*P.E.*: Percentage error between  $\omega_p \Delta t_1$  and  $|\omega_+ \Delta t_1|$  (the physical mode).

## APPENDIX B: DISTORTION OF THE DISTRIBUTION FUNCTION

This phenomenon is attributed to the process of divergence of characteristics due to the scatter gather operations in the force equation (involving  $f_p$ 's indirectly) with the net long term impact of numerical diffusion of  $f_g$  in phase space. This process gives rise to the phase density nonconservation and thus results in the numerical heating of the phase points. The PIC models also experience numerical heating [19] (see the section on heating time). In this appendix we obtain a gross estimate of the numerical heating attributed to this process in phase space and suggest ways to improve.

The process is attributed to the combined gather-scatter operations in interpolating  $f_p$  to its host cell for the charge calculation in the force equation. Therefore, although  $f_p$  associated with the phase points does not change with time, its mere interpolation to a fixed mesh (scatter operation and  $f_g$  calculation) and gathering the electric field in the force calculation (which depends on  $f_g$  through the charge density  $n_e$ ) will have the impact of smearing  $f_p$  at that phase point for that calculation. To see this, note that the actual force at the phase point should be proportional to  $f_p \Delta t$  in principle, but in practice it becomes proportional to  $f_p^* \Delta t$  where  $f_p^*$  is defined to result from the scatter gather operations on  $f_p$ . This causes trajectory errors in each time step which can result in the divergence of characteristics with a net long term diffusive impact on  $f_g$ .

Therefore, in order to determine its impact, one should estimate the net smearing impact of the scatter gather operations on the force for each phase point by obtaining an expression for  $f_p^*$ . In order to see this please refer to Fig. 1 part (c). The phase points within such a cell assign their associated  $f_p$ 's to this cell's grid points  $l_x - 1$  and  $l_x$  by the weights  $N_1$  and  $N_2$ , respectively; those grid points receive the following contributions respectively (scatter operation):

$$f_1 = \sum_{\forall p} N_1 f_p, \quad (\text{B1})$$

$$f_2 = \sum_{\forall p} N_2 f_p. \quad (\text{B2})$$

In the accompanying gather operation (e.g., in force calculation), an arbitrary phase point in that cell should receive the following contributions from those two grid points:

$$f_p^* = N_1 f_1 + N_2 f_2. \quad (\text{B3})$$

In practice then the associated gathered force at that phase point will depend on this quantity and not  $f_p$ . To obtain a gross estimate of  $f_p^*$ , we observe (please see below) that this error is most severe when  $N_1 = N_2 = 1/2$ ; i.e., when a phase point is exactly half way inside a cell. At that location,

$$\begin{aligned} f_p^* &= \frac{f_1 + f_2}{2} + f_p - f_p = f_p + \frac{f_1 - 2f_p + f_2}{2} \\ &= f_p + \frac{1}{2} \left( \frac{\Delta x}{2} \right)^2 \frac{\partial^2 f_p}{\partial x^2} + O(\Delta x^2). \end{aligned} \quad (\text{B4})$$

It can be shown that for other phase point locations, the  $1/2$  in front of the second quantity on the right-hand side should be replaced by  $(x_p/\Delta x)$ , where  $x_p$  is the distance of the phase point to its nearest grid point. This result [Eq. (B4)] can be generalized to  $2d$  in phase space if one considers a phase point at the center of the square cell in Fig. 1 part (b) and compute the net impact of the scatter-gather operations as above; i.e., in  $2d$ :

$$f_p^* = f_p + \left( \frac{\Delta x}{2} \right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial x^2} + \left( \frac{\Delta v}{2} \right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial v^2} + O(\Delta x^2 \Delta v^2). \quad (\text{B5})$$

