JOURNAL OF COMPUTATIONAL PHYSICS 102, 39–48 (1992)

A Comparison of Particle-in-Cell and Fokker–Planck Methods as Applied to the Modeling of Auxiliary-Heated Mirror Plasmas

RICHARD J. PROCASSINI*

Electronics Research Laboratory, University of California, Berkeley, California 94720

AND

BRUCE I. COHEN

Magnetic Fusion Energy Program, Lawrence Livermore National Laboratory, Livermore, California 94550

Received March 26, 1990; revised January 14, 1991

The transport and confinement of charged particles in an auxiliary-heated mirror plasma is modeled via two diverse computational tools: an implicit particle-in-cell (PIC) code and a bounce-averaged Fokker–Planck (F-P) code. The results from the PIC simulation are benchmarked against those obtained via use of F-P techniques, which has been the preferred means of analyzing plasma confinement and transport in mirror devices. The computer time required by each code to solve a specific test problem is presented, along with an itemization of the cost of the major processes involved in each method of solution. A qualitative discussion of the advantages and disadvantages of each code is also included.

1. INTRODUCTION

The magnetic mirror is an open-ended plasma configuration in which charged particles are confined by magnetic and electrostatic potentials. The confined-particle velocity distributions in such a device may deviate significantly from a Maxwellian distribution, and collisions have an essential influence. Therefore, the mirror fusion effort has relied heavily on methods of solving the Fokker–Planck (F-P) kinetic equation to determine the distribution of the trapped particles in the mirror plasma [1–3]. The F-P equation has been used to design mirror-based fusion experiments, to compare experimental results against theoretical calculations, and to predict the performance of mirror-based fusion reactors [1].

The solution of the F-P equation via numerical techniques can be a time-consuming proposition. A considerable amount of pre-processing is usually required to set up the computational domain over which the F-P equation is to be solved [4]. Progress in the development of implicit particle-in-cell (PIC) methods over the past decade [5, 6] has led to their use for simulations of large space- and time-scale plasma phenomena. The motivation of this paper is to show that these implicit PIC methods, when coupled with models which describe Coulomb collisional diffusion and auxiliary heating, are capable of reproducing the physics of mirror plasmas as predicted by a state-of-the-art F-P model. The PIC approach provides a self-consistent, first-principles solution to the confinement and transport of particles in a mirror plasma in a single run, while many F-P codes evolve only a single species in prescribed electrostatic potentials, thereby requiring an iterative scheme to determine the ambipolar confining potentials. It is hoped that the cost of a PIC simulation of mirror confinement and transport is competitive with that of the F-P calculation.

This paper is organized in the following manner. Sections 2 and 3 present brief reviews of the bounce-averaged F-P and PIC approaches, respectively. The choice of a suitable test case for a comparison of the two methods is the subject of Section 4. The test case that was chosen for this comparison is the simulation of neutral beam injection (NBI) into the end-plug of a tandem mirror plasma. The results obtained from the PIC code TESS [7, 8] are benchmarked against those of the F-P code SMOKE [4] in Section 5. The calculation of the ambipolar electrostatic potential by the PIC code was suppressed, thereby allowing a direct comparison with the results from the F-P code. A comparison of the computer time required by each code to perform this benchmark study, as well as a qualitative discussion of the advantages and disadvantages of each code are presented in Section 6. The conclusions of this benchmark comparison are discussed in Section 7.

* Present address: Inertial Confinement Fusion Program, Lawrence Livermore National Laboratory, Livermore, CA 94550.

0021-9991/92 $5.00

2. A REVIEW OF BOUNCE-AVERAGED FOKKER–PLANCK (F-P) METHODS

The relativistic F-P kinetic equation which describes the evolution of a distribution function $f(x, v, t)$ has the form

$$\frac{\partial f}{\partial t}+\mathbf{v}\cdot\frac{\partial f}{\partial \mathbf{x}}+\frac{\partial}{\partial \mathbf{p}}\cdot\left[q\left(\mathbf{E}+\frac{1}{c}\mathbf{v}\times\mathbf{B}\right)f+\mathbf{\Gamma}_{\text{coll}}+\mathbf{\Gamma}_{\text{rf}}\right]=S, \qquad (1)$$

where t is time, $\mathbf{x}$ is position, $\mathbf{v}$ is velocity, $\mathbf{p}\equiv\gamma m\mathbf{v}$ is momentum, $\gamma\equiv(1-v^2/c^2)^{-1/2}$ is the relativistic factor, m is the particle rest mass, c is the speed of light, $\mathbf{E}$ and $\mathbf{B}$ are the prescribed electric and magnetic fields, $\mathbf{\Gamma}_{\text{coll}}$ and $\mathbf{\Gamma}_{\text{rf}}$ are fluxes arising from Coulomb collisional and radiofrequency (RF) induced diffusion, and S is the particle source term. This equation is solved numerically by the SMOKE code [4] for either electrons *or* ions.

When the SMOKE code simulates neutral-beam heated plasmas, the particle source term S in (1) is given by a summation over beam components b,

$$S=\sum_b \alpha_b(z)(v_i^b+v_{cx}^b)\,n_i(z)\,s_b(\mathbf{v})-\sum_b \alpha_b(z)\,v_{cx}^b f, \qquad (2)$$

where $\alpha_b(z)$ is the spatial beam profile, v_{i}^b and v_{cx}^b are the ionization and charge exchange frequencies, $n_i(z)\equiv\int f d^3\mathbf{v}$ is the ion density, and $s_b(\mathbf{v})$ is the beam velocity distribution.

