A SCIENCE-BASED CASE FOR LARGE-SCALE SIMULATION

Volume 2

Office of Science
U.S. Department of Energy

September 19, 2004

"There will be opened a gateway and a road to a large and excellent science, into which minds more piercing than mine shall penetrate to recesses still deeper." Galileo (1564-1642)

[*on the "experimental mathematical analysis of nature," appropriated here for "computational simulation"*]

TRANSMITTAL

September 19, 2004

Dr. Michael Strayer, Acting Director
Mathematical, Information, and Computational Sciences Division
Office of Science
U.S. Department of Energy
Washington, DC

Dear Dr. Strayer,

This volume follows Volume I of *A Science-based Case for Large-scale Simulation*, which was delivered to the Director of the Office of Science of the U.S. Department of Energy on July 30, 2003. Both volumes are products of a workshop held June 23 and 24, 2003 in Washington, DC, during which over two dozen working groups, each composed of an appointed blend of scientists and engineers, mathematicians, and computer scientists, met and produced outlines that were subsequently fleshed out into the chapters of this volume: 11 chapters headlined by scientific missions of the Department, and 16 chapters focusing on enabling technologies from mathematics and computer science.

Volume I, whose 70 pages were produced expeditiously following the workshop, is intended for broad readership, and does not contain very much scientific detail. It abstracted the case for fostering simulation-based science and a new science of simulation from the products of the workshop (slide presentations of the working groups and early chapter drafts from their leaders). Its first three chapters provide motivation, historical perspective, and an illustration of large-scale simulation. The technical results of the workshop were condensed to little beyond lists of scientific opportunities in Chapter 4 and brief descriptions of supporting strides in mathematics and computer science in Chapter 5. These were followed by a summary chapter containing eight general recommendations.

Volume II, which you have before you, allows significantly more room for presentation of each of the technical frontiers digested in Chapters 4 and 5 of Volume I. In particular, each of the eleven chapters of this volume describing scientific applications is allotted a dozen pages. They introduce their subject, describe its impact on science and society, name scientific opportunities brought near by the rapid growth expected in simulation capability, and delineate research issues critical to simulation progress. These chapters also point to the enabling technologies chapters that follow by describing technology barriers. They list resources required to address the barriers (rooted, where convenient, in a discussion of resources currently used) and provide metrics of success for simulation in the field. Six of the chapters contain as “sidebars” illustrative short accounts that call attention to a particular scientific challenge with detail that might be out of proportion in

one of the standardized sections. (Each science application working group was given the option of including sidebars. Some chapters integrate their illustrations.)

Each of the eight chapters on mathematical methods introduces its subject, conveys important impacts on scientific applications, characterizes its research frontier, and specifies metrics of success. The eight computer science chapters follow the same outline as the mathematics chapters. Being software and systems oriented, they also in most cases list external dependencies and discuss how best to deliver their results to the applications.

Each of the three major sections of Volume II – scientific applications, mathematical methods, and computer science tools – is preceded by an introduction that comments on common elements of its chapters and makes a summary of the state of the art, *vis-à-vis* high-end simulation. These introductions draw, respectively, upon Chapters 4 and 5 of Volume I.

Volume II reproduces the Executive Summary of Volume I, which is updated only by the marking of another successful year of the Department's Scientific Discovery through Advanced Computing (SciDAC) initiative.

Taken together, the two volumes of *A Science-based Case for Large-scale Simulation* should be regarded as a meta-report authored by the simulation community at large. This meta-report will require continual refreshing of content, due to the rapid change in every one of its constituent fields. Several subjects of chapters in Volume II have benefited from dedicated workshops since the June 2003 combined workshop, as was intended in the charter of the original, and others are planned beyond this publication date. However, the thrust of the report remains reassuringly constant: simulation is an increasingly capable partner of theory and experiment in scientific progress, and its current limitations, most of which are well understood, are amenable to being pushed further and further outward.

In the 14 months that Volume I has been in circulation, a criticism has been levied against the basic charter that led to its creation, to wit, that advances in simulation capabilities were presumed, and the scientific community asked to respond. Could a more convincing science-based case for large-scale simulation be made if, instead, the scientific community first announced its needs, and those that advance the technologies of simulation were asked to respond? The editors of the present installment of our community's meta-report believe that the content assembled here is robust with respect to either phrasing of the question of scientific opportunity. It is granted that there are many quests in science and engineering that still defy simulation. Those documented in these volumes are, however, ripe for conquest by a balanced program that includes a prominent and increased role for simulation, and each makes its own case for importance. We cautioned in Volume I (page 7) and reiterate here that prioritizing among exciting scientific opportunities is beyond our charter and scope. We are gratified, however, by all such debate, and aim hereby to further equip the debaters.

We observe that this report is just one of many recent attempts to contribute to an enlightened assessment of simulation as a means of progress in science and engineering.

The CSTB project on *The Future of Supercomputing*, the report of the interagency *High-end Computing Revitalization Task Force*, and the NSF report *Revolutionizing Science and Engineering through Cyberinfrastructure* are three of the major community-wide studies that partially overlap the present effort in period of execution, in contributing personnel, and in scientific scope. In the workshop on *A Science-based Case for Large-scale Simulation* and in Volume I, we acknowledged our debt to many earlier such studies (going back to the National Science Board study convened by Peter Lax in 1982) and we compared our conclusions with some of them. The chapters of this volume are not exhaustive with respect to the opportunities for advances through simulation in their individual fields (e.g., neuroscience is beyond the purview of our biology chapter), and there are many fields in which simulation is highly prominent that are not represented here by a chapter (e.g., aerodynamics). Our choice of subject matter has strong alignment with ongoing programs in the Department of Energy's Office of Science.

Eight pages of front matter in Volume I were dedicated to listing the names and affiliations of the 315 scientists from government, academia, and industry who contributed to its contents, and thus directly as well to the contents of Volume II. In the interest of space, we do not repeat all of these attributions here. We intend to produce, in the near term, a book combining the material of Volumes I and II, which will add extensive references and some additional material. All contributions will be acknowledged afresh in that archival document. Chapter authors of Volume II include: Robert Armstrong, David Bailey, John Bell, E. Wes Bethel, David Brown, Phillip Colella, Michael Colvin, Peter Cummings, Lori Freitag Diachin, John Drake, Paul Fischer, Ian Foster, Al Geist, James Glimm, Frank Graziani, William Gropp, Francois Gygi, Steven Hammond, Charles Hanson, Robert Harrison, Bruce Hendrickson, Van Henson, Thomas Hughes, Stephen Jardin, William Johnston, Phillip Jones, Sallie Keller-McNulty, David Keyes, Dana Knoll, Kwok Ko, Ewing Lusk, Robert Malone, Juan Meza, Anthony Mezzacappa, George Michaels, William Nevins, Gordon Olson, Alex Pothen, Larry Rahn, Robert Rosner, Doron Rotem, Robert Ryne, David Serafini, John Shadid, Mark Shephard, Arie Shoshani, G. Malcolm Stocks, Robert Sugar, Lin-wang Wang, Mary Wheeler, Theresa Windus, Steve Yabusaki, and Kathy Yelick.

A handful of individuals must be further recognized. Without them, Volume II might have languished unassembled for some time further. Edward H. Barsis and Peter L. Mattern recently undertook an assessment of simulation drivers in the Department of Energy in a study sponsored by Sandia National Laboratories. They were provided access to early drafts of the scientific applications chapters of this volume and, through consultation with some of the above-listed authors of these chapters, they in turn assisted us in updating them. The same team of section editors that co-produced Volume I remained faithful to the project throughout its sixteen months: Thom H. Dunning, Jr. of Oak Ridge National Laboratory, who edited the chapters on scientific applications, Phillip Colella of Lawrence Berkeley National Laboratory, who did the same for mathematical methods, and William D. Gropp of Argonne National Laboratory, for computer science. The breadth and depth of their knowledge brought as much unity and coherence to each respective section as the diverse subject matter spanning each allowed. Finally, the editors gratefully acknowledge the labor beyond duty's call of technical

editor Gail Pieper of Argonne National Laboratory, who substantially cleaned up the presentation.

We recommend reading the two volumes of *A Science-based Case for Large-scale Simulation* together and we look forward to future installments of this and related meta-reports. By the appearance of the next installment, we expect that successes due to simulation will pile up in many areas discussed herein and certain exponents characterizing the state of the art will change; however, large-scale simulation will be at least as important to the next wave of grand scientific and engineering challenges as it is to today's.

Best regards,

David E. Keyes
Fu Foundation Professor of Applied Mathematics
Columbia University
New York, NY

Executive Summary

Important advances in basic science crucial to the national well-being have been brought near by a "perfect fusion" of sustained advances in scientific models, mathematical algorithms, computer architecture, and scientific software engineering. Computational simulation – a means of scientific discovery that employs a computer system to simulate a physical system according to laws derived from theory and experiment – has attained peer status with theory and experiment in many areas of science. The United States is currently a world leader in computational simulation, a position that confers both an opportunity and a responsibility to mount a vigorous campaign of research that brings the advancing power of simulation to many scientific frontiers.

Computational simulation offers to enhance, as well as leapfrog, theoretical and experimental progress in many areas of science critical to the scientific mission of the U.S. Department of Energy (DOE). Successes have been documented in such areas as advanced energy systems (e.g., fuel cells, fusion), biotechnology (e.g., genomics, cellular dynamics), nanotechnology (e.g., sensors, storage devices), and environmental modeling (e.g., climate prediction, pollution remediation). Computational simulation also offers the best near-term hope for progress in answering a number of scientific questions in such areas as the fundamental structure of matter, the production of heavy elements in supernovae, and the functions of enzymes.

The ingredients required for success in advancing scientific discovery are insights, models, and applications from scientists; theory, methods and algorithms from mathematicians; and software and hardware infrastructure from computer scientists. Only major new investment in these activities across the board, in the program areas of DOE's Office of Science and other agencies, will enable the United States to be the first to realize the promise of the scientific advances to be wrought by computational simulation.

In this two-volume report, prepared with direct input from more than 300 of the nation's leading computational scientists, a science-based case is presented for major, new, carefully balanced investments in

- scientific applications
- algorithm research and development
- computing system software infrastructure
- network infrastructure for access and resource sharing
- computational facilities
- innovative computer architecture research, for the facilities of the future
- proactive recruitment and training of a new generation of multi-disciplinary computational scientists

The three-year-old Scientific Discovery through Advanced Computing (SciDAC) initiative in the Office of Science provides a template for such science-directed, multidisciplinary research campaigns. SciDAC's successes in the first four of these seven thrusts have illustrated the advances possible with coordinated investments. It is now time to take full advantage of the revolution in computational science with new investments that address the most challenging scientific problems faced by DOE.

A SCIENCE-BASED CASE FOR LARGE-SCALE SIMULATION

Volume 2

Scientific Applications at Large Scales

Accelerating Computations for Accelerators
Astronomy and Astrophysics: Interpreting the Universe
Computing Life: From Molecules to Communities
Better Living through Computational Chemistry
Computing the Climate
Combustion Science: Enabling our Energy Future
Design of Materials: the Road to Technological Innovation
Nanoscience: Building a Better World one Atom at a Time
Plasma Science: Taming a Star
Simulating Quarks and Gluons with Quantum Chromodynamics
Subsurface Transport and Fate: Environmental Stewardship and Health

Mathematical Tools for Large-scale Simulation

Computational Fluid Dynamics
Discrete Mathematics
Meshing Methods
Multi-physics Solution Techniques
Multiscale Simulation
Solvers and Fast Algorithms
Transport Methods
Uncertainty Quantification

Computer Science for Large-scale Simulation

Visual Data Exploration and Analysis
Computer Architecture for High-end Computing
Programming Models and Component Technology for High-end Computing
Access and Resource Sharing
Software Engineering and Management
Data Management and Analysis: Keeping Ahead of the Data Avalanche
Performance Science
System Software and Large-scale Simulation

Table of Contents

Executive Summary 6
Table of Contents 8
Scientific Applications Overview 13
Role of Computational Modeling and Simulation 14
Computer Hardware 16
Scientific Applications 17
Data Deluge 18
Accelerating Computations for Accelerators 19
Impact on Science and Society 20
Scientific Opportunities 21
Beam–Beam Beam Heating Effects to Improve Storage Ring Collider Performance 21
Dark Current & Wakefields to Optimize Linear Collider Structure Design 22
Coherent Synchrotron Radiation to Improve Fourth-Generation Light Sources 24
Developing the Breakthrough Technologies for Far-Horizon Machines 24
Research Issues 25
Technology Barriers 26
Resources Required 27
Electromagnetic Systems Simulations 28
Advanced Accelerators 28
Metrics of Success 29
Astronomy and Astrophysics: Interpreting the Universe 30
Scientific Opportunities 31
Research Issues 36
Technology Barriers 39
Resources Required 40
Metrics of Success 42
Computing Life: From Molecules to Communities 43
Impact on Science and Society 43
Scientific Opportunities 43
Research Issues 44
Bioinformatics 44
Biophysical and Biochemical Modeling and Simulation 45
Systems Modeling and Simulation 45
Technology Barriers 46
Bioinformatics 46
Biophysical and Biochemical Modeling 47
Systems Modeling 48
Resources Required 49
Database Infrastructure for Bioinformatics 49
Computing Resources for Biophysical and Biochemical Simulations 49
Information Management Systems for Systems Biology 50
Metrics of Success 50

Sidebar #1 51
From Fold Recognition to Ab Initio Structure Prediction 51
Sidebar #2 53
Simulations of P. aeruginosa LPS Membrane Goethite Mineral Interactions 53
Sidebar #3 54
Modeling the Ribosome 54
Better Living through Computational Chemistry 55
Impact on Science and Society 55
Scientific Opportunities 56
Research Issues 58
Technology Barriers 59
Resources Required 61
Metrics of Success 63
Heavy element and environmental chemistry – a special responsibility of DOE 64
What is required to develop the next generation of computational chemistry software? 65
Computing the Climate 67
Impact on Science and Society 68
Scientific Opportunities 69
Research Issues 70
Technology Barriers 71
Resources Required 73
Metrics of Success 76
Why is increased resolution important? 77
Impact on Science and Society 78
Scientific Opportunities 78
Research Issues 82
Technology Barriers 83
Resources Required 84
Metrics of Success 86
Sidebar #1 87
Autoignition and Control of ‘Flameless’ Combustion 87
Sidebar #2 88
How Do Soot Particles Develop and Grow? 88
Design of Materials: the Road to Technological Innovation 90
Impact on science and society 91
Scientific Opportunities 92
Magnets of the Future: Predictive Modeling of Switching and Hysteresis 94
Advancing Modeling of the Fundamental Interactions in Complex Systems 95
Research Issues 96
Technology Barriers 97
Hardware Barriers 97
Software Barriers 97
Algorithm Barriers 99
Resources Required 99

Computational Resources 99
Human Resources 100
Nanoscience: Building a Better World one Atom at a Time 102
Impact on Science and Society 102
Scientific Opportunities 103
Research Issues 107
Technology Barriers 108
Resources Required 109
Metrics of Success 111
Plasma Science: Taming a Star 112
Impact on Science and Society 112
Scientific Opportunities 114
Research Issues 116
Technology Barriers 119
Resources Required 119
Metrics of Success 122
Simulating Quarks and Gluons with Quantum Chromodynamics 124
Impact on Science 124
Scientific Opportunities 127
Research Issues 129
Technology Barriers 130
Resources Required 131
Metrics of Success 133
Impact on Science and Society 135
Scientific Opportunities 136
Research Issues 138
Technology Barriers 139
Resources Required 140
Metrics of Success 141
Sidebar #1 142
Sidebar #2 144
Migration of High-level Radioactive Waste from Leaking Subsurface Tanks 144
Sidebar #3 146
Reduction of Global Warming through Geologic Sequestration of Carbon Dioxide 146
Mathematical Tools for Large-Scale Simulation 148
Computational Fluid Dynamics 151
Impact on Applications 151
Research Issues 154
Metrics for Success 155
Discrete Mathematics 157
Impact on Applications 157
Research Issues 158
Metrics of Success 160
Meshing Methods 161

Impact on Applications 161
Research Issues 163
Metrics of Success 164
Impact on Applications 166
Metrics for Success 168
Impact on Applications 169
Research Issues 173
Metrics for Success 174
Solvers and Fast Algorithms 176
Impact on Applications 177
Research Issues 179
Metrics of Success 181
Transport Methods 184
Impact on Applications 184
Research Issues 186
Metrics for Success 186
Uncertainty Quantification 188
Impact on Applications 189
Research Issues 190
Metrics for Success 192
Computer Science for Large-scale Simulation 193
Visual Data Exploration and Analysis 197
Impact on Applications 197
Research Issues 197
High-Capacity Visualization 198
Remote Visualization 199
Multiresolution Methods 200
Multidimensional and Multivariate Visualization 201
Coupling Analysis, Visualization, and Data Management 202
“Behavioral” Visualization 202
Delivering Visualization Technology to Application Scientists 203
Resources Required 203
Metrics of Success 205
Computer Architecture for High-end Computing 207
Impact on Applications 207
Research Issues 207
External Dependencies 209
Resources Required 210
Metrics of Success 211
Programming Models and Component Technology for High-End Computing 213
Component frameworks and programming models 213
Research Issues in HEC Programming Models 214
Programming Model Overview 215
Barriers to HEC Programming Models 215
Research Issues in HEC Component Frameworks 217

Barriers to HEC Components 217
Strategy for Overcoming Barriers 218
Metrics of Success 220
Access and Resource Sharing 221
Impact on Applications 221
Remote Access 222
Data Integration 222
Coupling Simulation and Experiment 222
Multidisciplinary Simulation 223
Orchestration of Science Process 224
Collaboration 224
Research Issues 225
External Dependencies 226
Physical and Services Infrastructure 226
Policy Challenges 227
Resource Requirements 227
Metrics of Success 227
Software Engineering and Management 229
What Is Software Engineering and Management? 229
Research Issues 230
External Dependencies 231
Delivering Software Engineering and Management to Applications 231
Resources Required 232
Metrics of Success 232
Data Management and Analysis: Keeping Ahead of the Data Avalanche 234
Research Issues 236
External Dependencies 239
Delivering Data Management Technology to Applications 240
Resources Required 242
Metrics of Success 242
Performance Science 244
Impact on Applications 245
Research Issues 246
External Dependencies 247
Delivering Performance Science to Applications 247
Resources Required 248
Metrics of Success 249
System Software and Large-scale Simulation 250
Impact on Applications 250
Research Issues 251
Resources Required 252
External Dependencies 253
Metrics of Success 253

Scientific Applications Overview

Computational modeling and simulation are among the most significant developments in the practice of scientific inquiry in the 20th Century. In the past two centuries, scientists have had extraordinary successes in identifying the fundamental physical laws that govern our material world, *e.g.*, the Standard Theory of elementary particles, the Schrödinger equation for atoms and molecules, the Navier-Stokes equations for fluid flow, and Maxwell's equations for electromagnetic fields. The solutions to these equations proffer an unprecedented level of understanding of the basic structure, interactions, and dynamics of matter. They also make it possible to obtain technologically and sociologically invaluable data on the impact of increases of greenhouse gases in the earth's atmosphere, the transport of underground pollutants, and the chemical processes involved in combustion, radionuclide separations, or enzymatic reactions, to name just a few. The difficulty is that these equations cannot be solved exactly for any but the simplest systems.

Computational modeling and simulation provide a means of solving the mathematical equations describing the physical laws of nature and predicting the behavior of natural and engineered systems. Computational modeling and simulation have been steadily advancing since the development of electronic computers in the Second World War. Extraordinary advances in computing technologies in the past decade have set the stage for a major further advance in computational modeling and simulation. In the 1990s, the peak speed of individual processors increased by a factor of one hundred, following the trajectory predicted by Gordon Moore. Over the same decade, the peak speed delivered to applications, as measured by Gordon Bell prizes, increased by nearly four orders of magnitude, thanks to parallelization. In the first decade of the 21*st* Century, an additional increase of a factor of one hundred is expected from Moore's Law, and while it is more difficult to predict how successful scientists will be in harnessing increased parallelism, the prospects are good for many important algorithms. These advances herald a new era in which computation contributes to scientific discovery with an impact comparable to that from experiment and theory. Combining these three scientific modalities, it will be possible to dramatically extend the exploration of the fundamental processes of nature, *e.g.*, the structure of matter from elementary particles to the building blocks of life, as well as advance the ability to predict the behavior of a broad range of complex natural and engineered systems, *e.g.*, nanoscale devices, microbial cells, fusion energy reactors, and the earth's climate.

To exploit this opportunity, the advances in computing technology must be translated into increases in the fidelity and performance of the scientific and engineering applications used to model the physical, chemical, and biological processes that underlie complex natural and engineered systems. This is a major undertaking. It will require advances in theoretical and mathematical science leading to computational models with increased fidelity and utility. It will require close collaboration among computational scientists, computer scientists, and applied mathematicians to translate these advanced mathematical

models into scientific application codes that can realize the full potential of today's and tomorrow's high-end computers. It will require educating a new generation of scientists and engineers who can join in the collaborative development of computational modeling and simulation tools and lead their application to address the nation's most pressing technical challenges.

In the eleven chapters in this section, experts in the various fields of computational modeling and simulation important to the mission of the Office of Science in the U.S. Department of Energy describe the opportunities for advancement in their fields and discuss the challenges that must be overcome if these opportunities are to be realized. The authors of the chapters discuss the impact that their field of science has on both science and society, describe the opportunities in science that could result from advances in computational science, and delineate the research issues in theoretical and computational science, computer science, and mathematics that must be addressed to achieve these advances. They also provide estimates of the resources that will be required to address these issues and apply the new capabilities to address the most important scientific challenges. Finally, in the last section of each chapter, the authors grapple with how to define a means of measuring success. This is an important issue, but it is not straightforward and many different views are offered. Clearly, in the end, the true measure of success is the advances in science enabled by advances in computational modeling and simulation capabilities. But, this is difficult to quantify except in historical hindsight.

While there is diversity in the themes within the chapters, there is also a measure of commonality. Below, some of the major themes that emerged in the chapters on the scientific applications are summarized.

Role of Computational Modeling and Simulation

In almost all of the scientific areas discussed here, computational science and engineering is already playing a critical role in advancing science.

Computational accelerator physicists are using simulation to increase the luminosity and the operating efficiency of existing particle accelerators and to optimize the design of next-generation accelerators to reduce both cost and risk. They are also using simulation to explore a new generation of more powerful, yet more compact and economical particle accelerators as a goal that is within reach in the next decade.

Computational biologists have long used simulation to gain insights into the structure of proteins and the mechanisms of enzymatic reactions. Now their challenge is to extend a new generation of powerful *ab initio* methods to such macromolecules, implement these methods on the most powerful computers available, and use them to make predictions about protein interactions and the structure and reactivity of protein complexes.

Computational combustion scientists are using simulation to describe complex interaction of fluid flow processes with the myriad chemical processes involved in sustaining the flame, producing both power and pollutants. Exciting opportunities include the nascent ability to combine direct numerical simulations of fluid flow with detailed chemical mechanisms on scales that directly compare to, and extend, advanced experimental

studies. This will provide an unprecedented view into the details of phenomena critical to many industrial chemical processes as well as combustion.

Computational environmental scientists are not only using simulation to better understand the impact of increases in greenhouse concentrations on global climate, but have recently begun to focus on the impact of such changes on regional climates – the place where all of us live. Such information is critical if we are to develop realistic responses to addressing the problems associated with climate change.

Computational environmental scientists are also involved in developing efficient, reliable approaches for cleaning up polluted soil and groundwater by using simulation to describe the complex interaction of physical, chemical, and biological processes in the subsurface. The powerful combination of advanced simulation technologies and a new generation of tools for characterizing subsurface properties and processes provide an unprecedented opportunity to advance our understanding of subsurface phenomena.

Computational high energy physicists are using simulation to test the Standard Model of high energy and nuclear physics, which is also the object of massive experimental campaigns at facilities all over the world. Our knowledge of this model is incomplete because it has been difficult to extract many of its most interesting predictions. The only existing method for doing so from first principles and with controlled systematic errors is through large-scale numerical simulations within the framework of lattice gauge theory.

Computational chemists have developed capabilities for predicting the structure, energetics and dynamics of small molecules, sometimes literally replacing experiment by simulation. They have now set their sights on extending these capabilities to much larger molecular systems, especially those important in catalysis, combustion, nanoscale science, and biochemistry, with very promising results.

Computational materials scientists are using simulation to better understand a broad range of materials, including materials important to energy, transportation, and computing. With continuing advances in computational techniques and increased computing capability, it will soon be possible to put in place a new paradigm for materials research in which modeling and simulation are integrated with synthesis and characterization to accelerate the discovery of new materials.

Computational nuclear physicists have entered the age of “precision cosmology.” Virtually all phases of the Universe – from its earliest moments to the present – are amenable to modeling and simulation, which allows us to connect what happened in the distant past to what is observed now. Areas actively being pursued include: large-scale structure of the universe; the formation and interaction of galaxies; star formation and stellar evolution and death; and numerical relativity.

Computational plasma physicists are actively engaged in merging the now separate macroscopic and microscopic models of plasmas and in extending the fidelity of these models by the inclusion of detailed simulations of the processes embodied in them. This integrated, high fidelity modeling capability will allow plasma scientists to develop understanding and insights into the complex fusion systems that will be critical in

realizing the long term goal of creating an environmentally and economically sustainable source of energy

Computer Hardware

Much of the progress in computer technology is being driven by increases in the computing speed of microprocessors. This had led to an increasing gap between the speed of the processors and that of the memory subsystem – unless the memory subsystem can keep the processor busy, valuable computing cycles are wasted. Although a number of scientific applications have been able to take good advantage of today's cache-based microprocessor systems, many others are experiencing lagging performance gains that can be directly linked to slow memory access. We find many of the chapter authors noting the need for *balanced* supercomputer systems, where balance is defined in terms of the ratio of processor speed versus memory speed, or processor speed versus the speed of the interprocessor switch, *etc*. The Japanese Earth Simulator is proof that it is possible to design much better balanced machines than those available in the U.S. today and demonstrates that the resulting performance gains can be enormous.

The computational scientists involved in developing applications for state-of-the-art computers need to be intimately connected with the computer scientists and engineers involved in designing the next generation of computers to be used for scientific research. The current trend in the design of parallel supercomputers involves the use of more, often less powerful processors (perhaps as many as 100,000 processors). There are very few scientific applications that currently scale to more than 10,000 processors and the techniques that would be used to allow most applications to run efficiently on 100,000 processors are not known. Close collaborations between computational scientists and computer designers will lead to a far better understanding of the limitations of computer designs for scientific and engineering simulations and could even lead to innovative solutions to this problem.

