Plasma Simulation on Magnetohydrodynamic Time Scales

J. U. Brackbill

CRPC-TR90040
March, 1990

Center for Research on Parallel Computation
Rice University
P.O. Box 1892
Houston, TX 77251-1892

AS111

PLASMA SIMULATION ON MAGNETOHYDRODYNAMIC TIME SCALES

J. U. Brackbill

Los Alamos National Laboratory
Los Alamos, New Mexico 87545, USA

Abstract: Progress in implicit-moment methods for plasma simulation on magnetohydrodynamic (MHD) time scales is described. A method is sought that is applicable not only to transport on MHD time scales, but also to plasma interactions with a wall and to sheath formation. The implicit field equations for a plasma in a DC magnetic field are given, and a method is described for their solution. An adaptive grid formulation is outlined, and a problem initialization presented to illustrate its application. The issue of nonlinear stability, which is still unresolved, is reviewed.

(Keywords: Plasma Simulation, Implicit-Moment-Method, Adaptive Grid)

Introduction

It is a major challenge in modeling magnetically confined plasmas to close the large gap between the time and space scales in magnetic fusion experiments, and the time and space scales on which one can solve the plasma kinetic equations. To understand the kinetics of energy confinement and impurity production in magnetic confinement experiments, one must bridge this gap. Methods to model kinetic effects in three dimensions on magnetohydrodynamic time scales in realistic geometries are the ultimate goal.

Current research in plasma simulation methods seeks to increase the capabilities of kinetic simulations along two lines: the development of new methods that can be used on the highly parallel computers of the future, and the development of new numerical techniques that can be used on the supercomputers of the present. Progress along the second line and its impact on modeling capabilities in the intermediate term is emphasized.

It is already a decade since new methods for simulating plasmas on long time scales were introduced. The methods include implicit methods, which allow one to integrate the equations of motion using large time steps [1, 2, 3], and gyrokinetic methods, which eliminate the fast time scales by asymptotic analysis[4]. The gyrokinetic methods have demonstrated capabilities for modeling drift instabilities in magnetized plasmas. The implicit methods are general purpose methods, with applications to edge effects as well as transport problems.

The implicit methods use time-implicit formulations of the equations of motion that eliminate the stability constraint on the size of the time step. The implicit equations yield time-resolved solutions for all those processes that evolve sufficiently slowly, and time-asymptotic solutions for processes which evolve too rapidly to be resolved by the time step.

Plasma Equations of Motion

The equations of motion for a plasma in a magnetic field include an equation to calculate the particle position, $\mathbf{x}_p$, from the particle velocity, $\mathbf{u}_p$,

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{u}_p \tag{1}$$

a momentum equation to calculate the particle velocity in the electric, $\mathbf{E}$, and magnetic, $\mathbf{B}$, fields,

$$\frac{d\mathbf{u}_p}{dt} = \frac{q_s}{m_s}\left[\mathbf{E} + \frac{\mathbf{u}_p \times \mathbf{B}}{c}\right], \tag{2}$$

(where q_s/m_s is the charge to mass ratio, and c is the speed of light), and Poisson's equation to calculate the electric field from the net charge density, n,

$$\nabla \cdot \mathbf{E} = 4\pi \sum_s n_s . \tag{3}$$

The particles are labeled by the subscript p, and the species (e.g. ions and electrons) by the subscript s.

Numerical Plasma Simulation

In a numerical calculation, Eqs. (1-3) are approximated by finite differences. Consider the approximation to derivatives with respect to time. Semi-discrete approximations to Eqs. (1-3) are,

$$\mathbf{x}_p^1 = \mathbf{x}_p^0 + \mathbf{u}_p^{1/2}\Delta t, \tag{1d}$$

$$\mathbf{u}_p^{1/2} = \mathbf{u}_p^0 + \frac{q_s}{m_s}\left[\mathbf{E}^\theta + \frac{\mathbf{u}_p^{1/2} \times \mathbf{B}}{c}\right]\frac{\Delta t}{2}, \tag{2d}$$

