Computer Physics Communications 48 (1988) 117–125
North-Holland, Amsterdam

THE FAST MULTIPOLE METHOD FOR GRIDLESS PARTICLE SIMULATION

John AMBROSIANO

Berkeley Research Associates, Springfield, VA 22150, USA

Leslie GREENGARD and Vladimir ROKHLIN

Department of Computer Science, Yale University, New Haven, CT 06520, USA

The numerical solution to N-body problems in gravitation or electrostatics has traditionally been obtained via particle-in-cell methods (PIC) since direct evaluation of all pairwise interparticle forces, requiring $\mathcal{O}(N^2)$ operations, is too expensive. Recently, hierarchical solvers, which use tree data structures and lumped-force approximations, have made gridless simulations feasible in $\mathcal{O}(N \log N)$ operations. In this paper, we explore the use of the fast multipole method (FMM) – a highly accurate order $\mathcal{O}(N)$ algorithm – in particle simulations. We briefly describe the FMM and its relation to other methods. Technical considerations of gridless simulations such as discrete particle fluctuations, sampling errors and boundary conditions are discussed and compared with PIC methodology. Examples of electrostatic simulations in plasma physics are presented.

1. Computing long range forces for large systems of particles

There are many examples in classical physics where models based on large-scale ensembles of particles interacting via long-range forces are extremely useful. Astrophysics and plasma physics are two prominent examples. Molecular dynamics is another, and in fluid mechanics there are methods for solving the Navier–Stokes equations based on the interactions of point vortices.

Although computers have made the calculation of many-body problems feasible, those which include long-range forces still present a challenge. A straightforward application of the pairwise force law to a collection of N particles requires $\mathcal{O}(N^2)$ arithmetic operations. This kind of calculation is quite reasonable given a few hundred particles, but the computational cost grows so quickly that calculations with thousands of particles are impractical and those with millions unattainable. For example, on a VAX 8600 computer, the evaluation of Coulombic interactions required 0.15 cpu-seconds for 100 particles, 15 seconds for 1000 particles leading to an estimate of 174 days for 1 000 000 particles.

Until very recently, particle-in-cell (PIC) methods have been regarded as the only efficient way to compute the simultaneous interactions within large systems of particles. Instead of calculating forces directly, PIC methods superimpose a grid of sample points (usually fixed) on which to compute the potential field associated with the forces. The particles contribute their mass, charge, or other intrinsic properties to the grid to create source densities for the potentials, which are then found from the numerical solution of an elliptic partial differential equation. The total operation count for a system of size $N = N_p + N_g$, composed of N_p particles and N_g grid points, is roughly $\mathcal{O}(N)$ if a fast elliptic solver is used. In typical calculations $N_p \geqslant N_g$.

2. Pros and cons of gridded calculations

Computations using PIC methods have largely been successful, particularly in plasma physics. Not only are the methods efficient when compared with direct calculation, but they usually offer a definite advantage. If the ratio of particles to cells is suitable large, then fluctuations inevita-

0010-4655/88/$03.50 © Elsevier Science Publishers B.V.
(North-Holland Physics Publishing Division)

bly associated with statistically small numbers of particles are smoothed to a manageable level. This smoothing takes place on the scale of a grid cell which represents the immediate neighborhood for a typical particle and encompasses a small fraction of the total number of particles. Thus, PIC calculations tend to smooth away local interactions leaving behind collective or global effects.

The smoothing effects of PIC on local interactions account for much of their appeal (and their success) in applications to plasma kinetic theory where the collisionless approximation is assumed. These systems are traditionally considered uncorrelated; i.e. the two-body correlation function is assumed to vanish. Where the length scale of interest is on the order of the Debye length (or greater), PIC works well provided the grid can resolve the Debye length and the number of particles per "Debye cell" is large. In gravitational problems, where there is no Debye shielding the method is less appropriate. For gravitating systems, local correlations become important and PIC methods are not very satisfactory unless measures are taken to correct the far-field solutions to include local forces. This idea of adding back in the local force correction is central to the P^3M method (particle–particle, particle–mesh) [1], which is regarded as a suitable PIC approach to correlated systems. Aside from gravitational problems, when fluids are modeled by point vortices, their interactions are often not well represented by PIC methods. In plasma physics, too, applications of PIC to cold plasmas and non-neutral plasmas meet difficulties.