The chapter authors also noted a clear need for a substantial increase in computing resources *across the board* for computational modeling and simulation. *Capacity* as well as *capability* computing resources are important. The majority of current research takes place on the computer systems available to small research groups (*capacity computing*). These computing resources are critical to the development of new methods and techniques as well as for exploratory calculations on scientific problems of interest. As computer technology evolves, these systems very cost-effectively extend and enhance the scope of activities of many research groups. To allow solution of the most challenging scientific problems, however, these resources must be augmented by access to the highest end computing resources available (*capability computing*). In the U.S. these resources are largely provided by DOE's Office of Science and NSF. However, none of these facilities provide the computing capability offered by machines such as the Earth Simulator, nor are the resources available sufficient to satisfy more than a fraction of the needs of the computational science and engineering community.

Scientific Applications

In planning scientific computing initiatives, it is very easy to focus on the computer hardware, relegating the computer and computational software to the background. This is a mistake – scientific applications are the "engines of discovery," computer hardware only enables these discoveries. As stressed by the President's Information Technology Advisory Committee (PITAC) in its 1999 report:

> "Software is the new physical infrastructure of the information age. It is fundamental to economic success, scientific and technical research, and national security. ... The Committee recommends that the Government make fundamental research in software both for computer systems engineering and for applications one of the Nation's highest R&D priorities."

The chapter authors agree with this statement, noting the importance of increased funding for both the development of new theoretical and mathematical models as well the implementation of new and existing models on computers. For high-end computers, the latter efforts require teams of computational scientists, computer scientists, and applied mathematicians. SciDAC has shown the efficacy of this team approach, but its investments must be expanded several fold if these benefits are to be fully realized in all fields of science and engineering important to the U.S. Department of Energy.

As the problems being addressed by computational simulations become more complex – *more like the real world* – new mathematical techniques and algorithms are needed to handle the broad range of temporal and spatial scales involved. As many as fifteen orders of magnitude present themselves in some fields. As many as ten orders of magnitude have been resolved in special simulations (with attention to floating point precision beyond what is provided as standard computational datatypes). More typically, at most two or three decades of scale are resolved today. The need for development and evaluation of new mathematical techniques and algorithms is a theme in most of the chapters. If the next generation of computer systems is to be built using hundreds of thousands of processors, then we must begin laying the groundwork for this now by increasing investments in numerical methods and algorithms. If we do not, these machines will provide increased capacity computing but little additional capability computing.

In many fields the software developed by computational scientists is used by a large number of researchers in the field as an aid to their computational and/or experimental studies. This offers greatly enhanced opportunities for advancing science but poses special problems for the scientific software developer. Scientific applications that are used by non-experts must be easy to use, robust, and reliable. This is best achieved by building a problem-solving environment for the scientific applications. Such an environment, which involves both software and hardware to allow visual interaction (and perhaps haptic, auditory, and other modes of interaction) with the data of the evolving

and of the completed computation is analogous to the "endstation" of an experimental device such an accelerator. Building an endstation requires close collaboration between scientists who are familiar with the language, practices, and computational models of the given field of science and computer scientists who are familiar with the many technologies requires to build such environments. This problem has received little attention in the past, when most of the users were experts; it must now be considered an integral piece of the software development process.

Data Deluge

Computational simulations are beginning to produce a flood of data – terabyte-size individual data sets and petabyte-size overall data sets – the magnitude of which will only increase with time (similar phenomena are occurring in certain areas of experimental and observational science, as well). Areas at the forefront of this data explosion include biology, climate and subsurface science, and combustion, although most areas will be in a similar situation in the next five to ten years. The data sets produced in the simulations have, themselves, become objects of research, with scientists mining the data sets in search of new discoveries. This requires the development of an infrastructure for storing, managing, and accessing the data sets as well as software tools to mine, analyze, and visualize the data in these sets.

Accelerating Computations for Accelerators

For more than 60 years, particle accelerators have been enablers of scientific discovery and technological progress. Accelerators give high energy to subatomic particles, which then collide with targets or with other particle beams or are used to produce secondary beams and/or radiation. They have led to important discoveries in fields such as high-energy physics, nuclear physics, materials science, chemistry, and the biosciences. Furthermore, accelerator technology is found in the service of fields far from basic research in the physical sciences: medical isotope production, medical irradiation therapy, pharmacology (e.g., rational drug design using protein crystallography at light sources), geologic exploration (e.g., placing a portable neutron generator inside a bore hole), industrial processes such as ion implantation, and even art authentication—the Louvre has its own particle accelerator.

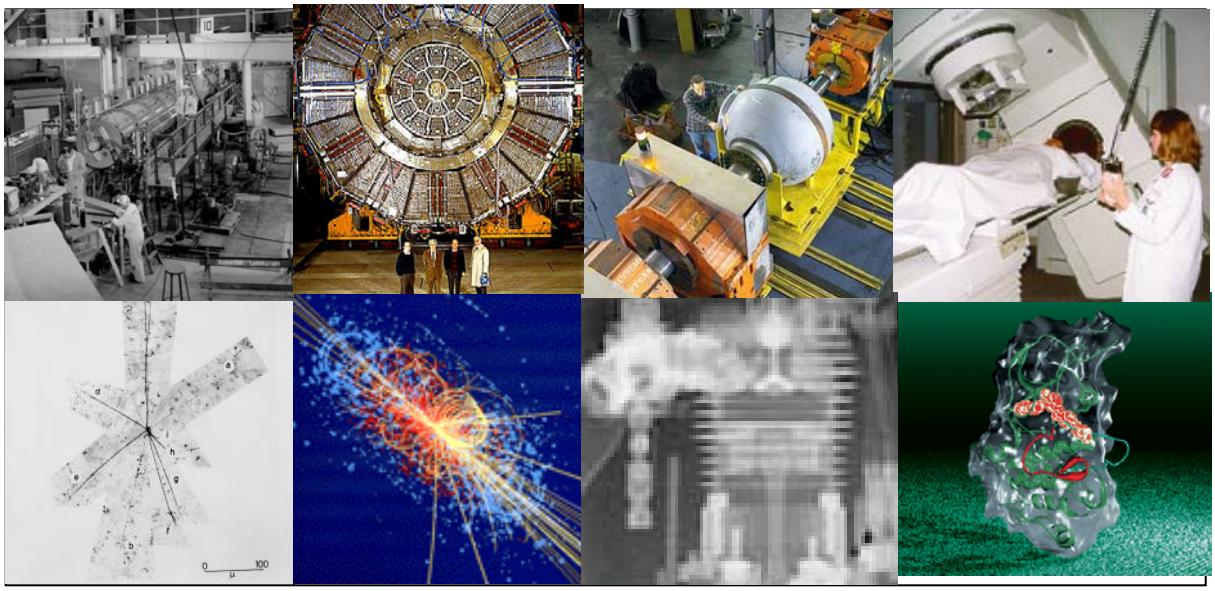


Figure 1. Particle accelerators have been responsible for a number of important Nobel prize-winning discoveries, ranging from the antiproton in 1959 (upper/lower left), to the W and Z particles in 1984 (upper/lower second from left). They are also used in national security applications such as x-ray and proton radiography (upper, second from right), an area that has important industrial spin-offs (lower, second from right). Particle accelerators also have many health-related applications, including medical therapy (upper right), and x-ray crystallography to determine the structure of proteins and other biological systems (lower right).

Particle accelerators come in a variety of shapes and sizes, from tiny accelerators in CRT-based televisions and computer monitors to a scale of many kilometers. The largest facilities are found in the nation's national laboratories and universities, and they are

supported by a host of programs within the U.S. government. Prominent among them is been DOE's Office of Science (SC), which has been responsible for developing some of the world's most powerful accelerators. The DOE/SC's existing portfolio of accelerators includes high-energy colliders (PEP-II, Tevatron, RHIC, and CEBAF), synchrotron light sources (SSRL, ALS, APS, NSLS), and spallation neutron sources (IPNS, SNS), on which the national investment totals more than $1 B in construction cost alone (see Fig. 2). This sum could grow to over $10 B within the decade should all accelerators, planned and proposed, be approved and built.

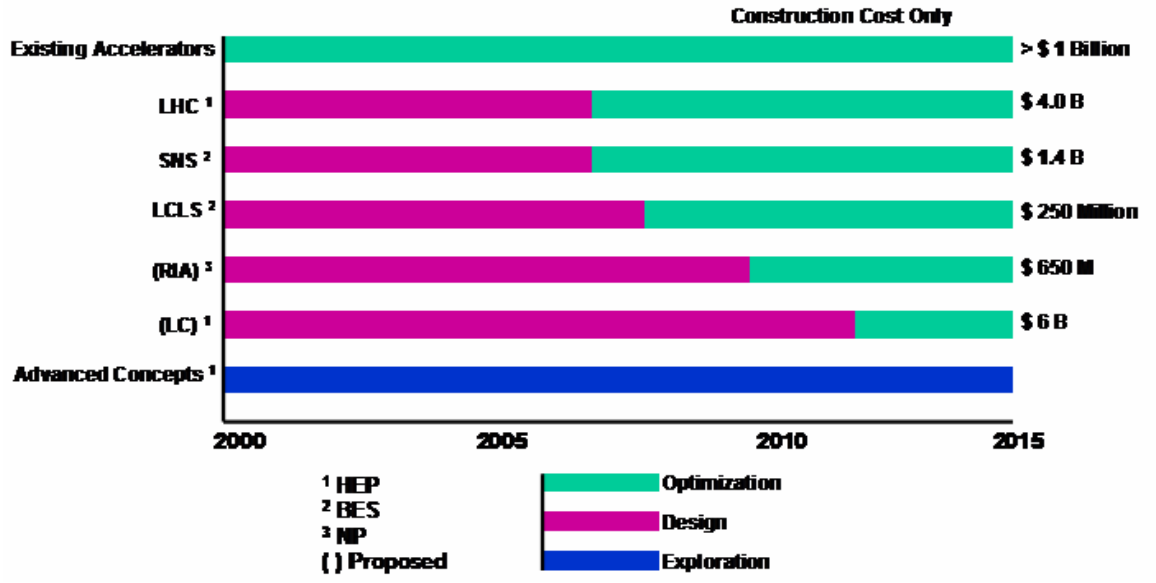


Figure 2. Timeline for DOE/SC's portfolio of particle accelerators that includes existing machines (PEP-II, Tevatron, RHIC, CEBAF), those under construction (SNS, LHC, LCLS), and proposed facilities (LC, RIA). Simulation can benefit the design phase and help with optimizing performance during operation. It is also the main tool for exploring novel acceleration concepts.

Impact on Science and Society

Particle accelerators are enablers of remarkable scientific discoveries, especially in the field of high energy and nuclear physics. Experiments associated with high-energy accelerators have led to important discoveries about elementary particles and the fundamental forces of nature, quark dynamics, and nuclear structure. A half-dozen Nobel prizes have been awarded to physics efforts based on particle accelerators, and the quest into the nature of matter continues with existing and planned facilities.

Beyond impacts on basic and applied science, accelerators are playing an important role in addressing the nation's energy, national security, and environmental issues—as drivers for heavy-ion fusion, as used in x-ray radiography, and as proposed for transmutation of nuclear waste, respectively, among many other uses. Accelerators and accelerator technology also have significant industrial applications. In the multibillion-dollar semiconductor industry, ion beams from accelerators are used to embed doped layers in

semiconductors. Klystrons that power accelerators are a billion-dollar industry worldwide for use outside of scientific contexts. Other applications include materials irradiation and ion plantation, and research with the SNS when completed will lead to improvements in the range and quality of products used in everyday lives through development of better materials.

But by far the most dramatic impact of accelerators on society has been found in the biomedical sciences, where over 10,000 cancer patients are treated every day in the United States with electron beams from linacs. Tumors are also treated with proton and neutron beams. Also, one in three patients benefits from nuclear medicine and radiopharmaceuticals using isotopes produced in accelerators. Synchrotron radiation facilities and spallation neutron sources are now having major impact on biomedicine through, for example, protein crystallography to guide drug design and neutron scattering to study osteoporosis.

Scientific Opportunities

Given the importance of particle accelerators across the many programs (HENP, BES, FES) in DOE/SC and the sizable cost associated with their construction and operation, it is imperative that their successful development and smooth operation be assured. Over the past decade, advanced simulation has proven to be an increasingly useful tool in the design and analysis of modern accelerators as they continually strive for higher energy and increased intensity. As Figure 2 shows, now is the critical time to accelerate the advancement in accelerator modeling so that the most powerful computing resources can be brought to bear on the challenging design and operational problems facing these complex scientific facilities, the majority of which will be in operation by the latter half of the decade.

Thanks to SciDAC (and its predecessor, the DOE Grand Challenge Program), a new suite of parallel simulation codes has been developed and applied to important accelerator problems with great success, laying the groundwork for the next advancement in simulation capability. The software efforts have focused on electromagnetic system simulation for developing parallel electromagnetic codes for the design, analysis, and optimization of accelerator structures, beam dynamics simulation for developing parallel beam dynamics codes and modules for treating multiple beam phenomena, and advanced accelerator concepts for developing parallel fully electromagnetic particle-in-cell codes for modeling laser- and plasma-based accelerators. With this comprehensive set of modeling tools, the accelerator community eagerly awaits next-generation hardware, envisioned two to three orders of magnitude more powerful, to enable significant advances that will benefit machine performance, new accelerator design, and advanced accelerator development. Meeting these objectives will help safeguard and maximize the nation’s return of investment over a wide spectrum of these expensive scientific instruments.

Beam–Beam Beam Heating Effects to Improve Storage Ring Collider Performance

Colliding beam interactions (beam-beam effects) affect the luminosity of all current and future storage ring colliders such as the Tevatron, PEP-II, RHIC, and LHC. Simulation is

essential for understanding beam-beam effects and for finding optimal parameter choices that lead to higher luminosity. In the "weak-strong" regime where one beam is at relatively low current with respect to the other, parallel simulations using present resources have proved successful in reproducing the lifetime signatures of the Tevatron at injection. For accelerators already operating in the "strong-strong" regime, such as PEP-II, self-consistent parallel simulations are already being used to compute the luminosity, but an enormous number of runs are needed to scan the huge parameter space for best operational choices. Even though PEP-II is operating at twice design luminosity, there is an urgent need is to raise it further to keep up with KEKB, its Japanese counterpart, which is delivering particle collision data at a much higher rate. A two- to three-orders-of-magnitude increase in compute power would extend the weak-strong simulation time on the Tevatron to approach control room time scale, enabling scientists to predict for the first time the lifetime of a storage ring collider and providing insight into the machine's performance. With a hundredfold improvement, the strong-strong simulation would be able to provide PEP-II with a fast, reliable prediction of operation parameters that will not only improve luminosity but also save valuable machine time for physicists to do more science. For proton colliders such as the Tevatron and LHC, the computing requirement is further greatly increased because of extended interactions over the bunch train, a large number of parasitic crossings (two beams sharing the same beampipe), and the need to simulate 10^2 more turns than for an electron collider such as the PEP-II. These runs are within reach only in the thousandfold regime, which would make possible optimization through multiple runs in a detailed parameter scan, leading to improved performance and increased return of investment on these large accelerator facilities.

Colliding beams also generate wall heating in a storage ring collider that could limit the operating current if the heating becomes excessive. PEP-II had to operate at below optimal current because of excessive heating in the Interaction Region (IR) as a result of repetitive excitation of trapped modes by the beam bunches. Under SciDAC, large-scale electromagnetic simulations helped guide improvements in the PEP-II IR chamber that allow the machine to operate at 15% higher current. But beam heating in the IR remains a concern as the current continues to be raised to meet new luminosity goals. Limited by today's computing resources, simulation cannot yet provide the resolution needed to predict this effect. A hundredfold increase in computational capability would allow the beam bunches and the IR chamber to be modeled accurately (including beamline devices) so that simulation can determine if and where wall heating becomes an issue as beam currents, bunch patterns, and crossing angle of the e+e- beams are varied to increase luminosity. Accelerator physicists will then have a powerful tool for avoiding a major limitation (beam heating) while exploring "what if" scenarios to improve the machine's performance.

Dark Current & Wakefields to Optimize Linear Collider Structure Design

Dark-current/RF breakdown are critical issues in advanced accelerating structure R&D, particularly at the high gradients envisioned for the Linear Collider design. Dark current can affect the main beam or increase the background at the detector downstream whereas RF breakdown can cause surface damage and limit structures from reaching higher gradients. Computation for these complex processes is difficult because multiphysics

(e.g., field evolution, surface emissions) as well as multiple scales in time (RF pulse length versus breakdown event) and space (small-scale damage in meter length structure) are involved. In addition, end-to-end simulation is necessary because these effects manifest themselves in the entire structure. While high-resolution modeling has already been established as a core LC capability, SciDAC has further brought about a qualitative leap from component design to system analysis in simulation capability (see Fig. 3). However, today's compute power has limited dark current simulation to a 30-cell constant gradient structure for a fraction of the RF pulse at steady state. A hundredfold increase in computer power would enable dark current to be simulated for a full RF pulse in an actual LC structure that consists of 55 cells with both damping and detuning features included. The results would benefit the LC design because dark current effects such as beam deflection and backgrounds could be accurately quantified. A thousandfold increase would enable the prediction of RF breakdown rates, achieving a fundamental breakthrough in high-gradient structure development.

Wakefields are beam-generated parasitic fields that can dilute beam emittance and disrupt the transport of long bunch trains down the linac. Calculating the long-range wakefields due to a 100 micron bunch in a meter long structure is challenging because the response spans a wide range in time and space scales. Only the basic LC detuned structure (cell-to-cell variation) has been modeled to date because of resource limitations. To accurately resolve the wakefields due to the first dipole band (~17 GHz) in this structure already requires 10 million degrees of freedom or over 200 GB of memory. Resolving the sixth band (~34 GHz) will need over 50 times more computing resources. With two orders of magnitude more computer power, wakefields in an actual LC structure design including both damping and detuning could be simulated, providing the first-ever accurate theoretical analysis of such a complicated structure. With three orders, the simulation would allow additional realism like structure imperfections and misalignment so that virtual wakefield experiments could be a reality. Realizing these golden opportunities would benefit not only the LC and other research areas in accelerator science and technology but also industrial applications such as computer circuit design, which requires electromagnetic modeling at a very large scale.

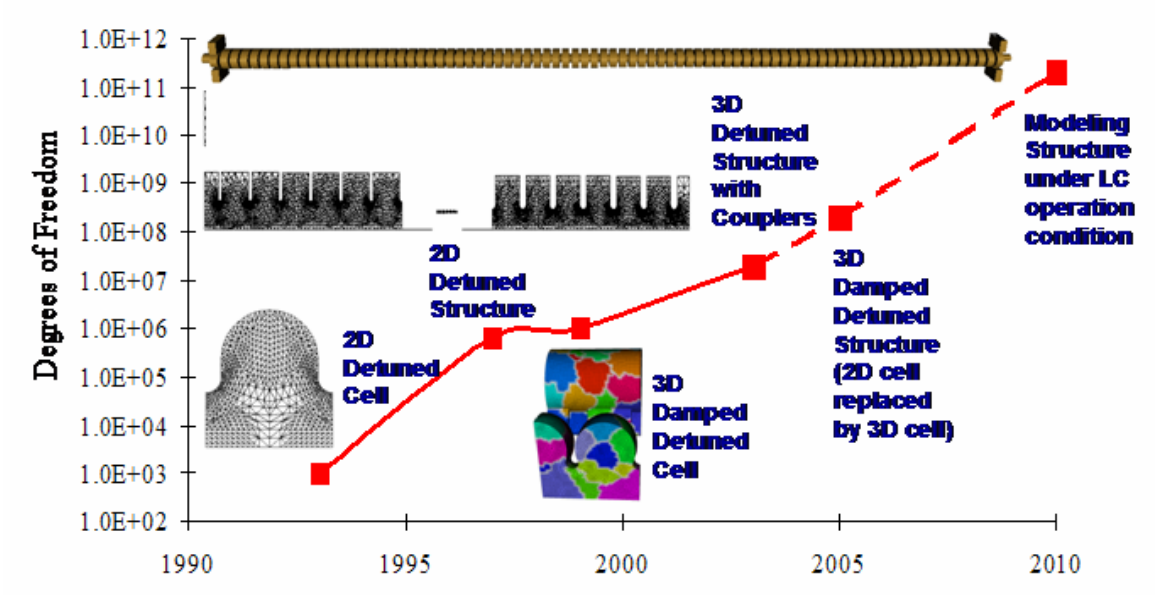


Figure 3. Increase in simulation capability in electromagnetic modeling versus time.

Coherent Synchrotron Radiation to Improve Fourth-Generation Light Sources

Coherent synchrotron radiation (CSR) can degrade beam quality or drive beam instabilities and affects all currently planned light sources that are free-electron laser (FEL) based, such as the Linac Coherent Light Source (LCLS). The most serious CSR-related concern is the microbunching instability, which can be accurately modeled only by high-resolution, three-dimensional, self-consistent simulations using macroparticles. With today's resources, this approach has proved capable of calculating the gain curves for the amplification of small-scale bunch modulations that are in good agreement with theory. A hundredfold increase would be needed to resolve microbunching instabilities in a beam line section in a FEL facility. A thousandfold increase would allow the entire beamline from gun to undulator (including the vacuum chamber as well as drift spaces) to be modeled, making it possible to predict microbunching in a real machine like the LCLS. In view of the increasing number of light sources to be built worldwide and their immense scientific potential, high-end computations are essential for addressing important issues to ensure these accelerators reach their designed performance.

Developing the Breakthrough Technologies for Far-Horizon Machines

The long-term future of experimental high-energy physics research depends on the successful development of novel ultra high-gradient acceleration methods. Laser/plasma systems have been shown to exhibit gradients and focusing forces more than 1,000 times greater than conventional technology; the challenge is to understand and control these high-gradient systems and then to string them together. Such technologies would enable the development of ultra-compact accelerators. The ability to place such compact accelerators at research organizations, high-tech businesses, and hospitals would have staggering consequences for science, industry, and medicine. Thanks to the SciDAC

program, we have developed a foundation of software tools that could be used to impact future accelerator design with a leap in computer and people resources. Using these tools, for the first time we can simulate current plasma/laser wakefield experiments in 3D with explicit particle-in-cell (PIC) models. These are at the scale of 1 GeV in energy and one centimeter to one meter in length. The ultimate role of advanced computing is to use it to design and analyze a 100+ GeV plasma accelerator stage in 3D before extensive capital is spent building one. With a hundredfold increase one could design and test the proposed plasma afterburner, which aims to reach the energy frontier via adding short meter long plasma sections at the end of an existing or planned linear collider! In addition, we could more closely couple the simulations to plasma wakefield accelerator (PWFA) and laser wakefield accelerator (LWFA) experiments by reducing the turnaround time from weeks to minutes, thereby greatly advancing the rate of scientific discovery. With a thousandfold improvement, we could model a 100+ GeV collider based on the computationally more challenging laser wakefield scheme whose length and time scales range over seven orders of magnitude.

Research Issues

Research issues in electromagnetic system simulation, which deals with particle and field interactions within the accelerator environment through Maxwell/Lorentz equations, fall in the areas of model formulation, simulation technology, and software and data management.

In model formulation, advances are needed in implicit time stepping, higher order basis functions, accurate boundary conditions, inclusion of losses, and improved surface physics. In simulation technology, the research areas are identified in the workflow diagram (Fig. 4), starting with geometry construction, followed by mesh generation, domain decomposition, solver application, verification with data, visualization, and solution refinement, and ending in performance evaluation. Work has already started in many of these areas under SciDAC, and fruitful collaboration has been established with the TSTT center on meshing and adaptive refinement and with the TOPS center on solvers and partitioning. Other productive joint efforts are under way in eigensolvers and visualization.

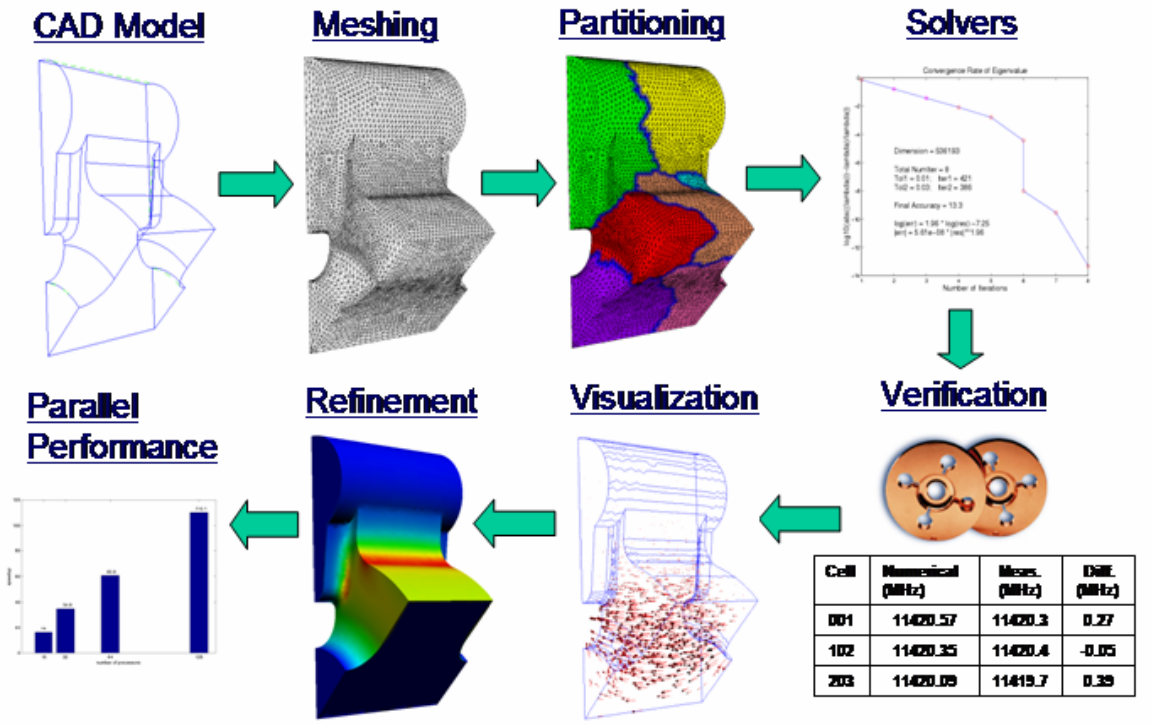


Figure 4. Workflow of a large-scale electromagnetic simulation.

Research issues in advanced accelerator simulation, which deals primarily with tightly coupled irregular particle data on a block-structured field mesh, can also be separated into the areas of model formulation, simulation technology, and software/data management. In model formulation, advances will be required in dispersionless (to one part in 10^4) electromagnetic field and/or Poisson solvers on block-structured meshes with complex boundaries and with mesh refinement, in efficient mesh relaxation methods for nonlinear iterative solutions to coupled field-particle interactions, in accurate impact and field ionization algorithms, and in relativistic fluid solvers on high aspect ratio meshes. In simulation technology, the research areas include effective domain decomposition for fields and particles including load balancing with adaptive meshes, efficient utilization of memory at all levels, run-time verification of reduced models, and performance optimization and evaluation. Research has already begun in several of these areas, and discussions with the APDEC ISIC center have begun on Poisson solvers for complex boundaries and with mesh refinement. Discussions have also begun on how to couple a PIC framework to the appropriate field solvers.

