Fast Algorithms for Classical Physics

Leslie Greengard

Some of the recently developed fast summation methods that have arisen in scientific computing are described. These methods require an amount of work proportional to N or $N \log N$ to evaluate all pairwise interactions in an ensemble of N particles. Traditional methods, by contrast, require an amount of work proportional to N^2. As a result, large-scale simulations can be carried out using only modest computer resources. In combination with supercomputers, it is possible to address questions that were previously out of reach. Problems from diffusion, gravitation, and wave propagation are considered.

Many problems in scientific computing require the calculation of all pairwise interactions in large ensembles of particles. A classical example is the N-body problem of gravitation. Given a configuration of N point masses m_i located at the points $\mathbf{x}_i$ in three-dimensional space, the gravitational field experienced by any one of them is obtained from Newton's law

$$\mathbf{E}(\mathbf{x}_j) = -\sum_{\substack{i=1 \\ i \neq j}}^{N} m_i \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|^3} \qquad (1)$$

where $|\mathbf{x}_i - \mathbf{x}_j|$ denotes the Euclidean distance between the points $\mathbf{x}_i$ and $\mathbf{x}_j$. The same equation, of course, governs the electrostatic interaction of charged particles according to Coulomb's law, which plays a fundamental role in a host of physical and chemical processes. The main difference is that mass is always of one sign, whereas charge can be positive or negative. In many situations, more complicated interactions are needed to describe the governing forces. In molecular dynamics and protein folding calculations, for example, forces acting on each atom may include factors such as bond vibration, structural deformation, and van der Waals attraction and repulsion (*1*). Nevertheless, the dominant force acting at significant distances from the atomic center is the Coulomb force.

Sums such as Eq. 1 also arise in continuum physics, in the approximation of the field attributable to continuous distributions of mass or charge, as

$$\mathbf{E}(\mathbf{x}) = -\int m(\mathbf{y}) \frac{\mathbf{x} - \mathbf{y}}{|\mathbf{x} - \mathbf{y}|^3} d\mathbf{y} \qquad (2)$$

A similar integral arises in magnetostatics, in the expression of the magnetic induction $\mathbf{B}$ attributable to a steady-state current density $\mathbf{J}$. This is the Biot-Savart law (*2*)

$$\mathbf{B}(\mathbf{x}) = \frac{1}{c}\int \mathbf{J}(\mathbf{y}) \times \frac{\mathbf{x} - \mathbf{y}}{|\mathbf{x} - \mathbf{y}|^3} d\mathbf{y} \qquad (3)$$

where c is the speed of light.

A rather different kind of interaction arises in problems of heat conduction or diffusion. Suppose that the temperature profile in space is given at time $t = 0$ by the function $W(\mathbf{x})$. Then, in a homogeneous isotropic medium with a diffusion constant of unity (*3*), the temperature at time $t > 0$ is given by

$$U(\mathbf{x}) = \frac{1}{\sqrt{(4\pi t)^{3/2}}}\int e^{|\mathbf{x}-\mathbf{y}|^2/4t} W(\mathbf{y}) d\mathbf{y} \qquad (4)$$

A discrete analog of this is the sum

$$U(\mathbf{x}) = \frac{1}{\sqrt{(4\pi t)^{3/2}}} \sum_{i=1}^{N} W_i e^{|\mathbf{x}-\mathbf{x}_i|^2/4t} \qquad (5)$$

If we want to evaluate the temperature field U at each of the N locations $\mathbf{x}_j$, we have a kind of N-body problem. As a final example, consider the acoustic field generated by a collection of pulsating sources of strength W_i. The (complex-valued) pressure is then given by a sum of the form

$$\phi(\mathbf{x}) = \sum_{i=1}^{N} W_i \frac{e^{ik|\mathbf{x}-\mathbf{x}_i|}}{|\mathbf{x} - \mathbf{x}_i|} \qquad (6)$$

where k is the ratio of the frequency of the oscillation to the sound speed in the medium. The intensity of the sound wave is proportional to $|\phi|^2$.

The scientific importance of such problems is clear. Unfortunately, our theoretical understanding of the detailed structure of interacting particle systems has been rather limited. With the advent of high-speed computers, however, we have the ability to simulate particle systems and study their behavior in detail. The most expensive part of such a simulation is the evaluation of the force at each source position, which requires an amount of work proportional to N^2, where N is the number of particles. This growth in cost has been a major limitation on the size of the systems that can be investigated, even on modern supercomputers. Many problems of practical interest require millions of particles and have been well beyond reach. However, recently developed methods based on a combination of ideas from mathematical physics and computer science have substantially reduced the amount of work required.

Fast Summation

Each of the preceding examples involves the evaluation of an integral of the form

$$U(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{y}) W(\mathbf{y}) d\mathbf{y} \qquad (7)$$

or a sum of the form

$$U(\mathbf{x}) = \sum_{i=1}^{N} W_i K(\mathbf{x}, \mathbf{y}_i) \qquad (8)$$

Straightforward computation of Eqs. 1, 5, 6, or, more generally, 8 is referred to as direct summation. Algorithms that reduce the cost of evaluating these sums at each of N target locations from $O(N^2)$ operations to, say, $O(N^{3/2})$, $O(N \log N)$, or $O(N)$ will be referred to as fast summation (*4*).