There are other liabilities of gridded calculations besides the sometimes unwanted smoothing of local forces. A grid imposes boundaries on the system of particles, whether such boundaries exist or not. Thus, the solution to a free-space problem cannot be obtained without computational overhead; for example, constructing a large grid to remove the boundaries far from the particles. Even in bounded problems, a grid can become a nuisance. Physical boundaries often do not fit the computational grid, making it necessary to adopt idealized boundary shapes that have only a passing resemblance to the actual shapes. Adding realism to boundary problems by using boundary-fitted curvilinear coordinates creates new (though surmountable) difficulties associated with the mapping. The finite element approach, which has been so successful in bounded continuum problems, takes on additional complexity when particles must be repeatedly reassigned to elements. For quadrilateral cells this adds approximately 15 percent to the cost of calculation [2]. The overhead in using an unstructured grid is unknown.

In addition to the problems associated with boundaries, a grid may not always sample the fields appropriately. If the grid coarse, fields may be undersampled, leading to aliasing errors (and possibly numerical instabilities). On a uniform grid, some portions of the calculation may be undersampled while others are not. In other circumstances features may arise from dynamics (e.g. shocks, fronts, vorticity layers) which are poorly resolved. Recent efforts to overcome these problems have included moving grids and other forms of adaptive grid refinement [3].

Finally, grid points add to the overall complexity of the computation without being central to the aim of the calculation. The overhead associated with the grid computation becomes more noticable as one goes from one dimensions to three. In one dimension it is possible to have many particles and adequate grid resolution without much overhead from the grid points; i.e. $N_g \ll N_p$. However, as one moves up to two and then three dimensions, one typically finds $N_g \sim N_p$.

The above considerations can lead to an interesting perspective on PIC methods. Grids, which were originally introduced as a computational convenience, can themselves become a difficulty if computational demands are high. When the main motivation is to follow particles as freely-moving Lagrangian phase points, one would prefer to do without a grid if the computational cost were not prohibitive.

3. Hierarchical solvers

Recently, some modern computer science techniques and some classical mathematical physics have combined to make gridless calculations with large numbers of particles feasible. These new methods have been called tree codes, hierarchical solvers and cluster methods. The most general technique, and the subject of this paper, is the fast multipole method (see ref. [4,10,13]).

The basic stratagem is to represent the force seen by a specific particle as the sum of forces from the individual particles in its immediate neighborhood plus coarser groupings of particles at longer range. This is also the idea behind the P^3M method, but here the goal is to perform the calculation without a mesh.

Trees are natural data structures in this context. In Appel's method for gravitating systems [5] a tree was constructed with particles as leaves and clusters of particles as the nodes of branches. Nearby particles were identified with the leaves of nearby branches. A similar method was implemented independently by Jernigan [6] and Porter [7]. The tree required continuous updating to avoid tangling as particles moved in space, while the complicated and arbitrary structure of the tree (what precisely is a "cluster?") made the errors in approximating lumps of particles as pseudo-particles difficult to analyze. Furthermore, the utility of the method for uniform distributions of particles was unclear. Nevertheless calculations could be performed with logarithmic rather than linear growth in the computational effort per particle.

Barnes and Hut [8] introduced a hierarchical $\mathcal{O}(N \log N)$ algorithm with a tree based on a partition of the space enclosing the particles rather than a cluster construction. In this approach, the entire space is identified with the root of the tree and then subdivided recursively into 2^n daughter cells, where n is the number of dimensions, until cells at the leaves contain only single particles. With this construction in place the evaluation of forces on a particle p begins by starting at the root of the tree and working toward the leaves. If the distance D from p to the center-of-mass of the cell is such that the $l/D < \theta$, where l is the cell dimension and θ is an error parameter, the cell's lumped center-of-mass contribution is added to the force on the particle. Otherwise its daughter cells are recursively examined for contributions to the force. The advantages of this approach are that a tree based on a spatial partition can easily be constructed at each timestep, requires no assumption about particle distribution and allows the error to be analyzed and estimated. An excellent review of hierarchical N-body methods has been given by Hernquist [9].

