JOURNAL OF COMPUTATIONAL PHYSICS 19, 58–76 (1975)

Canonical Momenta and Numerical Instabilities in Particle Codes*

BRENDAN B. GODFREY

Theoretical Division, Los Alamos Scientific Laboratory, University of California, Los Alamos, New Mexico 87545

Received November 4, 1974; revised April 21, 1975

A set of warm plasma dispersion relations appropriate to a large class of electromagnetic plasma simulation codes is derived. The numerical Cherenkov instability is shown by analytic and numerical analysis of these dispersion relations to be the most significant nonphysical effect involving transverse electromagnetic waves. The instability arises due to a spurious phase shift between resonant particles and light waves, caused by a basic incompatibility between the Lagrangian treatment of particle positions and the Eulerian treatment of particle velocities characteristic of most PIC–CIC algorithms. It is demonstrated that, through the use of canonical momentum, this mismatch is alleviated sufficiently to completely eliminate the Cherenkov instability. Collateral effects on simulation accuracy and on other numerical instabilities appear to be minor.

I. INTRODUCTION

The standard PIC–CIC plasma simulation scheme [1] is, in effect, an algorithm for solving numerically a system of hyperbolic partial differential equations, namely Vlasov's equation and Maxwell's equations. Thus, in common with other methods for treating such systems, particle codes are numerically unstable, if $\omega_{\max} \Delta t$ is too large. Here, $\omega_{\max}$ is the maximum normal mode frequency of the linearized system of equations, and Δt is the integration time step. Such constraints on Δt are often referred to as Courant conditions. In this sense, particle codes are subject to two Courant conditions. If Δt, measured in units of ω_p^{-1} (ω_p the plasma frequency), is greater than about 1.5, then the electrostatic [2] and electromagnetic [3] plasma modes independently become numerically unstable. Otherwise, if $\Delta t/\Delta x$ exceeds some constant, typically unity, short wavelength light waves become unstable. Here, Δx (in units of c/ω_p) is the characteristic size of a cell of the spatial grid.

There are, however, additional numerical instabilities unique to particle codes. These instabilities arise from a basic incompatibility between the Lagrangian

* Work performed under the auspices of the U.S. Energy Research and Development Administration.

treatment of particle positions and the Eulerian treatment of particle velocities and of electromagnetic fields. Spatial aliasing is a straightforward example. The simulation particles, distributed essentially continuously over the grid, can support an infinite set of wavevectors, $k_i = 2\pi l_i/L_i$, $-\infty < l_i < \infty$. The fields, defined only at mesh points, can support but a finite subset, $-N_i/2 < l_i < N_i/2$, of these wavevectors. Consequently, particle modes related by $k_i' = k_i + 2\pi\nu_i/\Delta x_i$, $-\infty < \nu_i < \infty$, are coupled via the fields. Here, Δx_i and N_i are the cell length and the number of cells in direction i, with $L_i = N_i \Delta x_i$. This effect has been studied in detail for electrostatic simulations [4, 5], where it gives rise to severe numerical problems, if the mesh cells are too large. Often the numerical instability resembles a legitimate two-stream instability. Presumably, spatial aliases also can drive spurious Weibel instabilities in electromagnetic simulations, though none have been reported.

A second manifestation of the Lagrangian–Eulerian dichotomy in particle codes is the numerical Cherenkov instability [3, 6]. The disparate treatment of particle positions and velocities leads to two distinct finite difference representations of $\omega - \mathbf{k} \cdot \mathbf{v}$ appearing in the electromagnetic dispersion relations for a simulation plasma. Because these two terms fail to cancel, a nonphysical pole is introduced, usually tending to destabilize the simulation. Aliases compound the problem. Should a light wave come into resonance with the pole, the results often are catastrophic [3].

People have, over the years, learned to cope with these numerical difficulties, usually by judicious choices of simulation parameters but occassionally by minor modifications of the differencing schemes [6–10]. Nonetheless, actually to eliminate their root causes remains a desirable goal. In this connection, the distinction made above between numerical limitations common to all differencing schemes for hyperbolic systems and those peculiar to particle codes as currently implemented is worth recalling. Although attempts to overcome the former may prove futile, it seems likely that significant progress in dealing with the latter is attainable.

One obvious apprach is to employ Eulerian differencing throughout, thereby eliminating both the Cherenkov instability and instabilities due to spatial aliases. However, the end result is not a particle code at all but rather a fluid code of the sort used in treating hydrodynamics problems. In general, such codes are considered unacceptable for plasma simulation, because of their excessive dispersiveness and consequent poor fine-scale resolution (Note, however, that several methods for reducing dispersion have recently been proposed. See, e.g. Ref. [11].) It appears also that particle codes are computationally more economical for studying plasmas.

A first step in the opposite direction involves determining velocities from canonical momenta instead of by direct integration. In this way, computation of particle velocities becomes partially Lagrangian. Interestingly, the change is just sufficient to make the two representations of $\omega - \mathbf{k} \cdot \mathbf{v}$ mentioned above equal, even including alias effects. (Alias contributions to other numerical instabilities are, unfortunately, not affected.) Thus, the numerical Cherenkov instability is totally eliminated. As an additional advantage, the form of the linear dispersion relations is simplified slightly.

To illustrate these various points, we begin in Section 2 by deriving a generic dispersion relation for multidimensional electromagnetic plasma computer simulations with no external fields. Great generality is possible primarily due to strong constraints placed on particle code algorithms by the underlying physics. That most codes use essentially identical methods of integrating the particle equations of motion also contributes to ease in obtaining generality. Periodic boundary conditions are assumed, a small but not necessarily insignificant limitation. Section 3 briefly considers each of the terms in the dispersion relation likely to cause numerical instability. Because of the severe coupling of effects in multidimensional simulations, a complete analysis is not possible. Nonetheless, an investigation limited to one-dimensional simulations yields some useful insights. The calculations of Sections 2 and 3 are reproduced in abbreviated form in Section 4 for the canonical momentum approach. (For completeness, an implementation of this algorithm is given in the Appendix.) From both the linear dispersion relations and the results of actual simulations, it is evident that the Cherenkov instability has been overcome. (REL, the code used, is a highly modified version of EMI, described in [12].) Finally, we observe in Section 5 that exact energy and momentum conservation are in general impossible in either conventional algorithms [1, 13] or the canonical momentum algorithm, because currents and forces are defined at different times.

This paper is, in some sense, intended as a sequel to Ref. [3], attempting to resolve several questions left unanswered there. In general, the notation remains the same, although the expression $\omega + \mathbf{k} \cdot \mathbf{v}$ has been replaced by $\omega - \mathbf{k} \cdot \mathbf{v}$ to conform with standard usage. All calculations are performed for nonrelativistic plasma simulations. Generalization to relativistic plasmas [14] should be straightforward, though perhaps not very rewarding, since the basic numerical effects are not expected to change significantly. Nonetheless, it may be that relativistic particle distributions emphasize different aspects of the numerical stability problem.

