JOURNAL OF COMPUTATIONAL PHYSICS 110, 150–163 (1994)

# A Splitting Algorithm for Vlasov Simulation with Filamentation Filtration

A. J. KLIMAS AND W. M. FARRELL

*NASA/GSFC, Laboratory for Extraterrestrial Physics, Code 690, Greenbelt, Maryland*

Received May 10, 1991; revised June 22, 1992

A Fourier–Fourier transformed version of the splitting algorithm for simulating solutions of the Vlasov–Poisson system of equations is introduced. It is shown that with the inclusion of filamentation filtration in this transformed algorithm it is both faster and more stable than the standard splitting algorithm. It is further shown that in a scalar computer environment this new algorithm is approximately equal in speed and far less noisy than its particle-in-cell counterpart. It is conjectured that in a multiprocessor environment the filtered splitting algorithm would be faster while producing more precise results. © 1994 Academic Press, Inc.

## INTRODUCTION

The evolution of a one-dimensional electron plasma on a finite interval, neutralized by a uniform and static background of positive ions, is governed by the Vlasov–Maxwell system

$$\frac{\partial F}{\partial \tau} + v\frac{\partial F}{\partial x} - E(x,\tau)\frac{\partial F}{\partial v} = 0 \tag{1}$$

$$\frac{\partial E}{\partial x} = 1 - \int_{-\infty}^{\infty} dv\, F(x, v, \tau) \tag{2}$$

$$\frac{\partial E}{\partial \tau} = \int_{-\infty}^{\infty} dv\, vF(x, v, \tau) \tag{3}$$

in which $x \in [-1, 1]$, $v \in (-\infty, \infty)$, and $\tau \in [0, \infty)$ are the independent position, velocity, and time variables. The quantity $F(x, v, \tau)$ is the probability distribution in the two-dimensional $(x, v)$ phase space, with the probability of finding an electron at $(x, v)$ within an infinitesimal phase space volume $dx\,dv$ being given by $F\,dx\,dv$. The electric field $E(x, \tau)$ is given by Maxwell's equations, (2) and (3), in the one-dimensional electrostatic limit; $E(x, \tau)$ is governed by the distribution of charge (right side of (2)) and current (right side of (3)) in the plasma. The Vlasov equation, (1), is equivalent to

$$F(x(s), v(s), s) = F(x(0), v(0), 0) \tag{4}$$

on the characteristics generated by

$$\frac{dx}{ds} = v \quad \text{and} \quad \frac{dv}{ds} = -E(x, s). \tag{5}$$

Klimas and Cooper [1] have shown that solutions of the Vlasov–Maxwell system can be obtained by solving the Vlasov–Poisson subsystem, (1) and (2), and then mapping the Vlasov–Poisson solution to the Vlasov–Maxwell solution. In the absence of (3) the position average of the electric field $E_0$ remains undetermined by (1) and (2) and is set to zero arbitrarily. In this paper, methods for numerically simulating solutions of the Vlasov–Poisson system, (1) and (2), with the constraint $E_0 = 0$ are discussed.

The methods that have been applied to the simulation of Vlasov–Poisson solutions can be divided into two groups; those that treat the Vlasov–Poisson system directly in the $(x, v)$ phase space and those that treat the system in one of several possible transformed spaces. Early studies dealt primarily with the latter methods. The position dependence of the Vlasov–Poisson system was usually expanded in Fourier modes and the velocity dependence in either Fourier modes, [2–8] Hermite polynomials, [9–13, 5, 7], or other related transforms [14]. The Fourier–Fourier transform method will be discussed in more detail below. While these approaches could be shown to contain certain attractive features, they generally were found cumbersome due to the appearance of computation intensive convolution type summations that represent the transformation of the nonlinear term $E(x, \tau)\,\partial F/\partial v$ in (1). With one exception [8] which will be discussed shortly, more modern approaches have dealt with the Vlasov–Poisson system in the $(x, v)$ phase space. Rather than (1) and (2), however, these approaches are based on the equivalent system (2), (4), and (5). The splitting algorithm, introduced by Cheng and Knorr [15, 16], is based on a fast and accurate method for generating the characteristic solutions of (5). Denavit [4, 17] has introduced a hybrid simulation method in which particle-in-cell (PIC) methods are mixed with Vlasov simulation concepts. The splitting algorithm and the hybrid

0021-9991/94 $6.00

method have been the primary choices for use in recent simulation studies [18–25].

It is an unfortunate property of solutions of the Vlasov–Poisson system that they often contain filamentation. From (4) it can be seen that the value of $F$ cannot change along a characteristic generated by (5). Often the characteristics mix together phase space regions where $F$ exhibits significantly different values and then steep gradients are generated in $F$ that have no mechanism to relax. In simulations that are done on a grid in the phase space, the grid almost inevitably becomes too coarse as this fine graining develops. This "filamentation" problem is well known in Vlasov simulation and has been discussed often [9, 10, 7, 26, 15]. Examples of the phenomenon will be discussed below. In practice it has been found that the steep gradients develop in $F$ primarily with respect to its velocity dependence but not its position dependence. However, no general proof of this property appears available.

Klimas [8] has introduced a filtration method for solving the filamentation problem in Vlasov simulation. In this method, a smoothed $F$ is simulated in such a way that no errors are introduced into the simulated solution by the smoothing. Further, with this method the electric field obtained from the smoothed $F$ through (2) is identical to that obtained from the unsmoothed $F$ when it can be obtained. This method will be discussed further below and examples of its application will be given. It is attractive also because its application is not limited to the one-dimensional electrostatic limit under consideration here. Unfortunately, however, this smoothing method was applied to the cumbersome Fourier–Fourier transformed Vlasov–Poisson system discussed above and no simple manner of application appears possible in the untransformed system.

In this paper it is shown that the splitting algorithm has an analogue in the Fourier–Fourier transformed phase space which is inherently superior to that in the untransformed space. The transformed splitting algorithm is more efficient than the untransformed one primarily for two reasons. First, the transformed free streaming phase of the splitting algorithm amounts to a simple shift of data in memory with no other computation necessary. This replaces the spline fitting or fast Fourier transformation and inversion [15, 16] necessary in the untransformed splitting algorithm at this phase. Second, it is not necessary to solve (2) to obtain the electric field in the transformed space; the procedure for obtaining the field is much simpler. Moreover, as shown by Klimas [8] the method for solving the filamentation problem can be applied easily and effectively in the transformed space. For simulations done on grids containing the same number of nodes it has been found that the transformed splitting algorithm, including the necessary steps to remove the filamentation, is considerably faster than the untransformed algorithm. Further, as will be demonstrated below, the transformed splitting algorithm is far more stable because of its capability to control filamentation. In applications, the standard splitting algorithm would generally be applied on grids with many more nodes to delay the point in the simulation at which phase space mixing has fine grained $F$ down to the size of the grid spacing and the simulation must be stopped.

Two applications of the transformed splitting algorithm with filamentation filtration will be discussed below. The *strong Landau damping* and *two-stream instability* solutions discussed by Cheng and Knorr [15] as part of their introduction of the splitting algorithm have been reexamined. A comparison of results obtained using the standard splitting algorithm, the transformed algorithm, and a particle-in-cell algorithm will be given.

## THE SPLITTING ALGORITHM

The splitting algorithm [15, 16] deals with the system (2), (4), and (5) in the $(x, v)$ phase space. Integration of the characteristic equations (5) is split into two steps. First the position of the characteristic is updated holding the velocity constant and then the velocity is updated holding the position and the electric field constant. At each update of the characteristics the mapping (4) must be applied. Since, in general, the phase space shifts generated in this manner do not lead exactly from one grid node to another some sort of interpolation is necessary to do the mapping. Spline fits to $F$ have been used for this purpose and also fast Fourier transforms (FFTs) have been used to accomplish both the phase space shifts and the interpolation simultaneously. Cheng and Knorr [15] showed that if the electric field is updated at the beginning of the velocity shift step then this algorithm leads to an approximation to the characteristics which is second order in the time step. Further, this algorithm is particularly attractive since it does not lead to the convolution summations which slow the transformed systems approaches.

Specifically, in the notation of Cheng and Knorr [15], a step in the integration of the system (2), (4), and (5) is accomplished by the shifts

$$F^*(x, v) = F^n(x - v\,\Delta\tau/2, v) \tag{6a}$$

$$F^{**}(x, v) = F^*(x, v + E(x)\,\Delta\tau) \tag{6b}$$

$$F^{n+1}(x, v) = F^{**}(x - v\,\Delta\tau/2, v) \tag{6c}$$

with $E(x)$ computed from $F^*(x, v)$ using (2). In the applications of this algorithm that will be discussed below all shifts, as well as the solution of (2) for $E(x)$, have been accomplished using FFTs. For example, assume $F(x, v)$ is given by a Fourier expansion in position as

$$F(x, v) = \sum_m F_m(v)\, e^{im\pi x}. \tag{7}$$

Then the shift of (6a) is equivalent to the application of a phase shift to the expansion coefficients

$$F_m^*(v) = e^{-im\pi v\,\Delta\tau/2} F_m^n(v). \tag{8}$$

The shift and fitting of (6a) is accomplished through the application of a forward FFT, the introduction of a phase shift, and then the application of a backward FFT. Similarly, the step of (6b) is accomplished through the application of a forward FFT with respect to $v \in [-v_{\max}, v_{\max}]$, the introduction of a phase shift

$$e^{im\pi E(x)\,\Delta\tau/v_{\max}} \tag{9}$$

then the application of a backward FFT. The electric field is obtained from (2) by letting