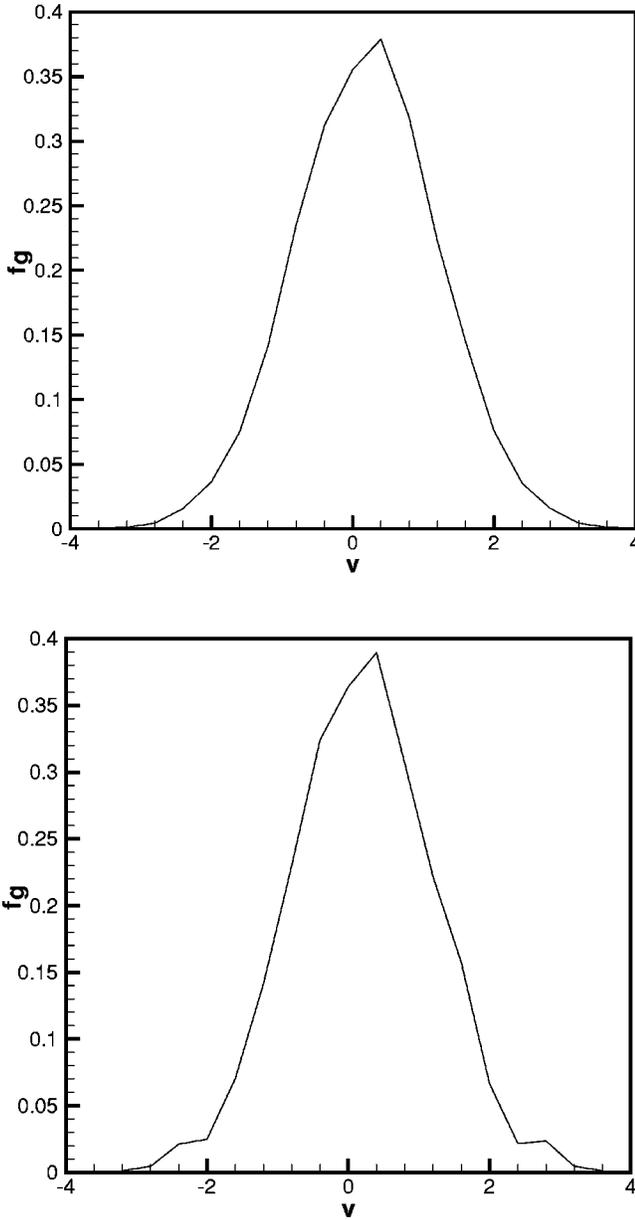


FIG. 23. Spatially averaged  $f_g$  for the system 256-D lengths across with  $\Delta v=0.4$  at normalized times 100 and 200  $/\omega_{pe}$ , respectively.

Sakanaka *et al.* [4] also reported errors of the same order which they termed numerical “diffusion.”

From these results, we conclude the following maximum limits of the error of gathering  $f_p$  per time step; i.e.,

$$(\Gamma_x)_{max} = \left(\frac{\Delta x}{2}\right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial x^2}, \quad (\text{B6})$$

$$(\Gamma_v)_{max} = \left(\frac{\Delta v}{2}\right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial v^2}. \quad (\text{B7})$$

