COMPUTING AT THE LEADING EDGE

RESEARCH IN THE ENERGY SCIENCES

The New Roles of Leading-Edge Computing3

The Numerical Tokamak Project

*D. C. Barnes, B. A. Carreras, B. I. Cohen, J. M. Dawson,
G. W. Hammett, S. C. Jardin, G. D. Kerbel,
P. C. Liewer, T. Tajima, and R. E. Waltz5*

Massively Parallel Processing12

Static and Animated Molecular Views of a Tumorigenic Chemical Bound to DNA

*Brian E. Hingerty, Suse Broyde, Rong Xu,
Suzanne Freeman O'Handley, and Thomas R. Krugh14*

Toward a High-Performance Climate Systems Model

W. P. Dannevik, M. C. MacCracken, and A. A. Mirin22

Collaborating to Solve a Major Data Storage Challenge31

Modeling Molecular Processes in the Environment

*T. H. Dunning, Jr., B. C. Garrett,
A. C. Hess, R. L. Ornstein, and R. A. Bair32*

Lattice Boltzmann Models for Flow in Porous Media

*Bryan Travis, Ken Eggert, Darryl Grunau, Shi Yi Chen, and
Gary Doolen42*

Parallel Algorithms for Modeling Superconductors

James L. Garner, Mark T. Jones, and Paul E. Plassmann48

MASTER

Parallel Computing at the Superconducting Super Collider Laboratory	
<i>T. Barts, B. Cole, and G. Bourianoff</i>	53
The Energy Sciences Network	60
The Advanced Combustion Modeling Environment (ACME)	
<i>M. L. Koszykowski, R. C. Armstrong, R. E. Cline, Jr., J. F. Macfarlane, N. J. Brown, and J. Y.-Chen</i>	61
Adaptive Methodologies for Computational Fluid Dynamics	
<i>John Bell, Marsha Berger, P. Colella, and J. Saltzman</i>	64
Integrated Distributed Computing	70
Lattice Simulations of Quantum Chromodynamics	
<i>Urs M. Heller</i>	72
Simulating High-Intensity Charged-Particle Beams for the Design of High-Power Accelerators	
<i>William Herrmannsfeldt, Kwok Ko, Cho Ng, Irving Haber, Alex Friedman, and David Grote</i>	79
Electronic Structure and Phase Stability of Random Alloys	
<i>G. M. Stocks, D. M. Nicholson, W. A. Shelton, and G. A. Geist</i>	86
The National Education Supercomputer Program	94
The National Energy Research Supercomputer Center	96
Bill McCurdy Talks About the Superkids	98

The New Roles of Leading-Edge Computing

Over the past quarter-century, a fundamental change has occurred in the way scientists and engineers view computation as a tool of research. In the 1960s, computation was a specialized tool whose application was largely limited to a few disciplines of physics, engineering, and chemistry, and which was widely considered to be merely an adjunct of theory. After a quarter-century of spectacular advances in computing hardware and numerical algorithms, we now commonly speak of experiment, theory, and computation as the three principal elements of modern scientific research. The change in our thinking is dramatically highlighted by discussions of large-scale "computational experiments" appearing in the scientific literature side by side with the results of physical experiments. Over the last decade, progress has accelerated still further as advances in microprocessor technology have revolutionized the business and personal use of computers in our society as well as the tools on the desks of our scientists and engineers.

The phrase "Grand Challenges of computational science" came into general usage in the late 1980s, when conferences on the subject sought to identify the major scientific goals that can be reached by using computation in its revolutionary role as a new mode of inquiry. The nature and scope of the Grand Challenges can be best understood by example. Within the mission of the Department of Energy, the complete mechanical, physical, and chemical design of efficient internal combustion engines and the comprehensive design of fusion-energy devices are such examples, as are the molecular-level design of ways to extract pollutants from water and soil and the design of new industrial materials and superconductors. The Grand Challenges range from the modeling of the global climate to the prediction of the functions and structures of biologically active molecules.

The purpose of this publication is to highlight selected scientific challenges that have been undertaken by the DOE Energy Research community. I would like to express my appreciation to the authors of the articles collected here. The high quality of the research reflected in their contributions underscores the growing importance both of the Grand Challenge scientific efforts sponsored by DOE and of the related supporting technologies that the National Energy Research Supercomputer Center (NERSC) and other facilities are able to provide. The continued improvement of the computing resources available to DOE scientists is a prerequisite to ensuring their future progress in solving the Grand Challenges.

The National Energy Research Supercomputer Center has been in existence since the beginning of the supercomputer revolution of the late 1970s and now plays a central role in providing computational technology for research in the energy sciences. As the principal supplier of production high-performance computing and networking services to the Energy Research community, the Center has met the general-purpose computing needs of energy scientists by providing a centrally located supercomputer resource, accessible to the remote user by a high-speed network. Our latest supercomputer is a 16-processor, 256-million-word CRAY Y-MP C90, the fastest Cray computer currently available. Another major service of NERSC is managing the Energy Sciences Network (ESNET), the nationwide wide-area network for the Department of Energy. NERSC is guided by its user community and its close connections with the scientific programs of the Office of Energy Research.

While NERSC began during the supercomputer revolution, its present and future are part of the microprocessor revolution. The explosion in microprocessor technology and the vast improvement in national networking capability have produced a new and powerful paradigm for computing using widely distributed resources. Many small-to-medium jobs are being off-loaded from the supercomputer to the workstation, where they often achieve faster turnaround at lower cost. With the advent of a standard UNIX interface and the proliferation of compatible software, the workstation has become a useful resource not only for production computing but also for pre- and post-processing activities such as text-editing and visualization. The emergence of easy-to-use portable communications software now allows scientists to distribute calculations over clusters of workstations connected by high-speed networks.

The ability to treat a heterogeneous distributed system as a single entity is now being developed by the computer industry and by service providers like NERSC. In the integrated distributed computing environment, the scientist will be able to log in to a collection of computers and access a single, unified (and distributed) file and storage system. We look forward to providing this capability to our users, and we plan to provide basic core services, through the Energy Sciences Network, to support wide-area distributed computing environments throughout the Energy Research community.

The other major result of the microprocessor revolution is the emergence of massively parallel computers as the next class of supercomputer. These machines consist of hundreds or thousands of microprocessors equipped with large memories and connected by the fastest communications hardware yet designed. The comparison of tremendous gains in this area with more modest progress in conventional vector processing leads us to believe that within several years, massively parallel computers will provide a major fraction of the supercomputing needs of the Office of Energy Research. Many of the articles in this issue cite results from massively parallel systems. NERSC's plans for the immediate future focus on the transition to massively parallel computing for energy researchers.

Computer networking technology is progressing as fast as other parts of the discipline. With the emerging Asynchronous Transfer Mode (ATM) technology, the performance of ESNET will undergo rapid evolution over the next several years—from the current 1.5 million bits per second to nearly a billion bits per second. The barriers to the integrated management of large volumes of widely dispersed data and to wide-area interactive visualization of scientific results are coming down. ATM technology is the basis for technologies that will allow simultaneous transmission of voice, data, and video.

The larger memories and faster speeds of new computers have caused data storage to become a major bottleneck for many scientists. The National Storage Laboratory, located at NERSC, is a collaboration between government and industry to bring large-scale hierarchical storage system performance and functionality into better balance with modern processing capabilities and new applications requirements. The National Storage Laboratory will initially focus on two major applications—fusion energy and climate modeling.

The expanding role of computation in science depends most critically on the pool of talented scientists who work in computationally intensive research and on our ability as a nation to attract and train young people from all parts of our diverse society for scientific careers. Our role in that effort is an important focus at NERSC. We were the first Department of Energy center to organize a supercomputing program for high-school students. We currently house the National Education Supercomputer, a CRAY X-MP dedicated to high-school education, and sponsor an innovative teacher training program.

There is national recognition of the growing role of computation in all parts of society, and the Federal Government's High-Performance Computing and Communications Initiative is spurring leading-edge computational science research at a number of Department of Energy institutions. We value our increasingly close ties with those institutions, with computer manufacturers and software houses, and with other research centers, because they help us bring new technology into the production computing environment and provide the highest level of service to our users.

Large-scale computing is indispensable to research in the energy sciences, and we at NERSC remain committed to providing the computational resources needed by our users to continue to expand the horizons of computational science.

C. William McCurdy, Director
National Energy Research Supercomputer Center
Lawrence Livermore National Laboratory

It comprises a consortium of researchers at six national laboratories of the DOE and the National Aeronautics and Space Administration (NASA), five universities, the National Energy Research Supercomputer Center at Livermore, the Advanced Computing Laboratory at Los Alamos, and the High-Performance Computing Center at Oak Ridge.

To obtain meaningful results, the models must include features known to be essential to tokamak transport. One is a true toroidal geometry, in which magnetic field lines follow spiraling (helical) paths around the torus. Another is the almost-free motion of particles along this twisted field, which connects points all along a given field line. Particles moving parallel to the magnetic field and those moving perpendicular to the magnetic field have greatly different time scales that must also be taken into account, as must the disparity in the speeds and responses of the relatively light electrons and the heavier ions in the plasma. Finally, the models must reflect a range of space scales that extends from the gyroradius of an ion (a few millimeters) to the size of the machine itself (a few meters).

A Twofold Approach

To reliably address such a mix of issues, we are following two main approaches and comparing their results. In the most fundamental approach, we computationally advance a collection of particles to sample the motion of actual plasma particles. We then use the resulting particle data to calculate plasma currents and charge densities by interpolating to a spatial grid and finally solve Maxwell's equations for the self-consistent electromagnetic fields.

The particle approach is more faithful to the physics, but its numerical resolution is limited. In an alternative approach, we use a set of fluid equations that deal with parallel and drift motions by advancing a few velocity moments of the particle velocity distribution with approximate models for the higher moments. Maxwell's equations are again solved to complete the self-consistent description. We are currently comparing solutions from these two methods with one another, with the predictions of analytic theory, and with experimental observations.

Computational Efforts

Although the nation's fusion program has long featured forefront scientific computing, one problem that until now has demanded more computing power than was available is turbulent transport calculations. Using the generation of computers just becoming operational, we can now carry out, in a true toroidal geometry, calculations that directly simulate a small to moderate-sized tokamak. However, the parameters of such calculations still fall short of those of larger machines. Accordingly, future calculations will push the largest available computers to their limits, driving the development of high-performance computing in several unique ways. Our calculations, which use both particle and fluid methods, have already identified and addressed several important issues in parallel computing. Among these are efficient algorithms for locating and following particles in complex geometries and load-leveling among many processors. Both particle and fluid methods will require the most advanced data storage and visualization tools to deal with the very large number of degrees of freedom represented in such calculations.

Numerical studies of plasma confinement in a tokamak configuration are complicated by several factors. Perhaps the most important is the extreme range of time scales in a fusion plasma, extending from 10^{-12} s for the fastest electrons to several seconds, the time scale required for the plasma profile to relax to a true steady state. The fluctuations of greatest interest have time scales in the range of a fraction to several milliseconds.

Great efforts have been made during the past decade to develop model equations and appropriate algorithms that embody these complex processes. Both the fluid and the particle approaches have been extended in the direction of each other. For example, particle-in-cell (PIC) methods, initially developed to examine the fastest time scales in a collisionless plasma, have been extended into the low-frequency regime of interest by using orbit-averaging and implicit techniques. On the other hand, fluid approaches originally applied to describe large-scale magnetohydrodynamic phenomena have been modified to include the essential kinetic effects

associated with nearly free motion of a plasma particle along the magnetic field.

The NTP is adapting both approaches to provide two independent numerical treatments in a region where the time scales overlap (Fig. 1). If validated in the overlap region, fluid models—which require many fewer degrees of freedom than PIC models—may provide a much more efficient means of studying transport. Comparison of the fluid and PIC approaches will also quantify several issues related to the numerical precision of the latter, including how the shot noise of individual particles affects the numerical solutions.

Particle Models

Several collaborating groups within the NTP are using PIC techniques with kinetic models to study turbulent transport in tokamaks. The groups are closely coupled in developing new algorithms, in porting and optimizing codes to new massively parallel computers, in undertaking physics applications, and in the other aspects of this physics and computing Grand Challenge. Particle codes have been developed in a variety of geometries: two- and three-dimensional slabs, three-dimensional cylinders, and most recently, three-dimensional toroids.

Most applications to date use electrostatic models, but partially electromagnetic models that include effects of finite plasma currents are being developed. Advanced algorithms have enabled much progress in physics applications whose time scales are considerably longer than the characteristically fast time scales of high-frequency plasma waves, particle transit times, and cyclotron periods.

The outstanding development in advanced kinetic algorithms is the gyrokinetic particle simulation model.¹ This model formally averages time over the cyclotron time scale while retaining spatial variations on the length scale of the Larmor radius. The validity of the gyrokinetics formalism depends on an ordering scheme in which temporal variations are slower than the cyclotron frequency, spatial variations in the equilibrium quantities are weaker than those in the perturbing electromagnetic fields, and velocity perturbations in the fields are smaller than the thermal speed. This set

of assumptions is well satisfied by a variety of instabilities that occur in the core of tokamaks and are believed to be the principal agents of anomalous transport.² In not being constrained to the cyclotron time scale, the simulation can employ a larger timestep that is more appropriate to the diamagnetic drift frequency. The model also features less electromagnetic noise from thermal fluctuations. As a result, low-frequency collective modes have been simulated more economically.^{3,4}

An additional innovation in gyrokinetic methods is the δf algorithm.^{5,6} In the δf algorithm, particles are used only to follow the deviations of the distribution from a local Maxwellian whose response to perturbing fields can be analytically represented if the deviations are of sufficiently small amplitude. This method achieves an additional savings in the number of particles required and is an effective simulation tool. Implicit methods and time-splitting techniques further improve the efficiency of the algorithms.^{7,8} The addition of toroidal effects has been especially important in improving the relevance of particle simulations to the understanding of tokamaks.⁹

NTP workers have ported and optimized gyrokinetic and other PIC codes to the multiprocessor supercomputers at NERSC and Cornell, and to massively parallel computers with diverse architectures such as the BBN-TC2000, the CM-2, Intel Hypercubes, and the Intel Delta. Code performance on these massively parallel

Figure 1. Schematic showing time and regions of applicability for fluid and kinetic simulation models in magnetic fusion. We will use both particle-in-cell and fluid approaches to examine fluctuation-driven transport in the overlap region. These methods will be applied to extend the accessible region of parameter space and to allow efficient identification of scaling regimes.

computers rivals or exceeds that on a single processor of the CRAY 2. Consortium members have also made progress in comparing the performance of SIMD and MIMD programming paradigms for these codes.

Several near-term physics calculations contribute to the NTP's kinetic simulation effort:

- Three-dimensional toroidal and quasi-toroidal simulations of ion-temperature gradient modes and trapped-particle instabilities relevant to core transport.
- Three-dimensional toroidal and quasi-toroidal simulations of ion-temperature-gradient modes with velocity

shear relevant to transport in the shear layer on the outer part of the tokamak.

- Hybrid gyrokinetic-magnetohydrodynamic three-dimensional simulations of energetic particles to study sawtooth activity, fishbone oscillations, and toroidal Alfvén eigenmodes excited by fusion alphas.¹⁰

Figures 2 and 3 are from gyrokinetic simulations of turbulent transport.¹¹ A combination of toroidal effects and the free energy contained in the ion temperature gradient excites an electrostatic instability that tries to relax by driving a plasma heat flux across the magnetic surfaces to the plasma boundary. The perpendicular component of the turbulent electric fields associated with the instability accelerates the plasma perpendicular to the magnetic field, producing an $E \times B$ flux of plasma and heat across the magnetic surfaces.

An important finding of our three-dimensional gyrokinetic simulations is that toroidal effects significantly increase saturated turbulent transport over what is observed in sheared-slab simulations without toroidal effects, and bring the simulation results into closer agreement with experimental observations in large tokamak experiments.¹²

Gyrofluid Models

The second modeling area involves the use of gyrofluid equations¹³⁻¹⁵ to simulate turbulent transport and other important tokamak phenomena. Traditional fluid equations, such as those of magnetohydrodynamics (MHD), have had a long and successful history of explaining large-scale equilibrium and stability properties of plasmas. However, they lack many "kinetic" effects (such as Landau damping) that are important for dealing with instabilities of the lower-frequency, shorter-wavelength drift type, believed to be responsible for plasma turbulence. The particle methods described above treat such instabilities automatically; only recently has a method been found to approximate them in fluid equations.¹³

Unlike traditional Navier-Stokes fluids, in which the mean free path of a particle is very short (so that nearby particles can

Figure 2. Gyrokinetic particle simulation of turbulence driven by an ion temperature gradient in a tokamak. Colors represent high- and low-amplitude potential fluctuations. (a) Linear mode structure. (b) Fully developed turbulence.

be assumed to move together as a fluid in a local thermodynamic equilibrium), the particles in a tokamak plasma may travel hundreds of meters along a magnetic field line before colliding with another particle. However, because they move at different speeds (typically a Maxwellian distribution), plasma particles that are initially close together end up far apart.

The collisionless mixing of particles is related to the Landau damping phenomenon often seen in plasmas. To model this process in fluid equations, we typically add a parallel heat conductivity to the temperature evolution equation to provide a damping rate of the order of the collisionless thermal mixing rate (the magnitude of the parallel wave number times the thermal speed). This can easily be evaluated with Fourier methods and turns out to be a fairly good Padé, or multipole, approximation to the linear kinetic plasma response. This procedure converges fairly rapidly if higher fluid moments are kept before introducing this dissipative closure approximation. When self-consistent electric fields are added to this model, it describes not only Landau damping (which provides an important energy sink for nonlinear turbulence simulations) but also inverse-Landau damping, which drives some types of plasma instabilities.

Another important kinetic effect is that an ion responds not to the local electromagnetic field but to the field averaged over the ion's gyro-orbit. We have recently developed Padé-like approximations that model the linear and nonlinear consequences of this effect. Such models enable us to formulate parabolic problems that can be solved by using fast Fourier transforms either alone or in combination with tridiagonal solvers.

The fluid approach provides an alternative, complementary path to kinetic simulations in addressing a class of similar phenomena. Extended to include kinetic effects, fluid models describe well the linear behavior addressed by the kinetic approach. The equations of fluid models embody important nonlinear conservation laws (particle numbers, momentum, energy, average magnetic moment, and so on) that the collective turbulence must satisfy. They also contain $\mathbf{E} \times \mathbf{B}$ and

related nonlinearities that are believed to be the dominant coupling mechanism between linear modes.

Because current gyrofluid models do not incorporate all kinetic effects, their range of applicability is not yet known, and comparisons between fully nonlinear particle and fluid simulations need to be carried out. However, as they are not formulated in terms of particle phase space, computational requirements are generally less severe than for kinetic simulations. This enables the same computing resources to simulate larger systems over longer time scales. In general, the character and demands of the gyrofluid model on both algorithms and computing capability differ significantly from those of particle-pushing. As a result, we are being led to develop a new set of computing skills and technologies.

NTP researchers have already ported and optimized three-dimensional gyrofluid codes on the Intel Gamma and Delta machines and on the CM-2/200. Porting and optimization of such codes on the Paragon and the CM-5 are planned. Some applications, such as the simulation of drift-wave instabilities and ion-temperature-gradient modes, have been completed and are being extended to toroidal geometries with the inclusion of electromagnetic effects and trapped particles. We continue to improve the fluid models by developing closure schemes

Figure 3. Isosurface rendering of the electrostatic potential for an ion-temperature-gradient instability obtained from a gyrokinetic calculation.

that retain more of the physics contained in the kinetic equations.

We are also undertaking new applications such as simulations of the dissipative trapped-electron mode and of toroidal Alfvén eigenmodes excited by alpha particles. We are using message-passing and data parallel programming techniques and also comparing the performance and optimization of algorithms. We have developed a fast Connection Machine block tridiagonal matrix solver based on parallel cyclic reduction and plan to extend it to block pentadiagonal systems. Development of faster linear system solvers continues. Figures 4 and 5 are examples of gyrofluid simulations of turbulence excited by ion-temperature-gradient modes and by toroidal Alfvén eigenmodes.¹⁵

Figure 4. Gyrofluid simulation of turbulence driven by an ion temperature gradient in a tokamak. Colors represent high- and low-density fluctuations.

Figure 5. Gyrofluid simulation of beam-driven toroidal Alfvén eigenmode instability in the Tokamak Fusion Test Reactor. Colors represent electrostatic potential.

The Enabling Computational Science Program

The NTP is arguably the most comprehensive software project ever undertaken in service of the nation's fusion program. Its complexity dictates a cooperative approach that combines modern computational tools, new computer-supported collaborative work technology, and new styles of computing. Computational physicists are working closely with computer science specialists (some from private industry) to create and perfect the necessary software tools. Industry will no doubt benefit from this cooperation, as will other scientific research projects. Advances in parallel processing, communications, database, visualization, and software technologies are providing the enabling computational science components of the project.

We are devising algorithms and methods to enable the NTP to use massively parallel computers efficiently. Among the most important are those for solving elliptic equations, performing fast Fourier transforms, and sorting N -body data for particle simulations. Particle-in-cell methods are particularly challenging for parallel machines.¹⁶ Each particle moves under the influence of force fields that depend on having access to data on all other particles. A key to success in such calculations is in keeping communication costs balanced with computing costs. Members of the Numerical Tokamak consortium are already developing parallel algorithms in a number of these critical research areas.

It would be a distinct advantage to have high-level programming languages complete with parallel constructs and intrinsics that can be compiled into efficient code on a variety of supercomputer architectures. The Numerical Tokamak consortium is creating both constructs and standards to measure progress in this important area. Our goal is to target a variety of massively parallel machines with a single source code. We will use a visual programming environment to select code modules created by different scientists and assemble them like building blocks to work cooperatively as a unit. An intuitive visual interface will make the Numerical Tokamak more accessible to a wider community of both theorists and experimentalists.

Immediate communication and rapid feedback are important to the NTP. Strong visual interaction among researchers is essential, for example, in comparing experimental and simulation data. The enabling technology is a gigabit-per-second network embodying extensions of current network software. The goal here is to enable a group of scientists working at widely separated sites across the country to cooperate and collaborate as if they were in a single room sharing visual access to the same data.

Managing the large amounts of data stored in a heterogeneous environment requires a comprehensive database management system. Current database technology must be extended from the workstation environment into the large-scale scientific computing environment, making use of large, high-bandwidth archival storage systems. This project will rely heavily on collaborative efforts between the DOE's High-Performance Computing Research Centers and archival storage vendors. For its part, the NTP consortium will systematize and automate the forms and procedures for exchanging data within the project.

The intuitive display of a large volume of physical information is an important element of the NTP. Visualization of three-dimensional vector fields produced by our simulations involves the manipulation of very large files (many billions of words for a time history) and requires powerful interactive workstations and supporting three-dimensional visualization software such as the AVS or IDL commercial software packages. The database management tools and high-speed network described above play key roles in facilitating our visualization tasks.

Goal of the Numerical Tokamak Project

The Numerical Tokamak Project will provide, for the first time, complete, reliable, and accurate numerical solutions of the model equations believed to describe plasma transport in tokamak geometries. Should these solutions agree with the growing body of experimental observations of transport rates and their scaling, and with details of fluctuation structures, we will have in hand convincing evidence

that transport can be reliably predicted in future machines. Such a capability will hasten the development and commercialization of fusion energy as well as saving billions of dollars in development costs.

Acknowledgments

We are grateful to the National Energy Research Supercomputer Center, the Jet Propulsion Laboratory, Oak Ridge National Laboratory's High-Performance Computing Center, and the Advanced Computing Laboratory at Los Alamos National Laboratory for providing computing services and assistance.

References

1. W. W. Lee, "Gyrokinetic Particle Simulation Model," *J. Comput. Phys.* **72**, 243 (1987).
2. J. D. Callen, "Transport Processes in Magnetically Confined Plasmas," *Phys. Fluids B* **4**, 2142 (1992).
3. W. W. Lee and W. M. Tang, "Gyrokinetic Particle Simulation of Low-Frequency Drift Instabilities," *Phys. Fluids* **31**, 612 (1988).
4. R. D. Sydora, "Three-Dimensional Gyrokinetic Particle Simulation of Low-Frequency Drift Instabilities," *Phys. Fluids B* **2**, 1455 (1990).
5. M. Kotschenreuther, "Particle Simulations with Greatly Reduced Noise," *Proceedings of the 14th International Conference on Numerical Simulation of Plasmas* (Annapolis, MD, 1991).
6. A. M. Dimits and W. W. Lee, *Partially Linearized Algorithms in Gyrokinetic Particle Simulation*, Princeton University PPPL-2718 (October, 1990), accepted for publication in *J. Comput. Phys.*; S. E. Parker and W. W. Lee, *A Fully Nonlinear Characteristic Method for Gyrokinetic Simulation*, PPPL-2851 (1992), to appear in *Phys. Fluids B* (January 1993).
7. J. U. Brackbill and D. W. Forslund, "Simulation of Low-Frequency Electromagnetic Phenomena in Plasmas," in *Multiple Time Scales (Computational Techniques*, Academic Press, 1985), p. 272; A. B. Langdon and D. C. Barnes, "Direct Implicit Plasma Simulation," in *Multiple Time Scales (Computational Techniques*, Academic Press, 1985), p. 336.
8. B. I. Cohen and T. J. Williams, "Semi-Implicit Particle Simulation of Kinetic Plasma Phenomena," *J. Comput. Phys.* **97**, 224 (1991); B. I. Cohen and T. J. Williams, *Implementation of a Semi-Implicit Orbit-Averaged Gyrokinetic Particle Code*, Lawrence Livermore National Laboratory, Livermore CA, UCRL-JC-109803 (February 1992), submitted to *J. Comput. Phys.*

9. M. J. LeBrun and T. Tajima, *Particle Simulation in Curvilinear Coordinate System*, University of Texas IFS-374 (1989), submitted to *J. Comput. Phys.*; M. J. LeBrun, T. Tajima, M. Gray, G. Furnish, and W. Horton, *Toroidal Effects on Drift-Wave Turbulence*, University of Texas IFS-563 (1992), submitted to *Phys. Fluids B*.
10. W. Park, S. Parker et al., "Three-Dimensional Hybrid Gyrokinetic-Magnetohydrodynamic Simulation," *Phys. Fluids B* **4**, 2033 (1992).
11. S. E. Parker and W. W. Lee, "Gyrokinetic Simulation of Ion Temperature Gradient Instabilities in 3D Toroidal Geometry," in *Proc. 14th Int. Numerical Simulation of Plasmas Conference* (Annapolis, MD, 1991).
12. M. Kotschenreuther, "Continuing Development and Applications of the δf Algorithm for Tokamaks," *Bull. Am. Phys. Soc.* **36**, 2435 (1991); M. Kotschenreuther et al., *Comparisons of Theoretically Predicted Transport from ITG Instabilities to L-Mode Tokamak Experiments*, University of Texas IFSR-532 (December 1991).
13. G. W. Hammett, W. Dorland, and F. W. Perkins, "Fluid Models of Phase Mixing, Landau Damping, and Nonlinear Gyrokinetic Dynamics," *Phys. Fluids B* **4**, 2052 (1992); G. W. Hammett and F. W. Perkins, "Fluid-Moment Models for Landau Damping with Application to the Ion-Temperature-Gradient Instability," *Phys. Rev. Lett.* **64**, 3019 (1990).
14. R. E. Waltz, "Three-Dimensional Global Numerical Simulation of Ion Temperature Gradient Mode Turbulence," *Phys. Fluids* **31**, 1962 (1988).
15. D. A. Spong, B. A. Carreras et al., "Fluid/Kinetic Models for Shear Alfvén TAE Modes in TFTR," *Bull. Am. Phys. Soc.* **36**, 2394 (1991); J.-N. Lebouef et al., "TEXT Tokamak Edge Turbulence Modeling," *Phys. Fluids B* **3**, 2291 (1991).
16. P. C. Liewer and V. K. Decyk, "A General Concurrent Algorithm for Plasma Particle-In-Cell Simulations Codes," *J. Comput. Phys.* **85**, 302 (1989).

