JOURNAL OF COMPUTATIONAL PHYSICS **133,** 146–159 (1997)
ARTICLE NO. CP975670

A 3-D Darwin–EM Hybrid PIC Code for Ion Ring Studies

Yu. A. Omelchenko and R. N. Sudan

Laboratory of Plasma Studies, Cornell University, Ithaca, New York

Received August 14, 1996; revised February 10, 1997

A new, 3-D electromagnetic (EM), hybrid, particle-in-cell (PIC) code, FLAME has been constructed to study low-frequency, large orbit plasmas in realistic cylindrical configurations. The stability and equilibrium of strong ion rings in magnetized plasmas are the first issues suitable for its application. In FLAME the EM-field is governed by Maxwell's equations in the quasi-neutral Darwin approximation (with displacement current neglected), the ion components are represented by discrete macro-particles, and the plasma electrons are modeled as a massless cold fluid. All physical quantities are expanded into finite Fourier series in the azimuthal (θ) direction. The discretization in the poloidal (r, z) plane is done by a finite-difference staggered grid method. The electron fluid equations include a finite scalar resistivity and macro-particles experience slowing-down collisions. A substantial reduction of computation time is achieved by enabling separate time advances of background and beam particle species in the time-averaged fields. FLAME has been optimized to run on parallel, MIMD systems, and has an object-oriented (C++) structure. The results of normal mode tests intended to verify the code ability to correctly model plasma phenomena are presented. We also investigate in 3-D the injection of a powerful annular ion beam into a plasma immersed in a magnetic cusp followed by an axially ramped applied magnetic field. A nonaxisymmetric perturbation is applied to the magnetic field and its effect on ion ring formation is analysed.

I. INTRODUCTION

The past several years have seen a rebirth of interest in the alternative fusion concepts as opposed to the tokamak approach. The field-reversed configuration (FRC) is unique in a hierarchy of possible candidates for the basis of the ultimate economical fusion reactor. The FRC is a compact system composed of a region of closed magnetic field lines surrounded by open lines [1]. Its simple topology significantly reduces design requirements for the external magnetic field coils and ensures short development cycles. The region of closed lines produced by currents flowing inside the plasma guarantees high β plasma confinement. A number of experiments have been staged recently to investigate the validity of the fundamental physical assumptions underlying a variety of FRC designs [2–5].

The concept of a large-orbit, field reversed ion ring, which stems from the original E-layer idea of Christofilos, was first proposed in 1974 [6]. The presence of large orbit ions in the fusion plasma has been shown to have a positive effect on the low frequency stability of the magnetic configuration (these studies have been reviewed in Ref. [7], which contains an extensive list of references). The FRC and ion ring configuration (IRC) have the same magnetic topology but the azimuthal diamagnetic current in the former is carried by small gyro-radius particles while in the latter it is carried by the energetic ions with orbits on the order of the major plasma radius (Fig. 1). A hybrid configuration with a significant fraction of the azimuthal current carried by large-orbit ions has also been proposed [8].

The project FIREX, which seeks to experimentally produce an ion ring with large field reversal factor $\zeta = \delta B/B > 1$ (δB is the diamagnetic self-field on axis and B is the external magnetic field), is currently underway at Cornell University. When ζ exceeds unity the resultant field at the axis of the ring is opposite in direction to the external field.

In FIREX ion rings will be produced by injecting a strong proton beam (with the number of particles $N_b \sim (2\text{–}5) \times 10^{17}$) through a magnetic cusp followed by a ramped solenoidal field (Fig. 2). The effect of the radial component of the cusp magnetic field acting on the particles results in the conversion of a substantial fraction of their axial energy (initially ~1 MeV) into azimuthal energy of magnetic gyration around the center axis, leading to the formation of a cylindrical layer of rotating ions advancing into the solenoidal field.

The formation of IRC by an energetic proton beam and its long-time stability are the primary issues for analytical and computational research in support of FIREX. Two particle-in-cell (PIC) codes have been developed to study these problems.

The $2\frac{1}{2}$-D (axisymmetric), hybrid PIC code FIRE was developed at an early stage of the experimental program and provided deep insight into the dynamics of the formation of a strong field-reversed ion ring [9]. In particular, it was found that:

• the ion beam is trapped and bunched into a ring on a fast timescale (of the order of several ion cyclotron periods)

0021-9991/97 $25.00

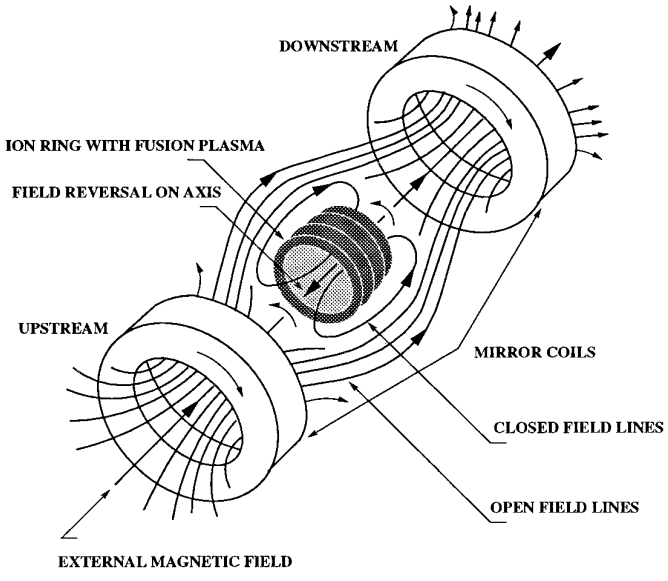

FIG. 1. Schematic drawing of the typical ion ring magnetic field reversed configuration.

by the self-generated compressional Alfvén waves propagating predominantly in the radial direction;

• the ring has a compact form and may contain up to 70% of the total number of injected particles;

• the field reversal on axis ($\zeta > 1.5$) occurs on the collisional diffusion timescale of several microseconds;

• once closed lines of force are formed the ring and the background plasma act as a collective entity with the result that the energetic ion momentum is shared with plasma ions. Thus the ring is significantly decelerated on reflection in the ramped solenoidal field compared to free particle motion, which gives enough time to switch an upstream mirror to trap the ring.

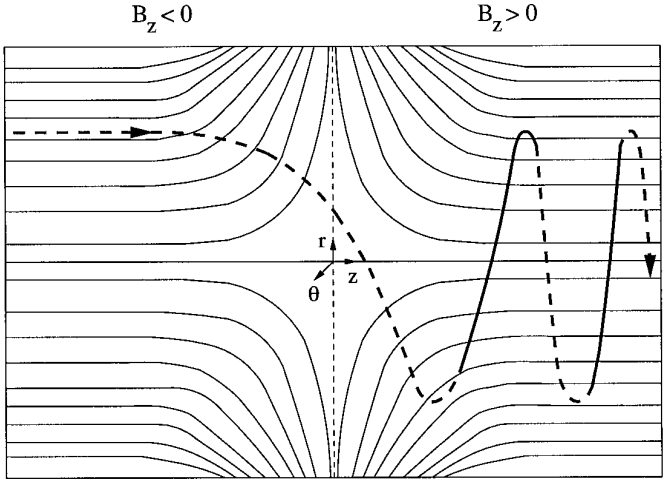

FIG. 2. Ion beam injection technique: ions pick up rotational momentum when crossing the magnetic cusp region.

An important consideration is the stability of the ion rings to low-frequency perturbations. FRC's produced by theta-pinch techniques are elongated axially and they appear grossly stable, which is consistent with theoretical considerations. The rings seen in the axisymmetric simulations [9] have a characteristic axial length ~10 cm, radial width ~5 cm, and major radius ~10 cm. Such rings are marginally stable according to the criterion developed by Ruchti and Lovelace [10]. They have also shown that the unstable rings will naturally evolve, through increase in the axial ion temperature, to stable shapes if in the initial formation process they are axially short. It has also been predicted that ion rings could stabilize compact torus configurations [11, 12, 13, 14]. A linearized 3-D hybrid code was constructed for stability studies of field-reversed ion rings [15] but lacked the ability to examine nonlinear effects that may lead to saturation of an instability. Recent hybrid simulations [16] demonstrated that ion beam injection provides an optimal way of adding the required kinetic effects to maintain a stable FRC. These studies were based on a self-consistent model that treated the background plasma as a fluid and an energetic ion component as a collisionless Vlasov species (particles). The MHD equilibrium of a partially tilted FRC system was chosen as the initial state for full MHD-beam simulations. Since the background plasma was treated as a fluid the model neglected most of the kinetic effects that may be present in a real FRC.

In our model, 3-D nonlinear hybrid techniques are applied to simulate low frequency plasma phenomena in a realistic cylindrical geometry. FLAME follows an evolutionary series of codes previously employed at Cornell [9, 15, 17–19]. It assumes the presence of a neutral gas, a preformed background plasma, and multiple energetic ion components. In our present approach we ignore the beam-induced, fast transient processes resulting in partial ionization of the neutral gas, and do not simulate the neutral motion.

Currently all ion species are treated as finite-size (macro) particles experiencing slowing-down collisions (see further). Although particles representing background ions introduce a noise level higher than that which exists in conventional MHD simulations, it significantly increases the model flexibility by accounting for large orbit particle dynamics near magnetic null points. The plasma electrons are treated as an inertialess fluid through a generalized Ohm's law. Since we wish to avoid the small scales associated with fast electron oscillations, quasineutrality is assumed throughout the code and displacement current is totally excluded from electrodynamic field equations.