Technology Barriers

Critical barriers to electromagnetic field simulation exist when dealing with systems several orders larger than present ones, including (1) lack of tools capable of generating good-quality meshes with billion elements, (2) partitioning methods for very large meshes that have to good parallel efficiency, (3) real/complex eigensolvers and linear

solvers, iterative or direct, capable of handling matrices of dimension 10^{10} with fast convergence, and (4) visualization techniques that allow fine features to be extracted from terabyte data sets at acceptable speed. Unless these formidable computer science and applied mathematics barriers are overcome, the applications will not be able to fully harness expected computing capability increases to achieve the advances needed for discoveries.

In the software and data management area, there is an increasingly critical need for tools that facilitate code development/integration/reuse, scalable I/O at terabytes per second for a thousand processors, and storage systems/archiving techniques that can handle datasets at the petabyte scale.

Despite the significant progress, critical barriers exist also exist in advanced accelerator simulation. The basic model formulation and simulation technology should scale well to 100,000 or more processors if the problem size scales accordingly. For many of the key problems, however, the problem size (memory requirement) remains fixed while the number of time-steps scales upward. Such simulations typically involve about 10^7 particles, and interprocessor communication becomes oppressive at more than about 10,000 processors because only a few thousand particles are being handled on each processor, with a fraction that could approach 1/2 sent to neighboring processors on each time step. Additional problems arise in the development and use of reduced physics models. In these models, 2D algorithms are embedded into 3D algorithms so the communication patterns are more complex. Better memory techniques clearly will be needed, or the processors will be "starved." Just as for electromagnetics and beam dynamics, for the software and data management area there is an increasingly critical need for tools that facilitate code development, integration, and reuse. In particular, the SciDAC experience has made it clear that interoperable software is critical so that more efficient field solvers, particle managers, and ionization routines can be added without restructuring the code.

Resources Required

Four examples of "high water marks" in accelerator simulations, all performed on 2,048 processors of the NERSC IBM SP3 are as follows:

- Quasi-static particle-in-cell (PIC) code: 208 Gflop/s, equivalent to 7% of peak
- Fully explicit electromagnetic PIC code: 300 Gflop/s, equivalent to 10% of peak
- Beam-beam code (a "weak-strong" model): 167 Gflop/s, equivalent to 5% of peak
- Nonlinear beam optics code: 304 Gflop/s, equivalent to 10% of peak

Typical execution times for these codes are currently 12–24 hours per run. The largest number of processors used to date is 4,096.

Accelerator design codes are often used in parameter studies and error studies involving tens to hundreds of runs. As a result, the time to solution for a single study can be as much as several thousand hours on present hardware.

Even on terascale systems, some problems, such as modeling beam dynamics in accumulator rings, involve simplifications in order to fit into acceptable execution time. For example, an accumulator may contain on the order of 100 microbunches, but only a

few microbunches are used. Only with petascale resources will it be possible to model all of the microbunches. Similarly, beam-beam simulations of hadron colliders are now typically performance for on the order of 100,000 atoms, equal to about a second of beam time. But in order to accurately extract the predicted beam lifetime from the simulation, it is desirable to simulation a few minutes of beam time, which would require approximately 100 times more computation—on the order of a petascale.

Accelerator modelers have begun developing tools to simulate beams in circular machines for hundreds of thousands or millions of turns in the presence of weak space-charge effects and machine resonances. In such a situation, the issue of numerical collisionality is much more stringent than in other types of accelerator simulations. Indeed, as a result of the long simulation time, the disparity of longitudinal motion and transverse motion, and the weakness of the space-charge, the numerical collisionality may overwhelm the physics being studied. Petascale resources are essential because the simulations are both very long and require low noise. Figure 5 places various accelerator simulation problems on a processing/memory diagram.

Electromagnetic Systems Simulations

Large-scale electromagnetic simulations will require a large, well-integrated, multidisciplinary team to support the research tasks involved in the simulation workflow. The beginning of such a simulation team (12 Ph.D.s and 6 graduate students) has been assembled under SciDAC with its members divided among computational scientists, applied mathematicians, computer scientists, and software engineers. As a result, the team expertise extends beyond accelerator physics and electromagnetics to include mesh generation, numerical linear algebra, parallel computing, and visualization, all of which are areas common to many ultrascale simulations. Embedding such experts within the team not only benefits the simulation but also strengthens the collaborations with core applied mathematicians and computer scientists, as they provide the right impedance match between the application and the enabling technologies. In view of the limited SciDAC funding, significant leverage is derived from the base program, accelerator projects and other DOE grants to support this team. A factor of two is the minimum increase required (assuming present leveraging level is the same or greater) to advance the present simulation capability to the level commensurate with a two-to-three-orders-of-magnitude leap in computing power.

Advanced Accelerators

SciDAC's accelerator teams have built the code infrastructure to make the jump to next generation computing resources. This includes the physics, algorithm, and software infrastructure. This team consists of 6 Ph.D.s and 4 graduate students. However, this effort has been highly leveraged with SciDAC directly supporting the equivalent of 1 Ph.D. and 1 graduate student. An example of how this combined effort can accelerate progress was in the area of converting advanced accelerator codes to an E-cloud code. This effort used a combination of physics insight, clever algorithms, and an advanced PIC software framework to build a fully parallelized code in one-tenth the time and man-hours than for the precedent with existing codes. The same sort of partnering as in the E-

cloud will be required for the future. There will be physics teams integrated with algorithm experts, software experts, and visualization experts.

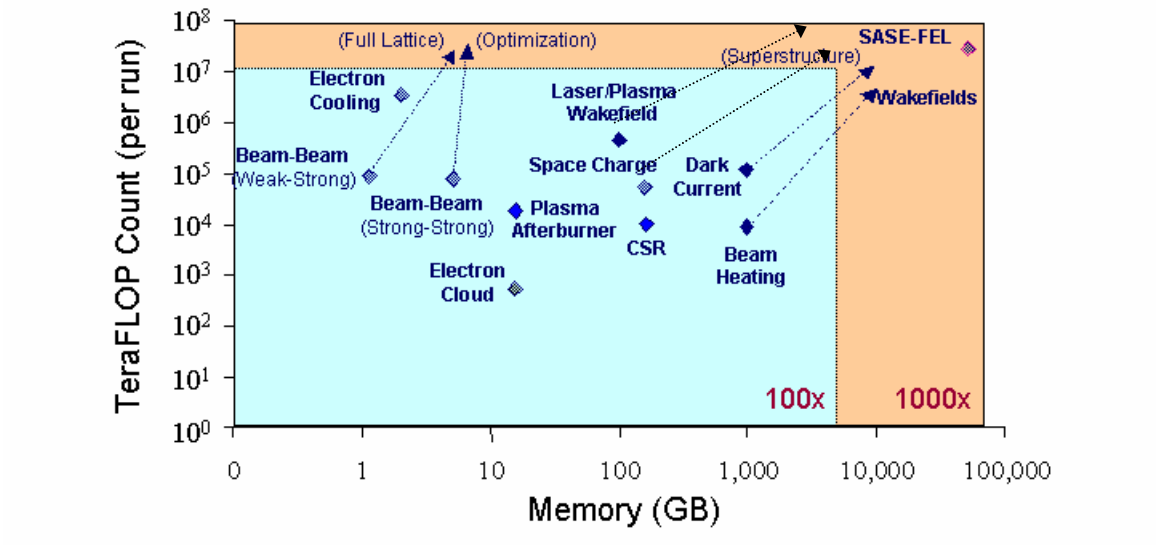


Figure 5. Computational requirements for large-scale simulation problems in accelerator science.

Metrics of Success

Accelerator modeling efforts will need to fulfill a set of metrics that address scientific discovery, accelerator performance, and computational, goals. In the area of new discovery, simulations should verify novel concepts in ultra-high gradient acceleration that could pave the way to compact accelerators and an afterburner option for the LC. Meeting these metrics of success would be a giant step for simulation that could set the stage for a fully integrated modeling of an entire accelerator. For accelerators, the simulations should result in increased luminosity, higher intensity, and shortened commissioning time for machines in operation such as the LHC and SNS. For new accelerators such as the LC and RIA, they should bring about design changes that would optimize performance and lower cost and risk. Computationally, it is essential that the code predictions be validated against experiments. In addition, the simulations should maximize the power of next-generation hardware to reach the longest physical timescales and the highest resolution for the most accurate model at the lowest computing cost.

Astronomy and Astrophysics: Interpreting the Universe

Astronomy—the study of the universe as a whole and of its component parts past, present, and future—is surely one of the earliest sciences pursued by mankind. Its origins are intimately tied to our search for understanding who we are and what our existence means. More recently (within the past half millennium) astronomy has played a central role in the rise of science as an experimental and deductive activity and, in the hands of luminaries such as Galileo and Newton, in the rise of physics as the fundamental physical science. This evolution is also marked by the words used to describe the field today: "astronomy" tends to refer to the more descriptive aspects of the subject, while "astrophysics" is used to describe activities related to the use of physical sciences (including both physics and chemistry) as explanatory tools for what astronomers observe.

Astronomy is now intimately linked to virtually all other sciences. For example, physicists study the nature of fundamental interactions by looking at the evolution of the very early universe and by studying the properties of highly evolved stars—exploding stars (e.g., supernovae), white dwarfs, and neutron stars. Biologists and chemists are examining the origins of life by considering the organic chemistry of the interstellar medium. Geoscientists interested in the origins of the planets are collaborating with astronomers who are finding numerous planetary systems orbiting near-by stars. The profound connections between astronomy and astrophysics and some of the deepest questions faced by mankind continue to this day: What is the origin of all matter and energy? What is the fate of the universe? What is the nature of space and of time? The very recent discovery of "dark matter," "dark energy," and the "accelerating universe" is but one example of the continuing quest to understand the universe.

Impact on Science and Society

Certainly no physical science has succeeded in attracting the enthusiasm and interest of the public to the degree that astronomy and astrophysics have. Astronomy clubs—filled with young and old enthusiasts—are found everywhere in the United States, and astronomy is commonly discussed in the mass media, from newspapers and news magazines to radio and television. It is the only science to have spawned its own literature genre—science fiction—and its own federal agency (NASA), and it plays a role in virtually every federal agency supporting scientific research.[1] Many of the most important areas of modern physical sciences—special relativity theory, gravity and the general theory of relativity, quantum mechanics (including nucleosynthesis and

[1] Astronomy and astrophysics science research is conducted under the auspices of federal agencies such as the Commerce Department, DOD, DOE (both Office of Science and NNSA), NASA, the NSF, and the Smithsonian Institution. It is also a primary focus for at least three major national laboratories (Goddard Space Flight Center, Jet Propulsion Laboratory, and Marshall Space Flight Center).

spectroscopy), plasma physics—grew out of research motivated by or related to astronomical questions. Even in the computational realm, astronomy stands out. It was the first physical science to demand computation. The ability to predict the seasons, notable events such as lunar and solar eclipses, and the motion of the planets hinged on the ability to compute, and the ambition and scope of some of the ongoing and planned astrophysical simulations[2] are the equal of any in science. Indeed, as was recognized by both the DOE/NNSA ASCI and DOE/SC SciDAC programs, some of the most important problems in modern astrophysics—such as the establishment of the universe's distance scale and the nucleosynthesis of the iron peak and heavier elements—can be broached only through "grand challenge" simulation capabilities.

One of the practical consequences of this deep connection between astronomy and the popular imagination is that astronomy and astrophysics have proved to be a strong recruiting tool for attracting students into the physical sciences. This is an essential point at a time when the physical sciences are finding it increasingly difficult to attract "the best and the brightest" of the youth of the United States.

Scientific Opportunities

Astronomy and astrophysics have a growing abundance of research opportunities. As we probe the universe using more and more sophisticated technology, the number of profound (and as yet unanswered) questions has actually increased rather than decreased. In the following, traversing all scales in the universe, we illustrate by example the richness of the questions and problems faced by modern astrophysicists. In all cases, simulations have played a central role; this role continues in the present and is sure to be the case in the future.

Large-Scale Structure and Cosmology. Largely as a result of a new generation of technologically advanced ground-based and space-based optical and microwave telescopes (such as the Keck telescopes, the Sloan Digital Sky Survey, COBE, HST, and WMAP; see Fig. 1), studies of the large-scale properties of the universe, and especially of its formation, have made enormous advances over the past decade. We have now entered the age of "precision cosmology." Most important, virtually all phases of the universe—from its earliest moments to the present—are now thought to be amenable to modeling and simulation, whose aims are to connect what is observed in the distant past to what is observed now in our corner of the universe. Furthermore, important new cross-disciplinary areas of science, such as particle astrophysics and the physics of quark-gluon plasmas, have led to entirely new sets of questions to be addressed, for which simulation will play an increasing and ever-more essential role. These are areas in which the frontiers of astronomy and of physics coincide and where we are as yet uncertain about

[2] As a prototypical example, consider simulations involving Type Ia supernovae, whose "outer scale" is that of the exploding star (about 10^9 cm) and whose "inner scale" is of atomic dimensions (about 10^{-8} cm). This represents a dynamic range of 17 orders of magnitude.

the most basic laws of nature. Much of what is of interest is very complex, and therefore simulation is an essential means by which theoretical progress can be made.

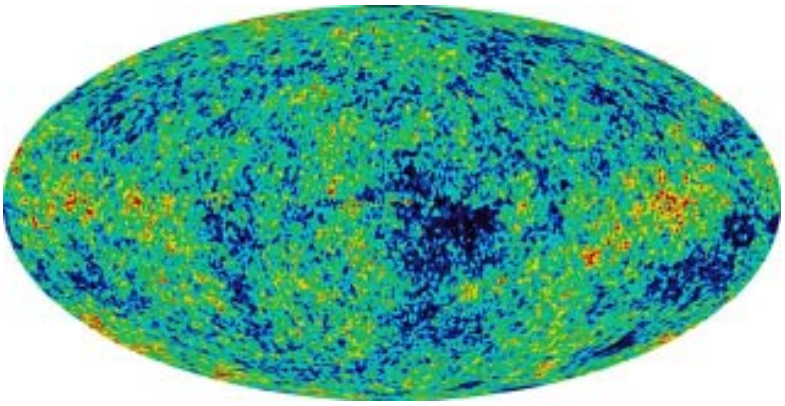


Figure 1. WMAP all-sky picture of the infant universe. The 13 billion year old temperature fluctuations, shown as color differences, correspond to seeds in the mass distribution at that time that grew to become the galaxies.

Galaxy Formation and Interactions. Over the past decade, it has become increasingly evident that the formation of large-scale structure in the universe—while seeded at the time of creation—followed the formation of much smaller-scale structures, namely, galaxies (see Fig. 2). Studies of individual galaxies, as well as their interaction in clusters, will continue to be carried out in concert with a major revolution in observations of these systems, which now use x-rays (for example, Chandra, to trace hot cluster and interstellar medium gas), optical emissions (to trace the stars), IR (for example, IRAS, ISO, and MSX space data, and eventually SOFIA, to trace the cold "baryonic" matter, composed of protons and neutrons, in interstellar medium clouds), and radio (for example, OVRO and BIMA, to trace the interactions of the cosmic microwave background photons with the hot electrons in the cluster gas). These interactions between theory and observations now demand much of theory, well beyond the simple models of just a few years ago. Simulation will make it possible to meet these demands.

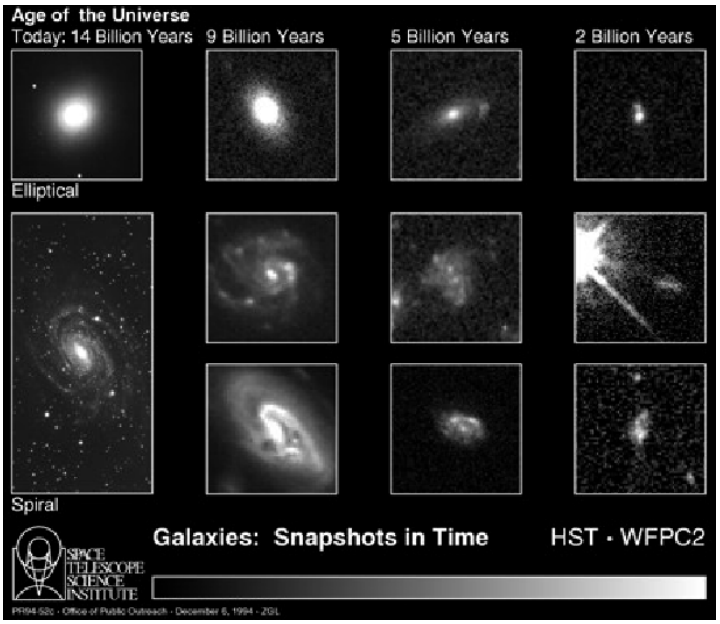


Figure 2. HST images of galaxies at various stages in the history of the universe. The emergence of the present-day Hubble types (ellipticals, spirals) is shown.

Star Formation. Simulations have played a central role in driving our modern understanding of how stars and planetary systems are formed. Using modern instruments observing outside the visible range, including the Very Large Array in the radio, the Owens Valley Radio Observatory, the James Clerk Maxwell Telescope, and the Berkeley Illinois Maryland Array in the millimeter, the NASA Infrared Telescope Facility, Kuiper Airborne Observatory, and Infrared Astronomy Satellite (as well as the ESA Infrared Space Observatory) in the infrared, and Einstein, ROSAT, Chandra, and XMM-Newton in the x-ray, astronomers have been able to penetrate the interstellar dust clouds that have long hidden from view the physical processes leading to gravitational collapse of interstellar gas and star formation. The next generation of instruments, including the Space Infrared Telescope Facility, the Atacama Large Millimeter Array, and the Large Millimeter Telescope, will open a far more detailed view. Simulation has moved us to the brink of being able to put the observations into a unified physical theory of star and planet formation that will allow us to predict the variety of planetary and stellar systems to be found in the universe. One of the most exciting areas in which simulations play an important role is in understanding the variety of evolutionary paths for planetary systems. Why do the gas giant planets in our solar system sit in well behaved orbits far outside the orbit of the Earth, while in many observed extrasolar planetary systems gas giant planets are found at distances from the parent star even less than the distance between the Earth and the sun, or in sweeping elliptical orbits? Ultimately, these models should allow us to

predict the frequency of potentially life-bearing planets in the Galaxy, as well as to understand the origins of our own Earth and sun.

Stellar Evolution. The evolution of stars is marked by a gradual consumption of the interior nuclear fuel and—in rotating stars that have internal convection layers—by a constant level of transient energy release mediated by internal magnetic fields: stellar "activity." In the past, our understanding of these processes relied largely on "one-dimensional" (spherically symmetric) evolutionary models of stars. Only recently have state-of-the-art, large-scale, multidimensional simulations of stellar evolution become possible, with their potential to elucidate new physics. And simulations of processes such as magnetic reconnection form an important bridge between astrophysics and the plasma sciences. This subject area has particular relevance to us for immediate, practical reasons. The magnetic activity of the star closest to us, our sun, is known to have consequences for the Earth's environment ("space weather") and is strongly suspected to influence global climate change.

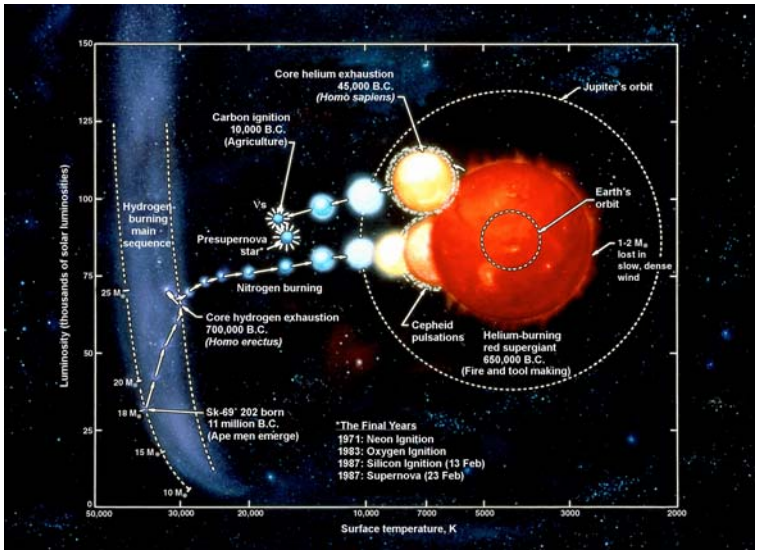


Figure 3. Life cycle of the 20-Solar-mass star that ended its life as the famous supernova SN1987A in the Large Magellanic Cloud, a nearby dwarf galaxy.

Stellar Death. The death of stars through spectacular stellar explosions known as supernovae (see Fig. 3) produce many of the elements in the universe necessary for life and serve as "standard candles," illuminating fundamental and profound aspects of our universe, such as its geometry, content, and ultimate fate. Most recently, through gamma-ray and x-ray observations (from CGRO, HETE, Chandra, and XMM-Newton) of the long-puzzling "gamma-ray bursts" (extremely bright and energetic bursts seen at cosmological distances throughout the sky), an indisputable association between these bursts and supernovae has been made. In addition to their place in the cosmic hierarchy,

the extremes of density, temperature, and composition encountered in supernovae provide an opportunity to explore fundamental nuclear and particle physics that would otherwise be inaccessible in terrestrial experiments: supernovae serve as cosmic laboratories, and supernova models are the bridge between observations (bringing us information about these explosions) and the fundamental physics we seek. In addition, proposed large-scale terrestrial experiments such as the Rare Isotope Accelerator, a priority for the U.S. nuclear physics community, and the proposed National Underground Science and Engineering Laboratory are both significantly motivated by supernova science. Much work remains to be done to elucidate the mechanisms for stellar death via explosion. With the advent of robotic telescopes (KAIT and NEAT) designed to maximize the success rate of finding supernovae, coupled to the use of large-aperture telescopes (such as the Keck, Subaru, VLT, and Gemini telescopes) to measure the detailed spectra of the exploding stars, theorists are faced with new opportunities for detailed testing of supernova models. The consequent demands on simulation will be severe. And we now stand at a threshold. The Laser Interferometric Gravitational Wave Observatory, an NSF-funded gravitational wave detector, is on line, along with other detectors around the globe. Galactic supernovae are among the sources expected to generate gravitational waves that can be detected by LIGO. A detection by LIGO would be the first direct evidence of gravitational waves and the dynamic nature of spacetime as an active, physical fabric and participant in universal phenomena (and not simply a void in which phenomena occur).

Numerical Relativity. Relativity has long had an intimate connection with astronomy and astrophysics—consider Eddington's classic observation of light from stars "bent" by the gravitational field of the sun, the first experimental test of Einstein's theory of general relativity, carried out near the beginning of the twentieth century. Much of the experimental data relevant to general relativity could, until recently, be captured by relatively simple approximations of the full Einstein field equations. However, with the (indirect) discovery of gravitational radiation from binary pulsars (leading to a Nobel prize for its discoverers), the realization that highly nonlinear aspects of general relativity may have astronomical verification has led to a major experimental effort to detect transient gravitational waves (*e.g.*, LIGO). Key to success will be a firm understanding of the physics leading to the gravitational radiation in the first place, and simulations of promising events such as black hole mergers and neutron star mergers are proceeding apace. Thus, much as cosmology has done over the past decade, relativity seems poised for a similar advance to "precision general relativity."

Astrophysical Data. With the advent of modern digital electronics, the replacement of film as both a recording and storage medium in astronomy is virtually complete. Coupled to the computerization of the observational tools themselves—the telescopes—this has driven a revolution in how astronomical data can be used: digitized data (obtained from both targeted and survey observations), coupled to powerful computing capabilities, have allowed astronomers to begin the construction of "virtual observatories," in which astronomical objects can be viewed and studied (on-line) across the electromagnetic spectrum. The vastly increased sophistication of data analysis tools has also affected the relationship between observations and theory/modeling. The observational results have

become more and more quantitative and precise and, therefore, more demanding of theory and simulations. One can also already anticipate the possibilities of mining these enormous data sets to (for example) measure the dark matter and dark energy in the universe through gravitational "lensing" and to find gamma-ray bursts by their optical afterglows, on the ground rather than through space-based missions.

Research Issues

Two cross-cutting issues recur in virtually all subfields of astronomy: multiscale and multiphysics phenomena.

Multiscale Phenomena. As alluded to earlier, the dynamic range in both time and space for typical astrophysical phenomena can be enormous. For this reason the practicalities of effectively simulating these phenomena require the development of subgrid models that correctly describe the physical processes not directly simulated. Subgrid modeling is a science in its own right and requires a judicious combination of theoretical work (in both physics and applied mathematics) and experimental studies that allow one to validate the model. Specific subgrid models that have to be developed to make progress in "cornerstone areas" are discussed below.

Multiphysics Phenomena. Many astrophysical problems involve a broad range of physical processes, not all of which can be captured by a single closed set of evolution equations. The successful coupling of distinct evolution equations, each describing a particular physical phenomenon or process, is still an art rather than a science. It is a forefront research area in its own right. Examples include (1) the coupling of N-body particle and single- or multicomponent fluid equations (in cosmology and galaxy formation/interaction studies); (2) the coupling of photons or neutrinos to hydrodynamics (*i.e.*, radiation hydrodynamics) in a completely self-consistent manner (in core collapse supernovae and gamma-ray burst modeling); and, at an even greater separation of scales, and (3) the coupling of neutrinos to stellar core nuclei in supernovae, which requires both state-of-the-art macroscopic, neutrino transport and microscopic, nuclear structure modeling.

Large-Scale Structure and Cosmology. Key to progress is the development of a "subgrid model" for star formation. At present, large-scale structure simulations treat stars as point masses and do not make any attempts at modeling the details of their formation. However, such details are essential for correct prediction of the large-scale distribution of galaxies by luminosity, morphology, and so forth. Observational data (e.g., from the Sloan Digital Sky Survey) will be of great help in building the requisite models.

Galaxy Formation and Interactions. Galaxy formation requires the development of a "subgrid model" for stellar evolution, from stellar birth through stellar death via supernovae. Challenges include (1) the incorporation of feedbacks to the interstellar and intergalactic media (via stellar winds and supernovae and their ejecta, and the radiation fields that accompany these phenomena; see Fig. 4), (2) the correct treatment of magnetic fields (and their influence on both the dynamics and the energetics), and (3) the inclusion of energetic particles (both their origins and their dynamical consequences).

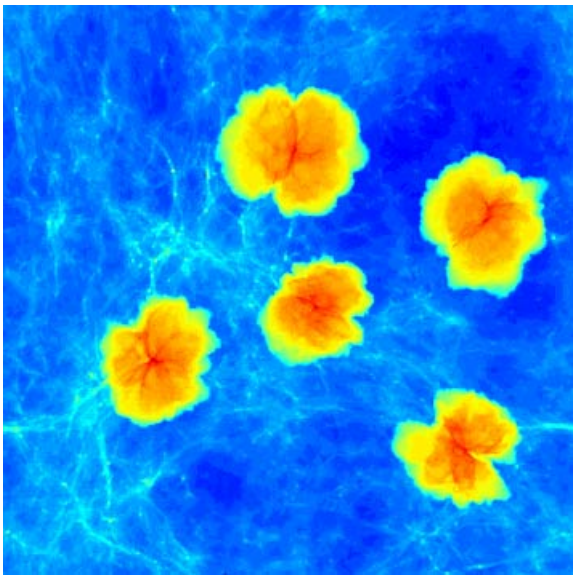


Figure 4. Snapshot from a simulation of the formation of the universe's first regions of ionized (by radiation) hydrogen.