Massively Parallel Processing

Conventional supercomputers fall far short of providing the teraflop-range computing power that many Office of Energy Research Grand Challenge applications will require. A number of these applications are discussed in this issue, including global climate modeling, accelerator design, lattice gauge theory, plasma simulation, flow through porous media, combustion, alloy theory, superconductors, molecular dynamics, and DNA modeling. Continuing advances in microprocessor technology and parallelism are making possible a new generation of architectures that will help meet the demand for greater compute power.

Current state-of-the-art microprocessors offer peak performance on the order of 100 megaflops, roughly equivalent to that of a CRAY X-MP processor. When hundreds to thousands of these processors are packaged together and interconnected by a high-speed communications network, their potential performance is one to two orders of magnitude greater than that offered by leading conventional vector multiprocessors, at comparable cost. Moreover, microprocessor-based platforms are expected to increase in performance at a much greater rate. Anticipated improvements in microprocessor technology should result in massively parallel computers that will deliver teraflop performance by the mid-nineties. These advances in hardware capability, coupled with advances in software technology and in parallel numerical algorithms, will provide the means to make significant progress in Grand Challenge computing in this decade.

While the principal focus of attention in the past has been on the dramatic increase in raw computing power, researchers have had difficulty realizing the full potential for two major reasons. First, the scientist must implement an algorithm that divides the problem into a large number of tasks that can be executed

in parallel. For some applications this is not particularly difficult; for others, such as the inversion of a large matrix, the optimal procedure is not at all obvious. Second, each processor must synchronize its work with the other processors because it needs data generated by them. An unbalanced workload, together with the inability to completely overlap communication with computation, leads to a degradation in performance.

NERSC is working closely with the High Performing Computing Research Centers (HPCRCs) and with other Energy Science research institutions to help make massively parallel processors a general-purpose computing resource. We envision an early massively parallel production environment becoming feasible within the next few years.

The transition to massively parallel computing will be an incremental process. But as more applications are assimilated to the massively parallel processing environment and as the HPCRCs and vendors succeed in developing appropriate advanced software, NERSC will help accomplish the transfer of these applications to the massively parallel production environment.

Toward this end, NERSC is participating in several activities. We are providing applications programming support to

assist users in migrating, parallelizing, and optimizing code for a variety of massively parallel processing platforms. For instance, we might take an existing vector code and collaborate with the code authors and with interested Energy Research computational scientists to convert the code to a parallel system of choice. At a later date, that same code, now ready for a production environment, could be migrated to a NERSC massively parallel processor. In our role of supporting computational science in Energy Research, we are providing information and insight on relevant systems and applications software such as portable parallel programming tools and linear-algebra packages. Additionally, we are conveying to vendors the computing and algorithmic requirements for important massively parallel Energy Research applications, and we plan to collaborate with them in developing tools and libraries. In particular, we have entered into a Cooperative Research and Development Agreement (CRADA) with Cray Research to collaborate in the development of a massively parallel production environment.

By entering this transitional phase now, we are ensuring that both NERSC and its users are prepared to embark on production parallel computing as soon as the technology is in place.

Evolution of peak speeds (red) and memory sizes (blue) of supercomputers. Massively parallel estimates are based on 1000-node machines or their equivalents. Projections are based on manufacturers' estimates.

guanine base is paired with cytosine and each adenine with thymine. During DNA replication the two strands unwind, and the replication machinery places the appropriate partner opposite each single-stranded template. If the wrong partner is placed opposite a given base, a mutation results.)

In 1970, Grunberger and co-workers⁶ proposed the base-displacement model, which suggested that the AAF, when bound to carbon-8 of guanine, causes the guanine to which it is attached to be removed from its normal position within the B-DNA double helix; the AAF takes the place of the guanine so that it lies within the helix. This distortion of the helix is accompanied by the disruption of base pairing between the AAF-modified guanine and its partner cytosine. In 1971, Fuchs and co-workers⁷ offered a similar model, which they termed insertion-denaturation.

Over the years, a large body of experimental data in solution, albeit not at high resolution, emerged from the two groups in support of their models. In 1989, our computer modeling studies produced a number of molecular views of AAF-modified B-DNA.⁸ We computed structures that place the AAF within the helix and new structures⁹ that place the AAF in the minor groove of the double helix.

We have now combined high-resolution nuclear magnetic resonance studies in solution and molecular mechanics and dynamics calculations to produce both static and animated atomic resolution views of double-stranded B-DNA bound to AAF. These views are in accord with experimental findings. Figure 2 shows the AAF-DNA segment studied.

Methodology

High-resolution nuclear magnetic resonance (NMR) spectroscopy is a technique that employs radio waves and a strong magnetic field to study molecular structure. The method yields many structural details, but the derived information must be combined with molecular mechanics or molecular dynamics calculations to yield views at atomic resolution.

Our NMR results produced a list of distance bounds between hydrogen atoms in the AAF-modified DNA as well as certain other information. The DNA is of the right-handed B-DNA type, but the AAF-

Figure 1. AAF bound to guanine. Ac represents the AAF acetyl group and dR represents the rest of the DNA chain.

modified guanine is rotated from its normal position in B-DNA, known as "anti," to a conformation called "syn," as originally proposed in the base displacement model.⁶ We used the program DUPLEX to incorporate this information into molecular mechanics calculations.^{10,11} In the molecular mechanics approach (for a review of this topic, see Ref. 12), the computer seeks to locate structures, among the endless choices in the flexible DNA molecule, whose potential energies are at a minimum and within the bounds of experimental data (if data are available and included in the calculations). Very large-scale computational searches are needed to achieve this goal because of the multitude of conformational possibilities and because of the multiple-minimum problem, which prevents the guaranteed location of the global—or even all important local—energy minima.

The potential energy is computed via a force field, represented by a number of terms that mimic as accurately as possible the forces that govern the molecular shape. These functions are dominated by fundamental terms that describe attractive and repulsive forces operating between all pairs of atoms. They are parameterized with data obtained for small molecules, and it is assumed that the small-molecule data are transferable to large molecules composed of the same types of atoms. Aqueous solvents and salts are mimicked by indirect treatments but are not represented explicitly in DUPLEX. The force field we employ is based on one that was devised and tested for nucleic acids by Olson and associates.^{13,14}

In our algorithm, the variables that govern the molecular shape are the rotations about flexible single bonds (torsion angles), together with continuous

Figure 2. AAF-modified DNA segment investigated.

puckering variations in the five-membered sugar rings (Fig. 3). Bond lengths, bond angles, and the dihedral angles that vary little are kept fixed. This results in a vast reduction in the number of variables that must be simultaneously optimized, compared with the Cartesian-space approach in which each atom is completely free to move. This can be done at virtually no cost for the applications discussed here, since our interest is focused on the large shape changes that are governed by torsion-angle movements.

Figure 3. Structure of a deoxydinucleoside monophosphate d(CpG), a DNA helix conformational building block. The hexagonal structure is C, the base cytosine, the hexagon-pentagon is G, the base guanine, and the other features are the sugar-phosphate-sugar backbone; the sugars are pentagonal. The arrows denote the rotations about the flexible single bonds, which—

together with the puckering variations in the two sugar rings—constitute the conformational variables in our approach. In the Cartesian-space treatment, by contrast, the number of variables is $3N - 6$, where N is the number of atoms.

We first choose a starting conformation for the molecule under investigation, compute the energy of that conformation, and then use the Powell algorithm¹⁵ to minimize the energy as a function of the flexible variables. In broad overview, the Powell method works as follows: a given variable is set to a starting value and two selected surrounding values, the energy is computed for each of these, a parabola is interpolated from these data, and its minimum is designated as the current variable value; the procedure is repeated for all variables, and this constitutes a single iteration. Numerous iterations are performed. Near the local energy minimum, all variables are flexed simultaneously, and the minimum of the multidimensional paraboloid is interpolated. This continues until a preset condition is met in which further small conformational changes produce very little energy change. At this point, we have a local minimum-energy conformation. We can then choose another starting conformation.

The NMR results give overall information on the helix type (for example, right-handed B-DNA or left-handed Z-DNA), intactness of base pairs, and orientation of the carcinogen-modified base with respect to its adjoining sugar ("anti" as in B-DNA

DNA and Carcinogenesis

DNA (deoxyribonucleic acid) is the molecule that carries the genetic information of living organisms. James Watson and Francis Crick discovered DNA's characteristic double-helix structure in the early 1950s. The information-carrying structural components of DNA consist of four subunits called bases: adenine (A), thymine (T), guanine (G), and cytosine (C). Triplets of such bases, called codons, specify the amino acids that constitute the building blocks of proteins. The relation between the bases that make up a codon and its corresponding amino acid is called the genetic code.

In the double-helix configuration, every A in one of the strands of DNA pairs with a complementary T in the other strand, and every G with a C. During replication, the two intercoiled strands of DNA unwind, and each parent strand becomes the template for the synthesis of a complementary daughter strand. An error in this replication process may cause a mutation in a region of DNA that codes for a particular protein molecule, thereby introducing an error into the structure of that protein when it is produced.

Carcinogens are chemical substances that can initiate a tumorigenic process in a cell. Some carcinogens can cause a mutation by attaching themselves to the DNA molecule, possibly distorting its normal helical shape and thereby interfering with faithful replication. It is critical to understand in molecular detail how this initiation process occurs. One approach is to correlate structural alterations with biological outcome (mutations and cancer). Ultimately, we would like to learn how to predict whether a given substance is mutagenic by observing its effect on DNA structure.

or "syn"). These results are used to select suitable starting conformations for energy minimization. In addition, we use the NMR data to derive an important list of interproton distances that are less than 5.0 Å. These distances are incorporated into the energy minimizations as constraints, via a penalty function; a similar approach is used to seek base-paired partners.^{10,11} Thus, the minimizer is directed toward the region of conformation space that the molecule actually occupies.

This approach is powerful because the energies are optimized in the search for structures that match the distances, but the search is carried out with only torsion angles (and sugar pucker) as the conformational variables. In this way, we are able to locate structures with realistic geometries and energetically favorable conformations. In practice, many dozens of consecutive trials are needed before all experimental distances are satisfied, with judicious selection of weights in the penalty function. A terminal minimization without any constraints ensures that final structures, within the overall bounds of the data, are unconstrained energy minima.

DUPLEX is running with fully optimized code on the CRAY Y-MP and CRAY-2 supercomputers. Currently, up to 12 base pairs of any sequence can be investigated on the CRAY Y-MP and 24 base pairs on the CRAY-2. One trial for a single-stranded dimer (Fig. 3) takes about five seconds of actual computing time; a single-stranded trimer, 15 seconds; a duplex trimer, five minutes; and a duplex dodecamer trial takes about six hours. Until we gained access to the U.S. Department of Energy supercomputers, we were working on a PDP 11/70 that was essentially dedicated to our work. On that computer, a single-stranded dimer trial took about five hours!

In the near future, we hope to adapt DUPLEX to massively parallel computers such as the Intel Paragon with its 2048 processors, which will be available at the new High-Performance Computing Center at the Oak Ridge National Laboratory. Our approach lends itself naturally to parallel computing, since large numbers of energy-minimization trials can be carried out independently. In addition, the various energy functions can themselves be evaluated in parallel within the program.

In the next stage, the static views achieved by molecular mechanics are computationally animated in an environment of water with salt present. This setting corresponds to the situation present in solution and mimics the natural milieu of the cell as well as possible with the current generation of supercomputers. To allow maximum freedom of motion, we do not use the NMR-derived distance bounds in these molecular dynamics simulations. In the molecular dynamics method (see Refs. 16 and 17 for reviews), we follow the movement in time of a molecule under the influence of the force field. In principle, if the trajectory of the molecule were followed for an infinite amount of time, all of conformation space would be surveyed, and the multiple-minimum problem encountered in static minimizations would be solved.

The overall philosophy of molecular dynamics simulations is as follows: the derivatives of the energy functions give the interatomic forces as a function of position, from which the instantaneous acceleration of each atom can be calculated by numerically solving Newton's equations of motion. We assign to the molecule a set of initial velocities, chosen randomly from a Maxwell-Boltzmann distribution at a selected temperature, with a given set of initial coordinates, and then compute new positions and velocities induced by the force field as a function of time. This produces animation. However, the size of the time step employed in most current molecular dynamics methods, including AMBER 3.1 (Ref. 18), which we use, is in the femtosecond (10^{-15} s) range. This is necessary to guarantee numerical stability in the explicit integration of Newton's equations of motion. (A new molecular dynamics algorithm based on the Langevin dynamics equation and the Implicit Euler scheme¹¹ is showing promise in addressing this problem by permitting larger time steps.)

Massively parallel computer architectures that permit simultaneous evaluation of the energy-function terms, while not affecting the time-step problem, will also permit longer simulations. Meanwhile, however, current supercomputers and algorithms can carry out molecular dynamics simulations of DNA for time frames in the range of only picoseconds to

nanoseconds. A 200-ps molecular dynamics simulation of the AAF-modified DNA segment shown in Fig. 2 with 3542 water molecules and 16 sodium ions takes about 140 hours on the CRAY Y-MP. Extending our calculations to the millisecond or even to the second range, where NMR data indicate that important motions occur, is a major computational challenge.

Our computations were carried out on the Cray supercomputers at the Department of Energy National Energy Research Supercomputer Center and at the National Science Foundation San Diego Supercomputer Center. Full details of this work are presented elsewhere.¹⁹

Results and Discussion

Our extensive computational searches, incorporating NMR-derived distances between hydrogen atoms on the AAF-modified DNA segment of Fig. 2, produced two structures that together provide good agreement with the NMR data. These structures are shown in Fig. 4. The structure of Fig. 4a represents roughly 80% of the conformations present in solution, with the remainder represented by Fig. 4b. In both these forms, part of the AAF protrudes into the B-DNA minor groove, but it is also stacked with an adjacent base in Fig. 4a, as in the base displacement⁶ or insertion-denaturation⁷ models. The AAF-modified guanine is displaced from its normal position and is not paired with its partner cytosine in either form. The DNA is more severely bent in the structure of Fig. 4b.

During the animation with molecular dynamics simulations, the NMR-derived features were preserved on average. Indeed, although we did not use the NMR-derived distances to constrain the motions, we did monitor the relevant separations between hydrogens. In some cases, their average values over the duration of the simulation showed improved agreement with the data. Thus, the dynamic structures more accurately reflect the average structure in solution, and the molecular dynamics serve to improve the quality of the structures computed as well as to provide valuable insight into the motions.

The simulations starting from the structure of Fig. 4a show the AAF moving so that the extent to which its distal ring protrudes into the minor groove varies. This

form also bent strongly during the 200-ps simulation. In the more bent starting structure of Fig. 4b, on the other hand, the DNA helix becomes less bent in this time frame, and the bulk of the AAF continues to protrude into the groove.

In combination, these simulations suggest that in the distorted AAF-modified B-DNA double helix, in which bending and flexing motions occur, the carcinogen moves between positions where it completely protrudes into the groove to more internal positions where it is stacked with a neighboring base. However, the simulations are orders of magnitude too slow to reveal the conversion between the two forms. Furthermore, both the dynamics and the NMR data indicate that a base pair next to the modification site shows increased mobility and that the base pair at the lesion site remains ruptured.

These distorting features may be responsible for the high susceptibility of this particular AAF lesion to repair and also for its mutagenicity.⁴ The repair enzymes, it is currently believed, are likely to recognize a perturbed double helix and to cleave out the AAF. In doing so, however, the process may take an error-prone pathway that sometimes results not only in removal of the AAF but also finally in a mutation. Various mechanisms for the mutational pathways have been discussed,^{20,21} including the possibility that the abnormally oriented, AAF-modified "syn" guanine fails to act as a faithful template during DNA replication. The mutagenicity is very likely a critical event in the pathway to tumorigenesis.⁵

Conclusions

We have used a combination of high-resolution NMR studies and molecular mechanics and dynamics calculations to produce both static and dynamic atomic resolution views of DNA modified by AAF, an aromatic amine that is tumorigenic in biochemically activated form. These views have provided some insights at the molecular level into how this substance may trigger an initial event in carcinogenesis. The computational endeavor involved represents a supercomputing Grand Challenge that is now addressable.

Future supercomputers should make it possible to carry out the animations for

Figure 4. Static views of (a) major and (b) minor conformers of AAF-modified DNA segment. The AAF is yellow and white (Ac moiety), its attached guanine with sugar-phosphate DNA backbone is red, and the rest of the DNA is blue. The left panels are dot surface views, and the right panels are space-filling. Color graphics are copyright 1991, Regents of the University of California, San Francisco Computer Graphics Laboratory.

much longer times than the currently limited picosecond-to-nanosecond range. Longer simulations are very important, since NMR data indicate that significant movements can occur even on a scale of seconds. In addition, to represent more realistically the environment of the cell, we should include important proteins in the simulations. To handle the vastly longer time frames required by more realistic animations, as well as the far greater number of atoms when proteins are included, we will need the much more powerful supercomputers that are already in various stages of planning and development.

Acknowledgments

This work was supported by DOE DE-FG02-91ER60931 and DE-AC05-84OR21400 with Martin-Marietta Energy Systems, National Institutes of Health NIH CA28038, RR06458, CA35251, and RR03317, and National Science Foundation DMB 8416009 and DMB 8611927. Computer graphics images were created by Dr. Teri Klein using the software UCSF MidasPlus from the Computer Graphics Laboratory of Professor Robert Langridge at the University of California, San Francisco. We thank Professor Robert Shapiro, Chemistry Department, New York University, for many helpful discussions with insightful perspectives.

References

1. L. Rehn, "Blasengeschwulste bei Fuchsin-Arbeitern," *Arch. Klin. Chir.* **50**, 588-600 (1895). Reprinted in translation in M. B. Shimkin, *Some Classics of Experimental Oncology: 50 Selections, 1775-1965*, NIH Publication No. 80-2150, pp. 45-51 (1980).
2. S. Nesnow, M. Argus, H. Bergman, K. Chu, C. Frith, and E. Weisburger, "Chemical Carcinogens. A Review and Analysis of the Literature of Selected Chemicals and the Establishment of the Gene-Tox Carcinogen Data Base," *Mutation Research* **185**, 1-195 (1987).
3. R. H. Wilson, F. DeEds, and A. J. Cox, Jr., "The Toxicity and Carcinogenic Activity of 2-Acetylaminofluorene," *Cancer Res.* **1**, 595-608 (1941).
4. F. A. Beland and F. F. Kadlubar, "Metabolic Activation and DNA Adducts of Aromatic Amines and Nitroaromatic Hydrocarbons," in *Handbook of Experimental Pharmacology: Chemical Carcinogenesis and Mutagenesis*, C. S. Cooper and P. L. Grover, Editors (Springer-Verlag, Heidelberg, Germany, 1990), pp. 267-325.
5. J. M. Bishop, "Retroviruses and Oncogenes," *Bioscience Reports* **10**, 473-491 (1990).
6. D. Grunberger, J. H. Nelson, C. R. Cantor, and I. B. Weinstein, "Coding and Conformational Properties of Oligonucleotides Modified with the Carcinogen N-2-Acetylaminofluorene," *Proc. Nat. Acad. Sci.* **66**, 488-494 (1970).
7. R. Fuchs and M. Duane, "Changes of Stability and Conformation of DNA Following the Covalent Binding of a Carcinogen," *FEBS Letters* **14**, 206-208 (1971).
8. R. Shapiro, B. E. Hingerty, and S. Broyde, "Minor-Groove Binding Models for Acetylaminofluorene Modified DNA," *J. Biomolecular Structure and Dynamics* **7**, 493-512 (1989).
9. D. Norman, P. Abuaf, B. E. Hingerty, D. Live, D. Grunberger, S. Broyde, and D. J. Patel, "NMR and Computational Characterization of the N-(Deoxyguanosin-8-yl) aminofluorene Adduct [(AF)G] Opposite Adenosine in DNA: (AF)G[syn]·A[anti] Pair Formation and Its pH Dependence," *Biochemistry* **28**, 7462-7476 (1989).
10. B. E. Hingerty, S. Figueroa, T. Hayden, and S. Broyde, "Prediction of DNA Structure from Sequence: A Build-up Technique," *Biopolymers* **28**, 1195-1222 (1989).
11. T. Schlick, B. E. Hingerty, C. S. Peskin, M. L. Overton, and S. Broyde, "Search Strategies, Minimization Algorithms, and Molecular Dynamics Simulations for Exploring Conformational Spaces of Nucleic Acids," in *Theoretical Biochemistry and Molecular Biophysics*, David L. Beveridge and Richard Lavery, Editors (Adenine Press, Schenectady, NY, 1990), pp. 39-58.
12. D. H. J. Mackay, A. J. Cross, and A. T. Hagler, "The Role of Energy Minimization in Simulation Strategies of Biomolecular Systems," in *Prediction of Protein Structure and the Principles of Protein Conformation*, G. D. Fasman, Editor (Plenum Press, New York, 1989), pp. 317-358.
13. E. R. Taylor and W. K. Olson, "Theoretical Studies of Nucleic Acid Interactions. I. Estimates of Conformational Mobility in Intercalated Chains," *Biopolymers* **22**, 2667-2702 (1983).

14. A. R. Srinivasan and W. K. Olson, "Polynucleotide Conformation in Real Solutions—A Preliminary Theoretical Estimate," *Fed. Proc.* **39**, 2199 (1980).
15. M. Powell, "An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives," *Comput. J.* **7**, 155–162 (1964).
16. M. Karplus and G. Petsko, "Molecular Dynamics Simulations in Biology," *Nature* **347**, 631–639 (1990).
17. J. A. McCammon and S. C. Harvey, *Dynamics of Proteins and Nucleic Acids* (Cambridge University Press, New York, 1987).
18. U. C. Singh, P. K. Weiner, J. Caldwell, and P. A. Kollman, AMBER, Version 3.1 (University of California, San Francisco, 1988).
19. S. F. O'Handley, "Structural Analysis of d(CCACC(AAF-G)CACC)·d(GGTGCGTGG) by NMR Spectroscopy," Ph.D. Thesis, Chemistry Department, University of Rochester, Rochester, NY (1991); S. F. O'Handley, D. G. Sanford, R. Xu, C. C. Lester, B. E. Hingerty, and S. Broyde, *Biochemistry* **32** (1993), in press.
20. T. A. Kunkel, "Misalignment-Mediated DNA Synthesis Errors," *Biochemistry* **29**, 8003–8011 (1990).
21. S. Broyde, B. E. Hingerty, R. Shapiro, and D. Norman, "Unusual Hydrogen Bonding Patterns in 2-Aminofluorene (AF) and 2-Acetylaminofluorene (AAF) Modified DNA," in *Nitroarenes. Occurrence, Metabolism and Biological Impact* (Plenum Press, New York, 1990), pp. 113–123.

We know little about what happens to the remaining half, particularly with respect to how the global carbon dioxide sink is partitioned between the ocean and the terrestrial biosphere.

All these uncertainties, combined with the practical impossibility of comprehensive, laboratory-based climate experiments, have created a unique and vital role for computer-based climate simulation experiments. By enabling climate scientists to quantify their models and to test them in detail against a wide array of observational datasets, such experiments can provide an objective basis for understanding the climate system and for estimating future changes.

The bewildering diversity of physical processes that together determine a given climate regime occurs over a wide range of space and time scales that are activated by nonlinear couplings. Models that attempt to encapsulate these processes are necessarily complex and overtax the capabilities even of current supercomputers. This disparity will become more acute with the next generation of models. Many estimates of future climatic change are based on atmospheric models run with limited chemical reactions, simplified oceans, and essentially no representations of terrestrial and marine biosystems.

Such limitations, together with the delayed responses currently evident in the global temperature record, point clearly to the need for more comprehensive and better-tested models. Future models of the climate system will need increased spatial resolution, enhanced representation of processes, and full coupling of additional major components of the earth system, including dynamic representation of the oceans, ecosystems, and interactive biogeochemistry. The dramatic range of physical processes and space and time scales embodied in such models, together with the enormous demands they make on throughput and data storage requirements, indeed pose a computational Grand Challenge.