The structure of this paper is as follows. The FLAME

computational model is presented in Section 2. Its subsections describe governing equations (for fields, ions, and electrons), dimensionless variables, and a discretization scheme adopted in the code. Section 3 presents the algorithm used to solve the numerical system, and Section 4 briefly outlines the FLAME programming structure and performance (a more complete discussion of these issues will be the subject of a separate paper). Section 5 is devoted to the presentation of the results of a series of tests based on the normal mode solutions obtained for the background plasma. Finally, in Section 6 we discuss the application of FLAME to the problem of ion ring formation in an applied magnetic field perturbed in the azimuthal direction.

II. COMPUTATIONAL MODEL

A. *Fields*

FLAME is a hybrid, PIC code designed to simulate plasma phenomena in the 3-D, cylindrical (r, θ, z) geometry. It follows low-frequency, quasi-neutral plasma motions described by the Maxwell equations with displacement current totally neglected,

$$\nabla \times \mathbf{B} = \frac{4\pi}{c}(\mathbf{j}_e + \mathbf{j}_i), \tag{1}$$

$$\frac{\partial \mathbf{B}}{\partial t} = -c\nabla \times \mathbf{E}, \tag{2}$$

where $\mathbf{j}_e$, $\mathbf{j}_i$ are the electron and total ion current densities.

Ion species are represented by macro-particles and the electrons are modeled as a cold massless fluid obeying Ohm's law:

$$\mathbf{E} = \frac{1}{\sigma}\mathbf{j}_e + \frac{\mathbf{j}_e \times \mathbf{B}}{\rho_e c}, \quad \sigma = \frac{\omega_{pe}^2}{4\pi(\nu_{ei} + \nu_{en})}. \tag{3}$$

In (3) ω_{pe} and ρ_e are the electron plasma frequency and charge density; ν_{ei} and ν_{en} are the characteristic frequencies of elastic electron collisions with background ions and neutrals [20], respectively.

The electron charge density is at all times assumed to equal the total ion charge density, $\rho_e = \sum_k \rho_i^k$ (quasineutrality condition). This assumption holds if $\nabla \cdot \mathbf{E}/4\pi\rho_e \ll 1$. This ratio in our simulations does not exceed 10^{-4}.

The FLAME field solver computes all components of **B** and **E** directly via Faraday's (2) and Ohm's (3) laws. We use a finite-difference representation to discretize spatial operators in the poloidal plane (r, z) and exploit the periodic nature of the azimuthal (θ) direction by representing all physical quantities as finite Fourier series, $\sim\exp(il\theta)$ (l is the azimuthal mode number). A Fast Fourier Transform (FFT) algorithm is used to switch between the cylindrical Fourier mode and configuration spaces.

The magnetic field **B** components are defined on a staggered poloidal grid mesh (commonly known as the Yee lattice) to ensure the conservative form of a discrete divergence analog, $\nabla \cdot \mathbf{B}$ (Fig. 3). The electric field, **E** in our model, is not advanced in time as it is normally done in fully electromagnetic codes, but instead is found from its relation to the electron density and current (3). We have found that it is imperative that this relation should be evaluated locally; i.e., the electric field should be computed at the same grid nodes as the electron quantities for significant numerical errors can result from nonlocal operations performed in spatial regions with large plasma density gradients. At the same time the **E**-field needed to advance the magnetic field has to be known at conventional Yee's locations. We solve this problem by introducing two computational vector fields corresponding to the physical electric field. The *primary* field, **E**, is computed via Ohm's law and defined along with the quantities describing plasma response (the ion and electron charge, ρ_e, ρ_i, and current, $\mathbf{j}_e$, $\mathbf{j}_i$ densities) on the Yee E_θ-grid (Fig. 3, grid 1). The (r, θ, z) components of the *secondary* electric field (needed to obtain a discrete analog of $\nabla \times \mathbf{E}$) are constructed by a central spatial averaging of the corresponding primary **E**-field components and defined together with a discrete curl operator, $\nabla \times \mathbf{B}$ at Yee's conventional locations (Fig. 3, grids 2, 1, 4). Thus, the resulting spatial finite-difference operators are accurate to the second order of mesh spacing, conserve $\nabla \cdot \mathbf{B}$ identically (with computer arithmetics precision) and provide an explicit algorithm for computing the self-consistent, low-frequency electromagnetic field.

The normalization of physical variables in the code is achieved by choosing the following scaling units: $\{\bar{r}, \bar{z}\} = \Delta r$ (poloidal coordinates), $\bar{\theta} = \Delta\theta$ (azimuthal angle), $\bar{v} = c$ (velocity), $\bar{t} = \Delta r/c$ (time), $\{\bar{\nu}, \bar{\omega}\} = 1/\bar{t}$ (frequency), $\{\bar{E}, \bar{B}\} = B_0$ (electric and magnetic fields), $\{\bar{\rho}, \bar{\mathbf{j}}\} = \{en_0, en_0 c\}$ (charge and current densities). Here Δr and $\Delta\theta$ are the radial and azimuthal grid steps; n_0 and B_0 are the characteristic values of the electron number density and the applied magnetic field, respectively.

Thus, the dimensionless equations (1), (2) are expanded in cylindrical coordinates as follows:

$$\frac{il}{r}B_z^l - \alpha\frac{\partial B_\theta^l}{\partial z} = c_m(j_{er}^l + j_{ir}^l),$$

$$\alpha\frac{\partial B_r^l}{\partial z} - \frac{\partial B_z^l}{\partial r} = c_m(j_{e\theta}^l + j_{i\theta}^l),$$

$$\frac{1}{r}\frac{\partial}{\partial r}(rB_\theta^l) - \frac{il}{r}B_r^l = c_m(j_{ez}^l + j_{iz}^l); \tag{4}$$

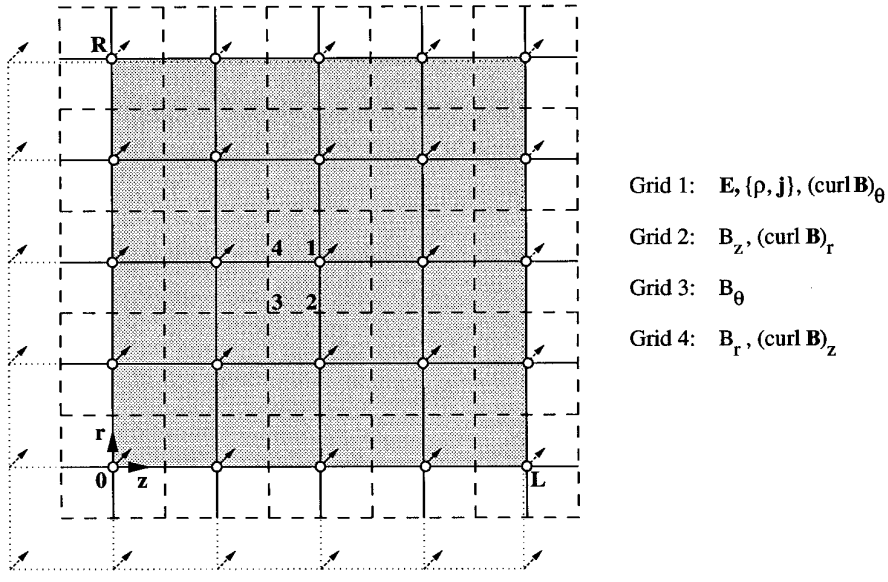

FIG. 3. The staggered 4-grid structure used in FLAME to define displayed computational quantities. Grid numbers show the corresponding locations of grid nodes, arrows indicate the reference points used to sort particles into grid-attached linked lists.

$$\frac{\partial B_r^l}{\partial t} = \alpha \frac{\partial E_\theta^l}{\partial z} - \frac{il}{r} E_z^l,$$

$$\frac{\partial B_\theta^l}{\partial t} = \frac{\partial E_z^l}{\partial r} - \alpha \frac{\partial E_r^l}{\partial z},$$

$$\frac{\partial B_z^l}{\partial t} = \frac{il}{r} E_r^l - \frac{1}{r} \frac{\partial}{\partial r} (r E_\theta^l). \tag{5}$$

Above we have introduced the poloidal grid aspect ratio, $\alpha = \Delta r/\Delta z$ and the dimensionless constant, $c_m = B_0/4\pi e n_0 \bar{r}$. Since the Ohm law is represented by a nonlinear relation, we numerically evaluate it in the configuration space,

$$\mathbf{E} = \eta \mathbf{j}_e + \frac{\mathbf{j}_e \times \mathbf{B}}{\rho_e}, \quad \eta = \frac{\nu_e/\Omega_{ce}}{\rho_e}. \tag{6}$$

where Ω_{ce} is the electron cyclotron frequency.