$$\nabla \cdot \mathbf{E}^\theta = 4\pi n^\theta . \tag{3d}$$

where Δt is the time step and n^θ is the charge density at $(t+\theta\Delta t)$. The charge density is calculated by convolving the particle distribution with a B-spline, S[5], with properties,

$$S(x;h) = \prod_{\alpha=1}^{3} S(x_\alpha;h_\alpha) \tag{4a}$$

$$h_\alpha = \int S(x_\alpha;h_\alpha)\,dx_\alpha . \tag{4b}$$

The convolution is written,

$$n^\theta = \sum_s \sum_p \int d^3x' S(x-x';h) q_s \delta(x'-x_p^\theta)/h^3 , \tag{5d}$$

and $x_p{}^\theta$, for example, is given by,

$$x_p^\theta = (1-\theta)x_p(t) + \theta x_p(t+\Delta t). \tag{6}$$

When $\theta=0$, the difference equations (1d-3d, 5d) are explicit. An evaluation of (5d) from the particle data, a solution of (3d), and a subsequent solution of (1d) and (2d) comprise a computation cycle in an explicit calculation.

Explicit Differencing

With an explicit scheme, the time step is limited by a linear stability condition, which for Eqs. (1d-3d) is given by,

$$\sin^2\left(\frac{\omega\Delta t}{2}\right) = \sum_s \left(\frac{\omega_{ps}\Delta t}{2}\right)^2 . \tag{7}$$

The plasma frequency for each specie is,

$$\omega_{ps}^2 = \frac{4\pi n_s q_s}{m_s}, \tag{7a}$$

and the sum over species is written simply, $\omega_p{}^2$. When $\omega_p\,\Delta t<2$, the roots of the dispersion relation are real and the difference equations are stable. When $\omega_p\Delta t>2$, the roots are complex and the equations are unstable. Since ω_p can be much higher than the frequency of interest, one pays a very high price to model low-frequency phenomena because the time step is so small.

Implicit Differencing

Implicit difference equations eliminate the stability constraint. When $1/2<\theta<1$, Eqs. (1d-3d,5d) are implicit, and the dispersion relation (with $\theta=1/2$) is

$$\tan\left(\frac{\omega\Delta t}{2}\right)\sin\left(\frac{\omega\Delta t}{2}\right) = \sum_s \left(\frac{\omega_{ps}\Delta t}{2}\right). \tag{8}$$

Even with very large Δt, the roots are real and lie in the interval, $0<\omega\Delta t<\pi$.

It is difficult to solve implicit equations. To calculate $\mathbf{E}^\theta$, one must calculate n^θ. To calculate n^θ, one must calculate $x_p{}^\theta$. To calculate $x_p{}^\theta$, one must calculate $\mathbf{E}^\theta$. Langdon[6] remarks that one cannot iterate (1d-3d,5d) to obtain a solution.

The equations can be solved approximately using either the direct implicit method[2], or the implicit moment method[1]. "The essence of the "direct" method is that we work *directly* with the particle equations of motion and particle-field coupling equations. These are linearized about an estimate (extrapolation) for their values at the new time level $(t+\Delta t)$[2]". The implicit moment method uses implicitly differenced fluid equations to estimate n^θ.

Implicit Moment Method

The derivation of the implicit moment equations begins with the definitions of the current density,

$$J_s^{1/2}(x) = \sum_p \int d^3x' S(x-x';h) q_s u_p^{1/2} \delta(x'-x_p^{1/2}), \tag{9}$$

and the pressure,

$$\Pi_s^0(x) = \sum_p \int d^3x' S(x-x';h) q_s u_p^0 u_p^0 \delta(x'-x_p^0). \tag{10}$$

If one expands the expression for the charge density about $x_p{}^0$ in powers of $\varepsilon=k\langle\delta x^2\rangle^{1/2}$, where k is a characteristic wave number, and $\langle\delta x^2\rangle$ is the mean square particle displacement in a time step, an approximate value of the net charge density is given by,

$$n^\theta = n^0 - \nabla\cdot J^{1/2}\theta\Delta t + O(\varepsilon^2), \tag{11}$$

and the net current density by,

$$J^{1/2} = J^0 - \nabla\cdot\Pi^0\frac{\Delta t}{2} + \frac{q_s}{m_s}\left[n_s^0 E^\theta + \frac{J_s^{1/2}\times B}{c}\right]\frac{\Delta t}{2} + O(\varepsilon^2). \tag{12}$$