Star Formation. The challenge in star formation studies is to span the dynamic range from star forming clouds of interstellar gas covering many light years, to stars and planets of thousands or tens of thousands of kilometers. The problem involves turbulence, magnetohydrodynamics, self-gravity (solution of the multidimensional Poisson equation), chemical networks, and multidimensional radiation transport, as well as "dusty" plasmas, coupled to the interstellar gas. Because plasma conditions (temperature, density, ionization state, magnetic field intensity) vary enormously in the physical regions of interest, it is not likely that a single evolution equation that correctly describes the physics in all regimes can be constructed. Instead, the challenge is to couple correctly, distinct evolution equations operating in distinct physical regimes. Current models treat cloud formation, dense-core formation, star formation, and planet formation independently and with major approximations to the physics. Future hardware and future software developments must allow coupling of the different scales and improvement of the physics at each scale.

Stellar Evolution. The challenge in simulating the evolution of stars is to develop three-dimensional models incorporating convection, interior rotation, pulsation, (nuclear) chemistry, radiation (both photon and neutrino), and magnetic fields and to integrate the resulting equations on time scales comparable to a star's lifetime. The dynamical time scales (for convection, for example) are enormously smaller than the evolutionary time scales, but at times (e.g., during "shell flashes") these time scales can become comparable. Hence, a single scheme for integrating the stellar evolution equations is unlikely to be successful. Hybrid schemes combining explicit and implicit schemes need to be developed.

Stellar Death. In the case of Type Ia supernovae, the missing ingredients for correctly describing the explosion mechanism are the development of models of a turbulent deflagration (flame) and a deflagration-to-detonation transition (transformation of a subsonic flame to a supersonic detonation) in the unconfined conditions of a supernova, incorporation of these processes into large-scale three-dimensional numerical simulations of the explosions (see Fig. 5), and an efficient use of mesh refinement to make the simulations feasible. In the case of core collapse supernovae, the key issue is to develop a three-dimensional, multifrequency, multiangle radiation (neutrino) transport capability. For either supernova class, the analysis of both the simulation data and the observational data (in order to allow comparison of time-resolved spectra and light curves) are computational challenges in their own right. Three-dimensional, multifrequency, multiangle radiation transport is also needed here to connect simulation data with observational data and to remove systematic errors in supernova "standard candle" determinations of cosmological parameters.

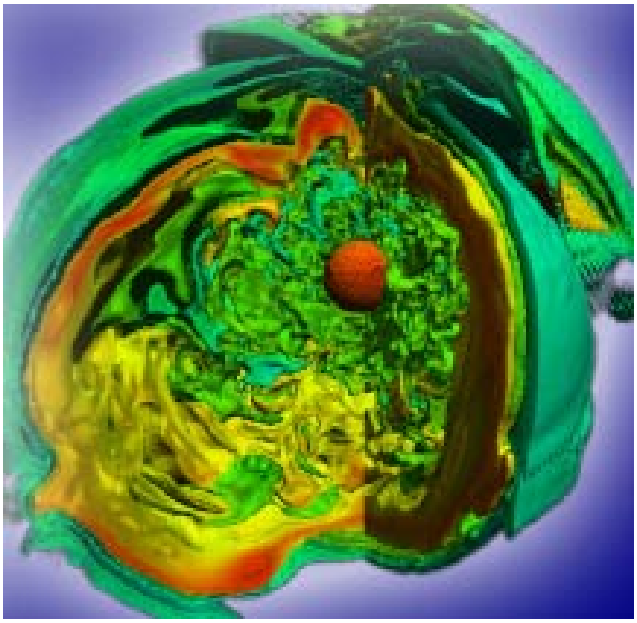


Figure 5. Snapshot from a stellar explosion simulation. Capturing the complex, turbulent dynamics in a supernova environment is a challenge for computational astrophysicists and visualization experts alike.

Numerical Relativity. The challenge in numerical relativity is to simulate black-hole mergers and neutron star mergers sufficiently long to understand the geometrodynamics of spacetime around such mergers and to predict the gravitational wave emission through all phases of the mergers. On a technical level, these simulations will present many of the challenges presented by supernova simulations, with the added complexity that the state of applied mathematics for numerical solution of the Einstein field equations seriously

lags that for the solution of the partial differential equations that describe radiation magnetohydrodynamics.

Astrophysical Data. The benefits of the digital revolution in astronomy, while opening new vistas, have also led to significant costs, mostly driven by the fact that the increased data capturing capabilities (from both telescopes and simulations) have led to a flood of new data, challenging both our ability to distribute and store/archive the data, and to analyze them effectively. Thus, the Sloan Digital Sky Survey has already produced a multi-terabyte dataset. The Supernova Factory using the NEAT Telescopes has surveyed half of the sky, multiple times, producing ~6 terabytes per year. The LSST (an 8-m class telescope with a 2.3 gigapixel CCD) will yield yet larger datasets. And the anticipated very large area optical telescopes (with diameters of 30 meters or more) will trump even these datasets. In all of these cases, unless sufficient attention is paid to data management and analysis, researchers will be unable to effectively use the new data (whether observational or computational). In part, this is an issue of computational infrastructure (e.g., storage, networking, and analysis engines and displays), but in part it relates to the development of standards and middleware that will make the data usable over long periods of time—well-defined data-interface and data-structure standards and the tools for making all this available over the network (e.g., Grid technology).

Technology Barriers

Six technology barriers predominate.

Capacity versus Capability Computing. The state-of-the-art supercomputers available to astronomers today are largely oversubscribed, especially for large simulations that require full use of the entire machine. Thus, queues for large simulations can be discouragingly long, leading to very long turn-around times for these simulations. Physics research—exploring the control parameter space of models by repeated simulations—becomes essentially impossible under these circumstances. While capability computing hardware must certainly be developed to address the target science discussed above, the present situation clearly indicates that future plans for computing resources must be designed for both capability and capacity computing.

Memory Size and Bandwidth. Significantly increased memory bandwidth and, more important, a balance between processor speed and memory bandwidth is the single most desired characteristic of future computer architectures across our science subareas. Moreover, for many forefront astronomy simulations, memory size is a paramount issue as well. For example, for many cosmological N-body as well as supernova simulations (the latter using adaptive meshes), current limits on total memory are the principal constraint on the size of the problems that can be addressed.

Communication Bandwidth. Global reduction operations are at the heart of many of the solution algorithms we use. For example, these operations are required to perform the inner product computations in iterative Krylov subspace methods for the solution of the large, sparse linear systems of equations that arise in radiation transport applications.

Large communication bandwidth will significantly reduce the wall clock time needed to perform such global reduction operations.

Algorithms. In astrophysics simulation, spatial and temporal ranges of 10 to 15 orders of magnitude are common, and increases in raw computing power, as dramatic as they may seem in the everyday world, will not be sufficient to address this problem. Thus, algorithm developments, much more so than leaps in hardware capabilities, will be the key to success for the ambitious simulations outlined here. For example, the development of (a) efficient multigrid solvers (e.g., for the solution of the Poisson equation for the gravitational potential) and implicit solvers (e.g., for the solution of our radiation transport equations), both for massively parallel computers and using adaptive meshes, (b) methods for adaptively varying the time integration scheme (from fully explicit to fully implicit) as the situation demands, and (c) scalable methods to perform global reduction operations are all examples of algorithmic advances that would have profound effects on the efficacy of astrophysics simulations.

Parallel I/O. Substantial efforts are under way to improve the performance of parallel I/O, which has been one of the major bottlenecks in degrading wall-clock performance on massively parallel computers, but these efforts have not yet succeeded in alleviating the bottleneck.

Validation. Because of limited resources, much of the astrophysics code development in the open computing community does not receive the proper level of code validation. By "code validation" we mean assurance that a given code accurately reproduces experimental results for values of the dimensionless control parameters that coincide with the expected regime of validity of the code. Indeed, a key scientific goal of laboratory astrophysics is to provide the experimental data for validating astrophysics codes.

Resources Required

Success in astrophysics research will require advances in hardware and software as well as substantial investment in personnel.

"Hard" Resources. It is relatively straightforward to describe the key characteristics of the next-generation astrophysics simulations, virtually independent of the particular subfield: There must be orders of magnitude increases in resolution and (spatial and temporal) scales covered (relative to existing three-dimensional simulations). The new three-dimensional simulations must include significantly more physics (*e.g.*, sophisticated radiation transport, cosmic magnetic fields). The use of adaptive meshes will see extensive use. And there will be a move toward mixed or fully implicit time integration schemes (in order to handle both longer physical time scales and physical processes such as radiation, which cannot be effectively treated with explicit schemes). All of these attributes of the next-generation simulations have implications for future computing demands. Compute engines with 100 Tflop/s to 10 Pflop/s sustained speeds and large total memory (> 10 terabytes) will be needed. These will likely be obtained through high individual processor speeds, large memory per processor, high memory bandwidth, low latency, and high communication. Petabytes of storage for cached and

archived data will also be needed, as well as network throughputs > 100 Gbyte/s on all networks (both local- and wide-area networks). Moreover, dedicated paths and bandwidth on demand will be essential for effective interactive and collaborative visualization, particularly involving researchers at geographically distributed sites. These will require both hardware (pipes) and software (the development of networking protocols) to provide and ensure such bandwidth. For visualization, terascale visualization hardware (memory, compute power) for data analysis, rendering, and display will be needed, and immersive visualization environments (Fig. 7), which offer unparalleled interactive explorations of complex multidimensional simulation data, must be part of any computational science resource portfolio.

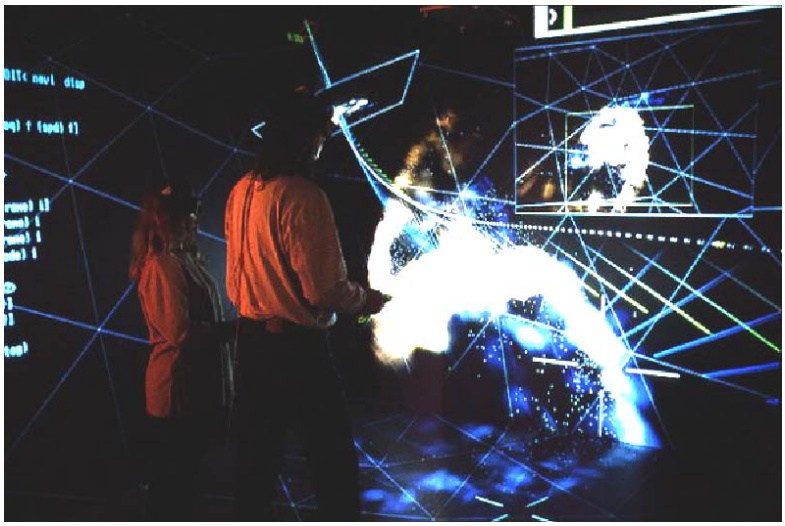


Figure 7. Astronomers can make use of immersive visualization environments to create "virtual observatories."

"Soft" Resources. Investment in hard infrastructure for computational astrophysics must be matched by investment in research personnel (funding for students, postdoctoral fellows, faculty, and staff). The target scientific problems delineated above will clearly require a long-term (more than 10 years), concerted effort by large, multidisciplinary teams. The SciDAC model, coupling applications teams and applied mathematics/computer science teams, or an even more tightly coupled model—for example the DOE/NNSA ASCI/Alliances program, where single teams of application scientists, applied mathematicians, and computer scientists are established—are both viable approaches. A crucial issue is the scale of funding necessary to make real breakthroughs possible.

For example, the SciDAC-supported Terascale Supernova Initiative is a geographically distributed, interdisciplinary team presently involving approximately 40 researchers at approximately 10 institutions (including astrophysicists, nuclear physicists, applied mathematicians, and computer scientists). Its goal is to understand core collapse

supernovae. A future effort with a somewhat expanded scientific scope and a significantly expanded applied mathematics and computer science component is envisioned. The ASCI-supported Center for Astrophysical Thermonuclear Flashes (the Flash Center) is similar, although it has a somewhat broader scientific scope, the core team is larger, and the participants are predominantly geographically collocated. It is focused on Type Ia supernovae, novae, and x-ray bursts. In either case, the essential elements include (1) a critical mass of expertise in the various subdisciplines needed to carry out successful large-scale simulations (from algorithm design and code construction to fundamental physics and astrophysics) and (2) the need to tightly couple disparate activities: the common experience dictates that the activities of the disparate disciplinary groups (astrophysics, applied mathematics, computer science, and physics) must be actively coordinated and managed in order for the overall effort to succeed. These large multidisciplinary efforts also provide significant benefits for the scientific community as a whole because the simulation codes and other computing infrastructure produced by these programs are turning out to be effective community-wide computational tools. In this way, the lessons learned in both the application science and in computational science are incorporated into tools used by the wider community in attacking science problems that may be totally distinct from science problems that motivated the focused center-based efforts in the first place.

Metrics of Success

In each of the subfields of astronomy, the aims of simulation are well defined. Success will be measured first and foremost by the extent to which the simulations reproduce the myriad observations and the extent to which they provide a theoretical framework within which all observed phenomena can be understood. In addition, the extent to which astrophysics codes are validated can be used as a further metric of success of a computational astrophysics program. The extent to which applications in computational astrophysics drive the development of “fundamental” application areas (such as computational fluid dynamics and magnetohydrodynamics, radiation transport and radiation hydrodynamics) and “enabling technologies” (such as data management and analysis, networking, and visualization), which are relevant to many application areas across the DOE Office of Science and across federal agencies, and the extent to which computational astrophysics as an application area helps foster the next generation of computational scientists in the United States will also be measures of success.

Computing Life: From Molecules to Communities

Biology is the study of life from the smallest molecules, *e.g.*, NO, a molecular messenger, to the largest ecosystems, *e.g.*, the Earth. One of the most important challenges for modern biology is to understand how cellular systems communicate, interact, and influence, or are influenced by, their environment. Biological systems have developed amazingly efficient and robust means of maintaining life: capturing and converting energy; communicating between cells, organs and communities; and regulating the most intricate processes of life. These are unifying themes shared by all living organisms from the simplest microbes to man. The genome projects have opened the door to understanding these processes by providing the means and technologies to catalog the basic "parts list" of life. Biologists are now poised to learn how these parts work together and function to shape this living planet. Scientific computing is critical to realizing this goal. The computational challenge is to model and simulate the dynamics and complexities of these biological systems with sufficient detail to predict how those parts function and interact in nature.

Impact on Science and Society

Extraordinary advances in molecular biology have been made in the past decade due in large part to discoveries coming from genome projects on human and model organisms. Biologists expect the next phase of the genome project to be even more startling in terms of dramatic breakthroughs in our understanding of biology and the future of biotechnology. This new biology will allow a level of quantitative understanding of biological systems that was previously unimaginable and will enable the creation of innovative biological solutions to many of humankind's most pressing challenges including human health, sustainable energy, control of atmospheric carbon, environmental cleanup, and effective defenses against bioterrorism. This transformation of biology into a quantitative science requires organization and querying of massive biological data sets and advanced computing technologies that will be combined into predictive simulations to guide and interpret experimental studies. Data management and analysis and computational modeling and simulation will play critical roles in the creation of the biology of the twenty-first century.

Scientific Opportunities

Progress in biology depends on the emergence of a new quantitative, predictive, and, ultimately, systems-level paradigm for the life sciences. New experimental methods must be developed to provide comprehensive, highly accurate datasets; computational infrastructure, software and algorithms must be developed to effectively use these data sets. In addition, a new generation of life scientists must be trained who are facile with the methods of both experimental biology and computational science. Finally, new models for organizing, managing, and funding the biosciences must be developed that will enable large-scale, multidisciplinary research projects in biology. Successful development of the new tools will require the sustained efforts of multidisciplinary teams of biologists, computational biologists and chemists, computer scientists, and applied

mathematicians, and applications of these tools will require teraflop/s-scale and beyond supercomputers as well as the considerable expertise required to use them. This research endeavor is a task for the entire biological community and will involve many agencies and institutions.

In several sidebars we provide success stories in protein structure prediction, large-scale molecular simulations of nucleosome structure and membrane-mineral interactions, first principles approaches to the basic chemical mechanism of DNA cleavage, and large-scale organization of patterns for immunological synapse formation. The progress in computational biology research illustrated in these sidebars provides a strong case history as to the scientific goals that can be accomplished in the future in biology.

Research Issues

The research challenges that have been identified for computational biology fall into three areas: database infrastructure for high-throughput bioinformatics, high performance computing for biophysical and biochemical simulation, and information management approaches to systems biology.

Bioinformatics

To a growing degree biological research involves integrating different types of biological data much of which has been gathered by research teams in different locations, curated by separate sets of experts, and stored in different databases. The facile use and integration of this data will therefore require means to seamlessly link these databases and extract data into a single software package for analysis and visualization. The diverse tasks in handling biological data are captured by the term "bioinformatics" that includes a range of computational analysis that is characterized in part by its reliance on biochemical, genomics, gene activity, and proteomics data, as the critical feature of the investigations. Sequence analysis, largely the prediction of genes and gene function by homology, has been a core task.

There are many challenges in bioinformatics. Most databases and software tools have been developed as research tools and but have not been "hardened" (to use computer science terminology) for widespread use by biologists in production environments. Definitions of biological terms vary among research laboratories, and the disparities make it difficult to integrate data from multiple sources. And finally, not only is the amount of data increasing rapidly, but the modes of analysis are changing from the comparison of single entities, such as the sequence of a particular gene, with similar entities from other organisms, to comparing multiple entities, such as entire genomes, with sets of other multiple entities. This means that the complexity and computational cost of the analysis is increasing much more rapidly than the amount of data that is available since the high throughput systems being applied now are creating a tsunami of new data.

Databases need to be constructed for key types of data such as genomes, pathways, regulatory networks, protein interactions, global protein expression profiling, scientific literature, and data generated from system or cell simulations. Methods for integrating or

transforming such data also need to be developed. The need for new types of data infrastructure is clearly evident, and it is necessary to move away from dispersed repositories arising from "data collecting" towards conceptually integrated "knowledge enabling" repositories. Bioinformatics infrastructure must be more than a storage and retrieval capability, and must support fundamentally new ways of doing science.

Biophysical and Biochemical Modeling and Simulation

To fully understand the biological machinery of a cell and its interactions with other cells and the environment, it is critical to know which proteins directly interact with each other and with other molecules in a cell, how these proteins are structurally docked to form a complex, and how the proteins of a complex interact dynamically to accomplish a biological function, such as the repair of a DNA defect. Such a detailed level of characterization of protein complexes is the prerequisite for understanding molecular and cellular functions, particularly functional mechanisms, at a systems level, *e.g.*, in a cell, a tissue or a whole organism.

The challenges to molecular-level simulations of biological processes include the large size of biomolecules and the long time scales of many biological processes, as well as the subtle energetics and complex milieu of biochemical reactions. Despite these challenges, there are a vast number of biochemical processes for which molecular simulations will have a major impact on our understanding. These problems include the elucidation of underlying protein-protein or protein-DNA interactions and the dissection of the catalytic function of enzymes. The promise of such modeling studies is rapidly growing as a result of the development of linear-scaling computational chemical methods and molecular modeling software for massively parallel computers. Computational biophysics and chemistry is most clearly identified as the area with the most need for high performance computing, with relatively mature methods and algorithms widely deployed on all current high performance computing platforms.

Systems Modeling and Simulation

Biosystems models encapsulate our understanding of the greater organizational levels found in biology such as cells and organisms. Simulation is becoming a key tool to further our understanding of biology at the systems level. It will be through computer-aided analysis of predictive mathematical models that we will gain the understanding required to manipulate genomes to proteomes, regulatory networks, metabolic pathways, and cells and microbial communities, to solve outstanding scientific problems.

The mathematical and computational challenges associated with modeling and simulating biological systems derive to a large extent from the complexity of such systems compounded by the wide range of temporal and spatial scale at which these systems operate. A major, and as yet unresolved, question is how to "map" genomic and molecular-level data onto higher-level conceptual models of observed system function. Extrapolating from the genomic to the community level requires building models containing multiple levels of complexity ranging from the molecular, cellular, and intercellular up to the community level. These levels must be linked so that we can predict how changes at one level propagate both upward and downward to adjacent levels

in the model. To achieve this ambitious goal, it will be necessary to build a hierarchal *information management* system within a high-performance computing environment that allows us to represent complex, dynamic cellular processes. This will include a unique set of tools to facilitate multiple approaches to biological modeling and simulation.

Technology Barriers

Bioinformatics

Developing new databases. New databases will be needed to describe the genome, proteome, metabolome, and genetic networks of relevant organisms. Databases for the most important organisms will require a high level of ongoing curation and refinement to incorporate the constantly emerging information about the gene products and pathways of each organism.

Improving database and software inoperability. Bioinformatics data is spread across hundreds of databases all over the world, is stored in a bewildering variety of data formats and software packages, and is accessible through numerous different interfaces. In addition, new data sources are constantly materializing and new data formats are being invented; new analysis techniques are being developed and new computational resources introduced; and new combinations of data and computational resources arise. The methodologies used to accomplish interoperability must therefore be flexible, scalable, and handle changes to the environment gracefully. In addition, as information is copied or derived from other data, its provenance must be accurately tracked in a manner analogous to citation of sources within the scientific literature. By combining database, systems, and software engineering expertise with challenging, biologically driven problems, environments providing integrated access to all of the information and tools required to address these problems will support the complex workflows required by scientists and be used to answer queries and workflows taking anywhere from seconds to days to execute, depending on the complexity of the task, the speed of data sources and analysis programs, and the capabilities of the underlying network.

Improving genome annotation systems. Genome annotation systems typically consist of a high-throughput computational analysis pipeline that runs gene finding software to identify genes, applies search programs (BLAST, HMMer, *etc*.) to identify sequence similarity to proteins or protein families for which data already exists, and executes programs that identify other genome features such as tRNAs, operons, terminators, and so forth. Integration of these multiple computational analyses provides functional predictions for many newly found proteins.

Typically, the outputs of these programs are stored either in a relational database or as flat files, and human annotators perform manual synthesis and refinement of those outputs before its release to the scientific community. Whereas current genome annotation requires significant amounts of manual intervention, and produce predictions of unknown quality, the next generation of annotation should provide automated prediction of protein function whose accuracy is on par with, or exceeds, that of the best human experts, and will probably need to utilize several approaches including: an expert-systems methodology for encoding the expertise of human annotation experts; fusion of

evidence generated from systematic application of multiple analysis tools including; and proper processing of multifunctional and multidomain proteins that identifies what region of a protein a predicted function is associated with.

Biophysical and Biochemical Modeling

A wide variety of biochemical simulation methods are available that vary in accuracy, scalability, and computational cost, but all strain the capabilities of the fastest supercomputers.

Quantum Mechanics and First Principles Molecular Dynamics. The most accurate methods for predicting the structure and energetics of biomolecules are quantum mechanical methods, which involve solving the electronic Schrödinger wave equation for the distribution of the electrons in atoms and molecules. There is a hierarchy of methods for solving the Schrödinger equation, ranging from those that computationally scale almost linearly with the number of atoms to much more accurate methods that scale as the seventh power of the number of atoms in the molecule. Although the best of these methods can achieve accuracies for energies and structures as good as or better than experimental methods, they are too computationally costly to be applied to most biochemical processes. Research is needed to develop versions of these methods that scale less steeply with system size, or to develop ways to empirically correct less costly methods using the knowledge gained from more accurate approaches.

Many biological processes are inherently dynamical, and *ab initio* or first principles molecular dynamics approaches could be used to address these problems. This capability would allow a large number of fundamental biophysical problems that have been inconclusively addressed by existing classical MD methods to be solved. These problems include the determination of the hydration structure of the DNA nucleoside bases, the energetic factors leading to DNA base pairing, and the hydration of DNA.

Classical Molecular Dynamics and Thermodynamics. These methods use an empirically derived force field to simulate the motion of each atom, and are highly developed for the simulation of nucleic acids and proteins. Typical molecular dynamics simulations involve hundreds of thousands of atoms that are simulated for many nanoseconds of time, with the largest published simulation being a 1 microsecond simulation of a small protein. The many nanosecond time scale is thought sufficient to capture structural relaxation and solvent reorganization, and is long enough in some cases to simulate transitions between different macromolecular conformations. However, many biological processes such as DNA replication (>1 millisec per base) or protein folding (millisecond to seconds) require far longer timescales. Improvements are required to increase the timescales possible in MD simulations. In addition, fundamentally new methods for optimizing macromolecular structures and sampling different configurations are needed.

Extended system equations of motion and associated numerical integrators have been developed that allow extensions from micro-canonical ensemble dynamics to sampling of states in the canonical ensemble as well as in the isothermal-isobaric ensembles. Recent advances in modern numerical integrators can now separate out the natural timescales of motions that depend on the strength of forces associated with each term in an empirical

force field, thereby increasing computational efficiency by an order of magnitude in biomolecular systems. Calculations performed using these multiple time step integration methods are very scalable, with each time step being a collective "move" that can be parallelized using standard domain decomposition paradigms.

Large proteins relax on timescales of an order of a second in solution, a benchmark atomistic simulations cannot approach at present. Although improved numerical integration and equations of motion have helped, several orders of magnitude improvement in efficiency needs to be obtained. By combining high performance computing with enhanced sampling methods it should be possible to achieve large reductions of computational effort in the sampling of configuration space, allowing the simulation of large complex biochemical systems on the timescales at which they operate.

Mathematical optimization research. Mathematical optimization is a more general approach for obtaining solutions to large nonlinear systems with numerous local minima; protein structure prediction is a recent example. Constrained optimization methods rely on the availability of sufficiently well defined constraints so that the desired solution is the only available minimum, or one of few available minima, in the optimization phase of the algorithm. Alternatively, global optimization and conformational search techniques attempt to systematically search the potential energy surface to find all low-lying minima including the global energy minimum. Various deterministic and stochastic global optimization techniques have been developed over the last decade, but application of these conformational search and optimization strategies to biology are computationally intensive since they typically require millions of evaluations of an energy function and its derivative. These optimization approaches are useful in many contexts including atomic-level structure prediction, for use in comparative modeling, docking of small ligands into protein active sites, and finding optimal protein-protein interaction geometries.