B. *Particles*

Macro-particles are used to compute the ion charge and current densities, ρ_i, $\mathbf{j}_i$ in the r.h.s.'s of Eqs. (4)–(6). Each simulation particle carries a charge, Q_p, and has an individual position, $\mathbf{r}_p$, and velocity, $\mathbf{v}_p$. The particles of a given ion species, k, are assigned a realistic mass-to-charge ratio, q_k/m_k. In FLAME macro-particles representing the same species are allowed to consist of different numbers of real particles. As a result they can possess variable numerical weights, w_p (normalized charges, Q_p). We find this technique particularly flexible and helpful when initializing a background plasma distribution in a realistic, uniformly discretized cylindrical space. Varying numerical weights of macro-particles in accordance with their radial positions results in a significant reduction of the total number of particles involved in the simulation. We have experimented with different weight settings and have not observed any meaningful changes in the basic simulation results.

Macro-particles are advanced in space and time with the nonrelativistic dimensionless Newton equations:

$$\frac{d\mathbf{v}_p}{dt} = q_p(\mathbf{E}_p + \mathbf{v}_p \times \mathbf{B}_p) - \nu_p \mathbf{v}_p, \quad \frac{d\mathbf{r}_p}{dt} = \mathbf{v}_p. \tag{7}$$

In (7) $q_p = (\bar{r}/r_c)(Z_k/M_k)$ is the dimensionless particle "field charge" ($r_c = m_p c^2/eB_0$, m_p is the proton mass; $Z_k = q_k/e$, $M_k = m_k/m_p$ are the k-species charge and mass numbers); $\mathbf{E}_p$ and $\mathbf{B}_p$ are the values of the electromagnetic field, $\mathbf{F} = \{\mathbf{E}, \mathbf{B}\}$ calculated at a particle position; ν_p is the normalized slowing-down collision frequency representing energy losses by ion components due to their collisions with free and bound electrons [21] (beam ions) and neutral (hydrogen) atoms [22] (background ions).

The coupling of the PIC algorithm with the finite-difference set of electrodynamic equations (4)–(6) is done via standard scatter–gather operations that transform charge and field quantities between the particles and the grid using a tri-linear cell volume weighting technique (Fig. 4). These procedures are applied by looping through grid-attached lists of sorted particles (Fig. 3) so that all particles in a given list have the same selection of neighboring grid nodes. The advantages and disadvantages of this implementation are

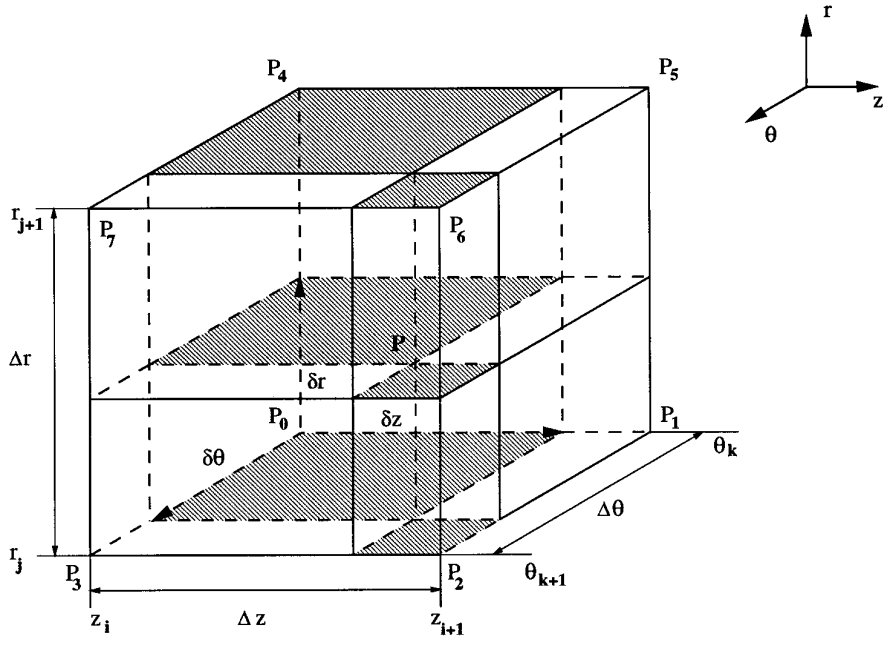

FIG. 4. The 3-D cell volume, tri-linear, particle interpolation scheme. $P(\delta r, \delta\theta, \delta z)$ is the relative particle position inside the reference grid cell, calculated with respect to the base node, P_0. The interpolation volume, δV_n^p, for a given cell vertex, P_n ($n = 0, ..., 7$), is found by calculating the volume of the cell fraction located opposite P_n with respect to P.

discussed further in the text. The scatter step involves distributing a particle's charge and current onto the grid, which is done by assigning their respective fractions to the nearest grid nodes proportionally to the ratio of the corresponding interpolation volume, δV_n^p, to the reference cell volume, V_{cell} (Fig. 4). The charge and current densities at a particular grid node, n are then found by summing up all contributions to it and dividing them by the cell volume, V_n, formed around this node:

$$\{\rho, \mathbf{j}\}_n = \frac{1}{V_n} \sum_p \beta_n^p w_p \{1, \mathbf{v}_p\}, \quad \beta_n^p = \frac{\delta V_n^p}{V_{\text{cell}}}. \tag{8}$$

Similarly, the gather operation interpolates the electromagnetic field from the nearest grid nodes to a particle position:

$$\mathbf{F}_{\text{p}} = \sum_n \beta_n^p \mathbf{F}_n. \tag{9}$$

The final dimensionless formulas (using the same notation as in Fig. 4) are

$$V_n = \begin{cases} r_j, & \text{if } r_j \neq 0 \\ \frac{1}{2} r_{1/2}^2, & \text{if } r_j = 0 \end{cases}$$

$$\beta_0 = \mu_u(1 - \delta r)(1 - \delta z)(1 - \delta\theta),$$

$$\beta_1 = \mu_u(1 - \delta r)\, \delta z (1 - \delta\theta),$$

$$\beta_2 = \mu_u(1 - \delta r)\, \delta z\, \delta\theta, \quad \beta_3 = \mu_u(1 - \delta r)(1 - \delta z)\, \delta\theta,$$

$$\beta_4 = \mu_l\, \delta r(1 - \delta z)(1 - \delta\theta), \quad \beta_5 = \mu_l\, \delta r\, \delta z(1 - \delta\theta),$$

$$\beta_6 = \mu_l\, \delta r\, \delta z\, \delta\theta, \quad \beta_7 = \mu_l\, \delta r(1 - \delta z)\, \delta\theta,$$

where

$$\bar{r}_j = \tfrac{1}{2}(r_j + r_{j+1}), \quad \mu_u = \frac{(r_j + r_{j+1} + \delta r)}{2\bar{r}_j}, \quad \mu_l = \frac{(2r_j + \delta r)}{2\bar{r}_j}.$$

C. *Boundary Conditions*

Boundary conditions for fields **E**, **B** on axis are satisfied in the l-mode space by taking advantage of the known asymptotic behavior of each particular mode. The electric field, **E**, and the perpendicular component of the magnetic field, **B**, are generally assumed to vanish at all conducting surfaces. The shorting out of the normal component of **E** can be validated either through the assumption of Child–Langmuir (space-charge limited) electron emission at the plasma–metal interface due to an ''infinite'' electron supply by the conducting wall at high magnitudes of the electric field that could result under experimental conditions, or by assuming infinite permittivity of the wall material, ε. The tangential component of the magnetic field then suffers a

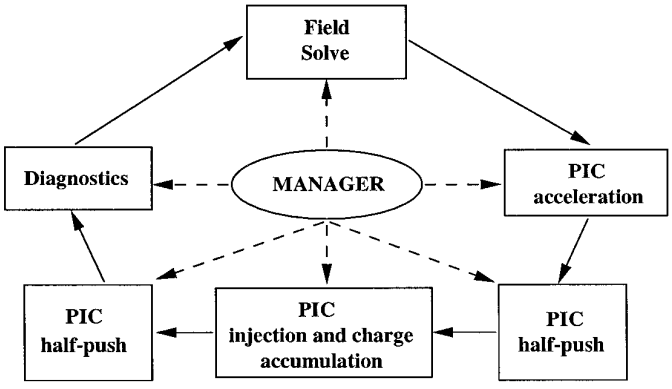

FIG. 5. FLAME algorithmic flow (parallel communication acts are omitted from the scheme).

discontinuity across plasma boundaries (it vanishes inside conductors). Particle boundary conditions absorb (beam species) or reflect (background ions) particles at all rigid boundaries.

Normal mode tests are carried out with appropriate changes made to the above boundary settings to ensure the correct solution behavior (e.g., the periodic boundary conditions are applied in the axial direction).

III. NUMERICAL ALGORITHM

The FLAME algorithmic flow is shown in Fig. 5. Below we outline a single timestep Δt advance of the field and particles quantities. The necessary FFT steps are straightforward in the context of the algorithm and omitted from the scheme. The standard leapfrog scheme is used for particle advance.

We start the time advance with the known quantities, $\mathbf{E}_l^{n-1}, \mathbf{B}_l^{n-1}, \rho_i^{n-1/2}, \mathbf{j}_i^{n-1/2}, \mathbf{v}_\mathrm{p}^{n-1/2}, \mathbf{r}_\mathrm{p}^n$ (n is indicative of a timestep number). The electron charge density is found from the quasineutrality condition, $\rho_\mathrm{e} = \rho_\mathrm{c} + \rho_\mathrm{i}$ (ρ_c is a cutoff value we introduced to ensure iteration convergence in low plasma density regions). To initialize the iteration we use $\mathbf{B}_l^{n-1/2} = \mathbf{B}_l^{n-1}$. The algorithm (in the normalized variables) then proceeds as follows.