In a computation cycle of the implicit moment method, n^0, J^0, and Π^0 are evaluated from the particle data. If terms of $O(\varepsilon^2)$ are neglected, solving for $\mathbf{E}^\theta$ requires the solution of a linear system of equations, including the moment equations, (11-12), and Poisson's equation, (3d). The value of $\mathbf{E}^\theta$ that one calculates this way approximates the solution of the implicit plasma equations to $O(\varepsilon^2)$. Provided $k\langle\delta x^2\rangle^{1/2}<1$, the errors are small.

The Tensor Dielectric

In a DC magnetic field, one is required to solve only a single elliptic equation for the electric field. The momentum equation can be solved for $J_s{}^{1/2}$. This value is substituted into into the continuity equation, (11), and the result substituted for n^θ in Poisson's equation, (3d), to derive a linear equation for $\mathbf{E}^\theta$, written,

$$\nabla\cdot\varepsilon\cdot E^\theta = 4\pi\sum_s n_s^0 - \nabla\cdot J_s^*\theta\Delta t. \tag{13}$$

The current, J_s'', is calculated from,

$$J_s'' = \frac{J_s' - J_s' \times \frac{\Omega_s \Delta t}{2} + J_s' \cdot \Omega_s \Omega_s (\frac{\Delta t}{2})^2}{1 + (\frac{\Omega_s \Delta t}{2})^2}, \quad (14)$$

where,

$$J_s' = J_s^0 - \nabla \cdot \Pi^0 \frac{\Delta t}{2}. \quad (15)$$

The dielectric tensor, ε, is defined by,

$$\varepsilon \cdot E^\theta = (1 + \varepsilon_{perp}) E^\theta + \varepsilon_{trans} E^\theta \times B + \varepsilon_{par} E^\theta \cdot B B. \quad (16)$$

(The dielectric, ε, is the same as the "implicit susceptibility" derived for the direct implicit method[2].) The definitions for the various quantities that appear in (14-16) are,

$$\varepsilon_{perp} = \sum_\sigma \beta_s, \ \varepsilon_{par} = \sum_s \beta_s \alpha_s, \ \varepsilon_{trans} = \sum_s \beta_s \alpha_s, \quad (17)$$

$$\alpha_s = \frac{q_s \Delta t}{m_s 2}, \ \beta_s = \frac{\omega_{ps}^2 \frac{\theta \Delta t^2}{2}}{1 + \left\{\frac{\Omega_s \Delta t}{2}\right\}^2}, \ \Omega_s = \frac{q_s}{m_s} B. \quad (18)$$

All of these intermediate variables are easily calculated.

The dielectric tensor contains information to anticipate the response of the plasma to the electric field, so that while the right side contains information at time t, the electric field can be calculated at time $t+\Delta t$.

Solution of Poisson's Equation

There are difficulties in solving Poisson's equation, (13) for the electric field. When the field equation is replaced by a finite difference approximation to model an inhomogeneous and magnetized plasma, one must invert a matrix that not only has variable coefficients, but also has no symmetry. (The symmetry is broken by the **E×B** drift term, which is anti-symmetric.) Many researchers, to avoid the cost of inverting a matrix with variable coefficients, have used a global iteration method[7]. In this method, a "variable-coefficient operator is approximately a simpler operator whose inverse may be obtained directly. In global iteration techniques, this approximate inverse is applied to the residue...of the full elliptic equation 2". The defect-correction iteration is in a form that can be solved using standard solvers such as fast-Fourier-transform (FFT). The method seems to work best for relatively homogenous plasmas in relatively weak magnetic fields[8, 9]. More recent results with an incomplete Cholesky decomposition and conjugate gradient iteration[10] in a simulation code, CELESTE, contradict the conventional wisdom. Eq.(13) is solved routinely in CELESTE with time steps that are orders of magnitude larger than previously reported using these standard methods[11].