For example, a comprehensive climate calculation may require an ocean grid of 0.3 deg latitude/longitude, an atmospheric grid of 1.0 deg, and 40 meshpoints in the vertical direction. Simulations

should be extendible to at least 1000 years, or about two circulation periods for the deep ocean, and should represent convective, advective, chemical, and other physical and biogeochemical processes. Such a calculation, using current numerical algorithms, would likely require over ten billion billion (10^{19}) arithmetic operations, taking over 100 days on a computing platform that delivers sustained tera-flops performance.

Pioneering simulation experiments such as those described above will play an important role in advancing the science of earth systems, and much can be learned from supporting studies with coarser resolutions or less comprehensive physics. Nevertheless, it is sobering to get a sense of how far we have to go. It is clear that substantially faster algorithms are imperative if we are to move forward to advanced simulations of the climate system during this decade.

In addition to placing greater demands on the main computing engine, comprehensive climate calculations require much greater storage capacity and data throughput than are available today. Stored data are used for visualization, program restarts, boundary conditions, diagnostics, and input to auxiliary calculations. To represent as few as five physical quantities at a single time level on a fine grid requires over 10 billion (10^{10}) bits of (uncompressed) storage capacity. Modern compression techniques can offer some relief if the data are to be used only for diagnostic purposes. Still, a formidable amount of storage is required.

One of the greatest challenges facing climate modelers is to understand the vast amounts of data produced by computer simulations. In fact, few disciplines rely more heavily on the need to visualize than does climate modeling. What is of interest is not so much the data per se but the information they contain. To make sense of all these data, we use software packages that use lines, shapes, and color gradations to depict physical quantities such as temperature and velocity. It is becoming commonplace to use workstations or other high-performance graphics engines, rather than the supercomputer itself, to visualize the data generated by models.

The US/GCRP and DOE CHAMMP Programs

The centerpiece of the U.S. effort to address global climate change is the U.S. Global Change Research Program (US/GCRP), developed by the Committee on Earth and Environmental Sciences of the Federal Coordinating Council for Science, Engineering, and Technology (FCCSET). The principal goal of the US/GCRP is to gain a predictive understanding of the physical, geological, chemical, biological, economic, and social processes that regulate the earth's environmental system. This understanding is necessary if we are to establish a scientific basis for formulating national and international policies related to natural and human-induced changes in the global environment. Nineteen federal agencies and departments participate in the US/GCRP, representing a new paradigm for interagency cooperation in the research arena. The implementation framework for the US/GCRP is organized around four Program Objectives: (1) Observations,

Data, and Information Management, (2) Process Research, (3) Integrated Modeling and Prediction, and (4) Assessment. Here we are concerned primarily with LLNL's contributions to the Department of Energy's (DOE's) role in Integrated Modeling and Prediction.

DOE is a pioneer in research on global change, having launched the first federal program in 1978. Today, the DOE's Office of Health and Environmental Research of the Office of Energy Research operates a long-standing core program that includes research on the global carbon cycle, studies of the effects of carbon dioxide fertilization on vegetation, and recent highly focused programs such as the Atmospheric Radiation Measurement (ARM) Program, the Program for Climate Model Diagnosis and Intercomparison (PCMDI), and the Computer Hardware, Advanced Mathematics, and Model Physics (CHAMMP) Program. CHAMMP, launched in collaboration with the High Performance Computing and Communication Initiative, is

The Greenhouse Effect

The sun is the energy source that drives the earth's global climate system. Most of the gases that make up the earth's atmosphere are nearly transparent to the visible radiation emitted by the sun. When this radiation falls on the earth's surface, some of it is reflected back into space and a portion is absorbed. The absorbed portion heats the earth's surface, which then re-radiates at longer (infrared) wavelengths. However, some of the same gases that are transparent to visible light strongly absorb or scatter radiation at these longer wavelengths, thereby preventing it from radiating away to space and effectively "trapping" it in the atmosphere. These are the "greenhouse" gases, so-called because their effect on global temperature is like that of a greenhouse.

Although present in relatively small concentrations (less than 1% of the total), greenhouse gases play a critical role in the earth's global heat budget—the result of subtracting net outgoing radiation from net incoming radiation. When the net global temperature of the earth's atmosphere is in balance, neither increasing nor decreasing, the outgoing radiation just equals the incoming radiation. An imbalance in the heat budget can have long-term heating or cooling effects on the atmosphere.

One of the most important greenhouse gases is carbon dioxide, which is emitted in copious amounts by many kinds of industrial activity such as fossil-fuel burning. Other important industrial greenhouse gases are methane, nitrous oxide, and chlorofluorocarbons, the latter widely used as a refrigerant and (until recently) as a propellant in aerosol products. There is evidence that in the more than two hundred years since the Industrial Revolution began, the global level of carbon dioxide and other greenhouse gases in the atmosphere has risen slowly but steadily.

Accompanying this rise has been an apparent increase of about one degree in the global temperature of the earth—although the evidence for this effect is not so clear-cut and is disputed by some climatologists. More important, there is as yet no direct way of linking a temperature rise to changing carbon dioxide concentrations. One reason that we are strongly motivated to build better models of the earth's climate system is to understand better the many complex processes that enter into the earth's radiation budget and their potential effects on climate and other aspects of man's environment.

aimed at developing a new generation of climate system models that capitalize on advances in massively parallel computing capabilities. CHAMMP activities, which reach across DOE, the national laboratories of other agencies, and the academic community, are organized into CHAMMP Science and Model Development Teams. Early phases of the model development efforts probe issues connected with using both finite-difference and Galerkin (spectral) numerical schemes to adapt existing climate models for efficient execution on both MIMD (multiple-instruction, multiple data) and SIMD (single-instruction, multiple-data) parallel architectures.

Capability for Modeling Climate Systems

Our capabilities are based on general circulation models of the ocean and atmosphere together with other computational components that represent a variety of physical, chemical, and biological processes (Fig. 1). We are developing versions of these components that are suitable for contemporary, high-performance computing architectures and are constructing a software framework that will enable us to couple groups of modules according to the application of interest. We describe here the atmospheric and oceanic general circulation models and their performance in a parallel-computing environment.

The Atmospheric Model

We have chosen the model originally developed by Arakawa and collaborators at UCLA.¹ The code uses a conservative finite-difference computational hydrodynamics scheme to explicitly advance in time the horizontal momentum, mass, pressure, and moisture on a three-dimensional staggered mesh. The model computes many additional physical processes, such as moist convection and radiation, along independent vertical columns. Production runs typically use a few thousand horizontal grid cells at each of ten or so vertical levels (Figs. 2 and 3). Finer-resolution experiments are planned for the near future.

To parallelize this model, we have implemented a two-dimensional latitude/longitude domain decomposition message-passing strategy. Subdomains consist of contiguous vertical columns of rectangular cross section in latitude/longitude, extending from the earth's surface to the upper atmosphere (Fig. 4). Our choice to decompose in only two dimensions was based on the fact that column processes (such as radiation and convection) strongly couple the elements within the column and do not naturally parallelize along it. Additionally, the number of meshpoints in the vertical direction is usually small. We chose two-dimensional over one-dimensional decomposition

Figure 1. Basic components of a comprehensive climate systems model. The physical processes that together control climate are divided into six major components. Each component encompasses a variety of physical process representations. Components are coupled by exchanges of data that may be multidimensional. For example, the atmospheric component is driven in part by the temperature distribution at the sea surface (a two-dimensional field), which in turn is determined by the oceanic module, and the atmospheric chemistry component is driven in part by the wind distribution (a three-dimensional field), which is determined by the atmospheric module. Only some of the principal coupling mechanisms are shown explicitly.

Figure 2. Snapshot of atmospheric motion near an altitude of about 10 km, from a simulation with the parallelized UCLA atmospheric general circulation model. This image, generated with the AVS graphics package, shows the trajectories of imaginary air parcels over a fixed time interval, the length of the resulting "streaklines" being related to the wind velocity along the path. The streaklines are also color-coded, red depicting higher windspeed and blue depicting lower windspeed. This image is from a simulation of conditions in October 1982.

Figure 3. Temperature distribution on a near-horizontal surface at an altitude of about 2 km, from a simulation with the parallelized UCLA atmospheric general circulation model. This field, corresponding to a simulated instant in October 1982, was generated by the graphics package developed under the auspices of the Program for Climate Model Diagnosis and Intercomparison (PCMDI). Colors range from blue (cooler) to red (warmer).

because the latter does not provide the necessary concurrency for mesh sizes of interest and has an asymptotically larger communications cost.

Subdomains are assigned to processors in a deterministic manner, and data are transmitted between subdomains in the form of messages. To minimize the amount of message traffic, the adjacent subdomains overlap. For example, each subdomain includes two border points at the ends of each row and column in order to accommodate a fourth-order difference method. The total extent of subdomain overlap is then four points at the end of each row or column and sixteen points at each corner. Once the quantities at the border of a subdomain are updated, their values are passed to the neighboring subdomain(s) and serve as boundary values for the ensuing timestep. Diagonal communications are accomplished by first passing messages horizontally and then vertically. Having the subdomains overlap allows this information to be passed only once and stored in the region requiring its use.

The Oceanic Model

The baseline ocean model for our climate systems modeling capability is the Geophysical Fluid Dynamics Laboratory Modular Ocean Model (MOM) of the National Oceanographic and Atmospheric

Administration (NOAA), which is based on the model of Bryan.² This is a three-dimensional, finite-difference-based, primitive equation model of the circulation and thermodynamics of the world ocean, including detailed bathymetry and coastline boundaries. To permit a larger time step in the mostly explicit time-advance scheme, we use a rigid-lid upper boundary condition to filter fast external gravity waves from the system. Consequently, we must solve a Poisson problem for the vertically averaged velocity stream function at each timestep. Temperature, salinity, and the departure of the velocity field from its vertical average are advanced by using fully explicit, conservative staggered-mesh hydrodynamics schemes. A typical grid count for production runs with this model is of the order of one million zones (Fig. 5).

Earlier attempts to parallelize ocean models, aimed at the shared-memory, vector multiprocessor environment,³ were based on one-dimensional domain decomposition with concurrent execution of longitude-depth slabs and vectorization on the longitudinal coordinate. In a pilot study, we carried out an analogous procedure using the shared-memory constructs on the LLNL BBN-TC2000. As expected from analytical performance estimates, this procedure is viable for a small number of processors (ten or fewer). For 64 processors,

Figure 4. Two-dimensional domain decomposition used for parallelizing the atmospheric general circulation model. While a uniform distribution of subdomains is shown, any logically rectangular pattern may be used. One processor is typically assigned to each subdomain to carry out the time advance for interior points. Messages are passed between processors to exchange border data from neighboring subdomains. On certain high-latitude coordinate lines, a portion of the time-advance algorithm requires interprocessor communication across all subdomains at that latitude, resulting in nonlocal message traffic. Subdomain configuration and size may be tailored to partly offset load imbalances generated by temporal variation of computational intensity within the atmospheric model.

parallel speedup approaches near-saturation because one-dimensional decomposition demands excessive communications overhead. Furthermore, because horizontal resolution in the ocean models is typically of the order of $2 \text{ deg} \times 2 \text{ deg}$, one-dimensional decomposition does not scale well for more than 100 processors. After exploratory studies with a simpler prototype ocean model using two-dimensional horizontal decomposition, we decided to implement the latter paradigm in the MOM code.

Portability

The rapid evolution of computing technology has produced a multiplicity of platforms available from a variety of vendors. Because we wish to carry out scientific studies in a reliable production environment while developing advanced computational versions for high-performance systems, it is of paramount importance to maintain a portable source code. This is especially true when the code is a rapidly evolving one, for which multiple sources would place unreasonable burdens on version control.

We have addressed two issues that affect portability: dynamic memory management and interprocess communication. Dynamic memory management is the ability to allocate and deallocate storage as needed during a calculation. Allocating memory at run time, rather than at compile time, allows memory to be used more efficiently and makes it much easier to resize subdomains dynamically (for example, when attempting to dynamically balance the load between processors). Because the language constructs for dynamic memory management are nonstandard, we are using the M4 preprocessor to implement macro constructs that are processed prior to compilation. This practice confines nonportable constructs to the bodies of the macros, allowing them to be easily modified to accommodate new architectures.

Message-passing constructs also vary from platform to platform. Each parallel architecture has its own message-passing library, and message-passing software designed to accommodate a variety of platforms is available. Here, too, we have

Figure 5. Horizontal temperature distribution at a depth of 60 m from a simulation with the Semtner and Chervin ocean general circulation model. Horizontal grid resolution for this experiment was $0.5 \text{ deg latitude} \times 0.5 \text{ deg longitude}$. The model was driven by prescribed surface atmospheric temperature, precipitation, and wind fields corresponding to annual-averaged observed climatic conditions. For the ocean component of the climate systems model, we implemented one- and two-dimensional domain decomposition paradigms on both shared-memory and distributed-memory platforms. Temperatures shown on the color bar increase from left to right.

chosen to construct macro commands for sending and receiving messages. These constructs, along with CPP precompile directives, allow movement between the CRAY, the BBN-TC2000, the Thinking Machines CM-5, and a Sun or IBM workstation cluster with a simple change of only three parameters.

Parallel Performance of Atmospheric Model

The parallel performance of any code depends on a number of factors. First, it is important to balance the computational load so that some processors do not idle while others are doing most of the work. It is also important to minimize interprocessor communications costs, again to avoid significant processor idle time. While some systems will allow message traffic simultaneously with computation, it is nevertheless informative to estimate communications costs by assuming that the two do not overlap.

One can define the speedup obtained from parallelization as the wall-clock time for a serial calculation divided by the wall-clock time for the same calculation in parallel. A perfectly efficient calculation will have a parallel speedup equal to the number of processors applied to the problem. Assuming the arithmetic time to be inversely proportional to the number of active processors, the speedup becomes a function of the communications time relative to the time to carry out arithmetic computations.

For the two-dimensional domain decomposition strategy used in the atmospheric model, the speedup S takes the form

$$S = \frac{N}{1 + c \left(\frac{N}{n} \right)^{1/2}},$$

where N is the processor count, n is the number of horizontal zones, and c is a machine- and algorithm-dependent constant. The formula assumes a load-balanced computation having subdomains of nontrivial size and reflects well-known scaling of communications overhead with subdomain surface-to-volume ratio for time-explicit domain decomposition methods. This is correct for the explicit time-advance scheme used for the hydrodynamic fields.

The longitudinal global filtering operation that is applied to certain terms in the momentum equation modifies the speedup scaling. The filtering operation is designed to eliminate unstable computational modes that originate from the convergence of meridians in polar regions. This introduces the need for non-nearest-neighbor subdomain communications. As a result, the parameter c is modified to the form $c' = c(1 + \alpha n^{1/2})/(1 + \beta n^{1/2})$, where α and β are constants that depend on the form of filter arithmetic. It should also be noted that c is inversely proportional to the amount of arithmetic required to implement the column physics parameterizations, which require no interprocessor communications.

Figure 6 shows a first characterization of how our atmospheric code performs on the Lawrence Livermore National Laboratory's BBN-TC2000. As the processor count increases, the speedup increases in good agreement with the predicted scaling formulas. These early results are from unoptimized code; we expect substantial performance increases from refined code. The agreement with speedup scaling predictions allows us to estimate with some confidence the parallel performance of next-generation climate systems models on future-generation platforms.

Figure 6. Observed and predicted speedup of the parallelized atmospheric model on the LLNL BBN-TC2000 platform. Data points are from independent runs of a fixed number of timesteps; curves are from the analytical performance model for the domain-decomposition, message-passing paradigm. The horizontal resolution in these tests was 4 deg latitude $\times$ 5 deg longitude. The blue curve is for hydrodynamics only (without filtering), the red curve includes filtering, and the yellow curve additionally includes column physics. It can be seen that filtering and column physics decrease and increase, respectively, the parallel efficiency of the model.

Toward the Future

As this article goes to press, we are readying model components for use on the Thinking Machines CM-5. Soon to follow will be transfer to the Intel Delta and Paragon systems. The use of portable memory management and interprocessor communications macros makes these ports relatively simple. For example, a single source code operates on the TC2000, a CRAY, or a workstation cluster.

With the support of LLNL Laboratory-Directed Research and Development and interagency funding, the process of constructing a computational framework that will allow autonomous packages to work together is just beginning. The framework will enable us to coordinate the modules making up the climate systems model, allowing us to add or substitute physics packages with a minimum of effort. A runtime database will allow components to exchange data with one another. We plan a graphical user interface to enable the scientist to define the problem at run time, monitor progress, and analyze results.

Our work to date suggests that advances in parallel and distributed supercomputing will return large dividends to research on global change. One goal of the CHAMMP program is to achieve, within this decade, a ten-thousand-fold increase in sustained computational throughput over current levels of a few hundred megaflops. While a substantial challenge, a combination of throughput increases and algorithm enhancements makes this goal appear feasible. However, a more difficult challenge remains. A faster, more comprehensive climate systems model is not necessarily a better one. Given the prospect of dramatically increased throughput, how can it best be applied? Substantial scientific issues remain to be addressed before we can say with confidence just how much we need to refine spatial resolution or how accurately physical processes are represented. Such issues are likely to remain with us well beyond the present decade.

Acknowledgments

A number of scientists at Lawrence Livermore National Laboratory have contributed to this project, including J. Bolstad, J. C. Brown, P. G. Eltgroth, P. Rouvillois, and M. F. Wehner.

The authors thank D. E. Shumaker and M. Collette for providing Figure 2, D. E. Shumaker and J. Albritton for providing Figure 3, and C. C. Covey for providing Figure 5.

We have been working in collaboration with the Department of Atmospheric Sciences at the University of California at Los Angeles. We thank A. Arakawa, C. R. Mechoso, J. A. Spahr, and J. D. Farrara for providing the atmospheric general circulation model and participating in very useful discussions.

References

1. A. Arakawa and V. Lamb, in *Methods in Computational Physics* 17 (Academic Press, New York, 1977), pp. 173-265.
2. K. Bryan, *J. Comp. Phys.* 4, 347 (1969).
3. A. Semtner, in *Advanced Physical Oceanographic Numerical Modeling* (Reidel Publishing, Dordrecht, 1986), pp. 187-202.

Collaborating to Solve a Major Data Storage Challenge

Large scientific, engineering, and commercial applications are straining storage and networking facilities, a condition compounded by the larger memories and faster speeds of new supercomputers, massively parallel processors, and high-performance workstations. An urgent need exists to bring large-scale hierarchical storage system performance and functionality into better balance with modern processing capabilities and new application requirements. The National Storage Laboratory (NSL), a collaborative, high-performance, storage testbed project centered at Lawrence Livermore National Laboratory (LLNL), has been organized to investigate, demonstrate, evaluate, and commercialize technologies that promise to remove network computing bottlenecks and provide needed storage system functionality. The NSL is located within the National Energy Research Supercomputer Center at LLNL and will be integrated into the Energy Sciences Network (ESNET) and LLNL's Open LabNet environment. The figure is a conceptual diagram of the NSL system architecture.

Three initial applications have been chosen to help set priorities, evaluate software and hardware performance, and study needed improvements.

- The Program for Climate Model Diagnosis and Intercomparison.
- Magnetic Fusion Energy modeling and experimentation.
- Digital storage and replay of computer-generated scientific movies.

The approach of the NSL is based on the premise that no single company, government laboratory, or research organization has the ability to confront all the system-level issues that must be resolved before there is significant advancement in storage systems technology. In an attempt to find solutions to these kinds of storage problems, LLNL has joined with industry and other laboratories and agencies. The primary objectives of the NSL are (1) to leverage Department of Energy laboratory

experience, expertise, and applications and that of industry to accelerate the development of U.S. technology for local and nationwide storage-system architectures and functionality, and (2) to facilitate and speed the availability of improved systems to the Department of Energy and U.S. industry.

The participants, aside from LLNL, are the IBM Federal Systems Company, IBM ADSTAR, Ampex Corporation, DISCOS Division of General Atomics, Maximum Strategy Incorporated, Network Systems Corporation, Zitel Corporation, IGM, PsiTech, CHI Systems, Cray Research, and the four National Science Foundation Supercomputer Centers. The industrial partners will provide over two million dollars worth of equipment and over three million dollars worth of personnel over the life of the project. Los Alamos National Laboratory and Sandia National Laboratories are also expected to join the collaboration under anticipated Technology Transfer Initiative funding. The NSL will encourage further collaboration with other interested parties. The experience gained through the NSL will be fed back into commercial products useful to the Department of Energy and U.S. industry, as well as to national storage system standards working groups.

Conceptual overview of the NSL architecture.

To address the problems outlined above, the Pacific Northwest Laboratory established the Theory, Modeling, and Simulation program in its Molecular Science Research Center. The program combines rigorous studies of fundamental processes (such as reaction, solvation, absorption, and crystallization) in model molecular systems with computational simulation of these processes in complex environmental systems. Current research includes solvation and reaction in aqueous solutions, structure and chemistry of minerals, and redesign of enzymes for bioremediation. These efforts require us to develop molecular modeling software that takes full advantage of advances in computer technology, including high-performance, massively parallel supercomputers. A research and development program has been initiated to develop the new generation of molecular modeling software required.

To support this work, the Chemistry Division of the DOE Office of Basic Energy Sciences has provided us with high-performance workstations, and the DOE's Office of Scientific Computing has allocated time on their supercomputers at the National Energy Research Supercomputer Center and at Florida State University.

Learning How Pollutants Behave in Water

Solution chemistry is a key factor in the transport of contaminants in soils and groundwater. How a contaminant migrates depends on its solubility in water and on its absorption on soil particles such as clay minerals. Chemical reactions can alter its molecular form, making it more or less soluble or harmful. We need a better understanding of solvation and reaction in aqueous solutions if we are to develop improved models of how pollutants behave in the subsurface.

Alkali and alkaline earth ions are important constituents in the waste tanks at the DOE's Hanford site. Cesium-137 and strontium-90 are the dominant radionuclides in the tanks. To gain a better understanding of the chemistry of these ions in aqueous solutions, we are developing computational models that describe the structure of water around the ions. This is the first step towards predicting their interactions with other species in

solution, such as minerals in the soil. We use the Molecular Dynamics method to simulate ionic aqueous clusters and solutions.¹ Equilibrium properties are obtained by averaging over many different configurations.

The reliability of these simulations depends on the accuracy of the water-water and ion-water interactions. Older simulations expressed these interactions as sums over pairs of atoms or sites in a molecule. However, this treatment neglects the polarizing effect of an ion on the distribution of the electrons in water molecules and vice versa. We include this effect by placing induced point dipoles at each charge site.² Each dipole is equilibrated with the fixed charges and with other dipoles in the system. This polarization potential is a many-body interaction that incorporates the effects of three-body, four-body, and higher interactions. To account for the repulsive many-body interactions, which are more important at short range, we include three-body exponential repulsions between the ion and pairs of water molecules. This model of ionic solutions is far more reliable than the two-body models, but it greatly increases the computational requirements.

We have developed potentials for the interaction of cesium ions with water that reproduce experimental values of the heats of formation and solvation enthalpy of small clusters $\text{Cs}^+(\text{H}_2\text{O})_n$, $n = 1-4$. We have then used these potentials in Molecular Dynamics simulations to explore the detailed structure of water molecules around the ion. We find that the water molecules are loosely arranged around the ion in concentric clusters called solvation shells, with about eight water molecules in the first solvation shell. Because the Cs^+ -water interaction is weak, we find that water molecules are rapidly exchanged between the first and second solvation shells. The simulations produced the two instantaneous configurations of water molecules in the first solvation shell around a cesium ion (Fig. 1). Both structures have eight water molecules in their first solvation shell. As can be seen, the water molecules are not tightly clustered around the cesium ion.

We have developed similar models for anion-water clusters.² Simulations of

Figure 1. Instantaneous configurations of the first solvation shell of water molecules around Cs^+ , produced by a Molecular Dynamics simulation of the ionic solution. The two snapshots are taken at two different times during the simulation.

small fluoride (F^-) water clusters reveal interesting differences in the solvation of anions. Figure 2 shows instantaneous configurations resulting from simulations of $\text{F}^-(\text{H}_2\text{O})_n$, $n = 1-4$. In cation-water clusters, such as those of cesium, the water binds to the ion with the oxygen atom closest to the ion. As water molecules are added to the cluster, they tend to arrange themselves more or less symmetrically around the ion. In anion-water clusters, in contrast, water binds to the anion with its hydrogen atoms closest to the ion. As water molecules are added, they tend to arrange themselves so as to retain, so far as possible, the favorable hydrogen-bonding interaction between the water molecules. For small clusters, the arrangements can be very asymmetric.

Tracking Pollutants in the Soil

Over 80% of the earth's continental crust is composed of the elements silicon, oxygen, and aluminum. When combined, they form two abundant classes of compounds known as the primary silicates and the aluminosilicates. In combination with organic material and various trace elements, these minerals aggregate to form the complex, ever-changing porous medium known as soil. To develop efficient and effective remediation strategies for contaminated soils, we must be able to accurately predict the migration of such pollutants through the soil to the water table or to water bodies on the surface.

Such predictions have at least three fundamental components. First is a characterization of geological features, including mineral composition and

hydrological features. Second is an analysis of the hazardous materials. Third is the construction of a detailed model describing how the waste plume disperses over time. A key component of such a model is a detailed description of the interactions of exposed rock and soil mineral surfaces with pollutants and their degradation products.

Several theoretical techniques can be used to investigate pollutant-water-mineral interactions. The most fundamental of these are *ab initio* approaches. We have applied a state-of-the-art solid-state method of quantum chemistry known as *ab initio* periodic Hartree-Fock (PHF) theory. The PHF method provides self-consistent field (SCF) solutions to the Hartree-Fock-Roothan equations subject to periodic boundary conditions. It is fully implemented in the Crystal computer program.³ Crystal can treat systems that are periodic in from zero to three dimensions, corresponding to molecules, polymers, finite-thickness slabs, and solids. One advantage of the PHF approach is that it avoids the inherent limitations of methods that model crystalline systems as finite molecular clusters.