1. *Field Solve.* Finite-difference solutions of Eqs. (4)–(6) are found:

$$\mathbf{j}_{\mathrm{e}l}^{n-1/2} = -\mathbf{j}_{\mathrm{i}l}^{n-1/2} + c_\mathrm{m}\nabla \times \mathbf{B}_l^{n-1/2},$$

$$\mathbf{E}^{n-1/2} = \eta \mathbf{j}_\mathrm{e}^{n-1/2} + \frac{\mathbf{j}_\mathrm{e}^{n-1/2} \times \mathbf{B}^{n-1/2}}{\rho_\mathrm{e}^{n-1/2}},$$

$$\mathbf{B}_l^n = \mathbf{B}_l^{n-1} - \Delta t \nabla \times \mathbf{E}_l^{n-1/2}, \quad \mathbf{B}_l^{n-1/2} = \frac{1}{2}(\mathbf{B}_l^n + \mathbf{B}_l^{n-1}).$$

Field solve procedures are repeated iteratively until all three components of $\mathbf{B}$ have converged to within a specified relative tolerance, $\varepsilon \simeq 10^{-4}$.

2. *Electric Field Prediction.* After the convergence is achieved the electric field is predicted at integral timesteps by

$$\mathbf{E}_l^n = 2\mathbf{E}_l^{n-1/2} - \mathbf{E}_l^{n-1}.$$

3. *Particle Force Gather.* Electromagnetic field is interpolated from the grid nodes to the particle positions:

$$\{\mathbf{E}, \mathbf{B}\}^n \to \{\mathbf{E}_\mathrm{p}, \mathbf{B}_\mathrm{p}\}^n.$$

4. *Particle Acceleration.*

$$\mathbf{v}_\mathrm{p}^{n+1/2} = \mathbf{v}_\mathrm{p}^{n-1/2} + \mathbf{q}_\mathrm{p}\,\Delta t(\mathbf{E}_\mathrm{p}^n + \mathbf{v}_\mathrm{p}^n \times \mathbf{B}_\mathrm{p}^n) - \nu_\mathrm{p}\,\Delta t \mathbf{v}_\mathrm{p}^n,$$

$$\mathbf{v}_\mathrm{p}^n = \tfrac{1}{2}(\mathbf{v}_\mathrm{p}^{n-1/2} + \mathbf{v}_\mathrm{p}^{n+1/2}).$$

5. *First Particle Half-Push.*

$$\mathbf{r}_\mathrm{p}^{n+1/2} = \mathbf{r}_\mathrm{p}^n + \frac{\Delta t}{2}\mathbf{v}_\mathrm{p}^{n+1/2}.$$

6. *Particle Injection and Charge Scatter.*

$$\{q_\mathrm{p}, \mathbf{r}_\mathrm{p}^{n+1/2}, \mathbf{v}_\mathrm{p}^{n+1/2}\} \to \{\rho_i, \mathbf{j}_i\}^{n+1/2}.$$

7. *Second Particle Half-Push.*

$$\mathbf{r}_\mathrm{p}^{n+1} = \mathbf{r}_\mathrm{p}^{n+1/2} + \frac{\Delta t}{2}\mathbf{v}_\mathrm{p}^{n+1/2}.$$

8. *Electric Field Correction* (*Optional*). The ion charge is interpolated in time

$$\{\rho_i, \mathbf{j}_i\}^n = \tfrac{1}{2}[\{\rho_i, \mathbf{j}_i\}^{n-1/2} + \{\rho_i, \mathbf{j}_i\}^{n+1/2}]$$

and the electric field, $\mathbf{E}$ is corrected via the Faraday and Ohm laws, using the ion charge, $\{\rho_i, \mathbf{j}_i\}^n$ and the magnetic field, $\mathbf{B}^n$.

Alternatively, one may start the time advance with the electric field $\mathbf{E}^{n-3/2}$. Then step 2 (the electric field prediction) should be replaced by

$$\mathbf{E}_l^n = \tfrac{3}{2}\mathbf{E}_l^{n-1/2} - \tfrac{1}{2}\mathbf{E}_l^{n-3/2}.$$

Below we present a few important comments on the actual implementation of the above algorithm in FLAME.

1. The iterative field solve (step 1) and the electric field correction at integral timesteps (step 8) are done independently of the particle advance and charge deposition steps

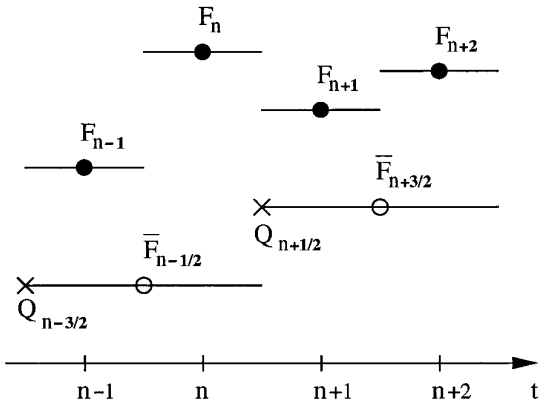

FIG. 6. The subcycling time integration sequence: circles and crosses mark the model times at which the charge, Q, and field, F, quantities are computed at given timesteps (in this example the particle advance period is twice as long as the field solve timestep).

3–7. Though this approach suffers a certain loss of accuracy compared to the fully iterative predictor–corrector schemes [19, 23], it results in a substantial reduction of code running time. The overall simulation accuracy can be easily controlled by a judicious choice of the timestep, Δt. We have found that this technique is well justified for practical applications so far considered.

2. The particle algorithm includes "field subcycling," which enables advancing the particles an integral number of timesteps used to advance the fields on the grid. Schematically this technique is described in Fig. 6. The actual length of particle push cycle allowed depends on the dynamic characteristics of a particular species. The particles are accelerated by the "slow" electromagnetic fields obtained by averaging the "fast" fields supplied by the field solver on each timestep. The current implementation of this algorithm reduces resolution of the time integration but results in a significant gain of CPU time. As in the above case the numerical accuracy can be controlled through an appropriate trade-off between the timestep value and the lengths of particle cycles.

3. The particle pushes (step 5 and step 7) are performed in a local (particle) Cartesian frame of reference, followed by a usual transformation of particle positions and velocities back to the cylindrical coordinates. We have taken care not to calculate the standard library **sqrt** and trigonometric functions for most instances, thus saving a significant percentage of particle push time.

IV. PROGRAMMING

Below we briefly discuss the FLAME programming structure and capabilities. The detailed account of these issues will be given elsewhere.

A. *Object-Oriented PIC (OOPIC) Coding*

The number of computational physics applications exploiting the object-oriented programming (OOP) style is constantly growing due to significant benefits it presents to the software developer when the code logic becomes complicated. This is particularly true when coding parallel PIC and Monte Carlo applications [24]. The resulting OOPIC architecture provides an opportunity for increasing code reliability, flexibility and extensibility [25]. The object-oriented methodology is employed in going from a mathematical Newton–Maxwell model to an approximate discrete formulation. The object-oriented paradigm enables the programmer to develop a code providing a natural discrete representation (cast) of physical objects in question. FLAME has been designed along basic guidelines of the OOP technology and incorporates a variety of classes characterizing particle and grid data as well as physical and programming (serial and parallel) operations-methods performed on them.

B. *Serial Performance*

We have conducted numerous tests with various profiling tools available on the IBM SP2 system of Cornell Theory Center to determine the serial (single node) performance of FLAME critical routines. We find that the field solve has the CPU performance of ~13–16 μs/grid node/iteration (we use the IBM ESSL FFT routines, which operate more efficiently on multiple data sequences, such as vector arrays). The PIC related procedures (particle charge deposition, acceleration, push, sorting) add up to ~18 μs/particle/push. With these benchmarks known we have done a number of runs by largely varying basic system parameters such as grid dimensions, particle numbers, and species advance cycles. Based on the timings observed empirical dependencies have been derived relating the FLAME serial performance to the above parameters with a great degree of accuracy (within 5%). By applying the benchmarking formulas it is possible to estimate the total CPU time required to complete a single serial FLAME production run. Taking into account the minimal parametric configuration dictated by the necessity of accurately resolving the typical physical spatial and temporal scales, we assume a $101(z) \times 41(r) \times 17(\theta)$ mesh, with 150,000 beam and 850,000 background plasma particles (with push periods 2 and 10 timesteps, respectively). Assuming then the total characteristic number of timesteps ~10,000 and that 4 iterations per step are required on average to achieve the convergence with relative tolerance $\sim 10^{-4}$) we arrive at the total of 20–25 IBM SP2 CPU hours. This estimate is in good agreement with actual charges in our account balance.

C. *Parallel Implementation*

FLAME has been designed to run on massively parallel facilities of MIMD type such as the IBM SP2. Its program-

ming structure enables it to run on a SIMD machine as well. The parallel computing is achieved, as usual, by partitioning the global computation domain into rectangular subdomains, assigning grid and particle data to each subdomain, and processing them concurrently [26]. One can see then that the numerical algorithm outlined above has to undergo straightforward modifications in that the time advances of the magnetic field and particle positions as well as the particle charge deposition must be followed by acts of transferring grid and (if necessary) particle data between neighboring subdomains. The electric field is computed locally by each processor.