The first breakthrough in this field came about 20 years ago with the development of particle-mesh methods, which allowed simulations of gravitational and Coulombic systems involving orders of magnitude more particles than previously possible. These methods are discussed in great detail by Hockney and Eastwood (*5*). Recent improvements have been made by Anderson (*6*), Almgren, Buttke, and Colella (*7*), and Couchman (*8*). Other useful methods include the multilevel approximation schemes of Brandt and Lubrecht (*9*, *10*) and the "wavelet"-based scheme of Beylkin, Coifman, and Rokhlin (*11*). I will limit this discussion, however, to a presentation of the ideas underlying yet another approach, based on multipole expansions, which was developed independently in the astrophysics (*12*, *13*), fluid dynamics (*14*), number theory (*15*), and potential theory (*16*, *17*) communities.

Beginning with a simple although physically uninteresting example, consider the calculation of

$$U(x_j) = \sum_{i=1}^{N} W_i (x_j - y_i)^2 \qquad (9)$$

for $j = 1, \ldots, N$, where the points in the sets $\{x_j\}$ and $\{y_i\}$ lie on the real line. Direct summation then requires $O(N^2)$ operations. On the other hand, we know that

The author is at the Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, USA.

$$(x_j - y_i)^2 = x_j^2 - 2x_j y_i + y_i^2 \qquad (10)$$

Substitution into Eq. 9 yields

$$U(x_j) = A_0 x_j^2 - 2A_1 x_j + A_2 \qquad (11)$$

where

$$A_k = \sum_{i=1}^{N} W_i y_i^k$$

Precomputation of the moments $\{A_k\}$ requires only $O(N)$ work, as does the subsequent calculation of U at each of the N target points with Eq. 11. It is easy to generalize this observation to summations such as Eq. 8 where the kernel of the interaction $K(\mathbf{x}, \mathbf{y})$ can be expressed as a finite series

$$K(\mathbf{x},\mathbf{y}) = \sum_{k=1}^{p} \phi_k(\mathbf{x})\psi_k(\mathbf{y}) \qquad (12)$$

A fast algorithm is obtained by the following two-step procedure. First, compute the moments A_k defined by

$$A_k = \sum_{i=1}^{N} W_i \psi_k(\mathbf{y}_i)$$

Second, evaluate $U(\mathbf{x})$ at each target position with

$$U(\mathbf{x}) = \sum_{k=1}^{p} A_k \phi_k(\mathbf{x})$$

The amount of work required is $O(Np)$ rather than $O(N^2)$. In mathematics, a kernel $K(\mathbf{x}, \mathbf{y})$ that can be expressed as a finite sum as in Eq. 12 is known as a degenerate kernel. Although the kernels of interest in physical applications are generally not of this type, the degenerate case serves as a useful model. Note that, no matter what the source distribution looks like, $U(\mathbf{x})$ is nothing more than a linear combination of the functions $\{\phi_1(\mathbf{x}), \ldots, \phi_p(\mathbf{x})\}$. Thus, there is a tremendous loss of information in the transformation from the data $\{\mathbf{y}_i, W_i\}$ to the target function U. The theme of encapsulating information in terms of moments in order to reduce computational cost recurs throughout this paper.

The far field. One of the reasons for the specific mention of the three summation problems (Eqs. 1, 5, and 6) is that each is representative of a broader class of interactions that influence the kind of fast algorithm that is appropriate. The difference in behavior between the three is demonstrated graphically in Fig. 1. Starting with a collection of points placed at random inside a unit disk in the xy plane (Fig. 1A), Fig. 1B shows the temperature field $U(\mathbf{x})$ in the xy plane for $t = 0.1$, assuming that at each point is a heat source assigned a random strength in the range $[-0.5, 0.5]$. It is obviously a globally smooth function. Assuming now that at each point is a randomly assigned mass in the range [0, 1], Fig. 1C shows the gravitational potential

$$\Phi(\mathbf{x}) = -\sum_{i=1}^{N} \frac{m_i}{|\mathbf{x} - \mathbf{x}_i|} \qquad (13)$$

induced in the xy plane, but only at a distance at least one disk diameter away from the center of the source distribution. Note that the gravitational field (Eq. 1) is just the gradient of this potential function. If we were to plot the potential in the immediate vicinity of the sources, the potential surface would look enormously more complicated. The far field, however, is smooth. Finally, Fig. 1D shows the real part of the complex-valued pressure field induced by acoustic sources of random strength with frequency $k = 5$. Despite the fact that we have stepped away from the sources, the far field has a noticeable structure on the scale of the wavelength $(1/k)$.

There is a distinct fast algorithm associated with each of these cases. We will consider them in order of mathematical complexity.