Since the bounce time of particles in a mirror plasma is much smaller than the collisional and RF diffusion times, or the ionization and charge-exchange times ($\tau_{\text{b}}\ll\tau_{\text{coll}}\sim\tau_{\text{rf}}\sim\tau_{\text{i}}\sim\tau_{\text{cx}}$), Eq. (1) may be converted to a bounce-averaged form by multiplying by $\mathbf{v}$ and integrating over the bounce motion $\int dz$, where z is position along the axis. The bounce-averaged version of (1), when expressed in terms of the independent variables total energy (ε) and magnetic moment (μ), is given by [9]

$$\tau_{\text{b}}\left(\frac{\partial f}{\partial t}-S\right)=\frac{\partial}{\partial\varepsilon}\left(D_{\varepsilon\varepsilon}\frac{\partial f}{\partial\varepsilon}+D_{\varepsilon\mu}\frac{\partial f}{\partial\mu}+D_{\varepsilon}f\right)+\frac{\partial}{\partial\mu}\left(D_{\mu\varepsilon}\frac{\partial f}{\partial\varepsilon}+D_{\mu\mu}\frac{\partial f}{\partial\mu}+D_{\mu}f\right), \qquad (3)$$

where $\varepsilon\equiv(\gamma-1)mc^2+q\Phi$ for electrostatic potential energy $q\Phi$, $\mu\equiv p_\perp^2/2mB_0$ and $\tau_{\text{b}}\equiv\int dz/|v_{||}|$ is the particle bounce time. Each of the diffusion coefficients in (3) has (in general) contributions from Coulomb collisional and RF-induced diffusion processes. For axisymmetric mirror applications, bounce-averaging reduces the number of phase space variables in the computational domain from 3 ($z, v_{||}, v_\perp$) to 2 (ε, μ). Since the bounce time is no longer resolved, the shortest time which must be resolved is on the order of the minimum of τ_{coll}, τ_{rf}, τ_{i}, or τ_{cx}.

The SMOKE code solves (3) for different "regions" (trapped-particle populations) in (ε, μ) phase space via a two-step process. First, the trapped regions are mapped from (ε, μ) phase space to a rectangular, Cartesian coordinate space (x, y) in order to simplify the computational domain. Equation (3) then becomes

$$J\tau_{\text{b}}\left(\frac{\partial f}{\partial t}-S\right)=\frac{\partial}{\partial x}\left(D_{xx}\frac{\partial f}{\partial x}+D_{xy}\frac{\partial f}{\partial y}+D_x f\right)+\frac{\partial}{\partial y}\left(D_{yx}\frac{\partial f}{\partial x}+D_{yy}\frac{\partial f}{\partial y}+D_y f\right), \qquad (4)$$

where $J\equiv(\partial\varepsilon/\partial y)(\partial\mu/\partial x)$ is the Jacobian of the transformation. Next, a Galerkin finite-element approximate solution of (4) is assumed

$$\hat{f}(x, y, t)=\sum_{i=1}^{N} c_i(t)\,B_i(x, y), \qquad (5)$$

where $c_i(t)$ is the ith coefficient and $B_i(x, y)$ is the ith basis function in the (x, y) space.

Equation (4) is therefore reduced to a matrix equation in terms of the coefficients $c_i(t)$

$$\mathbf{B}\cdot\frac{d\mathbf{c}}{dt}=\mathbf{A}\cdot\mathbf{c}, \qquad (6)$$

where the matrices $\mathbf{A}$ and $\mathbf{B}$ encompass the right and left sides of (4), respectively. An implicit scheme is used to advance (6) in time. This allows use of a time step which may be larger than any of the collisional/RF diffusion or NB injection times without loss of stability. Use of such a large time step will produce an equilibrium solution in a small number of steps; however, the transient evolution may not be correct.

3. A REVIEW OF PARTICLE-IN-CELL (PIC) METHODS

The TESS PIC code [7, 8] which is used for this benchmark study computes the trajectories of individual charged particles in either a prescribed or self-consistent electrostatic potential, and a prescribed magnetic field. A relativistic guiding center formulation of the equations of

motion in one spatial dimension ($\hat{z}$) along the axis of the mirror plasma is used,

$$\begin{aligned}\frac{dz}{dt}&=\frac{p_z}{\gamma m}\\ \frac{dp_z}{dt}&=qE_z-\frac{\mu}{\gamma}\nabla B_z+\left(\frac{dp_z}{dt}\right)_{\text{coll}}\\ &\quad+\left(\frac{dp_z}{dt}\right)_{\text{rf}}+\left(\frac{dp_z}{dt}\right)_{\text{nb}}\\ \frac{d\mu}{dt}&=\left(\frac{d\mu}{dt}\right)_{\text{coll}}+\left(\frac{d\mu}{dt}\right)_{\text{rf}}+\left(\frac{d\mu}{dt}\right)_{\text{nb}},\end{aligned}\tag{7}$$

where z is position, t is time, $p_z \equiv \gamma m v_z$ is the axial momentum, $\gamma \equiv (1-v^2/c^2)^{-1/2}$ is the relativistic factor, $\mu \equiv p_\perp^2/2mB_0$ is the magnetic moment, m and q are the particle rest mass and charge, and E_z and B_z are the electric and magnetic fields. A direct-implicit scheme [5, 6] is used to integrate (7). This implicit scheme provides numerically stable simulations, even if the time step is much larger than the electron plasma period ω_{pe}^{-1}. This allows one to simulate long-wavelength, low-frequency phenomena, without having to resolve high-frequency effects, such as electron plasma oscillations.

For multiple species, self-consistent simulations, the ambipolar potential is calculated via direct-implicit form of Poisson's equation

$$-\nabla\cdot\left(1+\sum_s \chi_s\right)\nabla\Phi(z)=4\pi e(Zn_i(z)-n_e(z)),\tag{8}$$

where the sum is over species s (ions and electrons), n_s is the free-streaming or explicit density, and χ_s is the implicit susceptibility, which is the implicit correction to the free-streaming charge density. Since the F-P model employed in this study is *not* capable of *self-consistently* calculating the ambipolar potential, it was decided to suppress the potential calculation in the PIC model. This was done in order to allow for a *direct* comparison, of the results from the two codes. In contrast to the simulations performed for this study, the implicit PIC code has been used to model many problems which have required the calculation of self-consistent electrostatic potentials, ranging from mirror plasma confinement to tokamak scrape-off layer transport [7, 10–12].

Equations (7) and (8) are solved via standard finite difference techniques. Several enhancements have been incorporated into the basic, implicit PIC model in order to simulate auxiliary-heated mirror plasma systems. These include (i) a self-consistent, relativistic, Monte Carlo, binary-particle Coulomb collision model [13] (which is a vectorized, relativistic implementation of the model developed by Takizuka and Abe [14]), (ii) a quasilinear RF diffusion model [15] (which is patterned after the theory of Bernstein and Baxter [9], as implemented by Rognlien [16]), and (iii) a neutral-beam ionization and charge-exchange package which employs standard Monte Carlo techniques [13, 15]. The time step cycle used in the TESS PIC code is shown schematically in Fig. 1.