1. *Partitioning*. The partitioning of the global computation region is done only in the poloidal plane so that FFTs are always local to processors. Domains defined on the staggered grids are partitioned separately into subdomains, which are then mapped onto individual processors. The general algorithm we have developed assigns an arbitrary number of subdomains to any particular processor. The geometrical configuration of the partitioned domain becomes available on every participating processor, but actual data are allocated locally only for domains owned by processors.

2. *Field Communication*. Here we use the term ''fields'' (as opposed to ''particles'') to refer to all spatial grid distributed computational quantities. To transfer them between the neighboring processors (when required by the algorithm) we devised a geometric approach similar in spirit (but not actual implementation) to the philosophy adopted by the developers of the LPARX parallel software [27]. The transfer protocol routine on each processor executes a loop only over those subdomains of a parallel (distributed across the processors) array that are assigned the processor *id* (owned by the processor). Each such domain is geometrically intersected then in another loop with all other subdomains consitituting the global computation domain. The data from the region common to the both intersected domains is copied via local or remote transfer routines. The intersection operation is in most instances precomputed so that the last loop is executed over a set of known ''neighbors'' and common regions. This technique is independent of the actual partitioning method.

3. *Particle Communication*. One of the remarkable features of the algorithm used for communicating grid data across the processors is that it can be extended to transferring particle objects as well. As already mentioned, FLAME organizes particles moving inside their ''home'' subdomain into linked lists ''attached'' to their ''base'' grid nodes, which are determined by taking integer part of particle coordinates with respect to the subdomain origin. After a particle's position has been advanced the sorting routine locates the grid cell the particle has moved to. If it is found to differ from the previous one then the particle is removed from the old list, its position is checked for boundary conditions, and it is added to an appropriate list.

This sorting algorithm has the following implications:

1. Particle charge deposition and acceleration are done by looping through lists of linked particles. This precludes vectorization but at the same time improves data locality by eliminating the necessity to fetch common grid data for each particle. As a result code performance improves on multiprocessor architectures due to a higher rate of cache (fast memory) hits. The efficiency of this mechanism increases with larger numbers of particles per cell.

2. Global particle array sorting is required only at code restarts and when the computation domain has to be repartitioned (e.g., for load balancing reasons). Thus the CPU overhead due to the necessity of maintaining linked lists is reduced to a minimum.

3. To transfer particles between two subdomains one only needs to intersect them in a proper way so that ''bunches'' of particles ''hanging'' on the boundaries are automatically packed into the buffer and sent out to subdomains where they belong. The code tracks the number and location of ''holes'' in particle storage arrays created by outgoing particles, and fills them in by incoming particles. The transfer procedure is looped until all ''ghost'' (migrating) particles have arrived at their destinations. The boundary conditions are applied simply by processing the particles attached to the global domain boundaries.

4. Having particles organized in linked lists will enable us in the future to implement Monte Carlo collision techniques which require operations over ensembles of particles located within a pre-determined spatial range (usually of order of the grid cell).

V. NORMAL MODE TESTS

The correct functioning of the first-order discrete plasma model can be verified by considering waves of known dispersion relation, e.g., normal modes. The normal modes of a cold cylindrical plasma bounded by a metallic wall at $r = R$ can be easily obtained [15] assuming that all first-order quantities vary as $\exp[i(l\theta + k_z z - \omega t)]$ and linearizing Eqs. (1)–(3) supplemented with an ion fluid equation (with resistivity and collisional terms neglected),

$$\omega^2 = v_A^2[R^2 \pm (R^4 - P^4)^{1/2}],$$
$$R^2 = \tfrac{1}{2}[\nu_m^2(1 + \kappa^2) + k_z^2(\kappa^2 + 2)], \quad P^4 = k_z^2(\nu_m^2 + k_z^2), \tag{10}$$

where v_A and Ω are the Alfvén velocity and ion cyclotron frequency, $\kappa = k_z v_A/\Omega$. The expression relating the transverse ''wave number'' ν_m (note that a radial mode $m = 1, 2, \ldots$, corresponds to $m/2$ radial ''wavelengths'' in

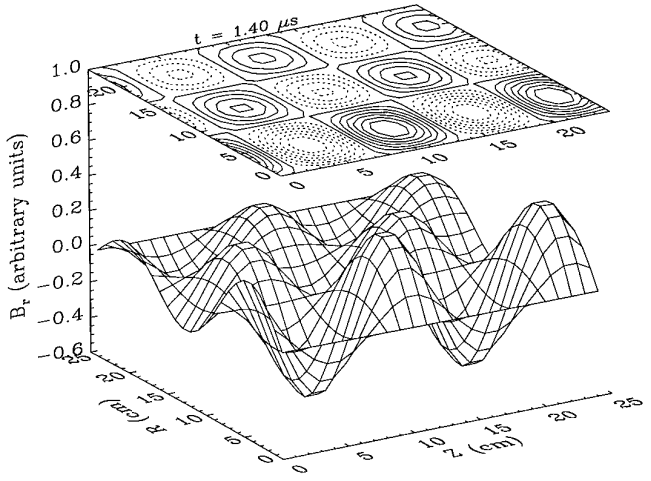

FIG. 7. A snapshot of the spatial mode structure of B_r for an $l = 0$ normal wave test run.

the system) to the longitudinal wave vector component, k_z, and azimuthal mode number, l, has the following form:

$$\frac{\mathrm{J}_{l-1}(\nu_m R)}{\mathrm{J}_l(\nu_m R)} + \frac{l}{\nu_m R}\left\{\frac{(\omega^3/\Omega^3)(\nu_m^2 v_\mathrm{A}^2/\Omega^2)}{[\kappa^2 - (\omega^2/\Omega^2)]^2 - \kappa^4(\omega^2/\Omega^2)} - 1\right\} = 0. \tag{11}$$

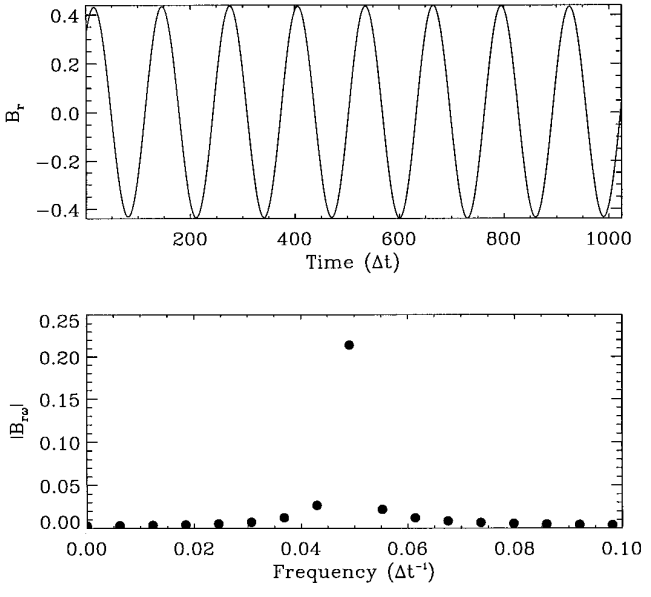

FIG. 8. B_r (in arbitrary units) in diagnostic cell ($l = 0$): (top) time history; (bottom) Fourier analysis.

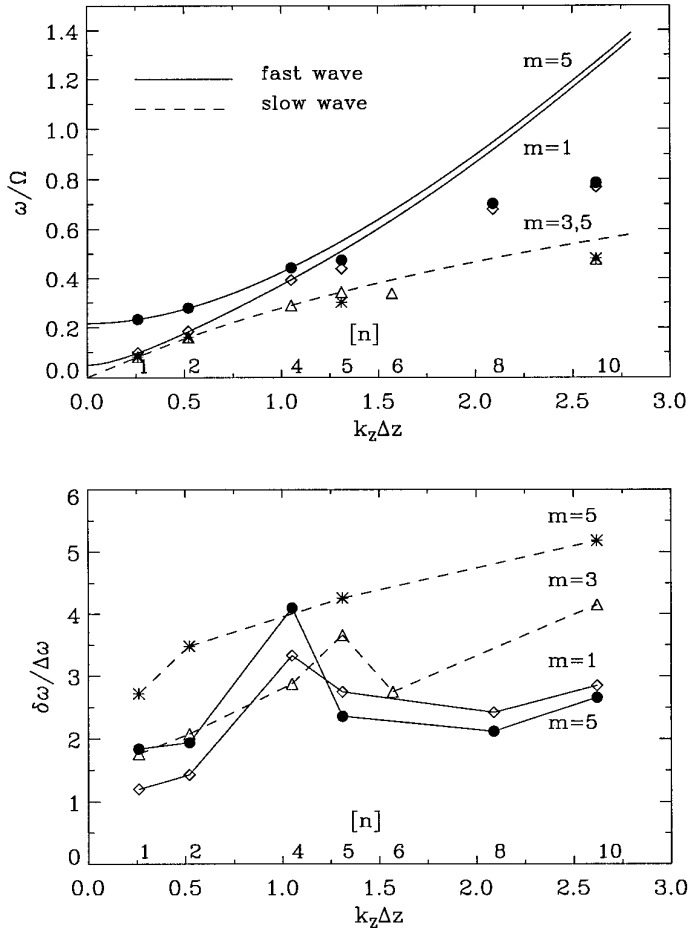

FIG. 9. The results of the $l = 0$ normal mode tests for slow- and fast-waves initialized with given radial mode numbers, m: (top) normalized theoretical and observed wave frequencies as functions of the longitudinal wave number, k_z; (bottom) observed wave bandwidths (divided by the frequency resolution scale).