$$E(x) = \sum_m E_m e^{im\pi x}. \tag{10}$$

Then, with $E_0 = 0$,

$$E_m = \frac{i}{m\pi} \int_{-v_{\max}}^{v_{\max}} dv\, F_m^*(v). \tag{11}$$

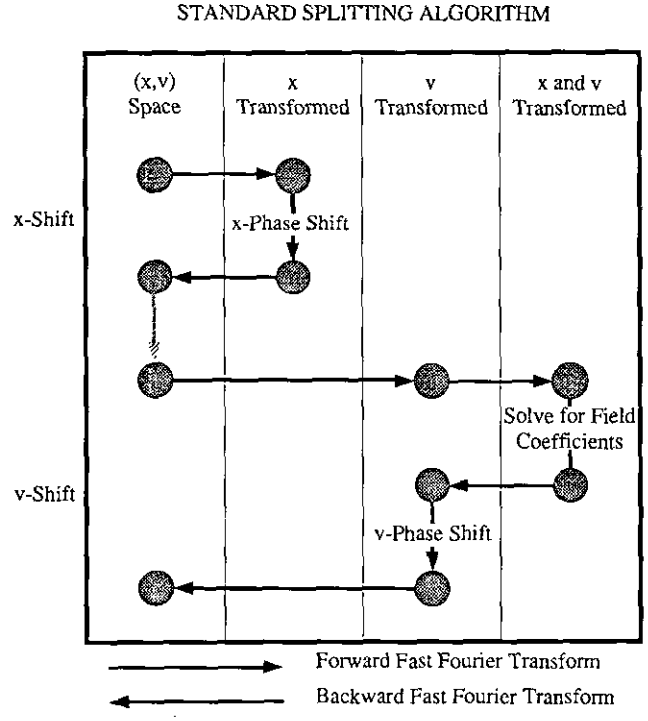


**FIG. 1.** From top to bottom, a single integration step in the splitting algorithm. Horizontal arrows represent forward or backward FFTs. Vertical columns indicate state of Fourier transformation ranging from none (left) to $x$ and $v$ transformed (right). The dashed vertical arrow is for graphical convenience and does not represent a computation component. Note, the field coefficients are computed in the middle of the velocity shift in order to avoid the velocity integration implied in (11).

The determination of the $E_m$ is done in the middle of the step (6b) after the forward FFT with respect to velocity has been applied since by then the integration over velocity has effectively already been done; it is simply related to the coefficient in the expansion representing the velocity average. The shift of step (6c) is identical to that of (6a) and, in fact, they are combined into one step. A single step forward in this algorithm is represented schematically in Fig. 1.

## THE FOURIER–FOURIER TRANSFORMED APPROACH

The Fourier–Fourier transformed approach to the Vlasov–Poisson system, introduced by Knorr [2], derives from the application of Fourier transformations to the system with respect to both position and velocity. Assuming solutions periodic in $x$ and letting

$$F(x, v, \tau) = \tfrac{1}{2} \sum_m e^{im\pi x} \int_{-\infty}^{\infty} dv\, e^{-i\pi v v} K_m(v, \tau) \tag{12}$$

and

$$E(x, \tau) = \sum_m e^{im\pi x} E_m(\tau), \tag{13}$$

then the set of functions

$$K_m(v, \tau) = \int_{-1}^{1} dx\, e^{-im\pi x} \int_{-\infty}^{\infty} dv\, e^{i\pi v v} F(x, v, \tau) \tag{14}$$

satisfies the system of first-order hyperbolic partial differential equations

$$\frac{\partial K_m}{\partial \tau} + m \frac{\partial K_m}{\partial v} = \nu \sum_n{}' \frac{K_{m-n}(0, \tau)}{m-n} K_n \tag{15}$$

in which the prime on the summation symbol signifies that the term for which $n = m$ is skipped. The relationship between the electric field expansion coefficients $E_m$ and the $K_m$ which is generated by (2)

$$E_m(\tau) = \frac{i}{m\pi} K_m(0, \tau) \tag{16}$$

has already been incorporated into this equation. Numerical simulations of solutions of (15) [6, 8] have been based on its first integral

$$K_m(v, \tau) = K_m(v - m\tau) + \sum_n{}' \int_0^{\tau} ds (v - ms) \times \frac{K_{m-n}(0, s)}{m-n} K_n(v - ms, \tau - s) \tag{17}$$

and discrete approximations to the time integral.

In relation to numerical simulation, there are two attractive features of (15) when compared to the original Vlasov–Poisson system. First, the characteristics are simple straight lines which are a priori available. Thus, with the establishment of an appropriate grid, interpolation between grid points can be entirely avoided. Second, and more significant, the problem of solving (2) for the electric field in the standard splitting scheme is replaced by simply evaluating the $K_m$ at $v=0$. The major disadvantages of (15) are the convolution summation it contains and the matrix inversion that results when discrete approximations to (17) are constructed; these must be done at each grid point for every time step forward. The summation can be done more efficiently [6] by taking advantage of its convolution form. This technique will be discussed more thoroughly below. However, this approach still remains considerably slower and more difficult than the splitting algorithm.

## FILAMENTATION AND FILTERED EQUATIONS

Cheng and Knorr [15] attempted to solve the filamentation problem by periodically filtering the probability distribution function with respect to its velocity dependence as the simulated solution was being generated. The filtering was done through a convolution

$$\bar{F}(x, v, \tau)=\int_{-\infty}^{\infty} du\, \mathscr{F}(v-u)\, F(x, u, \tau) \tag{18}$$

in which the filter $\mathscr{F}(v)$ was chosen to preserve the physically important lowest moments of the distribution. This periodic filtering, however, introduced error into the simulation. Filamentation is a real phenomenon in solutions of the Vlasov equation; it is not a spurious side effect of numerical simulation. Each time the filamentation was removed, in effect, a new solution was simulated. While this procedure allowed the simulation to continue, the relationship between the results and the initial state of the distribution function and field became more remote each time the filtering was done.

Klimas [27] solved the filamentation problem by carrying the filtering concept of Cheng and Knorr one step further. The system of equations that govern the evolution of $\bar{F}$ and $E$ was constructed from the Vlasov–Maxwell system. With the specific choice

$$\mathscr{F}(v)=\frac{\sqrt{2\pi}}{v_0}\, e^{-(v/v_0)^2/2} \tag{19}$$

in which the filter width $v_0$ is a constant parameter of choice, the filtered Vlasov–Maxwell system

$$\frac{\partial \bar{F}}{\partial \tau}+v\frac{\partial \bar{F}}{\partial x}-E(x, \tau)\frac{\partial \bar{F}}{\partial v}=-v_0^2\frac{\partial^2 \bar{F}}{\partial x\, \partial v} \tag{20}$$

$$\frac{\partial E}{\partial x}=1-\int_{-\infty}^{\infty} dv\, \bar{F}(x, v, \tau) \tag{21}$$

$$\frac{\partial E}{\partial \tau}=\int_{-\infty}^{\infty} dv\, v\bar{F}(x, v, \tau) \tag{22}$$

results. The method due to Klimas [27] is to solve this filtered system of equations in lieu of the original Vlasov–Maxwell system.

There are several features of the solutions of the filtered system (20), (21), and (22) that make this procedure for overcoming the filamentation problem attractive. First, Klimas [27] showed that an upper bound can be placed on $|\partial \bar{F}/\partial v|$ and, thus, a velocity grid spacing can be chosen before a simulated solution is generated with the understanding that it will be sufficiently fine to resolve $\bar{F}$. Generally, making the grid spacing comparable to or smaller than $v_0$ is sufficient. Since $v_0$ is a parameter of choice, the grid spacing can be chosen as desired. Second, from (18) it can be seen that $\bar{F}$ always gives a smoothed image of $F$, if it were possible to compute $F$. The smoothing is accomplished without introducing error into the evolution of either $F$ or $\bar{F}$ because the smoothed distribution is obtained from the equations that govern the evolution of the smoothed distribution. Since the filter width can be chosen, the desired resolution of the smoothed image can be chosen. As an aside, the filter (19) can be thought of as a rather idealized detector response in (18) and then the filtered Vlasov–Maxwell system becomes a theory for an observable distribution of electrons. Third, the electric field that is computed from the filtered system is identical to the one that would be obtained from the original system if it could be computed. With the choice of filter (18) the zeroth and first moments of $F$ and $\bar{F}$ are identical. Thus, the distributions of charge and current that appears on the right sides of (21) and (22) are identical to those that appear on the right sides of (2) and (3). The result is that the computed field is unchanged by the choice to compute a smoothed distribution and, further, the computed field is independent of the filter width used. Finally, Klimas [27] showed that this approach is generalizable with no change to the full three-dimensional electromagnetic Vlasov–Maxwell system.

Computing solutions of the filtered Vlasov–Maxwell system is no more difficult than computing solutions of the original Vlasov–Maxwell system in the Fourier–Fourier transformed space. The transformed filter is given by

$$\mathscr{F}(\nu)=e^{-(\nu/\nu_0)^2/2}, \tag{23}$$

the filtered Fourier–Fourier modes are given by

$$\bar{K}_m(\nu, \tau)=\mathscr{F}(\nu)\, K_m(\nu, \tau), \tag{24}$$

and their evolution is governed by the slight generalization of (15)

$$\frac{\partial \bar{K}_m}{\partial \tau} + m\frac{\partial \bar{K}_m}{\partial v} = v \sum_n{}' \frac{\bar{K}_{m-n}(0,\tau)}{m-n}\bar{K}_n - \varepsilon m v \bar{K}_m \tag{25}$$

in which

$$\varepsilon = \frac{1}{v_0^2} = (\pi v_0)^2. \tag{26}$$

Klimas [27–29] has given a number of examples in which the smoothed distribution has been simulated on this basis. However, simulating solutions of the system (20), (21), and (22) in the original $(x, v)$ phase space does not appear tractable; i.e., a simple extension of the standard splitting algorithm to a filtered equation system appears beyond reach.