2. The Electromagnetic Dispersion Relation

In studying a number of apparently quite dissimiliar particle code algorithms, we have found their dispersion relations to be very much the same [3]. We shall expand upon this observation by developing a dispersion relation general enough to encompass most PIC–CIC nonrelativistic electromagnetic plasma simulation codes currently in use. The derivation is heuristic in the sense that the final result possesses a wider range of validity than any of the steps leading up to it. Nonetheless, three basic limitations appear as the calculation proceeds. First, the time integration scheme for the particle equations of motion may vary from the standard leap-frog scheme [1] by terms of order no lower than $(F\Delta t)^2$, where F represents any self-consistent field quantity and Δt is the integration time step. Second, finite difference approximations to derivatives in Maxwell's equations must be employed in a reasonably consistent fashion. Third, the spatial grid must be rectangular, uniform, and periodic. The first of these is necessary to give some definiteness to the final result. The second requires, among other things, that space and time differences commute and that certain common vector identities are satisfied. It does not, however, preclude use of high-order approximations to the Laplacian [7, 15] or of smoothing of short wavelength field fluctuations. The third assumption allows the equations to be spatially decomposed by Fourier transform. Relaxing this requirement causes no difficulty in principle, but complicates the algebra immensely.

We begin by treating Maxwell's equations, expressed in terms of the scalar and vector potentials, ϕ and $\mathbf{A}$,

$$\frac{\partial^2}{\partial t^2}\phi - \nabla^2\phi = \rho + \frac{\partial}{\partial t}\Lambda, \tag{1}$$

$$\frac{\partial^2}{\partial t^2}\mathbf{A} - \nabla^2\mathbf{A} = \mathbf{j} - \nabla\Lambda, \tag{2}$$

where Λ is the completely arbitrary gauge parameter [16],

$$\Lambda = \frac{\partial}{\partial t}\phi + \nabla\cdot\mathbf{A}. \tag{3}$$

The electric and magnetic fields, **E** and **B**, are given by

$$\mathbf{E} = -\nabla\phi - \frac{\partial}{\partial t}\mathbf{A}, \tag{4}$$

$$\mathbf{B} = \nabla\times\mathbf{A}. \tag{5}$$

Units are chosen such that both the speed of light and the plasma frequency are unity. If, now, these differential equations are replaced by corresponding finite difference equations and the difference equations Fourier transformed, we obtain expressions of the form

$$[\omega^2 - \mathbf{k}^2]\,\phi = -\rho - i[\omega]\Lambda, \tag{6}$$

$$[\omega^2 - \mathbf{k}^2]\,\mathbf{A} = -\mathbf{j} - i[\mathbf{k}]\Lambda, \tag{7}$$

$$\Lambda = -i([\omega]\,\phi - [\mathbf{k}]\cdot\mathbf{A}), \tag{8}$$

$$\mathbf{E} = -i([\mathbf{k}]\,\phi - [\omega]\,\mathbf{A}), \tag{9}$$

$$\mathbf{B} = i[\mathbf{k}]\times\mathbf{A}. \tag{10}$$

Brackets around quantities designate their finite difference representations. Note that some quantities may have multiple representations in the same differencing scheme, although we do not explicitly take account of this possibility.

For internal conistency of Eqs. (1)–(3), the charge and current densities, ρ and $\mathbf{j}$, must satisfy the continuity equation

$$\frac{\partial}{\partial t} \rho + \nabla \cdot \mathbf{j} = 0. \tag{11}$$

When, as in most codes, this equation does not follow automatically from the mesh interpolation algorithm (for exceptions, see, e.g., [17]), one must appropriately adjust the longitudinal (i.e., curl-free) component of the current. Failure to do so, in addition to impairing accuracy, introduces a spurious low-frequency longitudinal mode, which may, under certain circumstances, lead to instability. It is equivalent to correct the current explicitly, to adjust the vector potential according to Eq. (3) [18], or to demand simply that $\nabla \cdot \mathbf{E} = \rho$ [19]. In any case, we find after a short calculation that

$$\mathbf{j} = \tilde{\mathbf{j}} + [\mathbf{k}]([\omega]\,\rho - [\mathbf{k}] \cdot \tilde{\mathbf{j}})/[\mathbf{k}]^2, \tag{12}$$

where $\tilde{\mathbf{j}}$ is the original, uncorrected current. The new current $\mathbf{j}$, as defined in Eq. (12), identically satisfies

$$[\omega]\,\rho - [\mathbf{k}] \cdot \mathbf{j} = 0, \tag{13}$$

the finite difference representation of Eq. (11), provided $[\omega]$ and $[\mathbf{k}]$ are defined consistently in Eqs. (12) and (13).

Turn now to the Vlasov equation, which in linearized form reads

$$\frac{\partial}{\partial t} f + \mathbf{v} \cdot \frac{\partial}{\partial \mathbf{x}} f = -\mathbf{F} \cdot \frac{\partial}{\partial \mathbf{v}} f^0. \tag{14}$$

Here, f^0 and f are the zero- and first-order particle distribution functions. We assume that there are enough particles per grid cell (a half-dozen or so, depending on circumstances. See, e.g., [20, Fig. 3]) that f^0, f, and their first derivatives can be treated as continuous functions. The force $\mathbf{F}$ is taken as a first-order quantity. Thus, we exclude from consideration external (i.e., zero-order) forces. (Langdon [21] has considered at length the case of a constant external magnetic field in electrostatic simulations.) For definiteness, we consider only the standard leap-frog particle integration algorithm, in which particle positions and forces are defined at integer times and particle velocities at half-integer times. Note, however, that any algorithm differing from leap-frog by terms of second or higher order in $F \Delta t$ must yield the same linear dispersion relation. (Such terms are dropped in linearizing in F.) Charge and current densities are determined at integer and half-integer times, respectively.

Ordinarily, in deriving dispersion relations for particle-codes, one abandons the differential form of the linearized Vlasov equation, Eq. (14), in favor of the integral form [3, 4], using it to interconnect f evaluated at successive integer or half-integer time steps. It is, however, possible to proceed directly from the differential equation, thereby obtaining a simpler and perhaps more convincing derivation, provided one notices an interesting fact: In the leap-frog algorithm, the particle equations of motion may be thought of as integrated exactly, although with only approximate forces.

$$\frac{d}{dt}\mathbf{x} = \mathbf{v}, \tag{15}$$

$$\frac{d}{dt}\mathbf{v} = \Sigma_{m,\mathbf{n}}\mathbf{F}_{m,\mathbf{n}}S_1(\mathbf{x} - \mathbf{x}')\,\Delta t\;\delta(t - m\,\Delta t). \tag{16}$$

Here, $(\mathbf{x}')_i = (\mathbf{n})_i(\Delta\mathbf{x})_i$, with $\mathbf{n}$ a vector of indices labeling points on the spatial mesh. $S_1(\mathbf{x})$ is the interpolation function for the forces. Strictly, $\mathbf{F}S_1$ should be replaced in Eq. (16) by $\mathbf{F} \cdot \mathbf{S}_1$, $\mathbf{S}_1$ a matrix of interpolation functions, to allow separate interpolation algorithms for the various components of $\mathbf{F}$. For the sake of notational simplicity, we forego this minor generalization. The finite difference analog of Eq. (14) is, then, simply

$$\frac{\partial}{\partial t}f(t, \mathbf{x}, \mathbf{v}) + \mathbf{v}\cdot\frac{\partial}{\partial \mathbf{x}}f(t, \mathbf{x}, \mathbf{v})$$
$$= -\Sigma_{m,\mathbf{n}}\mathbf{F}_{m,\mathbf{n}}\cdot\frac{\partial}{\partial \mathbf{v}}f^0S_1(\mathbf{x} - \mathbf{x}')\,\Delta t\;\delta(t - m\,\Delta t). \tag{17}$$