For purposes of code verification we will consider two cases (i) $l = 0$ and (ii) $l \neq 0$, $k_z = 0$, in which boundary condition (11) reduces to simple forms [15]. The first case checks the accuracy of the finite difference operators in the poloidal (r, z) plane, while the second one verifies the behavior of discrete nonaxisymmetric modes. The periodic boundary conditions are applied in the axial direction. The radial boundary conditions are adjusted in each case in accordance with the behavior of the electric and magnetic fields near the cylindric wall. The typical run duration is 1024 timesteps. In order to separate the grid (finite dispersion) effects from the temporal errors (introduced by choosing a finite timestep Δt) we vary Δt with every run to keep the wave period to timestep ratio constant, $T/\Delta t = 128$. The fields and currents are sampled in time in an arbitrary diagnostic cell. In all test runs we assume a hydrogen plasma ($Z_i = M_i = 1$), the applied magnetic field,

TABLE 1

Summary of $l = 1$, $k_z = 0$ Runs

m	$\nu_m R$	ω/Ω (analytic)	ω/Ω (observed)
1	1.874	0.043	0.043
−1	1.808	−0.042	−0.042
2	5.357	0.132	0.132
−2	5.307	−0.125	−0.125
3	8.560	0.200	0.199
−3	8.510	−0.221	−0.221
12	36.880	0.368	0.369
20	62.025	0.456	0.586

$\mathbf{B}_0 = B_0\hat{z}$, $B_0 = 10$ kG, and equal poloidal grid steps, $\Delta r = \Delta z = 1$ cm.

A. $l = 0$

In this case it follows from Eq. (11) that for a given m the radial wave number, ν_m is found by inverting $J_1(\nu_m R) = 0$. One obtains then the corresponding normal wave frequency, ω from Eq. (10) by specifying k_z and choosing the appropriate sign.

The probe wave is initialized by setting at $t = 0$ the cylindrical components of the electromagnetic field and ion current in accordance with the linearized solution of the collisionless Maxwell and ion motion equations [15]:

$$E_r = |E_r|\cos(k_z z - \omega t)J_1(\nu_m r),$$

$$E_\theta = -|E_\theta|\sin(k_z z - \omega t)J_1(\nu_m r),$$

$$B_z = \frac{c\nu_m}{\omega}|E_\theta|\cos(k_z z - \omega t)J_0(\nu_m r),$$

$$B_r = \frac{ck_z}{\omega}|E_\theta|\sin(k_z z - \omega t)J_1(\nu_m r),$$

$$B_\theta = \frac{ck_z}{\omega}|E_r|\cos(k_z z - \omega t)J_1(\nu_m r),$$

$$j_{ir} = \frac{\omega_p^2}{4\pi}\frac{\omega}{\Omega^2 - \omega^2}\left[-\frac{\Omega}{\omega}|E_\theta| + |E_r|\right]\sin(k_z z - \omega t)J_1(\nu_m r),$$

$$j_{i\theta} = \frac{\omega_p^2}{4\pi}\frac{\omega}{\Omega^2 - \omega^2}\left[-|E_\theta| + \frac{\Omega}{\omega}|E_r|\right]\cos(k_z z - \omega t)J_1(\nu_m r),$$

$$E_z = 0, \quad j_{iz} = 0,$$

$$\frac{|E_r|}{|E_\theta|} = \frac{\omega^3/\Omega^3}{(\omega^2/\Omega^2)[1 + \kappa^2] - \kappa^2}, \tag{12}$$

where $\omega_p = 4\pi e_i^2 n_i/m_i$ is the ion plasma frequency. We use the following parameters: $N_r \times N_z = 25 \times 25$, $n_e = 5.2 \times 10^{14}$ cm^{-3}.

An example of the normal wave shape in space is shown in Fig. 7. The numerical solution exhibits the characteristic harmonic behavior throughout many wave periods (Fig. 8), which is clearly manifest in the monochromatic shape of its Fourier image. The results of axisymmetric ($l = 0$) normal mode runs are summarized in Fig. 9, where typical analytical dispersion curves are plotted for slow ("−") and fast ("+") waves along with observed wave frequencies as functions of $k_z\,\Delta z = 2\pi n/N_z$. For the sufficiently long wavelengths $k_z\,\Delta z \leq 1$ ($n \leq 4$) little numeric dispersion is present, while for $k_z\,\Delta z \geq 1$ the wave frequency is noticeably less than that predicted by analytic (gridless) theory. The finite grid effects also account for the appearance of spectral

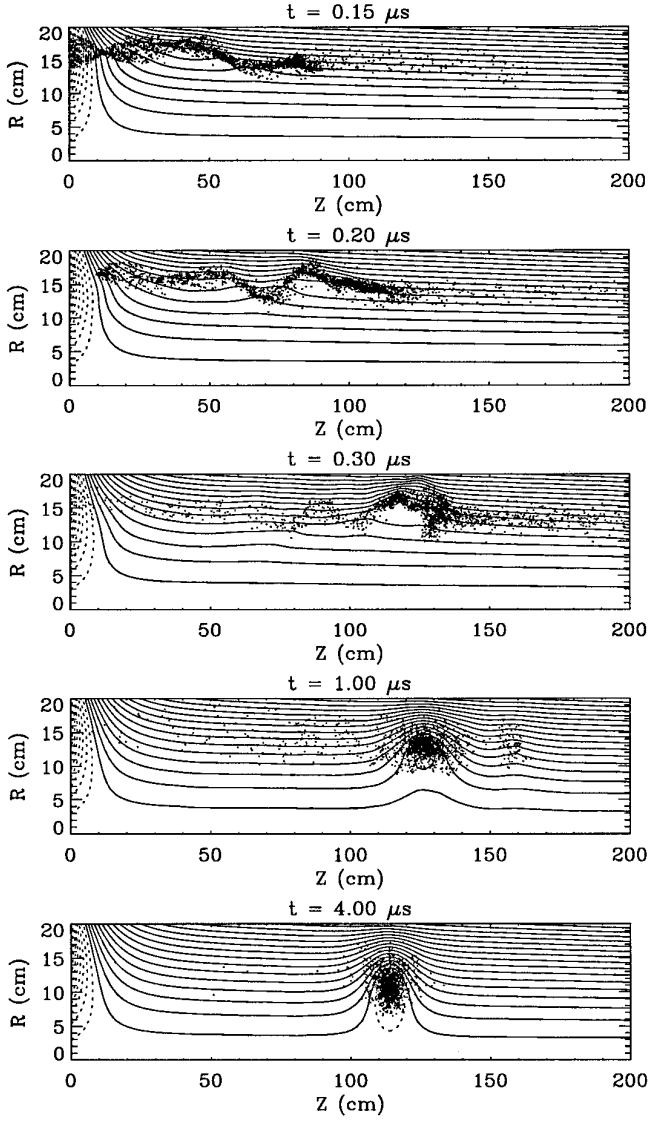

FIG. 10. Axisymmetric FLAME simulation: the formation of an ion ring field-reversed configuration.