We have applied the PHF method to clay minerals, materials that strongly influence the migration of subsurface pollutants. Clays are among the most common decomposition products of aluminosilicates. Because of their large surface areas, these materials play a dominant role in ion-exchange processes and in the transport of chemical species in soils. However, they are difficult to characterize experimentally because their particles are so small (typically 2 to 10 μm in diameter) and because

Figure 2. Instantaneous configurations of $F^-(H_2O)_n$ clusters, $n = 1-4$, produced by Molecular Dynamics simulations.

they tend to exhibit both structural and substitutional disorder. Figure 3 illustrates the structural characteristics typical of these materials in kaolinite ($Al_2Si_2O_9H_4$). In general, clays are layered hydrous aluminosilicates that adsorb atomic and molecular species either between their layers, at external surfaces parallel to their layers, or on their external edges.

We have recently completed a study of kaolinite,⁵ pyrophyllite, and talc. All of these materials are of low symmetry and contain 20 to 40 heavy atoms per unit cell. As a consequence, *ab initio* calculations burden even the most powerful computers available today. Of particular chemical interest is the charge-density distribution, electrostatic potential, and electric fields between the clay layers, the so-called "interlayer space." The electrostatic potential can be used to help identify regions of the minerals that attract metal ions. Figure 4 depicts a portion of the electrostatic potential perpendicular to two layers of kaolinite. Clearly visible are the positions

of the hydrogen atoms internal to a given layer as well as those that span two layers.

The most notable aspect of the interlayer region is the large electric fields in the neighborhood of the hydrogen atoms. The fields reveal a strong network of hydrogen bonds between layers that make the interlayer region inaccessible to most chemical species. To penetrate into the kaolinite lattice, molecules (or ions) must disrupt the attractive interactions and push the layers apart. Experiments indicate that few molecular or ionic species can accomplish this.⁵ These results are consistent with studies that reveal that kaolinite preferentially adsorbs cations on the external edges of its growth surfaces.⁶ Interlayer electrostatics in kaolinite can be sharply contrasted with those in pyrophyllite and talc, whose layers are not bridged by hydrogen atoms. The layers of pyrophyllite and talc are held together by much weaker electrostatic forces, and metal ions are more readily intercalated into these materials.

Engineering Enzymes to Clean Up Pollution

Synthetic or foreign substances released to the environment are sometimes readily decomposed by natural microorganisms that have an appropriate metabolism or that have evolved new enzymatic pathways. However, such organisms may not be able to degrade dangerous compounds fast enough to allay safety concerns. Because enzymes derive their selectivity from a molecular

lock-and-key mechanism, efficient degradation of some compounds may require substantial changes in the enzymes produced by the microorganisms. Although microorganisms can evolve enzymes able to attack most chemical groups, efficient pathways may evolve very slowly if enzymes must undergo multiple, simultaneous genetic changes.

Accordingly, it may be more efficient to "design" multiple changes into an enzyme than to wait for them to occur biologically,

Figure 3. Crystal structure of the clay mineral kaolinite. The crystal is projected into the bc plane with the ab plane parallel to the clay layers (where the set of vectors $\{a,b,c\}$ refers to the unit cell axes). Inner hydrogen positions were optimized by Hess and Saunders.⁴

either by natural selection or under laboratory-controlled conditions. To carry out rational redesign that alters an enzyme's specificity and/or efficiency, we must have adequate structural data about the enzyme, its active site, its interactions with a substrate, and an understanding of its mechanisms of action. We chose cytochrome P-450cam as our initial enzyme to study. Cytochrome P-450cam catalyzes the hydroxylation of camphor and several substrate analogues such as norcamphor. While the hydroxylation of camphor is 100% efficient and produces only one product, for norcamphor the efficiency drops to only 12% and three products are formed.⁷

Our molecular-dynamics simulations of substrate-bound P-450cam complexes suggested that the loss in efficiency results from excessive motion of the norcamphor within the active site.⁸ These simulations showed that norcamphor motion is coupled to the motion of the sidechain of one of the amino acid residues (phenylalanine-87) that lines the active site. The simulations suggested that the efficiency could be increased by reducing the mobility of this residue. We did this by increasing the size of the residue at position 87. We first simulated this site-directed mutation by replacing phenylalanine with tryptophan. The simulations suggested that the efficiency of the mutant should be about twice that of the native species.

Figure 4. Electrostatic potential map spanning two layers of kaolinite in the *bc* plane. Indexes in parenthesis indicate directions relative to the crystallographic axis system.

Ensuing experimental studies found that the efficiency of hydroxylation of norcamphor essentially doubled (from 12% to 23%).⁹ This is the first time that site-directed mutagenesis and Molecular Dynamics simulations have been used together to predict and verify the effect of a mutation of cytochrome P-450cam designed to increase the efficiency of norcamphor hydroxylation. More recently, we predicted that a second but different amino acid substitution at the active site increases the efficiency and restricts the regiospecificity of P-450cam for the hydroxylation of norcamphor; both properties were experimentally verified.

Computational Modeling of Molecular Processes in the Environment

Our research has identified the computer resources needed to solve critical molecular-level problems in environmental restoration. A substantial increase in computing capability will be needed over what current technology can provide. Inherent limitations on conventional supercomputer systems make large-scale parallel computing the only viable route to solving these problems in a timely fashion. We need, first, molecular-modeling software that takes full advantage of the computing power offered by massively parallel systems. In addition, efficient and effective analysis of computational results requires a new generation of molecular-science graphics and visualization applications, powerful molecular information databases, and new graphical user interfaces. To this end, we have established development efforts in each of these areas as fundamental components of the Molecular Science Computing Facility now under construction in PNL's Environmental and Molecular Sciences Laboratory. Each of these projects is outlined below.

The gains from these efforts will be enormous. Molecular-modeling software running on massively parallel computers will increase current computing capability by 10 to 100 times. Scientists using such software with advanced visualization applications and knowledge systems will be able to successfully attack the complex, molecular-level Grand

Challenges posed by the DOE's environmental programs.

Molecular Modeling Codes for Massively Parallel Computers

Highly parallel computers present new opportunities to model complex molecular processes. To realize its full potential, however, molecular-modeling software must minimize serial computations and maximize the efficiency of parallel computations. Current software, optimized for serial computers, is often ill-suited for massively parallel machines. Not only must this software be restructured for use on parallel computers, but often an entirely new approach is needed to achieve the degree of parallelism required to make full use of the thousands of processors in massively parallel computers.

Recent exploratory work includes implementation of the DISCO code on the Intel Touchstone Delta computer, located at Cal Tech and sponsored by the Concurrent Supercomputing Consortium. DISCO, originally developed by a team headed by J. Almlöf at the University of Minnesota,¹⁰ embodies the direct Hartree-Fock method of approximating the electronic wave functions of molecules. We rewrote key sections of DISCO for the Delta.¹¹ The most computationally intensive part of the application is the calculation of two-electron repulsion integrals, a step that takes about 99% of the computational time on a serial computer.

We parallelized this process by replicating the key matrix assembled by the program (the Fock matrix) and using a master/slave procedure to perform dynamic load-balancing in computing the required integrals and constructing the matrix elements. Running in parallel on 16 Delta nodes, the code performed roughly as well as on a single-processor CRAY X-MP. Running on all 512 processors of the Delta, its throughput was roughly equivalent to 28 CRAY 2 processors (Fig. 5). PNL scientists are now using the program for modeling solvent-solute interactions.

We have also written a new Molecular Dynamics program, PARADYN, for distributed-memory parallel computers. However, the Molecular Dynamics approach used in PARADYN does not scale favorably to large numbers of

processors on today's massively parallel computers—on the Intel Delta, we obtain an efficiency of just 16% for 512 processors. The bottleneck is the amount of time spent in communicating the computed forces at the end of each time step. The next generation of parallel computers will deploy improved interprocessor communications, permitting PARADYN to operate efficiently with three to four times as many processors and to undertake correspondingly larger computational studies.

Graphical User Interfaces for Computational Chemistry Codes

Graphical interfaces help the user prepare his or her computational "experiments," and molecular visualization applications reduce the deluge of computed numbers to images that exploit our native ability to process visual information. Both techniques will be essential to users of advanced molecular science applications. Without them, we will not be able to take full advantage of the new generation of molecular-modeling software. A key feature of our interface research and development effort is chemistry-specific expert assistance, organized to minimize the amount of time and input required of the user in setting up computations. Another key objective is to enable computational "experiments" that transparently use multiple applications across various computers.

Our initial research work on graphical user interfaces for complex quantum chemistry programs has resulted in the Computational Chemistry Input Assistant (CCIA), a prototype user interface for the General Atomic and Molecular Electronic Structure System of codes¹² (GAMESS). The CCIA prototype, which runs on Apple Macintosh computers, includes interactive features familiar to Macintosh users. The interface leads the user through the process of creating the hundreds of input parameters required for GAMESS, presenting only as many options as needed to define a given calculation and ensuring that the options selected are consistent and reasonable. Most dialog windows have context-sensitive help that provides information pertinent to the data requested in that window. CCIA includes a Basis Set Adviser (Fig. 6) that bases its recommendation on the performance of each basis

set in similar calculations on related molecules. The prototype reference database, which currently includes molecular geometries, energetics, and vibrational frequencies for 72 molecules from experiments and up to a dozen basis-set families, is being expanded to include additional properties and molecules.

Visualization Codes for Modeling Simulation Analysis

Our visualization efforts focus on graphically analyzing data from molecular modeling applications, emphasizing the applications developed by the molecular science software effort. Recently, we have emphasized graphics relevant to analyzing mineral systems. Many visualization requirements for these systems may be met by extending commercial visualization systems such as AVS (from AVS, Inc.) and Explorer (from Silicon Graphics Inc.) with additional molecular-modeling modules. Use of such toolkits to build applications provides a flexible environment for responding to research requirements and can dramatically reduce both prototyping and development time. Although not all of our requirements can be met in this fashion, this approach provides a very efficient way to address many of our needs.

Figure 5. Parallel DISCO calculations on the imidazole molecule ($C_3N_2H_4$). The red line shows the relative speed of the parallel Delta computation versus a single-processor Delta computation for increasing numbers of processors. The blue line shows the speedup of a hypothetical 100% parallel code.

Figure 6. Input screen for the Computational Chemistry Input Assistant's Basis Set Adviser. The user specifies the properties to be computed and the accuracy desired. The Adviser checks a database of molecular computations and recommends appropriate basis sets.

Basis Sets (Seek Advice)
INFO

Areas to Emphasize

☒ **Molecular Structure**
☐ **Vibrational Modes**
☒ **Relative Energetics**
☐ **Moments/Polarizabilities**
☐ **Transition Moments**
☐ **Spin Density Properties**
☐ **Excitation Energies**

Relative Emphasis

Less ← | → More

Selection Criteria

Proceed
Default

Figure 7. Surface electrostatic potential for a single layer of kaolinite. The red isosurfaces encapsulate the spatial minimum in the electrostatic potential, classically the most energetically receptive region of space for positively charged species. Note the proximity of the isosurfaces to the ring oxygen atoms.

Using this approach, we have produced the Space Viewer application. Space Viewer is a general-purpose molecular graphics program built upon AVS. The new modules allow researchers to build and view crystals. Crystals can be displayed in several formats, and structural geometry can be analyzed interactively. Space Viewer enables interactive analysis of computations, including isosurfaces and contour plots of electron density, electrostatic potential, and density differences. Figure 7 is a surface electrostatic potential for a single layer of kaolinite rendered by Space Viewer.

Acknowledgments

We wish to thank our colleagues in the Theory, Modeling and Simulation and Computer and Information Science programs in the Molecular Science Research Center whose work contributed to this article: L. X. Dang, S. S. Xantheas, M. I. McCarthy, R. A. Kendall, M. A. Thompson, D. F. Feller, J. B. Nicholas, D. R. Jones, and E. N. Thornton.

Our work is supported by the Division of Chemical Sciences, Basic Energy Sciences, Office of Energy Research, U.S. Department of Energy and by the Laboratory Directed Research and Development Program at PNL. Computer resource allocations at the National Energy Research Supercomputer Center (Livermore, California) and Florida State University (Tallahassee, Florida) were provided by the Office of Scientific Computing, Office of Energy Research. PNL is operated for the DOE by Battelle Memorial Institute under contract DE-AC06-76RLO 1830.

References

1. For an excellent review of molecular dynamics methods, see M. P. Allen and D. J. Tildesley, *Computer Simulations of Liquids* (Clarendon Press, Oxford, 1987).
2. J. Caldwell, L. X. Dang, and P. A. Kollman, *J. Am. Chem. Soc.* **112**, 9145 (1990); L. X. Dang, J. E. Rice, J. Caldwell, and P. A. Kollman, *J. Am. Chem. Soc.* **113**, 2481 (1991).
3. R. Dovesi, C. Pisani, C. Roetti, and V. R. Saunders, *Program Crystal, Quantum Chemistry Program Exchange*, Program #577, Indiana University, Department of Chemistry, 1988; C. Pisani, R. Dovesi, and C. Roetti, *Hartree-Fock Ab Initio Treatment of Crystalline Systems*, (Springer-Verlag, New York, 1988).
4. A. C. Hess and V. R. Saunders, *J. Phys. Chem.* **96**, 4367 (1992).
5. C. T. Johnston and D. A. Stone, *Clays and Clay Miner.* **2**, 121 (1990); T. Tipton, D. A. Stone, K. KuBlat, and W. B. Person, *J. Phys. Chem.* **93**, 291 (1989).
6. G. E. Brown Jr., in "Mineral-Water Interface Geochemistry," *Rev. In Mineral.* **23** (Mineralogical Society of America, Washington, D.C. 1990); C. J. Chisholm-Brause, P. A. O'Day, G. E. Brown Jr., and G. A. Parks, *Nature* **348**, 6301 (1990).
7. W. M. Atkins and S. G. Sligar, *J. Am. Chem. Soc.* **109**, 3754 (1987); W. M. Atkins and S. G. Sligar, *J. Biol. Chem.* **111**, 2715 (1988).
8. M. D. Paulsen and R. L. Ornstein, *Proteins: Structure, Function, and Genetics* **11**, 184 (1991); M. D. Paulsen, M. B. Bass, and R. L. Ornstein, *J. Biomolecular Structure and Dynamics* **9**, 187 (1991); M. D. Paulsen and R. L. Ornstein, *J. Computer-Aided Molec. Design* (in press); M. B. Bass, M. D. Paulsen, and R. L. Ornstein, *Proteins: Structure, Function, and Genetics* **13**, 26 (1992).
9. M. B. Bass, D. Filipovic, S. G. Sligar, and R. L. Ornstein, "Increasing the Coupling Between Norcamphor Oxidation and NADH Reduction in the P-450cam Reaction by Molecular Dynamics," *Protein Engineering* (to appear).
10. J. Almlöf, K. Faegri, Jr., M. Feyereisen, and K. Korsell, *DISCO, a direct SCF and MP2 code*.
11. M. Feyereisen and R. A. Kendall, "An Efficient Implementation of the Direct SCF Algorithm on Parallel Computer Architectures," *Theoretica Chimica Acta* (to appear).
12. GAMESS, version 17 (June 1992); originally developed by M. Dupuis, D. Spangler and J. J. Wendoloski, NRCC 1980; version 17 developed by M. W. Schmidt, K. K. Baldridge, J. A. Boatz, J. H. Jensen, S. Koseki, M. S. Gordon, K. A. Nguyen, T. L. Windus and S. T. Elbert.

Modeling flow in porous media on so small a scale poses a substantial challenge. A model must be able to accurately treat multicomponent and multiphase flow accompanied by complex chemical and biological activity and must also accommodate extremely complex geometries. A unique feature of flow and transport in soils and rocks is the highly multiply-connected topology and varied channel geometries through which fluids and solutes must pass. This feature makes the use of traditional numerical algorithms, designed for simple geometries, very awkward and cumbersome for studying small-scale flow. Such considerations demand a new approach that can provide high-resolution results in complex geometries efficiently and easily. The exciting new field of cellular automata gives us a tool to attack this problem.

Cellular Automata Models

Traditional mathematical treatments of fluid flow take the continuum approach. Flow is considered as a transport of some smooth, infinitely fine quantity such as mass, momentum, or energy. Dynamic equations are derived by considering the net flux through a small volume, and there is little or no reference to the underlying atomistic nature of the flow. This approach has been extremely successful in describing a wide variety of fluid-flow situations. However, the resulting continuum equations are highly nonlinear. Because of the difficulty or impossibility of finding analytic solutions to such equations, we must often seek numerical solutions.

In numerical solutions, the fluid continuum is discretized in a micromodel in which parcels of fluid move between small, well-defined volumes. It has occurred to many researchers that a "bottom-up" approach that starts with a discrete, atomistic model of fluids might prove more fruitful than the continuum approach. The difficulty with this idea is that a direct model of molecular dynamics requires far too many calculations to be feasible. What is needed is a discrete method in which particles represent the collective behavior of many atoms or molecules.

Discrete fluid models have evolved to embody a level of detail and the essential features needed to reproduce Navier-

Stokes behavior on the macroscale. Cellular-automata models can reflect complex behavior and, under the right conditions, can reproduce fluid flow dynamics.

In simple terms, cellular-automata methods start with a set of objects (particles) that move on a regular geometric structure, or lattice, according to a set of rules that determine what happens when particles collide at the lattice's nodes. In this sense, a cellular-automata model is an atomistic approach to simulating physical processes. Although simple in principle, cellular automata can reproduce a surprising range of complex dynamics. With the proper choice of interaction rules and geometric lattice, cellular automata systems can mimic the flow of fluids even in multiphase situations. Additionally, a slight modification of the interaction rules enables the lattice geometry to represent not only a continuum but an arbitrarily connected structure such as the pore systems of soils and rocks. These features make lattice cellular automata an ideal computational tool for studying small-scale single and multiphase flow in porous media.

However, the discrete-particle approach tends to be "noisy," with computational artifacts sometimes obscuring the results. A variation on the basic method that addresses this problem is called the Lattice Boltzmann Cellular Automata algorithm. In this approach, probabilities replace discrete particles. The result is a much smoother and more accurate solution.

Advances in Experimental Techniques

The development of cellular-automata methods has been accompanied by complementary advances in experimental capabilities. We can now capture the pore structure of a rock and use it to create artificial experimental structures that embody the rock's internal geometry. Photomicrographs of thin sections of rock are first digitized. The digitized pattern is then used as a template to etch, onto the face of a piece of glass, the porous structure of the thin section. Finally, two such pieces of etched glass are used to make a glass sandwich that has the same pore structure as the thin section. We can use these porous

glass sandwiches to easily observe and measure the dynamics of two-dimensional flow and transport.

Even more impressive, x-ray tomography can capture the three-dimensional structure of the pores of a small sample of rock at a resolution of a few microns. This information is then digitized and used as a template to reconstruct the pore geometry in transparent plastic, allowing us to see in detail what's going on inside the rock when fluids are moving through it. The digitized record of the pore structure can also be used directly in the lattice Boltzmann numerical model. Since tomography replicates the exact geometry of a pore system, the lattice Boltzmann model enables us to separate out the relative contributions of geometry, flow dynamics, and reactions between the fluids and minerals of the porous material.

Model Applications

Conventional pumping methods recover at the surface only about one-third of the oil in an oil-bearing formation. After the pressure behind the initial flow dissipates, the remainder must be recovered by more expensive methods such as flooding the well with water or detergents. To increase the efficiency of recovery, it would be valuable to understand how the various fluids in the well behave under pressure. We have used the lattice Boltzmann approach to help us understand this process better.

Two examples illustrate the power of this computational method. In both cases, we use a fluid such as oil to fill a sample of a porous medium. We then inject a slug of another fluid, such as water, into the sample, followed by a continuous stream of oil.

In the first example, we used the etching technique described above to create a digitized, two-dimensional micromodel of the porous medium and simulated the injection process with the lattice Boltzmann method. In Fig. 1, a slug of water is injected into an oil-filled sample, displacing the oil. As oil once again enters behind the water slug, it surrounds and entraps trailing water ganglia. Still later, the slug of water is beginning to break up as it advances and begins to leave the sample. This behavior is quite

different from the idealized "piston" motion that large-scale models commonly assume for small-scale dynamics. Careful quantitative comparisons will take some time to complete, but the results typically compare very well with movies of actual experiments in glass micromodels created from thin sections. In this simulation, done on the Connection Machine 200 (CM-200) at the Advanced Computing Laboratory at Los Alamos National Laboratory, we used 2048 horizontal nodes and 1476 vertical nodes.

The second example features a three-dimensional model based upon the tomography technique described above. The geometry reflects the actual three-dimensional pore structure of a small sample (1 mm square) of an oil-bearing Berea sandstone sampled at 10- μ m resolution. (The sample was provided by Mobil Corporation.) In the sequence shown in Fig. 2, a slug of water is forced into oil-filled pores. The water first displaces the oil, which then sweeps in behind the water slug, displacing it but leaving small blobs of water behind, since water preferentially wets the rock surface. We ran this simulation on the CM-200, using $100 \times 100 \times 100$ lattice nodes to cover the volume of the sample. On our current computers, we can achieve a resolution of $512 \times 512 \times 512$.

An important variable needed for large-scale traditional models of two-phase porous flow is the relative permeability vs saturation curve for a material. Relative permeability is a measure of how easily a fluid will flow through porous material when subjected to a fixed pressure gradient. Our lattice Boltzmann algorithm can be used, for instance, on the geometries of the two examples to determine their relative permeability curves by computing the average flow rate through the sample for each phase as the composition of the inflow changes from one extreme (all water) to the other (all oil).

It appears reasonable to expect that the relative permeabilities of the fluids in a two-phase flowing mixture depend on the saturation level of each phase. Although this conceptual model works, it also leads to difficulties. For example, it predicts extremely large negative pres-

Figure 1. Three snapshots at different times in a numerical simulation of a water slug moving through an oil-saturated rock sample. The geometry of the pore space was taken directly from a digitized photomicrograph of a thin section of an oil-bearing sandstone. (a) A slug of water (blue) is injected into the oil-filled sample from the left, displacing most of the oil (red) toward the right. The interface between the two fluids is relatively uniform. (The irregular yellow areas are grains of rock.) (b) The slug of water has entirely entered the sample. It is now irregular in shape and is losing its definition. Oil is once again entering at the left, surrounding ganglia of water left behind. (c) Still later, the slug of water is starting to exit the sample. It is no longer moving as a unit but more like a collection of strings or blobs.

tures at low phase saturations, the physical meaning of which is suspect. A more complete theory would recognize that a particular saturation value can be assigned to an infinite variety of spatial distributions—that is, that the same amount of oil or water can be distributed in a volume in countless ways. Energy and entropy considerations can be used to constrain the possible ways of arranging the fluid masses in a pore space. New theories attempt to provide a better model of how fluids tend to distribute in a volume. Our lattice Boltzmann algorithm, which is based on first principles, can be used to calculate, under various conditions, the shape assumed by the interface between the fluids. The results can then be compared with the predictions of the new theories.

Prospects for the Future

These examples demonstrate the feasibility of using the Lattice Boltzmann Cellular Automata algorithm to model complex, small-scale flow in porous media. We are doing several studies to answer specific technical questions concerning wettability effects in oil reservoirs and to test out a new first-principles approach to describing fluid flow in porous media. The lattice Boltzmann algorithm is being extended to include other forces such as heating and chemical reactions, which are very important in porous formations. We believe this numerical method, coupled with recent advances in experimental capability, represents a true breakthrough in

Figure 2. A three-dimensional simulation of the flow in porous media. Here the oil is made transparent to help reveal the three-dimensional nature of the pore space. The pore geometry corresponds to that of a small sample of an oil-bearing Berea sandstone, as determined by high-resolution x-ray tomography. The red material represents the solid matrix of the sample. The remaining (light blue) region is pore space. (a) The oil phase initially fills the pore space completely. A slug of water (dark blue) is being forced in on the right. (b) The water first displaces the oil. The shape of the oil-water interface at this scale is very irregular. (c) The water is in turn displaced by injected oil, but leaves small blobs of water behind, since water preferentially wets the rock surface.

porous-flow technology. Applications to energy and environmental problems will be far-ranging, and may soon extend to disciplines such as materials science, where the method can be used in the manufacture of porous ceramics and filament-resin composites, and to biology, where it could be used to design biocompatible materials such as artificial bones, contact lenses, or medication-releasing implants.

Bibliography

G. Doolen, Ed., *Lattice Gas Methods for PDEs*, Proc. NATO Advanced Research Workshop, Los Alamos, NM, Sept. 6–8, 1989 (North-Holland Elsevier Science Publishers, Amsterdam, 1991).

- U. Frisch, B. Hasslacher, and Y. Pomeau, "Lattice-Gas Automata for the Navier-Stokes Equation," *Phys. Rev. Lett.* **56**, 1505–1507 (1986).
- U. Frisch, D. d'Humieres, B. Hasslacher, P. Lallemand, Y. Pomeau and J. Rivet, "Lattice Gas Hydrodynamics in Two and Three Dimensions," *Complex Systems* **1**, 649–707 (1987).
- B. Hasslacher, "Discrete Fluids," *Los Alamos Science*, No. 15, 175–217 (1987).
- D. Rothman, "Cellular-Automaton Fluids: A Model for Flow in Porous Media," *Geophysics* **53**, 509–518 (1988).
- S. Wolfram, "Cellular Automaton Fluids 1: Basic Theory," *J. Stat.Phys.* **45**, 471–526 (1986).

For homogeneous (nonlayered) type-II superconductors, the Ginzburg-Landau theory has been very successful in predicting vortex structure and configurations. For layered type-II systems, we use an extension of the Ginzburg-Landau model developed by Lawrence and Doniach. Unfortunately, analytic solutions to these models are known for only a few special cases. To make predictions from such models, it is therefore essential to develop effective computational methods for solving them.