## THE FOURIER–FOURIER TRANSFORMED SPLITTING ALGORITHM

Combining (6c) with (6a), the standard splitting algorithm consists of two steps, a shift of the position variable and a shift of the velocity variable in the probability distribution. These shifts are the successive solutions of the pair of equations

$$\frac{\partial F}{\partial \tau} + v\frac{\partial F}{\partial x} = 0 \tag{27a}$$

$$\frac{\partial F}{\partial \tau} - E(x)\frac{\partial F}{\partial v} = 0 \tag{27b}$$

over time intervals of length $\Delta\tau$ and with $E(x)$ computed using (2) at the beginning of the velocity shift. This pair of equations plus the instructions to find their solutions successively in this manner are an equivalent expression of the standard splitting algorithm. The filtering operation that was applied above to (1) to obtain the filtered Vlasov equation (20) can be applied to this pair of equations to obtain

$$\frac{\partial \bar{F}}{\partial \tau} + v\frac{\partial \bar{F}}{\partial x} = -v_0^2 \frac{\partial^2 \bar{F}}{\partial x\, \partial v} \tag{28a}$$

$$\frac{\partial \bar{F}}{\partial \tau} - E(x,\tau)\frac{\partial \bar{F}}{\partial v} = 0. \tag{28b}$$

This pair of equations is the basis for the filtered splitting algorithm. As above, this pair of equations is to be solved successively over time intervals of length $\Delta\tau$ and with $E(x)$ computed using (21) at the beginning of the velocity shift. In practice, this operation has been found difficult. Instead, the Fourier–Fourier transformed representation of (28a) and (28b),

$$\frac{\partial \bar{K}_m}{\partial \tau} + m\frac{\partial \bar{K}_m}{\partial v} = -\varepsilon m v \bar{K}_m \tag{29a}$$

$$\frac{\partial \bar{K}_m}{\partial \tau} = v \sum_n{}' \frac{\bar{K}_{m-n}(0,\tau)}{m-n}\bar{K}_n, \tag{29b}$$

has been utilized. Although this pair of equations reside in the Fourier–Fourier transformed space, (29a) will nevertheless be said to generate the position shift and (29b), the velocity shift. The $\bar{K}_m(0,\tau)$ in (29b) or, equivalently, the electric field modes given by

$$E_m(\tau) = \frac{i}{m\pi}\bar{K}_m(0,\tau) \tag{30}$$

are to be computed at the beginning of the velocity shift step and held constant in time through that step. In the Fourier–Fourier transformed space the filtered splitting algorithm follows from the successive solution of this pair of equations.

The position shift is given by the solution of (29a),

$$\bar{K}_m(v, \Delta\tau) = e^{-\varepsilon m(v - m\,\Delta\tau)\,\Delta\tau} e^{-(1/2)\,\varepsilon m^2(\Delta\tau)^2} \times \bar{K}_m(v - m\,\Delta\tau, 0). \tag{31}$$

Multiplication by the exponentials in this equation accomplishes the smoothing in velocity necessary to remove filamentation from $\bar{F}(x, v, \tau)$. Otherwise, assuming that the grid has been established properly, the position shift entails a shift of the $\bar{K}_m$ in computer memory from grid node $v - m\tau$ to grid node $v$. The forward and backward FFT pair associated with the position shift in the standard splitting algorithm (see Fig. 1) has been avoided.

The velocity shift is done most efficiently using the method first discussed in this context by Denavit and Kruer [6]. Using (30), a backward FFT is applied to (29b) to obtain

$$\frac{\partial \bar{K}(x, v, \tau)}{\partial \tau} = -i\pi v E(x)\,\bar{K}(x, v, \tau). \tag{32}$$

The solution

$$\bar{K}(x, v, \Delta\tau) = e^{-i\pi v E(x)\,\Delta\tau}\bar{K}(x, v, 0) \tag{33}$$

is then forward transformed back to the Fourier–Fourier transformed space. Note the phase shift in this equation is the $v$-phase shift of the standard splitting algorithm (see Fig. 1). The pair of FFTs applied here with respect to the position variable are equivalent to those of the standard algorithm. However, the pair of FFTs with respect to the

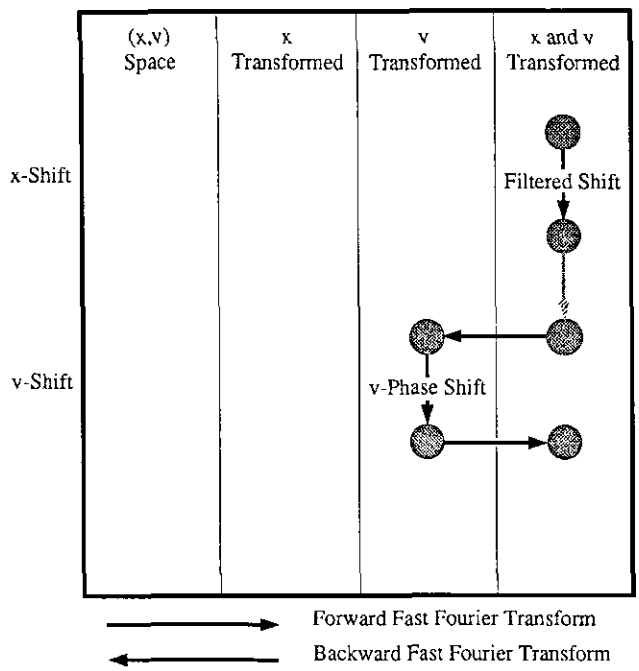


**FIG. 2.** From top to bottom, a single integration step in the filtered splitting algorithm. Horizontal arrows represent forward or backward FFTs. Vertical columns indicate state of Fourier transformation ranging from none (left) to $x$ and $v$ transformed (right). The dashed vertical arrow is for graphical convenience and does not represent a computation component.

velocity variable, in the standard algorithm, are avoided. A single integration step forward in this filtered splitting algorithm is represented schematically in Fig. 2. Comparison of this figure with Fig. 1 reveals the overall reduction in complexity in the Fourier–Fourier transformed space. Of the six FFTs necessary to carry out this step in the standard splitting algorithm, four have been eliminated. The operations that remain are approximately equivalent in both algorithms.

## RESULTS

Simulation results of two solutions of the Vlasov–Poisson system will be presented here. They are the "strong Landau damping" and the "two-stream instability" solutions discussed by Cheng and Knorr [15] in their paper in which the splitting algorithm was introduced. Results obtained using the standard splitting algorithm ("unfiltered" results), the transformed and filtered splitting algorithm ("filtered" results), and a one-dimensional electrostatic particle-in-cell algorithm ("PIC" results) [30] will be compared. The Vlasov simulations were done on a 64 (position) by 512 (velocity) element grid. Further details of the PIC algorithm are given below.

### A. *Strong Landau Damping*

The initial probability distribution for the strong Landau damping solution is given by

$$F(x, v, 0) = \sqrt{2\pi}\,(1 - 0.5\cos \pi x)\, e^{-(2\pi v)^2/2} \qquad (34)$$

For this initial state and for the results to be presented in the following the unfiltered distribution is on a grid with spacing in velocity $\Delta v = 0.01$ while for the filtered distribution $\Delta v \simeq 0.04$. The filter width $v_0 \simeq 0.05$ for the filtered distribution is comparable to the grid spacing and small compared to the thermal width $v_{th} = 1/2\pi \simeq 0.16$ of the initial distribution. Thus, the filtering has little effect and the two initial distributions are quite similar. The effects of varying $\Delta v$ in the unfiltered simulation and $v_0$ in the filtered simulation will be discussed.

This solution is called *strong* Landau damping because the initial spatial perturbation on the distribution function is large enough to produce an initial electric field through (2) or (21) that contains energy density equal to the thermal energy of the electrons. Nevertheless, as will be shown below

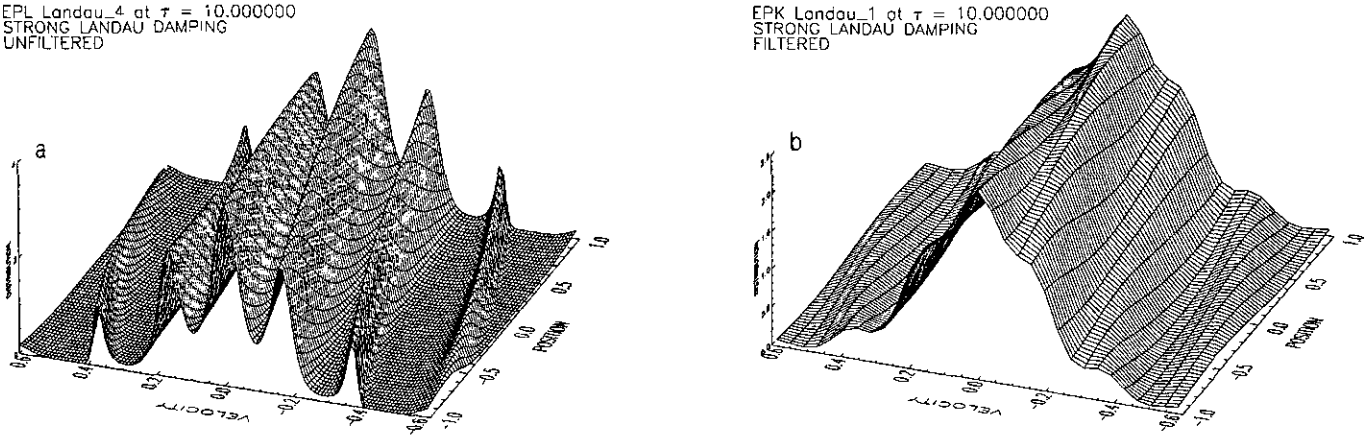


**FIG. 3.** The probability distribution for the unfiltered and filtered Vlasov simulations of the strong Landau damping solution at $\tau = 10$.