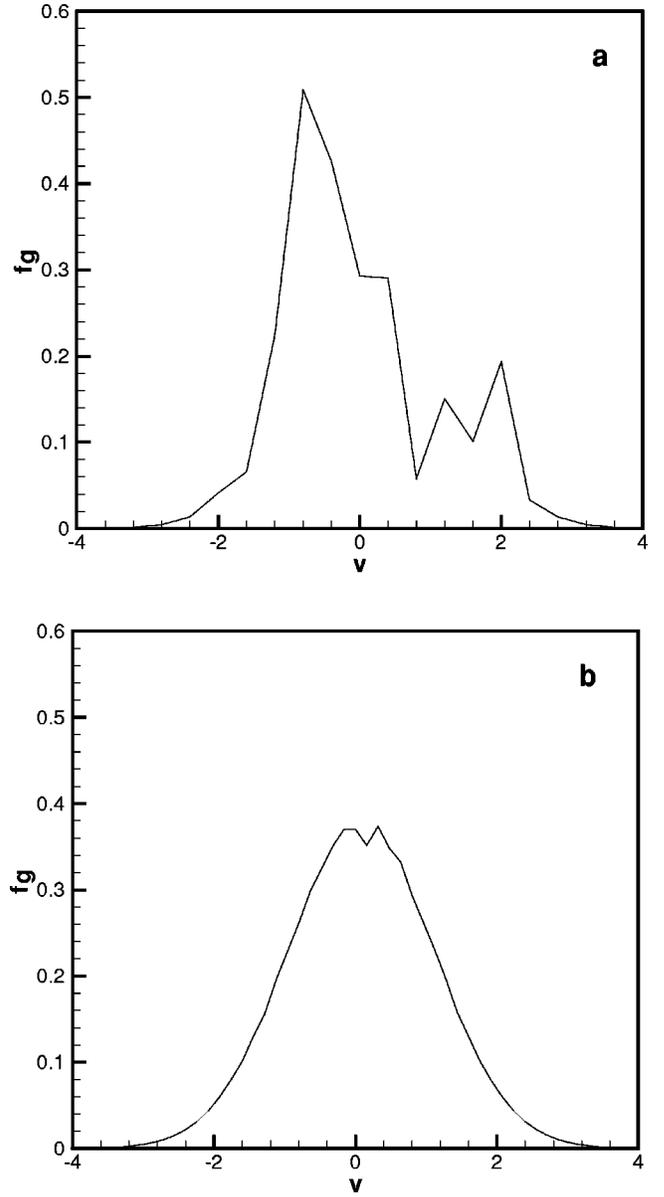


FIG. 24. (a) The  $f_g$  at one spatial grid for the system 256-D lengths across with  $\Delta v=0.4$  at normalized time 100  $/\omega_{pe}$ . (b) The  $f_g$  at one spatial grid for the system 16-D lengths across with  $\Delta v=0.16$  at normalized time 100  $/\omega_{pe}$ .

From Eq. (B5) one can prove by induction that if  $f_p$  remains unchanged with time (fixed characteristics), then after  $n$  time steps one has

$$f_p^*(n\Delta t) = f_p + (n-1) \left[ \left(\frac{\Delta x}{2}\right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial x^2} + \left(\frac{\Delta v}{2}\right)^2 \frac{1}{2} \frac{\partial^2 f_p}{\partial v^2} + O(\Delta x^2 \Delta v^2) \right]. \quad (\text{B8})$$

One of the first observations here is that for the cases  $f_p = f_p(t)$  (i.e., when characteristics are not retained), more

terms will appear on the right-hand side of Eq. (B8) which can contribute to greater errors.

Equations (B6) and (B7) then give valuable insights on characteristic divergence due to this type of numerical diffusion and can be used to determine when trajectory errors exceed allowable limits at which point a simulation will cease validity. For example from Eqs. (B6) and (B7), it is evident that  $\Gamma_v$  should be larger than  $\Gamma_x$  initially for quiet start runs. However, as  $f_p^*$  gets distorted to scales larger than  $\Delta x$  due to velocity diffusion, it can result in nonzero  $\Gamma_x$  in such a way that  $\Gamma_x$  may eventually become dominant. Second,  $\Gamma_v$  is different at different  $v$  from the outset; and it is maximum where  $\partial^2 f_p / \partial v^2$  is maximum (which takes place at  $v = \sqrt{3/2}v_{th}$ ).

Therefore, in each force calculation trajectory (velocity) errors can be determined from  $(f_p^* - f_p)\Delta t$ . For the two sets of the simulations performed for this paper we then have

$$\left(\frac{f_p^* - f_p}{f_p}\right)_{max} \Delta t = \frac{(\Gamma_v)_{max}}{f_p} \Delta t \left(v = \sqrt{\frac{3}{2}}v_{th}\right) = 0.36\% \quad (\text{B9})$$

when  $\Delta v = 0.4v_{th}$  and equal to 0.06% when  $\Delta v = 0.16$ , as the maximum percentage of trajectory (velocity) errors per time step for each case. The actual observed trajectory errors were far smaller though since these calculations corresponded to the points of maximum  $\Gamma_v$  in phase space. For example Fig. 23 displays two snapshots of the spatially averaged  $f_g$  at 100, 200  $1/\omega_{pe}$ , respectively, for the system 256-D lengths across with  $\Delta v = 0.4$ . These results indicate accumulative errors of roughly 10% in 1000 time steps at the point  $v/v_{th} = \sqrt{3/2}$  which corresponds to the ordinate grid

point with  $v = 1.84$  in the figure. However, as one examines  $f_g$  at a given spatial grid for this case, one observes that it has ceased its Maxwellian profile at that grid point [Fig. 24(a)]; i.e., the trajectory errors have destroyed the system's initial quiet start profile, while the overall system is still Maxwellian. Figure 24(b) though which corresponds to  $\Delta v = 0.16v_{th}$  still maintains its Maxwellian quiet start profile at time  $t = 100 1/\omega_{pe}$ . Examining the data more closely indicated that in the former, the initial quiet start was maintained up to 250 time steps, while in the latter it was maintained up to 1400 time steps which is 5.6 times longer (same time steps were used in both cases). This indicates close agreement with Eq. (B7) for it shows that as  $\Delta v \rightarrow 2.5\Delta v$ ,  $\Gamma_v \rightarrow 6.25\Gamma_v$ , and the duration of quiet start in principle should be reduced by roughly  $1/\Gamma_v$  which is 6.25.