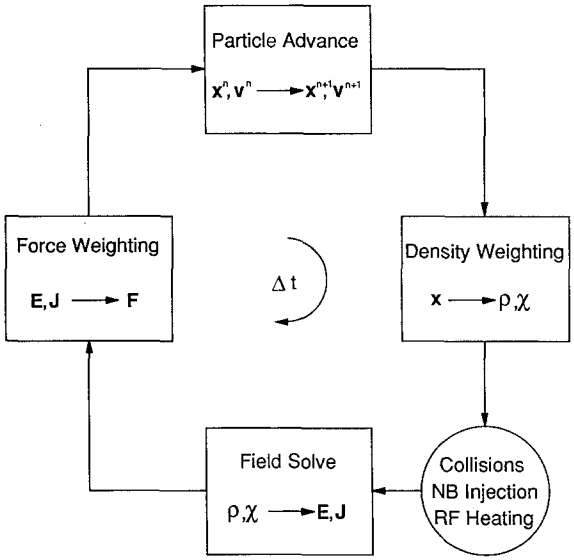

FIG. 1. A schematic diagram of the particle-in-cell (PIC) code time step cycle.

4. SELECTION OF THE BENCHMARK TEST CASE

The selection of a test case which will serve to benchmark the PIC code against the F-P code is not a trivial matter. Benchmarking the results of one computer code against those of another code can be a difficult undertaking. The situation is further complicated when the codes under consideration are as different in nature as a PIC code (which advances individual, discrete particles) and a F-P code (which evolves a continuum distribution function). In many ways, this benchmark is another case of "comparing apples and oranges." This section initially describes the constraints placed upon the size of the time step by each of the codes and then examines two mirror plasma systems which are possible candidate test cases for this comparison.

One of the important numerical quantities to be dealt with in simulations of mirror plasmas is the ratio of the collisional diffusion time to the particle bounce time $\tau_{\text{coll}}/\tau_b$, where

$$\tau_{\text{coll}}^{\alpha/\beta}\equiv\frac{m_\alpha^2 v_{\text{th},\alpha}^3}{4\pi q_\alpha^2 q_\beta^2 n_\beta \ln(\Lambda_{\alpha/\beta})}\tag{9}$$

and

$$\tau_b = \frac{2\pi L_b}{\langle v_\perp \rangle}, \tag{10}$$

where v_{th} is the particle thermal velocity, $\ln(\Lambda)$ is the Coulomb logarithm, and L_b is the half-distance between particle turning points in a mirror magnetic well. Note that the velocity that enters into (10) is the average perpendicular or gyrovelocity $v_\perp$. This result is obtained from the time-independent, fluid momentum equation (in the absence of any electric fields).

As noted above, the bounce-averaged F-P code SMOKE does *not* need to resolve τ_b, but (in general) only the shortest of τ_{coll}, τ_{rf}, τ_i, or τ_{cx}. On the other hand, the PIC code TESS must resolve fractions of τ_b, since the spatial dimension is gridded and the electrostatic and magnetic potentials are defined only on this grid. For accurate simulations, one must ensure that a particle with an average parallel velocity $\langle v_{||} \rangle$ traverses less than one grid cell per time step

$$\frac{\langle v_{||} \rangle \Delta t}{\Delta z} = f, \tag{11}$$

where $f \lesssim 1$, and Δz and Δt are the grid spacing and the time step, respectively. This constraint is sometimes referred to as a particle Courant condition, in analogy to the Courant–Friedrich–Lewy (CFL) condition encountered in the numerical treatment of fluid flow [17]. For the system where $L_b \simeq L$ the system length and $\langle v_\perp \rangle \simeq \langle v_{||} \rangle$, then the PIC code requires that

$$\tau_b \simeq \frac{2\pi N_g}{f} \Delta t, \tag{12}$$

since $\Delta z \equiv L/N_g$ with N_g the number of grid cells which span the system length L. This turns out to be a significant constraint on the size of the time step that may be used for mirror plasma simulations. For example, if one takes $f = 0.75$ and uses $N_g = 64$ cells over the distance L, then (12) requires 536 time steps per bounce time.

4.1. Test Case 1—An ECRH-Heated Tandem Mirror End Plug

This proposed test case simulates the transport and confinement of electrons *only* in prescribed electrostatic and magnetic fields. The electron density is fed by passing particles from a central cell. The electrons are heated by electron cyclotron resonance heating (ECRH) injection at two locations: fundamental injection ($\omega_{rf} = \omega_{ce}$) at the outer half-maximum B-field location and second-harmonic injection ($\omega_{rf} = 2\omega_{ce}$) near the bottom of the magnetic well. Figure 2 shows the magnetic and potential profiles for this system, along with the center locations at which RF wave energy is injected into the plasma.

This test case includes another important time scale: the energy diffusion time due to wave-particle interactions. The RF diffusion time is given by

$$\tau_{rf} \equiv \frac{(\Delta\varepsilon)^2}{D_{\varepsilon\varepsilon}} \sim \frac{(\Delta\varepsilon)^2}{E_{rf}^2}, \tag{13}$$

where $\Delta\varepsilon$ is the average energy kick received by a particle and E_{rf} is the peak RF electric field [9].

The initial conditions for this test case are $\langle n_e \rangle = 10^{12}\,\mathrm{cm}^{-3}$ and $\langle kT_e \rangle = 4\,\mathrm{keV}$. The electron pitch-angle scattering time is $\tau_{coll}^{e/e} = 1.32 \times 10^{-3}$ s, the RF diffusion time for electrons is $\tau_{rf} = 4.27 \times 10^{-4}$ s, and the electron bounce time is $\tau_b = 6.63 \times 10^{-7}$ s. The important time-scale ratios for the initial plasma are, therefore,

$$\frac{\tau_{rf}}{\tau_b} = 6.43 \times 10^3$$

and

$$\frac{\tau_{coll}^{e/e}}{\tau_b} = 1.99 \times 10^3.$$

The final conditions are $\langle n_e \rangle = 4.0 \times 10^{12}\,\mathrm{cm}^{-3}$ and $\langle kT_e \rangle = 40\,\mathrm{keV}$. Therefore $\tau_{coll}^{e/e} = 1.05 \times 10^{-2}$ s, $\tau_{rf} = 4.75 \times 10^{-3}$ s, and $\tau_b = 2.10 \times 10^{-7}$ s, such that