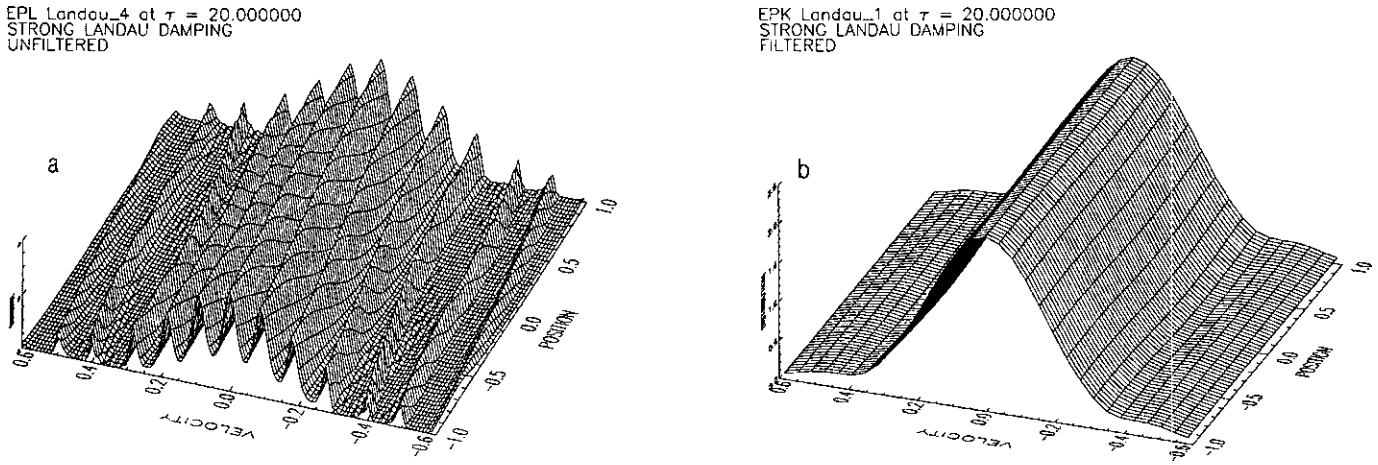


**FIG. 4.** The probability distributions for the unfiltered and filtered Vlasov simulations of the strong Landau damping solution at $\tau = 20$. Note, the viewing positions for the two distributions differ considerably. The unfiltered distribution appears considerably foreshortened because it is being viewed from almost vertically above to reduce confusion.

the field strength decays rapidly with increasing time and, although it recovers some of its initial strength later, it remains relatively weak. With the exception of the trapped characteristics which will be discussed below, the characteristics are well approximated by the free-streaming characteristics which result in the limit of zero electric field. In this case, free-streaming filamentation develops rapidly. The distribution function is sheared as it is shifted farther in position at higher positive or negative velocities than at lower ones. The continuation of this process results in the long filamentary structure (the distribution is periodic in position) shown on the left side of Fig. 3 for the unfiltered simulation. At this time the width of the filaments and the spacing between them has reduced to something approaching the filter width $v_0$ in the filtered simulation and the unfiltered and filtered distributions have begun to differ significantly.

As shown in Fig. 4, by $\tau = 20$ the filtered distribution has lost all traces of the filamentation which is so evident in the unfiltered distribution. Although difficult to see in this figure, irregular noise is emerging on the unfiltered distribution between the filaments. This noise is due to aliasing in the Fourier transformations with respect to velocity which are done during the velocity shift step in the splitting algorithm. The width of the filaments has now begun to approach the velocity grid spacing, thus leading to the failure of the Fourier transformations.

Finally, as shown in Fig. 5 at $\tau = 40$, the unfiltered distribution has become heavily filamented and the simulation will shortly be dominated by noise, due to aliasing, which is growing rapidly. The unfiltered simulation must be stopped at this point. The filtered distribution, however, remains smooth and shows the appearance of two phase space vortices on either side of the thermal core of the distribution.

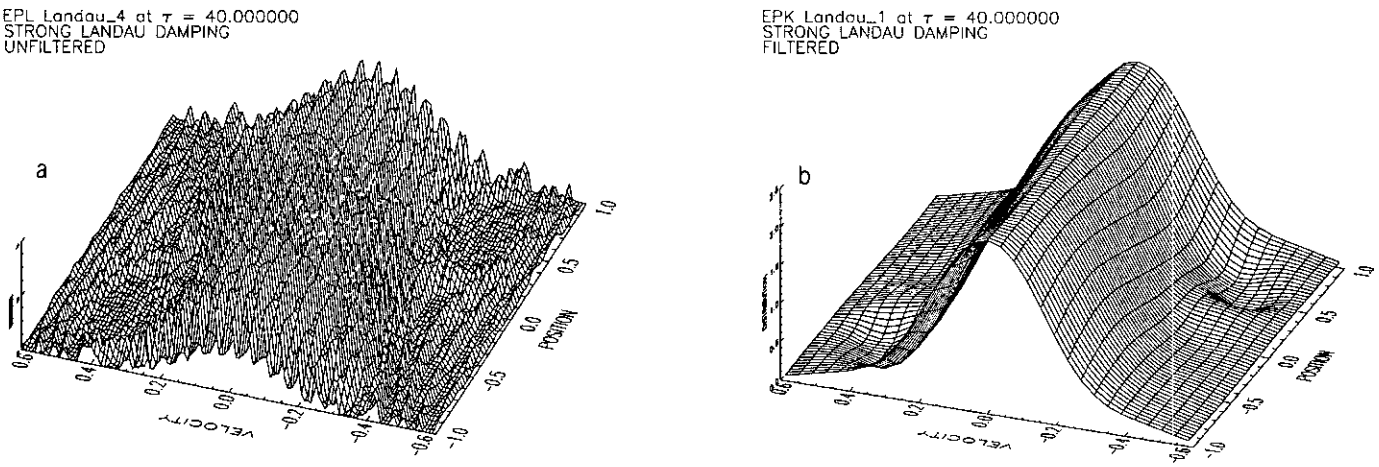


**FIG. 5.** The probability distribution for the unfiltered and filtered Vlasov simulations of the strong Landau damping solution at $\tau = 40$. Notice, the viewing positions for the two distributions differ considerably. The unfiltered distribution appears considerably foreshortened because it is being viewed from almost vertically above to reduce confusion.

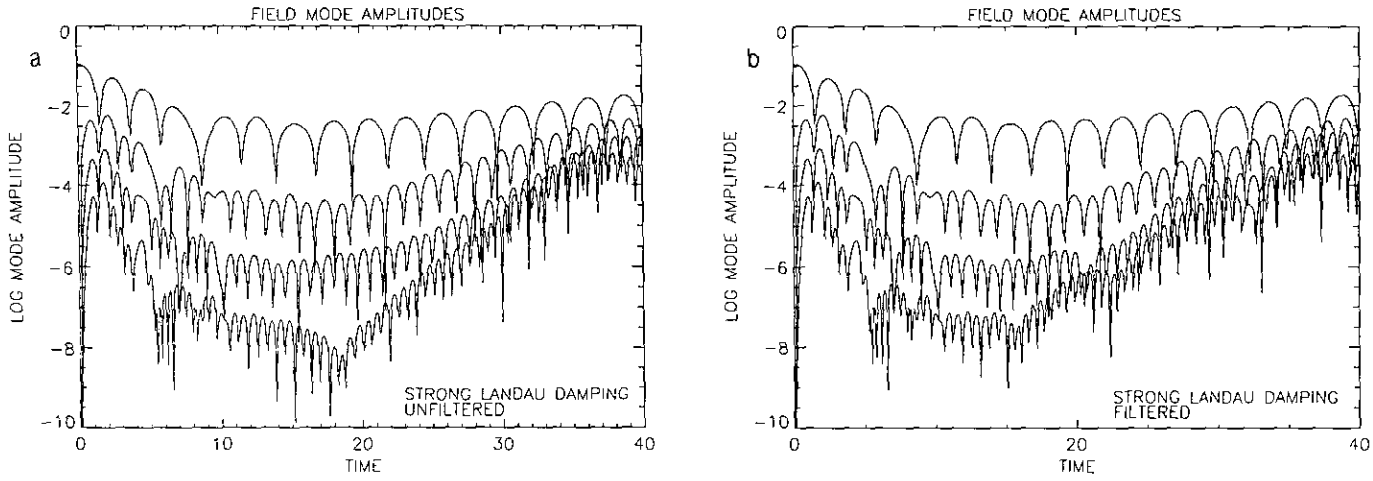


**FIG. 6.** The evolution with time of the dominant field modes for the unfiltered and filtered Vlasov simulations of the strong Landau damping solution.

Evidence for these vortices can also be seen in the unfiltered distribution; they propagate in position at the velocity at which they appear on the distribution and are due to trapping in the electric field.

The electric fields of the unfiltered and filtered simulations of this strong Landau damping solution are dominated by several of their Fourier modes at the lowest wavenumbers in the transformation with respect to position. The evolution of these modes with time is shown in Fig. 6. It can be seen that the fields computed through the unfiltered and filtered simulations, as expressed through the contributions of the dominant field modes, are essentially identical as predicted above.

However, as shown in Fig. 7, the evolution of the distributions of all field modes differs considerably for the two simulations. The onset of aliasing in the Fourier transformations of the splitting algorithm due to velocity filamentation on the distribution function leads to the spurious growth of all of the high wavenumber (short wavelength) field modes. The unfiltered simulation has been done on velocity grids with spacing two and a half times that of the results shown here to verify the spurious nature of this noise. Indeed, the noise onset time (as determined through its first appearance in figures like Fig. 7) was found to vary inversely with the grid size and the amplitude of the noise at $\tau = 40$ when the simulations were ended was found much larger in the simulation done on the coarse grid than on the fine grid. The filtered simulation shows a remarkably controlled field mode distribution with no evidence of numerical problems up to $\tau = 40$ when the simulation was ended.