Smooth Interactions

Given a set of points $\mathbf{x}_i = (x_i, y_i, z_i)$ and a set of source strengths W_i, the function

$$U(\mathbf{x}) = \sum_{i=1}^{N} W_i e^{-|\mathbf{x}-\mathbf{x}_i|^2/4t} \qquad (14)$$

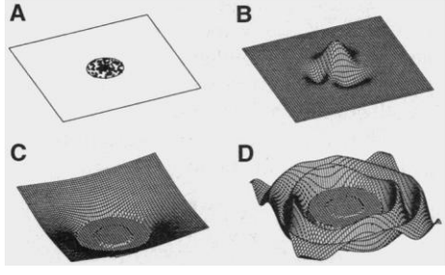

Fig. 1. We consider three different functions $U(\mathbf{x})$ of the form described by Eq. 8. (**A**) The source distribution $\{\mathbf{x}_i\}$ for each case. The unit sphere enclosing the sources is referred to as the source sphere. (**B**) When the interaction is governed by a diffusion process, the function $U(\mathbf{x})$, given by Eq. 5 with $t = 0.1$, is globally smooth. (**C**) When the interaction is gravitational, the function $U(\mathbf{x})$ is smooth in the far field. We plot the gravitational potential induced by the sources, but only at a sufficient distance from the source sphere, whose boundary is raised slightly for identification. (**D**) When the interaction is that of an acoustic field, the function $U(\mathbf{x})$ is highly structured even at a distance from the sources. The spatial scale of the oscillation is determined by the parameter k in Eq. 6. Here, $k = 5$.

is referred to as the (discrete) Gauss transform (*18*). The reason that the temperature function in Fig. 1B looks so smooth is that U is infinitely differentiable in each coordinate direction for $t > 0$. Furthermore, the derivatives are rapidly decaying. A more precise statement requires the introduction of some notation. In particular, we will use the Hermite functions $h_n(x)$, defined by

$$h_n(x) = (-1)^n \frac{d}{dx}(e^{-x^2})$$

Suppose now that a source $\mathbf{x}_i$ is located in a box with center $\mathbf{c} = (c_1, c_2, c_3)$ and side length $\sqrt{t}$. Then the heat kernel

$$e^{-|\mathbf{x}-\mathbf{x}_i|^2/4t}$$

can be expressed as a Hermite series

$$e^{-|\mathbf{x}-\mathbf{x}_i|^2/4t} = \sum_{n_1,n_2,n_3=0}^{\infty} \Phi_{n_1n_2n_3}(\mathbf{x})\Psi_{n_1n_2n_3}(\mathbf{x}_i) \qquad (15)$$

where

$$\Psi_{n_1n_2n_3}(\mathbf{x}_i) = \frac{1}{n_1!n_2!n_3!}\left(\frac{x_i - c_1}{\sqrt{4t}}\right)^{n_1}\left(\frac{y_i - c_2}{\sqrt{4t}}\right)^{n_2}\left(\frac{z_i - c_3}{\sqrt{4t}}\right)^{n_3}$$

and, for $\mathbf{x} = (x, y, z)$ (*18*)

$$\Phi_{n_1n_2n_3}(\mathbf{x}) = h_{n_1}\left(\frac{x - c_1}{\sqrt{4t}}\right) h_{n_2}\left(\frac{y - c_2}{\sqrt{4t}}\right) h_{n_3}\left(\frac{z - c_3}{\sqrt{4t}}\right)$$

One can also derive a precise estimate of the error that results from truncating the series in Eq. 15 after p terms

$$\left| e^{-|\mathbf{x}-\mathbf{x}_i|^2/4t} - \sum_{n_1,n_2,n_3=0}^{p} \Phi_{n_1n_2n_3}(\mathbf{x})\Psi_{n_1n_2n_3}(\mathbf{x}_i) \right| \le \left(\frac{1}{p!}\right)^{3/2}\left(\frac{1}{2}\right)^{2p} \qquad (16)$$

The expansion of Eq. 15 together with its error bound (Eq. 16) is the only analytical tool needed to construct a fast Gauss transform. Observe that the heat kernel is very nearly degenerate; that is, one can easily choose p so that the heat kernel is approximated by a finite series with any desired precision. The case $p = 4$ already causes the error to be less than 0.01%. It remains only to organize the computation so that the finite series can be used effectively.

For simplicity, let us assume that the sources and targets for the discrete Gauss transform (Eq. 14) lie in a cube B_0 of unit volume, and let us begin by subdividing B_0 uniformly into smaller boxes of side length $\sqrt{t}$ (if $t \ge 1$, B_0 needs no further refinement). Choose p sufficiently large that the

heat kernel is approximated to the desired precision. For each small box B with center c, compute the moments

$$A_{n_1n_2n_3} = \sum_{\mathbf{x}_j \in B} W_j \Psi n_1 n_2 n_3(\mathbf{x}_j)$$

for $n_1,n_2,n_3 \leq p$. Because each source contributes to exactly one expansion, the amount of work required to form the moments for all nonempty boxes is proportional to Np^3. Consider now a target point $\mathbf{x}$ that lies, say, in box C. Because the heat kernel decays exponentially fast in space, we need only consider the influence of the near neighbors in our box decomposition of B_0. Indeed, ignoring all but the nearest $(2n + 1)^3$ boxes, we incur an error of the order $e^{-n^2/4}$. For $n = 6$, this is approximately 10^{-4}. The influence of the sources contained in each of these nearby boxes can be obtained by evaluating the expression

$$\sum_{n_1,n_2,n_3=0}^{p} A_{n_1n_2n_3} \Phi_{n_1n_2n_3}(\mathbf{x})$$

where $\{A_{n_1n_2n_3}\}$ are the moments precomputed for that box. The amount of work required is of the order $(2n + 1)^3p^3$ for each target, which is a constant. The total amount of work, therefore, is proportional to $N + M$, where N is the number of sources and M is the number of targets. For a more complex but more efficient version of the method, we refer the reader to the original paper by Greengard and Strain (*18*). For 100,000 sources and targets in two space dimensions, the algorithm is about three orders of magnitude faster than direct summation.