These approaches have two properties in common: they are $\mathcal{O}(N \log N)$ and they are approximation methods – in general less accurate than direct calculation.

4. The fast multipole method (FMM)

In refs. [4,10,13], an algorithm is presented for the evaluation of Coulombic interactions in a system of N particles which requires $\mathcal{O}(N)$ operations and is accurate to within round-off error. The method shares with the Barnes and Hut technique the same quad-tree structure (octal in three dimensions) to partition space as pictured in fig. 1. However, there are several significant differences.

The fast multipole method, as it is called, is also a divide-and-conquer approach, but is based on certain analytic observations concerning multipole expansions. The first such observation is that when two sets of particles are "well-separated", as shown in fig. 2, it is possible to determine the number of terms needed in the expansion to compute the forces of interaction to any desired accuracy. Moreover, the number of terms is independent of the number of particles. It depends only on the geometry. In the method, then, sets of

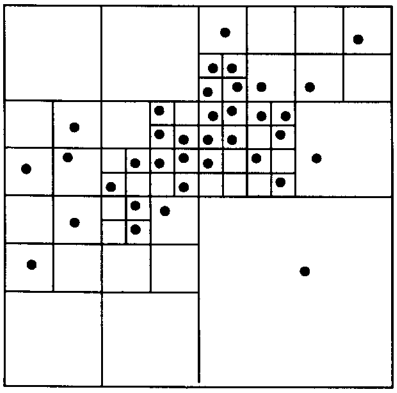

Fig. 1. A quad-tree structure constructed around an arbitrary distribution of particles.

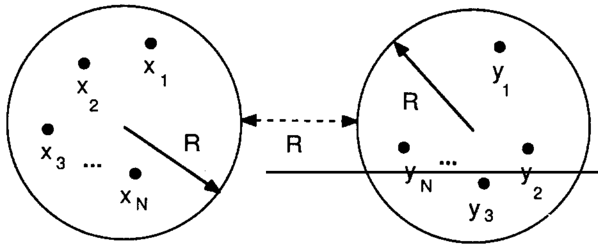

Fig. 2. Two "well-separated" sets of particles. For a given separation, the desired error tolerance determines the number of terms required in a multipole expansion of the potential.

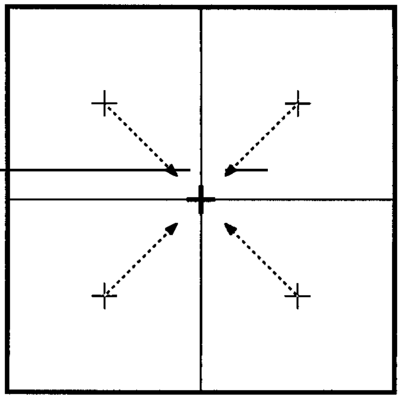

Fig. 3. A schematic representation of how shifting theorems allow expansions for parent cells to be constructed from those centered in the daughter cells.

particles are identified as cells in the quad-tree and "well-separated" is taken to mean "at least one cell away" at a fixed level of refinement. Given a separation distance and a specified error tolerance, the number of terms needed in the expansion is determined. More specifically, the number of terms k is related to the error tolerance ϵ by the expression

$$\frac{1}{2^k}\sum_i |q_i| \leqslant \epsilon, \qquad (1)$$

where q_i is the charge of particle i. At every level in the hierarchy, a k-term multipole expansion at the center of each nonempty cell is constructed. Forces are then evaluated recursively from the tree root to the leaves.