The strong Landau damping solution has also been simulated using a one-dimensional electrostatic PIC algorithm [30]. To obtain comparable spatial resolution in the PIC and the Vlasov simulation results, the PIC simulations

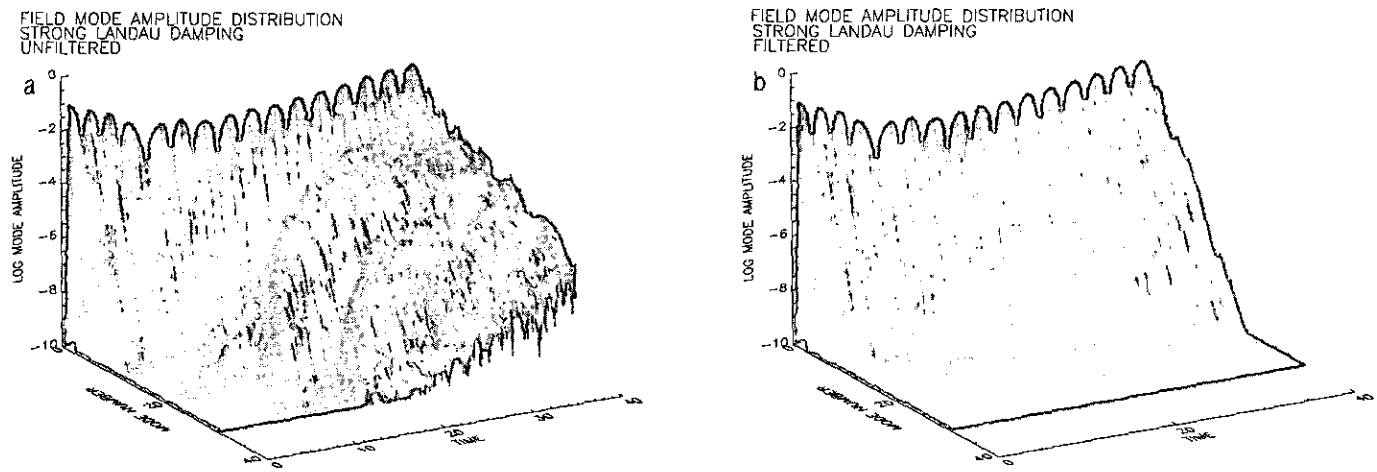


**FIG. 7.** The evolution with time of the distributions of all field mode amplitudes contained in the unfiltered and filtered Vlasov simulations of the strong Landau damping solution. Time increases to the right and wavenumber increases from back to front. The vertical scale is logarithmic.

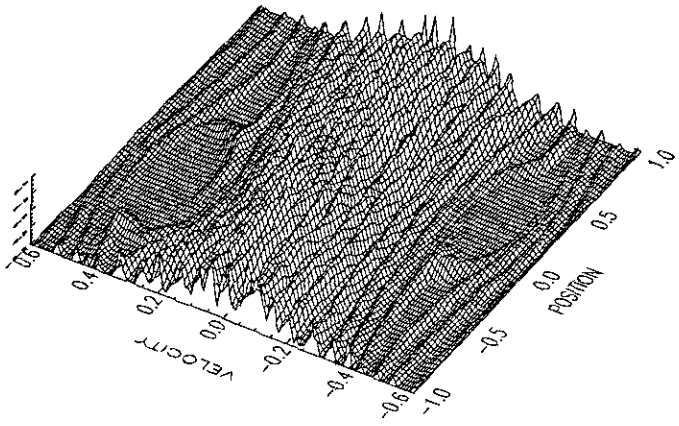


FIG. 8. The probability distribution for the PIC simulation of the strong Landau damping solution at $\tau = 40$. Note, the distribution appears considerably foreshortened because it is being viewed from almost vertically above to reduce confusion.

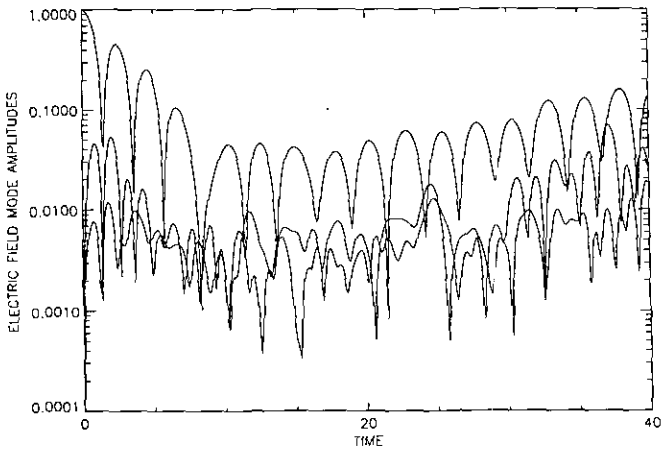


FIG. 10. The evolution with time of the dominant field modes for the PIC simulation of the strong Landau damping solution.

were done on a position grid with 64 elements. The electric field was updated at each integration step by simple particle counting to effect the integral over velocity in (2) and trapezoidal integration to undo the derivative with respect to $x$ in (2). Thus, this is a simple but efficient algorithm. The results that will be presented here were obtained using 64,000 particles.

Figure 8 shows the probability distribution obtained from the PIC simulation of the strong Landau damping solution at $\tau = 40$. To represent the result as the surface plot shown the particles were further binned in velocity on a grid with spacing $\Delta v = 0.008$, a value slightly smaller than that used to obtain the unfiltered Vlasov simulation results displayed above. Comparison of this figure with Fig. 5 shows that the PIC simulation has resolved the distribution function approximately as well as the unfiltered Vlasov simulation has at the end of the respective simulations.

The noise level in the PIC simulation is quite high. Comparison of Fig. 9 with Fig. 7 shows that almost all of the field modes are dominated by noise due to the discrete distribution of particles in this simulation method. Figure 10 shows the evolution with time of the three dominant field modes in this simulation. Comparison of this figure with Fig. 6 shows that the evolution of the strongest field mode is very similar in all three simulations. The next two strongest field modes in the PIC simulation are only qualitatively similar to the Vlasov modes and higher wavenumber modes are dominated by noise. Thus, while many of the features of the distribution function are resolved they contain sufficient noise to mask any but the dominant features in the field.

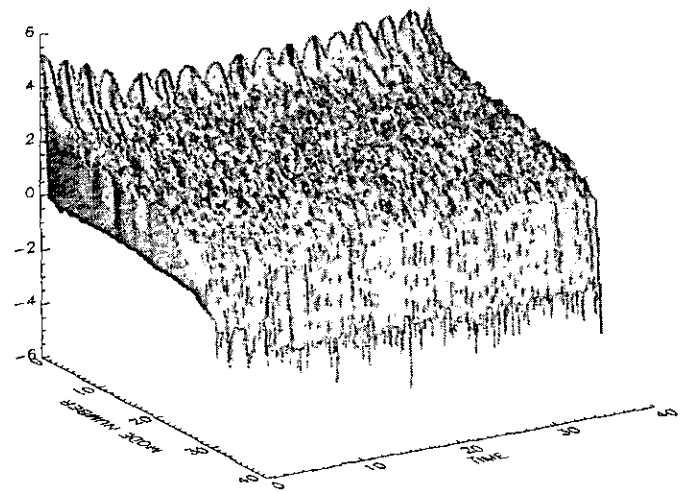


FIG. 9. The evolution with time of the distribution of all field mode amplitudes contained in the PIC simulation of the strong Landau damping solution. Time increases to the right and wavenumber increases from back to front. The vertical scale is logarithmic.

### B. *Two-Stream Instability*

The initial probability distribution for the two-stream instability solution is given by

$$F(x, v, 0) = \sqrt{2\pi}\,(1 - 0.05\cos \pi x) \times (2\pi v)^2\, e^{-(2\pi v)^2/2}. \tag{35}$$

The evolution of this solution is fundamentally different from that of the strong Landau damping solution discussed above. The strong Landau damping solution is notable for its rapid decay in field strength with the field energy transferred into particle energy in the process. Most of the characteristics rapidly approach the free-streaming characteristics. Further, the initial spatial perturbation on the distribution function is large. The result is free-streaming filamentation of large amplitude. The two-stream instability starts with a weak spatial perturbation on the distribution

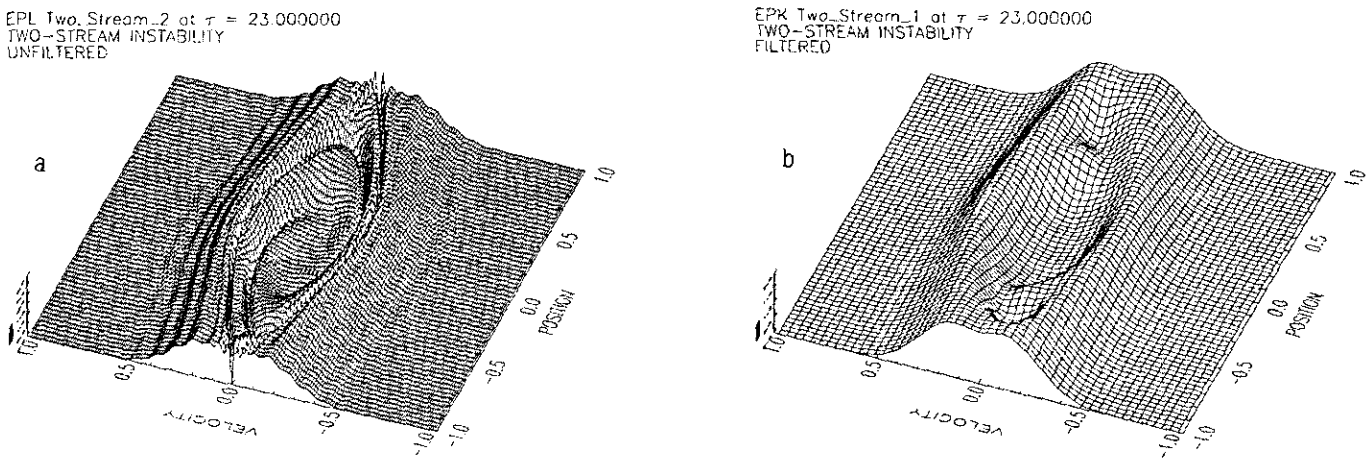


**FIG. 11.** The probability distribution for the unfiltered and filtered Vlasov simulations of the two-stream instability solution at $\tau = 23$.

function and leads to growth of the field strength and consequent trapping of a significant portion of the total particle population; few characteristics can be characterized as approximately free-streaming. In this case free-streaming filamentation is negligible but filamentation of a different type forms nevertheless.