Systems Modeling

Conceptual Systems Models and Tools. It is impossible to predict *a priori* how a cell or system will behave without extensive experimental data, but experimental data are difficult to interpret without an initial conceptual framework. In advancing the technology, there are many complex issues associated with the *dynamic abstraction* and *representation of biological information* for modeling or simulation efforts. For example, which features of the system should be represented as systems of differential equations or as networks of connectivity? There is also a major requirement for large-scale *knowledge production* from basic experimental and computational data sources, often from integrating multiple diverse data streams. Biologists recognize this level as the familiar inductive reasoning process that results in "telling a biological story" based on multiple clues. This process typically requires a large number of *feature extraction* tools from both experimental and computational raw data sources; for example, interpreting mass spectrometry peptide data or recognizing patterns in genomic sequences. Finally there is a wide range of challenging issues surrounding large-scale data handling and management to ensure the pedigree of information and full, integrated access to disparate databases.

Network analysis. In addition to the deterministic behavior described by classical differential equations, effective models will need to incorporate stochastic behavior and uncertain parameters. In particular, solving hybrid systems of differential equations, discrete equations, and stochastic equations using computational approaches remains a challenge. Both the theory for understanding the long-time behavior of these highly complex generalized dynamical systems, and the numerical methods for actually solving them, are still in their infancy. Furthermore, computational analysis tools are needed for the development and verification of models, including capabilities for: sensitivity analysis, parameter estimation, automated bifurcation analysis, and model reduction and verification.

Information Management. There is a need to develop an environment and a community resource that facilitates model development through access to existing models and associated software libraries, to multiple distributed sources of biological systems data, and to high performance computer systems to run comprehensive simulations. An integrated computational infrastructure for collecting, interpreting, and distributing the information gathered from the experimental efforts should be developed. This resource would include tools to model and simulate key processes in individual cells, key aspects of entire individual cells, microbial communities, biofilms, and other systems of relevance.

Resources Required

Database Infrastructure for Bioinformatics

There will be several large-scale DOE facilities that will have the need to capture bulk data from many different measurements and instruments in large-scale data archives. Each facility will need tailored database solutions linked to teraflop/s computer facilities for real time data reduction. These facility databases archives will need terabit/s networks linking them for the establishment of a distributed knowledge base that can be mined to drive modeling and simulations of large scale biological systems. Research will be needed to determine how to store these data; develop representations and models for data and metadata from many different measurements and assays; develop data exchange and format standards for facilities and the community; and design efficient query and retrieval methods for large datasets. A high performance hardware infrastructure will be needed to provide rapid and flexible access to very large (petabyte) data volumes. There will be many types of data, each with algorithmic research and development challenges for analyzing high data throughput.

Computing Resources for Biophysical and Biochemical Simulations

Biophysical and biochemical modeling requires high performance computing well beyond what is currently available, although several other factors are critical including the size of primary system memory, the size and speed of secondary storage, and the overall architecture of the computer, *i.e.*, the latency and bandwidth of the switch and the speed with which local memory can be accessed.

Using computers in the 100 Tflop/s class, we expect major progress in the predictive power of classical MD and first principles MD (FPMD) techniques for simulations of biological systems. We estimate that access to such computers will enable FPMD simulations of systems comprising several thousand atoms for several picoseconds, as well as of systems comprising proportionally smaller numbers of atoms in the nanosecond range; classical simulations will be able to realize size and timescales two to three orders of magnitude larger than the estimates for simulations using FPMD.

Even given the very broad range of simulation methods required by computational biology, it is possible to provide some guidelines for the most efficient computer architectures. Regarding the size of primary memory, it is usually most efficient if a copy of the $6N$ set of coordinates describing a time step of an N-atom molecular dynamics simulation or the $N \times N$ matrices describing the N-electron quantum chemical wavefunction can be stored on each processing element. For the biological systems of the sort described in this white paper, this corresponds to a minimum of several hundred megabytes of RAM per processor. However, for the real problems that need to be solved, we will have to take advantage of all of the available memory in the system and a NUMA (non-uniform memory access) model made tractable by using tools such as Global Arrays that hide the memory locale and latency of access from the user/developer. The availability of such tools from computer science is critical to the success of the biochemical/biophysical modeling efforts described herein.

Similarly, general estimates can be made for the minimal interprocessor communication rates. Since the goal of parallel processing is to distribute the effort of a calculation, for tightly coupled methods such as quantum chemical simulations, it is essential that the time to communicate a partial result be less than the time to simply recalculate it. Assuming Gflop/s speeds for individual processing elements in the parallel computers, this translates roughly to an upper bound of Gbyte/s interprocessor communication speeds.

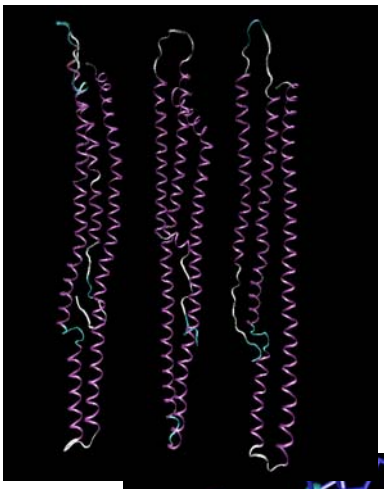
Information Management Systems for Systems Biology

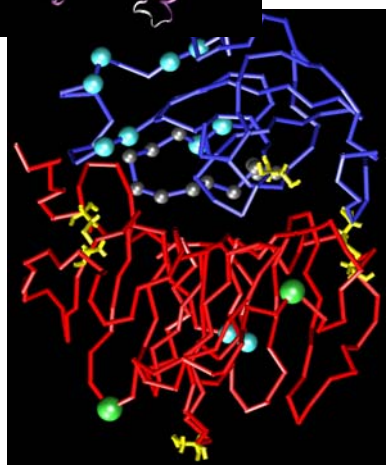
Systems biology has the need for new types of databases (both hardware and operating system) that can accommodate large data volumes with great schema complexity that allow rapid, flexible query retrieval. To help drive systems biology approaches, integrated information systems that address the complexities of the systems need to be built. The challenge is to provide systems for experiment design, sample specification, sample tracking and metadata recording, workflow management, process optimization and documentation, quality assurance, and sharing such data (petabyte scale) across facilities or projects.

Metrics of Success

The ultimate metric of success for computational biology will be its adoption as a central tool in biological research. Just as computational modeling is the first step in any modern engineering effort or large scale physics experiment, we envision a future where biological experiments are conceived, designed and analyzed by computational simulations, and that a major goal of biological experiments will be to provide parameters

for and validate models. A key to such adoption will be the unambiguous solution of major biological research questions by simulation-driven approaches.





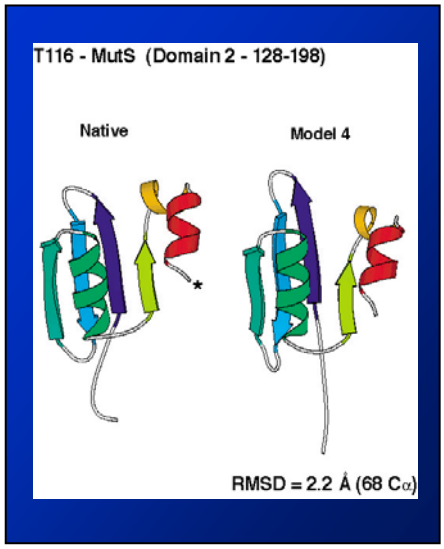
PROSPECT successfully recognized the multi-domain vitronection in CASP4.

Sidebar #1

From Fold Recognition to Ab Initio Structure Prediction

Three structure prediction methods, PROSPECT, ROSETTA, and SPSC, are ordered in their increasing reliance on existing structural information on known proteins and increasing computational complexity of the underlying structure prediction method. When used with next generation computers, they are ready for immediate use in predictions of protein structure for relevant protein complexes.

PROSPECT predicts a protein structure based on determining the best sequence-structure alignment between a query sequence and template structures in its structure database. PROSPECT assigns a confidence value to each prediction, based on assessment of various statistical and physical properties of the predicted structure using SVM (Support Vector Machine) technique. PROPSECT allows the user to specify protein-specific information, e.g., a disulfide bond between two residues or a binding site involving a group of residues, and uses this information as threading constraints. PROSPECT has been thoroughly tested through the CASP3 and CASP4 structure prediction experiments.



The ROSETTA code demonstrated some of the strongest predictions in the most recent CASP4 competition. The ROSETTA method combines sequence-structure alignments on small pieces of known tertiary templates combined with statistical and physical-based energy functions to produce backbone structures for the hardest category of prediction: proteins with no sequence similarity or structural homology to known proteins. The current protein domain size limit for this approach is about 150 residues, although a large fraction of proteins either

fall in this size range or can be decomposed into domains within this size for use in GTL4.

on structure
g ROSETTA.

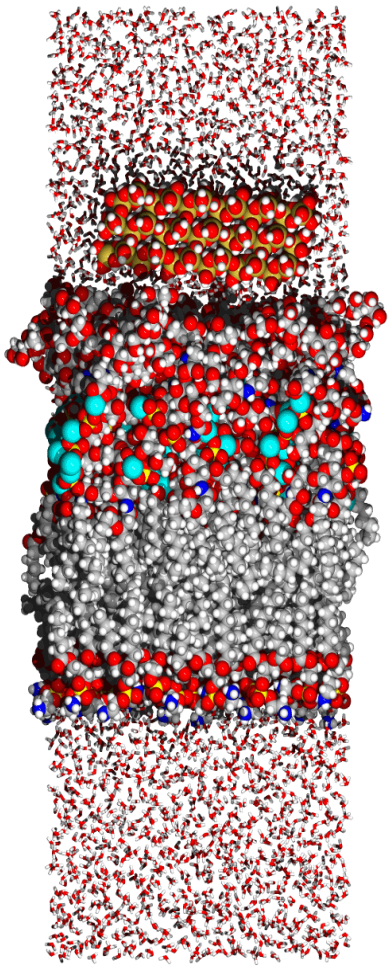
Stochastic Perturbation with Soft Constraints (SPSC) indirectly uses information from known proteins to predict secondary structure, but not in the tertiary structure or in generating the terms of the energy function. The SPSC approach is characterized by the use of an all atom energy function that includes a novel hydrophobic solvation function derived from solution scattering experiments that shows ability for discrimination against misfolded structures. The SPSC method and energy function were used for blind prediction in the ab initio category for the first time in CASP4, and gave the best prediction for one of the most difficult targets of the competition, a new fold protein of 240 amino acids.

Comparison between experiment (center), submitted SPSC prediction (right) with RMSD of 8.46Å, and next generation run of SPSC (left), that converged to a final prediction with RMSD of 7.7Å

Sidebar #2

Simulations of P. aeruginosa LPS Membrane Goethite Mineral Interactions

This work shows that large size-scale simulation of multiple nano-second dynamics is ready for immediate deployment in evaluating function for relevant proteins and their complexes.



Molecular representation of LPS membrane of Pseudomonas aeruginosa in near contact with the goethite mineral

The interactions of lipopolysaccharides located on the outer membrane of gram-negative bacteria with mineral surfaces are modeled to study how such interactions affect metal uptake and mineral dissolution. This is leading to better ability in predicting the molecular processes involved in microbial metal binding and microbial attachment to mineral surfaces. The scale of these interactions ranges from the molecular scale, where individual ions bind to specific functional groups, to membrane interactions with minerals and other surfaces that can take place on the scale of microns.

Molecular modeling studies of the rough lipopolysaccharide (LPS) membrane of *Pseudomonas aeruginosa* interacting with a model for the goethite mineral surface are being carried out to analyze the effect of the presence of the mineral on structural and dynamical properties of the polysaccharide core, the adhesion properties of the two surfaces, and to compare result to experimental data. The calculations are being done with the computational chemistry code, NWChem, that handles both electronic structure and molecular dynamics on massively parallel computers, and is portable and scalable.

For the model for the LPS membrane, good agreement was found for the lipid layer density compared to reported values for a phosphatidylethanolamine bilayer membrane. This property is very sensitive to the atomic charge model used in the MD simulations. We have used NWChem and massively parallel computers to obtain an improved set of partial charges from complete *ab initio* electronic structure calculations on a realistic model with more than 100 atoms. Nanosecond MD trajectories for the 40,000 atom LPS membrane system were generated and analyzed by using a variety of methods, including RMS deviations, energetic decomposition, water, ion and functional group distributions, and essential dynamics analysis. These calculations are providing direct information on how LPS membranes attach to mineral surfaces.

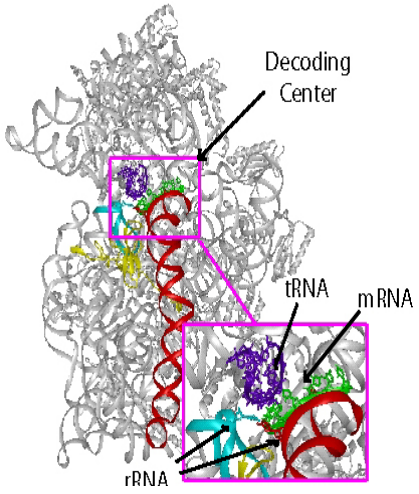
Sidebar #3

Modeling the Ribosome

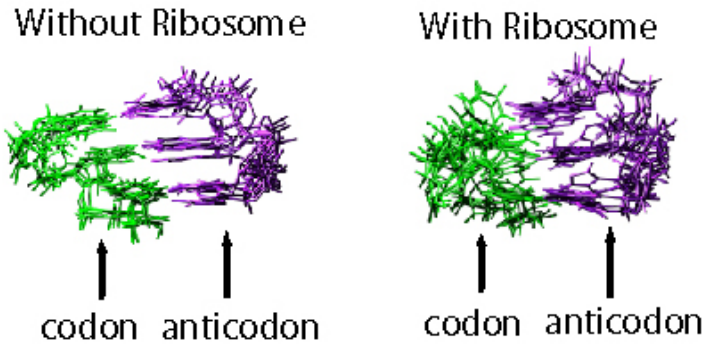
This work provide another perspective as to how large size-scale molecular dynamics simulations lead to understanding of functional mechanism that will be important in the GTL initiative.

The ribosome is a molecular motor that possesses the apparatus necessary to read the genetic code and translate it into specific proteins. As one of the most highly conserved structures in the cell, the ribosome is crucial to all forms of life. This molecular machine is a complex consisting of three RNA molecules and over fifty proteins, arranged into two major subunits. The large subunit is the largest asymmetric structure solved to date; the entire complex has yet to be solved to atomic resolution. Scientists are currently engaged in an effort to determine, on the molecular level of detail, how the many components of the ribosome act in concert to accomplish its function. The major part of ribosomal function (elongation) consists of three steps: (1) decoding, where the ribosome selects the correct amino acid to add to the nascent polypeptide chain, (2) peptidyl transferase, where the amino acid is added, and (3) translocation, where the ribosome moves along the messenger RNA. These are the first molecular dynamics simulations that address step (1) by simulating the decoding center in complex with correct and incorrect tRNA. These simulations have helped to elucidate the mechanism by which the ribosome discriminates between correct and incorrect tRNAs.

In particular the simulations have revealed that the ribosome destabilizes incorrect codon-anticodon pairs. The molecular mechanisms of steps (1) and (2) are rapidly being uncovered. Step (3), however, requires knowledge of the structure of the entire ribosomal complex, which has yet to be determined to atomic resolution. New RNA homology methods have been developed that successfully modeled the *E. coli* small ribosomal subunit (this has not been crystallized). An *in silico* structure of the entire ribosomal complex is being constructed, which will put us in a unique position to understand translocation via all-atom molecular dynamics simulations.



Decoding center of the small ribosomal subunit was simulated with molecular dynamics simulations to elucidate the mechanism of discrimination.



Superimposed configurations of molecular dynamics trajectories for an incorrect codon-anticodon pair without ribosome and with ribosome

Better Living through Computational Chemistry

Chemistry—the science of molecules—is the science of the everyday world. Molecules are the fundamental units of matter: smaller than this, matter loses its macroscopic identity. An understanding of the structure, interactions, and reactions of molecules is thus of critical importance to a wide range of phenomena, from the fate of contaminants in the environment, through the production of plastics from crude oil, to the occurrence and treatment of genetic diseases. In a world attempting to balance energy usage, environmental quality, human health, and economic prosperity, chemistry is, and will continue to be, the key science required to meet the challenge.

The fundamental physical laws defining molecular behavior are known and embodied in quantum mechanics as the Schrödinger and Dirac equations. As the Nobelist Paul Dirac stated in 1929, "The underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble." Chemists have spent the past 75 years developing solutions to these equations with much success, though still restricted for such first-principles simulations to relatively simple molecules.

Chemists are now about to cross a remarkable threshold and expect a dramatic expansion in their ability to make reliable predictions about molecular structure and processes. This advance is due to the confluence of advances in theoretical, computational, and experimental capabilities, allowing understanding and characterization matter at the atomic and molecular level. By integrating the chemist's capabilities in the areas of synthesis and characterization with computational modeling and simulation, it will soon be possible to use computation to design molecules to do what we want, and to control how we make them.

Impact on Science and Society

Chemistry is a central science. It is an intellectual quest in its own right, but it is also a critical element of many of the other sciences important to the U.S. Department of Energy. For example, with reference to other chapters, chemical processes are responsible for the energy produced and pollutants released by an automobile engine (combustion science). Catalytic chemical processes are used to remove these pollutants from the tailpipe emissions. Likewise, an understanding of chemistry is important for predicting and mitigating the spread of pollutants in underground plumes (subsurface science), processing high-level radioactive wastes, predicting and alleviating the long-term effects of greenhouse gases and stratospheric ozone depletion (climate science), tailoring the properties of nanomaterials for a broad range of applications (material and nanoscale sciences), and understanding and manipulating the biochemical processes on which life is based (biology).

The U.S. chemical industries play a major role in the national economy and contribute directly to the high quality of life enjoyed by its citizens. There is little in everyday life that does not involve these core U.S. industries. The U.S. petroleum, chemical, and pharmaceutical industries are the world's largest producers of chemicals, ranging from "wonder" drugs to paints, from cosmetics to plastics, from fuels to fertilizers. The chemical industry represents 10% of all U.S. manufacturing, employing more than one million Americans. It is also one of the few industries that possess a favorable balance of trade. The petroleum and chemical industries contribute approximately $500 billion to the GDP of the United States. These industries maintain their global competitive position by their ability to produce new products using energy-efficient, low-cost, and environmentally clean processes. An understanding of and an ability to predict the structures, energetics, and reactions of molecules are essential to achieving these goals.

Scientific Opportunities

Computational chemistry has already had an enormous impact on fundamental chemical science. It has provided new insights into the structure and properties of molecules, especially those molecules whose existence is fleeting (e.g., free radicals) yet whose role in many chemical processes is critical. Computational chemistry has also dramatically advanced our understanding of how chemical reactions occur—in the energetic changes that drive chemical reactions, in the structural changes that occur as reactions proceed, and in the detailed dynamics of the collision complex. These insights were initially qualitative; but as computational methods, software, and computer hardware have advanced, chemists are now able to make accurate predictions about a wide variety of chemical phenomena, sometimes literally replacing experiment by computation. With the additional investments envisioned in this report, the capabilities of computational chemistry will dramatically expand, enveloping a far broader range of chemical phenomena than possible today.

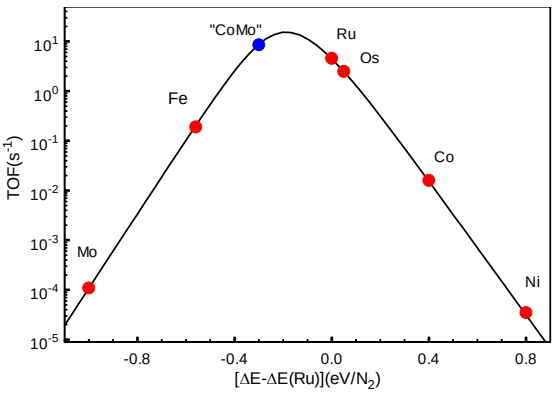


Figure 1. Calculated trends in ammonia synthesis rates (TOF) shown as a function of the dissociative chemisorption energy of nitrogen. The figure shows, in agreement with experimental evidence, that Ru, Os, and Fe are the best catalysts and how a CoMo alloy should be as good as the best elementary catalysts.

Chemical Catalysis. The chemical industries take a cheap feedstock (e.g., a hydrocarbon) and convert it into a higher-value product by rearranging the atoms or adding functional groups. Catalytic processes are directly involved in the synthesis of 20% of all industrial products. A catalyst's role is to make a desired chemical reaction proceed much more efficiently than it otherwise would. To date, catalysts have been largely designed by using trial and error, synthesizing and testing a potential new catalyst to determine whether yields are improved, unwanted byproducts reduced, and/or energy requirements decreased.

This process is both expensive and time-consuming, and it rarely leads to novel new catalysts. Advanced experimental tools are still unable to provide data on all of the steps involved in catalytic processes. Computational modeling and simulation can fill this gap, enabling the design of catalysts from first principles. For example, ab initio catalyst design requires quantitative information about transition states for key reactions. This information is accessible only by computational methods. In one successful example (see Fig. 1), a combined simulation/experimental approach was used to design the first new ammonia catalyst since Haber and Bosch's work in the early 1900s. However, to make a major step forward in catalyst design, we must gain at least a factor of a thousand increase in the performance of simulations.

*Combustion Chemistry***.** The chemical reactions involved in combustion determine the rate at which fuel is consumed and energy is released, as well as the quantity of pollutants produced. Thus, a thorough understanding of the chemistry of combustion systems is required to meet two important design goals for tomorrow's advanced combustion systems: reducing unwanted emissions and improving system performance. The hydrocarbon fuels involved in everyday combustion systems are complex mixtures of large hydrocarbon molecules. For example, gasoline primarily consists of iso-octane and n-heptane as well as performance enhancing additives. Laboratory studies cannot obtain all of the data needed to model combustion. Many species have only a fleeting existence, yet are critical for sustaining the combustion process; many reactions only occur at the high temperatures attained in flames, yet the harsh conditions of combustion systems are difficult to reproduce in the laboratory. Recent advances in computational chemistry and the efficient implementation of these advances on terascale computers have provided a new level of capability for characterizing the chemical species and reactions involved in the combustion of the key components of gasoline, iso-octane (C_8H_{18}) and n-heptane (n-C_7H_{16}). With an increase of a factor of a hundred in computing power, researchers will be able to begin studies of n-hexadecane ($C_{16}H_{34}$), one of the major components of diesel fuel. This capability could revolutionize the design of new combustion systems on which the nation depends for 85% of its energy needs.

Nanoscale Science. Nanoscale structures and processes exhibit qualitatively new behavior—the systems are too large to be treated by quantum mechanics today and may be governed by microscopic collective phenomena, yet are too small to behave according to the laws of continuum mechanics. Chemistry is at the heart of nanoscale science, much of which is the study of very large molecules. Although a number of experimental techniques for characterizing nanoscale systems have been developed, many details of nanoscale structures and processes remain unobserved or unobservable. Computational methods are already being used to simulate the properties of carbon nanospheres (buckyballs) and nanotubes, and the flow of fluids confined in spaces just a few nanometers in extent. These simulations clearly establish the connection between the structure of the molecules involved in nanoscale processes (e.g., hydrocarbon lubricants) and the unusual processes often observed (e.g., alternate "freezing" and "thawing") behavior in fluid flow. An increase of two to-three orders of-magnitude in computational capabilities will enable a molecular understanding of friction, the design of new lubricants, and the development of a new generation of molecular electronic devices.

Molecular-scale devices offer several advantages over conventional technology, including miniaturization that will allow the scaling of component size to the ultimate level of atoms and molecules A concerted integrated theoretical effort has been aimed at developing the simulation and design tools to quantitatively model electronic transport through organic molecules in an open environment where the molecules are placed between semi-infinite (macroscopic) metallic electrodes. The ultimate goal is to compute current-voltage curves in order to reproduce and optimally predict real experiments.

*Biochemistry***.** Although biological systems have many unique features, at the most fundamental level, they are molecular machines. Proteins and enzymes, molecules that are essential to life, have long been the province of experimental and computational chemists and biochemists. Computational chemists pioneered protein simulations, continue to be leaders in this field, and are now developing new approaches for molecular simulations that hold great promise for elucidating the mechanisms of enzymatic reactions, for understanding and characterizing protein-protein interactions, and for understanding the intimate relationship between molecular structure and biological function. Carbohydrates, drug molecules, neurotransmitters, the permeabilities and structural properties of cell membranes, and nucleic acids (the material of genetics) are equally important, and computations will be key to their ultimate understanding. Advances in computational chemistry are critical to understanding biological systems at their most fundamental level, and reliable predictions will require a factor of at least three orders of magnitude in computational performance.

Research Issues

Significant theoretical, mathematical, and computational obstacles must be overcome to realize the full potential of computational modeling and simulation in understanding and controlling chemical change. We are confident, however, that these problems will be solved with appropriate investments in basic research. Our confidence is based on both experience and recent theoretical developments that, in selected areas, point a way forward. In this section, we summarize some of these research issues.

One of the key needs for the improved design and control of chemical processes is a*ccuracy*. We can now routinely make reliable predictions (errors of less than 1 kcal/mol) of the heats of formation of small- to medium-sized gas phase molecules (10–20 atoms) with reasonable computational effort. In addition, for some simple gas-phase reactions, we can predict reaction rates over broad temperature ranges to within a factor of two to three. However, the computational cost of these calculations scales as N^7, where N is related to the number of atoms in the molecule. This means that we cannot rely solely on increases in computational power if we want to make predictions about the much larger molecules involved in many chemical processes important to the DOE. We must expand the progress that has already been made in reducing the scaling of molecular simulations. In fact, expected increases in applications software capabilities during the next few years will finally allow calculations on molecules that are large enough to realize a significant reduction in the scaling exponent. In addition, we need to include and quantify the effects

of relativity to predict reliably the properties and behavior of (radioactive) heavy-metal compounds.

The ability to treat the environment in which a reaction is taking place, *e.g.*, a solvent, requires new theoretical and algorithmic developments in order to make reliable predictions about the reaction rates in the condensed phases that are so important in chemical production systems, on atmospheric particles, and in biological systems. Modeling chemistry at the nanoscale also introduces new challenges, including multiscale methods and new infinite (open, as opposed to periodic) boundary conditions and also multiscale methods. Excited states also need new theoretical treatments. Electronically excited states play a key role in many important chemical processes, including combustion and photosynthesis, yet we currently have no efficient computational approach that can quantitatively describe such excited states for other than the smallest molecules.

Another critical need is the extension of atomistic simulations, such as molecular dynamics, to much longer length and time scales. Macroscopic time scales in molecular dynamics simulations will not be reached solely by increases in hardware because of fundamental causal limitations on how many time steps can be executed simultaneously on a computer, whether parallel or serial. At present, it is routinely possible to simulate atomistic systems, or systems represented as interacting atoms such as proteins and polymeric systems, for periods of the order of tens of nanoseconds using first-principles dynamics. However, far longer timescales—microseconds, milliseconds and beyond—are needed to simulate such processes as phase transitions, rare events, enzyme kinetics, and protein folding. New methods are also required for effective sampling of infrequent events, exploring “rough” energy surfaces, and modeling reaction dynamics on surfaces with many minima.