Let us next consider two extreme regimes. First, suppose that $t = 10^{-10}$. Then the effect of each heat source is vanishingly small beyond a distance of about 10^{-5}. The question of fast computation of the sum Eq. 14 becomes substantially one of sorting and finding near neighbors. Second, suppose that $t = 1$. Then all sources have nonnegligible interactions, but only one box is constructed, one set of moments is computed, and one expansion is evaluated for each target. As time marches forward, there is less and less information content in the temperature field $U(\mathbf{x})$. This is to be expected from a diffusion process.

Tree Codes

The evaluation of gravitational or Coulombic interactions requires a different set of tools for organization of the computation. The force is long-ranged and not globally smooth. Working in the astrophysics community, Appel (*12*), Barnes and Hut (*13*), and others developed what have come to be known as "tree codes" to overcome the computational obstacle presented by the N-body problem. They are based on the observation that although the gravitational field may have a complex local structure, the far field is smooth. In a tree code, a cluster of particles is replaced by some simpler representation, which is used to compute the influence of the cluster at sufficiently great distances. Although there is generally some sacrifice in accuracy, the amount of time required to compute all interactions is proportional to $N \log N$ rather than N^2 (*19*).

Before discussing the organizational aspects of a tree code, let me make a precise statement about the approximation of the far field. I will consider the potential energy function (Eq. 13) rather than the gravitational field (Eq. 1) because it is slightly less cumbersome to deal with. I again begin with some notation:

Let (r, θ, ϕ) be the spherical coordinates of a point with Cartesian coordinates (x_1, x_2, x_3). That is

$$r = \sqrt{x_1^2 + x_2^2 + x_3^2}$$
$$\theta = \cos^{-1}(x_3/r)$$
$$\phi = \tan^{-1}(x_2/x_1)$$

The potential energy can be expanded in terms of special functions known as spherical harmonics, which are characterized as (*20*)

$$\frac{Y_n^0(\theta, \phi)}{r^{n+1}} = A_n^0 \frac{\partial^n}{\partial z^n}\left(\frac{1}{r}\right) \quad (17)$$

For $m > 0$

$$\frac{Y_n^m(\theta, \phi)}{r^{n+1}} = A_n^m \left(\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right)^m \left(\frac{\partial}{\partial z}\right)^{n-m}\left(\frac{1}{r}\right) \quad (18)$$

and

$$\frac{Y_n^{-m}(\theta, \phi)}{r^{n+1}} = A_n^m \left(\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right)^m \left(\frac{\partial}{\partial z}\right)^{n-m}\left(\frac{1}{r}\right) \quad (19)$$

where

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}} \quad (20)$$

Suppose now that a collection of k source is contained inside a sphere of radius a. Let m_i denote the strength of the ith source, and let $Q_i = (\rho_i, \alpha_i, \beta_i)$ denote its position in spherical coordinates. Then for any $\mathbf{x} = (r, \theta, \phi)$ with $r > a$, the potential $\phi(\mathbf{x})$ is given by its multipole expansion (*2*, *20*)

$$\phi(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta, \phi) \quad (21)$$

where

$$M_n^m = \sum_{i=1}^{k} m_i \rho_i^n Y_n^{-m}(\alpha_i, \beta_i) \quad (22)$$

Furthermore, for any $p \geq 1$,

$$\left| \phi(\mathbf{x}) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta, \phi) \right| \leq \frac{A}{r - a}\left(\frac{a}{r}\right)^{p+1} \quad (23)$$

where

$$A = \sum_{i=1}^{k} |m_i| \quad (24)$$

From the error bound Eq. 23, it is clear that there are two ways to improve accuracy. One is to increase p, referred to as the degree of the expansion, until some desired degree of precision is achieved. The other is to fix p and insist that the ratio a/r be sufficiently small. Observe that we have achieved a tremendous amount of data compression in the multipole expansion. At a distance $2a$ from the center of the sphere, setting $p = 20$ provides six digits of accuracy in the evaluation of the potential.

The nonadaptive scheme. If the source distribution is essentially uniform in a cube, a fairly simple nonadaptive algorithm can be constructed. We need to introduce a hierarchy of boxes that refine the domain into smaller and smaller regions. Level 0 will refer to the entire box, and level $l + 1$ is obtained by subdividing each box at level l into eight equal parts, which are considered its children. The number of distinct boxes at level l is then 8^l. Within this data structure, boxes at the same level of refinement that share a boundary point are said to be near neighbors. Boxes that are at the same refinement level but are not near neighbors are said to be well separated. Finally, with each box i is associated an interaction list consisting of the children of the near neighbors of i's parent that are well separated from i itself (Fig. 2).

The idea of the algorithm is to use the box hierarchy to cluster particles at finer and finer spatial scales and to compute interactions between distant clusters by means of multipole expansions. At levels 0 and 1, there are no pairs of boxes that are well separated. At level 2, on the other hand, 64 boxes have been created, and there are a number of well-separated pairs. Suppose now that, for each box i at level 2, we create the multipole expansion about

the box center induced by the contained sources. We can then use these multipole expansions to compute the interactions between the particles in all well-separated pairs of boxes. After all such well-separated interactions have been accounted for, it remains only to compute the interactions between particles contained in each level 2 box with those contained in the box's near neighbors. The algorithm does this recursively. Consider, for example, a level 3 box, and let us seek to determine which other boxes at that level should be interacted with by means of multipole expansions. Those boxes outside the region of the parent's near neighbors are already accounted for (at level 2), and interactions with near neighbors cannot accurately be computed by means of an expansion. The remaining boxes correspond exactly to the interaction list defined above. The nature of the recursion is now clear. For each box at each finer level, a multipole expansion of degree p is formed, expressing the far field induced by the sources it contains. This expansion is then evaluated for each particle in the region covered by its interaction list. We halt the recursive process after log N levels of refinement.