The simulated unfiltered and filtered distribution functions at $\tau = 23$ are displayed in Fig. 11. At this time the field strength has grown sufficiently to trap many of the particles in the opposing beams of the initial state. The result is the swirling phase space vortex shown in the figure. Many of the details of this vortex are smoothed away in the filtered result with the filter width that was used. Higher resolution could be attained through the use of a narrower filter and at the expense of a larger computational effort. On the other hand, the detail is beginning to overwhelm the unfiltered computation at $\tau = 23$. Again, numerical noise is beginning to emerge due to filamentation which, in this case, is present within the vortex. The depression in the initial distribution in the vicinity of $v = 0$ has been wrapped around the walls of the vortex by the swirling characteristic motion thereby leading to structure near or at the velocity grid spacing and consequently to aliasing in the Fourier transformations of the unfiltered splitting algorithm.

At the end of the simulations, at $\tau = 40$ (see Fig. 12), the filtered distribution shows a deep depression at the position of the vortex but essentially none of the details of its interior. The unfiltered distribution has become significantly contaminated by noise of this time.

Figure 13 shows the evolution with time of the unfiltered and filtered distributions of field mode amplitudes for this solution. The unfiltered simulation develops noise in the high wavenumber modes as the two-stream instability comes to saturation. The behavior of this noise both with time and with velocity grid spacing is quite different from that of the strong Landau damping solution discussed

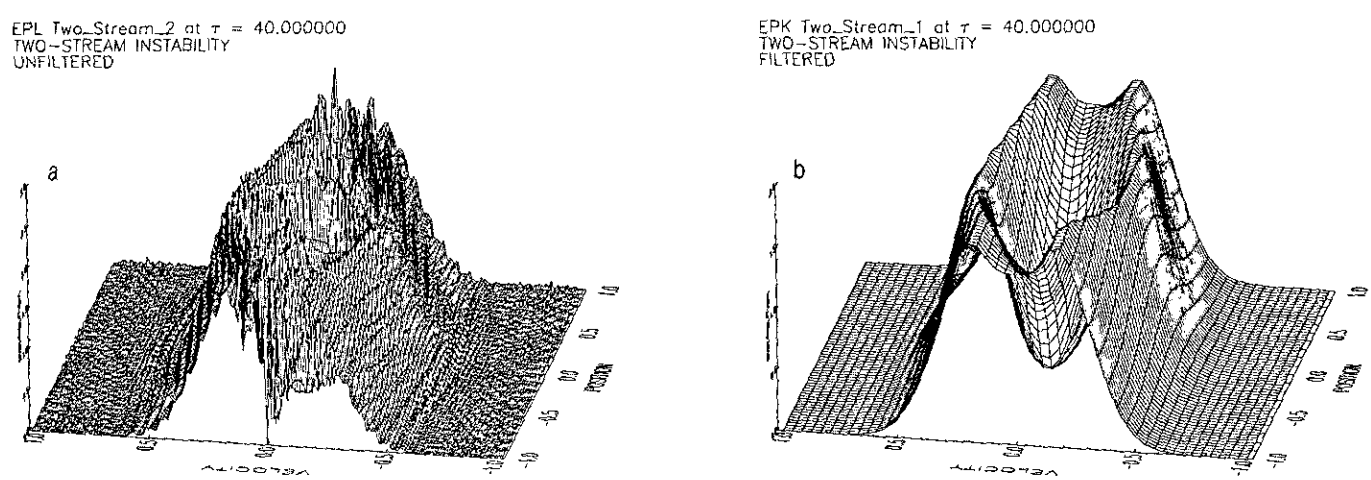


**FIG. 12.** The probability distribution for the unfiltered and filtered Vlasov simulations of the two-stream instability solution at $\tau = 40$.

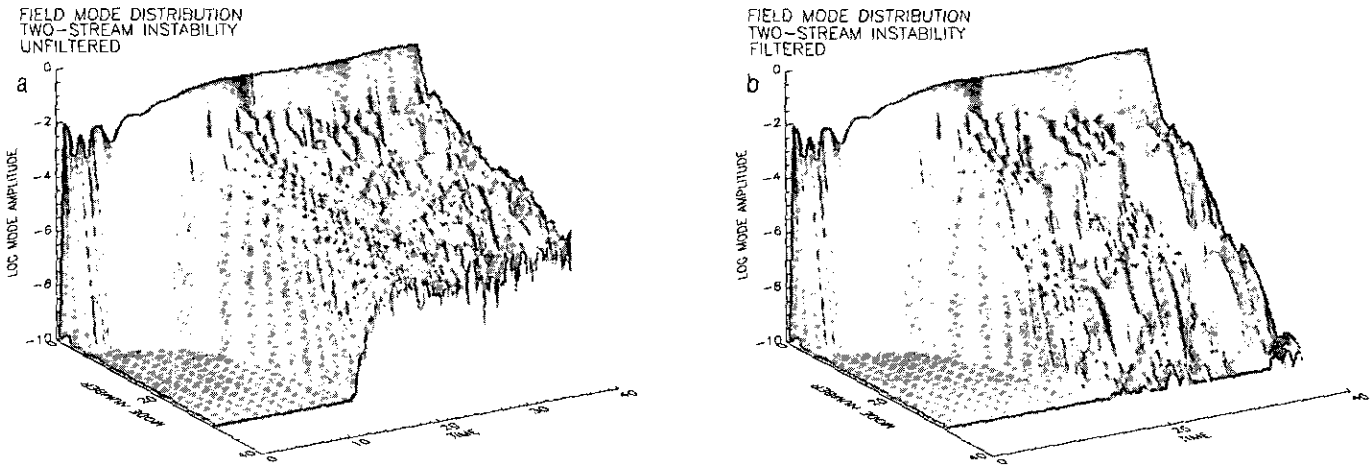


**FIG. 13.** The evolution with time of the distributions of all field mode amplitudes contained in the unfiltered and filtered Vlasov simulations of the two-stream instability solution. Time increases to the right and wavenumber increases from back to front. The vertical scale is logarithmic.

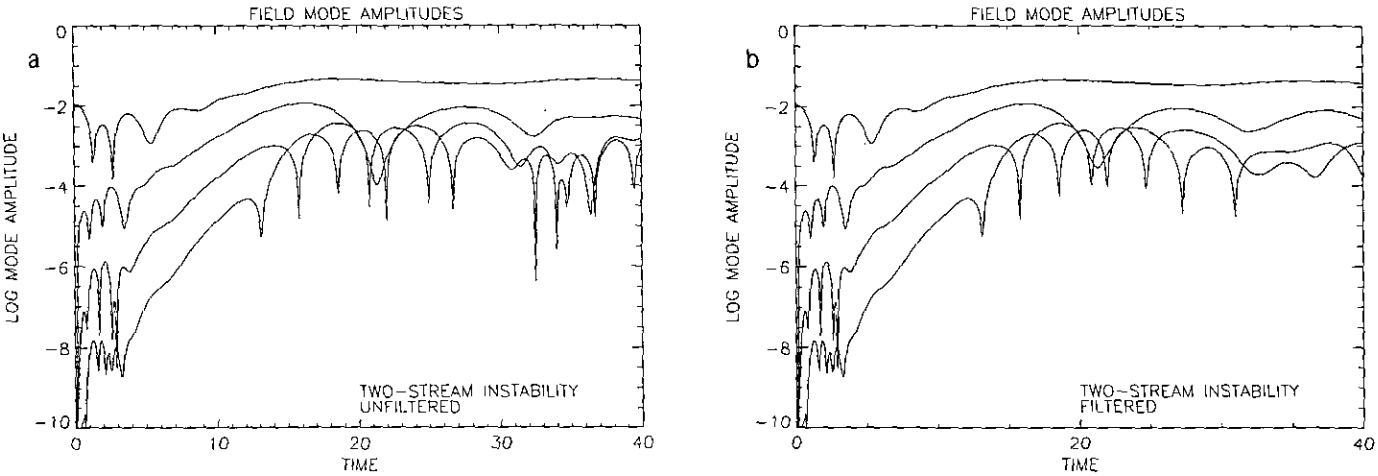


**FIG. 14.** The evolution with time of the dominant field modes for the unfiltered and filtered Vlasov simulations of the two-stream instability solution.

