THE FAST MULTIPOLE ALGORITHM

Accurate computation of the mutual interactions of N *particles through electrostatic or gravitational forces has impeded progress in many areas of simulation science. The fast multipole algorithm provides an efficient scheme for reducing computational complexity.*

Problems that involve computing the mutual interactions within large sets of particles pervade many branches of science and engineering. When the particles interact through electrostatic (Coulomb) or gravitational (Newton) potentials, the long range of the resulting forces creates a computational headache. This is because forces arise for all pairs of charges (or masses) in such systems. Because the number of pairs increases quadratically with the number of particles N included in a simulation, the computational complexity of simulations carried out without well-designed computational strategies is said to be of order $O(N^2)$. This quickly renders computational studies of the so-called N-body problem practically impossible as systems increase in size to levels relevant for realistic problems. Thus, from a computational point of view, the problems a biophysicist encounters simulating ion conduction through cellular membranes are essentially the same as those an astrophysicist encounters simulating accretion of planetary systems.

Examples from biomedicine illustrate the dilemma $O(N^2)$ algorithms pose for computing the Coulomb forces that arise in atomic simulation. All biomolecules carry partial charges centered around their atoms, resulting in Coulomb interactions. The overall biomolecular charge is usually small—local neutrality limits the effect of such forces to some degree, although accurate simulations cannot neglect them. When unbalanced charges or dipole moments arise in molecular systems, long-range Coulomb forces can dominate biomolecular-system arrangement and dynamics, and faithful descriptions of these forces are essential.

An example of this is DNA in biological cells. DNA contains two negatively charged phosphates for each pair of bases that establish the genetic code. Positive ions (such as Na+) that exist in physiological fluids neutralize these negative charges, but positive ions spread diffusively so that Coulomb forces remain strong in spite of them. Lipid bilayers that form membranes and

1521-9615/00/$10.00 © 2000 IEEE

JOHN BOARD
Duke University
KLAUS SCHULTEN
University of Illinois, Urbana-Champaign

are the staging ground for many biomolecular processes are ubiquitous in biological cells. The lipids contain head groups that may be charged but invariably feature a strong electric dipole. Because lipids are more or less aligned in parallel in membranes, the dipole moments sum rather than cancel each other. Water molecules that also carry strong dipole moments are always present near lipid bilayers and tend to counterbalance the lipid dipoles, but do so only to a limited degree (strong Coulomb forces remain). These forces imply the presence of strong electric fields across biological membranes that need to be accurately described if simulations of membrane processes are to be meaningful.

The computational complexity of naive Coulomb solvers severely constrains progress: systems with at most a few thousand atoms can be studied with $O(N^2)$ solvers on very fast computers. However, at least 200 lipids must be included in a lipid bilayer simulation to have an acceptable volume-to-surface ratio that can help model bulk properties. Additionally, at least one water molecule layer must be added to the simulation on each side of the bilayer. Thus, the smallest simulated volume is about 100Å × 100Å, which is filled with over 30,000 atoms of lipids and water. If proteins are embedded in such a system, the simulated system's size can quickly reach 200,000 atoms. The DNA-coded information is controlled through proteins that can recognize DNA sequences. Simulations that seek to understand this control must include a segment of DNA, proteins, and the ubiquitous water along with physiological ions. The smallest system of this type contains well over 30,000 atoms. Modern biology poses many exciting challenges that require simulations of systems with hundreds of thousands to millions of atoms—viral infection, the conversion of light energy into chemical energy at photosynthetic membranes of bacteria or plants, the description of DNA and proteins in chromosomes, or the transcription of genetic information into proteins at ribosomes. These challenges motivated computational scientists to seek practical solutions to the *N*-body problems inherent in Coulomb and gravitational interactions. The first such algorithm that reduced the computational effort to $O(N)$ was Vladimir Rokhlin and Leslie Greengard's fast multipole algorithm (FMA).[1]

Historical context

Rokhlin and Greengard's work arguably provided the first numerically defensible method for reducing the *N*-body problem's computational complexity, but they weren't the first to work on the problem. Some form of the *N*-body problem is at the core of many computational problems, but astrophysical simulations of gravitating bodies and the evaluation of electrostatic interactions between charged particles (such as the atoms in a biomolecular simulation) have motivated much of the reported work.

Before the development of the FMA and related algorithms, those running *N*-body simulations had two choices: either attack the $O(N^2)$ complexity of the problem with brute force or truncate the potential's infinite range to a more manageable but less accurate value. Astrophysicists tended to gravitate to the former solution, the Grape (*gra*vity *p*ip*e*) project in Japan being the most notable effort. The project built a series of massively parallel machines to rapidly evaluate gravitational interactions, culminating in Grape-4;[2] over 1,000 processors handled the computational complexity. Even so, $O(N^2)$ eventually overwhelms any number of processors, so the maximum problem size is limited.

Rokhlin and Greengard arguably provided the first numerically defensible method for reducing the N-body problem's computational complexity.