Nonlinear Stability

For very low-frequency phenomena, the fluctuations due to numerical causes may dominate the results. Horton *et al* [12] argue that "In particle simulations a practical limit is encountered to the number of gyroradii contained in the gradient scale length.... Since the amplitude of the fluctuations scale proportional to {this number}...{simulations} are restricted to the domain of relatively high fluctuation levels". One result of high fluctuation levels is plasma heating. Cohen *et al* [13] systematically examined the energy conservation properties of the direct-implicit method for a wide range of time steps and plasma conditions. They found that plasma heating did occur under certain conditions. In addition, nonlinear numerical instabilities have been observed with implicit methods when $\omega_{pe}\Delta t$ (ω_{pe} is the electron plasma frequency) exceeds the number of particles in a cell [13]. The nonlinear instability causes plasma heating.

(Lee[14] argues that the low fluctuation level of the gyrokinetic model relative to ordinary plasma simulations is a unique advantage. Direct comparisons of gyrokinetic and implicit plasma simulations are in progress to evaluate the correctness of this conjecture.)

If the heating is too large, results of calculations of slowly growing instabilities may not be meaningful. To reduce heating, numerical dissipation can be used, but the dissipation also will reduce the growth rate of physical instabilities.

In calculations with CELESTE, heating is observed with $\omega_{pe}\Delta t$=10000. The heating rate increases with temperature, as one would expect if the heating were due to errors in the estimated fields, (13). The growth time for the exponential increase in energy is presently several hundred time steps, and seems not to depend on the number of particles per cell, in contrast to results with the direct method [13]. Further efforts to improve the accuracy of the calculation of the effective charge and of the dielectric tensor are expected to reduce heating to acceptable levels without adding additional numerical dissipation. However, the problem of nonlinear stability of the implicit simulation remains an important, and yet unresolved, issue.

Real Geometries

In CELESTE, the particle equations of motion are solved on a grid of

arbitrarily shaped quadrilaterals. This capability has been built into the formulation of the particle equations of motion and into the data structures[15, 16, 17]. One is free to redefine the grid for each cycle of a computation as needed, and according to a prescription that improves the accuracy or efficiency of the calculation. For example, one can use a body-fitted grid [18] to model the geometry of magnetic confinement experiments by causing the computational domain to conform to the structure and shape of the boundary [17].

Natural Coordinates

The solution of the equations of motion on an arbitrary grid uses natural coordinates, (ξ,η), which are calculated by mapping each of CELESTE's quadrilateral cells on to a unit square. At each vertex, the natural coordinates assume integer values, (i,j), which are constant as the mesh moves. Elsewhere, the mapping between physical and natural coordinates is given by bilinear interpolation, which can be written as a tensor product of linear B-splines,

$$x_p = \sum_{i,j} x_{i,j} S_1(\xi - i, \eta - j) = \sum_{i,j} x_{ij} S_1(\xi - i) S_1(\eta - j) \qquad (19)$$

The natural coordinates of a particle, whose physical coordinates are $\mathbf{x}_p$, are defined by inverting this mapping. The particle acceleration, Eq. (2d), is calculated by interpolating the electric and magnetic fields from the grid values.

The particle shape function, S is defined in terms of the natural coordinates. With this choice, the grid defines the particle size in physical space, but h, the support of S, which defines the particle size in natural coordinates, is always 1.

The Grid Generator

The grid may be completely redefined from one cycle to the next, except that the new grid also must map on to a logical rectangle. In CELESTE, the number of grid points is fixed, but the grid points may be moved about as needed. Presently, one may choose to use an Eulerian or an adaptive grid.

The adaptive grid is generated by solving a variational problem, in which one minimizes a functional. The functional may be complex, allowing one to control grid smoothness, orthogonality, spacing and orientation [19, 20]; or it may be simple, giving control only over spacing, such as the functional ,

$$I = \int_V \frac{\left[\nabla\xi^2 + \nabla\eta^2\right]}{w} dV . \qquad (20)$$

To minimize I, one solves finite-difference approximations to the Euler equations.

With w=1, minimizing I gives a body-fitted grid[18].
When w is not constant, its variation controls the mesh spacing. Where w is larger, the grid points are closer together, and where w is smaller, the grid points are further apart. If w is an appropriate function of the data, the grid will adapt to provide increased resolution where it is needed by moving grid points. For example, if w is proportional to the current density in a calculation of flow in a time-dependent magnetic field, the grid will cluster to resolve steep gradients in the magnetic field intensity [21].