Although data management and analysis have not posed limitations on computational chemistry in the past, on current generation supercomputer systems computational simulations are creating as-yet-unresolved data management and analysis problems. For example, a single time step of a million-atom simulation produces tens of megabytes of data, which, integrated over millions to billions of time steps, yields terabytes to petabytes of data. Traditional “store and retrieve” techniques cannot be used to analyze these simulations. Computational chemistry algorithms for data analysis “on-the-fly” must be developed and refined.

We have not come to the end of innovation in theory, mathematics, and algorithms, just as we have not come to the end of improvements in computational speed and memory capacity. These innovations will extend the power and utility of computational chemistry well beyond extrapolations based on today’s scaling laws.

Technology Barriers

Applied and numerical mathematics. One cannot overstate the impact of improved numerical methods on computational chemistry. Examples include the standardized robust and well-implemented linear algebra routines (e.g., BLAS, ScaLAPACK), and the

recent applications in chemistry of the fast-multipole method (FMM). These accomplishments were, in large part, supported by the DOE. Major problems are simulating multiple length scales in nonperiodic systems, long-time simulations, efficient sampling of infrequent events; dynamics and statistical mechanics of high-dimensional systems; and reduced scaling of electronic structure methods with molecular size and computational precision. Other necessary developments include improved FMM to reduce both the break-even point and the cost for high precision, numerical methods (e.g., multiresolution, multigrid, low-separation rank) suitable for efficient and reliable computation in three, six and higher dimensions as well as solving inverse and optimization problems such as commonly appear in either process design or the interpretation of experimental data.

Algorithms. Modern numerical methods trade precision for speed, and to do so with maximum efficiency to guaranteed precision requires robust and sophisticated algorithms. For instance, the FMM algorithm describes how to compute the potential due to collection of charges to a specified precision in a time that increases linearly with the number of charges. On a parallel computer, an algorithm must also describe the distribution and movement of data and work between the processors. Additional algorithms and techniques are necessary to accommodate the latency and bandwidth of memory, which are not keeping pace with processor speed. Anticipating future computers with tens to hundreds of thousands of processors, we need algorithms that exploit many levels of parallelism and include dynamic load balancing. Algorithms that self-adapt to the available architecture are also desirable. Furthermore, the intricacy of contemporary chemical models is exceeding the ability of humans to implement them on modern computers. Increasingly important will be automated generation of near-optimal algorithms and software from high-level mathematical expressions.

Software. Essentially all of the complexity of the theories, algorithms, computer hardware, and programming models are visible at the software level. Even now, only by using sophisticated design and software engineering techniques is this complexity manageable in computational chemistry applications. If we are to successfully use next-generation computers and algorithms, we must aggressively seek new and yet more sophisticated software approaches. Improved and fully interoperable parallel programming tools supporting multiple levels of parallelism are essential and must provide higher abstraction levels than current tools. The following remarks are generic to computational science but are emphasized in the context of chemistry. The poor quality of hardware vendor software and the increasingly user-*un*friendly environments of large computers must be improved. In addition, performance analysis and debugging tools must be capable of dealing with large and long-running simulations. Domain-specific frameworks could ease management of workflow and resources.

Hardware. Chemistry algorithms are very diverse, and no single machine architecture meets all needs in a way that is balanced for each in terms of the performance and capacities of processor, memory, interconnect, and I/O. Future algorithms are expected to be less regular and not so dominated by dense linear algebra operations as are many current methods. Processor support to tolerate high memory latency is clearly important. Larger caches might enable classical molecular dynamics to become fully cache resident

on big parallel computers. This capability will dramatically increase the speed of each time step in these simulations. To achieve long simulated times, however, we must use as many processors as possible; such scalability requires low-latency and high-bandwidth interprocessor communications. Electronic structure calculations require high-performance for matrix multiplication and fast Fourier transforms. Fault tolerance is again a major concern. Further, fault tolerance on systems of thousands of processors is a concern, which may have algorithmic as well as hardware and systems software solutions.

Resources Required

In this section, we estimate the resources (theory, software, and hardware) required by the DOE Office of Science to advance computational chemistry in support of DOE's missions. We do not include resources needed to overcome the technology barriers.

Advancing theory is crucial to the future of computational chemistry. This places a high premium on innovation and will be accomplished largely by research groups of single principal investigators—a largely under-realized opportunity for the DOE Office of Science. However, their vitality is threatened by diminishing funding and the relatively small size of university single-PI grants. The Office of Science investment in basic theoretical research, including statistical mechanics, electronic structure, reaction dynamics, and classical dynamics, must be expanded significantly and new PIs brought into the program. The rapid and effective transfer of ideas from research to production computing must also be encouraged and supported.

We must also create and support a full suite of computational chemistry simulation software that efficiently implements our models on computers ranging from PCs to building-sized ultrascale computers with millions of processors. In an accompanying side bar, we analyze the required resources with reference to the known development costs of a particular code. The expected complexity of future computers and algorithms, the breadth of the science applications, and our experience to date lead us to the following findings.

- The major applications software systems will best be developed with broad, community-based collaboration including computational chemists, computer scientists, mathematicians, and end users.
- A disciplined approach to software development is vital. Although computers become obsolete in three to four years, scientific applications have lifetimes of decades.
- National computational chemistry facilities to coordinate, develop, and support key applications should be considered. The U.K. Collaborative Computational Projects are successful, long-lived role models for the envisioned effort, although their funding level is inadequate for the current challenges.
- Additional software development resources are needed to exploit current opportunities: about 20 full-time staff in electronic structure, 20 in molecular dynamics/statistical mechanics, and 10 in reaction dynamics, plus postdoctoral fellows and graduate students.

- Development of expert systems and domain-specific problem-solving and collaborative environments will require 15–20 additional full-time staff. Such tools will become essential as the scale of data management and computation increase (not just the size but the complexity and number of calculations).

In the long-term, as much as one-third of the manpower must be devoted to software support and maintenance.

Whereas a large fraction of these resources will be required to enable capability computing on ultrascale computers, capacity computation on smaller machines must not be neglected. Chemists need a large high-performance computing capacity because of the vast number of molecules of interest, the complexity of the calculations, and the large number of chemists using computational tools. In turn, capability computing is required for large, high fidelity simulations, to ensure accurate predictions of critical molecular properties, and to provide highly accurate benchmark results against which to test more approximate methods.

Currently with density functional theory-based codes, we sustain about 0.75 Tflop/s on 256 Itanium 2 processors, realizing about 50% of peak. The turnaround time is typically a couple of days. The current algorithmic scaling is $O(N^3)$, where N is the number of electrons. The high sustained performance comes from BLAS3 dense linear algebra, which dominates the workload. A factor of 10 to 20 increase in system size is currently coveted, with corresponding 1000 to 8000 increases in computing time, respectively. Given today's sustained performance of slightly under 1 Tflop/s, a 1 Pflop/s capability would deliver this increase.

No single machine is appropriate for all of the algorithms represented in chemistry and chemists emphasize *balance* in all aspects of the hardware, with our large temporary disk I/O requirements perhaps differentiating chemistry from many other scientific disciplines. This diversity suggests opportunities to improve cost effectiveness by matching the computer architecture to the algorithms, as illustrated by the recent MD-GRAPE and WINE processors—hardware specifically designed to speed MD simulations. Beyond 5–10 years, it is difficult to anticipate the details of computer architectures, but it is essential that chemists have early access to advanced architectures (e.g., FPGAs and PIMs), including the use of simulators, to evaluate and influence the designs, and to optimize our software.

We estimate that about 40 (peak) Tflop/s-years are annually dedicated to chemistry applications associated with non-ASCI DOE work, aggregating allocations on DOE's supercomputer centers, laboratory and group clusters, and desktop workstations. The discussion in previous sections concerning hundredfold to thousandfold increases in resources is relative to this figure. These next few orders of magnitude are needed as soon as possible. In combination with concurrent advances in theoretical and computational methods, this expansion of compute power will enable truly revolutionary scientific breakthroughs. We cannot emphasize enough that without resources for the actual computations, we will fail to realize the scientific discoveries that the entire infrastructure is intended to enable. For instance, the limited computer and manpower resources

presently available have discouraged most computational chemists from investing the effort necessary to develop massively parallel software.

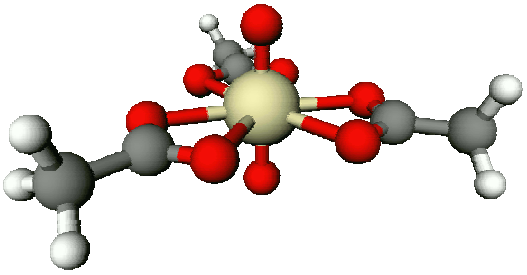
Metrics of Success

Science-based metrics should be used to measure improvements in the performance, predictions and impact of computational chemistry. Combined with measured execution rates on specific hardware, these metrics may also provide some measure of the efficacy of that hardware for selected applications. The reduced dependence of computational chemistry calculations on accuracy, molecule size, and/or time span may be measured and compared against past performance. Community-defined chemistry benchmarks (i.e., the time to solution of specific chemical problems with prescribed levels of theory, accuracy, and precision) can quantify these advances. Another family of increasingly large, community-defined chemistry benchmarks could measure increased capability. The expanded simulation capabilities will also be demonstrated by new scientific discoveries. Another measure of the broader impact of computational chemistry is the increased uptake of computation by industry, which typically lags use in national laboratories and universities by several years. In addition, approximately 15% of journal publications in chemistry currently employ simulation in some form; this percentage can be tracked.

Sidebar #1

Heavy element and environmental chemistry – a special responsibility of DOE

Four decades of nuclear weapons production at U.S. Department of Energy facilities has resulted in the interim storage of millions of gallons of highly radioactive mixed wastes in hundreds of underground tanks, extensive contamination of the soil and groundwater at thousands of sites, and hundreds of buildings that must be decontaminated and decommissioned. The single most challenging environmental issue confronting the DOE, and perhaps the nation, is the safe and cost-effective management of these wastes. What is the physical and chemical form of the wastes in the tanks and in the ground? How can the radioactive wastes be safely processed? How can the processed waste be safely stored? Understanding the chemistry of radionuclides has long been a special responsibility of DOE's Office of Science, and, given the difficulty of handling radioactive material and the loss of experimental expertise in heavy-element chemistry in the United States, improved ability to model the chemistry of these species is becoming a critical national need.



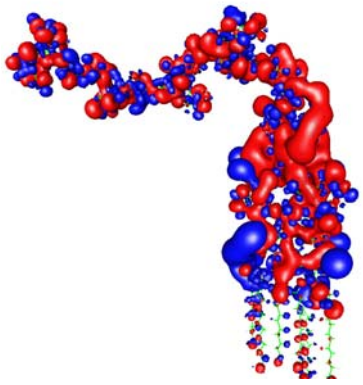
Uranyl triacetate. The structure, thermo-chemistry, and vibrational spectra of model radioactive species can be reliably predicted with relativistic quantum chemistry on high-performance computers. Fully detailed models, and other properties such as electronic spectra, are currently not feasible, in general.

Computational chemistry can provide fundamental information on the complex interactions of radioactive materials with other species, often enabling the replacement or curtailment of expensive experiments. Interactions in the soil determine how contaminant plumes spread and transform. Interactions within the complex chemical mixtures inside the waste tanks determine how to transfer, process and safely store the wastes for thousands of years. Such information is needed to solve DOE's cleanup problems in a comprehensive, cost-effective, permanent way. Reliable predictions require the combination of the theory of relativity with quantum mechanics and will require 100 to 1000 times current computational capabilities.

Sidebar #2

What is required to develop the next generation of computational chemistry software?

Solution of significant chemical problems requires the use of multiple theoretical models. As a result, computational chemistry packages contain a broad range of functionality and are large, complex software systems. Consider some of the major U.S. chemistry packages for solving the electronic Schrödinger equation: GAMESS, GAUSSIAN, NWChem, and QChem. These packages have up to one million lines of code (e.g., NWChem), each has particular strengths for certain types of simulation (e.g., QChem includes new linear scaling algorithms), each has a large following in the chemistry community (e.g., GAMESS at 10,000 sites and NWChem at 1,000 sites worldwide), and each is long lived and required substantial investments to develop (e.g., GAUSSIAN is now over 30 years old, with an investment estimated at 500 to 1,000 person-years).



Electrostatic potential of lipopolysaccharide and counter-ions (936 atoms) from NWChem constrained RESP fit to HF (6-31G basis, 7943 functions).*

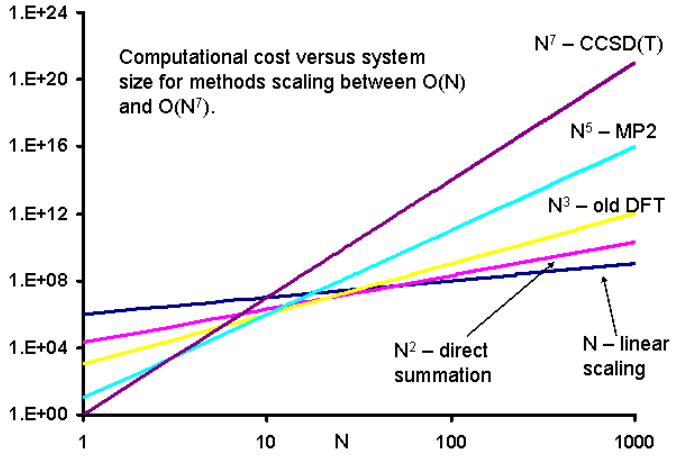
To develop a new generation of software packages for solving the electronic Schrödinger equation on the computers being considered in this report will require a substantial investment of resources. To estimate the level of investment, consider NWChem, a software package designed from the ground up for massively parallel computers using the latest software engineering practices. NWChem was intended to run efficiently on 100 to 1,000 processors, with some tasks recently redesigned for up to 10,000 processors. The NWChem project, which began in 1992, involved a core group of five computational chemists, a computer scientist, and an applied mathematician, augmented by a worldwide group of more than twenty contributors. Over 100 person-years were invested directly in the development of NWChem v1.0.

Successfully exploiting a thousandfold increase in the complexity of future computers, algorithms, theory, and scientific problems, as well as supporting a much larger base of users, cannot be accomplished with the small core team plus contributors such as that which developed NWChem. There is a need in the next decade for the following resources defined relative to the size of the NWChem team: electronic structure (3x), molecular dynamics (3x), reaction dynamics (1x), and problem-solving and collaborative environments (2x).

Sidebar #3

Search for the computational Holy Grail – linear scaling algorithms in chemistry

The combination of larger, faster computers with theoretical and algorithmic advances will lead to a dramatic increase in chemists' ability to simulate molecules and their interactions. To understand the algorithmic side of the interplay, consider this schematic depiction of the relative computational cost of several algorithms used to compute molecular energies.



The CCSD(T) is an accurate method yielding results that rival experiment. However, its cost increases as N^7, where N is the number of atoms in the molecule. To simulate a molecule twice the size requires a computer 128 times larger.

Clearly, reducing the scaling exponent is critical to future progress in computational chemistry. As shown in the figure, however, there is a "break-even point" below which methods with a lower scaling exponent actually cost more than methods with a higher exponent. In addition to the exponent, we must also work to reduce the prefactor, which is the cost of simulating a system of unit size.

Traditional methods for solving the DFT equations scale as N^3, as a result of dense linear algebra operations. New algorithms circumvent this cubic term, leaving terms scaling as the square of the number of atoms. Algorithms such as the fast multipole method, developed recently by applied mathematicians, exploit multi-level representations of the molecule and can reduce the scaling to linear (N). Development of linear scaling algorithms for highly accurate methods, such as CCSD(T), is more challenging. Here, some progress has been made in reducing the scaling exponent, but much work remains to be done to obtain linear scaling.

New insights, as well as additional computing resources, are required to advance computational chemistry. Nevertheless, recent progress makes us confident that we will succeed, and also that larger computers will push us past the "break-even point" into the regime in which the algorithms will scale linearly with molecule size and enable molecular simulations on an unprecedented scale.

Computing the Climate

The National Academy report entitled *Climate Change Science: An Analysis of Some Key Questions* (June 6, 2001) identified the need to "reduce the wide range of uncertainty inherent in current model predictions of global climate change" by producing advances in "the understanding and modeling of the factors that determine atmospheric concentrations of greenhouse gases and aerosols and the so-called feedbacks that determine the sensitivity of the climate system to a prescribed increase in greenhouse gases." For policy makers to have the best information from climate scientists, leadership-class research programs must be maintained within the national laboratory and university systems.

Figure 1. Our laboratory and home. The atmosphere, oceans, and biosphere interact over about fifteen ranges of spatial scale, from molecular diffusion to global circulation.

Climate simulations based on the mathematical description of ocean and atmospheric flows coupled with chemical and biological process models are providing scientists with new insights into earth's complex climate system (see Fig. 1). The new coupled models are able to balance the fundamental physical quantities of energy, momentum, mass, as well as freshwater, salinity and chemical species among terrestrial, ocean, ice and atmospheric pools. The modeling of the global carbon cycle to better understand the interactions, effects and feedbacks in the climate system is one goal of present scientific research (see Fig. 2).

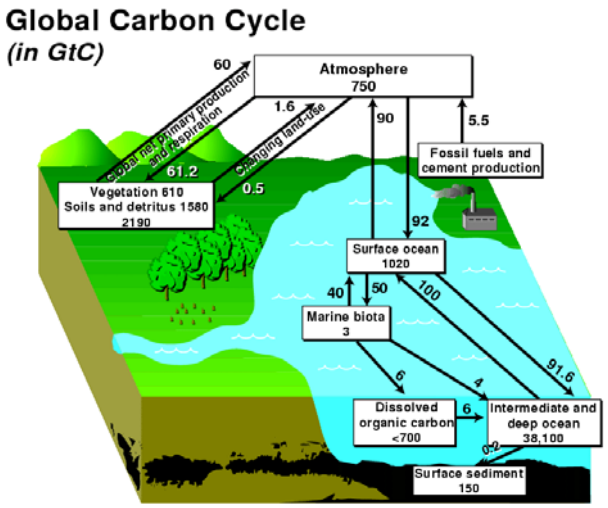


Figure 2. Major carbon transfers between components of the atmosphere, oceans, and biosphere.

Impact on Science and Society

An international consensus is emerging that humans are changing Earth's climate. Climate change is expected to continue and even accelerate. Clearly, future climate change will have important impacts on many sectors of society, including agriculture, water resource management, energy production and demand, human health, and recreation. Natural ecosystems and biodiversity will also be affected. The cost of adaptation to climate change could be large, and we must attempt to anticipate and quantify potential damage resulting from climate change. Adaptation strategies might reduce the damage, but such strategies will also have an associated cost.

Greenhouse gases such as carbon dioxide have long residence times; hence, delay in reducing these gases may dramatically increase costs and decrease effectiveness of mitigation strategies. Accurate long-term predictions that include as many known feedbacks as possible will be required to evaluate the impacts of climate change and the effectiveness of emission-reduction scenarios and carbon-sequestration methods. Policy makers need such tools now. A better understanding of potential societal impacts is needed to properly weigh the costs of mitigating climate change (e.g., by developing new carbon-free energy sources or developing carbon sequestration technologies) against the costs of allowing climate change and its impacts to occur. Demonstration and implementation of carbon sequestration methodologies and new carbon-free energy production technologies will require decades to develop. Whatever policies are followed in the future, anthropogenic climate change will continue for decades, and the explanation of observed changes will require high-end climate modeling.

Scientific Opportunities

Improved climate models are essential tools for a more complete understanding of climate and impacts. Climate models are the only means of integrating our knowledge of the components (atmosphere, ocean, land surface, and sea ice) that make up the complex climate system. And they are the only means for carrying out "experiments" on the climate system to study the projected changes and impacts of different scenarios. In order to be useful to regional planners, climate models must make credible predictions on a regional spatial scale (e.g., within a state). Because of the coarse resolution and other limitations of today's climate models, predictions are considered reliable only averaged over continental and larger scales, but not on a regional scale. In order to make reliable and useful region-scale predictions, climate models need greatly increased spatial resolution, improved treatments of subgrid-scale physical phenomena (e.g., clouds), and inclusion of additional physical, chemical, and biogeochemical processes, such as the chlorophyll concentrations in Fig. 3.

The United Nations-sponsored Intergovernmental Panel on Climate Change (IPCC) is a highly regarded multinational scientific body that performs extensive studies of potential climate change and publishes their findings on a five-year cycle. The IPCC is beginning to collect scientific results for the Fourth Assessment, to be completed in 2007. The Community Climate System Model (CCSM2) has been developed as a multi-agency initiative with support from the National Science Foundation and the Office of Science in the U.S. Department of Energy. CCSM2 will be one of the primary climate models used in the next IPCC assessment. With an increase in dedicated computing resources in the 100 Tflop/s range, new studies could be performed with a higher resolution atmospheric model providing much improved spatial detail.

An increase of storage and processing capability by a factor of 3, at the current resolution, would allow the addition of dynamic vegetation to a fine-scale land model. Another factor of 2 would provide enough power to routinely include troposphere chemistry. Ocean biogeochemistry could be included in the coupled model for an additional factor of 3 to 5. Interactive carbon in a full carbon cycle will require a further factor of 2. Extending the atmospheric model to include a full stratosphere and increasing the vertical resolution requires another factor of 5 increase in capability. These additional physical mechanisms may not be fully exploited unless the ocean and atmospheric horizontal resolution is substantially increased. An eddy-resolving ocean model would require another factor of 1200. In the ten-year time frame, it will be important to include cloud-resolving atmospheric simulations in a fully coupled earth-system-modeling framework. The cumulative requirement supporting these developments is estimated to be a factor of nearly four orders of magnitude in aggregate flops. To accomplish such runs in today's turnaround times requires petaflops computing. A more extensive discussion of these issues is given below.

Beyond better prediction of the actual climate, putting reliable, efficient simulators into the hands of geophysicists and environmental scientists would allow exploration of "what if" scenarios, such as the effect of aerosols, of massive landcover changes, or of the collapse of the thermohaline circulation.

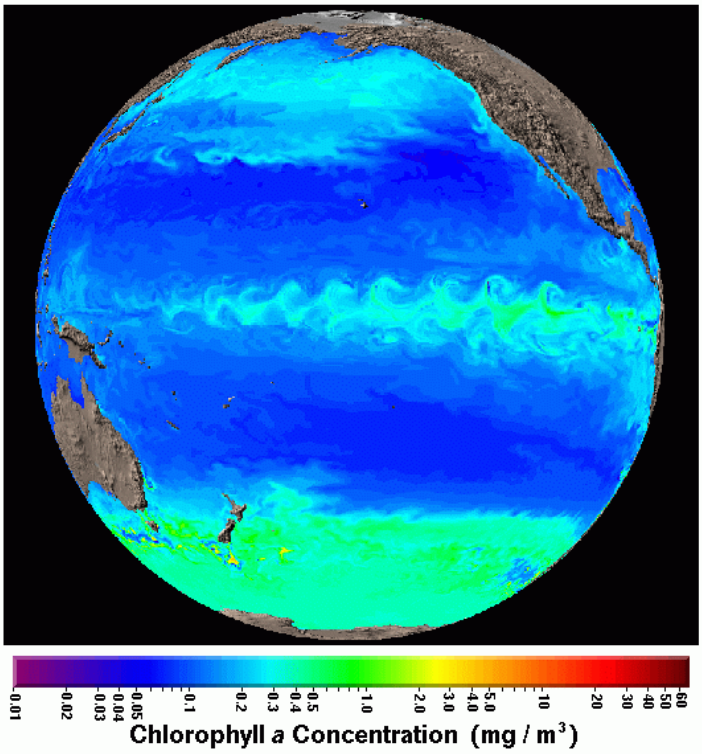


Figure 3. Color-simulated chlorophyll distributions in the Pacific Ocean using the LANL Parallel Ocean Program. Ocean biogeochemistry includes phytoplankton growth in response to the upwelling of nutrients.

Research Issues

The question of the extent and significance of natural climate variability on the decadal to century time scales and our ability to accurately predict future climate states must be periodically revisited and answered with the best scientific grounding possible. The great advances that have been made in understanding and modeling the individual components of the climate system, the atmosphere, the ocean, the ice, and land are now being applied to understand feedbacks in an earth system model that couples all of the individual components. What causes the various climate oscillations—the El Nino, the Pacific decadal oscillation, and the North Atlantic oscillation—and how these interact with each other and anthropogenic factors are a matter of active research. These issues have bearing on the sensitivity of the climate system to switches between stable states and on the frequency of extreme weather events such as droughts and catastrophic storms. The predictability of climate depends in large measure on the ability of models to capture and

faithfully reproduce the balance of physical processes in what mathematicians refer to as a dynamical system.

Research issues exist in every facet of climate modeling, ranging from physical process parameterizations, to submodels, to the fully coupled model. Within each of these facets, research issues exist with respect to (sub)model validation by comparison with regional or global measurements and observations. It is often beneficial to reformulate some portion of a (sub)model. This reformulation involves research to find a better way to represent a given process or system. Examples are different "dynamical cores" used in atmospheric component models, such as spectral transform, semi-Lagrange, and finite volume. Each is a different approximation to the same physical equations, and each uses different techniques to solve the equations. The suitability of these methods for reproducing the nearly two-dimensional turbulence in the atmosphere and the conservation of chemical species advected by the winds continues as an active area, with significant overlap with research in computational fluid dynamics and numerical methods.

Furthermore, research continues on methods by which model codes can be structured and written so that the codes can be easily ported onto a variety of computer architectures without sacrificing performance. This software engineering effort is particularly challenging today because the return of vector-based architectures to the domestic market has forced code developers to try to cope with the somewhat opposite poles of vector-based and cache-based programming styles.

A most important issue for simulation relates to the data that feeds it: the United States must lead the world in establishing a permanent climate observing system capable of sustaining observations and measurements of the real climate system over decades to centuries. Comprehensive datasets are needed to compare with model predictions (model validation), increase our understanding of the complicated nonlinear behavior of the climate system, and provide a basis for development of better models.

Technology Barriers

A number of barriers to progress in climate modeling are linked directly with the computer hardware. The performance of today's computational climate models suffers from inadequate memory bandwidth and high memory latency that cannot be masked by multiple levels of cache, thereby making it difficult to achieve more than 10% of peak performance of the processor. Performance is dramatically improved on computers with high memory bandwidth.

High-latency interconnection networks are also a bottleneck. The poor performance in interconnection networks has limited the cost-effective scaling of climate models to merely hundreds of processors and constrained the algorithms and coding styles that scientists use to make progress.

Other features of the computer architecture are also important. Climate modeling codes perform better on computers with faster, tightly integrated processors (SMP compute nodes and vector processors). Since the timestep size decreases with increasing resolution for the prevailing explicit, operator-split climate model integrators, faster processors are needed to maintain the same rate of simulation throughput and scientific productivity (e.g., simulation years per processor day).

The growth of computer capability and capacity must take place in a balanced fashion for the climate community to obtain maximum benefit from increased investments in computer hardware. In addition, emphasis must be placed on making dedicated resources available for the long simulations and concentrated studies required to support climate change applications. The climate community has found it difficult to obtain the resources and throughput needed for long simulations at existing computer centers, whose mission is to serve large numbers of users.