Classical molecular simulation has many more complications than the astrophysical case in that many other forces act between atoms in addition to the $1/r$ interaction (forces constraining bond lengths and angles, van der Waals forces, and more). Nonetheless, the Coulomb interaction's infinite range makes it the computationally dominant factor in such simulations. Researchers in the molecular simulation community also took the brute-force approach with Coulomb's law, with custom machines[3,4] dedicated to molecular dynamics simulation. As in the gravitational case, however, complexity trumps hardware—time limited these machines as to what size of biomolecular system they could study. The Illinois 60-processor transputer-based machine ran for over two years to obtain a successful simulation of a lipid bilayer of 200 lipids with 32,000 atoms that included water. Although this was a groundbreaking simulation at the time (it demonstrated a high degree of accuracy in comparison with experimental observation), the effort required to execute it was not easily repeatable, and the approach could not be scaled to the systems 10 to 1,000 times larger that computational biologists wanted to study.

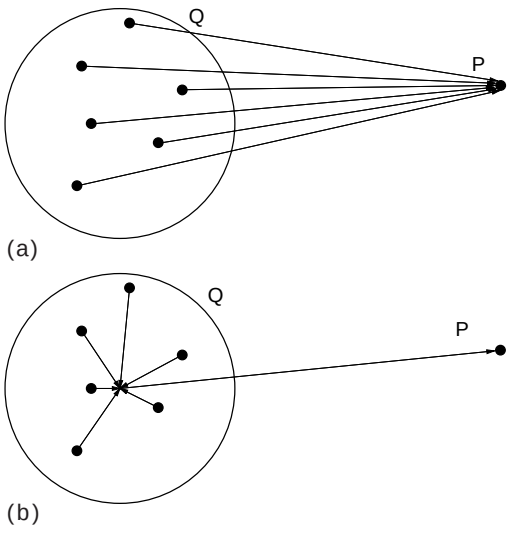

Figure 1. Rather than interact with each of the distant particles individually (a), the particle at P can interact with an approximate aggregation of the distant group, such as its centroid-located net charge (b).

Most researchers in the molecular simulation community took an easier way out. By truncating the Coulomb interaction's infinite range to 8Å–20Å or so, they reduced the $O(N^2)$ complexity to a linear problem over a fixed, finite neighborhood easily addressed with neighbor lists and other techniques. Of course, this ignored all interactions beyond the cut-off radius, thus reducing the simulation's fidelity. One argument is that because the many other forces involved in a molecular simulation are modeled at best to a few significant figures of accuracy by largely empirically determined linear or quadratic expressions, being completely faithful to the electrostatics isn't as important. Although some problems can be successfully studied this way, truncation clearly has limitations that make it unsuitable for many simulations, especially the membrane and DNA simulations discussed earlier.

The method

Matthew Pincus and Harold Scheraga suggested the first basic idea behind the FMA in the biophysical literature in 1977.[5] They described, and others later implemented, an approximation where the effect of a group of distant, charged particles on a particle of interest is described by replacing the entire distant group with a single pseudoparticle that embodies the group's properties. The key properties of the distant group of particles are its net charge, its dipole moment, and its quadrupole and higher multipole moments. Mathematically, these properties are conveniently represented by the multipole expansion of the distant group. The infinite but rapidly converging multipole series expansion is truncated at a convenient number of terms, in practice usually three to eight, with more terms giving higher accuracy in the approximation. The particle of interest can now interact with the entire distant group by instead interacting with the single multipole expansion that represents the group, instead of with all the distant group's individual members (see Figure 1).

The second key idea of the FMA and related methods is to use a hierarchical decomposition of space to rationally separate the simulation region into areas that are suitably distant from each other to invoke the multipole expansion approximation. In Figure 2's oct-tree decomposition, ever-larger regions of space that represent increasing numbers of particles can interact through individual multipole expansions at increasing distances. The first practical algorithms[6,7] combined the two ideas for use in astrophysical simulations. Both methods have a computational complexity of $O(N \log N)$ in the number of particles N, an improvement over $O(N^2)$. Additionally, the monopole moment is large in the Newtonian case, because all mass is positive. Thus, the monopole term alone, and certainly the first two terms of the multipole series (monopole plus dipole terms), computed the gravitational interactions quite accurately.

The electrostatic problem is complicated because charge distributions' monopole moments are usually small; positive and negative charges roughly cancel each other out. By adding more terms to the multipole series, we can adapt the Barnes-Hut algorithm to the electrostatic case, but the resulting method is difficult to rigorously analyze for its numerical robustness. Enter Greengard and Rokhlin's fast multipole algorithm in 1987. Their introduction of a *local expansion* further reduced the procedure's complexity from $O(N \log N)$ to $O(N)$, at least in certain important cases. Additionally, their method's machinery was amenable to a rigorous numerical analysis that bounded the method's error, removing the somewhat ad hoc feel of the earlier methods. We can now confidently determine how many terms are required in a multipole expansion to achieve a certain guaranteed level of accuracy.

The local expansion idea is critical to their improved scheme. Now, distant groups of particles interact with entire groups of target particles at once: both the distant group and the target group are represented by multipole expansions. An interaction between these groups essentially

involves a convolution of the coefficient arrays describing their respective multipole expansions.