What distinguishes the FMM from other methods, aside from its careful attention to accuracy, is the way in which the expansions in the tree are constructed and the way the forces on individual particles are subsequently evaluated. The algorithm takes full advantage of potential theory, using theorems to shift the centers of multipole expansions, to convert multipole expansions into local expansions (Taylor series), and to shift the centers of local expansions. For example, in constructing the multipole expansions for all nonempty boxes at all levels of spatial refinement, the expansions are first calculated relative to the centers of cells at the finest level. The centers of the expansions are then shifted to combine moments from daughter cells to form expansions centered at parent cells as shown in fig. 3. A similar procedure is subsequently used to obtain the net forces for all particles. Rather than directly evaluating a multipole expansion at a target location, that expansion is converted into a local expansion about the center of the target cell. This local expansion is then shifted to each of the target cell's daughters and the process repeated at the level of the daughters. Contributions from cells that are well-separated are accounted for by converting their multipole expansions to local expansions and adding them to those obtained from the parent cells. Ultimately, at the leaves, which contain only a few particles each, the local expansions are evaluated to obtain the far field, while forces from neighboring particles are obtained by direct pairwise computation. Thus, the algorithm performs an upward pass to construct the multipole expansions and a downward pass to construct the local expansions and compute forces. The particles themselves are only accessed at the finest level. The results of this procedure is a computational effort that scales as N.

It is worth noting that the FMM is not an approximation method in the sense that forces may be evaluated to machine accuracy if desired. For example, on a 32-bit machine this requires twenty terms. For large numbers of particles the method is *more* accurate than the direct calculation which requires many more operations and is more susceptible to round-off error.

5. Gridless particle simulations

The fast multipole method provides an accurate means of performing gridless N-body calculations with reasonable computing resources. Given this new opportunity, it is worthwhile to explore the methodology for performing gridless simulations and to draw parallels, where possible, with PIC. This we will do within the context of plasma simulation.

There are three basic aspects of gridless calculations that differ from gridded ones. First, fluctuations from statistically small numbers of particles are not smoothed by the interpolation (shape) functions that weight particle masses or charges to the grid. Since there is no interpolation, another method is required. One may insteady modify the force law at close range to represent distributed particles shapes. Second, gridless simulations are truly Lagrangian in character; i.e. the physics is sampled on a set of moving nodes that follow the flow in phase space rather than on a set of fixed points. Therefore sampling requirements are not as clear. Finally, the methods of enforcing boundary conditions in gridless calculations are less obvious than in the gridded case. However, as we will discuss, the lack of a grid poses no serious problems for enforcement of boundary conditions. In fact there is much to suggest that FMM calculations may be *better* for bounded problems than PIC.

Fluctuations may be suppressed in gridless calculations by modifying the force law for close-range encounters. This is what happens in a PIC simulation. Interpolation of charge to a grid gives the particle a cloud-like shape, about a cell size, over which the charge is effectively distributed. Overlapping clouds feel a diminished force which vanishes when the clouds are concentric. Exactly the same strategem can be employed in gridless calculations, with the FMM providing a convenient framework. Since forces between immediate neighbors must be computed directly in the FMM (i.e. as binary encounters), the nature of the force law can be specified. In the two-dimensional calculations that follow, we let the force between particles be linear in their separation distance r when $r < \epsilon$, and Coulombic ($1/r$) when $r \geqslant \epsilon$. This avoids the buildup of entropy in collisionless plasma simulations, and provides a convenient parallel to gridded methods on which to base a comparison.

Aliasing is a form of error common to most computations. When a wave is undersampled, the physical picture of the wave that emerges from discrete calculation is a long wavelength alias of the shorter physical wave. In many cases, long wavelength aliases can feed back on the calculation to produce instabilities, some of which are self-limiting. In gridded calculations, the rule is that a physical wave must be sampled twice per wavelength by the grid to avoid aliasing. In gridless simulations, one does not expect aliasing errors of exactly the kind described above. Yet some sort of sampling theorem must apply to the finite system of particles even in the absence of a grid. A disturbance whose wavelength is shorter than the average interparticle spacing cannot be represented. Thus a similar notion to the one above for gridded methods would lead us to require that there be at least two particles per wavelength on average for a gridless calculation to resolve it. In one of our examples below we see a suggestion that errors in sampling the Debye length during a gridless simulation may contribute to a self-limiting heating instability whose exact cause is presently unknown. We also find that the level of spurious heating in the gridless case is much more benign than in a comparable PIC simulation.