Its Fourier transform reads

$$f(\omega', \mathbf{k}', \mathbf{v}) = i\mathbf{F}_{\omega,\mathbf{k}}\cdot\frac{\partial}{\partial \mathbf{v}}f^0S_1(\mathbf{k}')/(\omega' - \mathbf{k}'\cdot\mathbf{v}), \tag{18}$$

where $\omega' = \omega + \mu\omega_g$ and $(\mathbf{k}')_i = (\mathbf{k})_i + (\boldsymbol{\nu})_i(\mathbf{k}_g)_i$, with $\omega_g = 2\pi/\Delta t$ and $(\mathbf{k}_g)_i = 2\pi/(\Delta\mathbf{x})_i$. The indices μ and $\boldsymbol{\nu}$ are analogous to m and $\mathbf{n}$ above.

The force $\mathbf{F}$ in Eq. (16) is $\mathbf{E} + \mathbf{v} \times \mathbf{B}$. From Eqs. (9) and (10), its transform is

$$\mathbf{F} = i(\phi - \mathbf{v}\cdot\mathbf{A})[\mathbf{k}] - i([\omega] - [\mathbf{k}]\cdot\mathbf{v})\,\mathbf{A}. \tag{19}$$

The perturbed charge and current densities are related to f by

$$\rho_{m,\mathbf{n}} = \iint S_2(\mathbf{x} - \mathbf{x}')\,f(m\,\Delta t, \mathbf{x}, \mathbf{v})\,d\mathbf{x}\,d\mathbf{v}, \tag{20}$$

$$\mathbf{j}_{m,\mathbf{n}} = \iint S_2(\mathbf{x} - \mathbf{x}')\,\mathbf{v}f((m + \tfrac{1}{2})\,\Delta t, \mathbf{x}, \mathbf{v})\,d\mathbf{x}\,d\mathbf{v}. \tag{21}$$

$S_2(\mathbf{x})$ represents the interpolation algorithms for ρ and $\tilde{\mathbf{j}}$, which, incidentally, need not be the same [7], although again we do not explicitly take account of this possibility. It is, moreover, not always true that ρ, $\tilde{\mathbf{j}}$, and $\mathbf{F}$ are defined on the same spatial mesh. However, any resulting phase factors in the Fourier transforms can be absorbed into $S_2(\mathbf{k})$, so that the form of the equations is unchanged. The same is true for any additional smoothing imposed on the charge or currents.

$$\rho_{\omega,\mathbf{k}} = \Sigma_{\mu,\mathbf{v}} S_2(-\mathbf{k}') \int f(\omega', \mathbf{k}', \mathbf{v})\, d\mathbf{v}, \tag{22}$$

$$\tilde{\mathbf{j}}_{\omega,\mathbf{k}} = \Sigma_{\mu,\mathbf{v}} S_2(-\mathbf{k}')(-1)^{\mu} \int \mathbf{v} f(\omega', \mathbf{k}', \mathbf{v})\, d\mathbf{v}. \tag{23}$$

The factor $(-1)^{\mu}$, arising because $\tilde{\mathbf{j}}$ is determined at half-integer times, is quite significant.

The foregoing equations can be combined to give

$$\begin{aligned}
[\omega^2 - \mathbf{k}^2]\, \phi + i[\omega]\, \Lambda = {} & -\Sigma_{\mu,\mathbf{v}} S_1(\mathbf{k}')\, S_2(-\mathbf{k}') \\
& \times \int \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \\
& \cdot \frac{\partial}{\partial \mathbf{v}} f^0 (\omega' - \mathbf{k}' \cdot \mathbf{v})^{-1}\, d\mathbf{v},
\end{aligned} \tag{24}$$

$$\begin{aligned}
[\omega^2 - \mathbf{k}^2]\, \mathbf{A} + i[\mathbf{k}]\, \Lambda = {} & -\Sigma_{\mu,\mathbf{v}} S_1(\mathbf{k}')\, S_2(-\mathbf{k}') \\
& \times \int \{(-1)^{\mu}\, (\mathbf{v} - [\mathbf{k}][\mathbf{k}] \cdot \mathbf{v}/[\mathbf{k}]^2) + ([\mathbf{k}][\omega]/[\mathbf{k}]^2)\} \\
& \times \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \\
& \cdot \frac{\partial}{\partial \mathbf{v}} f^0 (\omega' - \mathbf{k}' \cdot \mathbf{v})^{-1}\, d\mathbf{v},
\end{aligned} \tag{25}$$

or, if the sum on μ is explicitly evaluated [22],

$$\begin{aligned}
[\omega^2 - \mathbf{k}^2]\, \phi + i[\omega]\, \Lambda = {} & -\Sigma_{\mathbf{v}} S_1(\mathbf{k}')\, S_2(-\mathbf{k}') \int \frac{\Delta t}{2} \operatorname{ctn} \left((\omega - \mathbf{k}' \cdot \mathbf{v}) \frac{\Delta t}{2} \right) \\
& \times \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \cdot \frac{\partial}{\partial \mathbf{v}} f^0\, d\mathbf{v},
\end{aligned} \tag{26}$$

$$\begin{aligned}
[\omega^2 - \mathbf{k}^2]\, \mathbf{A} + i[\mathbf{k}]\, \Lambda = {} & -\Sigma_{\mathbf{v}} S_1(\mathbf{k}')\, S_2(-\mathbf{k}') \\
& \times \int \left\{ (\mathbf{v} - [\mathbf{k}][\mathbf{k}] \cdot \mathbf{v}/[\mathbf{k}]^2) \frac{\Delta t}{2} \csc \left((\omega - \mathbf{k}' \cdot \mathbf{v}) \frac{\Delta t}{2} \right) \right. \\
& \left. + ([\mathbf{k}][\omega]/[\mathbf{k}]^2) \frac{\Delta t}{2} \operatorname{ctn} \left((\omega - \mathbf{k}' \cdot \mathbf{v}) \frac{\Delta t}{2} \right) \right\} \\
& \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \cdot \frac{\partial}{\partial \mathbf{v}} f^0\, d\mathbf{v}.
\end{aligned} \tag{27}$$

To solve this sytem, we must specify Λ. The desired dispersion relations are then just the vanishing of the three nontrivial eigenvalues of the matrix of coefficients of ϕ and $\mathbf{A}$ from either (24) and (25) or (26) and (27). The various dispersion relations given in the literature can readily be obtained as special cases.

3. Numerical Instabilities

It is not possible to provide in a single article any reasonably complete analysis of the electromagnetic dispersion relation. We shall, therefore, limit ourselves to a few general remarks together with one specific calculation.