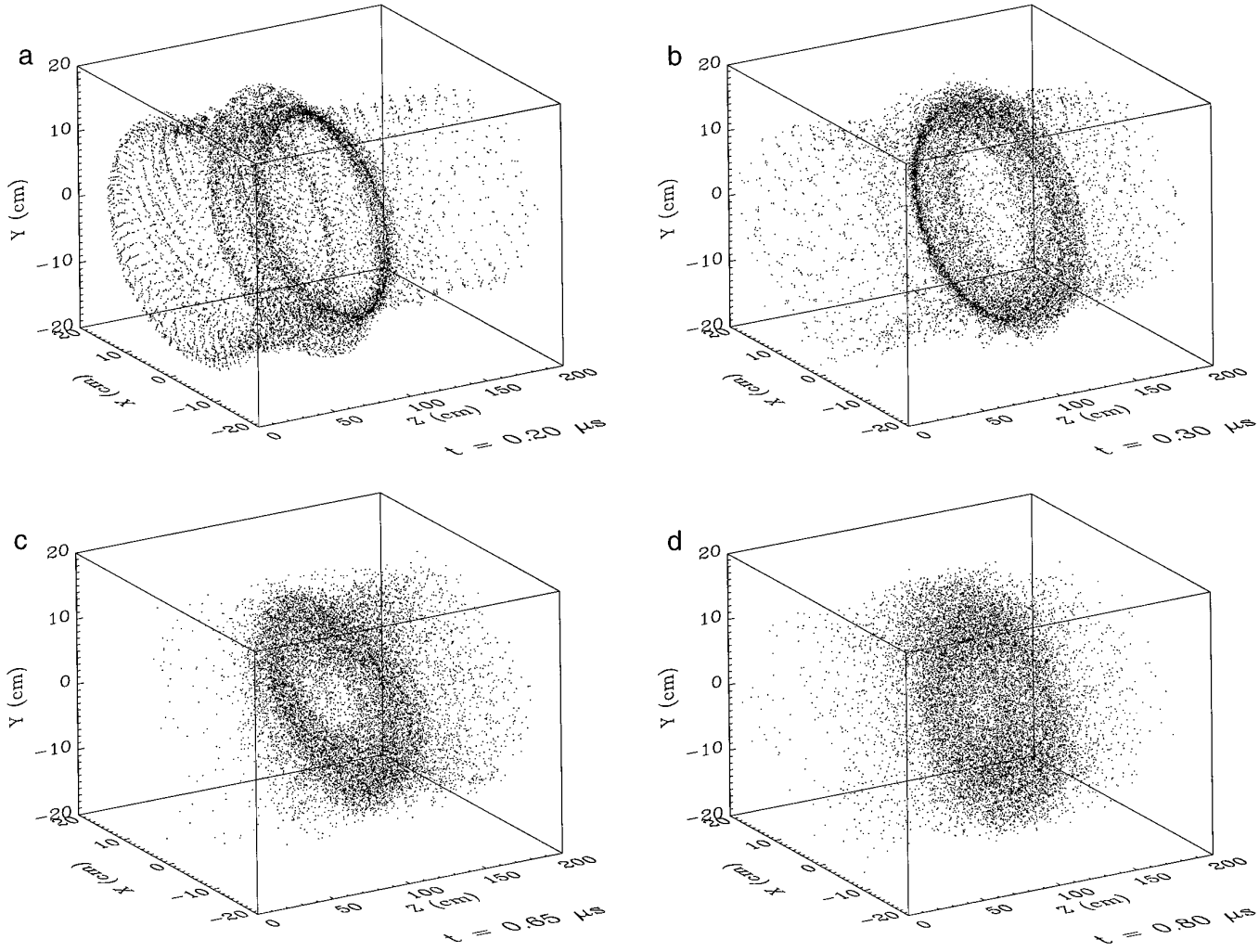

FIG. 11. 3-D Cartesian snapshots of the ion beam evolution: the beam advances into the dipole-modulated ramped solenoidal magnetic field and forms a ring (the injection period is 0.15 μs).

spread in the wave Fourier image (Fig. 9, *bottom*) that accompanies larger radial and longitudinal wave numbers. This effect is well known, and the computation accuracy can be controlled in each particular problem by choosing appropriate grid spacings.

B. $l \neq 0$, $k_z = 0$

In the limit of infinite longitudinal wavelength, Eqs. (10), (11) yield, respectively,

$$\frac{\omega}{\Omega} = \pm \frac{\nu_m c}{\omega_p}, \tag{13}$$

$$\frac{\mathrm{J}_{l-1}(\nu_m R)}{\mathrm{J}_l(\nu_m R)} \pm \frac{lc}{\omega_p R} - \frac{l}{\nu_m R} = 0. \tag{14}$$

The linearized solution (we omit the argument $\nu_m R$ of the Bessel functions) in this case is

$$B_z = \delta B(\cos\omega t \cos l\theta + \sin\omega t \sin l\theta)\mathrm{J}_l,$$

$$B_r = B_\theta = 0,$$

$$E_r = \frac{\delta B}{4\pi\rho_i}\left[-\frac{\Omega}{\omega}\frac{l\mathrm{J}_l}{r} - \nu_m \mathrm{J}_l'\right](\cos\omega t \cos l\theta + \sin\omega t \sin l\theta),$$

$$E_\theta = \frac{\delta B}{4\pi\rho_i}\left[\frac{\Omega}{\omega}\nu_m \mathrm{J}_l' + \frac{l\mathrm{J}_l}{r}\right](\cos\omega t \sin l\theta - \sin\omega t \cos l\theta),$$

$$E_z = 0,$$

$$j_{ir} = -\frac{\nu_m}{4\pi\rho_i}\frac{\delta B}{\omega}\Omega(\sin\omega t \cos l\theta - \cos\omega t \sin l\theta)\mathrm{J}_l',$$

$$j_{i\theta} = \frac{\delta B l}{4\pi\rho_i r}\frac{\Omega}{\omega}(\cos\omega t \cos l\theta + \sin\omega t \sin l\theta)\mathrm{J}_l,$$

$$j_{iz} = 0. \tag{15}$$

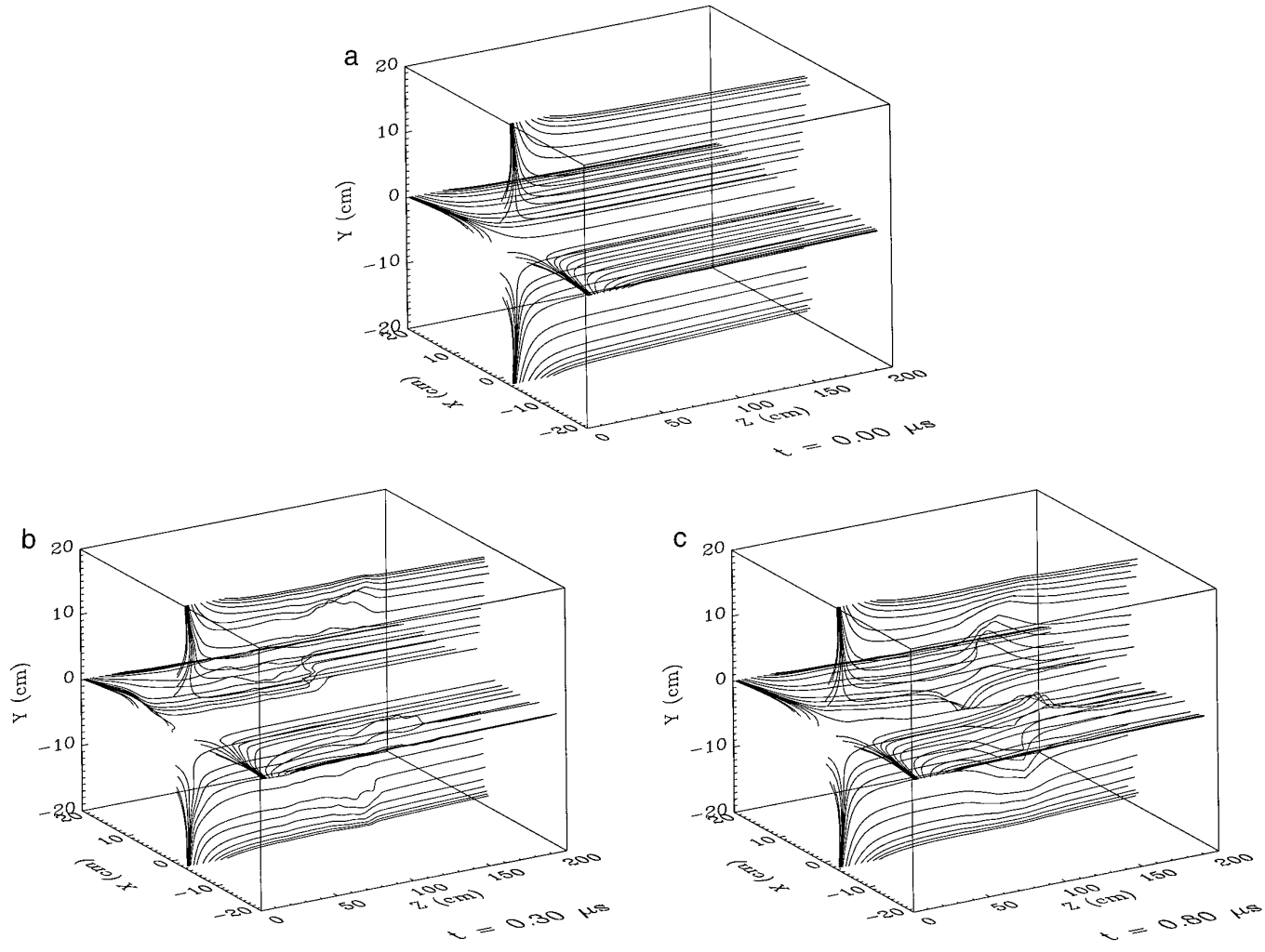

FIG. 12. 3-D Cartesian snapshots of the total magnetic field: field lines originate at the poloidal planes, $\theta = 0, \pi/2, \pi, 3\pi/2$, and are distorted by the transitory azimuthal component of the beam generated field.

We have performed a series of $l = 1$ runs on a $3(z) \times 25(r) \times 8(\theta)$ mesh. The analytical frequencies were obtained by specifying $\nu_m R$, then solving Eq. (14) for the ion density, and finally choosing the sign in Eq. (10) and computing the corresponding value of ω. The results of these tests are described in Table 1 and indicate good azimuthal properties of the code. In fact, in all but one of the considered cases the field and current oscillate in time with strictly monochromatic frequencies (the spectral widths of their Fourier images are close to the frequency resolution scale). The only exception is the $m = 20$ wave. This oscillation is not monochromatic (its spectral width exceeds the frequency resolution spacing by three times and its numerical frequency is appreciably different from the analytical value), which is well understood since the radial ''wavelength'' of this mode is close to the minimal one resolved on the chosen grid.