The enforcement of boundary conditions would at first seem difficult in a gridless calculation. Indeed, the method is ideally suited for unbounded problems such as those that arise in astrophysics. However, the FMM, because of its basis in multipole expansions, provides a powerful framework for including boundary effects. There are two basic approaches to boundaries in the FMM: images and boundary integrals.

The familiar method of images enforces Dirichlet or Neumann conditions at a boundary surface by suitable replacement of the surface with image particles. Such a method would be very costly in a direct calculation because at a plane surface, for example, N particles would require N images, increasing the effort by a factor of four. If we added an opposing boundary plane

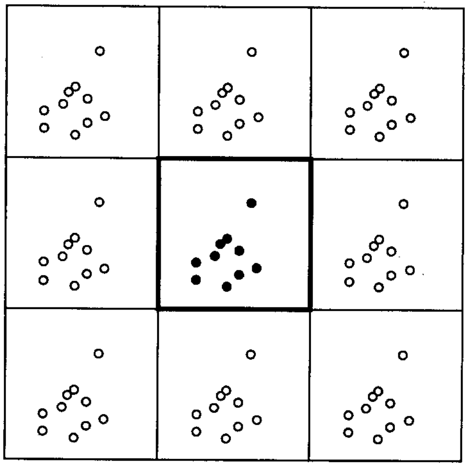

Fig. 4. A system of particles and its periodic images.

(e.g., two capacitor plates), the infinity of images, even if truncated to a finite number for computation, would be disastrous. However, in the FMM, the root of the tree contains a multipole representation of the entire assembly of particles which can be shifted to other locations. Consider the case of doubly periodic boundaries in two dimensions. Periodicity in the fields requires an infinite pattern of periodically repeated image boxes, as shown in fig. 4. Given the k-term multipole expansions Φ for the isolated system, it is possible (see ref. [4] for details) to construct a k by k matrix T, such that $T \cdot \Phi$ is a k-term local expansion describing the field due to the infinite number of well-separated image boxes. This procedure adds very little to the cost of performing the computation for the isolated system. It is also well to note that obtaining the periodic solution this way (by analytic summation of the images) is more accurate than the results of a discrete Fourier transform (DFT) on a grid.

There is another way to enforce boundary conditions using the FMM that is well-suited to complicated domains. In fact, in ref. [11], one of the authors (V.R.) had originally formulated much of the basic theoretical framework of the FMM to solve elliptic boundary value problems in complicated regions.

It is well known from classical potential theory that boundary value problems for the Laplace equation can be reduced to second kind integral equations. For example, to solve the Dirichlet problem

$$\nabla^2 \phi = 0 \tag{2}$$

within a region Ω such that

$$\phi(x) = f(x) \tag{3}$$

on the boundary surface $\partial\Omega$, we construct a solution of the form

$$\phi(x) = \oint_{\partial\Omega} \phi_D(x, t) \sigma(t) \, \mathrm{d}t, \tag{4}$$

where $\phi_D(x, t)$ is the potential at x due to a normally oriented dipole of unit strength located at a boundary point t. To satisfy the boundary condition, the unknown dipole density $\sigma(t)$ must satisfy

$$\pi\sigma(t) + \oint_{\partial\Omega} \phi_D(t, t')\sigma(t') \, \mathrm{d}t' = f(t). \tag{5}$$

For purposes of computation, the boundary surface can be discretized and the resulting linear system solved by iteration. Each iteration can be performed rapidly using the FMM to evaluate the potentials of N dipoles at N locations.

The appeal of boundary methods is that the dimensional complexity of the problem is reduced; e.g. two-dimensional problems become one-dimensional. Very complicated boundaries can be represented by simply distributing points along the boundary surface, and exterior as well as interior problems can be solved without a grid.