$$\frac{\tau_{rf}}{\tau_b} = 2.27 \times 10^4$$

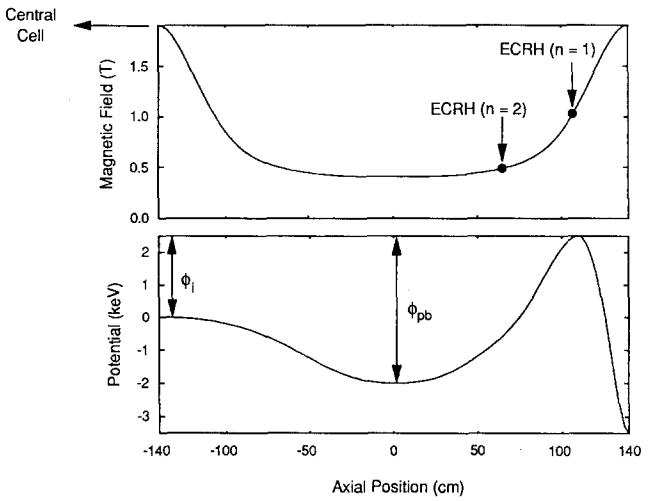

FIG. 2. Magnetic field and electrostatic potential profiles for the ECRH-heated tandem mirror end plug test case. The center locations of ECRH injection are shown on the magnetic field profile.

and

$$\frac{\tau_{\text{coll}}^{e/e}}{\tau_b} = 4.99 \times 10^4.$$

Since $\tau_{rf} < \tau_{\text{coll}}^{e/e}$, the F-P code need only use a time step $\Delta t_{F-P} \lesssim \tau_{rf}$ to obtain an accurate simulation of the approach to steady state. However, the PIC code (with $N_g = 64$ cells and $f = 0.75$) requires a time step which is 1/536 of a bounce time. It is estimated that the F-P code may need to simulate about 10 collisional diffusion times $\tau_{\text{coll}}^{e/e}$ for the system to come to equilibrium. Therefore, the F-P code requires $N_{t,\text{F-P}} \simeq 1111$ time steps for $\Delta t_{\text{F-P}} \simeq \tau_{rf}$, while the PIC code would require $N_{t,\text{PIC}} \simeq 2.67 \times 10^8$ time steps for $\Delta t_{\text{PIC}} \simeq \tau_b/536$ to reach a steady state. This is not an encouraging result for a simulation of this system based upon the PIC method!

This number could be substantially reduced *if* it is possible to accelerate both the collisional and RF-induced diffusion, thereby reducing the time-scale ratios $\tau_{\text{coll}}^{e/e}/\tau_b$ and τ_{rf}/τ_b, but maintaining the ratio $\tau_{\text{coll}}^{e/e}/\tau_{rf}$. Acceleration of the collisional diffusion is relatively straightforward, but acceleration of the RF-induced diffusion is much more complicated, requiring sophisticated energy-dependent acceleration factors and time-step subcycling [16]. These features are *not* easily incorporated into a PIC code such as TESS.

Therefore, one must conclude that the bounce-averaged F-P code is much better suited to the simulation of ECRH-heated mirror plasmas than is the direct-implicit PIC code.

4.2. Test Case 2—A NBI Sloshing-Ion Tandem Mirror End Plug

This proposed test case simulates the transport and confinement of ions *only* in a prescribed magnetic field without an ambipolar potential. The density is fed by neutral beam injection (NBI) of "sloshing" ions and gas-puff ionization. An equilibrium is reached when the loss of particles (resulting from collisional detrapping of the loss-cone distribution) balances the injection of particles into the trapped region of velocity space via NBI. The magnetic field profile and the orientation of the neutral beam are shown in Fig. 3.

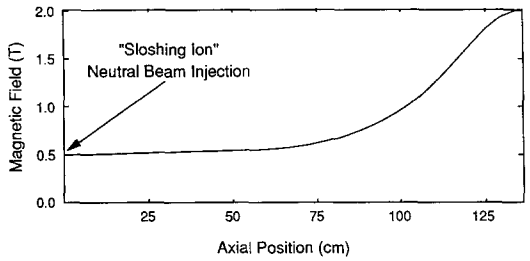

FIG. 3. The magnetic field profile and neutral beam orientation used in the "sloshing ion" tandem mirror end plug test case.

The initial or background ion conditions for this test case are $\langle n_i \rangle = 5.0 \times 10^{11}\ \text{cm}^{-3}$ and $\langle kT_i \rangle = 100$ eV. Therefore, the ion pitch-angle scattering time is $\tau_{\text{coll}}^{i/i} = 8.22 \times 10^{-4}$ s and the ion bounce time is $\tau_b = 1.18 \times 10^{-4}$ s, such that

$$\frac{\tau_{\text{coll}}^{i/i}}{\tau_b} = 6.96.$$

The final conditions associated with the beam ions ($i*$) are $\langle n_i \rangle = 5.0 \times 10^{12}\ \text{cm}^{-3}$ and $\langle \varepsilon_i \rangle = 5$ keV. Therefore $\tau_{\text{coll}}^{i*/i*} = 2.07 \times 10^{-2}$ s and $\tau_b = 1.63 \times 10^{-5}$ s, such that the important time-scale ratio is

$$\frac{\tau_{\text{coll}}^{i*/i*}}{\tau_b} = 1.27 \times 10^3,$$

while the ratios of the 90° scattering and slowing down times to the beam-ion bounce time are (assuming $\langle kT_e \rangle = 60$ eV)

$$\frac{\tau_{90}^{i*/i*}}{\tau_b} = 1.01 \times 10^3,$$

$$\frac{\tau_s^{i*/i*}}{\tau_b} = 1.48 \times 10^3,$$

$$\frac{\tau_{90}^{i*/e}}{\tau_b} = 1.32 \times 10^4,$$

and

$$\frac{\tau_s^{i*/e}}{\tau_b} = 4.70 \times 10^2.$$

The time step used in the F-P code need only resolve (approximately) the beam-ion drag time $\Delta t_{\text{F-P}} \lesssim \tau_s^{i*/e}$ to obtain an accurate approach to steady state. For $N_g = 64$ cells and $f = 0.75$, the PIC code again requires a time step which is $\frac{1}{536}$ of the bounce time. In order to reach an equilibrium, the F-P code needs to simulate about 47 beam-ion 90° scattering times $\tau_{90}^{i*/i*}$, such that $N_{t,\text{F-P}} \simeq 100$ time steps for $\Delta t_{\text{F-P}} \simeq \tau_s^{i*/e}$. The PIC code would therefore require $N_{t,\text{PIC}} \simeq 47(\tau_{90}^{i*/i*}/\tau_b)(\tau_b/\Delta t_{\text{PIC}}) \simeq 47(1.01 \times 10^3)(536) = 2.54 \times 10^7$ time steps to reach equilibrium. However, this number can be significantly reduced by accelerating the collisional diffusion, thereby reducing the time-scale ratio $\tau_s^{i*/e}/\tau_b$. By choosing a PIC collision acceleration factor $A_{\text{coll}} = 300$, there would be $\simeq 1.6$ bounces per drag time. This would reduce the required number of time steps to the range of $N_{t,\text{PIC}} \sim 8.0 \times 10^4 - 1.0 \times 10^5$, which although large, is not unreasonable if the number of simulation particles is kept low.