These results indicate that the length of a simulation to be a function of both  $\Gamma_v$  and  $\Gamma_x$  while its time step is governed by the Courant condition discussed in Appendix A. Therefore, to improve the performance of a Vlasov code we make the following suggestions:

(1) Make the mesh nonuniform along  $v_x$ , choosing finest resolution where  $\Gamma_v$  is highest. This is possible when one implements a finite element scheme as here.

(2) Estimate analytically how long quiet starts will last by making spatial average of  $\Gamma_v$  at an arbitrary cell.

(3) Determine how long the overall system remains Maxwellian by repeating the preceding step but using  $\Gamma_x$ . Note that  $\Gamma_x$  increases the scale of the quiet start.

(4) One can attempt adding antidiffusion terms to offset the impact of  $\Gamma_v$  and  $\Gamma_x$ .

- 
- [1] T.P. Armstrong, R.C. Harding, and G. Montgomery, *Methods in Computational Physics* (Academic Press, New York, 1970), Vol. 9, p. 30.
- [2] G. Joyce, G. Knorr, and H.K. Meier, *J. Comput. Phys.* **8**, 53 (1971).
- [3] J. Nüherberg, *J. Appl. Math. Phys.* **22**, 1057 (1971).
- [4] P.H. Sakanaka, C.K. Chu, and T.C. Marshall, *Phys. Fluids* **14**, 611 (1971).
- [5] D. Nunn, *J. Comput. Phys.* **108**, 180 (1992).
- [6] J. Denavit and W.L. Kruer, *Phys. Fluids* **14**, 1782 (1971).
- [7] Iu.L. Klimontovich, *Zh. Eksper. Theor. Fiz.* **33**, 982 (1957) [*Sov. Phys. JETP* **6**, 753 (1958)].
- [8] J. Canosa, J. Gazdag, J.E. Fromm, and B.H. Armstrong, *Phys. Fluids* **15**, 2299 (1972).
- [9] D.W. Hewett, *J. Comput. Phys.* **16**, 49 (1974).
- [10] M.M. Shoucri and R.R.J. Gagné, *J. Comput. Phys.* **23**, 242 (1977).
- [11] C.Z. Cheng and G. Knorr, *J. Comput. Phys.* **22**, 330 (1976).
- [12] M.M. Shoucri and R.R.J. Gagné, *J. Comput. Phys.* **27**, 315 (1978).
- [13] A. Ghizzo *et al.*, *J. Comput. Phys.* **108**, 105 (1993).
- [14] L.E. Johnson, *J. Plasma Phys.* **23**, 433 (1980).
- [15] Y. Kurihara, *Mon. Weather Rev.* **93**, 33 (1965).
- [16] O.C. Zeinkiewicz and K. Morgan, *Finite Elements and Approximation* (Wiley, New York, 1983), Chap. 2.
- [17] G. Dahlquist and A. Björck, *Numerical Methods*, translated by N. Anderson (Prentice Hall, Englewood Cliffs, NJ, 1974), Sec. 5.4.
- [18] S.T. Zalesak, *J. Comput. Phys.* **31**, 361 (1979).
- [19] R.W. Hockney and J.W. Eastwood, *Computer Simulation Using Particles* (Institute of Physics, University of Reading, Berkshire, 1988).
- [20] J.M. Dawson, *Rev. Mod. Phys.* **2**, 421 (1983).
- [21] Kittel, *Elementary Statistical Physics* (Wiley, New York, 1958), P. 135.
- [22] F. Kazeminezhad, J.N. Leboeuf, F. Brunel, and J.M. Dawson, *J. Comput. Phys.* **104**, 398 (1993).
- [23] J.M. Dawson, *Phys. Rev.* **118**, 381 (1960).
- [24] F.C. Grant and M.R. Feix, *Phys. Fluids* **10**, 969 (1967).
- [25] J. Denavit, *J. Comput. Phys.* **9**, 75 (1972).