Other barriers to progress are associated with the quantity of data that must be handled in climate simulations. As the data accumulates from coupled simulations run for centuries, it has become difficult to analyze by using current tools. The distributed archive and analysis centers are making good progress in assembling the software tools needed for such analyses, but there is a need for high-performance networks and high-performance switches linking the centers. As the community begins to use data assimilation techniques for climate studies, the bandwidth required to exchange data will grow dramatically. Some of the most significant advancements in climate simulations have used new algorithms to improve accuracy and to achieve greater throughput. One area of promising research is the iterative solution of implicit methods, which are attractive because they may allow larger time steps. Currently, however, these methods suffer computationally because they require global reduction operations on each iteration, operations that result in serious performance degradation on machines with high-latency interconnects.

The same considerations apply to spectral transform methods used in the atmospheric simulations, because the transforms are global in extent. Fast transform methods, such as the FFT, still yield superior operation counts in many solution algorithms but may scale poorly on massively parallel machines and at high resolution. New methods based on icosahedral meshes and smoothly mapped meshes show promise for better scalability.

Fundamental research into new mathematical methods for the simulation of complex, interacting fluid flows are also needed to advance climate simulation capabilities over the next decade. Specific topics of interest include semi-implicit and operator split methods to allow long time integrations in the presence of fast moving gravity waves, Lagrangian vertical coordinate systems and conservative remapping schemes that allow accurate thermodynamic simulation of transport and moist processes, and fast methods for solution of elliptic systems.

Also of considerable interest are software engineering practices that allow a community of hundreds of computational climate scientists, mathematicians, computer scientists, and numerical analysts to develop and maintain community codes. It is difficult to write codes that will run effectively across the wide range of today’s computer architectures and that will be extensible to tomorrow’s model upgrades. Compiler technology is not

keeping pace with high-performance scientific computing demands. Parallelism constructs that are stable and robust are also sorely needed. As the memory hierarchies deepen and more architecture specific layers are added, modelers require more support in dealing with software issues. A significant effort is required, and more support needed, to adapt to new architectures and maintain the pace of scientific development

Resources Required

Increased computational resources could significantly improve climate system simulations. Efforts to model biogeochemical cycles and their relationship to climate require development of comprehensive coupled models on a scale unprecedented in any computational science discipline. To develop models sufficient for addressing these questions and issues will require the following changes to the current state of the art:

Increase the spatial resolution of the grids of the coupled model components. At present, the "standard" atmospheric simulation uses a grid with 300 km horizontal resolution and 26 vertical levels. For the ocean, the corresponding resolutions today are 100 km in the horizontal and 40 levels in the vertical. The resolution targets are about 10 km in both the atmosphere and ocean, for different reasons. It has been demonstrated that 10 km resolution is needed to resolve oceanic mesoscale eddies. A similar resolution is needed in the atmospheric component to obtain predictions of surface temperature and precipitation in sufficient detail to analyze the regional and local implications of climate change.

Increase the completeness of the coupled model by adding to each component model important, interactive physical, chemical and biological processes that heretofore have been omitted because of their computational complexity. Inclusion of atmospheric chemistry, both tropospheric and stratospheric, and biogeochemistry in the ocean is essential if scientists are to understand the ecological implications of climate change.

Increase the fidelity of the model. We need to replace parameterizations of subgrid physical processes by more realistic and accurate treatments as our understanding of the underlying physical processes improves, often as the result of observations field programs such as the DOE Atmospheric Radiation Measurement Program.

Increase the length of both control and climate-change-scenario runs. Longer control runs will reveal any tendency for the coupled model to drift and will also improve our estimates of model variability. Longer climate change scenario runs will permit examination of critical issues such as the potential collapse of the global thermohaline circulation that may occur on time scales of centuries in global warming scenarios.

Increase the number of simulations in each ensemble of control runs or climate-change-scenario runs. *Increase the number of climate-change scenarios investigated.* These issues are both examples of perfectly parallel extensions of present-day simulations: each instance of another scenario or ensemble member is completely independent of every other instance. Ensemble members are distinguished by small perturbations in their initial conditions, which are quickly amplified by the nonlinearity of the equations. The use of

ensembles provides an important measure of the range of variability of the climate system. These are summarized in Table 1.

Table 1: Compute factors for addressing improvements to climate models.

Issue	*Motivation*	*Compute Factor*
Spatial resolution	Provide regional details	10^3-10^5
Model completeness	Add "new" science	10^2
New parameterizations	Upgrade to "better" science	10^2
Run length	Long-term implications	10^2
Ensembles, scenarios	Range of model variability	10
Total Compute Factor		10^{10}-10^{12}

We emphasize that these compute factors are relative to the capability of computer systems typically used now for coarse-resolution (300 km atmosphere, 100 km ocean) simulations.

In addition to the performance requirements listed above, storage requirements would also increase for each of the possible paths identified. The storage requirements for sample atmosphere and ocean resolutions are shown in Table 2. The table also shows the storage required for each additional tracer in an atmospheric chemistry or ocean biogeochemistry simulation. A low-resolution coupled model run for a century produces almost 1 TB of output. In a high-resolution configuration, a single century run would produce 23 TB.

Table 2: Storage requirements for atmosphere and ocean resolutions.

Component	**Resolution**	**History output (GB/sim-yr)**	**Each tracer (GB/sim-yr)**
Atmosphere	T42 (300 km)	7.5	0.02
Atmosphere	T85 (150 km)	29	0.08
Atmosphere	T170 (75 km)	110	0.3
Ocean	1° (100 km)	1.7	0.2
Ocean	0.1° (10 km)	120	17

We cannot rely solely on increases in computer power if we are to attain our goals in a timely way. A significant role will be played by scientists finding better ways to model the climate and by continued vigilance in validating model predictions against the best observational records available. History has shown that development of new algorithms and numerical methods can reduce computing requirements by orders of magnitude. Similarly, new ways of formulating models and approximating physical, biological, and chemical processes can substantially reduce cost or improved fidelity of the models. Such improvements are harder to forecast than the technological improvements in computer hardware, like that embodied in Moore's law, but when they do occur they enable major breakthroughs in understanding and predictive capability.

The research objectives for climate science require a diversified investment in environmental science disciplines as well as crosscutting, enabling technologies. The development of multidisciplinary teams, such as those funded in the DOE SciDAC program, have been successful in bringing resources to bear on specific problems and providing timely solutions to difficult problems.

For coupled atmosphere-ocean model simulations with 1° (100 km) ocean/ice resolution and T85 (150 km) atmosphere/land resolution, we achieve 3.5 simulated years per processor-day on 192 processors of an IBM SP4 and 16 simulated years per processor-day on 176 processors of the Earth Simulator (with some components not fully optimized). For ocean-only simulations at eddy-resolving 0.1° (10 km) resolution, we achieve 0.12 simulated years per processor-day on 500 processors of an IBM SP3 and 3.6 simulated years on 640 processors of the Earth Simulator. These latter two simulations run at 7% and 30% of peak, respectively. These percentages are typical of the 5–10% of peak performance climate models achieve on cache-based machines and 30–40% of peak on vector-based machines.

The scaling of computational complexity of the climate simulations we would like to reach can be estimated straightforwardly with today's primarily explicit codes. Cost goes up as the square of the inverse of the spatial mesh parameter and linearly in the number of vertical levels, the number of passive tracers, and the duration of the integration.

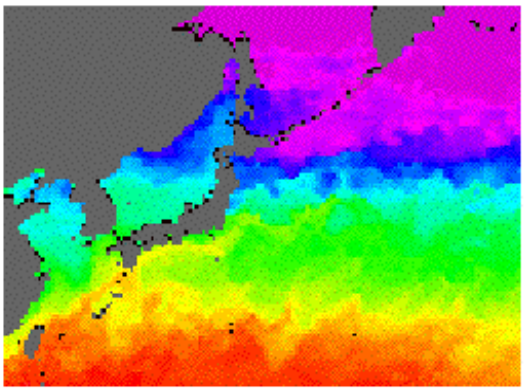
The ocean model at 1° and the atmospheric model at T42 each scale well in fixed-size (strong scaling) parallel sense up to 256 processors of an IBM SP. With these relatively small grids, scaling to thousands of processors is not achievable. For future problem sizes of 0.1° and T170 or better, scaling beyond 1,000 processors seems achievable. The scaling of high-resolution atmospheric models is not well understood, however, because model parameters must be changed, requiring substantial runs at new resolutions to determine their best values.

Metrics of Success

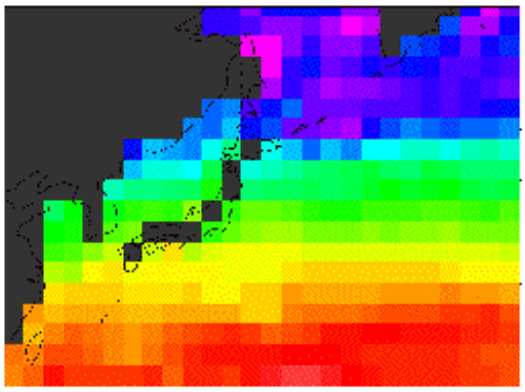
The objective of high-end climate modeling is increased scientific understanding of the climate system and the possible effects of human -induced climate change. Science is the driver so we pose metrics in terms of scientific productivity and not processing rates. The climate community expresses simulation throughput in simulated years per wall-clock day. It has been a goal to achieve overnight turnaround of a “standard” 20-year atmospheric run in support model development. This is roughly equivalent to 15,000 times real time. Even for the standard low-resolution runs we do not today achieve much better than one thousand times real time, as described in the previous section. These metrics assume that the science drives the resolution and complexity requirements. The U.S. climate modeling community is currently behind in its science goals because it falls far short of achieving the needed throughput rate with high-resolution models. Of course, the greater the simulation throughput, the more data there is to be processed and analyzed. Data analysis often takes longer than data generation, in wall-clock time.

The ultimate measure of success is the U.S. climate community’s ability to support climate change simulations of scenarios suggested by the IPCC Assessment and the U.S. National Assessment of Climate Change. Adequate computing capacity and networking infrastructure are crucial to meeting these strategic commitments and to credibly influencing global policy making bodies.

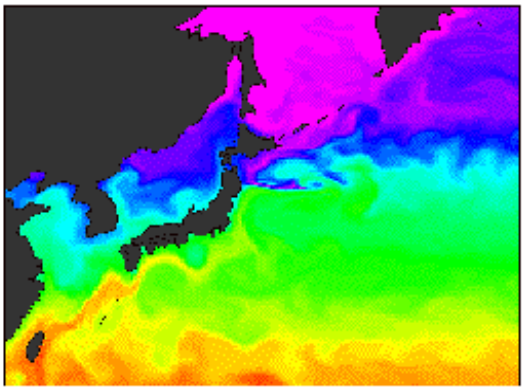
a



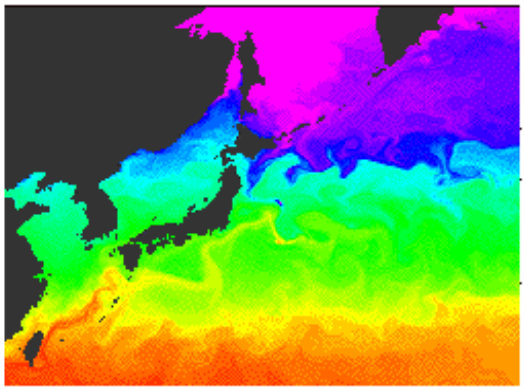
b



c



d



Combustion Science: Enabling our Energy Future

Combustion is the process by which fuel and oxidizer react chemically to generate heat and mechanical work. Combustion is described in terms of reacting flow processes that include the combined effects of chemical reactions, fluid dynamics, and transport. Many important classes of combustion problems also involve multiphase behavior such as the dynamics of fuel sprays, particulates, and chemical reactions at surfaces. In some problems, thermal radiation plays a critical role. The science of combustion seeks to gain a predictive understanding of these processes and their interactions.

Impact on Science and Society

Although combustion is one of our oldest scientific endeavors, it remains a critical mission area for the Department of Energy. Combustion of fossil fuels continues to provide over 85% of the energy required for transportation and stationary power generation. Combustion is also responsible for most of the anthropogenic pollution in the environment. Carbon dioxide and soot resulting from combustion are major factors in the global carbon cycle and climate change. Soot, NOx, and other emissions have important consequences for both the environment and human health. In addition, related reacting flow processes play a central role in a broad range of science and technology applications including the synthesis of nanomaterials, chemical vapor deposition for creating semiconductor devices, and reforming and catalytic processes needed for new fuel cell systems. A predictive understanding of combustion processes will play a key role in providing efficient, clean, and sustainable energy for our nation's future as well as establishing a solid scientific basis for building new materials and designing new chemical processes.

Figure 1. Two of the laboratory turbulent jet flames that are part of the International Workshop on the Measurement and Computation of Turbulent Nonpremixed Flames. http://www.ca.sandia.gov/TNF/

Scientific Opportunities

In spite of the fundamental scientific and technological importance of combustion, our knowledge of combustion processes is surprisingly incomplete. Much of the difficulty in combustion science results from the complex interaction of turbulent flow processes with the myriad chemical processes occurring in a flame. New diagnostic techniques developed over the

past two decades years have given us quantitative and detailed measures of the structure of many combustion processes. They are, however, still far from quantifying the full range of species involved in complex chemical reactions, and far from resolving the finest spatial structures characteristic of interfacial and high-pressure processes. Computational implementation of theory and models, coupled with experiments such as depicted in Figure 1, has enabled great progress in our understanding of idealized, and aspects of more complex, combustion processes, but have been unable to directly explore the full complexity of realistic and important regimes of combustion.

For a broad range of problems in combustion science, this is about to change. With the recent advances in computer hardware and new algorithm technologies depicted in Figure 2, combustion scientists are beginning to perform the first predictive direct simulations of realistic complex flames using detailed representations of the chemistry and transport.

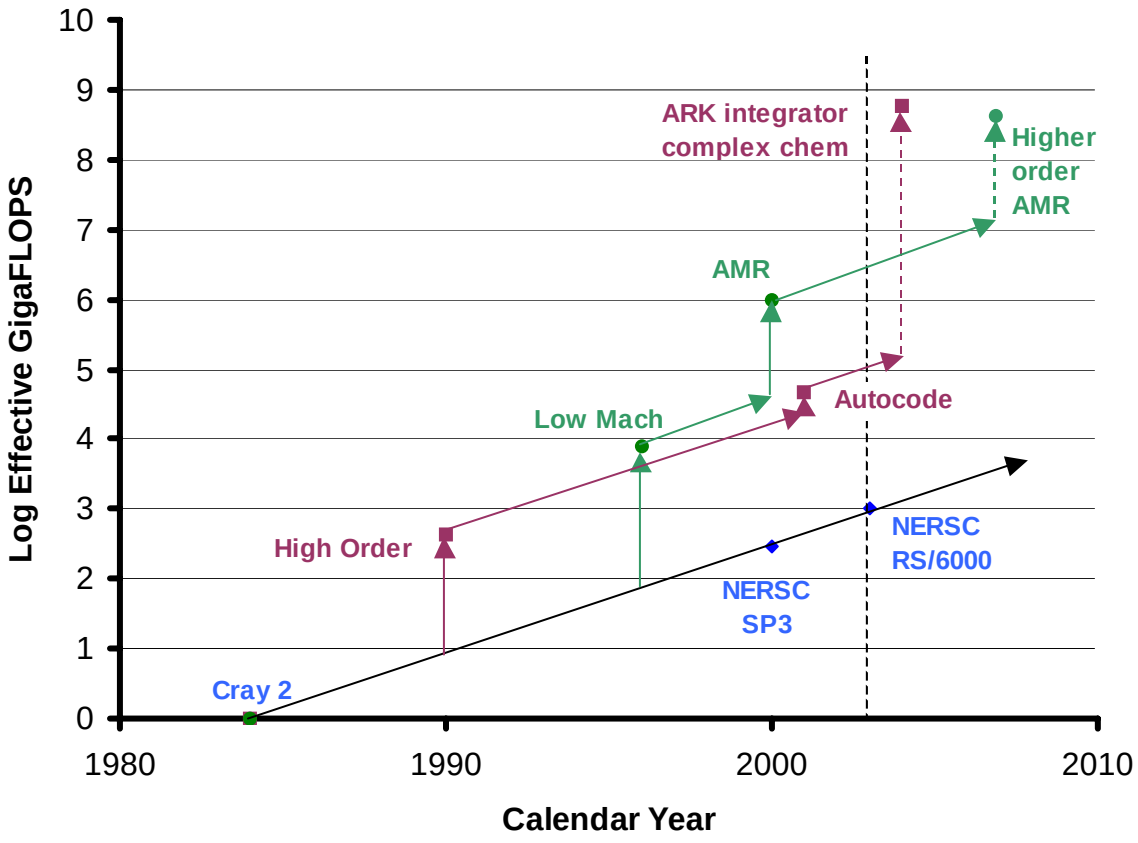


Figure 2. Turbulent gas-phase combustion simulations are modeled using either the compressible reacting flow equations or a low Mach number model. In both formulations, the equations include models for fluid dynamics, chemical kinetics and transport. Algorithmic advances for both types of simulations continue to yield enhancements to our computational capability comparable to, if not

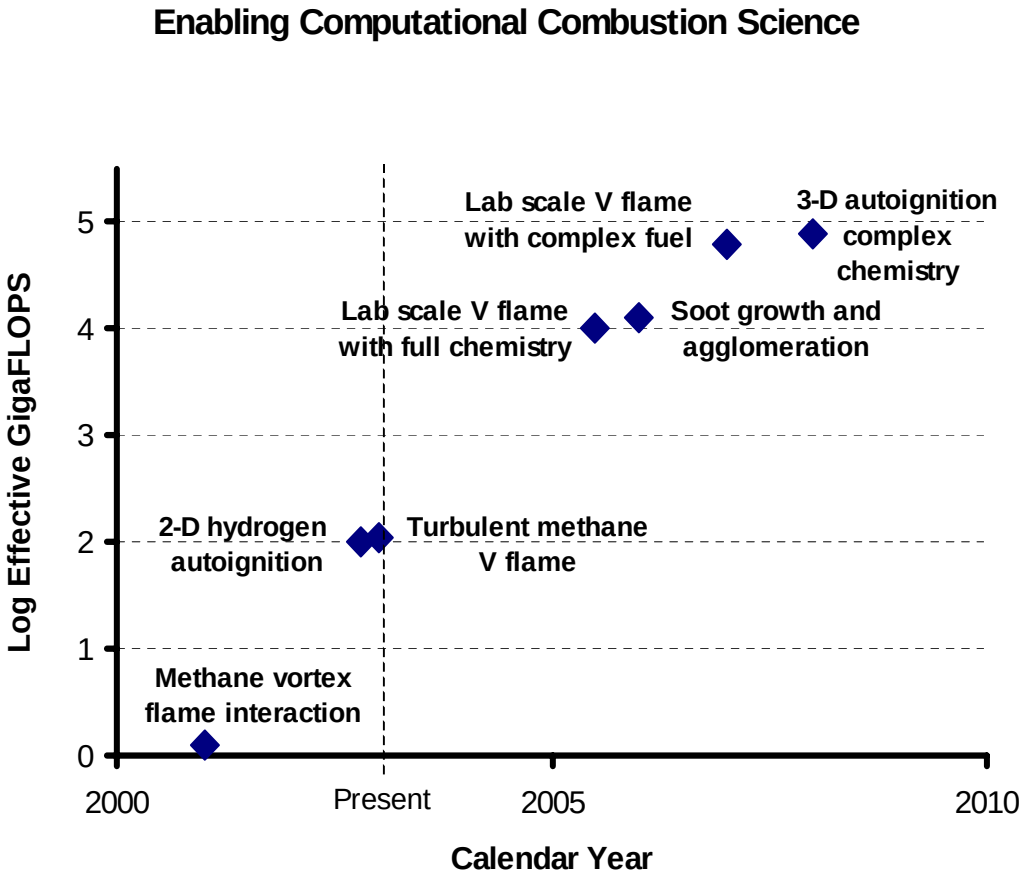


Figure 3. Combustion simulations have reached the point where we can explore the detailed chemical and fluid-dynamical aspects of important combustion processes. With additional computing power we will be able to address many long-standing combustion issues including turbulence / chemistry interactions, autoignition, and soot formation.

With moderate extensions of the rapidly evolving state-of-the-art in these technologies, long-standing problems in combustion science can be solved. A few examples of the combustion topics that can be addressed as computational power increases are illustrated in Figure 3 and are further described below.

One area that is well primed to exploit increased computing resources is the exploration of fundamental turbulence-chemistry interactions in laboratory-scale, atmospheric pressure flames. A recent computation of a turbulent premixed methane-air 'V' flame is shown in Figure 4. While this computation exceeded the resources normally accessible, a hundred-fold increase in available computational power will enable routine detailed simulations of turbulent natural gas combustion in laboratory-scale flames. For the first time, researchers will be able to probe the detailed dynamical and chemical properties of these types of flames over the full range of length scales observed. They will be able to quantify how turbulence alters the chemical pathways in the flame and how chemistry affects the turbulent flame speed. With an additional order of magnitude in compute power and continued algorithmic advances, they will be able to predict the pollutant emissions from such flames, understand how the presence of larger hydrocarbons affects the flame chemistry, and quantify pressure effects on flame dynamics. They will also be

able to investigate the chemical behavior and emissions characteristics of turbulent jet diffusion flames, such as those pictured in Figure 1, where mixing plays a dominant role in the dynamics.

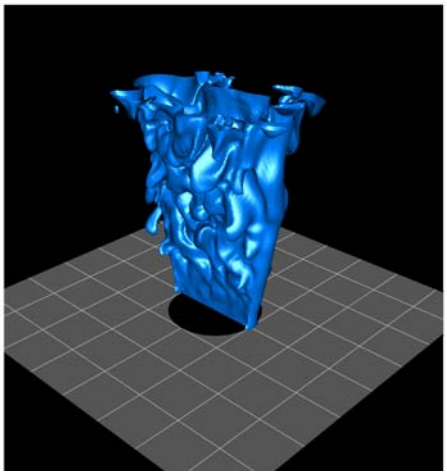


Figure 4. Instantaneous flame surface from numerical simulation of a turbulent premixed methane flame.

Another important example of how increased computing power would enable answering key questions in combustion science is the simulation of autoignition in a high-pressure turbulent environment. The interaction of autoignition and turbulence is a transient process that initiates at very small scales and depends on the details of the chemistry. This process is presently intractable for theory and eludes our finest measurement capability, but is beginning to yield to investigations using large-scale direct simulations. For example, results from a recent two-dimensional study of autoignition at higher pressure are illustrated in the first accompanying sidebar. These simulations were limited to two dimensions and simple hydrogen-air chemistry, but a hundred-fold increase in computer power will enable these simulations to be extended to realistic fuels such as n-heptane in two dimensions and simple fuels in three dimensions. With a thousand-fold increase and anticipated algorithmic advancements, it will begin to be possible to simulate realistic fuels at high pressure in three dimensions.

Increases in compute power will also open up new avenues for exploring multiphase flow issues arising in combustion. There are many such opportunities, including particle formation, spray breakup and combustion, catalytic combustion, and combustion in super-critical flows. Recent algorithmic advances, for example, have produced tools that are just beginning to track the growth of a complex hydrocarbon molecule into a soot particle by reactions within an enveloping soup of combustion products. The impact these approaches can have on the fundamental questions surrounding soot birth and grow are described in more detail in the sidebar. A hundred-fold increase in computational capacity will allow us to track the chemistry of polycyclic aromatic hydrocarbons, capture the early stages of soot growth, and understand the detailed properties of early soot particles. With further algorithmic advances and a thousand-fold increase in hardware capability, we will be able to gain enough information to develop predictive science-based models of the growth and oxidation of soot particles that can be used in simulations of combustion processes. This, combined with other anticipated advances, would provide the scientific basis to evaluate and optimize new combustion processes and fuels to mitigate particulate emissions and the associated heath hazards.

These examples illustrate how high-performance computing is now enabling combustion scientists to make contact with turbulent laboratory-scale combustion systems, and is extending their reach beyond the limits of current measurement capabilities and theories to new fundamental understanding. There are many other types of simulations that can contribute needed understanding for problems involving sprays, particles, and reactive surfaces. The continued rapid evolution of computational technologies provides an opportunity to remove the barriers to understanding very complex, yet important, combustion problems. Indeed, this is an opportunity to embark on a new era of combustion science that can have a profound and urgently needed impact on our nation's progress to a clean, sustainable, and economical energy future.

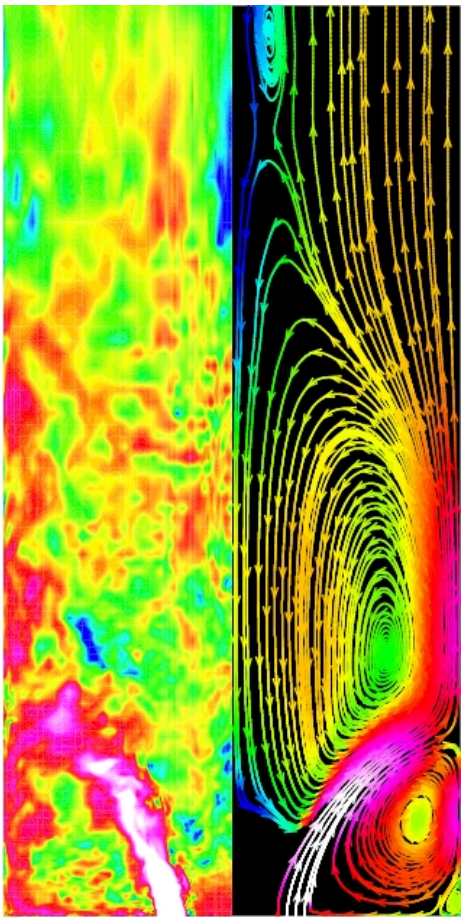


Figure 5. Fully coupled Large Eddy Simulation of hydrogen-enriched lean-premixed combustion in a swirling flow burner for comparison with laboratory experiments. The instantaneous computed velocity field is shown on the left, and the average flow field on the right.

Research Issues

Clearly, the opportunities provided by advances in computational technology must be accompanied by progress in other research areas to assure the envisioned progress in combustion science. One major research issue that is intimately linked to these research opportunities is the generation of mechanisms to describe the chemical reactions occurring in combustion systems. As we explore reacting flow phenomena with increasing fidelity there will be a critical need to develop and validate new chemical mechanisms of increasing complexity and accuracy. We must also be able to find suitable reduced descriptions of the kinetics, with known chemical fidelity, derived from comprehensive chemical mechanisms. Developing tools for automating this process will be a crucial element of combustion research and will require close interaction with the chemistry community. Similar supporting research will also be required to obtain transport data, thermodynamic data, and other information needed to model multi-component mixtures precisely.