An additional and crucial benefit of Greengard and Rokhlin's approach is that it is not restricted to the $1/r$ potential. Multipole-like formulations can be constructed for any power law potential and for other functional forms; we can apply the same complexity-reducing FMA mechanics to these cases with similar results. The $1/r$ case enjoys some special properties that simplify its analysis, but extension of these methods to other classes of potential functions is an active and fruitful area of current work.

Researchers are studying very large astrophysical simulations with hybrids of the FMA and the earlier Barnes-Hut scheme. In the biophysical-simulation world, the Ewald summation method is an additional competitor. Since the development of the FMA, scientists have created various fast versions of the nearly 80-year-old Ewald method that are faster than multipole codes in some cases, although their error behavior is harder to quantify. The Ewald codes also handle periodic boundary conditions automatically; FMA-derived codes can be extended to this case with extra effort. Nonetheless, FMA and its offspring remain important, and the newest formulations promise to again challenge Ewald codes for the title of fastest electrostatic solver.

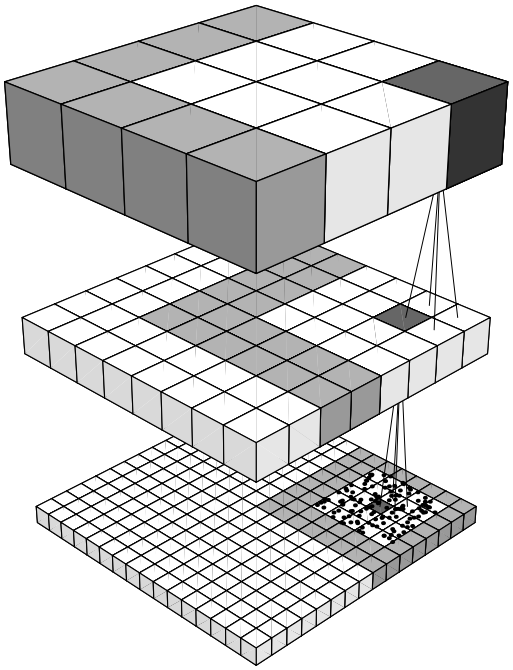

Figure 2. Multipole algorithms use hierarchical spatial decomposition to separate the simulation space into regions sufficiently far apart from each other to interact according to the approximate method in Figure 1. At increasing distances, ever-larger regions of space can be lumped into single approximations.

References

1. L. Greengard and V. Rokhlin, "A Fast Algorithm for Particle Simulation," *J. Computational Physics*, Vol. 73, No. 2, Dec. 1987, pp. 325–348.
2. J. Makino and T. Makoto, *Scientific Simulations with Special-Purpose Computers: The GRAPE Systems*, John Wiley & Sons, New York, 1998.
3. D.J. Auerbach, W. Paul, and A.F. Bakkers, "A Special-Purpose Computer for Molecular Dynamics: Motivation, Design, and Application," *J. Physical Chemistry*, Vol. 91, No. 19, 10 Sept. 1987, pp. 4881–4890.
4. K. Boehncke et al., "Molecular Dynamics Simulation on a Systolic Ring of Transputers," *Transputer Research and Applications 3*, A.S. Wagner, ed., IOS Press, Amsterdam, 1990.
5. M.R. Pincus and H.A. Scheraga, "An Approximate Treatment of Long-Range Interactions in Proteins," *J. Physical Chemistry*, Vol. 81, No. 16, 11 Aug. 1977, pp. 1579–1583.
6. A.W. Appel, "An Efficient Program for Many-Body Simulation," *SIAM J. Scientific and Statistical Computing*, Vol. 6, No. 6, Jan. 1985, pp. 85–103.
7. J.E. Barnes and P. Hut, "A Hierarchical O(*N*log(*N*)) Force-Calculation Algorithm," *Nature*, Vol. 324, No. 6096, 4 Dec. 1986, pp. 446–449.

John Board is the Bass Associate Professor and associate chair in the Electrical and Computer Engineering Department at Duke University. He is also director of the Center for Computational Science and Engineering at the university. His research interests include the application of high-performance computing techniques to problems in the biological and physical sciences. He received his MS in electrical engineering from Duke, and his D.Phil in theoretical physics from Oxford University. Contact him at the Dept. of Electrical and Computer Eng., Duke Univ., Box 90291, Durham, NC 27708; jab@ee.duke.edu.

Klaus Schulten is Swanlund Professor of physics and director of the Theoretical Biophysics Group at the Beckman Institute at the University of Illinois, Urbana-Champaign. His research interests focus on the architecture, functions, and mechanisms of molecular aggregates in biological cells. In his research he employs large-scale molecular dynamics simulations as well as quantum mechanical and statistical mechanical descriptions. His research group develops molecular graphics, simulation, and collaboratory software. He received his Diplom in physics at the University of Münster, Germany, his PhD in chemical physics at Harvard, and his habilitation degree at the University of Göttingen, Germany. Contact him at the Beckman Inst., Univ. of Illinois, Urbana, IL 61801; kschulte@ks.uiuc.edu.