Based upon the previous arguments, one may conclude that the NBI tandem-mirror end plug is a reasonable test case for a comparison of the F-P and PIC codes.

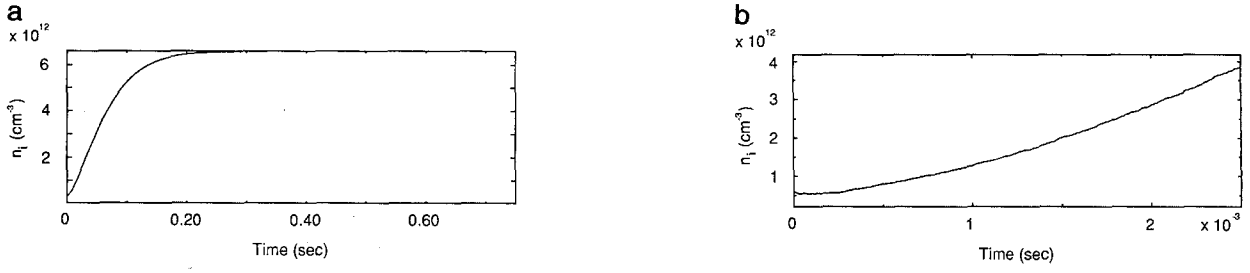

FIG. 4. The average-density time history of the ions from (a) the SMOKE F-P code and (b) the TESS PIC code.

5. BENCHMARK OF THE PIC CODE RESULTS AGAINST THE F-P CODE RESULTS FOR THE NBI TEST CASE

The tandem mirror end plug system with sloshing ions was evolved to a steady state by running the SMOKE F-P code for $N_{t,\text{F-P}} = 100$ time steps ($\Delta t_{\text{F-P}} = 7.5 \times 10^{-3}$ s). By accelerating the collision rate in the PIC model by $A_{\text{coll}} = 300$, the two codes were run for the same number of collision times $T_{\text{F-P,PIC}} \simeq 98\tau_s^{i*/e}$, which required $N_{t,\text{PIC}} = 81,100$ times steps in the PIC calculation ($\Delta t_{\text{PIC}} = 3.08 \times 10^{-8}$ s). Even though the final time in the PIC simulation represents only a small portion of the time required to reach equilibrium, the results from the PIC code are in reasonable qualitative agreement with those from the F-P code. The average-density time histories from SMOKE and TESS are presented in Fig. 4, while the average-kinetic-energy time histories from the two codes are shown in Fig. 5. The density and kinetic energy histories obtained from TESS are still increasing, while those from SMOKE have leveled off.

The ion velocity-space density contours $f_i(v_{||}, v_\perp)$ (see Fig. 6) and the distribution functions $f_i(v_{||})$ (see Fig. 7) and $f_i(v_\perp)$ (see Fig. 8) are compared at two axial locations: the magnetic-well midplane (and center of the NB footprint) at $z = 0$ cm, and near the turning points of the beam ions at $z = 100$ cm. The F-P code maps the trapped-particle distribution along the magnetic field lines to the desired location, while the PIC code determines the distribution of all particles (those trapped and in the loss cone) within a region which has a $\delta = 5$-cm half-width on either side of the desired location. The velocity-space density $f_i(v_{||}, v_\perp)$ is numerically integrated to obtain the distribution functions $f_i(v_{||})$ and $f_i(v_\perp)$.

There is general agreement between the velocity-space density contour plots produced by the two codes, including the location of the density maximum in velocity space at the well midplane ($z = 0$ cm). Note that there are particles in the loss cone in the velocity-space density contour plots from TESS (see Fig. 6b). These passing particles, which are *not* dealt with by SMOKE, come from two sources. The first component of the passing particle density is due to the creation of ions in the loss cone via charge exchange or ionization of the neutral-gas puff or the mirror trapped particles (the neutral beam only injects particles into the trapped region of velocity space). The second passing-particle density component arises from the Coulomb collisional pitch-angle scattering of trapped ions into the loss cone.

The ion distribution functions $f_i(v_{||})$ and $f_i(v_\perp)$ calculated by TESS agree well with those predicted by SMOKE. The peak of the $f_i(v_{||})$ distribution at $z = 0$ cm from TESS is smaller than that from SMOKE. This can be attributed to the fact that the results from TESS are not at steady state. As time went on, one would expect to see additional particles transferred to lower values of $p_{||}$ from the beam density peak around $p_{||} \simeq 1.6 \times 10^{-16}$ g-cm/s, due to the collisional drag of beam ions on background electrons.

The average density and kinetic energy profiles of the ions from the SMOKE F-P code are shown in Fig. 9a, while those from the TESS PIC code are presented in Fig. 9b. Note that the kinetic energy profile shown from SMOKE is the total kinetic energy ($\varepsilon_{\text{tot}} \equiv (\gamma - 1)mc^2$), while TESS provides the components of kinetic energy parallel and perpendicular to the axis of the device ($\varepsilon_{||,\perp} \equiv p_{||,\perp}^2/2m$).

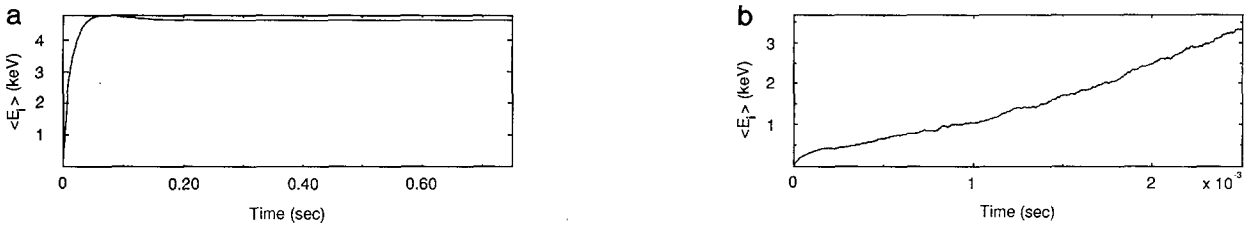

FIG. 5. The average-kinetic-energy time history of the ions from (a) the SMOKE F-P code and (b) the TESS PIC code.