The amount of work done at each level is of the order $O(N)$. To see this, note first that approximately p^2N operations are needed to create all expansions, because each particle contributes to exactly p^2 expansion coefficients. Second, from the point of view of a single particle, there are at most 189 boxes (the maximum size of the interaction list) whose expansions are evaluated, so $189p^2N$ operations are needed for all evaluations.

At the finest level, we have created $8^{\log_8 N} = N$ boxes, and it remains only to compute interactions between nearest neighbors. Because we have assumed that the distribution is homogeneous, there is, on the average, one particle per box, so that this last step requires about $27N$ operations. The dominant cost is the formation and evaluation of expansions at each level, which is proportional to N log N. In the Barnes-Hut algorithm, p is chosen to be 1, 2, or 3. If greater precision is desired, the distance from a multipole expansion center to an evaluation point must be increased in accordance with the error bound Eq. 23. This requires a change in the definitions of interaction list and near neighbor (*21*). In the algorithm described above, however, one chooses p to yield any prescribed accuracy, again in accordance with Eq. 23. Such a scheme was proposed (in two space dimensions) by Van Dommelen and Rundensteiner (*14*).

The adaptive algorithm. When the distribution of particles is nonuniform, as in most astrophysical problems, it is clear that a somewhat different strategy must be used. During the refinement process, each box is examined to determine whether it actually contains any particles. If so, it is subdivided further. If not, it is pruned from the tree structure and ignored at subsequent levels. The cost of this adaptive algorithm is more difficult to state precisely because it depends on the total number of refinement levels, which is not determined a priori. In most cases of practical interest, this turns out to be proportional to log N. For example, if the interparticle spacing collapses as N^{-s}, where s is independent of N, then s log N levels are needed. It is, therefore, quite reasonable to refer to the adaptive algorithm as also being of the order N log N.

The Fast Multipole Method

The fast multipole method (FMM) (*16, 17, 22–25*) can be viewed as a tree code, but one which makes use of several further an-

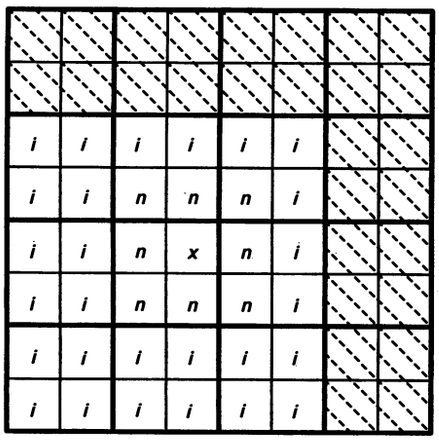

Fig. 2. The near neighbors and interaction list of a box in two space dimensions. The thick lines correspond to level 2, and thin lines correspond to level 3. For the level 3 box marked by *x*, the nearest neighbors are indicated by the letter *n*, and the interaction list by the letter *i*. Interactions between sources in the dashed region and those in the box *x* have already been computed at level 2.