The boundary integral method and the N-body calculation can be combined into a very powerful technique – one with the potential to handle bounded particle simulations of extraordinary complexity.

6. Examples

We present some examples of electrostatic simulations in two dimensions. The first two are intended to demonstrate that application of the FMM to problems traditionally solved by PIC methods can produce excellent results. The last example is a study of spontaneous heating in a cold plasma.

Our first example is that of an electron beam emitted near a grounded conducting plane. A number of particles are loaded every timestep at $x = 0$ with a beam velocity v_{x0} in such a way as to simulate a beam of uniform density. To enforce the boundary condition on the conducting plane at $x = 0$, we use the simple artifice of having one positive image particle at a symmetric position for every electron. Fig. 5 shows the result after the beam has propagated some distance. Since there are no neutralizing ions, the beam simply spreads from the electrostatic repulsion. The head of the beam spreads less because it feels no forces from the right whereas the center of the beam experiences repulsion from all sides. The high degree of symmetry and regularity evident in the figure suggests a high degree of accuracy. In fact, small perceived irregularities were later found to be the result of poor resolution by the laser printer.

The results of the second example are more quantitative. The problem is a standard one in plasma physics. We simulate an instability that occurs when electrons stream through ions. Periodic boundary conditions are enforced as described in section 5. Drifting electrons are loaded atop ions ($m_i/m_e = 10$) initially at rest on a regular lattice totaling 5000 particles altogether. Both species are initially cold. The evolution of the instability can be seen very well in the pair of phase-space plots (v_x versus x) of fig. 6. In fig. 6a, we can see a wave growing whose wavelength agrees very well with theory. Fig. 6b shows the trapping of electrons and ions in the wave as the instability saturates. Another picture of the simulation is shown in fig. 7. Having sampled the electric field at a large number of points in the x-direction, we take the Fourier transform and plot the spectrum of waves versus time. The wavenumber associated with the interparticle separation is indicated by an arrow. In this plot we can see the unstable wavenumber grow quite clearly and then decay as trapping distributes the energy to other modes. Notice the spectrum is very clean and without the sort of noise that one might expect from a gridless calculation. We have used the linear force law described in section 5 within a radius $\epsilon = 1.5\delta$ where δ is the average interparticle spacing.

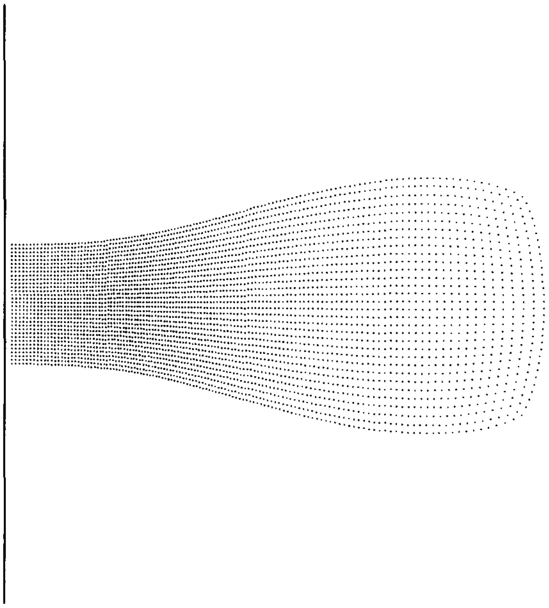

Fig. 5. A beam of electrons propagating to the right of a grounded conducting plane.