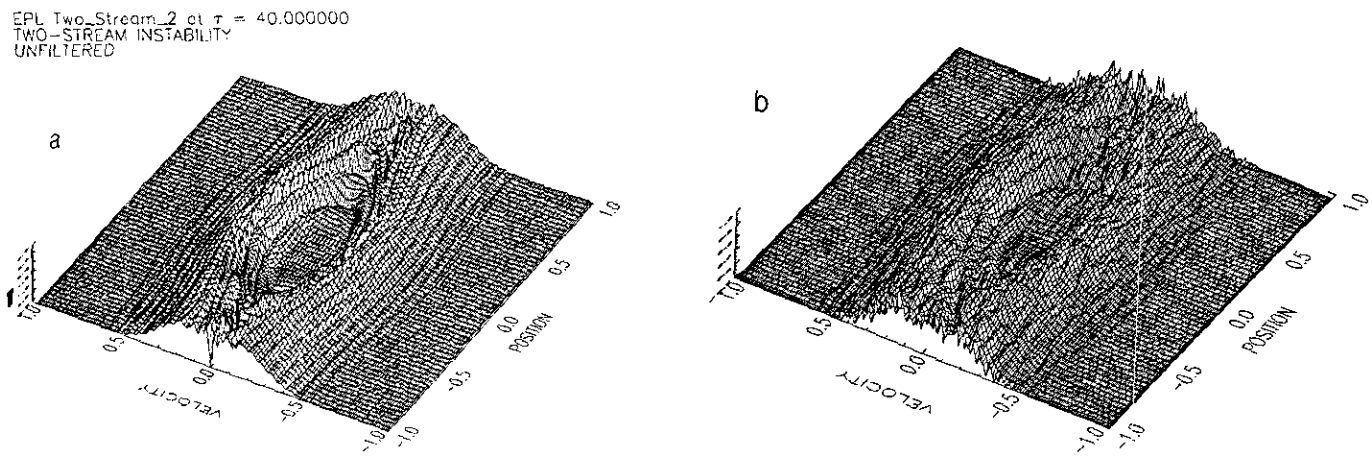


**FIG. 15.** The probability distribution for the unfiltered Vlasov and PIC simulations of the two-stream instability solution at $\tau = 40$.

above. In this case, as time increases the noise reaches a maximum level at about the same time that the instability saturates and then shows no tendency for further growth. In the unfiltered Landau damping simulation the noise level was found to always increase with time until it became the dominant feature of the simulation. Halving the velocity grid spacing used to obtain the two-stream instability results shown in Fig. 13 leads to no significant change in the field mode amplitude distribution for small and large wavenumbers. Increasing the grid spacing leads to a reduction of the noise at the largest wavenumbers and to an earlier onset of noise for moderate wavenumbers in a manner similar to the earlier onset times found in the Landau damping simulation when the velocity grid spacing was increased.

The evolution with time of the dominant field modes at the lowest wavenumbers for both the unfiltered and the filtered simulations are shown in Fig. 14. Again, the fields computed from the unfiltered and the filtered simulations are essentially identical. The differences that do appear lie in the lower amplitude, higher wavenumber, mode amplitudes.

This solution has also been simulated using the PIC algorithm discussed above. A second view of the probability distribution at $\tau = 40$ obtained using the unfiltered splitting algorithm is compared to the distribution obtained at that time using the PIC algorithm in Fig. 15. Again, the results are remarkably similar except for the presence of excess noise on the PIC distribution.

The presence of excess noise in the PIC simulation results is further confirmed through a comparison of Fig. 16 with Fig. 13. As in the strong Landau damping simulations discussed above, the evolution of the dominant field mode in the PIC simulation is in good agreement and the next strongest mode is in qualitative agreement with the Vlasov simulation. The other less significant modes in the PIC simulation are dominated by noise.

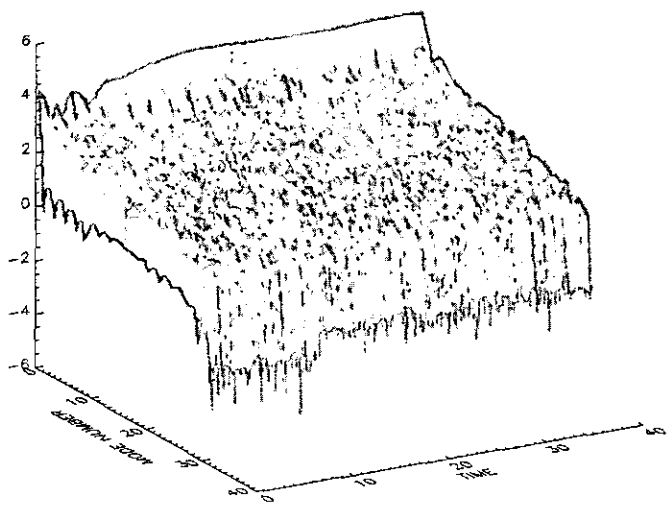


**FIG. 16.** The evolution with time of the distribution of all field mode amplitudes contained in the PIC simulation of the two-stream instability solution. Time increases to the right and wavenumber increases from back to front. The vertical scale is logarithmic.

## DISCUSSION OF RESULTS

There are two possible sources for aliasing in the Fourier transformations with respect to velocity used in the standard splitting algorithm. The first of these, filamentation, has been a central issue of the discussion above. In this case structures with increasingly finer scales and larger derivatives in the velocity variable form on the distribution function as it evolves in accordance with the Vlasov–Poisson system (1) and (2). Ultimately the scale sizes reduce to the velocity grid spacing and the Fourier transformations become aliased. A second source of noise that can be thought of as aliasing is introduced by the presence of the boundaries of the finite velocity grid. The Fourier transformations imply the distribution function is periodic in velocity but, regardless of its initial form, it will generally evolve to a function that is nonperiodic. Consequently, Gibbs oscillations form at the boundary of the grid and become the source of spurious noise which can propagate into the simulated distribution. The usual method for controlling this edge noise is to either fill the grid with zeros near its velocity boundaries or, as is the case in the results discussed here, to ensure that the boundary of the grid is sufficiently far from the thermal core of the distribution function under consideration so that the Gaussian shape of the distribution ensures extremely low amplitude Gibbs oscillations.

The noise that contaminates the unfiltered simulation of the strong Landau Damping solution is almost certainly due to filamentation. The formation of free-streaming filamentation as time progresses is evident in the simulated distribution and the inverse dependence on grid size of the noise onset time in the field mode amplitude distribution strongly supports that interpretation. The source of the noise that contaminates the unfiltered simulation of the two-stream instability solution is not as clear. For that case, the noise distribution and onset time do not clearly support one or the other of the aliasing mechanisms. To study this issue further, another unfiltered simulation of the two-stream instability solution has been carried out using the same velocity grid size and a smaller velocity grid spacing $\Delta v = 0.003$ so that the distribution function is much larger at the velocity boundaries of the grid than in all of the simulation results discussed above. This simulation is an attempt to maximize the contamination due to the Gibbs oscillations at the grid boundaries.

Figure 17 shows the simulated distribution function at $\tau = 23$ using a logarithmic scale on the vertical axis. The undulations on the flanks of the distribution are due to the

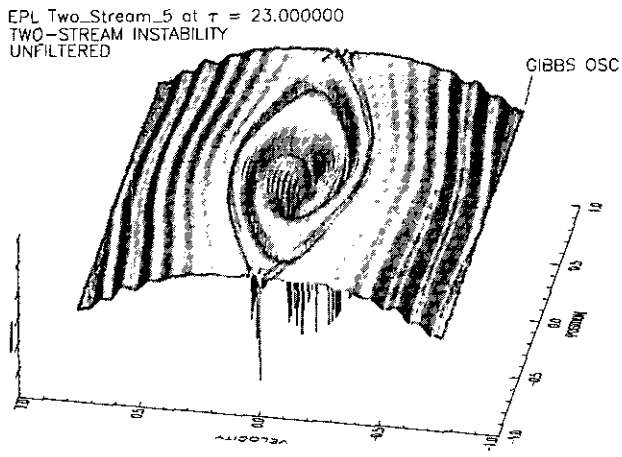


**FIG. 17.** The probability distribution for the unfiltered Vlasov simulation of the two-stream instability solution at $\tau = 23$. The vertical scale is logarithmic. The velocity grid spacing $\Delta v = 0.003$ is intentionally too small for the grid size.

formation of free-streaming filamentation on the untrapped characteristics. This filamentation, however, is well resolved by the fine velocity grid and is not a source of noise. Much earlier in the simulation Gibbs oscillations were observed to form at the velocity boundaries of the grid. By $\tau = 23$ the Gibbs oscillations have been converted into disordered noise in the thin fuzzy appearing regions (labeled "GIBBS OSC") near the boundaries of the simulated distribution in Fig. 17. This noise has not propagated into the interior of the grid. However, the phase space vortex that is clearly evident in this figure is contaminated with noise. The appearance of this noise is clearer in Fig. 18 which shows the same result as Fig. 17, but using a linear scale on the vertical axis. The noise is concentrated in the interior of the vortex where there are large derivatives due to the wrapping of the initial distribution that has occurred. It appears that the source of this noise is the trapping filamentation that forms

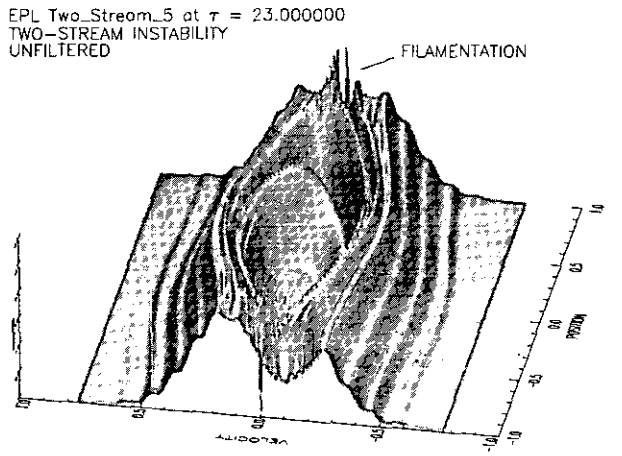


**FIG. 18.** Same as Fig. 17 except that the vertical scale is linear.

in the interior of the vortex. Since this simulation maximizes the contribution of the Gibbs oscillations to noise on the simulated distribution, it must be concluded that the dominant source of noise in the unfiltered two-stream instability results discussed above is filamentation.