When an external magnetic field is applied either parallel or perpendicular to the layers of the material, one can use physical symmetry to reduce the dimensions of the computational problem from three to two. However, because we are interested in layered systems to which magnetic fields are applied from an arbitrary direction, we must solve the full three-dimensional model. For problems of even modest size, such models result in systems with over 10^6 unknowns. To solve such large systems, we must use a combination of highly efficient algorithms and massively parallel computing.

Massively Parallel Optimization

The Ginzburg-Landau model is a phenomenological model that expresses the local free energy, ΔF , of the physical system in terms of a complex-valued order parameter, ψ . Magnetic field effects are included through the introduction of a vector potential A . The local free energy can be written as the sum of three terms,

$$\Delta F = F_{cond}(\psi) + F_{kin}(\psi, A) + F_{field}(A), \quad (1)$$

where F_{cond} is the condensation energy, F_{kin} is the kinetic energy, and F_{field} gives the field energy. Note that the order parameter and vector potential are coupled only through the kinetic-energy terms. For homogeneous (nonlayered) systems, these three terms are given (in cgs units) by the equations

$$\begin{aligned} F_{cond} &= \alpha|\psi|^2 + \frac{\beta}{2}|\psi|^4, \\ F_{kin} &= \frac{1}{2m} \left| \left(\frac{\hbar}{i} \nabla - \frac{e}{c} A \right) \psi \right|^2, \\ F_{field} &= \frac{1}{8\pi} |\nabla \times A|^2. \end{aligned} \quad (2)$$

For layered systems, the Lawrence-Doniach model consists of generalizations of these three terms.

The total free energy over some volume Ω , the free-energy functional F , is given by the integral

$$F = \int_{\Omega} \Delta F[\psi(x), A(x)] dx. \quad (3)$$

The global minimizer of this integral gives the physical configuration of the order parameter and vector potential. For a homogeneous (nonlayered) material, the minimum-energy solution was first predicted by Abrikosov. In this case, the vortices are packed in a hexagonal lattice. This solution, which can be verified computationally, is shown in Fig. 1. The figure

Figure 1. Magnitude of the order parameter shown for the two-dimensional homogeneous model. The plane of the image is perpendicular to the applied magnetic field; thus we see a two-dimensional slice through the vortex "tubes." We obtain the global minimum for the free-energy functional by a hexagonal packing of the vortices.

shows the magnitude of the order parameter in pseudocolor for a slice taken through the superconductor perpendicular to the applied magnetic field. The order parameter goes to zero in the center of each of the vortices, where a unit quantum of magnetic flux penetrates the bulk of the material. Since the system is isotropic, a two-dimensional computational model is sufficient to obtain these results.

However, the physical systems of most interest are layered, not homogeneous. To model these systems with an external magnetic field applied from an arbitrary direction, we must use the full three-dimensional model. In our computational results, the free-energy functional is defined on a three-dimensional rectangular mesh. For our parallel implementation, we geometrically partition the spatial domain in two of these dimensions (Fig. 2). The complex order parameter ψ is defined within the superconducting layers, and the vector potential A is defined within the insulating layers. The number of magnetic flux quanta and the angle of the applied magnetic field are enforced by the imposition of pseudo-periodic boundary conditions on the model.

We obtain numerical solutions for the order parameter and the vector potential by first applying a finite-difference discretization method to the free-energy expression for the Lawrence-Doniach model. The resulting discrete problem can then be solved by large-scale optimization techniques. The solution of this problem yields an approximation to the equilibrium distribution of superconducting electrons and vector potential that minimizes the free-energy functional.

Previous attempts to solve the nonlinear optimization problem used simulated annealing and steepest-descent algorithms. These techniques required many hours of CRAY time and could solve only homogeneous, two-dimensional problems with very

Figure 2. Three-dimensional layered model partitioned in two dimensions.

few vortices. In joint work with Stephen Wright at Argonne, we have developed a more sophisticated, second-order optimization algorithm based on a damped, inexact Newton method. The inclusion of damping alleviates a major problem of previous methods: the numerical degeneracy of the model resulting from the gauge invariance of the system. However, this second-order method requires the approximate solution of large, sparse systems of linear equations with millions of unknowns.

To attack these computationally demanding problems, we are using the Intel Delta parallel computer both for its large memory (8 gigabytes) and its computing power (512 i860 processors). (The Delta is located at Cal Tech and is sponsored by the Concurrent Supercomputing Consortium.) Because the Delta is a distributed-memory computer in which processors communicate via message-passing, the algorithms and their implementations are very different from those on a traditional sequential computer. We must parallelize four major computations: the evaluation of the function, the gradient, and the Hessian, and the solution of a series of large, sparse linear systems.

We can take steps to simplify the parallelization of these evaluations. First, we need many fewer grid points in the thickness direction of the three-dimensional model than in its length and width directions. We can, therefore, partition the grid for distribution to processors only in the latter directions. Second, with this

Figure 3. Adjacent values, or "ghost" points, required for evaluating the function, the gradient, and the Hessian.

partitioning of the grid, a processor needs information only from the edge of each of its neighbors to do the evaluations. For example, Fig. 3 shows the "ghost" points that processor 2 requires to evaluate its portion of the function, gradient, and Hessian.

To solve approximately the large, sparse system of linear equations, we use the conjugate gradient method preconditioned by an incomplete Cholesky factorization (ICCG). We have found that the incomplete factorization preconditioner works very well for these systems, whereas simpler methods fail to converge in a reasonable amount of time. The crux in the parallelization of the ICCG algorithm is a reordering of the matrix according to a coloring of the graph associated with the nonzero structure of the sparse matrix. We have found this approach to be very successful. The performance of the linear system solver scales linearly with the number of processors used and the problem size.

Computational Results

We have solved problems with 2×10^6 unknowns at speeds of up to 4.26 gigaflops on the Intel Delta. These systems involved matrices with over 10^8 nonzeros. Our results for the three-dimensional model are the first detailed calculations of the vortex structures and configurations in layered type-II systems.

Figure 4 shows isosurfaces of the magnitude of the magnetic field for a system near

H_{c2} . The external magnetic field is applied at an angle fixed by the pseudo-periodic boundary conditions imposed on the model. This view is taken parallel to the layers of the material; we can see the vortices tilted with respect to the layers. The figure shows the "pinching" of the vortices as the magnetic-field isosurfaces pass through the superconducting sheets and then spread out again in the insulating layers. Also visible is the characteristic "staircase" structure of the vortices, observed experimentally when the applied magnetic field is tilted.

It is energetically favorable for a vortex to remain parallel to a superconducting layer. As a consequence, there can be a difference between the vortex angle in the bulk of the material and the angle of the applied field. A phenomenon known as "vortex locking" has been predicted when the angle of the applied field is nearly parallel to the layers; our model reproduces this effect.

Since we are interested only in the minimum-energy configuration of the model, we use the parallel input-output capabilities of the Delta to write that state to disk. The data are then postprocessed and viewed with the AVS graphics package.

An arena of future interest is the effects of an electric current on the vortex configuration. The effects can be modeled with the time-dependent Ginzburg-Landau theory. With this approach, we can integrate the minimum-energy solutions forward in time and observe the resulting vortex dynamics.

Figure 4. View parallel to the layers of the vortex structure in the layered three-dimensional model.

We would like to be able to observe the vortex dynamics in real time. Accordingly, we have used the BBN-TC2000, the Titan graphics workstation, and the Doré graphics language to set up a prototype real-time, interactive system. We use the P4 package developed by Butler and Lusk. The master program is initiated on the Titan, which then spawns a parallel program on the remote BBN-TC2000. Data are communicated through a socket connection back to the graphics workstation, where we monitor the results in real time. We are currently implementing this approach for our superconductivity calculations on the Intel Delta. Its practicality will depend on successful use of the high-bandwidth (T3) network connection between Argonne and Caltech.

Summary

In our current collaboration, we continue to study vortex-locking and to calculate characteristic magnetization curves.

We are also using the model to extract detailed information about the three-dimensional structure of the vortex. The advent of massively parallel computers and the development of efficient computational algorithms have made possible this first "computational" glimpse into a layered type-II superconductor.

Acknowledgments

This work was supported by the Applied Mathematical Sciences Program of the Office of Energy Research, U.S. Department of Energy, under Contract W-31-109-Eng-38. James Garner also received support for this work from the National Science Foundation under grant number DMR-9200267. Our scientific collaborators in this work were Stephen Wright and David Levine of Argonne's Mathematics and Computer Science Division and Roy Benedek, Shelly Shumway, and Katherine Strandburg of Argonne's Materials Science Division.

Figure 1. The Superconducting Super Collider will be built and operated near Waxahachie, Texas by 1999. This enormous machine consists of a LINAC and four circular accelerators: the Low-Energy Booster (LEB), the Medium-Energy Booster (MEB), the High-Energy Booster (HEB), and the Collider.

Figure 2. The magnets to steer and focus the beams of protons will be implemented in modules called half-cells. A typical half-cell is illustrated here. Dipole magnets bend and steer the 20-TeV proton beam around the racetrack-shaped collider ring. A quadrupole magnet is used to keep the beam in focus. The spool pieces are fairly intricate structures with correction magnets to rectify small imperfections in the dipoles and quadrupoles. The collider will be constructed with 968 half-cells. The two proton beams will travel in opposite directions around the collider, each beam in a separate pipe with a complete lattice.

tools in bringing the Super Collider to fruition. We will rely on them heavily to record and help interpret the complex experimental data when it begins to flow from the accelerator. Computing power at the SSCL consists of over 2200 Macintoshes (used mainly for administrative tasks), almost 900 workstations, about 75 compute, file, and mail servers, five VAX mainframes, a distributed-processing "ranch" of 68 workstations, and the 64-node Intel hypercube machine. The workstation ranch consists of 32 SPARCstation 2s and 32 HP 9000/720s used as compute servers, and four 4D/360 SGIs with six CPUs each used as file servers. Finally, the SSCL logs a total of 10,000 computing hours per year on CRAY supercomputers housed at the National Energy Research Supercomputer Center (NERSC) at Livermore, California and at the Florida State University Computer Center in Tallahassee, Florida.

Detector Design

Designing detectors for the extremely rare events that occur when the high-energy proton beams collide is one of the tasks of Physics Research Division. This division designed and implemented the ranch of workstations, known as the Physics Detector Simulation Facility (PDSF), described above. The PDSF uses the Cooperative Processes Software (CPS) developed at Fermilab. The CPS consists of a job manager program and a set of sub-routines used for interprocess communication, remote subroutine calls, and synchronization. These tools support a wide range of parallel computing tasks.²

The PDSF has drawn on the foregoing tools and on some CERN library tools to develop BCD/CPS, a parallel implementation of the Bottom Collider Detector (BCD) GEANT3 simulation for CPS processor ranches.³ GEANT3 is a CERN program to generate a geometrical description of the BCD detector, to input an extended particle-decay database, to initialize a data structure, to input BCD dipole field maps, and to book user-defined histograms. Either of two other programs (both distributed by CERN), ISAJET or PYTHIA, a Lund Monte Carlo program, is used to input event data, and the CERN program FZ is used to output event data.

This implementation offers orders of magnitude more computing speed than other methods. Figure 3 is a plot from the FZ event output. Because this type of computing is sufficiently flexible to enable use of its cycles in either a parallel mode (many workstations running one code) or a

Figure 3. Schematic of colliding proton beams showing a collision event, magnets, and various particle detectors. The event shown was generated by ISAJET 6.36. For scale, the gap between the pole tips of the central dipole magnet is 5 m. The plot was produced by the program BCD/CPS, which runs on a ranch of workstations that includes SPARCstations, HP 9000/720s, and SGIs running in parallel mode. Gamma rays are shown by a dotted blue line, charged particles (except muons) by a solid red line, neutral hadrons or neutrinos by a dot-dash black line, and muons by a dashed green line. Inset table identifies key components.

single-workstation mode, it shows much promise for efficient use of CPU cycles and available memory.

Lattice Analysis

A major goal of our simulation studies is to understand how to shape the proton beams and keep them in the proper race-track orbits. To do this, we are using TEAPOT (Thin Element Accelerator Program for Optics and Tracking), a code with a long history of simulating accelerator magnet configurations known as "lattices."⁴ Our approach is roughly similar to the finite-element method of analyzing mechanical structures. All elements are replaced by sequences of drift regions, in which particle trajectories are straight lines, and by infinitely thin elements in which deflections are calculated exactly. The feasibility of this approach depends heavily on available computing power.

To determine the stability of the proton beams with the TEAPOT code, we would like to track hundreds of simulated protons for millions of turns. (The actual proton beams will circulate a million times in just five minutes.) On a scalar computer, such a task would require months of continual CPU time. Two years ago, Yiton Yan of the SSSL reported on the results of implementing ZTRACK, a post-TEAPOT tracking code, on CRAY-2 multiprocessors.⁵ Following this work, a decision was made to study TEAPOT's performance on a massively parallel computer, gradually adding more refinements.⁶

After studying a variety of parallel architectures, we chose a 64-node Intel hypercube machine as a prototype TEAPOT production machine. The choice of a parallel computer for the TEAPOT code was largely determined by data-flow considerations (Fig. 4). As each simulated pass of a proton beam through the Super Collider requires extensive global data to simulate each lattice element, the number of calculations per step (particle passage through an element) is fairly large. As the individual processors of SIMD (Single Instruction, Multiple Data) machines have limited access to global data, we ruled them out. We then considered MIMD (Multiple Instruction, Multiple Data) machines. These range from machines with many processors but modest computing power per processor to those with fewer but more powerful processors. On the former, TEAPOT could be programmed with each lattice element assigned to a processor. On the latter, however, it would be better to assign individual particles to a given processor.

In a scheme in which particles are successively assigned to CPU nodes, no communication between nodes is necessary. (Fortunately, each particle in a very high-energy beam can be considered independently because in this limit the current-dependent effects are small.) The TEAPOT code performs very well in this mode in simulating the bulk of accelerator phenomenology. Accordingly, we chose to acquire a MIMD machine with fewer but more powerful processors. Of those machines, the Intel iPSC/860 (hypercube) had sufficient local memory for the lattice elements and had the most powerful individual processors. This choice also enabled us to build upon hypercube experience from ORNL (Oak Ridge National Laboratory). The Intel hypercube also looked very promising for developing more sophisticated versions of TEAPOT that require interprocessor communication.

Running TEAPOT on the Hypercube

The Intel hypercube machine was delivered in January 1991. Each of its 64 nodes is an i860 microprocessor with double-precision arithmetic at an observed peak rate of 30 megaflops. (The theoretical 64-bit

Figure 4. Data flow in the Thin-Element Accelerator Particle Optics and Tracking code (TEAPOT). This diagram shows graphically why the massively parallel Intel hypercube was chosen to run this code. The hypercube is a MIMD machine with enough memory in each node to store global data that includes characteristics and effects of each lattice element. Data for 12,000 single elements and coordinates of each particle are stored.

peak speed is 60 megaflops.) Two nodes have 32 MB of memory, and each of the remaining 62 nodes has 8 MB of memory. The nodes are arranged in a hypercube configuration and are connected to the LAN (local area network) through an 80386-based microcomputer called the System Resource Manager (SRM).

We initially implemented TEAPOT on the hypercube with several particles (protons) assigned to a single node. Fifteen types of element can be used as TEAPOT input data, including drift spaces, magnets (dipoles on up), simple horizontal or vertical kicks, monitors, markers, radiofrequency cavities, and collimators. Input data for each particle are its x and y coordinates, its momenta in the x and y directions, its time of flight, and the total momentum deviation, dp/p . We have written stand-alone versions of the tracking kernel of TEAPOT, one in C and the other in Fortran 77 (ZTRACK).

TEAPOT on the hypercube has performed tracking simulations of all four of the Super Collider's circular accelerators. These production runs take anywhere from a few minutes to a few days. For longer runs, the code generates restart files. In general, TEAPOT runs at an average of eight megaflops per processor. Each processor, then, is roughly equivalent to three Sun SPARCstation 2s in computing power. The hypercube's total average processing power for this code shows that this machine is about 20% faster than a CRAY Y-MP running in fully vectorized but uniprocessing mode (Table 1). On a Y-MP with 16 processors, the code executes roughly 12 times faster than with a single processor.

We have used production runs of TEAPOT on the hypercube mainly to study the high-energy booster and collider designs. By following the survival patterns of particles launched with chosen displacements and momenta, we can determine the stability of the proton beam. Figure 5 is an example of a plot generated by TEAPOT. Contours show survival time in terms of how many turns of the accelerator each particle makes before being "lost" (colliding with a wall or other obstacle). The plots are directly related to the beam's half-life, which is the ultimate criterion for how effectively the beam is confined.

Code Improvements

During the year and a half that we have been running TEAPOT on the hypercube, we have refined the code in many respects. For example, we can now model lower-energy beams, which require consideration of space-charge effects. To include these effects, we must recalculate the electromagnetic fields by solving their time-dependent Maxwell equations. These calculations are done in a particle-in-cell (PIC) fashion, requiring communication of data among the nodes.⁷ We have developed a very efficient scheme for solving Maxwell's equations by mapping the

Machine	Version	Run time (hours)
CRAY Y-MP	Scalar, Fortran	103.5*
	Vector, Fortran	13.8*
	Fortran	11.5
	C	11.3
64-node iPSC/860	Assembly (estimated)	5.5

*Single processor.

Table 1. Computation times for tracking codes.

Figure 5. Contour survival plot produced by the TEAPOT code on the Intel hypercube for 441 protons launched in the collider and circulating 100,000 times. Contours are for equal particle lifetimes (expressed in number of turns in the accelerator) on a transverse section through the accelerator aperture at a fixed location in the ring as a function of transverse position x and momentum offset dp/p .

Figure 6. Performance data for selected TEAPOT subroutines (Kick3D and Base3D) that calculate space charge. LOOP generates performance data for the entire loop, including Kick3D, Base3D, and two other routines. Trials were performed with 1, 8, 16, and 32 hypercube nodes (processors). (a) Speedup achieved as a function of number of nodes. (b) Efficiency in terms of overhead per node. For node-to-node communication, the total overhead per node declines as more nodes are added.

Figure 7. First six frames of a sequence showing how microbunches of protons from the LINAC coalesce into a macrobunch in the Low-Energy Booster. Each frame is a section of the beam pipe with the longitudinal direction vertical. Each section is one LEB radio-frequency "bucket" long. Frames were generated at intervals of 10.0 ns; a proton makes one complete revolution of the LEB in 2.4 μ s. Each frame contains nine LINAC microbunches with a frequency of 428 MHz. The LEB radiofrequency buckets are at a frequency of 47.5 MHz. The colors identify the separate microbunches as they come from the LINAC; each microbunch comprises 6000 particles.

three-dimensional cylindrical grids onto a one-dimensional array of computing nodes that uses equal grids in the z direction. Performance is similar to that on a CRAY Y-MP. Figure 6 graphically shows the efficiency and speed of some parallel calculations as a function of the number of processors used. Figure 7 is a series of plots tracing the evolution of a beam of particles moving in a space charge.

Our experience shows that system software, including compilers and subroutine libraries, is still in its infancy. Two upgrades of the system software have dramatically improved the stability of the code and its sustained performance. We are further improving sustained performance by writing critical segments of the code in assembly language.

Summary

The use of different computing resources with various architectures has proved fruitful in designing both the accelerators and detectors for the Superconducting Super Collider. We are confident that this approach will produce not only the best, most cost-effective designs but will position us to apply the latest computer technology to experiments in particle physics well into the next century.

Acknowledgments

Many researchers contributed to this article. Laird Cornell and Lee Roberts were responsible for implementing the PDSF and developing the BCD/CPS code. ISAJET was written by Frank Paige of the SSCL (formerly of Brookhaven National Laboratory) and Serban Protopopescu of BNL, and PYTHIA was written by Hans-Uno Bengtsson of the University of Uppsala and Torbjorn Sjostrand of CERN. Long Chyr Chang did the parallel programming that added space charge to TEAPOT. M. L. Sloan of the University of Texas Center for Fusion Research, in Austin, helped to formulate the space-charge implementation.

The Defense Advanced Research Projects Agency supported some of the TEAPOT/hypercube project through a joint DARPA/DOE program for high-performance computing. Shinji Machida provided the descriptions for the TEAPOT

space-charge plots. Valerie Ditta graciously supplied up-to-date statistics on numbers of computers at the SSCL. Yiton Yan made many valuable suggestions for the entire article. Haregeweyn Ferede and Fesseha Mariam were very helpful in providing timely background information. The Machine Simulation Group also provided strong support for the TEAPOT/hypercube project. Critical reading of this manuscript by Yiton Yan, Long Chyr Chang, Haregeweyn Ferede, and Lee Roberts is much appreciated. All groups and individuals not further identified are on the SSCL staff.

References

1. N. Baggett, Ed., *The SSCintillator*, Superconducting Super Collider Laboratory (July 1992).
2. L. Cornell, *High Energy Physics Computing at the SSCL*, Superconducting Super Collider Laboratory, SSCL-400 (March 1991).
3. L. Roberts, *BCD/CPS: An Event-level GEANT3 Parallelization via CPS*, Superconducting Super Collider Laboratory, SSCL-413 (April 1991).
4. L. Schachinger and R. Talman, "TEAPOT: A Thin-Element Accelerator Program for Optics and Tracking," *Particle Accelerators* **22**, 35 (1987).
5. Y. Yan, "Supercomputing for the Superconducting Super Collider," *Energy Sciences Supercomputing, 1990*, A. Mirin and G. Kaiper, Eds., pp. 9-13, National Energy Research Supercomputer Center, Lawrence Livermore National Laboratory (1990).
6. G. Bourianoff and B. Cole, "Parallel Processing at the SSC: The Fact and the Fiction," *Conference Proceedings No. 255, Particles and Fields Series 48*, Alex Chao, Ed., Corpus Christi, Texas, pp. 251-254 (1991).
7. L. C. Chang, G. Bourianoff, B. Cole, and S. Machida, *A Parallel Implementation of Particle Tracking with Space Charge Effects on an Intel iPSC/860*, Superconducting Super Collider Laboratory, SSCL-581 (August 1992).

The Energy Sciences Network

The Energy Sciences Network (ESNET) is a nationwide computer data communications network that supports energy-related scientific research. Managed and funded by the Office of Scientific Computing of the Department of Energy's Office of Energy Research, ESNET provides access to major energy research installations, connecting various energy research computing centers and facilitating the dissemination of information among scientific collaborators.

ESNET, which became fully operational in early 1990, is engineered and operated by NERSC. Its implementation currently consists of T1 (1.5 million bits per second) leased phone circuits that connect multiprotocol routers. The routers currently in use support several communication protocols, including the Department of Defense Internet Protocol, DECNET Phase IV, X.25, and the Open Standards Interconnection ConnectionLess Network Protocol.

ESNET is intended to facilitate remote access to major scientific facilities, disseminate needed information among scientific collaborators, and provide widespread access to existing Energy Research supercomputer facilities. ESNET has direct links to Japan and Germany, and is connected to the research networks of other federal agencies. Through collaborative arrangements, the various federal agencies cooperate in providing data transport services both domestically and internationally. A small number of high schools are connected on a pilot-program basis in support of the National Education Supercomputer Program and the Department of Energy's commitment to education.

The Network Operations Center provides round-the-clock monitoring and control for the various network components that comprise ESNET. Routine monitoring verifies the network's integrity and provides statistics for troubleshooting and long-term

planning. Its capabilities include distributed displays, an online trouble ticket system for problem tracking, and electronic mail boxes for information, operations, and trouble calls.

ESNET is being upgraded to provide next-generation capabilities and to support the continuing growth of traffic, which roughly doubles every six months. The upgrade will use a technology called Asynchronous Transfer Mode (ATM). ATM is an emerging technology based on the high-speed transmission of fixed-length "cells." This service is being provided on an "early-availability" basis, although it is not expected to be generally available before 1994. The ATM technology is considered the basis for a broad-band technology (called B-ISDN, for Broadband Integrated Services Digital Network) that will support future multimedia "universal" transmission. B-ISDN will be capable of simultaneously carrying voice, data and video communications traffic, and bandwidth-intensive applications. These applications could include, for example, the transfer of high-definition graphics images and links between high-capacity local area networks.

With the new ATM technology, data communications networking will evolve from the current T1-based capacity to substantially faster speeds, with steps to 45, 155, and 622 million bits per second in the next few years (a total speedup of more than 400 times the current bandwidth). Currently, approximately 20 pages of a standard dictionary can be transmitted in one second at T1 rates. At the highest rate planned, about 8,000 pages of information will be transmitted per second. This network upgrade will not only allow researchers to work more effectively, but will enable entirely new capabilities. It will also accelerate commercial schedules for making the new technology available on a nationwide basis.

Workstations send messages that the interface equipment segments into cells for streaming through the ATM network, where they are reassembled into messages for delivery. This technology represents a first step toward the concept of a "national data highway."

Communications "backbone" of ESNET.

The Advanced Combustion Modeling Environment (ACME)¹ addresses these issues by creating a computational modeling environment for the combustion scientist that

- Enables and manages the development of complex combustion models.
- Allows easy access to high-performance, next-generation computing systems.
- Allows simultaneous development of both the computational and modeling sciences.

As part of the ACME effort, we have developed POET (Parallel Object-Oriented Environment and Toolkit). POET is a toolkit for parallel computing whose purpose is to provide the user with a transparent link to the power of parallel distributed computing. Our approach is similar in design methodology to the X toolkit (Xt) for window management. We have designed a high-level, object-oriented framework that isolates the description of the physical model from the code that implements the parallel algorithm and data flow. Consequently, the combustion scientist need be concerned only with application-specific code.

POET Architecture

Existing tools for developing parallel software generally fall into two categories. One category comprises high-level tools and compilers that hide the parallelization details, making them easy to use but also hiding the features that can lead to bottlenecks. The other category consists of low-level tools for message-passing that create scalable code but require such detailed knowledge of algorithms and software that they are difficult for the non systems programmer to use. The purpose of POET is to allow a physical scientist to create an integrated and scalable application code that transparently accesses parallel computing resources and avoids the traditional pitfalls associated with parallel computing. POET is a high-level, object-oriented framework for parallel computing that is designed to directly integrate existing application codes. It is written in C++ in an extensible manner that guarantees scalable code.