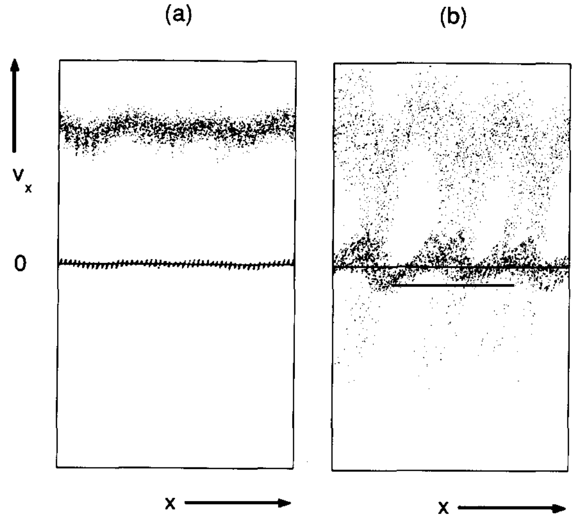

Fig. 6. Phase space, v_x versus x, showing electrons and ions plotted together during an electron-ion two-stream instability. (a) The unstable wave is seen growing; (b) electrons and ions are trapped by the wave as the instability saturates.

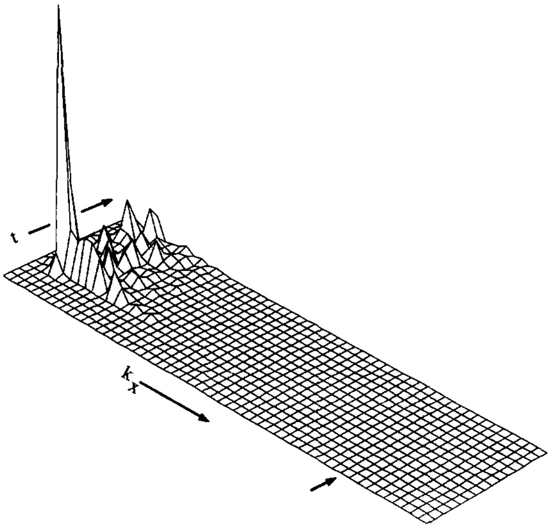

Fig. 7. Evolution of the waves spectrum along the x-direction during simulation of a two-stream instability. The arrow represents the wavenumber associated with the average interparticle spacing.

Finally, we consider the phenomenon of spontaneous heating. In PIC simulations, the phenomenon falls into one of two catagories: secular buildup of entropy from fluctuations tracable to finite timesteps and small numbers of particles; and a self-limiting instability, usually called the finite grid instability, due to undersampling of the Debye length. We have discovered that heating of the first kind occurs for the same reason and has the same remedy as in PIC – namely more accurate time integration and the use of distributed charges or clouds rather than point particles. Details of the study are deferred to another paper. The second type of heating (i.e. due to undersampling on the Debye scale) also seems to occur in gridless simulations, but appears to be much more benign pointing to a substantial advantage of the gridless method.

In our final example, we initialized a number of non-drifting ions and electrons with a very small temperature and allowed them to interact. We chose the cloud radius ϵ to give an average number of particles per cloud of about nine. From this we can infer the size of the cells in a comparable gridded calculation.

The amount of numerical heating expected from the finite grid instability is well known and theoretically understood [12]. The instability results from undersampling the Debye length $\lambda_D = v_{th}/\omega_p$, where v_{th} is the thermal speed and ω_p the plasma frequency. Heating stops when the Debye length can be seen by the grid. Thus the instability saturates when

$$v_{th} = \omega_p \Delta/\pi, \tag{6}$$

where Δ is the grid spacing. Fig. 8 shows the numerical heating (increase in kinetic energy) of the gridless simulation versus time. The kinetic energy is seen to rise and then quickly saturate. An estimate of the thermal speed at saturation for a comparable PIC simulation is also provided. Note that this level is about 125 times greater than the value obtained in the gridless simulation. The origin of spurious heating in the gridless simulation is unknown although there is reason to believe it is related to errors in sampling the Debye