The most striking, though probably not the most important, aberration introduced by finite differencing is the asymmetry of (26) and (27). Thus, the coefficient of $\mathbf{A}$ in Eq. (26) does not equal the coefficient of ϕ in Eq. (27). There are two sources of asymmetry, the factor $(-1)^{\mu}$ in Eq. (25), due to defining $\mathbf{j}$ and $\mathbf{F}$ at different times, and the existence of alias vectors $\mathbf{k}'$ not parallel to $[\mathbf{k}]$ (in multidimensional simulations). Without symmetry, we cannot guarantee the eigenvalues of the matrix of coefficients to be real. What the complete ramifications of this situation are is unclear.

Of more immediate concern is the infinity of additional resonances introduced by spatial and temporal aliasing. Each may be thought of as creating nonphysical normal modes which are equivalent to physical normal modes in every respect but one. The phase velocity $\omega'\mathbf{k}'/|\,\mathbf{k}'\,|^2$ is incorrect. Just as the true modes can interact to drive two basic instabilities, the Weibel [23] and two-stream [24], so we must expect the alias modes to interact with both physical modes and, to a lesser extent, one another to trigger analogous numerical instabilities. Indeed, the effects of aliases would render particle codes worthless were it not for the fact that most of the nonphysical oscillations have very low amplitudes. If, for instance, ω' is sufficiently large, the alias phase velocity exceeds that of most particles, so that the waves have no source of energy. On the other hand, if $\mathbf{k}'$ is too large, the phase velocity falls on the flat, central portion of $f^0(v)$, and the mode again is but weakly coupled to the plasma. Additionally, the interpolation factors $S_1(\mathbf{k}')\,S_2(-\mathbf{k}')$ strongly suppress large $\mathbf{k}'$ modes. Conventional wisdom states that aliases are squeezed out between these two limits, if $\omega_p \Delta t$ and $\mathbf{k}_D \cdot \Delta\mathbf{x}$ are both chosen to be modest fractions of unity. The components of $\mathbf{k}_D$ are the reciprocal Debye lengths in the corresponding directions.

A third major modification of the dispersion relation involves the coefficients $\mathbf{A} \cdot \partial f^0/\partial \mathbf{v}$ in Eqs. (26) and (27), in both cases a sum of terms each proportional to

$$([\omega] - [\mathbf{k}] \cdot \mathbf{v})(\Delta t/2)/\sin((\omega - \mathbf{k}' \cdot \mathbf{v})(\Delta t/2)). \tag{28}$$

In the limit of vanishing $\Delta\mathbf{x}$ and Δt, (28) reduces to unity, and so contributes no physical resonances whatever. However, for finite $\Delta\mathbf{x}$ and Δt no such cancellation occurs, and there are spurious modes for every ω' and $\mathbf{k}'$, including the fundamental frequency and wavenumber, ω and $\mathbf{k}$. In other words, problems associated with the disparity between numerator and denominator persist even when alias effects are suppressed. Tracing through the derivation in Section 2, we can see easily how the mismatch arises. The numerator of (28) is the finite difference representation of $\omega - \mathbf{k} \cdot \mathbf{v}$ from the basically Eulerian treatment of interparticle forces, while the denominator is a second representation of that same quality but due to the Lagrangian treatment of particle positions. In this sense, there is an essential incompatibility at the heart of most particle code algorithms. It has been shown that the extraneous modes can couple directly into light waves of matching phase velocity to produce a violent numerical instability. What effect, if any, these modes have upon electrostatic oscillations is, as yet, unknown.

The Coulomb gauge,

$$[\mathbf{k}] \cdot \mathbf{A} = 0, \tag{29}$$

is most commonly employed in plasma simulations. Specializing to this gauge, we obtain from Eqs. (24) and (25)

$$[\mathbf{k}]^2 \phi = \Sigma_{\mu\nu} S_1(\mathbf{k}')\, S_2(-\mathbf{k}') \int (\omega' - \mathbf{k}' \cdot \mathbf{v})^{-1} \times \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \cdot \frac{\partial}{\partial \mathbf{v}} f^0\, d\mathbf{v}, \tag{30}$$

$$[\omega^2 - \mathbf{k}^2]\, \mathbf{A} = -\Sigma_{\mu\nu} S_1(\mathbf{k}')\, S_2(-\mathbf{k}')(-1)^{\mu} \int \mathbf{v}_{\perp}(\omega' - \mathbf{k}' \cdot \mathbf{v})^{-1} \times \{([\omega] - [\mathbf{k}] \cdot \mathbf{v})\, \mathbf{A} - (\phi - \mathbf{v} \cdot \mathbf{A})[\mathbf{k}]\} \cdot \frac{\partial}{\partial \mathbf{v}} f^0\, d\mathbf{v}. \tag{31}$$

By $\mathbf{v}_{\perp}$ we mean the component of $\mathbf{v}$ perpendicular to $[\mathbf{k}]$,

$$\mathbf{v}_{\perp} = \mathbf{v} - [\mathbf{k}][\mathbf{k}] \cdot \mathbf{v}/[\mathbf{k}]^2. \tag{32}$$

Note that no generality has been lost in choosing (29), since for any differencing scheme in any gauge one can always construct an equivalent scheme in any other gauge.

The Maxwellian distribution is a convenient choice for $f^0(\mathbf{v})$ in that we are able to evaluate explicitly all the integrals in (30) and (31). It is, moreover, a rational starting point for any general stability analysis, since an algorithm incapable of treating a physically stable equilibrium distribution is almost certainly of little use in simulating more complicated plasma configurations, where it may not be possible to distinguish between real and numerical phenomena. With a modest amount of algebra, we find

$$[\mathbf{k}]^2\,\phi = \Sigma_{\mu\nu} S_1(\mathbf{k}')\,S_2(-\mathbf{k}')\,\frac{[\mathbf{k}]\cdot\mathbf{k}'\phi - [\omega]\,\mathbf{k}'\cdot\mathbf{A}}{2\,|\,\mathbf{k}'\,|^2}\,Z'(\zeta), \tag{33}$$

$$\begin{aligned}[\omega^2 - \mathbf{k}]^2\,\mathbf{A} = {} & \Sigma_{\mu,\nu} S_1(\mathbf{k}')\,S_2(-\mathbf{k}')(-1)^{\mu} \\ & \times \left(\frac{[\mathbf{k}]\cdot\mathbf{k}'}{|\,\mathbf{k}'\,|^2} - \left\{1 + \left(\frac{[\omega]\,|\,\mathbf{k}'\,|^2}{\omega'[\mathbf{k}]\cdot\mathbf{k}'} - 1\right)\zeta Z(\zeta) + \frac{1}{2}\,Z'(\zeta)\right\}\mathbf{A}\right. \\ & \left. + \frac{\mathbf{k}_\perp'}{\sqrt{2}\,|\,\mathbf{k}'\,|\,\mathbf{v}_T}\,\frac{[\mathbf{k}]\cdot\mathbf{k}'\phi - [\omega]\,\mathbf{k}'\cdot\mathbf{A}}{|\,\mathbf{k}'\,|^2}\,\{Z(\zeta) - \zeta Z'(\zeta)\}\right).\end{aligned} \tag{34}$$

$Z(\zeta)$ and $Z'(\zeta)$ are the plasma dispersion function and its first derivative [25] with argument

$$\zeta = \omega'/\sqrt{2}\,|\,\mathbf{k}'\,|\,v_T\,. \tag{35}$$

The quantity $\mathbf{k}_\perp'$ is that part of $\mathbf{k}'$ perpendicular to $[\mathbf{k}]$, while v_T is the plasma thermal velocity.