Because of the similarity between the paradigms for object-oriented programming and the physical structure of multiple-instruction, multiple-data

(MIMD) parallel computing, the parallel-processing community is becoming increasingly aware of the benefits of object-oriented computing to MIMD processing. In most MIMD computers, the CPU and the memory reside together on a single unit, and computations proceed by message-passing among collections of these units. The object-oriented design methodology is based on collecting into a single object both the processing method and the data to be processed.

Computations proceed by passing messages between objects. Consequently, the object-oriented paradigm is a natural fit for MIMD architectures and has been successfully applied in the POET design.

Figure 1 indicates three distinct approaches to software development.² The first (Fig. 1a) is the traditional approach, often used when porting existing code from vector machines to parallel machines. Most often, the user is responsible for all aspects of the code: the application-specific code, the parallel algorithm, and the data flow. This approach requires a specialist in parallel code development to achieve an effective speedup.

The second approach (Fig. 1b) consists of providing an object-oriented class library that has already been designed for a parallel implementation. However, typical C++ class libraries require the user to write the main routine that drives the objects instantiated from the library. The user is responsible for the highest-level portion of the application and consequently must be aware of the environment in which the application is to run. With unsophisticated users, this can lead to highly complicated structures that cancel all the benefits gained from the object-oriented class library. For example, consider the X Windows programming environment. X Windows requires constant arbitration between the server and the application's window event loop. It would be disastrous to hand over the top level of an X program to a naive user and to trust that the main event loop would be consulted at the right time and often enough to avoid poor behavior or outright deadlock. In Xt, however, the top level is handled by an object called the top-level shell. The user's main program exists only to configure objects, called *widgets*, by

adding callback routines or modifying attributes under the top-level shell. A widget is realized by calling the main-loop function for the top level, from which control will not return until the application ends.

As we define it, a toolkit is a library of objects that also includes a top-level control object. We refer to POET as a toolkit because it is more than a collection of objects that simply provide computational processing capabilities. The POET approach (Fig. 1c) seeks to provide a mechanism similar to the Xt approach for MIMD parallel computations. That is, POET separates the physics specific to a problem from the parallel algorithm needed to solve it. Our goal is to embed as much of the computational control as possible into the toolkit. The physics specific to the user's particular application problem is handled as a user add-on. In our first implementation, these add-ons take the form of callbacks that specialize a general algorithm embodied in an object selected from the toolkit. Taking into account the object the user has requested, the toolkit decomposes the problem across multiple processors. The callbacks, however, need not be aware of

this decomposition, nor of the architecture of the parallel environment, because the physics that specialized the problem is independent of the object chosen for the solution. The following section gives a detailed Fortran example for a turbulent reacting probability density function code.

In this first implementation, we have provided a Fortran interface because users in the computational physics community that are most likely to benefit from this toolkit are either solely or primarily familiar with Fortran. Special limitations imposed by integrating Fortran required the use of callbacks rather than inheritance, which would be more appropriate to C++, the native language of POET. The top level of the toolkit is initialized by the Fortran subroutine PTINITIALIZE. POET objects, called *pidgets*,³ are sent messages through the Fortran subroutine PTSNDMSG. Fortran uses only these two functions to access the toolkit. Our current Fortran interface is somewhat cumbersome and inefficient. However, as previously mentioned, our most likely user has developed Fortran models as applications and therefore needs this interface.

Figure 1. Different approaches to parallel software development require varying degrees of user responsibility (orange). (a) The traditional approach requires the user to be responsible for all aspects of development—the parallel algorithm, the data flow, and the application specific code. (b) The second approach provides an object-oriented class library (blue), but the user must still write the main routine and is therefore still responsible for the highest level of the application. (c) POET is a toolkit that handles as much of the computational control as possible (blue). The user need only be concerned with the particular application-specific code required to solve a problem.

Adaptive Methodologies for Computational Fluid Dynamics

John Bell, Lawrence Livermore National Laboratory; Marsha Berger, Courant Institute of Mathematical Sciences, New York University; P. Colella, University of California, Berkeley; J. Saltzman, Los Alamos National Laboratory

Fluid-dynamics phenomena typically have features that vary on many length scales and exhibit complex spatial and temporal behavior. In combustion, for example, energy is released in a region, such as a flame front, that is much thinner than the length scales that characterize the gross motions of the fuel and the surrounding air. In flow past an immersed body, the forces acting on the body are completely determined by the thin, viscous boundary layers near it and by the behavior of the spatially confined wake behind it. Fluid flow in an engineering device is sensitive to the details of the device's shape.

Over the past ten years, the Department of Energy's Applied Mathematical Sciences program has funded a research effort in computational fluid dynamics aimed at developing numerical methods that accurately and efficiently represent such effects. These adaptive methods, as they are known, are based on the idea of varying the computational effort in different regions of space to resolve the various effects with optimal efficiency. Typically, such methods prove cheaper by a factor of 10 to 100 for a given level of accuracy.

One such method is known as adaptive mesh refinement, in which mesh points in

a finite-difference calculation are inserted and removed as the calculation progresses. This technique maintains a uniform level of accuracy and resolves important features of a problem. The first illustration is an example of a typical mesh distribution for such a calculation in two space dimensions. The density of mesh points varies as a function of space at any given time and changes as the calculation evolves in time.

In the technique of front-tracking, we represent a thin layer, such as a shock front, a boundary between two different materials, or the flame front mentioned above, by an infinitely thin boundary that moves through a finite-difference grid. By taking advantage of our analytical knowledge of the front's dynamics, we obtain a representation that is much less expensive computationally than if we were to resolve the details of the front's internal structure. The second illustration shows the results of calculating a shock wave that is interacting with an interface between two gases, carbon dioxide (CO_2) and methane (CH_4). Front-tracking enables us to compute efficiently and accurately the considerably different acoustic and thermodynamic properties of the two gases that produce the resulting complex wave interactions.

Typical adaptive mesh distribution for a two-dimensional calculation.

Finally, we are developing a variety of flexible and general techniques for representing complicated boundary geometries. The bottom illustration shows the time evolution of jets formed by high-pressure gas flowing through holes in a baffle plate into a cylindrical chamber (only one-quarter of the cylinder is shown). The chamber wall and the plate are gray, and the pressurized gas underneath the plate is shown in red. The jets (moving upward in the figure) pass through four circular holes in the plate and emerge into the chamber, driving a compression wave (red) ahead of them, followed by the expanding jet material (blue). In this example, we have used adaptive mesh refinement to resolve the complicated fluid dynamics of the jet.

The methods described here are being used in a variety of applications, including space science, shock hydrodynamics, turbulence, and defense technologies. The Department of Energy is funding this work as a Grand Challenge in computational fluid dynamics and combustion dynamics, involving researchers at Lawrence Livermore National Laboratory, Los Alamos National

Laboratory, New York University, and the University of California at Berkeley, in collaboration with Sandia National Laboratory Livermore's Combustion Research Facility. Our goal is to develop these methods into a set of tools for solving combustion problems having both fundamental scientific interest and practical engineering applications.

Shock wave impacting the interface between two gases. Front-tracking was used to resolve the resulting wave interactions. The black curve shows the position of the interface between the two gases. The different acoustic properties of the CO_2 and CH_4 lead to the complex wave interactions observed in the calculation.

(a) Gas (red) is injected through holes in a flat plate. (b) The emerging gas jets drive a compression wave (red), followed by expanding jet material (blue). In (c) and (d), the compression wave and the jet material undergo further evolution. Adaptive mesh refinement was used to resolve the dynamics.

Currently the user must configure and message the pidgets needed to implement a solution. We are developing a high-level intelligent interface that will completely isolate the user from the details of pidget creation. The frame-based interface will provide abstracted representations of standard combustion models. A graphical user interface will enable the user to customize a particular problem. Automatic code generation will provide the link between the pidgets and the user's problem representation. In addition, we expect eventually to incorporate automatic code generation for chemical mechanisms.

The frame-based user interface will provide the mechanism for managing the complexity of the models. As users customize each application to include more detailed chemistry and turbulence, models will become increasingly difficult to manage and verify. In addition to providing an environment natural for the user, high-level algorithms that provide constraints on model development will force the user to create code that is consistent, both computationally and with the modeling algorithms.

Chemically Reacting Flow Problem

To demonstrate POET, we have investigated a probability density function (PDF)⁴ code for a problem involving diffusion of a jet flame.⁵ The PDF algorithm primarily involves Monte Carlo calculations and is highly amenable to an efficient parallel implementation. POET is used to partition the algorithmic portions of the code (such as the equation solver and Monte Carlo simulation) from the application-specific code.

In this problem, a cylindrical jet injects a premixed fuel that is then ignited. The problem comprises three parts: a turbulent-motion model, a chemical-reaction model, and a coupling between chemical reactions and turbulence. We compute downstream exhaust with an advancing grid that begins at the inlet and advances downstream until chemical activity is complete, as shown in Fig. 2. Symmetry enables us to model this problem with a one-dimensional grid that effectively represents a radial slice of the cylindrical nozzle.

The algorithm also incorporates a piecewise Monte Carlo (PMC) method that

transports (or mixes) particles to and from nearest-neighbor cells. This mixing is a result of fluid-mechanical turbulence and Fickian diffusion. The chemistry that accounts for most of the computation is local to each cell as particles from one node cell are required to communicate only with their right and left neighbor cells.

The POET implementation builds upon a PMC object. The first step is to spatially decompose the domain of the linear grid. The PMC object then uses nearest-neighbor dependence to define three types of cell: left boundary, right boundary, and interior. The data necessary to compute the properties of an interior cell come from the cell itself as well as its right and left neighbors. For this particular application (a one-dimensional PDF), a specialized routine handles interior cells. It receives information that represents all of the data relevant to the cell and its neighbors. If interprocessor communication is necessary to access the information, POET takes care of the message-passing necessary to move the information from processor to processor. Similarly, specialized routines are necessary to define the data transfer for the right and left boundaries. These three routines are sufficient to accomplish the PDF calculation. The user completes the functionality of the PMC object by specifying the independent routines that provide the physics calculations particular to the PDF problem.

Results

The model was implemented on a distributed network of 25 IBM RS/6000 workstations located at the U.C. Berkeley mechanical engineering department. The Monte Carlo portion of the problem, which comprises the entire one-dimensional grid of cells plus boundary conditions for the two cells on the left and right ends of the grid, accounts for 99.3% of the computer time, even with minimal chemistry. At the appropriate time, statistics computed from the particles distributed over the cells contribute to an update of the flow field, either as an iteration converging on a steady state or as a time step in a nonsteady problem. The addition of realistic chemistry increases the computation required by 10- to 100-fold.

Figure 2. (a) Schematic of the turbulent reacting jet problem solved with POET. The particular problem discussed here is a steady-state turbulent flame. It is computed by advancing a one-dimensional grid along the flow direction from the inlet outward until the flame chemistry ceases meaningful activity. Though the relatively less demanding fluid mechanical calculation is done on a host processor outside of the toolkit, the computationally intensive chemistry and turbulent diffusion calculation is done in parallel using Fortran callbacks. (b) The advancing one-dimensional grid is spatially domain decomposed on the distributed processors. Three Fortran callbacks that encapsulate the physics of the reacting jet are provided by the user. The two boundary subroutines, *LBndry* and *RBndry*, handle the two appropriate end cells, and the interior subroutine, *Inside*, handles the center cell and its two neighbors. *LBndry* and *RBndry* are expected to modify only the boundary cells, and *Inside* is expected to modify only the center cell.

Figures 3 to 5 show the results of our calculation. Figure 3 shows the temperature profile of the jet flame. Figures 4 and 5 indicate concentrations of nitric oxide (NO) and the hydroxyl radical (OH). To make an impact on the combustion community, it is critical that we be able to predict concentrations of such pollutants. Species such as OH and NO can be measured experimentally to verify models.

Because of limitations on memory and available computational time, a strictly serial code is incapable of addressing more than a few chemical species and reactions. The parallel version, generated with the help of POET and running on our collection of IBM workstations, can do a hydrogen-jet calculation that includes tens of chemical species and reactions in less than a day. Figure 6 shows how these calculations (including full chemistry) are speeded up as a function of the number of processors. We plan to implement the

code next on an Intel Paragon (a massively parallel computer) and expect to produce these results in minutes.

POET provides the power necessary to compute PDF chemically reacting flow problems with realistic chemistry on massively parallel and distributed systems. Moreover, by encapsulating the PMC algorithm in an autonomous object and thus insulating the specific physics of a PDF problem from the solution algorithm, we produce code that can be reused on other chemically reacting flow problems. Ultimately, we will need a parallelized equation solver to update the flow field; however, in this early implementation we hope to avoid the issue by solving the entire flow field on every processor. Since this step requires little computation and the communication of very few averaged numbers to each processor, the lack of scalability should have little impact on the speedup necessary for this application.

Figure 3. Temperature profile for simulated diffusion of a jet flame. Blue indicates low temperatures and red high temperatures (image generated by the Advanced Visualization System).

Figure 4. NO concentration for the same jet flame allows the combustion scientist to predict the formation of pollutants. Blue indicates low concentration and red high concentration (image generated by the Advanced Visualization System).

Figure 5. Intermediate species concentrations, such as OH, can be compared with experimental results to verify models. Blue indicates low concentration and red high concentration (image generated by the Advanced Visualization System).

Summary

Interfacing the PDF jet flame diffusion model with POET has enabled the first investigation of a turbulent, nonpremixed flame that includes full chemical mechanisms. This benchmark calculation provides an opportunity for examining the limitations of various reduced chemical mechanisms. The design methodology we have described, in concert with the use of an object-oriented language, has produced code that can be continually developed by both computational and physical scientists. The modular structure of POET allows a physical scientist to easily modify both the fluid mechanics and the chemistry of a model without affecting the code that distributes the problem to the various processing elements.

We plan extensions of POET to support higher dimensionality and applications requiring efficient equation solvers. Advanced computing techniques, such as generating automatic code for chemical mechanisms, will impose the partitioning necessary to handle the models as their complexity increases. This approach will create a well-defined path for combustion modelers and other computational scientists to formulate and access high-performance models customized to their particular application.

The approach to software development described above is vital to the full and efficient use of next-generation systems. It is particularly important in computational physics, where state-of-the-art techniques are being applied in both physical science and computer science. Partitioning supports model development and easy access to distributed resources otherwise untapped by the naive user.

Figure 6. Linear speedups have been observed and are generally indicative of a scalable application.

Acknowledgments

This work was supported in large part by the U.S. Department of Energy.

Notes and References

1. M. L. Koszykowski, R. Armstrong, R. E. Cline, Jr., J. Macfarlane, J.-Y. Chen, and N. Brown, *The Advanced Combustion Modeling Environment*, American Chemical Society, Washington, DC, August 24–28, 1992.
2. This list is not meant to be complete. The examples highlight the difference between a toolkit approach and other approaches.
3. The name "pidget" is analogous to the widget in the X toolkit.
4. S. B. Pope, "The Probability Approach to the Modeling of Turbulent Reacting Flows," *Combustion and Flame* **27**, 299 (1976).
5. J.-Y. Chen, R. W. Dibble, and R. W. Bilger, "PDF Modeling of Turbulent Nonpremixed CO/H₂/N₂ Jet Flames with Reduced Mechanisms," *Twenty-Third International Symposium on Combustion*, The Combustion Institute (1990), pp. 775–780.

Integrated Distributed Computing

For the past several decades, energy researchers have been able to draw on a variety of dispersed computing resources, from supercomputers to minicomputers to workstations to personal computers. For the most part, however, the scientist has had to treat each resource as a distinct entity, requiring its own log-in identifier and password, and having its own private file system. That paradigm is now being transformed.

Cluster Computing in the Energy Sciences

The microprocessor revolution, improved network bandwidth, and the emergence of UNIX as a de facto standard are enabling the scientist to exploit the aggregate power of his computing resources. A new paradigm, known as "cluster computing," is evolving from the confluence of these technologies. At many locations, such as Florida State University, Fermi National Accelerator Laboratory, or the Superconducting Super Collider Laboratory (see the article beginning on page 53), energy scientists now run loosely coupled "distributed-parallel" calculations using software tools such as PVM or P4 to link "farms" or "ranches" of workstations. Koszykowski, MacFarlane, and collaborators (see the article beginning on page 61) describe combustion simulations on a cluster of IBM RISC machines at the University of California at Berkeley. More and more codes designed for moderately to massively parallel processors, such as the climate models of Dannevik and co-workers at

Lawrence Livermore National Laboratory (see the article beginning on page 22), now also execute on high-end workstation clusters.

Toward an Integrated Distributed Computing Environment

As a next step, scientists eagerly await the day when they can treat a heterogeneous distributed system as a single entity rather than a collection of individual components. The ability to log in to a distributed computing environment (DCE) is just around the corner, as is the availability of a single, unified (and distributed) file and storage system. One can envision a collection of "resource centers," coupled via ESNET, coordinating to serve the Energy Research community in a distributed manner. A common way in which scientists will make use of a DCE is discussed in the article by Garner and coworkers at Argonne National Laboratory (page 48). There, under the control of a local graphics workstation, a large-scale superconductor computation is spawned to a remote, massively parallel processor, with the generated data automatically routed back to the local graphics device for postprocessing. The figure shows a conceptual overview of a DCE.

Energy Research scientists can be among the first to realize the benefits of integrated distributed computing on a wide scale. As noted in the sidebar on page 60, the bandwidth of ESNET will be increased to near-gigabit speeds over the

next several years. This improvement in networking, together with advances in hardware and in operating-system software, will provide the necessary technological base for an integrated distributed computing environment. ESNET can provide scientists with "core DCE support"—namely, services that should be basic and common to most, if not all, wide-area distributed computing environments in the Energy Research community. Examples of such services include a single global log-in with network-wide authentication and centralized accounting. Additionally, a

common global file structure, available to researchers regardless of location or the resource being used, could lead to a "national file system." Further issues to be addressed include resource scheduling and a common programming environment.

The challenge of providing an integrated distributed computing capability for the Energy Research user community is a formidable one. The benefits are so great and the opportunity so compelling, however, that the first steps are already being taken to build a test bed and begin the journey.

Conceptual depiction of a DCE. Using a single log-in through a local workstation or X-terminal, the scientist could execute an application distributed across a variety of resource centers, making use of a local midrange graphics device, a remote data base device, and a remote, massively parallel processor serving the Department of Energy.

so-called Standard Model, which describes the strong, electromagnetic, and weak interactions.

Finding nonperturbative solutions of the equations of QCD is in many ways similar to solving fluid dynamics problems (especially when employing techniques of molecular dynamics, described below). We begin with a simple—but highly nonlinear—set of equations that implicitly contain all the necessary information (up to the boundary conditions).

The starting point of calculations in QCD is the path-integral approach used in quantum field theories. To enable a numerical attack, we must first discretize space-time into a regular grid called a lattice. The quarks are then described by fields $\psi(x)$ attached to the sites of this lattice, and the gluons by special complex unitary matrices, $U_\mu(x)$, belonging to the color group $SU(3)$, attached to the links (Fig. 1). ("Color charge" is a property of quarks.) The physical observables are extracted from the expectation values

$$\langle \Omega \rangle = \frac{1}{Z} \int [d\psi][d\bar{\psi}][dU] \exp\{-S(U) - \bar{\psi}M(U)\psi\} \Omega(U, \psi, \bar{\psi}). \quad (1)$$

The function $S(U)$ in the exponential is the gluon action, containing a kinetic term and local interactions, while $M(U)$ contains the kinetic term of the quark fields and their local interactions with the gluons. Z is a normalization factor, often referred to as a partition function because of its similarity to partition functions in statistical mechanics. The integration is over the fields at each site and each link of the lattice. The fermion fields are somewhat peculiar; they are anti-commuting and therefore cannot be represented on a digital computer. Fortunately, they can be integrated analytically using Gaussian methods. The result is

$$\begin{aligned} \langle \Omega \rangle &= \frac{1}{Z} \int [dU] \det M(U) \\ &\quad \times \exp\{-S(U)\} \Omega(U, M^{-1}(U)) \\ &= \frac{1}{Z} \int [dU] \exp\{-S(U) \\ &\quad + \text{Tr} \log M(U)\} \Omega(U, M^{-1}(U)). \end{aligned} \quad (2)$$

The price we pay for the peculiarity of the quark fields is a very time-consuming computation of $\det M(U)$, where $M(U)$ is a huge but sparse matrix. We still must integrate over the U -field on each link of the lattice. As the integration is typically over more than 10^6 dimensions, the task is impossible for any kind of standard integration method. However, the exponential factor in Eq. (2) means that the contribution of most field configurations to the integral is negligible, and we can use stochastic methods. Because $\det M(U)$ is non-local, standard Monte Carlo methods such as the Metropolis algorithm are still impractical. The best algorithms for computing Eq. (2) are based on an analog to statistical mechanics.

We first introduce a fictitious time and momenta conjugate to the U -fields, each with a Gaussian distribution $\exp(-p^2/2)$. The expression $S(U) - \text{Tr} \log M(U)$ is now viewed as the potential term, and $\sum p^2/2$ as the kinetic term of a Hamiltonian. We then replace the average in Eq. (2) by an average over $\Omega(U(t), M^{-1}(U(T)))$ in fictitious time, where $U(t)$ is the solution of the equations of motion resulting from the Hamiltonian. This procedure is referred to as the Molecular Dynamics algorithm. To ensure ergodicity, momenta p are from

Figure 1. Artist's conception of a QCD simulation on a lattice, showing three quarks that make up a hadron. The inset shows the quark fields attached to a lattice element.

time to time replaced (refreshed) by new random Gaussian variables. This combination is known as the Hybrid Molecular Dynamics (HMD) algorithm.

To solve the equations of motion numerically, we must discretize the fictitious time. The resulting finite step size introduces errors that must be controlled (kept smaller than, for example, the statistical errors). Better still would be to use a few different step sizes and then extrapolate to zero step size. Since each simulation is very time-consuming, such an extrapolation is rarely possible.

Normally we apply a "leapfrog" integration scheme to solve the discretized equations of motion. This approach is simple and thus easily implemented and is also (up to round-off errors) explicitly invariant under time reversal. The latter property is important in the variant of the algorithm in which the errors resulting from discretization of the equations of motion in fictitious time are compensated by periodically performing an accept/reject Metropolis step.

This last variant, in contrast to the other two, which suffer from finite-step-size errors, is an exact algorithm known as the Hybrid Monte Carlo algorithm. Unfortunately, technical reasons prohibit us from using the Hybrid Monte Carlo algorithm for some systems of interest. In such cases, we must resort to the inexact HMD algorithm and try to control the errors stemming from finite step size.

The equations of motion to be solved contain a term like

$$\text{Tr}\{M^{-1}(U)\partial M(U)/\partial U\}.$$

Thus, for each step in the integration, the large but sparse matrix $M(U)$ must be inverted. It is this task that makes lattice QCD simulations so expensive. The inversion is done with an iterative method, usually the Conjugate Gradient algorithm.

Computational Requirements

To gain an impression of the computational demands, consider that one iteration of the Conjugate Gradient algorithm for a lattice size $16^3 \times 32$ requires between 8×10^7 and 3.8×10^8 floating-point operations, depending on the exact way in which the fermion-gluon interaction is discretized on the lattice; that is, depending on the

exact form of the matrix M . Convergence of the Conjugate Gradient algorithm can take over 250 iterations each time it is applied; for physically more interesting parameter values, more than 1000 iterations are required.

To keep the acceptance rate sufficiently high (above 50%) for the Hybrid Monte Carlo algorithm, or step-size errors reasonably small in the case of the HMD algorithm, we must use step sizes as small as $dt = 0.005$ to 0.01 .

Finally, to collect reasonable statistics for the observables, computed as fictitious time averages, the equations of motion must be integrated for a length of 1000 to 2000 time units. Combining all these numbers, we see that a state-of-the-art computation requires between 2×10^{15} and 1.5×10^{17} floating-point operations.

In a lattice QCD code, the same operations nearly always must be performed on all the lattice sites. The data needed reside either on that site or on one of its nearest-neighbor sites. Accordingly, most parts of the simulation programs, where almost all the CPU's time is spent—in particular the implementation of the Conjugate Gradient algorithm—lend themselves to rather straightforward vectorization or parallelization. Indeed, codes used for lattice gauge theory are among the most efficient on both vector and (more important) massively parallel supercomputers.

Coding Challenges

The author is a member of a large, DOE-sponsored effort known as the High-Energy Monte Carlo Grand Challenge (HEMCGC) collaboration.¹ This work was started as one of DOE's original "Grand Challenges" in 1988. Running on the ETA-10 supercomputer at the Supercomputer Computations Research Institute (SCRI) at Florida State University, the codes achieved, on one processor, sustained speeds of about 500 megaflops.

In the spring of 1990, the HEMCGC migrated its computations to the newly acquired Connection Machine model CM-2 system at the Supercomputer Computations Research Institute, where it has since been using close to one-half the available CPU time. When programmed in a high-level language, such as CMFortran or Fortran/Paris, a lattice QCD simulation

code runs at about 1.5 gigaflops on the full 64K-node CM-2.

As the CPU spends more than 95% of its time in the Conjugate Gradient inverter subroutine, its code has been given special attention. It was handcoded in CMIS, the Connection Machine Instruction Set. This treatment maximized usage of the pipelining capabilities of the Weitek floating-point chips as well as enabling bidirectional, nearest-neighbor communication in all four dimensions of the lattice simultaneously.

The resulting code runs, for optimal geometries, faster than six gigaflops on the full CM-2. In actual calculations, the programs run at sustained rates of four to five gigaflops. The range of speeds results from simulations on different lattice sizes (and hence geometries) and from the fact that we used two different forms of the fermion-gluon interaction matrix M . Computationally, they differ in the balance needed between communication and floating-point operations. Communication takes longer, and thus a code with less communication per floating-point operation runs faster.