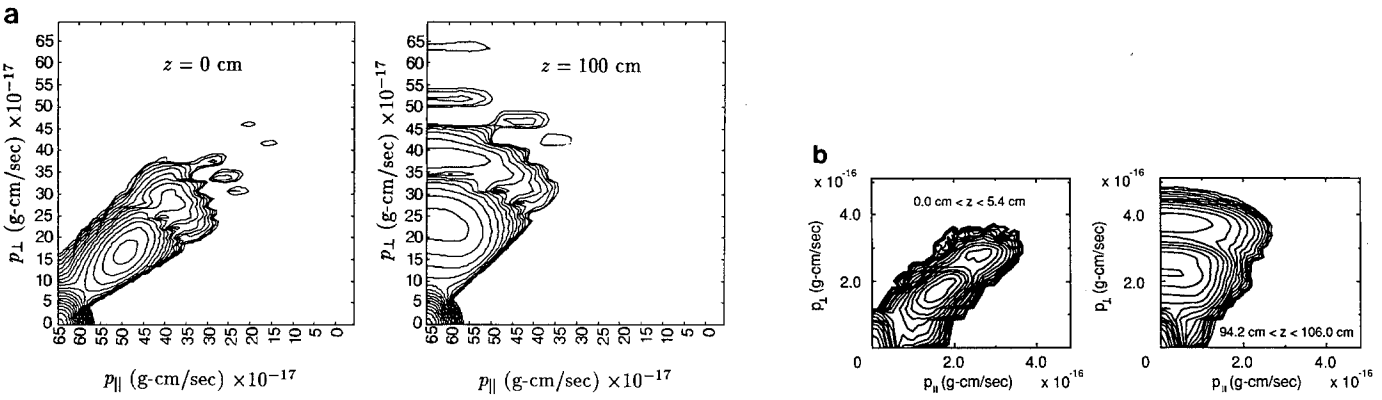

FIG. 6. The velocity–space density contours at the magnetic well midplane and the ion turning points from (a) the SMOKE F-P code and (b) the TESS PIC code.

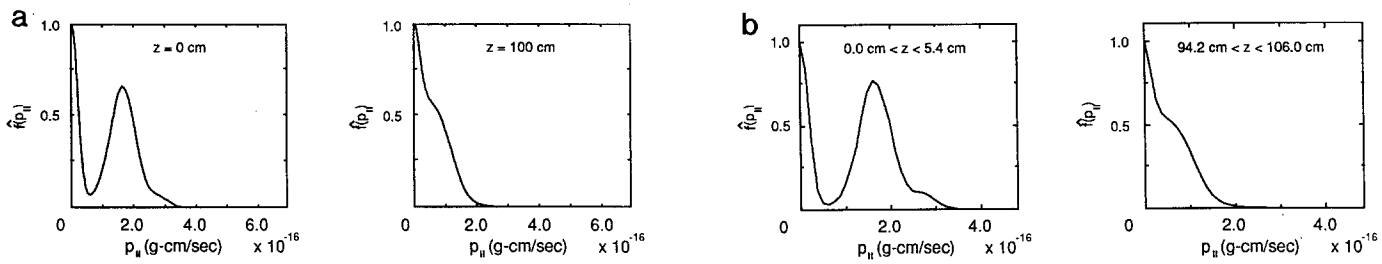

FIG. 7. The $f_i(v_{||})$ distributions at the magnetic well midplane and the ion turning points from (a) the SMOKE F-P code and (b) the TESS PIC code.

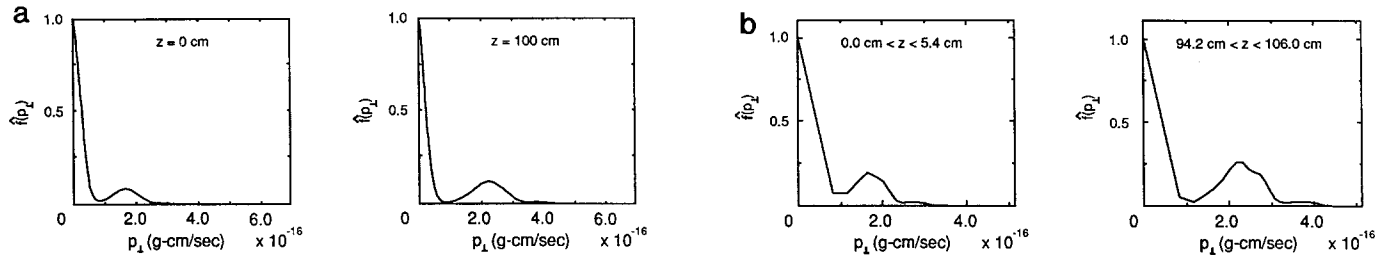

FIG. 8. The $f_i(v_\perp)$ distributions at the magnetic well midplane and the ion turning points from (a) the SMOKE F-P code and (b) the TESS PIC code.

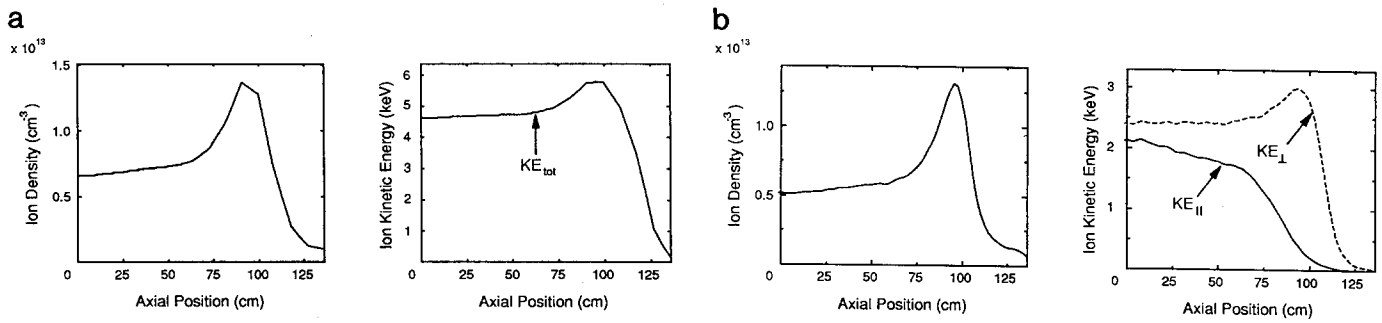

FIG. 9. The average density and kinetic energy profiles of the ions from (a) the SMOKE F-P code and (b) the TESS PIC code.