For one-dimensional simulations, in which $[\mathbf{k}]$ and $\mathbf{k}'$ are parallel, Eqs. (33) and (34) decouple. Langdon and others have studied in detail the resulting electrostatic dispersion relation [4, 5]. The transverse electromagnetic dispersion relation may conveniently be rewritten as

$$[\omega^2 - k^2] = \Sigma_{\mu,\nu} S_1(k')\,S_2(-k')(-1)^{\mu}\,([\omega]/\omega' - 2[k]/k')\,\zeta Z(\zeta). \tag{36}$$

There are many ways in which Eq. (36) can give unstable modes of oscillation. In the first place, the vacuum equation $[\omega^2 - k^2] = 0$ may itself predict instability, for k near $k_{\max} = k_g/2$. As explained in the Introduction, stability of the vacuum equation implies the Courant condition that Δt be less than some expression involving Δx, often simply $\Delta t \leqslant \Delta x$. When plasma reactance from the right side of Eq. (36) is included, a more restrictive Courant condition becomes necessary. However, it is easy to show that for realistic simulation parameters the change in Courant condition is minor. (For an example of the influence of plasma reactance on light wave stability, see Ref. [3, Section 5].) In any event, by choosing Δt sufficiently small one can always avoid such instabilities.

If we could neglect all but the $\mu = \nu = 0$ term in the double sum in Eq. (36), we would find that light waves are Landau-damped at a rate given approximately by $\pi^{1/2}\zeta\exp(-\zeta)$. For v_T small, as is appropriate to nonrelativistic simulations, the damping is quite weak and so is unobjectionable. In fact, the aliases usually are not negligible, and each contributes a term like that of the fundamental but multiplied by the corresponding coefficient of $-\zeta Z(\zeta)$ from (36). Since the coefficients are of both signs, some contribute to Landau growth rather than damping.

Determining analytically which dominates is a messy process, more so than in the electrostatic case, because time aliases cannot be ignored. Nonetheless, the results are very similar to those of electrostatic simulations, namely, that appreciable instability occurs only for $\Delta x/\lambda_D > 1$. However, instability growth rates are smaller by factors of order v_T^2. In combination, these two rules indicate that Δx must be of order unity for strong numerical growth, but such large cells are objectionable for other reasons and in practice would scarcely ever be employed. Figure 1 gives the frequencies and growth rates computed numerically from Eq. (36) for the differencing scheme described in Section 4 of Ref. [3] with $\Delta t = \Delta x = 0.2, \lambda_D = 0.1, \beta = -0.01$, and $N = 1$. Note that β is reduced slightly from its ideal value of zero in order to avoid possible reactive effects.

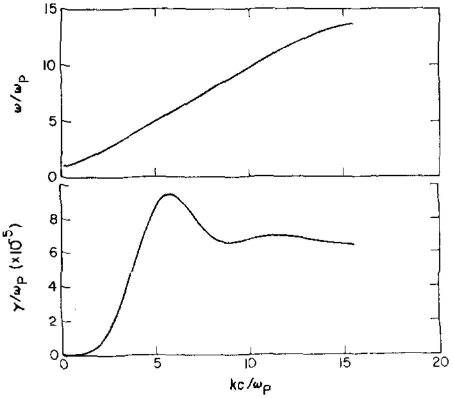

FIG. 1. Numerical solution of Eq. (36) for the "implicit differencing scheme" [3, 8] with $\Delta x = \Delta t = 0.2$, $\beta = -0.01$, and $\lambda_D = 0.1$.

These conclusions do not, incidentally, necessarily hold true for relativistic simulations, in which $\partial f^0/\partial v$ often is very large near $v = 1$. A single time alias with phase velocity in this region possibly could dominate the sum in (36), providing strong Landau-growth irrespective of $\Delta x/\lambda_D$. The only reasonably sure way to guarantee stability here is to force $\omega/k > 1$, although use of an "energy-conserving" scheme [7] might help.

In the absence of numerical distortion, Eq. (36) also admits zero-frequency solutions heavily damped except near $k = 0$. One expects that the addition of aliases might drive this mode unstable, creating something reminiscent of a Weibel instability [23]. For this to happen, the right side of Eq. (36) must be negative for k small. Because of the many terms involved, we have not sought analytic criteria for instability. Instead, a brief computational search of (36) for low-frequency numerical instabilities was carried out but proved unsuccessful. This does not, of course, preclude such instabilities.

Returning to the multidimensional simulation dispersion relations of Eqs. (33) and (34), we encounter only two qualitatively new features, alias wavevectors not parallel to $[\mathbf{k}]$ and coupling between longitudinal and transverse modes. The first of these should not prove too important for electrostatic or high-frequency electromagnetic waves, since alias contributions to Landau growth tend to be proportional to $[\mathbf{k}] \cdot \mathbf{k}'$. Counterstreaming aliases remain, in general, the most significant. It is possible, however, that off-angle aliases may produce the appearance of temperature anisotrophy, increasing the chances for spurious Weibel instabilities. This seems especially likely, if an anisotrophic grid is employed. The consequence of coupling between (33) and (34) remains an open question.

We have not yet encountered the numerical Cherenkov instability per se. In order to investigate its interesting behavior, let us expand Eq. (31) for small $\mathrm{Im}(\omega)$ to obtain, for $[k] > 0$,

$$\begin{aligned}\mathrm{Im}(\omega) \approx \pi \Sigma_{\mu,\nu} S_1(k')\, S_2(-k')(k')^{-1}\, (-1)^{\mu}\, [k]\, \partial\langle v_\perp{}^2\rangle(\omega'/k')/\partial v \\ - \langle f^0\rangle(\omega'/k')([\omega] - \omega'[k]/k')/\partial[\omega^2 - k^2]/\partial\omega, \qquad (37)\end{aligned}$$

where

$$\langle f^0\rangle(v) = \int f^0(v, \mathbf{v}_\perp)\, d\mathbf{v}_\perp\,, \qquad (38)$$

$$\langle v_\perp{}^2\rangle(v) = \int f^0(v, \mathbf{v}_\perp)\, v_\perp{}^2\, d\mathbf{v}_\perp\,. \qquad (39)$$

For simplicity, we have assumed a one-dimensional simulation with $f^0(v, \mathbf{v}_\perp)$ isotropic in the transverse velocities.