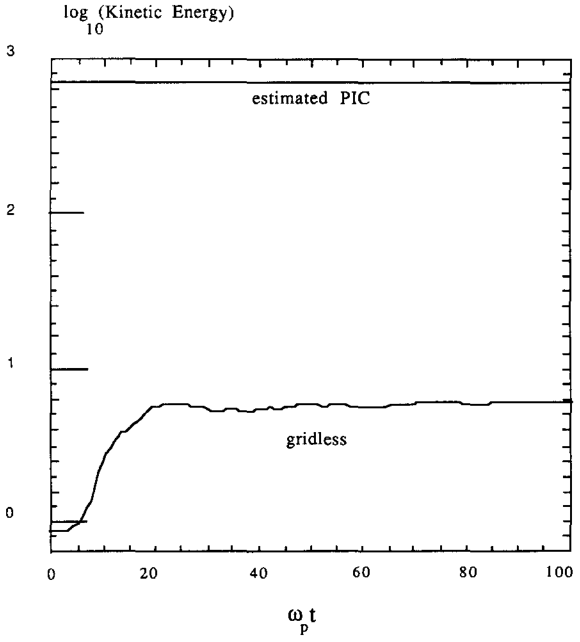

Fig. 8. Log of total kinetic energy versus time for an initially cold plasma. The apparent heating instability grows, but quickly saturates. Above is a numerical heating estimate (from eq. (6)) for a comparable PIC simulation.

length as in the PIC case. There are two aspects of the phenomenon that lend credence to this speculation. The first is that the heating saturates instead of rising secularly. The second is that the increase in kinetic energy per particle is less as one decreases the average interparticle spacing. This test case using a small number of particles is only representative. We expect that realistic simulations of cold plasmas using greater numbers of particles would heat even less. However, the above results based on estimates of comparable PIC simulations are not definitive. Direct, detailed comparisons are needed. Such a study is in progress.

7. Remarks

We have attempted to give an idea of the new opportunities available in N-body simulations – and in particular in plasma simulations – afforded by the fast multipole method. Particle-in-cell methods are, of course, extraordinarily flexible and useful techniques whose methodology is quite mature. However, hierarchical solvers, and in particular the FMM, provide a new and powerful option for many difficult problems and one that promises to grow in capability.

While presently available computer codes deal with two-dimensional problems, the three-dimensional theory has been worked out [13] and will be implemented soon. Astrophysical and fluid mechanical problems are ready candidates for application. Plasma physics applications are likely to be concentrated in the areas of cold plasmas and beams, and to plasmas in complicated regions.

There are several applications of the FMM underway. A direct implicit electrostatic method is being developed. Electromagnetic simulations using the Darwin approximation should also be possible.

Finally, the current interest in parallel computation is likely to stimulate further work with these algorithms. The formation and manipulation of expansions in the FMM are intrinsically independent procedures, and well-suited to parallel architectures.

References

[1] R. Hockney and J. Eastwood, Computer Simulation Using Particles (McGraw-Hill, New York, 1981) p. 22.

[2] J.U. Brackbill, private communication.

[3] J.U. Brackbill and H.M. Ruppel. J. Comput. Phys. 65 (1986) 314.

[4] L. Greengard and V. Rokhlin, A Fast algorithm for Particle Simulation, Yale Univ. report YALEU/DCS/RR-459 (1986); also to appear in J. Comput. Phys.

[5] A. Appel, SIAM J. Sci. Statist. Comput. 6 (1985) 85.

[6] J. Jernigan, I. A. U. Symp. 113 (1985) 275.

[7] D. Porter, thesis, Physics Dept. Univ. of California, Berkeley (1985).

[8] J. Barnes and P. Hut, Nature 324 (1986) 446.

[9] L. Hernquist, Comput. Phys. Commun. 48 (1988) 107.

[10] J. Carrier, L. Greengard and V. Rokhlin, A Fast Adaptive Algorithm for Particle Simulation, Yale Univ. report YALEU/DCS/RR-496 (1987).

[11] V. Rokhlin, J. Comput. Phys. 60 (1985) 187.

[12] C.K. Birdsall and A.B. Langdon, Plasma Physics via Computer Simulation (McGraw-Hill, New York, 1985) p. 179.

[13] L. Greengard and V. Rokhlin, Rapid Evaluation of Potential Fields in Three Dimensions, Yale Univ. report YALEU/DCS/RR-515 (1987).