Adaptive gridding can be useful in PIC calculations. For example, when w is proportional to the number of particles per zone, where the particles are sparse the zones will be larger. This extends the range of densities that can be represented by making the particle size larger in regions of low density.

Conversely, the adaptive grid can be used to generate inhomogeneous initial conditions. One notes that it is convenient to load a constant number of particles in each cell initially. Further, it is better to give each particle the same significance, i.e. mass and charge. Since particles in a collisionless plasma may move long distances from their initial positions, one must expect that particles that were initially widely separated will eventually be close together. To avoid having neighboring particles with radically different weights, all particles should begin with approximately the same weight.

It is only possible to achieve both aims on a mesh with variable spacing. To generate the correct spacing, one uses an adaptive grid, even in two dimensions.

In one dimension, the Euler equation that minimizes the adaptive grid functional, (20), is

$$\frac{\partial}{\partial\xi}\left[\frac{1}{w}\frac{\partial x}{\partial\xi}\right] = 0. \qquad (21)$$

In finite difference form, it is useful to add a "time-dependent" term to avoid singularity when w=0, and write,

$$\delta(x_i^1 - x_i^0) - \frac{1}{2}(w_{i+1/2} + w_{i-1/2})(x_{i+1}^1 - 2x_i^1 + x_{i-1}^1)$$

$$= -\frac{1}{2}(w_{i+1/2} - w_{i-1/2})(x_{i+1}^1 - x_{i-1}^1), \qquad (22)$$

where δ is a small parameter with appropriate dimensions, and x_i^0, i=1,N, is a uniformly spaced grid with N points.

In this example, the charge density, n, is given by,

$$n = n_0\left(1 + e^{-\frac{x}{L}}\right), \qquad (23)$$

with L=10. The particles are generated on the grid x^1, with spacing shown in Fig. (1). The number of particles per cell is constant, Fig. (2), but the charge density calculated from the particles using (5d), Fig. (3), is close to the prescribed density, (23).

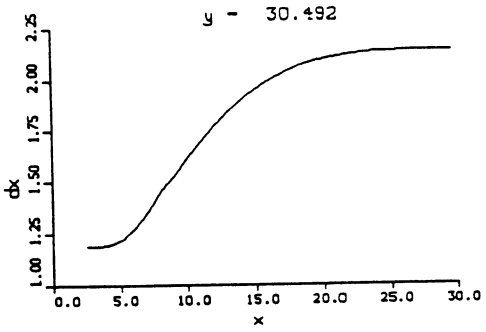

Figure 1. The mesh spacing is plotted against the grid index, i.

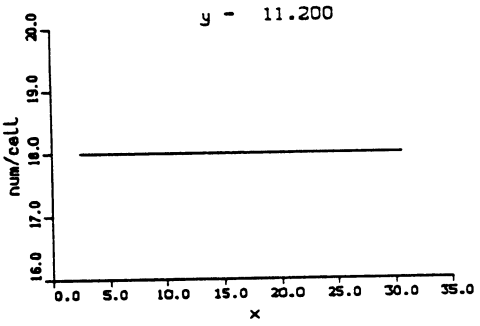

Figure 2. The number of particles per cell is plotted against the cell index.

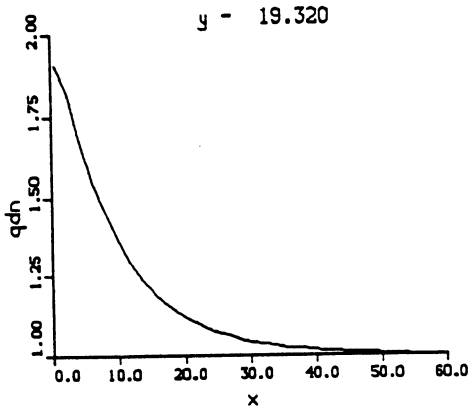

Figure 3. The charge density is plotted against the physical coordinate. The variation is approximately the prescribed profile, Eq. (23).