When thermal velocities are comparatively small, as in beam problems, the term containing $\langle v_\perp{}^2\rangle\,(v)$ can be ignored. Moreover, with $\langle f^0\rangle\,(v)$ sharply peaked, usually only one alias contributes strongly to the sum.

$$I_m(\omega) \approx -\pi S_1(k')\, S_2(-k')(k')^{-1}(-1)^{\mu}\langle f^0\rangle(\omega'/k')([\omega] - \omega'[k]/k')/\partial[\omega^2 - k^2]/\partial\omega \qquad (40)$$

With this expression all the qualitative results of Ref. [3] on the Cherenkov instability can be reproduced straightforwardly. Incidentally, working back from Eq. (40) into Section 2, one finds that the Cherenkov instability arises when, due to numerical effects, resonant particle velocities oscillate in phase, rather than 180° out of phase, with the vector potential of a light wave.

To the extent that thermal effects, as represented by the $\partial\langle v_\perp{}^2\rangle/\partial v$ term, are not negligible, they tend to ameliorate the Cherenkov instability.

4. Lagrangian Treatment of Velocities

The analysis we have performed thus far depends to a fair degree on the particle velocity differencing scheme, represented by Eqs. (16) and (19). These expressions constitute a natural finite difference representation of the differential equation

$$\frac{d}{dt}\mathbf{v} = \mathbf{E} + \mathbf{v} \times \mathbf{B}. \tag{41}$$

Let us recast (41) into the equivalent form [26],

$$\frac{d}{dt}(\mathbf{v} + \mathbf{A}) = \nabla(\mathbf{v} \cdot \mathbf{A} - \phi). \tag{42}$$

The combination $\mathbf{v} + \mathbf{A}$ is, of course, the particle canonical momentum. A part of the force is now in the form of a total, or convective, derivative $d\mathbf{A}/dt$, which can be finite differenced within the Lagrangian rather than the Eulerian framework. This seemingly small change is, in fact, critical: By partially removing the Eulerian–Lagrangian mismatch previously discussed, we eliminate the Cherenkov instability.

The leap-frog algorithm corresponding to Eq. (42) can be represented as

$$\begin{aligned}\frac{d}{dt}\mathbf{v} = \Sigma_{m,\mathbf{n}}\{&\mathbf{A}_{m-(1/2),\mathbf{n}}S_1(\mathbf{x} - \mathbf{v}\,\Delta t/2 - \mathbf{x}')\\ &- \mathbf{A}_{m+(1/2),\mathbf{n}}S_1(\mathbf{x} + \mathbf{v}\,\Delta t/2 - \mathbf{x}') + \mathbf{F}_{m,\mathbf{n}}S_1(\mathbf{x} - \mathbf{x}')\,\Delta t\}\,\delta(t - m\,\Delta t),\end{aligned} \tag{43}$$

where by $\mathbf{F}$ we mean the finite difference expression for $\nabla(\mathbf{v} \cdot \mathbf{A} - \phi)$. Once again, the interpolation functions $S_1(\mathbf{x})$ are taken to be the same only for notational simplicity. (In practice, one may wish to use only an approximation to (43) in order to increase computation speed; see the Appendix. So long as the algorithm actually employed agrees to first order with the center-differenced-scheme, and no external fields are involved, the linear dispersion relations are unchanged.) If, now, we repeat the derivation of Section 2, replacing (16) by (43), we find that Eqs. (24)–(27) are unchanged except for the substitution

$$([\omega] - [\mathbf{k}] \cdot \mathbf{v})\,\Delta t/2 \rightarrow \sin((\omega - \mathbf{k}' \cdot \mathbf{v})\,\Delta t/2). \tag{44}$$

The replacement is precisely that needed to reduce (28) to unity, so that the Cherenkov instability is indeed eliminated.

In most instances the Cherenkov modes are removed only at the cost of introducing a spurious low-frequency electrostatic mode, typically leading to severe instability. However, in contrast to the conventional differencing schemes of the preceding section, the present algorithm is not gauge invariant: The use of differing gauges yields differing stability properties. In particular, the Coulomb gauge, because it explicitly separates the longitudinal and transverse fields equations, is not subject to this new numerical problem. (This situation is more readily understandable within the context of finite element methods [27].) Invoking this gauge, we obtain

$$
\begin{aligned}
[\mathbf{k}^2]\,\phi = \Sigma_{\mu,\nu} S_1(\mathbf{k}')\,S_2(-\mathbf{k}') \int \Big\{ &\cos\Big((\omega - \mathbf{k}'\cdot\mathbf{v})\,\frac{\Delta t}{2}\Big)\,\mathbf{A} \\
&- (\omega' - \mathbf{k}'\cdot\mathbf{v})^{-1}\,(\phi - \mathbf{v}\cdot\mathbf{A})[\mathbf{k}]\Big\} \cdot \frac{\partial f^0}{\partial \mathbf{v}}\,d\mathbf{v}, \qquad (45)
\end{aligned}
$$

$$
\begin{aligned}
[\omega^2 - \mathbf{k}^2]\,\mathbf{A} = \Sigma_{\mu,\nu} S_1(\mathbf{k}')\,S_2(-\mathbf{k}')\,\Big\{ &\mathbf{A} - (-1)^{\mu} \int \mathbf{v}_{\perp}(\omega' - \mathbf{k}'\cdot\mathbf{v})^{-1} \\
&\times (\phi - \mathbf{v}\cdot\mathbf{A})[\mathbf{k}] \cdot \frac{\partial f^0}{\partial v}\,d\mathbf{v}\Big\}. \qquad (46)
\end{aligned}
$$

Both (45) and (46) are somewhat simpler than their counterparts (30) and (31) of Section 3. The unusual cosine term in Eq. (45) should prove to be comparatively unimportant, since it is of order $(\Delta t)^2$ and involves no resonances.

For the reasons outlined in Section 3, we specialize to the case of a Maxwellian simulation plasma, obtaining

$$
\begin{aligned}
[\mathbf{k}^2]\,\phi = \Sigma_{\mu,\nu} &S_1(\mathbf{k}')\,S_2(-\mathbf{k}') \\
&\times \Bigg(\frac{[\mathbf{k}]\cdot\mathbf{k}'}{|\,\mathbf{k}'\,|^2} \Bigg\{ \frac{\phi - \omega'\mathbf{k}'\cdot\mathbf{A}/|\,\mathbf{k}'\,|^2}{2v_T{}^2}\,Z'(\zeta) + \frac{\mathbf{k}'\cdot\mathbf{A}}{\sqrt{2}\,|\,\mathbf{k}'\,|\,v_T}\,Z(\zeta)\Bigg\} \\
&- \frac{\Delta t}{2}\,\sin\Big(\omega\,\frac{\Delta t}{2}\Big)\,\exp(-\mathbf{k}'^2 v_T{}^2\,\Delta t^2/8)\,\mathbf{k}'\cdot\mathbf{A}\Bigg\}, \qquad (47)
\end{aligned}
$$

$$
\begin{aligned}
[\omega^2 - \mathbf{k}^2]\,\mathbf{A} = \Sigma_{\mu,\nu} S_1(\mathbf{k}')\,S_2(-\mathbf{k}')\,\Bigg(&\Bigg\{1 + \frac{[\mathbf{k}]\cdot\mathbf{k}'}{2\,|\,\mathbf{k}'\,|^2}\,Z(\zeta)\Bigg\}\,\mathbf{A} \\
&+ \frac{\mathbf{k}_{\perp}'}{\sqrt{2}\,|\,\mathbf{k}'\,|\,v_T}\,\frac{[\mathbf{k}]\cdot\mathbf{k}'\phi - [\omega]\,\mathbf{k}'\cdot\mathbf{A}}{|\,\mathbf{k}'\,|^2}\,\{Z(\zeta) - \zeta Z'(\zeta)\}\Bigg). \qquad (48)
\end{aligned}
$$

The analysis of Eqs. (47) and (48) for one-dimensional simulations yields results similar to those previously obtained for Eqs. (33) and (34). The growth rates of numerical high-frequency electromagnetic instabilities are reduced somewhat, though this hardly seems important. Figure 2 illustrates the change. Again, we have been unable to find any low-frequency instabilities. For multidimensional simulations, the coupling between longitudinal and transverse waves probably is a bit stronger here due to the added terms in Eq. (47). We suspect that this extra coupling does not significantly modify stability properties, especially for modest values of Δt and $\Delta\mathbf{x}$.