VI. 3-D ION RING FORMATION

Before proceeding with essentially 3-D FLAME production runs we repeated the axisymmetric simulations performed earlier with the FIRE code and found an excellent match between the new and old results (compare Fig. 10 with Figs. 8(a), 10(a) in Ref. [9]).

The first practical problem requiring the application of such a major computational tool as the FLAME code was the investigation of the 3-D stability of an ion ring. In particular, we are interested in whether any departures from strict axisymmetry would be deleterious for ring formation. An example of a loss of axisymmetry is the presence of azimuthal fluctuations in the beam density and current caused by static aberrations of the ramped applied magnetic field, $l \neq 0$. Our objective was to find out if these perturbations may amplify to such an extent that they have

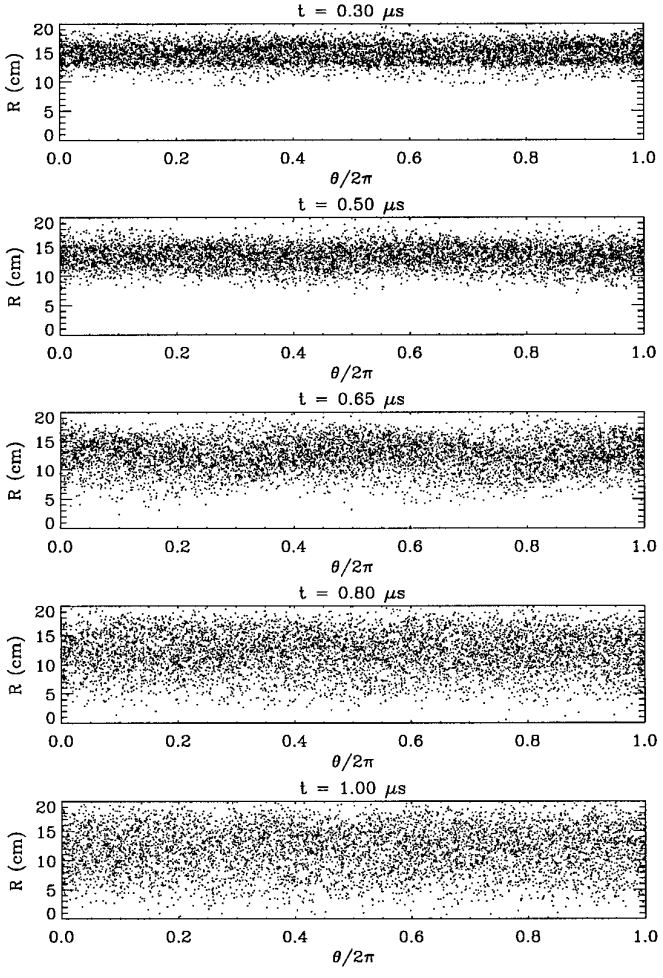

FIG. 13. Ion ring evolution in the (r, θ) phase space: the beam exhibits a harmonic modulation at $l = 2$ mode, which disappears due to the large-orbit particle phase mixing.

a disruptive effect on a ring trapping and the formation of a field-reversed configuration.

The grid resolution in this study was $101 \times 41 \times 23$ ($l_{\max} = 5$ modes were retained in the field equations) and the numbers of beam and background particles, ~210,000 and 1,200,000, respectively. The simulation was run in parallel on a network of 6–10 IBM SP2 processors (the actual number of processors varied throughout the run). The fields were advanced each 0.1 nsec while the beam and background particles were updated each 0.2 nsec and 1 nsec, respectively. The total energy was conserved within a few percent of the energy injected into the system.

The preliminary results hint the good azimuthal stability properties of the rings previously observed in the axisymmetric ($2\frac{1}{2}$-D) simulations. In particular, we have studied the ''quiet'' injection of an unperturbed ion beam into a ramped applied magnetic field distorted by static dipole ($l = 2$) radial and azimuthal harmonics of the order of several percent of the radial component of the axisymmetric solenoidal field at the characteristic beam injection radius. We have found that this does not significantly interfere with the beam trapping and ring formation mechanisms previously studied under the assumption of axisymmetric evolution of the beam–plasma system. The beam is trapped by the self-field on the same spatial and temporal scales (Fig. 11) previously derived from the axisymmetric studies. We have also discovered that the maximum azimuthal perturbations of the ring-generated magnetic field (Fig. 12) are of the order of 20% of the averaged self-field strength at the ring location after 1 μsec. However, the self-field that diffuses to the axis remains much more uniform throughout the simulation. The temporal evolution of the maximum field-reversal factor follows very closely the time history during the axisymmetric formation of the ring (obtained in $2\frac{1}{2}$-D simulations), and even speeds up thereafter due to the closer proximity of the ring to the axis.

The 3-D simulation also reveals that the beam modulation at $l = 2$ which develops during the ring formation and disappears later on due to the phase mixing of the large-orbit beam particles (Fig. 13), results in an increase in the radial beam temperature with a concomitant increase in the ring radial thickness.

VII. CONCLUSION

The results from the normal mode and 3-D ion beam simulations clearly demonstrate the general robustness of the developed algorithm and the ability of FLAME to model complex plasma phenomena in three dimensions.

An important consideration yet to be taken into account is the long-time stability of the ion rings to low-frequency perturbations, particularly the tilt and precession, $l = 1$ modes. The studies of the low-frequency stability of field-reversed rings in a dense uniform plasma background have been earlier carried out using a linearized 3-D hybrid code, RINGHYBRID [15, 28]. We intend to further explore these important issues through a thorough investigation of the self-consistent, nonlinear behavior of a stationary, strong field-reversed ring with respect to these modes. The results of our research are expected to provide further insight into the general theory of equilibrium and stability of field-reversed configurations.

ACKNOWLEDGMENTS

This work was supported by U.S. DOE Contract DE-FG02-93ER54221 and was done using the IBM SP2 system of the Cornell Theory Center. The preparation of graphics materials was assisted by A. V. Omelchenko.

REFERENCES

1. M. Tuszewski, *Nucl. Fusion* **28,** 2033 (1988).
2. J. A. Fillips, D. A. Baker, and R. F. Gribble, *Nucl. Fusion* **35** (1995).

3. A. L. Hoffman *et al*., *Fusion Technol.* **23,** 185 (1993).
4. H. Momota *et al*., *Fusion Technol.* **21,** 2307 (1992).
5. Y. Ono *et al*., *Trans. Fusion Technol.* **27,** 369 (1995).
6. R. N. Sudan and E. Ott, *Phys. Rev. Lett.* **33,** 355 (1974).
7. J. Finn and R. N. Sudan, *Nucl. Fusion* **22,** 1443 (1982).
8. R. N. Sudan, Particle ring fusion, in *Unconventional Approaches to Fusion*, edited by B. Brunelli and G. G. Geotta (Plenum, New York, 1982), p. 311.
9. Yu. A. Omelchenko and R. N. Sudan, *Phys. Plasmas* **2,** 2773 (1995).
10. C. B. Ruchti and R. V. Lovelace, *Phys. Fluids* **27,** 1789 (1984).
11. R. N. Sudan and P. K. Kaw, *Phys. Rev. Lett.* **47,** 575 (1981).
12. C. Litwin, R. N. Sudan, and A. D. Turnbull, *Phys. Fluids* **27,** 2791 (1984).
13. A. D. Turnbull and R. N. Sudan, *Phys. Fluids* **29,** 1923 (1986).
14. C. Litwin and R. N. Sudan, *Phys. Fluids* **31,** 423 (1988).
15. A. Fiedman, R. N. Sudan, and J. Denavit, *J. Comput. Phys.* **40,** 1 (1981).
16. D. C. Barnes and R. D. Milroy, *Phys. Fluids B* **3,** 2609 (1991).
17. A. Friedman, R. L. Ferch, R. N. Sudan, and A. T. Drobot, *Plasma Phys.* **19,** 1101 (1977).
18. A. Mankofsky, A. Friedman, and R. N. Sudan, *Plasma Phys.* **23,** 521 (1981).
19. A. Mankofsky, R. N. Sudan, and J. Denavit, *J. Comput. Phys.* **70,** 89 (1987).
20. H. S. W. Massey and E. H. S. Burhop, in *Electronic and Ionic Impact Phenomena* (Oxford Univ. Press, Oxford, 1952).
21. F. C. Young *et al*., *Phys. Rev. Lett*. **49,** 549 (1982).
22. D. J. Rose and C. Melville, in *Plasmas and Controlled Fusion* (MIT Press, Cambridge, MA, 1961), p. 32.
23. D. S. Harned, *J. Comput. Phys.* **47,** 452 (1982).
24. K. Moriarty, T. Trappenberg, and C. Rebbi, *Comput. Phys. Commun.* **81,** 153 (1994).
25. J. Reynders, D. W. Forslund, P. J. Hinkel, M. Tholburn, D. G. Kilman, and W. F. Humphrey, *Comput. Phys. Commun.* **87,** 212 (1995).
26. P. C. Liewer and V. K. Decyk, *J. Comput. Phys.* **85,** 302 (1989).
27. S. C. Baden, S. R. Kohn, S. M. Figueira, and S. J. Fink, *The LPARX User's Guide v. 1.0* (unpublished).
28. A. Friedman, R. N. Sudan, and J. Denavit, *Plasma Phys.* **29,** 3317 (1986).