6. COMPARISON OF THE F-P AND PIC CODES

Having benchmarked the PIC code results against those from the F-P code, it is now time to examine the cost required by each code to perform this specific plasma simulation and to discuss the benefits and limitations of each code.

6.1. Simulation Cost Comparison

Since these codes are based upon different physical descriptions of a plasma and employ dissimilar numerical techniques, the criteria to be used in performing a cost comparison are not readily evident. Therefore, only the total run time (CPU *plus* input/output *plus* system time *plus* memory charges) of each code is directly compared. The total run time is then broken down into its major components, and the average cost of each component per time step is presented. Each code was generated using the Cray CFT compiler and run on a Cray X-MP/2 computer.

6.1.1. *Fokker–Planck (F-P) Code Cost Breakdown*

The SMOKE F-P code was run for $N_{t,\text{F-P}} = 100$ time steps with $\Delta t_{\text{F-P}} = 7.5 \times 10^{-3}$ s, such that the final time was $T_{\text{F-P}} = 0.75$ s. The time step is approximately equal to the beam-ion drag time $\tau_s^{i*/e}$; hence the fastest collisional time scale in this system is resolved. The total run time (CPU + I/O + SYS + MEM) for the SMOKE F-P code was $T_{\text{run,F-P}} = 75.34$ min on the Cray X-MP/2, or an average of 45.20 s per time step.

The major processes required to solve the F-P matrix equation (6) for $\hat{f}(x, y, t)$ are:

1. Calculation of the Rosenbluth potentials for that are used to determine the Coulomb collisional contribution to the diffusion coefficients: the total time required for this process was $T_{1,\text{F-P}} = 19.12$ min, or an average of 11.47 s per time step.

2. Mapping the trapped regions from the (ε, μ) phase space to the rectilinear (x, y) phase space, including the calculation of the Jacobian: the total time required for this process was $T_{2,\text{F-P}} = 13.34$ min, or an average of 8.00 s per time step.

3. Calculation of the neutral beam source term via Eq. (2): the total time required for this process was $T_{3,\text{F-P}} = 7.80$ min, or an average of 4.68 s per time step.

4. Inversion of matrix equation (6): the total time required for this process was $T_{4,\text{F-P}} = 3.35$ m, or an average of 2.01 s per time step.

Not surprisingly, the calculation of the Rosenbluth potentials turned out to be the single most costly process performed by the F-P code. The matrix equation (6) was inverted using a sparse matrix package which was developed at Yale University [18] for optimized execution on either Cray 1 or Cray X-MP computers.

6.1.2. *Particle-In-Cell (PIC) Code Cost Breakdown*

The TESS code was run with $N_g = 64$ grid cells with an average particle displacement per time step of $f = 0.75$, such that (12) requires $\Delta t_{\text{PIC}} = \tau_b/536 = 3.08 \times 10^{-8}$ s. The value of the Coulomb collisional acceleration factor was $A_{\text{coll}} = 300$, such that *only* $N_{t,\text{F-P}} = 81$, 100 time steps were required ($T_{\text{PIC}} = 0.74$ s). The number of simulation ions present in the system, when averaged over the length of the run, was found to be $\langle N_i \rangle = 2250$. The total run time (CPU + I/O + SYS + MEM) for the TESS PIC code was $T_{\text{run,PIC}} = 188.04$ min on the Cray X-MP/2, or an average of 61.83 μs per particle per time step.

The major processes performed by the PIC code to solve for the ion distribution function and its moments are (neglecting the cost of solving Poisson's equation for the ambipolar potential, which is not required because this is not a self-consistent simulation):

1. Integrating the trajectory of each ion every time step (particle pushing) according to Eq. (7): the total time required for this process was $T_{1,\text{PIC}} = 19.90$ min, or an average of 6.54 μs per particle per time step.

2. Calculation of Coulomb collisional diffusion by the Monte Carlo binary particle model: the total time used by the Coulomb collision package was $T_{2,\text{PIC}} = 104.41$ min, or an average of 34.33 μs per particle per time step.

3. Injection of particles (impact ionization and charge-exchange of the neutral beams and gas feed) by the neutral beam package: the total time required for this process was $T_{3,\text{PIC}} = 7.81$ min, or an average of 2.57 μs per particle per time step.

The Coulomb collision package accounts for the majority of the computer time used by the PIC code. This is directly attributable to the complexity of the binary-particle collision model, which conserves both the momentum and kinetic energy of the interacting particles. Conservation of these quantities is necessary for accurate energy balances and collisional diffusion rates in relativistic plasmas, such as ECRH-heated mirror devices.

As mentioned above, the bounce-averaging process allows one to use time steps in the F-P code which are, in general, much larger than the particle bounce time. In contrast to the bounce-averaged F-P model, and in common with the PIC model, mirror simulations using a standard F-P code (those that evolve the particle velocity distribution in $(z, v_{\|}, v_{\perp})$ space) would require resolution of a cell transit time in accordance with the aforementioned CFL condition. To simulate the NBI problem described above with a standard F-P code, where the grid resolution was the same as in the PIC model, one would require $N'_{t,\text{F-P}} = 2.54 \times 10^7$

time steps. If the collision rate in the standard F-P model could be accelerated by $A_{\text{coll}} = 300$ (as was done in the PIC simulation), $N''_{t,\text{F-P}} \simeq 8.5 \times 10^4$ time steps would still be needed. The computer time needed to perform such a run would be $T''_{\text{run, F-P}} \simeq (N''_{t,\text{F-P}}/N_{t,\text{F-P}})\, T_{\text{run, F-P}} = (8.5 \times 10^4/100)(75.34\ \text{min}) = 6.4 \times 10^4\ \text{min}$, or over 1000 h! This is a staggering figure, especially in light of the fact that the PIC run required only about 3 h.

6.2. The Advantages and Disadvantages of Each Code

The major benefits and limitations of each code, with regard to the simulation of auxiliary-heated mirror plasmas, are briefly outlined below.