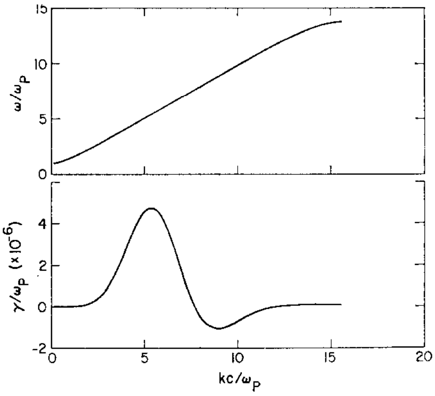

FIG. 2. Numerical solution of Eq. (48) for a one-dimensional simulation with $\Delta x = \Delta t = 0.2$, $\beta = -0.01$, and $\lambda_D = 0.1$.

In summary, the use of canonical momenta in advancing particle velocities solves the problem of numerical Cherenkov instabilities but does not appear to have much effect on other aspects of particle code numerical stability. We have implemented a version of this algorithm in an otherwise standard one-dimensional electromagnetic plasma simulation code [12]. With a single cold beam, the code was stable for $v_0 \Delta t$ far in excess of the stability limits derived in Ref. [3]. Successful simulations were performed also for hot Maxwellian distributions.

Incidentally, in Ref. [3] we observed large, sharp spikes in the transverse field power spectra at $\omega = kv_0$ for numerically stable cold beam simulations. Figure 3 of Ref. [3] is an example. These spikes were attributed to the Cherenkov resonances of Eq. (31). However, we have found in simulations employing the canonical momentum algorithm that these spikes not only persist but are enhanced slightly. The resonance explanation must, therefore, be incorrect. We now believe that the spikes are nothing more than highly energetic van Kampen modes [28] or, equivalently, Doppler-shifted frozen-in noise from the initial particle loading. (That these spikes might be due to van Kampen modes was suggested by Dawson [29].) In partial verification of this hypothesis, we have produced similar spikes in the longitudinal field spectra by introducing short wavelength noise into the charge density at time zero. (Ordinarily, particles are loaded uniformly in space.)

5. Global Conservation Laws

Before concluding, we wish to exploit the differential form of the simulation plasma Vlasov equation to draw a few general conclusions on momentum and energy conservation. The exact (not linearized as in Eq. (17)) Vlasov equation for the standard time-centered leap-frog algorithm is

$$\frac{\partial}{\partial t}f + \mathbf{v}\cdot\frac{\partial}{\partial \mathbf{x}}f + \Sigma_{m,\mathbf{n}}(\mathbf{E}_{m,\mathbf{n}} + \mathbf{v}\times\mathbf{B}_{m,\mathbf{n}}) \cdot\frac{\partial}{\partial \mathbf{v}}fS_1(\mathbf{x}-\mathbf{x}')\,\Delta t\,\delta(t-m\,\Delta t) = 0. \tag{49}$$

Taking the first velocity moment and then integrating over **x** gives us the time rate of change of the particle momentum,

$$\mathbf{P}_{m+(1/2)} - \mathbf{P}_{m-(1/2)} = \Sigma_{\mathbf{n}}(\bar{\rho}_{m,\mathbf{n}}\mathbf{E}_{m,\mathbf{n}} + \bar{\mathbf{j}}_{m,\mathbf{n}}\times\mathbf{B}_{m,\mathbf{n}})\,\Delta t, \tag{50}$$

where

$$\bar{\rho}_{m,\mathbf{n}} = \iint S_1(\mathbf{x}-\mathbf{x}')f(m\,\Delta t,\mathbf{x},\mathbf{v})\,d\mathbf{x}\,d\mathbf{v}, \tag{51}$$

$$\bar{\mathbf{j}}_{m,\mathbf{n}} = \iint S_1(\mathbf{x}-\mathbf{x}')\,\mathbf{v}f(m\,\Delta t,\mathbf{x},\mathbf{v})\,d\mathbf{x}\,d\mathbf{v}. \tag{52}$$

We would like to eliminate $\bar{\rho}$ and $\bar{\mathbf{j}}$ in terms of **E** and **B** via Maxwell's equations in order to obtain the Poynting flux on the right side of Eq. (50), thereby demonstrating conservation of total (i.e., particle plus field) momentum. However, ρ an **j**, not $\bar{\rho}$ and $\bar{\mathbf{j}}$, appear in the finite differenced Maxwell's equations of Section 2. We can arrange that $\rho = \bar{\rho}$ by choosing $S_1 = S_2$, but **j** and $\bar{\mathbf{j}}$ can never be equal, or even related in a simple way, because they are defined at different times. Therefore, total momentum is never conserved except in the special case of $S_1 = S_2$ in an electrostatic simulation.

Similarly, we obtain the time evolution of the particle energy by forming the second velocity moment of (49),

$$U_{m+(1/2)} - U_{m-(1/2)} = \Sigma_{\mathbf{n}}\bar{\mathbf{j}}_{m,\mathbf{n}}\cdot\mathbf{E}_{m,\mathbf{n}}\,. \tag{53}$$

Here, clearly, the right side of Eq. (53) cannot under any circumstance be transformed into the field energy, so the total energy is never conserved exactly. For both energy and momentum, the lack of an exact conservation law is due to determining the current at a time different from that of the forces. Defining both at the same time, on the other hand, does not seem practical. Fortunately, experience indicates that energy and momentum are, in general, approximately conserved to sufficiently high order. The dilemma posed is, therefore, not burdensome in practice.

We have calculated corresponding expressions for the canonical momentum algorithm, although the results are too cumbersome to be reproduced profitably here. Again, usually neither momentum nor energy is conserved exactly. There is, however, one important exception: If total momentum is defined as the sum of particle canonical momenta, then it is conserved in those directions along which there is no spatial variation. Because this result follows directly from Eq. (42), it is true even for many non-time centered schemes.

The existence of simulation particle exact equations of motion should make possible other interesting, and certainly more powerful, studies. In particular, the development of multiparticle correlation functions and their associated theory [30] suggests itself. From there, one could use simulations to test some of the basic statistical assumptions on which much of plasma theory is based. All this is, of course, no small undertaking.