Even with our very efficient codes, a simulation takes—for one set of parameters and one form of fermion matrix M —between a few weeks and more than a year to run on one-half of a 64K CM-2. The remarkable thing is that such resources are needed to simulate a system of spatial size $16^3 \times 32$! If we choose a box long enough to comfortably fit one of the hadrons inside, the resulting resolution is barely adequate (see Fig. 1); if, instead, we try to increase the resolution, the size of the box shrinks, and the “squeezed” hadron is affected by severe finite-size effects. The experience we have gained so far indicates that a “definite” computation of, say, the hadron spectrum will require a system with at least two to three times the number of sites in each direction. Such an extension will require supercomputers several orders of magnitude faster than the current generation.

Nevertheless, we have already achieved much, and can anticipate more, with the current generation of massively parallel supercomputers. We have obtained interesting physics results even from preliminary simulations. Algorithms can be

tested and tuned on almost realistic systems, and new and better algorithms can be developed. Especially when testing algorithms, it is important to stay as close as possible to the realistic system we wish to simulate. It is for such systems that, for example, the iterative inversion scheme takes longest to converge and thus is subject to the greatest improvement. Improvement, however, typically incurs more overhead, which on smaller and less realistic systems can impose a penalty on the new algorithm that can cause us to misjudge its worth.

The Lattice QCD Calculation

The most important theoretical goal of lattice QCD is to compute the hadron spectrum and to compare it with the experimentally known hadron masses to see whether QCD is correct. However, in the path-integral approach suitable for numerical simulations, we cannot compute the masses directly. Instead we must measure correlation functions [see Eq. (2)] such as

$$C(t) = \langle O(t)O(0) \rangle, \quad (3)$$

where $O(t)$ is an operator—or interpolating field—that couples the vacuum to the particle whose mass we want to determine. One can show that the correlation function behaves as

$$C(t) = c_1 \exp(-m_1 t) + c_2 \exp(-m_2 t) + \dots, \quad (4)$$

where m_i are the energies (masses of the states that have the quantum numbers of the interpolating field $O(t)$). The lowest mass in the channel, m_1 , can thus be extracted from the exponential fall-off of $C(t)$ as $t \rightarrow \infty$.

Other interesting quantities are the probability distribution of the quark in a meson relative to the antiquark or, in a proton, the probability distribution of the third quark relative to the first two. Figure 2 shows examples of such wave functions measured by the HEMCGC collaboration.

Wave functions give us information about the sizes of the hadrons. For example, we can extract from them the charge radii and compare those with numbers known from experiment. Knowing the sizes of particles can also serve as a diagnostic tool to help judge the reliability of

our simulations. As can be seen in Fig. 2, the wave function of the proton fills almost the entire lattice. This indicates that the lattice volume should be increased further, since the particle seems to be squeezed by its limited space. Indeed, we have found that, for the same parameters, going from a lattice of spatial size 12^3 to one of 16^3 reduces the mass of the proton by about 15%.

Quarks in nature are almost massless, a circumstance that raises two other difficulties for lattice QCD simulations. First, the convergence speed of the Conjugate Gradient inversion of the matrix M decreases roughly as $1/m_q$ with the quark mass, m_q . Thus, for quarks of smaller mass, we need more Conjugate Gradient iterations to obtain convergence. At the same time, a smaller quark mass reduces the pion mass, since chiral symmetry requires that the latter is proportional to $\sqrt{m_q}$. This relationship in turn induces larger finite-size effects, which are typically of order $e^{-m_\pi L}$.

The upshot is that when working with a smaller quark mass, we must make the lattice size accordingly bigger, again increasing the cost of the simulation. What this means is that our current simulations are feasible only at quark masses that are too large and thus are not yet realistic. As a result, although all masses come out in the right "ballpark," the ratio of the proton and rho-particle masses comes out too large, as does the pion-to-rho mass ratio. An encouraging trend, though, is that as the mass of the quark gets smaller, the mass ratios move in the right direction, getting closer to the experimentally known values.

In addition to the masses and charge radii of hadrons, we are interested in their lifetimes and other elements of their interaction matrices. (Most elementary particles, except the proton, the electron and the neutrinos, are unstable and eventually decay.) Some matrix elements, like the pion decay constant, f_π , and the "wave function at the origin" for vector mesons (which is related to their electromagnetic width), are measured relatively easily from current matrix elements between the vacuum and a one-particle state. Such measurements are already feasible, and the HEMCGC collaboration is attempting to make those calculations. Others, which are matrix elements of four-fermion operators between different one-particle states, require lattice sizes that are impractically large for currently available resources without making some approximation.

Finite-Temperature Calculations

Another, related line of lattice QCD studies is concerned with its predictions at finite temperatures. At low temperatures, QCD operates in the ordinary phase of hadronic matter (briefly described above). When the temperature is raised, the system undergoes a change to a chirally symmetric state that consists of a plasma of quarks and gluons. Lattice gauge theory aims at studying this change to determine whether it is a real phase transition (and if so, of what order) or simply a rapid crossover from one type of behavior to another.

Studies of high-temperature plasmas have an important bearing on issues

Figure 2. Cross sections of hadron wave functions. Brighter colors correspond to higher probability density. (a) Pion wave functions as a function of relative separation of quark and antiquark. (b) Proton wave function with the u quarks fixed at $(x,y,z) = (0,-2,0)$ and $(0,2,0)$. Shown is the probability distribution of the d quark. (Figures produced by T. DeGrand and M. Hecht.)

related to the physics of the early universe. At the very high temperatures and pressures prevailing immediately after the Big Bang, all matter existed in the form of a quark-gluon plasma. The answer to the question of whether the transition between this plasma phase and hadronic matter represents a real phase change can have an important bearing on our understanding of how the universe evolved. Physicists hope to explore the properties of the quark-gluon plasma in heavy-ion collision experiments that produce the necessary conditions. Such experiments are planned for the Relativistic Heavy Ion Collider (RHIC), under construction at Brookhaven National Laboratory, and for CERN in Switzerland.

Numerical studies of systems at finite temperatures reveal that pure SU(3) gauge theory—QCD without fermions—predicts a first-order deconfinement transition. QCD with four light “flavors” (another property of quarks) predicts a first-order transition that restores chiral symmetry. For QCD with only two light flavors, in contrast, we have not as yet observed a real phase transition but only a rapid crossover from the chirally broken hadronic phase to the chirally symmetric plasma phase. Two light, almost massless, quarks are found in nature, the so-called “up” and “down” quarks (in addition to heavier ones). As the lightest of the heavier ones, the “strange” quark has a mass close to the transition or crossover temperature and is therefore likely to be unimportant for understanding the physics of the transition/crossover.

Finite temperature is introduced on a four-dimensional lattice by making it asymmetric with the temporal direction shorter. The temperature is then given by

$$T = 1/(aN_t), \quad (5)$$

where a is the lattice spacing and N_t is the number of sites in the temporal direction. Lattice sizes needed for finite-temperature studies comparable in quality to the zero-temperature studies described earlier are $16^3 \times 4$ to $16^3 \times 8$. They are thus a little less demanding than the zero-temperature studies. On the other hand, to locate the phase transition or crossover from hadronic matter to a quark-gluon plasma

it is necessary to perform simulations at several parameter values. Once found, these parameters must be converted to a physical temperature. Equation (5) shows that to do so we first need to know the lattice spacing in physical units. This quantity can be obtained from a (zero-temperature) spectrum calculation that determines the hadron masses, such as the mass of the rho particle, in lattice units; that is, it produces a pure number for $m_\rho a$. We then use the experimentally known value of the rho mass to obtain the lattice spacing and finally the transition temperature in physical units.

Table 1 shows the results for the transition or crossover temperature. The term n_f denotes the number of quark flavors considered, where $n_f = 0$ refers to a quenched calculation. Experiments at RHIC should reach higher temperatures and therefore produce the quark-gluon plasma phase.

Another property we expect to change in going from the low-temperature hadronic phase to the high-temperature plasma phase is the topological susceptibility, which is related to large-scale structures of vacuum fluctuations. Strictly speaking, however, the topological charge is defined in the continuum limit only for “smooth” configurations. In a configuration produced by a numerical simulation, the charge is masked by short-distance fluctuations and lattice artifacts. To expose the underlying topological charge density, the configuration must be made smoother. This is done by a “cooling” algorithm that iteratively relaxes a configuration towards a local minimum of the action functional, $S(U)$, in Eq. (1). Since the relaxation is done with a local algorithm, the large-scale structure of the configuration remains unchanged in the

n_f	T_c (MeV)
0	230
2	140
4	100

Table 1. Critical temperature for finite-temperature QCD with n_f flavors.

Figure 3. Topological charge distribution generated by a simulation just below the transition temperature from a hadron plasma to a quark-gluon plasma. Blue indicates a positive density and orange a negative density, with lighter shades showing higher densities. (a) In a typical configuration, the distribution is masked by quantum fluctuations and lattice artifacts. (b) Charge distribution in a "cooled" configuration. (The figures, from a simulation done by Sergiu Sanielevici, were generated by the visualization software SciAn.)

cooling process and is thereby revealed, as shown in Fig. 3. Topological susceptibility is strongly suppressed in the high-temperature phase, in accordance with theoretical expectations.

Conclusions

Numerical simulations of QCD, the strong-interaction sector of the Standard Model of particle physics, are a very demanding computational undertaking. Such simulations are well-suited to massively parallel systems. However, we are limited by the current generation of supercomputers to preliminary and exploratory studies. Reliable simulations that approach realistic values of parameters such as quark mass and the coupling constant require computers running at sustained speeds of a teraflop or more. Such large-scale simulations are the theoretical

counterparts of the high-energy accelerator experiments now performed at Fermi National Laboratory in Illinois and at CERN in Geneva, Switzerland, and planned for the Superconducting Super Collider in Texas and the RHIC in Brookhaven, New York. Indeed, detailed calculations of strong-interaction effects are necessary to interpret the data collected in such experiments and to extract their implications for the Standard Model. Only large-scale numerical simulations promise to give us that capability.

Reference

1. K. M. Bitar, R. Edwards, U. Heller, A. D. Kennedy, W. Liu, T. A. DeGrand, S. A. Gottlieb, A. Krasnitz, J. B. Kogut, R. L. Renken, M. C. Ogilvie, P. Rossi, D. K. Sinclair, K. C. Wang, R. L. Sugar, M. Teper, and D. Toussaint, *Intl. J. Supercomputer Applications* **4**, 48 (1990).

range from simple "envelope-equation" models to elaborate fluid and discrete-particle simulations in one, two, and three dimensions. Because of a beam's low plasma frequency and short residence time in the accelerator, particle-in-cell techniques are particularly effective. A particle-in-cell (PIC) program uses a mesh of zones in two or three dimensions to keep track of local electric and magnetic fields. Particles that happen to reside within a zone during a particular time-step contribute charge and current terms to the field calculation that the program makes between particle time steps.

The Induction LINAC Systems Experiments (ILSE) facility and experimental sequence proposed by the Lawrence Berkeley Laboratory represents the main thrust of the U.S. program.² In ILSE, as in almost all modern accelerators, transverse confinement is afforded by quadrupole electrostatic or magnetic lenses. Such lenses focus the beam in one transverse direction while defocusing it in the other. The successive lenses are of opposite polarity, leading to the term "alternating-gradient" (A-G) focusing. Since the beam is wider in a given direction as it passes through an element that is focusing in that direction, and since the focusing force on the particles is proportional to their distance from the axis, the focusing forces

are stronger than the defocusing forces, so that the particles experience a net focusing force.

In most conventional accelerators, a "matched" beam condition is obtained by balancing the average focusing force against the transverse particle temperature. However, for the space-charge-dominated beams of greatest interest in HIF research, the balance is predominantly between the electrostatic self-field of the beam and the confining force (Fig. 1).

We use the three-dimensional PIC code WARP (Ref. 3) to follow the dynamics of a space-charge-dominated beam in a user-specified lattice of sharp-edged focusing and bending elements. The self-field is obtained by solving Poisson's equation on a "warped Cartesian" mesh, one that is bent along with the accelerator centerline. Special "residence corrections" applied to the forces acting on particles ensure that neighboring particles receive similar impulses as they traverse a sharp-edged element regardless of the number of timesteps on which they land within the element. Correction terms that account for rotation of the coordinate system are incorporated into the equations of motion; Poisson's equation is solved in the warped coordinates, yielding the electrostatic potential.

Longitudinal Compression in HIF Drivers

In an induction linear-accelerator driver, current is amplified by a drift-compression process wherein the beam's tail is accelerated more than the head, thereby enhancing the current. This uneven acceleration results in a shorter beam with a higher concentration of ions and thus higher current. The MBE-4 facility at Lawrence Berkeley Laboratory⁴ has been used to study this effect experimentally. We have used the WARP code (and others with simplified geometries) to model the effect; these calculations were done at the National Energy Research Supercomputer Center.

Figure 2a shows the results of modeling the initial beam. The upper half of the figure is a top view of the beam (x - z projection) and the lower half a side view (y - z projection). The beam is even longer and thinner than the figure suggests; the tick

Figure 1. Periodic focusing systems, showing (a) a single-particle orbit without space charge and (b) effect of space charge in depressing the betatron phase advance per cell.

marks are labeled in meters relative to the initial centroid of the beam. Because of the A-G focusing employed, the beam is fat in x where it is thin in y , and vice versa. Figure 2b shows the initial position and velocity in the axial direction (z - v_z). The various colors indicate where in the beam a group of particles was located at the beginning of the simulation.

The drift-compression process comes to an end when space-charge repulsion slows the inward motion to a standstill. We have used simulations and theory to obtain the scaling of both the degree of compression and the distance to peak compression with beam current and velocity tilt. Figure 2c shows the beam after it has moved 9.144 m down the system; at this point it is near peak compression. The beam has grown fatter as a result of compression, and some axial mixing has occurred. The final (z - v_z) plot, in Fig. 2d,

shows that the inward motion (in the beam frame) of the beam's ends has slowed as a result of space-charge repulsion by the body of the beam. A number of the particles have passed through the beam's center.

The velocity profile must be carefully tailored if the head-to-tail spread in axial velocity is to be effectively nullified at the point of peak compression. Just as an optical lens focuses the different colors of visible light at different points (a phenomenon called chromatic aberration), a magnetic lens focuses particles of different speeds onto different spots. Too great a spread in the particle velocities will lead to excessive chromatic aberration in the final focusing optics and hence to too large a focal spot at the fusion target. Misaligned elements and cross-sectional enlargement of the beam resulting from axial compression can lead to emittance

Figure 2. (a) Initial MBE-4 beam as modeled using the WARP code. The beam is moving from left to right. Colors indicate initial position of particles in the z direction. The upper image is a top view of the beam (x - z projection) and the lower image a side view (y - z projection). (b) Axial velocity versus axial position (v_z vs. z) at the start of the MBE-4 simulation. Faster particles are at the left. (c) Simulated MBE-4 beam after it has moved 9.144 m, showing x - z (upper image) and y - z (lower image) projections. The beam has grown much shorter, and somewhat fatter, as a result of the drift compression process used to shorten the pulse and enhance the current. (d) MBE-4 simulation at 9.144 m near the point of peak compression, showing v_z - z .

growth as particles sample unwanted multipole fields. We are using both simulation and theory to study these effects.

Combining Beams in HIF Drivers

The principal challenge in designing accelerator systems for high-intensity beams is to preserve beam quality. Applications of charged-particle-beam technology such as electron-positron colliders, free-electron lasers, and HIF drivers all demand beams with both high current and low emittance. The requirement for low emittance has motivated us to search for possible causes of emittance growth and the resulting limits on beam brightness. One such limiting mechanism is the conversion of space-charge potential energy into kinetic energy, which occurs whenever a beam with a nonuniform current profile propagates in a linear focusing channel.

An extreme case of nonuniform current density, and one of particular interest to HIF, occurs when several beams are combined to produce a single beam of higher total current. Beam-merging is useful because it is difficult to transport a high-current beam at low energy. In collaboration with our colleagues, we have studied the particular case of five beams formed simply by placing a mask with five holes in the path of a high-intensity electron beam.⁵ As the beams merge, a fluorescent screen shows a generally homogeneous spot that corresponds to the total area of the beam. However, the initial configuration of the beam reappears further down the beam line. This is an interesting and unexpected observation. This phenomenon occurs despite the substantial growth in emittance that results from the nonuniform current density. Apparently enough particles describe orbits that are nearly linear so that the original structure of the transverse beam is imaged after one-half of a betatron oscillation.

We used the program SHIFT to simulate numerically the propagation of the merging beams.⁶ The simulation must represent the experimental apparatus very accurately to quantitatively reproduce the observed reappearance of the original configuration. We used a gray-scale graphic representation of the simulation data to reproduce the results from

the fluorescent-screen monitors. This process takes advantage of the inherent ability of the eye to recognize details in patterns that would be very complex to represent mathematically. Figure 3 compares the results of the simulation with the experimental observations.

The Next Linear Collider

There is now general agreement in the high-energy physics community that to continue exploring the energy frontier with electron-positron interactions, we must abandon circular colliders for linear colliders. This realization has led to active research by physicists throughout the world on the next generation of linear colliders. We are now at the point where we can discuss in fair detail the design of such a "next linear collider," or NLC.⁷

The two key parameters of the NLC are its energy and luminosity. A broad consensus on the values of these parameters has emerged over the past couple of years. The energy should be 0.5 TeV (total energy of electron plus positron), upgradable to at least 1.0 TeV. One reason for this choice of energy range is the great potential of such a collider for doing significant high-energy physics research in the era of the Superconducting Super Collider (see the article beginning on p. 53). Another is that this energy range is a natural next step, as it is greater than that of the Stanford Linear Collider (SLC) by a factor of 5 to 10. To obtain a collision event rate sufficient to perform detailed measurements, the luminosity of the collider should increase with the square of its energy. An NLC in the tera-electron-volt range requires a luminosity between 10^{33} and $10^{34}/\text{cm}^2/\text{s}$.

An energy increase of the magnitude needed can be obtained in two ways: either by increasing the length of the collider 10 to 20 times over that of the SLC (3 km) or by raising the strength of its accelerating field to 5 to 10 times that of the SLC gradient (20 MV/m). The current consensus is that we should first increase the accelerating field by a factor of 3 to 5, up to about 50 to 100 MV/m. To limit the radiofrequency (rf) power required, the field should be generated by structures similar to those used by the SLC but at a higher frequency of 10 to 30 GHz. At the

Stanford Linear Accelerator Center, the frequency choice for the NLC is 11.4 GHz, or four times the present SLC frequency. The trade-off between accelerator length and strength of the accelerating field is ultimately governed by overall cost and upgradeability. A broad optimum occurs at the point where the linear costs (accelerating structure, magnets, tunnel, etc.) equal the cost of the rf power source.

We can now sketch a rough design of a linear collider that can achieve both the desired energy and luminosity. Figure 4 shows a possible layout. The design calls for two complete linear accelerators, one for electrons and one for positrons. Each LINAC is supplied with particle beams by a damping ring that is followed by a preacceleration section consisting of two bunch compressors and a 16-GeV LINAC. After passing through the main linacs and the final focusing system, the beams collide at a small crossing angle inside a large particle detector.

Designing a Klystron for the Next Linear Collider

A klystron is a device that amplifies or generates rf electromagnetic radiation (Fig. 5). We are simulating designs of X-band klystrons for the Next Linear Collider by running the two-dimensional programs Condor and Mask on the CRAYs at the National Energy Research Supercomputer Center in Livermore. The high peak fields involved pose problems for the output cavities and output coupler. The waveguide coupler causes special concern because it perturbs the symmetry of the output cavity. The result is the generation of transverse fields that can deflect the beam into the walls of the klystron. As klystron beams for the XC-series of tubes have sufficient energy density to melt copper within one modulator pulse, they must be carefully designed to prevent their interception.

The fully electromagnetic PIC simulation concerns the part of the beam tube (Fig. 5) that includes all the rf cavities. Figure 6 is a single snapshot, generated by Condor, of the particle distribution. Cavities are simulated by a port with an rf voltage located on the outer well. Phase and amplitude of the rf are determined by solving equations for the equivalent circuit

Figure 3. Phosphor screen pictures obtained from experiment, and comparable simulation plots, at various distances z from the beam mask.

Figure 4. Schematic of a possible layout for the Next Linear Collider (not shown to scale). Most of the length (and cost) is in the two "main LINACs."

Figure 5. Schematic drawing of a high-power, high-gain klystron. Gain is determined by the number of gain cavities.

Figure 6. Output of the Condor program for a klystron. Cavities are simulated by rf voltages across a port in the wall. Equivalent circuit equations are used to determine rf voltage phase and amplitude for each cavity.

Figure 7. Output coupler for the X-band klystron designed using the three-dimensional rf modeling capabilities of the PIC program MAFIA. The arrows show the direction and strength of the electric field in the operating mode.

for each cavity. This technique saves an immense amount of computer time that would otherwise be required to simulate the spaces in the cavities that do not contain the beam and also leads very quickly to the steady-state condition that is the objective of the calculation.⁸

The most critical portion of a high-power klystron is its output cavity. It is the design of this cavity that largely determines the klystron's efficiency, and it is here that rf high-voltage breakdown is most likely to occur. Because the output cavity includes the waveguide for extracting power, the simulation is intrinsically three-dimensional, a fact that is ignored in the simulation shown in Fig. 6.

Figure 7 shows the output cavity we designed for the X-band klystron with the three-dimensional program MAFIA on the IBM RS/6000 workstation. The arrows indicate the strength and direction of the rf electric field. It is evident that the waveguide coupler introduces a strong asymmetry in the field distribution. The goal of our three-dimensional modeling effort is to design the coupler cavity and the output coupler with dimensions that provide good power coupling to the waveguide and a field distribution that is symmetric near the axis. It is also important to minimize high local peak fields while maintaining efficient extraction of the rf power from the beam.

In principle, this modeling effort can replace the cold-test procedure for the cavity that is normally the basis of all tube and accelerator structure design. In practice, it is used to complement the experimental process by reducing the number of iterations that engineers must make with a trial-and-error approach.

In many ways, the more powerful workstations and the resources available on the CRAY supercomputers are interchangeable. The three-dimensional program MAFIA is available on workstations, and the similar program ARGUS is available on the CRAY system. Although both of these programs have rf field-solving

and three-dimensional PIC capabilities, it is not truly practical to run them in the three-dimensional mode with particles on either platform. True production runs for three-dimensional PIC calculations will probably have to await the availability of massively parallel computers.

Acknowledgments

This work was supported by the U.S. Department of Energy under contracts DE-AC03-76SF00515, DE-A105-83ER40112, and W-7405-ENG-48.

References

1. W. J. Hogan, R. Bangerter, and G. L. Kulcinski, "Energy from Inertial Fusion," *Physics Today* (September 1992), pp. 42-50.
2. T. J. Fessenden, et al., "ILSE, The Next Step Toward a Heavy Ion Induction Accelerator for Inertial Fusion Energy," *Proc. 14th International Conf. on Plasma Physics and Controlled Nuclear Fusion Research*, IAEA, Wurzburg, Germany, Sept. 30-Oct. 7, 1992.
3. A. Friedman, D. P. Grote, and I. Haber, "Three Dimensional Particle Simulation of Heavy Ion Fusion Beams," *Phys. Fluids B* **4**, 2203 (1992).
4. T. J. Fessenden et al., *Proc. 1987 Particle Accelerator Conf.* (IEEE Cat. No. 87CH2387-9, 1987), pp. 898-902.
5. I. Haber, D. Kehne, M. Reiser, and H. Rudd, "Experimental, Theoretical and Numerical Investigation of the Homogenization of Density Nonuniformities in the Periodic Transport of a Space-Charge Dominated Beam," *Phys. Rev. A* **44**, 5194-5204 (1991).
6. I. Haber, in *Proc. Workshop on High-Current, High-Brightness, and High-Duty Factor Injectors*, G. H. Gillespie, Y. Y. Kuo, D. Keefe, and T. P. Wangler, Eds., AIP Conf. Proc. No. 139 (AIP, New York, 1985), p. 107.
7. R. Ruth, "The Next Linear Collider," *18th Annual SLAC Summer Institute on Particle Physics*, Stanford, CA, July 1990, SLAC-PUB-5406.
8. K. Eppley, "Algorithms for the Self-Consistent Simulation of High-Power Klystrons," *Linear Accelerator and Beam Optics Codes Workshop*, San Diego, CA, AIP Conference Proceedings 177, C. Emlinger, Ed., pp. 313-322 (1988).

system is viewed as a collection of nuclei and electrons. The major advantage of such an approach is that predictions of properties and identification of the mechanisms that give rise to phase transformations do not depend on input parameters. Rather, they are consequences of the properties of the underlying electron "glue" responsible for metallic cohesion. The disadvantage of this approach is that it is very difficult to implement in all but relatively simple situations.

Almost of necessity, any *ab initio* theory of phase stability in alloys must include three key elements. The first is a theory of many-electron interactions sufficiently accurate to enable calculation of the small energy differences between different alloy phases. Second is a means of implementing this theory for relevant phases (elemental, ordered intermetallic, disordered solid-solution). Third are techniques for treating compositional rearrangements at finite temperatures—these give rise to the most important entropic contribution to the free energy of the disordered solid solution.

We outline here one such first-principles approach. It assumes that much can be learned about metallic alloys by first studying the electronic structure and energetics of ideal, random solid solutions. We can visualize a random solid solution as an underlying periodic lattice on which the atoms of the alloy are randomly distributed (see Fig. 1). In many respects this is the natural, high-temperature solid-state phase. The reason for this is simply entropic. As indicated above, there is a contribution to the free energy that results from the different possible ways that the atomic species can be distributed amongst the available lattice sites. Since this term is negative (stabilizing), and is proportional to the absolute temperature, there should always be some temperature at which the disordered solid-solution phase is stable with respect to the pure elements and to the possible ordered compounds. Thus, in addressing questions of alloy stability, it makes sense first to consider the properties of the high-temperature disordered phase and then to study its instabilities either to

Figure 1. (a) Three unit cells of an idealized face-centered cubic (fcc) lattice. In the fcc lattice, the basic structural unit can be thought of as a cube with atoms on each corner and at the center of each face, whereas in the body-centered cubic (bcc) lattice there is an atom at each corner of the cube and an atom at its center. (b) In an $A_cB_{(1-c)}$ random solid solution, the A and B (red and blue) atoms are randomly distributed on the sites of an underlying fcc lattice with probabilities c and $(1 - c)$, where c is the concentration of the A species. (c) In the $L1_2$ ordered structure, which forms at the composition $A_{0.25}B_{0.75}$, the A atoms occupy the corner positions in the basic unit cube and the B atoms occupy the face centers. (d) The $L1_0$ ordered structure forms at the composition $A_{0.50}B_{0.50}$. In this structure, the A and B atoms are arranged in alternating layers.

phase separation (separation into regions of pure A and pure B) or to the formation of specific ordered phases such as those illustrated in Fig. 1.