The run times for the simulation algorithms discussed here have been investigated. In every case the unfiltered and filtered splitting algorithms were compared on identically sized grids. If $N_x$ is the number of grid elements in position and $N_v$ the number in velocity, then for $N_x = 64$ and $N_v = 512$ it was found that the filtered splitting algorithm (in Fourier–Fourier transformed space) is 4.4 times as fast as the standard unfiltered splitting algorithm. For $N_x = 128$ and $N_v = 512$ the ratio is 4.2 and for $N_x = 64$ and $N_v = 1024$ the ratio is 3.3. The comparison of run times between the Vlasov simulations and the PIC simulations depends considerably on a subjective choice of grid sizes and particle numbers. With $N_x = 64$ and $N_v = 512$ and with 64,000 for the number of particles, it was found that the PIC algorithm resolves details on the distribution function approximately as well as the unfiltered splitting algorithm but, of course, the PIC algorithm is not subject to noise due to filamentation. On the other hand, with that many particles only the dominant features in the field are resolved by the PIC algorithm while the splitting algorithms give much more information concerning the field evolution. If, on balance, this is a reasonable comparison, then the PIC algorithm and the filtered splitting algorithm were found to run at approximately the same speed; the PIC algorithm is 1.3 times faster. Note, however, that this comparison was done on a scalar computer. While PIC algorithms are not amenable to vectorization it is clear that the filtered splitting algorithm is largely vectorizable. In a multiprocessor environment the filtered splitting algorithm would certainly be the fastest of the three algorithms.

## SUMMARY AND CONCLUSIONS

A Fourier–Fourier transformed version of the splitting algorithm for simulating solutions of the Vlasov–Poisson system of equations has been introduced. It has been shown that the splitting algorithm is more efficient in the transformed space. It has further been shown that filamentation filtration can be included in the transformed splitting algorithm with little additional computational effort.

Two solutions of the Vlasov–Poisson system have been simulated using both the standard and the transformed splitting algorithms. For these two solutions and for the short simulations discussed it has been shown that the transformed splitting algorithm with filamentation filtration is more stable and considerably faster than the standard splitting algorithm. Using the standard splitting algorithm, significant levels of noise contaminated the distribution function and the field at the ends of the simulations because

of the filamentation on the distribution function. In contrast, the filtered simulations remained noise-free and stable.

Comparisons of simulations using the transformed splitting algorithm with much longer simulations that have been done with versions of the standard splitting algorithm [18, 21–23] are anticipated.

A PIC algorithm has also been used to simulate the Vlasov–Poisson solutions. It has been shown that in a scalar environment the PIC and the filtered splitting algorithms are approximately equal in speed. It is conjectured that in a multiprocessor environment the transformed splitting algorithm would be faster while, simultaneously, producing more details of the simulated solution.

## ACKNOWLEDGMENT

This research has been supported in part by NASA Grant 432-36-00-54.

## REFERENCES

1. A. J. Klimas and J. Cooper, *Phys. Fluids* **26**, 478 (1983).
2. G. Knorr, *Z. Naturforsch A* **18**, 1304 (1963).
3. J. Denavit, B. W. Doyle, and R. H. Hirsch, *Phys. Fluids* **11**, 2241 (1968).
4. J. Denavit, in *Proceedings, Fourth Conference on Numerical Simulation of Plasma*, edited by J. P. Boris and R. H. Shanny (Naval Research Laboratory, Washington, DC, 1970), p. 305.
5. T. P. Armstrong, R. C. Harding, G. Knorr, and D. Montgomery, "Solution of Vlasov's Equation by Transform Methods," in *Methods in Computational Pysics*, Vol. 9 (Academic Press, New York, 1970), p. 29.
6. J. Denavit and W. L. Kruer, *Phys. Fluids* **14**, 1782 (1971).
7. G. Joyce, G. Knorr, and H. K. Meier, *J. Comput. Phys.* **8**, 53 (1971b).
8. A. J. Klimas, *J. Comput. Phys.* **50**, 270 (1983).
9. F. C. Grant and M. R. Feix, *Phys. Fluids* **10**, 696 (1967a).
10. F. C. Grant and M. R. Feix, *Phys. Fluids* **10**, 1356 (1967b).
11. T. P. Armstrong, *Phys. Fluids* **10**, 1269 (1967).
12. T. P. Armstrong and D. Montgomery, *J. Plasma Phys.* **1**, 425 (1967).
13. T. P. Armstrong and D. Montgomery, *Phys. Fluids* **12**, 2094 (1969).
14. G. Joyce, G. Knorr, and T. Burns, *Phys. Fluids* **14**, 797 (1971).
15. C. Z. Cheng and G. Knorr, *J. Comput. Phys.* **22**, 330 (1976).
16. R. R. J. Gagné and M. M. Shoucri, *J. Comput. Phys.* **24**, 445 (1977).
17. J. Denavit, *J. Comput. Phys.* **9**, 75 (1972).
18. M. M. Shoucri, *Phys. Fluids* **22**, 2038 (1979).
19. J. Denavit, *Phys. Fluids* **28**, 2773 (1985).
20. A. Simon, S. Radin, and R. W. Short, *Phys. Fluids* **31**, 3649 (1988).
21. A. Ghizzo, B. Izrar, P. Bertrand, E. Fijalkow, M. R. Feix, and M. Shoucri, *Phys. Fluids* **31**, 72 (1988).
22. P. Bertrand, A. Ghizzo, M. R. Feix, E. Fijalkow, P. Mineau, N. D. Suh, and M. Shoucri, "Computer Simulation of Phase Space Holes Dynamics," in *Nonlinear Phenomena in Vlasov Plasmas*, edited by F. Doveil, Les Editions de Physique (BP 112, 91944 Les Ulis Cedex, France, 1989), p. 109.
23. A. Ghizzo, P. Bertrand, M. M. Shoucri, T. W. Johnston, E. Fijalkow, and M. R. Feix, *J. Comput. Physics* **90**, 431 (1990).
24. L. Demeio, *J. Comput. Phys.* **99**, 203 (1992).
25. L. Demeio and J. P. Holloway, *J. Plasma Phys.*, in press.
26. G. Knorr, *J. Comput. Phys.* **13**, 165 (1973).
27. A. J. Klimas, *J. Comput. Phys.* **68**, 202 (1987).
28. A. J. Klimas, *J. Geophys. Res.* **95**, 14, 905 (1990).
29. A. J. Klimas and R. J. Fitzenreiter, *J. Geophys. Res.* **93**, 9628 (1988).
30. J. Denavit and W. L. Kruer, *Comments Plasma Phys. Cont. Fusion* **6**, 35 (1980).

JOURNAL OF COMPUTATIONAL PHYSICS 110, 164–170 (1994)

# Methods for Fast Computation of Integral Transforms

SHAY GUERON

*Center For Applied Mathematics, 504 ETC Building, Cornell University, Ithaca, New York 14853*

Received August 1, 1990; revised November 19, 1992

This paper is concerned with two aspects of the numerical calculation of integral transforms. The first is finding a necessary and sufficient condition that enables converting an integral transform into a correlation (convolution) form. The condition and the transformation that implements it are generalizations of the Gardner transformation and derived in the paper. This technique can be applied to a wide class of integral transforms and is shown to reduce the computational complexity and storage requirements of the resulting algorithm. The second issue addressed in the paper is the accuracy of the calculation of the correlation integral, obtained by the above transformation, for a given number of samples. It is shown how the standard FFT method can be applied in combination with various numerical integration rules. This proves to be an important factor in expediting the computations, reducing the storage requirements, and improving the accuracy.


## 1. INTRODUCTION

The numerical approximation of integral transforms is important in many scientific and engineering problems and can present a formidable task in terms of computer time and storage requirements. The need for fast algorithms becomes especially crucial when the kernels of the transforms (or the transformed functions) are oscillatory.

In this paper we treat two problems related to the numerical calculation of integral transforms. The first is finding a necessary and sufficient condition that enables the conversion of a given integral transform into a correlation (convolution) integral. The condition and the corresponding change of variables are derived in Section 2 and generalize the well-known Gardner transformation (Gardner *et al.* [8]). We show that many of the commonly used integral transforms can be converted into a correlation form by means of a change of variables.

Correlation integrals can be calculated with significantly reduced computational effort and memory requirements. This reduction of effort can be achieved by applying the FFT algorithm to the discretized problem. Traditionally, the discrete evaluation of the transform is performed by means of the rectangular quadrature rule. In Section 2 we introduce a modification of this standard approach. At the cost of some extra multiplications, finer quadrature rules can be applied, thereby reducing the number of discretization points needed to obtain a desired accuracy. This modification is fast, easy to implement, reduces storage requirements and improves the accuracy.

In Sections 3 and 4 we give two examples that present typical difficulties involved with the change of variables and demonstrate ways to overcome them. With our approach, one can apply the same integration rules that might have been used for the original problem. Consequently, the advantage of the transformation depends only on the inherent properties of the problem (the integrands). The first example shows how our transformation can change the properties of the integrand and make it singular and that the flexibility to use different integration rules is crucial. In the second example we use the new technique to improve the fast Hankel transform (Siegman [17]).

## 2. CHANGING AN INTEGRAL TRANSFORM INTO A CORRELATION FORM

Consider the integral transform

$$F(x) = \int_a^b K(x, y)\, g(y)\, dy, \qquad x \in [c, d], \qquad (2.1)$$

where the kernel, $K(x, y)$, and the transformed function, $g(y)$, may be real or complex. Computing $F(x)$ can be a time-consuming task requiring a large storage space. In this section we investigate the cases where Eq. (2.1) can be converted into the correlation integral

$$F_1(u) = \int_{a_1}^{b_1} H(u + v)\, \tilde{g}(v)\, dv, \qquad u \in [c_1, d_1]. \qquad (2.2)$$

(Note that replacing $v$ with $-v$ in Eq. (2.2) results a convolution form. Thus, we do not distinguish between correlations and convolutions through our discussion.) We shall show that an integral transform of the form shown above has properties which may be exploited to reduce the



0021-9991/94 $6.00