6. Conclusions

On the basis of our analysis, it appears that the numerical stability problems encountered in multidimensional electromagnetic plasma simulations should be no more severe than those dealt with at present in one-dimensional electrostatic simulations, provided the canonical momentum algorithm is used. However, we will have greater confidence when investigations with non-Maxwellian distributions have been performed and the effects of coupling between longitudinal and transverse modes better delineated. The need to consider external fields and nonperiodic boundary conditions must also be stressed. Most of all, considerably more practical experience is required.

Appendix

It is quite convenient to implement the canonical momentum algorithm of Section 4 for one-dimensional electromagnetic simulations, because longitudinal and transverse quantities are so weakly coupled. For nonrelativistic problems exact time-centering is possible [12] (Nielson [31] has independently considered nonrelativistic implementations in both one and two dimensions),

$$\begin{aligned}
V_\perp{}^m &= P_\perp{}^m(e/mc) - A_\perp^{m+(1/2)} + A_\perp^{m-(1/2)}], \\
V^{m+(1/2)} &= V^{m-(1/2)} + (E^m + \tfrac{1}{2}V_\perp{}^m(d/dx)[A_\perp^{m+(1/2)} + A_\perp^{m-(1/2)}])(e/m)\,\Delta t, \\
X^{m+1} &= X^m + V^{m+(1/2)}\,\Delta t, \\
P_\perp^{m+1} &= P_\perp{}^m.
\end{aligned}$$

Here, X is the particle position, V the longitudinal velocity, E the electrostatic field, $V_\perp$ the transverse velocity, $P_\perp$ the transverse canonical momentum, and $A_\perp$ the transverse vector potential. The species charge to mass ratio is e/m.

For consistancy as well as convenience, $A_\perp$ is treated by linear interpolation, while $dA_\perp/dx$ is treated by nearest grid point interpolation. The electric field E can be area-weighted by either method, depending on one's taste. Change and current are computed in the usual way at integer and half-integers times, respectively, by linear interpolation. The field-solving algorithm is described in Ref. [3, Section 4], although other methods are acceptable.

It appears necessary to sacrifice time-centering for relativistic simulations employing the canonical momentum approach, since the relativistic energy factor γ appears in the equations of motion. However, second-order-accurate noncentered algorithms prove satisfactory in practice.

Implementation in two dimensions is currently under investigation.

Acknowledgments

The author is endebted to Eric Lindman, Barry Newberger, Clair Nielson, Lester Thode, and others for their many valuable insights and suggestions.

References

1. R. L. Morse, *Meth. Computational Phys.* **9** (1970), 213; C. K. Birdsall and D. Fuss, *J. Computational Phys.* **3** (1969), 494.
2. A. B. Langdon, unpublished.
3. B. B. Godfrey, *J. Computational Phys.* **15** (1974), 504.
4. E. L. Lindman, *J. Computational Phys.* **5** (1970), 13; A. B. Langdon, *J. Computational Phys.* **6** (1970), 247; A. B. Langdon, in "Proceedings of the Fourth Conference on Numerical Simulation of Plasmas" (J. Boris and R. Shanny, Eds.), p. 467, Naval Research Laboratory, Washington, D.C., 1970.
5. H. Okuda, *J. Computational Phys.* **10** (1972), 475.
6. J. P. Boris and R. Lee, *J. Computational Phys.* **12** (1973), 131.
7. H. R. Lewis, *Meth. Computational Phys.* **9** (1970), 307; H. R. Lewis, *J. Computational Phys.* **10** (1972), 400; A. B. Langdon, *J. Computational Phys.* **12** (1973), 247.
8. C. Nielson and E. Lindman, in "Abstracts: Sherwood Theoretical Meeting," University of Texas, Austin, 1973, paper D9; C. Nielson and E. Lindman, in "Proceedings of the Sixth Conference on Numerical Simulation of Plasmas," p. 148, Lawrence Berkeley Laboratory, Berkeley, Calif., paper E3, 1973.
9. I. Haber, R. Lee, H. H. Klein, and J. P. Boris, in "Proceedings of the Sixth Conference on Numerical Simulation of Plasmas," p. 46, Lawrence Berkeley Laboratory, Berkeley, Calif., paper B3, 1973.
10. D. Nicholson, B. Cohen, A. N. Kaufman, C. E. Max, and A. B. Langdon, *Bull. Amer. Phys. Soc.* **18** (1973), 1361; A. B. Langdon, unpublished.
11. J. P. Boris and D. L. Book, *J. Computational Phys.* **11** (1973), 38; E. Turkel, *J. Computational Phys.* **15** (1974), 226; D. Gottlieb and E. Turkel, *J. Computational Phys.* **15** (1974), 251.

12. D. W. Forslund, E. L. Lindman, R. W. Mitchell, and R. L. Morse, LA-DC-72-721, Los Alamos Scientific Laboratory, Los Alamos, N. Mex., 1972.
13. O. Buneman, *J. Computational Phys.* **1** (1967), 517.
14. B. B. Godfrey, B. S. Newberger, and K. A. Taggart, *IEEE Plas. Sci.* **3** (1975), 68.
15. O. Buneman, *J. Computational Phys.* **11** (1973), 250.
16. J. D. Jackson, "Classical Electrodynamics," Chap. 6, Wiley, New York, 1962.
17. R. L. Morse and C. W. Nielson, *Phys. Fluids* **14** (1971), 830; O. Bunemen, "Relativistic Plasmas," p. 205, Benjamin, New York, 1968.
18. C. E. Nielson, unpublished.
19. J. P. Boris, in "Proceeding of the Fourth Conference on Numerical Simulation of Plasmas" (J. P. Boris and R. Shanny, Eds.), p. 3, Naval Research Laboratory, Washington, D.C., 1970.
20. B. B. Godfrey and B. S. Newberger, *Plas. Phys.* **17** (1975), 317.
21. A. B. Langdon, unpublished.
22. I. S. Gradshteyn and I. M. Ryznik, "Table of Integrals, Series, and Products," p. 36, Academic Press, New York, 1965.
23. E. W. Weibel, *Phys. Rev. Lett.* **2** (1959), 83.
24. J. M. Dawson, *Phys. Rev.* **118** (1960), 381.
25. B. D. Fried and S. D. Conte, "The Plasma Dispersion Function," Academic Press, New York, 1961.
26. H. Goldstein, "Classical Mechanics," pp. 19–21, Addison–Wesley, Reading, Mass., 1950.
27. B. B. Godfrey, Seventh Conference on Numerical Simulation of Plasmas, Courant Institute, New York, 2–4 June 1975.
28. N. G. van Kampen, *Physica* **21** (1955), 949; K. M. Case, *Ann. Phys.* **7** (1959), 359; I. Lerche, *J. Math. Phys.* **8** (1967), 1838.
29. J. Dawson, Sixth Conference on Numerical Simulation of Plasmas, Lawrence Berkeley Laboratory, Berkeley, Calif., 16–18 July 1973.
30. ~~I. L. Klimontovich, *J. Exptl. Theoret. Phys.* **34** (1958), 173 [*Sov. Phys. JETP* **34** (1958),~~ 119]; N. Rostoker, *Nucl. Fusion* **1** (1961), 101.
31. C. W. Nielson, private communication.