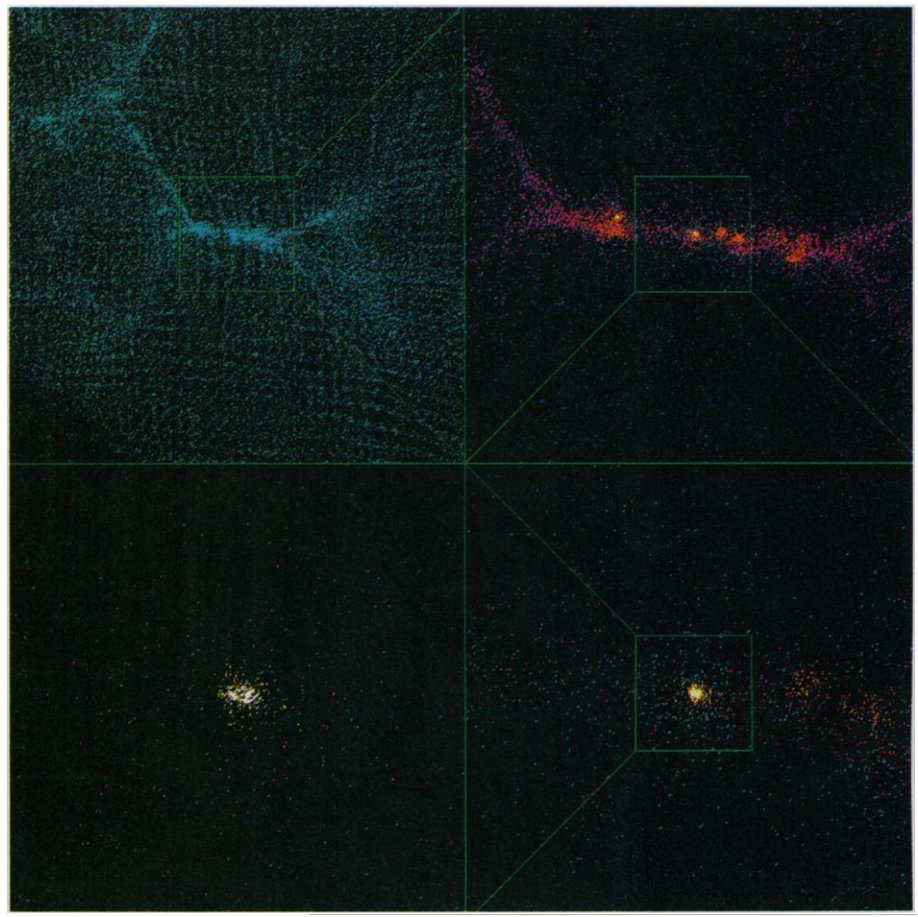

Fig. 3. A snapshot from a simulation of the formation of a galaxy in the early universe. The upper left panel is taken from a simulation that only includes gravitational interactions, whereas the other panels include the effects of gas dynamics. Particles colored cyan only interact through gravity. Particles that interact hydrodynamically as well as gravitationally are colored according to their density, with density increasing from black to blue to red to white. The upper left panel is 640 kpc across. Successive magnifications are shown clockwise, with the lower left panel measuring 10 kpc across. Such galaxies are believed to provide sites for the most distant quasars. The simulation is by Katz *et al.* (*38*).

alytical observations. The main observation is that away from a collection of sources, the potential field can be expressed as a local expansion of the form

$$\phi(\mathbf{x}) = \sum_{n=0}^{p} \sum_{m=-n}^{n} L_n^m Y_n^m(\theta, \phi) r^n \quad (25)$$

where L_n^m are local expansion coefficients. The difference between the FMM and the N log N tree codes is that multipole expansions are not evaluated at coarse levels of the box hierarchy. Instead, local expansions like Eq. 25 are used to accumulate information from the multipole expansions corresponding to the boxes in the interaction list. The resulting algorithm requires only $O(N)$ work. Perhaps more important, however, is the fact that the FMM has a richer analytical structure and has been extended to the acoustic scattering case by Rokhlin *et al.* (*26–28*) as well as to a variety of other problems and boundary conditions (*29–33*).

Acoustic scattering. The first thing to recall about the acoustic scattering case is that the far field does not have a simple structure (Fig. 1D). Thus, even though the far field attributable to gravitational or Coulombic sources can be captured very accurately in a few multipole moments, no such possibility exists here (*34*). Nevertheless, the pressure field

$$\phi(\mathbf{x}) = \sum_{i=1}^{N} W_i \frac{e^{ik|\mathbf{x}-\mathbf{x}_i|}}{|\mathbf{x}-\mathbf{x}_i|}$$

can still be expressed as a multipole expansion

$$\phi(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \alpha_n^m Y_n^m(\theta, \phi) h_n(kr) \quad (26)$$

where h_n is a spherical Hankel function and α_n^m are far-field expansion coefficients. So that we can see how many terms are needed to accurately represent the induced field, suppose that the acoustic sources are contained inside a sphere of radius a. Then, as shown in (*27*, *28*), the error in truncating the expansion (Eq. 26) after p terms is given by

$$\left| \phi(\mathbf{x}) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \alpha_n^m Y_n^m(\theta, \phi) h_n(kr) \right| \approx \left(\frac{a}{r} \right)^p \quad (27)$$

but only if $p > |k|a$. In the gravitational case, the error bound is unaffected by the size of the sphere and depends only on the ratio a/r. Here, the number of terms grows linearly with both the frequency and the size of the scattering region. By judicious use of both far field and local expansions, as well as the ability to transform quickly from one type to the other, it is possible to obtain a fast algorithm that scales like $N^{3/2}$, $N^{4/3}$, or N log N, depending on the complexity of the implementation (*27*, *28*).

Discussion

Of the fast summation methods discussed in this article, the ones that have been used

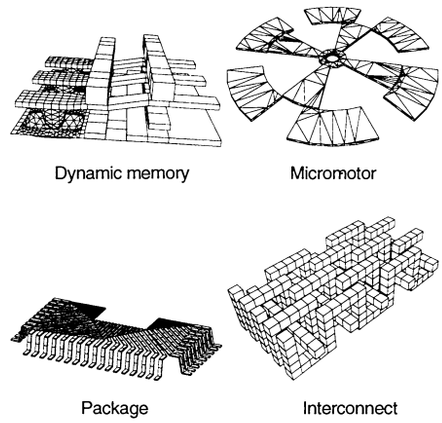

Fig. 5. The programs FASTCAP and FASTHENRY, developed at MIT, use fast multipole algorithms to reduce the time required to analyze electrostatic and magnetic interference in complex structures from days to minutes. The structures shown are three bits of an integrated circuit dynamic memory, an electrostatically actuated micromotor, a microprocessor package, and an integrated-circuit interconnect structure.