The resolution required to resolve gradients or to keep the number of particles per cell constant changes in time, and, consequently, the grid can be regenerated as often as every time step. During grid generation, the particles are stationary in physical space, but their natural coordinates must be recomputed.

Applications

A discussion of past and present applications of the implicit plasma simulation methods conveys the opportunities these methods provide.

A computational study of edge effects in magnetized plasmas has yielded the discovery of high frequency sheath oscillations with implications for future experiments[22]. Calculations of edge effects in inhomogeneous plasmas in two dimensions has uncovered new transport mechanisms. On much longer time scales, the comparison of the results of implicit calculations of drift instabilities with those from the gyrokinetic model give perspective on the relative capabilites of the two methods[11].

Discussion

There are as yet unsolved problems in modeling bounded plasmas. In bounded plasmas, the need to improve the definition of particle orbits beyond the boundary of the domain becomes more acute as the time step increases. Magnetized plasmas at a conducting boundary provide interesting examples of the nonphysical behavior that may arise. Additional physics must be added to the simulation models. For example, collisions in strongly magnetized plasmas must be modeled. An implicit, Monte-Carlo model for collisions provides a partial solution, but more work is needed[23].

References

1. J. U. Brackbill, D. W. Forslund, in *Multiple Time Scales* J. U. Brackbill, B. I. Cohen, Eds. (Academic Press, Orlando, 1985), pp. 271-310.

2. A. B. Langdon, D. C. Barnes, in *Multiple Time Scales* J. U. Brackbill, B. I. Cohen, Eds. (Academic Press, Orlando, 1985), pp. 335-375.

3. R. J. Mason, in *Multiple Time Scales* J. U. Brackbill, B. I. Cohen, Eds. (Academic Press, Orlando, 1985), pp. 233-270.

4. W. W. Lee, *J. Comput. Phys.* **72**, 243 (1987).

5. C. d. Boor, *A Practical Guide to Splines* (Springer-Verlag, 1978).

6. A. B. Langdon, *J. Comput. Phys.* **30**, 202 (1979).

7. P. Concus, G. H. Golub, *SIAM J. Numer. Anal.* **10**, 1103 (1973).

8. D. C. Barnes, T. Kamimura, (Nagoya Institute of Plasma Physics, 1982),

9. M. Tanaka, *J. Comput. Phys.* **79**, 209 (1988).

10. T. J. Jordan, in *Parallel Computations* G. Rodrigue, Eds. (Academic Press, Orlando, 1982), pp. 1.

11. J. U. Brackbill, D. W. Forslund, *Bull. Am. Phys. Soc.* **34**, 2045 (1989).

12. W. Horton, R. D. Estes, D. Biskamp, *Plasma Phys.* **22**, 663 (1980).

13. B. I. Cohen, A. B. Langdon, D. W. Hewett, R. J. Procassini, *J. Comput. Phys.* **81**, 151-168 (1989).

14. W. W. Lee, W. M. Tang, *Phys. Fluids* **31**, 612 (1988).

15. T. Abe, *J. Comput. Phys.* **83**, 424 (1989).

16. J. U. Brackbill, H. M. Ruppel, *J. Comput. Phys.* 314 (1986).

17. T. Westermann, in *Thirteenth Conference on the Numerical Simulation of Plasmas* R. J. Mason, Eds. Santa Fe, NM, 1989), pp. IM3.

18. J. F. Thompson, Z. U. A. Warsi, C. W. Mastin, *Numerical Grid Generation: Foundations and Applications* (Elsevier, New York, 1985).

19. J. U. Brackbill, J. S. Saltzman, *J. Comput. Phys.* **46**, 342-367 (1982).

20. A. E. Giannakopoulos, A. J. Engel, *J. Comput. Phys.* **74**, 422-439 (1988).

21. R. D. Milroy, J. U. Brackbill, *Phys. Fluids* **25**, 775 (1982).

22. J. R. Myra, e. al, *Bull. Am. Phys. Soc.* **34**, 2038 (1989).

23. C. W. Cranfill, J. U. Brackbill, S. R. Goldman, *J. Comput. Phys.* **66**, 239 (1986).