In addition to providing us with a reference state from which to study phase stability, disordered solid solutions are interesting in their own right. Indeed, they often find applications based on some specific property that is characteristic of the disordered phase. For example, the high electrical resistivity that results from compositional disorder is exploited in heating elements. Disordered alloys, especially those with face-centered cubic (fcc) lattices (Fig. 1), also form the basis of most commercial alloys, as they are more likely than ordered alloys to be ductile. Indeed, alloy designers have only recently developed ordered intermetallic alloys that are ductile. However, disorder plays a role even in these, as they are typically used at compositions that do not correspond to the simple ratios required by the stoichiometric compound or contain additions that are randomly distributed.

Electronic Structure of Disordered Alloys

Most modern first-principles calculations of the electronic structure and energetics of condensed matter are based on the local density approximation (LDA) to density functional theory (DFT). Density functional theory is, in principle, an exact method of calculating the energetics of an electron system in the field of the atomic nuclei, in particular the total energy of the ground state ($T = 0$ K). The LDA converts DFT into a practical, albeit approximate, computational scheme. The LDA-DFT methodology requires as inputs only the composition and atomic numbers of the constituent species. Thus, it provides a direct link between the properties of a material and its underlying electronic structure that does not depend on any adjustable parameters.

In the LDA-DFT method, the problem of calculating the charge density and energy of the ground state is reduced to solving a set of Hartree-like self-consistent field (SCF) equations for the motion of a single electron moving in the effective field generated by the nuclei and the other electrons. The potential that enters in the

Schrödinger equation, which is at the heart of the SCF equations, is calculated from the charge density, which, in turn, is generated from the solutions of the Schrödinger equation. Typically, the SCF equations are solved iteratively, the major task being to solve the Schrödinger equation (or, for heavy elements, its relativistic equivalent, the Dirac equation) for systems containing roughly 10^{23} atoms.

For ordered systems, solution of the Schrödinger equation can be greatly simplified by making use of the periodicity of the crystal lattice. Thus, a wide variety of so-called band-theory methods are now available for calculating the electronic structure and energy of ordered solids. For disordered alloys there is no such simplification, as the crystal potential in each atomic cell is different from that in every other because every atom is surrounded by a different arrangement of other atoms. This complexity is further compounded by the fact that the way in which atomic species are distributed amongst the available lattice sites is not known. Consequently, calculated properties must be averaged over an ensemble of possible configurations consistent with the concentrations of the constituent species.

We have extended the implementation of the LDA-DFT methodology to random substitutional alloys via the LDA Korringa, Kohn, and Rostoker coherent potential approximation (LDA-KKR-CPA) method developed in collaboration with our colleagues Frank Pinski at the University of Cincinnati, Duane Johnson at Sandia National Laboratories–Livermore, Balazs Györffy of the University of Bristol, Julie Staunton of the University of Warwick, both in the UK, and Beniamino Ginatempo at the University of Messina in Italy. In its present form, this method is the basis of all our calculations of the electronic structure, energetics, and phase stability of disordered alloys.

The LDA-KKR-CPA method uses the coherent potential approximation for the complex task of averaging over configurations. The CPA is a mean-field theory of the effects of disorder. It directly yields the averaged Green's function, from which we can obtain other quantities of interest, such as the electronic charge density and energy, by integrating over the occupied

electron energies (the so-called occupied band).

As is typical of mean-field theories, the averaging process implicit in the CPA requires the solution of a set of integral equations. The KKR-CPA method formulates these equations by using multiple-scattering Green's function techniques. The essence of the CPA is to model the effects of disorder on the electronic structure through the properties of an effective scatterer (or single-site coherent potential) that is the same on every site. Thus, the problem of treating scattering from a random array of real scatterers is replaced by the much simpler problem of describing the scattering from a periodic array of effective scatterers. In the KKR-CPA method, the effective scatterer chosen is one that gives, in some sense, the "best" representation of the real disordered system.

Solving the equations of the KKR-CPA method turns out to be a computationally demanding task, requiring repeated evaluation of a three-dimensional integral over the Brillouin zone (the unit cell of reciprocal space). At each point in the Brillouin zone, calculation of the integrand requires the inversion of a matrix—the KKR matrix—that has the dimension $N(l_{\max} + 1)^2$, or $2N(l_{\max} + 1)^2$ if the conduction electrons are treated relativistically, where N is the number of sublattices in the basic unit cell ($N = 1$ for simple fcc and bcc structures; see Fig. 1) and $l_{\max}$ is the angular momentum cutoff used in the calculation. When dealing with alloys of transition metals, we typically include s, p, d, and f states (thus $l_{\max} = 3$). Hence, for fcc and bcc lattices (treating the electrons nonrelativistically), the matrix is of dimension 16, whilst for the perovskite structure (five atoms per unit cell and relativistic electrons; see below) the matrix size is 160. To complicate matters further, the integrand can be nearly singular over a complicated surface in reciprocal space.

Despite these difficulties, we have developed efficient numerical methods for solving these equations. Still, the number of matrix inversions is large—typically 300 to 2000 per energy. For a single self-consistent field iteration, the KKR-CPA equations must be solved for 100 to 150 energies. However, taking the inverse of a complex matrix is a task to which vector

processors are well-suited. Consequently, the availability of successive generations of Cray supercomputers has contributed greatly to the development of the KKR-CPA method. Indeed, constructive interplay between theoretical, algorithmic, and computational developments has been a general feature of the whole field of first-principles calculations of the properties of solids.

One quantity of interest in a disordered solid solution is its heat of mixing. This is a measure of the energy required to form the alloy from its constituent species. If the heat of mixing is positive, the alloy should phase-separate at low temperatures; if it is negative, the alloy should form ordered compounds. For a binary $A_cB_{(1-c)}$ alloy at $T = 0$ K, the heat of mixing is simply $E_{\text{mix}} = E_{\text{alloy}} - cE_A - (1 - c)E_B$, where E_{alloy} , E_A , and E_B are the energies of the disordered alloy and of the A and B pure metals, respectively. Simple arguments based on regular solution theory predict that E_{mix} is proportional to $c(1 - c)$.

Figure 2 shows the results of our calculations of energies of mixing for $\text{Cu}_c\text{Zn}_{(1-c)}$ and $\text{Ag}_c\text{Al}_{(1-c)}$ in the fcc phase. The former displays a behavior that is close to regular solution theory, while the latter does not. Both are consistent with measured enthalpies of mixing and with a number of other properties of these systems.

Figure 2. Energies of mixing of two fcc solid solutions, $\text{Cu}_c\text{Zn}_{(1-c)}$ and $\text{Ag}_c\text{Al}_{(1-c)}$ alloys.

Figure 3. Short-range-order diffuse scattering patterns calculated by the Györfy-Stocks method for (a) $\text{Cu}_{0.93}\text{Pd}_{0.07}$, (b) $\text{Cu}_{0.75}\text{Pd}_{0.25}$, and (c) $\text{Cu}_{0.60}\text{Pd}_{0.40}$. The calculated peaks are in good agreement with experimental values.

Ordering and Phase Stability

Rather than reviewing current approaches to ordering and phase stability, we show here some results that illustrate what can now be achieved by building on the LDA-KKR-CPA theory of the ground state.

Even in the disordered state, atoms do not in general arrange themselves randomly. In most alloys, they have a propensity either to cluster or to show short-range order (SRO). In an alloy that exhibits clustering, the probability of finding like atoms as neighbors is greater than would be expected from a random distribution. Similarly, in an alloy that exhibits SRO the probability that unlike atoms are near neighbors is greater than in a random distribution. The propensity of atoms in the disordered state either to cluster or to exhibit SRO can be quantified in the so-called Warren-Cowley SRO parameter, $\alpha(\vec{q})$ and can be measured directly in diffuse-scattering experiments with x rays, electrons, and neutrons.

Figures 3 and 4 show the results of calculations of SRO diffuse scattering for selected $\text{Cu}_c\text{Pd}_{(1-c)}$ and $\text{Pd}_c\text{Rh}_{(1-c)}$ alloys. The calculations are based on the so-called Györfy-Stocks theory of ordering and phase stability. In this method, SRO diffuse scattering is calculated from the results of an LDA-KKR-CPA calculation of the electronic structure in the homogeneously disordered phase. Other than the CPA, the only major approximation is to treat the configurational entropy within mean-field theory. The power of the theory lies in the fact that it can be executed without recourse to arbitrary or adjustable parameters.¹ Thus, the predictions of the theory regarding the nature of the ordering process are couched in terms of the underlying electronic structure.

Figure 3 shows calculated SRO diffuse-scattering patterns for copper-rich $\text{Cu}_c\text{Pd}_{(1-c)}$ alloys at three concentrations. The calculated peak positions in the SRO diffuse-scattering maps are indicative of this system's propensity to form long-period ordered structures at low temperatures and are in excellent agreement with experiment for all concentrations.

The diffuse scattering pattern calculated for $\text{Pd}_c\text{Rh}_{(1-c)}$ alloys is completely different from that for $\text{Cu}_c\text{Pd}_{(1-c)}$ (Fig. 4a is an

example). In $\text{Pd}_c\text{Rh}_{(1-c)}$ alloys, the positions of SRO diffuse-scattering peaks are independent of composition and are indicative of low-temperature phase separation. Indeed, as shown in Fig. 4b, the phase boundary calculated from the mean-field-theory free energy is in quite good agreement with what is observed experimentally, especially considering that the only inputs to the theory are the atomic numbers of palladium and rhodium.

Parallelization of the KKR-CPA Method

Despite major achievements in first-principles calculations of the properties of condensed matter, researchers have barely scratched the surface of this vital discipline. Accordingly, as we address increasingly complex problems we face a continued demand for more powerful methods and the computational resources needed to execute them.

From a computational perspective the 1980s might be characterized by the increasing power and availability of vector computers. The 1990s promise the rapid development of massively parallel computers. Thus, it seems opportune to ask if these anticipated advances in computational power will be accompanied by progress in first-principles calculations of materials properties comparable to that achieved by vector machines. In beginning to address this issue, we undertook the parallelization of the KKR-CPA code.

The KKR-CPA code offers a number of opportunities for effective parallelization. As we have noted, the bulk of the computational effort demanded by the code is expended on inverting the KKR-matrix at the large number of $\vec{k}$ points (300 to 2000) necessary to evaluate the $\vec{k}$ -space integral required to solve the KKR-CPA equations for a single energy. In addition, we must solve the KKR-CPA equations for a large number of energies throughout the occupied energy band in order to perform the energy integration needed to obtain the charge density.

Both the $\vec{k}$ -space integration and the energy integration offer opportunities for parallelization. In the former, we can assign the work associated with setting up and inverting the KKR-matrix at a given $\vec{k}$

point to an individual processor. In the latter, the work associated with solving the KKR-CPA equations for a single energy is assigned to a given processor. In both cases, it is trivial to ensure that the calculation being performed on one processor is independent of all others.

Which strategy is the most effective depends to some extent on the target parallel machine. We have been using MIMD (multiple-instruction, multiple-data) machines with limited to moderate parallelism but relatively powerful individual processors. These machines consist of CRAY Y-MPs and CRAY-2s with four to eight processors, located at the National Energy Research Supercomputer Center, Florida State University, Ohio State University, and Cray Research, Inc., and a 128-node Intel iPSC/860 located at Oak Ridge National Laboratory. Accordingly,

Figure 4. (a) Calculated short-range order diffuse scattering pattern for a $\text{Pd}_{0.25}\text{Rh}_{0.75}$ alloy. (b) Phase diagram of $\text{Pd}_c\text{Rh}_{(1-c)}$ alloys. The magenta shows the calculated miscibility gap. The blue shows our experimentally determined values.

we parallelized the code over the energy integration by adopting a master-slave paradigm. A single processor, the master, knows the full problem, including performance of the energy integration, construction of the charge density, iteration of the LDA equations, and calculation of the total energy. Each of the remaining nodes is assigned the subproblem of solving the KKR-CPA equations for a single energy. The master generates a pool of tasks (energies) that it disburses to the slaves. When a slave node has completed its task, it sends the solution back to the master, which then sends it another energy. This procedure continues until all the energies have been calculated.

In general, solving the KKR-CPA equations requires different numbers of iterations at different energies. Consequently, we had to consider load-balancing issues. Since the energy integration is performed over a contour in the complex plane, and energies that lie far off in the complex plane take fewer iterations, we order energies according to the size of their imaginary part, those with a small imaginary part being placed at the head of the task queue.

Because the node calculation primarily involves setting up and inverting the KKR-matrix, the process is highly vectorizable. Thus, on vector processors such as the CRAY Y-MP, the algorithms execute at speeds near their theoretical limits. For

large systems, we have obtained execution rates of the order of 20 and 300 megaflops on single Intel i860 and CRAY Y-MP processors, respectively.

Since no slave-slave and little master-slave communication is required, the method also parallelizes effectively. Indeed, for large systems the code can be made essentially 100% parallel. To illustrate this, we obtained performance figures for a load-balanced calculation of the electronic structure of $\text{Ba}_{0.6}\text{K}_{0.4}\text{BiO}_3$, a high-temperature superconductor with a perovskite crystal structure (Fig. 5). Since we performed a fully relativistic calculation, the dimension of the KKR matrix was 160. For this problem we obtained execution rates of over 2.3 gigaflops on an eight-processor CRAY Y-MP and over 2.5 gigaflops on ORNL's 128-node Intel iPSC/860.

In addition to running the code on true parallel machines, we have also implemented it on ten networked IBM workstations under Parallel Virtual Machine (PVM). PVM is a UNIX-based software package that allows the creation of a (virtual) parallel machine from a set of independent computers. The software uses standard communication pathways (Internet, Ethernet etc.). By using PVM to

Figure 5. Idealization of the perovskite-structure high-temperature superconductor $\text{Ba}_{0.6}\text{K}_{0.4}\text{BiO}_3$. This system comprises five interpenetrating simple cubic sublattices. One of these is disordered and contains barium (green) and potassium (yellow) randomly distributed; the other four, which are ordered, consist of one bismuth sublattice (blue) and three oxygen sublattices (red).

Figure 6. Performance, in megaflops per \$million, of the relativistic, complex-lattice KKR-CPA code on three different systems—an eight-processor CRAY Y-MP, a 128-node Intel iPSC/860, and Parallel Virtual Machine software implemented on a cluster of ten networked IBM workstations—during a calculation of the electronic structure of $\text{Ba}_{0.6}\text{K}_{0.4}\text{BiO}_3$.

create virtual parallel machines ranging from ten IBM RS/6000 workstations to a network of several CRAYs, we have obtained performances between 250 megaflops and 9 gigaflops.

An analysis of cost performance, in contrast to raw performance, throws an interesting sidelight on our parallelization work. The results are shown in Fig. 6. We attained the best cost/performance ratio on a network of ten IBM workstations, but the Intel iPSC/860 also outperformed traditional Cray machines. This gratifying performance, when coupled with high execution rates, shows that indeed parallel processing holds great promise for first-principles approaches to the properties of disordered alloys. We are optimistic that the next decade will see advances in our understanding of these and other types of materials as great as those obtained over the last.

Acknowledgments

The illustrations in this article were produced with the help of Ross Toedte and Ray Flanery. In addition to the individuals mentioned in the text, it is a pleasure to acknowledge significant contributions by Sam Faulkner, Florida Atlantic University, Bill Butler, Oak Ridge National Laboratory, Andrea Barbieri, Meiko plc., Bristol, UK, Walter Temmerman and Dzidka Szotek, Science and Engineering Research Council, Daresbury, UK, Julie

Staunton, University of Warwick, UK, Herman Winter, Kernforschungszentrum, Karlsruhe, Germany, Tony Gonis, Patrice Turchi, and Marcel Sluiter, Lawrence Livermore National Laboratory, and Phong Vu, Cray Research, Inc.

This work was supported by the Division of Materials Science, Office of Basic Energy Sciences, U.S. Department of Energy (GMS, DMN), by the Applied Mathematical Sciences Research Program, Office of Energy Research (GMS, DMN, WAS, GAG), by the Assistant Secretary of Conservation and Renewable Energy, Office of Industrial Technologies, Advanced Industrial Concepts Materials Program (DMN), all under subcontract DEAC05-84OR21400 with Martin-Marietta Energy Systems, Inc., and by a grant of computer time at NERSC from DOE-BES-DMS.

Note

1. The applications shown in this article rest on one further assumption, that the full expression for the electronic energy used in the formal derivation of the KKR-CPA theory of $\alpha(\vec{q})$ can be approximated by the so-called band-structure energy alone. This approximation neglects certain charge-transfer contributions but is expected to be valid for systems where charge transfer is small, as is the case for the systems discussed in the text.

The National Education Supercomputer Program

The National Education Supercomputer Program (NESP) was inspired by the conviction that government must increasingly use its world-class scientists, engineers, and facilities to help motivate and educate America's students and teachers in math and science. Sponsored by the Department of Energy, NESP provides a unique instructional tool for science and mathematics students in junior high school, high school, and community colleges throughout the United States.

The program consists of several elements, including a CRAY X-MP supercomputer, the high-school science student honors program in supercomputing, the high-school teachers' supercomputing workshops, and other special workshops such as one for the Southwest Indian Polytechnic Institute.

The goals of the NESP include:

- Performing large-scale computer simulations and modeling to enhance classroom science and mathematics instruction and to demonstrate the importance of supercomputers in scientific research.
- Supplying a reliable, progressive computing environment that supports efficient, wide-range network connectivity.
- Developing distributed software that allows students nationwide to use their local microcomputers in conjunction with the supercomputer at NERSC.
- Exploring and developing projects that foster interdisciplinary education.

- Providing a mechanism to facilitate communication between teachers throughout the country and to encourage sharing curriculum and ideas among teachers, students, and scientists.

The National Education Supercomputer

The National Education Supercomputer (the NES) is a CRAY X-MP donated to the program by Cray Research, Inc. The machine, housed and operated at NERSC, is used exclusively for educational purposes. Any high-school science or math class with a modem and a personal computer can use this supercomputer; up to 100 classes can use the NES at any one time.

The DOE High-School Science Student Honors Program in Supercomputing

The DOE High-School Science Student Honors Program in Supercomputing was piloted and established at NERSC in 1985. It is designed to identify and honor high-school students who demonstrate a high proficiency in science and mathematics. The program motivates and encourages participating students to continue post-secondary studies in mathematics, science, engineering, and computer science disciplines that support DOE's overall mission. In doing so, it introduces students to DOE's research programs and laboratory capabilities. Also known as the "Superkids" program, it brings one student from each state, the District of Columbia, Puerto Rico,

American Samoa, a representative from the Department of Defense Dependents Schools, and eight foreign students to the Laboratory each summer. Typical student projects include climate modeling codes, sophisticated telecommunications, scientific visualization, and programming on supercomputers.

Students gain hands-on experience during individualized study sessions in which

they use microcomputers and workstations networked to the NES. They attend lectures on state-of-the-art supercomputer applications with Laboratory scientists and programmers who are experts in a number of disciplines, including aerodynamics, climate, chemistry, physics, fusion energy, engineering, robotics, graphics, and computer modeling. These scientists and programmers share their computing and scientific experience with the students and assist them in exploring their individual interests in computational science. The program is dedicated to reaching the students who will constitute the next generation of researchers and computing science specialists.

Teachers' Supercomputing Workshops

Teachers from across the nation attend tutorials and work on projects to learn how to use the NES and to gain insight on how it can be used in their own classrooms. In addition, the teachers collaborate with Laboratory professionals and other participants in the workshop to develop curriculums. After the summer workshop, the teachers are expected to complete curriculum development and to plan how to introduce the material to students throughout the following year. At the end of the school year, teachers submit a report detailing the curriculum built, telling how it was used in their schools, and summarizing their students' project events and results. This information is collected and published for distribution to future teacher participants and interested schools.

An important goal of the teachers' workshops is the development of "master" teachers. Master teachers then conduct workshops for teachers in their local areas, giving more and more teachers and students access to this national resource. As more "master" teachers are

developed and as the teachers' workshops increase in scope and diversity, the program will touch at least 200 to 250 teachers each year, who will in turn influence thousands of students.

The National Energy Research Supercomputer Center

The National Energy Research Supercomputer Center (NERSC) is the principal supplier of production high-performance computing and networking services to the nationwide energy research community. For years, the Center—located at and operated by Lawrence Livermore National Laboratory—has met the general-purpose computing needs of energy researchers by providing a centrally located supercomputer resource accessible to remote users by a high-speed network. NERSC was the first such center to offer this service, and it served as a model for other centralized supercomputing facilities, including some of the centers funded by the National Science Foundation.

Originally called the Controlled Thermonuclear Research Center and subsequently known as the National Magnetic Fusion Energy Computer

Center, NERSC was formed in 1974 in response to the computational demands of the national magnetic fusion energy program. In 1983, DOE's Office of Energy Research, noting NERSC's success in the magnetic fusion community, expanded the Center's role to serve the full range of national energy research programs—high-energy and nuclear physics, the Superconducting Super Collider, basic energy sciences, health and environmental research, and scientific computing as well as fusion energy.

When first founded, the Center relied on CDC 7600 machines—the fastest and most powerful at the time. In 1978, NERSC acquired one of the first CRAYs built and, since then, has expanded its collection of supercomputers with a series of even more powerful CRAYs. Today, NERSC houses a

A roomful of computing power. NERSC currently houses a CRAY Y-MP C90; two CRAY-2s, a 4-processor and an 8-processor, each with 128 million words of memory; and a CRAY X-MP dedicated to the high-school education program.

CRAY Y-MP C90, a 16-processor, 256-million-word machine that is the fastest CRAY computer currently available; two CRAY-2s, a 4-processor and an 8-processor, both with 128 million words of memory; and a CRAY X-MP dedicated to the high-school education program.

A successful general-purpose computing environment depends not only on the capabilities of the central supercomputers but also on factors such as network speed and storage capacity. When NERSC was first formed, access to remote users was provided by telephone lines. This was then upgraded to satellite communication. The current network, called the Energy Sciences Network, is

based on fiber optic technology and offers transmission at rates of over one million bits of data per second. NERSC's archival storage system is one of the world's largest and fastest. Its present capacity is over twelve trillion bytes, to be at least doubled in the next year or two.

Over the years, NERSC has provided an invaluable service to the energy research community. NERSC serves some 4500 users located at national laboratories, universities, and in private industry. With increasing concern about energy-related issues, such as global warming and alternative energy sources, the demand for NERSC's services will undoubtedly grow.

The CRAY Y-MP C90, NERSC's latest supercomputer, is a 16-processor, 256-million-word machine, the fastest CRAY computer currently available. The C90's central processor provides a peak performance of one billion floating-point operations per second (one GFLOPS). With a peak performance of 16 GFLOPS, the C90 enables a higher level of productivity for scientists and engineers.

Bill McCurdy Talks About the Superkids

The “Superkids” program gives our best and brightest students an experience in science education that’s unlike any other that I know of. If we want to teach them modern astronomy, for example, we can’t give them viewing time on the Keck telescope. If they’re interested in high-energy physics, we can’t get them beam time on Stanford’s SLAC accelerator. If we take them on a tour of the Lab’s Nova laser facility, they’re told not to touch anything. But with supercomputers it’s different. Here they can get hands-on experience, using the same kinds of machines that scientists around the country use for leading-edge research in fusion energy, in global climate modeling, in chemistry and materials science, and so on.

They sense that we’re not making this up for them: it’s real, and they’re doing the same kinds of things—often using the same computer codes—being done in forefront research. And their experiences with real science aren’t limited to dialogues with computers. In high-school science, students have little contact with anyone doing forefront research. High-school students have trouble seeing scientific research in terms of real human beings with their personal interests and feelings about their work. So we bring in prominent scientists, such as Edward Teller, for brown-bag sessions with the kids that are sometimes presentations, sometimes discussions, and sometimes both. Both, but always interesting and provocative. This kind of interaction helps to make science less of an abstraction and more of a human activity for them.

The most important part of the Superkids’ experience is not the computers they use, but the science they do. They run global climate simulations with computer codes that are being used right now by scientists trying to understand climate change. They run seismic analysis codes to see how

buildings, freeways, and other structures behave during an earthquake. Another program does quantum chemistry. It’s the same code used by drug researchers to model interactions between molecules and thereby to screen a vast number of candidate drugs before performing expensive experiments.

More important, they invent applications of those codes. For example, they model what happens to global climate a hundred or a thousand years from now if the earth’s orbit is altered by an errant astronomical body that enters the solar system. This is the sort of ability to dream—to *imagine* science—that will someday make them leaders in science. And I think it’s what makes the program work.

The Superkids program has produced a cascade effect in educational innovation at the Laboratory. We now sponsor a very similar program for high-school teachers, who respond very much like the Superkids. It improves their image of themselves as teachers, and it gives them teaching tools they

wouldn’t otherwise have. It provides encouragement and recognition in the difficult job of tracking students to science and mathematics.

One of the nation’s educational goals is to make our students first in the world in math and science by the year 2000. One component of our strategy for reaching this goal is to provide better science and math education for more people, and the other is to educate leaders. The students who come to the Lab are between 14 and 18 years old. They’re chosen from 50 states, four territories, and several foreign countries. In general, they’re picked on the basis of accomplishment. What they learn from their role models here forms impressions that will last their entire careers. These are some of the people who will become our future scientific leaders.

END

DATE

FILMED

3/10/94