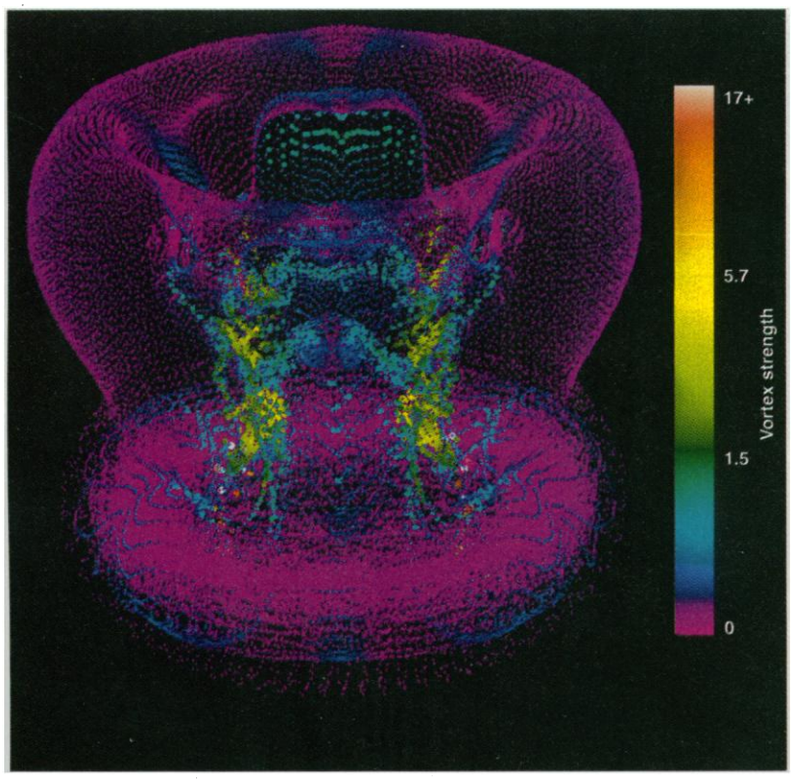

Fig. 4. The positions of 81,920 discretized vortex particles initially on the surface of a sphere are shown. Each of the particles initially carries a vorticity corresponding to the solution of potential flow past a solid sphere, that is, μA, where $\vec{\mu} = 3/8\pi \sin\theta \hat{e}_\phi$ and A is the area of the projected surface element associated with that particle. The simulation has been evolved according to the Biot-Savart law for 160 time steps. The vector directions are not shown, but the magnitude of the vortex strength is color-coded. The simulation is by Salmon *et al.* (*41*).

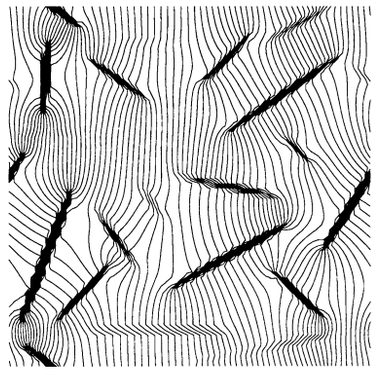

Fig. 6. A contour plot of the electrostatic potential in a composite medium. The medium consists of a uniform background of unit conductivity in which are embedded a collection of long, slender inclusions of conductivity 10^{-6}. The outlines of the inclusions can be discerned because of the accumulation of equipotential lines in poorly conducting regions (*44*).

most extensively in applications are the tree codes and fast multipole methods for problems in astrophysics, fluid dynamics, and potential theory, although a developing area of applications is computational chemistry (*35–37*). A few examples are collected here.

Figure 3 shows a snapshot of a large-scale simulation of quasar formation by Katz *et al.* (*38*), involving 10^5 particles. With a tree code (*39*), the calculation was approximately two orders of magnitude faster than would have been possible with direct summation.

In incompressible fluid dynamics (*40*), the velocity field is obtained from the vorticity field through a Biot-Savart type integral (Eq. 3). Salmon *et al.* (*41*) have developed a fast parallel tree code for both gravitational and fluid dynamical applications. The evolution of a vorticity distribution, discretized as particles and initially on the surface of a sphere, is shown in Fig. 4.

Finally, we would like to indicate how fast algorithms can be used to solve a variety of problems in electrostatics and magnetostatics (Fig. 5). These problems are usually formulated as partial differential equations, but their solution can be expressed as the field attributable to an unknown charge or dipole distribution on the surface of the object. For example, to solve the Dirichlet problem $\nabla^2 U(\mathbf{x}) = 0$ in a domain Ω with $U(\mathbf{x}) = f(\mathbf{x})$ on $\partial\Omega$, we can represent the solution by

$$U(\mathbf{x}) = \int_{\partial\Omega} \frac{\sigma(\mathbf{y})}{|\mathbf{x} - \mathbf{y}|}\, d\mathbf{y}$$

To satisfy the boundary condition, the unknown charge density $\sigma(\mathbf{y})$ must satisfy the integral equation

$$\int_{\partial\Omega} \frac{\sigma(\mathbf{y})}{|\mathbf{x} - \mathbf{y}|}\, d\mathbf{y} = f(\mathbf{x})$$

To solve this equation, we can discretize the boundary into N nodes, and the preceding equation is converted into a dense, linear system of equations. White and collaborators have developed several software packages for solving such systems on the basis of a fast multipole accelerated iterative procedure (*42, 43*). My group has also used the fast multipole method to look in detail at the electrostatic field inside composite media with complex geometries (*44*). A typical potential field is shown in Fig. 6.

Fast hierarchical algorithms are beginning to play a substantial role in scientific computing. Although I have concentrated on multipole algorithms, which encapsulate information in terms of moments, other tools from approximation theory give rise to complementary approaches (*9–11*). I have not attempted to present a comprehensive review. Instead, I have tried to explain the theoretical foundations of the methods and to give some sense for the kinds of problems currently being addressed.