6.2.1. *Advantages of the SMOKE F-P Code*

1. The main advantage of SMOKE is that it is bounce averaged. This procedure reduces the number of independent variables (and hence the complexity of the problem) from 3 $(z, v_{||}, v_\perp)$ to 2 $(v_{||}, v_\perp)$, or equivalently, (ε, μ). The minimum time step required in a given simulation is therefore considerably reduced, since it is not necessary to resolve the particle bounce time τ_b.

2. SMOKE employs an implicit scheme to evolve the distribution function in time. This allows one to use a time step which is much larger than the shortest diffusion time in the system, providing a rapid approach to equilibrium.

3. An aspect of SMOKE which can be considered either an advantage or a disadvantage is that the code only evolves the distribution function of trapped particles. Therefore, the code does not calculate $f(v)$ for particles that are injected into or scatter into the loss cone.

6.2.2. *A Disadvantage of the SMOKE F-P Code*

The SMOKE F-P code is only capable of evolving the distribution function of a single species in a given run. This means that it is not possible for SMOKE to calculate the ambipolar potential in a single run.

6.2.3. *Advantages of the TESS PIC Code*

1. The main advantage of TESS is that it can advance multiple species in a self-consistent ambipolar potential. It is therefore possible to determine the effect of the ambipolar potential on particle transport and confinement in a single run.

2. TESS uses an implicit algorithm to advance the particle trajectories. This allows one to accurately evolve the distribution function and its moments using time steps that are much larger than those required by explicit PIC codes in order to maintain numerical stability.

3. TESS follows the trajectories of both trapped and passing particles. This allows one to determine the distribution function and the moments of particles that escape out through the loss cone. In fact, TESS can model an almost arbitrarily complicated time-dependent phase-space boundary between trapped and passing particles, a feature that has not been incorporated into the SMOKE F-P code.

6.2.4. *A Disadvantage of the TESS PIC Code*

Since TESS follows particle orbits and the self-consistent potential as they evolve in time, it is subject to the constraint on the time step given by (12), which arises from the fact that the PIC code must obey a particle Courant condition $(\langle v_{||} \rangle\, \Delta t/\Delta z \lesssim 1)$ for accurate simulations.

7. SUMMARY AND CONCLUSIONS

The TESS particle-in-cell code has been used to follow the trajectories of charged particles in a neutral-beam-heated mirror plasma. TESS accurately reproduces the steady-state distribution function and moments predicted by the SMOKE F-P code, even though the PIC simulation was only run part of the way to equilibrium. The PIC code requires considerably more computer time to reach steady state than does the bounce-averaged F-P code.

Based upon this observation, it is reasonable to conclude that a bounce-averaged F-P code is better suited than a PIC code to determine the equilibrium distribution function of a single species confined by magnetic and electric potentials in a mirror device. However, for multiple-species calculations, where a self-consistent ambipolar potential is desired, only the PIC code is capable of providing all of the physics in a single run. Therefore, the PIC code may be used to accurately simulate the particle dynamics during the startup of a mirror plasma, wherein both the self-consistent potential and particle orbits evolve in time. However, it should be noted that the total run time for both the F-P and PIC models will increase when a calculation of the self-consistent ambipolar potential is included. In the PIC code, where the cost of the potential solver is usually a small fraction of the cost of advancing the particle trajectories, the total run time may rise significantly if the number of particles must be increased to ensure that the thermal fluctuations in the self-consistent electric field are adequately small.

ACKNOWLEDGMENTS

The authors thank Y. Matsuda for allowing them to use the SMOKE F-P code and for providing them with a tutorial describing the input and output of the code. Discussions with T. D. Rognlien and C. K. Birdsall are gratefully acknowledged. This research was supported by a grant from the Plasma Physics Research Institute of the Lawrence Livermore National Laboratory under the auspices of the U. S. Department of Energy, Contract Number W-7405-Eng-48. Additional funding was provided to the Plasma Theory and Simulation group at the University of California at Berkeley by the U.S. Department of Energy under Contract Number DE-FG03-86ER53220.

REFERENCES

1. R. F. Post, "The Fokker–Planck Equation in Mirror Research," Report UCRI-89200, Lawrence Livermore National Laboratory, Livermore, 1983 (unpublished).
2. J. Killeen and K. D. Marx, *Methods Comput. Phys.* **9**, 421 (1970).
3. J. Killeen, A. A. Mirin, and M. E. Rensink, *Methods Comput. Phys.* **16**, 369 (1976).
4. Y. Matsuda and J. J. Stewart, *J. Comput. Phys.* **66**, 197 (1986).
5. B. I. Cohen, A. B. Langdon, and A. Friedman, *J. Comput. Phys.* **46**, 15 (1982).
6. B. I. Cohen, A. B. Langdon, D. W. Hewett, and R. J. Procassini, *J. Comput. Phys.* **81**, 151 (1989).
7. B. I. Cohen, J. C. Cummings, R. J. Procassini, and C. K. Birdsall, in *Proceedings, 12th Conf. Num. Sim. Plasmas, San Francisco, 1987*, Paper PW15.
8. R. J. Procassini and B. I. Cohen, "The TESS Computer Code User's Manual," Report UCRL-ID 104092, Lawrence Livermore National Laboratory, Livermore, 1990 (unpublished).
9. I. B. Bernstein and D. C. Baxter, *Phys. Fluids* **24**, 108 (1981).
10. R. J. Procassini, C. K. Birdsall, and E. C. Morse, *Phys. Fluids B* **2**, 3191 (1990).
11. R. J. Procassini, C. K. Birdsall, and B. I. Cohen, *Nucl. Fusion* **30**, 40 (1990).
12. R. J. Procassini and C. K. Birdsall, *Phys. Fluids B* **3**, 1876 (1991).
13. R. J. Procassini, Ph.D. thesis, Chaps. 3, 4, University of California, Berkeley, 1990 (unpublished).
14. T. Takizuka and H. Abe, *J. Comput. Phys.* **25**, 205 (1977).
15. R. J. Procassini and B. I. Cohen, "Auxiliary Plasma Heating and Fueling Models for Use in Particle Simulation Codes," Report UCID-21654, Lawrence Livermore National Laboratory, Livermore, 1989 (unpublished).
16. T. D. Rognlien, *Phys. Fluids* **26**, 1545 (1983).
17. R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow* (Springer-Verlag, New York, 1983), pp. 42, 47.
18. S. C. Eisenstat, M. C. Gursky, M. H. Shultz, and A. H. Sherman, "The Yale Sparse Matrix Package," Research Report 114, Department of Computer Sciences, Yale University, 1983 (unpublished).