REFERENCES AND NOTES

1. J. A. McCammon and S. C. Harvey, *Dynamics of Proteins and Nucleic Acids* (Cambridge Univ. Press, Cambridge, U.K., 1987).
2. J. D. Jackson, *Classical Electrodynamics* (Wiley, New York, 1975).
3. J. Crank, *The Mathematics of Diffusion* (Clarendon, Oxford, 1964).
4. The notation $O(N^\alpha)$ means that the computational cost of a procedure is proportional to N^α.
5. R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (McGraw-Hill, New York, 1981).
6. C. R. Anderson, *J. Comput. Phys.* **62**, 111 (1986).
7. A. Almgren, T. Buttke, P. Colella, *ibid.* **113**, 177 (1994).
8. H. M. P. Couchman, *Astrophys. J. Lett.* **368**, 23 (1990).
9. A. Brandt, *Comput. Phys. Commun.* **65**, 24 (1991).
10. ——— and A. A. Lubrecht, *J. Comput. Phys.* **90**, 348 (1990).
11. G. Beylkin, R. Coifman, V. Rokhlin, *Commun. Pure Appl. Math.* **44**, 141 (1991).
12. A. W. Appel, *SIAM J. Sci. Stat. Comput.* **6**, 85 (1985).
13. J. Barnes and P. Hut, *Nature* **324**, 446 (1986).
14. L. Van Dommelen and E. A. Rundensteiner, *J. Comput. Phys.* **83**, 126 (1989).
15. A. M. Odlyzko and A. Schönhage, *Am. Math. Soc. Trans.* **309**, 797 (1988).
16. V. Rokhlin, *J. Comput. Phys.* **60**, 187 (1985).
17. L. Greengard and V. Rokhlin, *ibid.* **73**, 325 (1987).
18. L. Greengard and J. Strain, *SIAM J. Sci. Stat. Comput.* **12**, 79 (1991).
19. For a related review article, see L. Greengard, *Comput. Phys.* **4**, 142 (1990).
20. P. R. Wallace, *Mathematical Analysis of Physical Problems* (Dover, New York, 1984).
21. J. K. Salmon and M. S. Warren, *J. Comput. Phys.* **111**, 136 (1994).
22. J. Carrier, L. Greengard, V. Rokhlin, *SIAM J. Sci. Stat. Comput.* **9**, 669 (1988).
23. L. Greengard, *The Rapid Evaluation of Potential Fields in Particle Systems* (MIT Press, Cambridge, MA, 1988).
24. C. R. Anderson, *SIAM J. Sci. Stat. Comput.* **13**, 923 (1992).
25. F. Zhao, *Res. Rep. AI-TR-995* (Massachusetts Institute of Technology, Cambridge, MA, 1987).
26. V. Rokhlin, *J. Comput. Phys.* **86**, 414 (1990).
27. ———, *Appl. Comput. Harmonic Anal.* **1**, 82 (1993).
28. R. Coifman, V. Rokhlin, S. Wandzura, *IEEE Antennas Propag. Mag.* **35**(no. 3), 7 (1993).
29. B. K. Alpert and V. Rokhlin, *SIAM J. Sci. Stat. Comput.* **12**, 158 (1991).
30. L. Greengard, *ibid.* **11**, 603 (1990).
31. A. Greenbaum, L. Greengard, A. Mayo, *Physica D* **60**, 216 (1992).
32. M. Gu and S. C. Eisenstat, *Dept. Comput. Sci. Res. Rep. 932* (Yale University, New Haven, CT, 1992).
33. G. Russo and J. Strain, *J. Comput. Phys.* **111**, 291 (1994).
34. Specialized data compression ideas can be used, however. F. X. Canning, *J. Electromagn. Waves Appl.* **4**, 893 (1990); B. Bradie, R. Coifman, A. Grossmann, *Appl. Comput. Harmonic Anal.* **1**, 94 (1993).
35. H.-Q. Ding, N. Karasawa, W. A. Goddard III, *J. Chem. Phys.* **97**, 4309 (1992).
36. J. A. Board, J. W. Causey, J. F. Leathrum, A. Windemuth, K. Schulten, *Chem. Phys. Lett.* **198**, 89 (1992).
37. L. Greengard and V. Rokhlin, *Chem. Scr. A* **29**, 139 (1989).
38. N. Katz, T. Quinn, E. Bertschinger, J. M. Gelb, preprint.
39. F. R. Bouchet and L. Hernquist, *Astrophys. J.* **68** (suppl.), 521 (1988).
40. A. J. Chorin and J. E. Marsden, *A Mathematical Introduction to Fluid Mechanics* (Springer-Verlag, New York, 1990).
41. J. K. Salmon, M. S. Warren, G. S. Winckelmans, *Int. J. Supercomput. Appl.* **8**, 129 (1994).
42. K. Nabors and J. White, *IEEE Trans. Comput. Aided Des.* **10**, 1447 (1991).
43. M. Kamon, M. J. Tsuk, J. White, *IEEE Trans. Microwave Theory Tech.* (special issue on Electrical Performance of Electronic Packaging), in press.
44. L. Greengard and M. Moura, *Acta Numer.* **1994**, 379 (1994).

45. This work was supported by the Applied Mathematical Sciences Program of the U.S. Department of Energy, by an NSF Presidential Young Investigator award, and by a Packard Foundation Fellowship.