While chemistry is central to combustion processes, there are other processes for which substantial research is needed to yield the required predictive models. These include models of spray breakup and mixing, radiation properties and interactions with chemistry, soot and other particles, and models of reactive interfaces such as those found in catalytic combustion and exhaust treatment. Many of these multi-physics processes

are also important in other fields, and will benefit from a broad collaborative and interdisciplinary initiative in computational research.

The software for performing combustion simulations represents only a portion of the suite of tools needed for computational combustion science to reap the benefits of a dramatic increase in computing power. New analysis paradigms must be developed and implemented to elucidate the basic underlying processes and validate new reduced models. Combustion simulations can involve hundreds to thousands of dependent variables that are coupled in highly nonlinear ways. Extracting the relationships between these variables, determining the sensitivity of solutions to parameters, and exploring reduced descriptions of the chemical kinetics requires sophisticated new mathematical tools. Of particular importance in this area are tools such as computational singular perturbation, proper orthogonal decomposition, uncertainty quantification, sensitivity analysis and other techniques that focus on addressing the complexity of coupled transport-chemical processes in the context of multi-dimensional reacting flow simulations.

Our ability to perform simulations of realistic flames with detailed chemistry creates exciting new research opportunities for developing models to study the large-scale behavior of combustion systems. In particular, detailed simulations can be used to establish baselines for understanding of how fundamental combustion processes vary across the many length scales present in turbulent flames at realistic conditions. If we can quantify such scaling behavior and encapsulate the dominant modes of turbulent reacting flows in lower dimensional models, we will have the tools needed to predict larger scale dynamics with known fidelity. Approaches based on Large Eddy Simulation (LES), an emerging multi-scale approach that takes advantage of breakthroughs in predictive models at the smallest scales, are just now allowing us to tackle such problems. For example we are beginning to apply LES models to examine the effects that mixing hydrogen into the fuel has on the flame stability and nitrous oxide emissions. An example of such a simulation is illustrated in Fig. 5. The development of these types of models will provide the scientific underpinnings that can revolutionize the design of new combustion systems.

Technology Barriers

Advances in combustion simulation face a number of technological barriers. Although simulation methodologies are available for many of the computational problems we have posed, additional development is required to harness the power of both existing and new computer architectures for these problems. One such area is the need for research in applied mathematics to extend low Mach number models to include adaptive approaches for closed chambers and techniques for including long wavelength acoustic effects. Another critical area of research is scalable algorithms for multi-physics reacting flow problems. Particular issues in this area include the development of scalable solver techniques for variable coefficient and nonlinear implicit systems and the development of

improved load-balancing strategies for heterogeneous physics. Substantial increases in capability can also be achieved by developing improved discretization procedures that not only provide improved representations of the basic physical processes but also improve the coupling between these processes.

Another issue facing combustion science is that of managing software complexity. The simplest simulations are multi-physics algorithms that incorporate fluid mechanics, chemical kinetics, and transport. More complex problems will also require algorithms that treat particles, radiation, and/or multi-phase processes and interfaces. Adaptive mesh and multi-scale methodologies are often required to solve problems with the necessary fidelity. These challenges are further complicated by the desire to rapidly implement new multi-physics and computational algorithms among a collaborative team of scientists. The possibility that new software frameworks can facilitate this for complex combustion codes is being explored in the current SciDAC program.

Some of the sophisticated data analysis tools described in the above section are actually most efficiently implemented as algorithms in the simulation code itself. The complexity of the integration of such new algorithms in such a way that they interact properly with the other solver technologies can be a barrier to implementation. For example, the validation of reduced models often requires a statistically relevant number of similar, repeated simulations. Integrating uncertainty quantification approaches into the simulation allows potentially more efficient interrogation of parameter dependencies and model certainties. Barriers to data analysis are also found in the data-management issues associated with multi-terabyte datasets. Managing this complexity and making the resulting tools available to collaborations and the larger community is a high priority in combustion science.

Combustion science and the supporting simulations rely on a very diverse set of chemical inputs and models, and also produce new data and models. New data informatics approaches, such as those now being pursued in the DOE National Collaboratory Program, are needed to provide for the rapid collaborative development and exchange of chemical mechanisms, thermodynamics data, validated model descriptions, and annotated experimental data that will support the computational studies.

Resources Required

Developing the software methodologies needed to solve the major outstanding questions in combustion science posed above will require several multidisciplinary research teams. Experience in the SciDAC program suggests that each team should consist of a minimum of ten people split amongst combustion scientists, applied mathematicians and computer scientists. The commonality of the issues in combustion science with other application areas suggest that approximately half of the personnel on a software development team can perform most effectively if embedded in enabling technology centers analogous to the SciDAC integrated software infrastructure centers. While this core team might persist for five or more years, others may be considered close collaborators working for shorter periods on specific combustion science issues. These scientists (including

graduate students and postdocs) will specify simulation requirements, analyze results, and will generally collaborate on the development of new code capabilities and analysis tools. Finally, this broad collaboration of computational scientists must be supported by facilities that enable the maintenance of software capabilities, support of new collaborations, sharing of critical data and results, and integration of new state-of-the-art approaches.

Additionally, expansion of programs in experimental and theoretical aspects of chemical science will be required to take advantage of a significant increment in available computational capabilities. The results from successful large-scale scientific simulations will create new opportunities for validation and discovery from experiments and for new insights and modeling approaches from theory. Furthermore, interdisciplinary teams must be formed to accomplish parallel advances in the underlying chemical science challenges, including the development of kinetic and thermo-chemical data, transport processes, and chemical mechanisms.

Based on computational experience gained under SciDAC we can fairly reliably estimate the computer resources needed for combustion applications. Turbulent reacting flow computations that resolve the detailed structure of a premixed flame will require approximately 3×10^{16} aggregate flops and 8 terabytes of memory. Such a simulation will generate 25 terabytes of data that will need to be archived for several months while the data is analyzed. For autoignition problems with simple fuels in three dimensions or complex fuels in two dimensions the required aggregate flops and volume of generated data will be comparable but the resident memory size will be somewhat smaller. Computational estimates for multiphase problems are typically somewhat higher. For example, molecular modeling of soot particle growth and agglomeration for 1 microsecond of evolution for a 0.5 micron agglomeration will require approximately twice the computational resources of turbulent gas-phase combustion problems.

For an illustration of scaling laws in computational combustion, consider the low Mach number adaptive mesh code used to generate Figure 4. Current production runs employ 128-1024 processors of the NERSC SP3. Parallel scaling is good to 128 processors, but deteriorates in the 128-1024 range. This deterioration is primarily attributable to poor scaling of sparse unstructured linear system solvers on large numbers of processors, which require global communication. Parallel scaling would improve if communication were improved relative to the processor speed. The observed scaling law for overall computational complexity would be $O(N^4)$ for an explicit method on a uniform grid, where N is the reciprocal of the mesh interval and a coefficient reciprocal in the Mach number. For the semi-implicit adaptive meshing employed, scaling is observed to be between $O(N^3)$ and $O(N^4)$. At current problem sizes, the adaptive mesh refinement algorithm reduces overall operation count by a factor of 10 to 100, depending upon the Mach number, independent of the speedup factor obtained from its parallelization. Higher order spatial discretizations under development could probably save an additional order of magnitude. Current computations for methane-air flames employ a rather minimalist representation of carbon chemistry. A more comprehensive carbon chemistry mechanism would increase computational requirements by a factor of three. Including nitrogen chemistry would increase the time for the minimalist mechanism by a factor of

eight. Changing the fuel from methane to the much more complex propane mechanism would increase requirements by a factor of 24. Changing the fuel to heptane (a surrogate for diesel fuel) would increase requirements by a factor of about 360. All of these estimates assume a relatively low turbulence intensity. More realistic levels of fluid mechanical turbulence could boost requirements by an additional three orders of magnitude. An ultimate simulation would also incorporate additional multiphase and pressure effects whose complexity is difficult to estimate from current experience.

Metrics of Success

Some combustion simulations focus on detailed comparisons with experimental data. For these types of problems the appropriate metric for success is the ability to predict properties of flames and validate these predictions against experiments with a given level of fidelity. For laboratory-scale turbulent premixed methane-air flames computational combustion scientists have already been able to predict flame morphology and propagation properties. The next milestone in this type of study, achievable with a hundred-fold increase in compute power, would be predicting the distribution of chemical species such as OH and CH in a flame that can be measured using planar laser induced fluorescence. Further milestones at the thousand-fold level would be predicting the level of emissions of pollutants such as NOx, quantifying the chemical pathways involved in the burning of fuels with larger hydrocarbons, and exploring modifications to flame dynamics at high-pressure.

Other computational simulations will focus on exploring regimes that are not accessible to detailed experimental measurements, such as autoignition at very high pressure. For these simulations, careful verification of spatial resolution requirements and scaling behavior of algorithms must be introduced to ensure the fidelity of the simulations. Then comparisons with the limited detailed data and more global experimental data (such as the pressure-time history of autoignition in a closed chamber) will provide opportunities for validation and discovery of new phenomena. A major metric of success will be the demonstration of accurate simulations at high pressure with detailed chemistry models in three-dimensions. With a thousand-fold increase in computer power, highly scalable algorithms, and new approaches for handling complex chemistry, such a simulation would enable detailed exploration and the discovery of fundamental new information about modes of flame propagation and the potential for combustion control.

For problems of soot formation, an early metric for success would be successful molecular simulation of a 500-atom system for 10 microseconds. With a thousand-fold increase in computing resources our goal would be to simulate systems up to 10nm in size for 1 millisecond. The successful comparison of derived physical and optical properties with available experiments and simulations of laminar and unsteady sooting flames will provide opportunities for discovery and validation of new soot models and would be the first step in understanding how to minimize soot formation.

In all cases, the ultimate measure of scientific success is the production of new, validated, and predictive understanding of here-to-fore unresolved complex combustion problems.

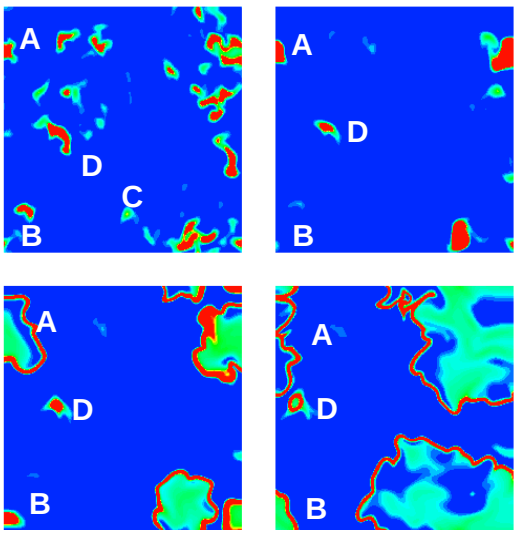
The implementation of reduced models in simulation approaches such as LES will also directly facilitate transfer of this new knowledge to the creation of new design capabilities that will allow combustion science to successfully impact the development of new combustion technologies.

Sidebar #1

Autoignition and Control of 'Flameless' Combustion

Autoignition is the process that lights a combustible mixture by the application of heat, but without a flame or spark. But how does autoignition progress in fluctuating and incompletely mixed gases, and how we might control the process? At stake are novel approaches to high efficiency, low emission combustion technologies. Experiments demonstrate that autoignition can initiate combustion in a gas mixture that will not burn in the usual mode where fuel is consumed in a thin high-temperature sheet of flame. An important implication of such 'flameless' combustion is that the chemical energy is released at lower peak temperatures, affecting the fundamental competition between the production of NOx species (enhanced at high temperatures) and the complete oxidation of hydrocarbon species.

Our present understanding of autoignition is primarily from experimental data and simulations limited to zero or one-dimensional studies. Most of this work assumes perfectly mixed gases with no spatial variations. Scientists exploring new approaches to efficient energy conversion need, however, detailed information concerning the modulation of the relevant ignition chemical kinetics by the fluctuating local mixing environment that leads to localized regions of chemical activity or 'hot spots'. Ultimately, it is the aggregate behavior of temporally and spatially evolving 'hot spots' that determines the overall ignition timing and fuel consumption rate. Such a detailed characterization of autoignition is well beyond what we can hope to measure with sophisticated laser-based experiments, or predict using current theories. But large-scale computer simulations offer the opportunity to study these complex phenomena in great detail.



Islands of hydroperoxyl (HO_2), an early indicator of autoignition, are shown forming in 2-D simulations of a fluctuating hydrogen-air mixture (in time from top left to lower right). Note that kernel A grows most rapidly, B and D more slowly, and that C is extinguished by the

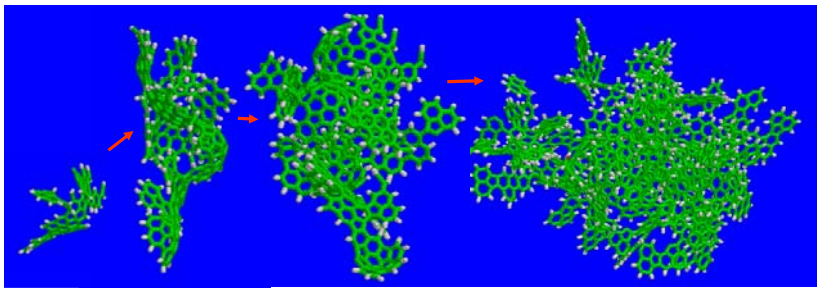
Recently, two-dimensional Direct Numerical Simulations (DNS) of autoignition in mixtures of hydrogen and air have found new chemical pathways that are observed in such fluctuating environments. The resulting detailed data (see the Figure) provided a basis for a new approach to describing transient

autoignition in terms of relevant flow and thermochemical parameters. A more realistic series of three-dimensional runs to study n-heptane (a surrogate for more complex hydrocarbon fuels) autoignition at the high pressures of interest will require about 1000 times the current computational capability. This simulation would provide the first full characterization of a realistic autoignition process, including its topologies and propagation dynamics. The data would provide new fundamental understanding of the effects of mixing on the dynamics of autoignition, thus giving us the keys to innovative 'flameless' combustion technologies as well as other important ignition phenomena.

Sidebar #2

How Do Soot Particles Develop and Grow?

Predictive models of soot formation and oxidation that provide detailed chemical structures of the particles currently do not exist, a fact that greatly limits our ability to control this important chemical process. The fundamental difficulty in developing high fidelity models is the inherent "meso" character of soot. Models of soot formation are most sensitive to events in the earliest stages and therefore, hinge on understanding fundamental transitions from molecular to particle length and time scales. Its "meso" character requires the development of new multiscale techniques in theory. It is this multiscale character of high fidelity soot simulations that will be enabled by enhanced computational resources.



The first step in soot formation is thought to be the construction of a sub nanometer polycyclic aromatic hydrocarbon (PAH) seed as depicted in the figure. The most computationally intensive component of the calculations of this step, namely the generation of the forces between reacting species, scales for chemical accuracy as N^7 or higher for N atoms. Current simulations are only feasible because they employ less accurate methods (with N^{2-3} scaling) and rudimentary searches of coordinate space. Calculations of sufficient accuracy to replace measurements for this step in soot formation will require computer resources 100 times those currently available and a commensurate improvement in theoretical and algorithmic methods.

Soot begins to acquire its characteristic properties as it grows to nanometer particles sizes (about 1000 molecules) under conditions where the molecular properties of the species that condensed into the particle are largely lost. New combined Monte Carlo and molecular dynamics methods have been developed as viable tools to predict the detailed evolution of particles and surfaces subject to reactive flow environments. These methods tend to scale linearly with the size of the molecular system involved, but are very

sensitive to the accuracy of parameters from detailed molecular computations. Understanding details of the internal structure of soot, that are important for optical properties, free radicals, etc., requires understanding of intramolecular reactions and mesoscale rearrangements that can be addressed by Accelerated Molecular Dynamics methods. The size of the systems, the sensitivity to the parameters, and the number of statistical samples needed to provide realistic mechanisms for fluid flow simulations will require capacity computing at multi-teraflop levels.

For the several orders of magnitude larger sample sizes of still longer trajectories that also include oxidation chemistry that are required for investigating a set of combustion conditions, algorithmic advances and 1000 times the current computational resources will be required. The anticipated detailed soot model cannot be incorporated into computational fluid dynamics simulations without reduction techniques that simplify the model to that appropriate for conditions at each grid point in the simulation. The resulting combustion simulations will require computers at 100x the current level. However, such simulations will be able for the first time to treat soot emissions with the same fidelity as the much more chemically simple NOx emissions, allowing for the first time accurate estimations of the NOx/soot trade-off that dominates the operation point of many combustion devices.

The transition from molecular scale to mesoscale information required in soot modeling is characteristic of many frontier areas in chemistry and physics, for example, nanoscience, catalysis, climate change and environmental remediation. The application of large-scale computational resources (in the Tflop/s to Pflop/s range) to these areas will lead to synergistic advances across broad areas of the physical sciences.

Design of Materials: the Road to Technological Innovation

Materials science is concerned with the discovery of new materials and the understanding, control, and exploitation of their properties. The results of past materials research permeate our everyday lives, from the chips in the computer on which this text was written to the structural and magnetic materials used in generation of the electricity that powers it. At the most basic level materials science asks the simple question "How do we take the ninety or so elements that comprise the periodic table and put them together in combinations that produce materials with useful properties?

Traditionally the search for new materials and the refinement of existing ones has been accomplished by *Edisonian* trial and error, guided by simple models and the skill and intuition of countless experimenters. Today, however, new materials are increasingly assembled atom by atom or involve previously unimagined complexity; their properties are probed by billion dollar experimental facilities (Advanced Light Source, Spallation Neutron Source) capable of revealing microscopic detail. In addition accurate, robust simulations that are founded in the fundamental equations appropriate to the real material and utilizing the computational power of new generations of high performance computers now have an unprecedented impact on the development of new materials and devices.

Scale	**Quantum**	**Nanoscopic**	**Mesoscopic**	**Macroscopic**
Length (m)	10^{-11} — 10^{-8}	10^{-9} — 10^{-6}	10^{-6} — 10^{-3}	$> 10^{-3}$
Time (s)	10^{-16} — 10^{-12}	10^{-13} — 10^{-10}	10^{-10} — 10^{-6}	$> 10^{-6}$

Table 1. **Scales spanned by materials science.**

A fundamental problem faced by much materials research is that the properties of real materials depend on phenomena that occur at different length and time scales (Table 1). At the smallest scale, properties are determined by the "electron glue" that holds the atoms together (bonding or cohesion). This is the domain of quantum physics. At the macroscopic level, many materials properties – strength, fracture, magnetism – are as much influenced by microstructure – crystallites or grains within the material – as the intrinsic bonding of the atoms of the ideal crystal. Between these length scales is the world of nanoscience ($1 – 100 \times 10^{-9}$ m), where materials often display new or unusual properties that hold exciting possibilities for future scientific discovery and technological innovation.

Describing each of these extremes and more importantly bridging the disparate length and time scales associated with them (multiscale modeling) poses the grand challenge of

theoretical and computational materials science. Making progress in addressing these challenges promises to revolutionize the design and development of new materials.

Impact on science and society

Advanced materials drive economic, social, and scientific progress, shape our everyday lives and play a crucial, enabling role, in virtually all technologies. Indeed the current information age is built on the twin foundations of semiconductor processor and magnetic storage technologies developed over the last forty years. The exponential growth rate in both processing power and storage density has been made possible through exploitation and control of materials properties on ever smaller length scales and increasing complexity. Structural materials that are stronger, lighter, retain their strength at higher temperatures, or adsorb energy when deformed, enable more efficient energy production as well as more efficient and safer automobile and airline transportation.

Currently, storage capacity (areal density or Gbits/in^2) of magnetic disc drives is doubling ever year (Fig. 1). This phenomenal rate of increase– up from the already impressive 60% per year in the early 90's and 30% per year prior to that – was facilitated by the introduction of Giant Magneto-Resistance (GMR) read heads and was the result of a scientific discovery made less than ten years previously. Impressive as these advances are, they cannot continue for more than a few years without significant new scientific breakthroughs because the individual storage elements will be so small as to be unstable (super-paramagnetic limit) and of no use for long-term storage of information.

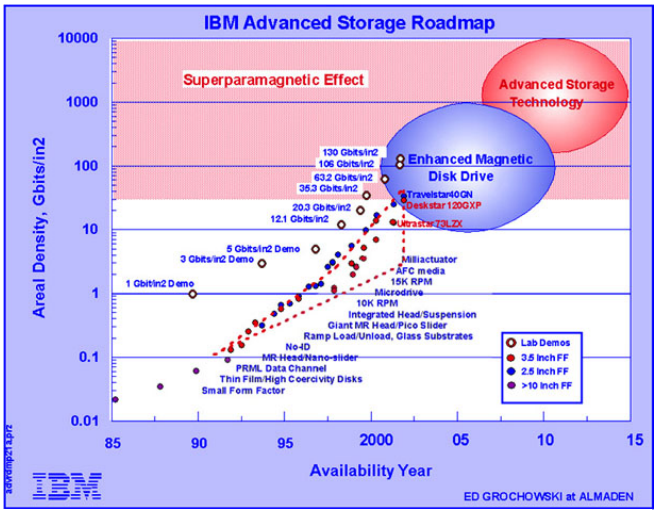


Figure 1. Advanced storage roadmap showing the annual increase in areal density of magnetic disc drives over the last two decades and future projections.

As cast, the ordered inter-metallic compound, Ni_3Al, is brittle. However recent scientific discoveries involving addition of small amounts of boron, slight modification of the Ni:Al ratio, and control of microstructure, has resulted in a new class of commercial

alloys that are ductile, strong at high temperature, and corrosion resistant. These alloys are now resulting in substantial energy and cost savings in the steel, automotive, and chemical industries (Fig. 2). In 2001, the development of these alloys was listed as one DOE Basic Energy Sciences' 100 most significant scientific advances of the previous 23 years.

Figure 2. Nickel-Aluminide alloys in action: forms (top left) for automobile engine part casting (top right); heat exchanger (bottom left); rolling-mill rollers for steel-sheet manufacture (bottom right).

In numerous other areas of materials science the basis for future scientific breakthroughs is being laid – understanding the origins of high temperature superconductivity, transition metal oxides with totally new properties and functionality, and the exploration of the fascinating world of nanostructured materials.

Scientific Opportunities

During the next two decades the opportunity exists to develop a new paradigm for materials research in which modeling and simulation are integrated with synthesis and characterization to accelerate discovery. During the last two decades, application of first principles quantum theories of the electronic structure of materials, coupled with simulations using idealized models, has resulted in a revolution in the understanding of many simple systems – ideal crystals and alloys, surfaces, and localized defects. Future development of multiscale modeling capabilities will allow the study of microstructure and its influence on strength and fracture, as well as the synthesis and processing routes required to control microstructure. Significantly increasing the size and complexity of systems that can be studied at the quantum level can make it possible to solve fundamental problems not currently accessible to theoretical description – dynamics of electron spin, strong electron correlations, and high temperature superconductivity. In addition theory and modeling can be used to take maximum advantage of experiments

performed at the nations advanced characterization facilities through direct calculation and simulation of that which is measured.

Material Optimization for Energy and Transportation

Structural materials from nickel-based superalloys used for turbine blades to lightweight aluminum alloys used for automotive parts are pillars of the energy and transportation industry. Without exception, these materials are compartmented on a micron scale by boundaries of complex shapes that divide spatial regions of different composition and/or different crystallographic orientation – broadly called the "microstructure". A materials microstructure controls most of its structural properties – strength, wear, and corrosion resistance. Predicting how the microstructure emerges from an initially structureless melt during solidification (casting, welding, etc), and how it evolves during post-solidification processing, is an essential prerequisite for material optimization and is one of today's most important theoretical computational challenges.

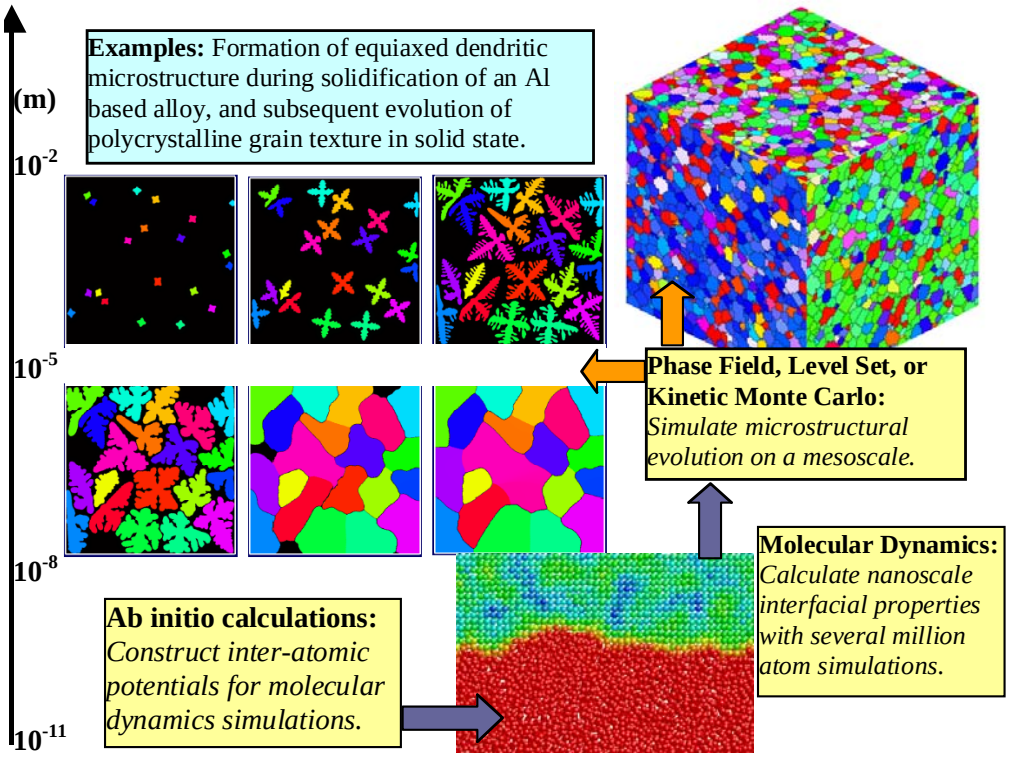


Figure 3. Multiscale integration of computational approaches used to predict microstructures of structural materials from the atom up.

The core of this challenge is the accurate prediction of how phase and grain boundaries move in response to driving forces such as temperature, concentration, or stress. This problem is intrinsically multiscale because the two key anisotropic properties that control

this motion, the interface energy and mobility, are determined by details of inter-atomic (quantum physics) forces acting on nanometer/picosecond length/time scales. Whereas the highly nonlocal fields that determine the local driving force for motion are determined by bulk transport of mass or energy on macroscopic length and time scales. Furthermore, the vast parameter space that characterizes the interface anisotropy (*e.g.*, five dimensional for grain boundaries in three dimensions!) approaches biological complexity.

Progress in solving this multiscale problem has recently been accomplished through the integration of atomistic scale simulations and mesoscale models (Fig. 3). Quantum mechanical ab-initio simulations have been used to guide the construction of inter-atomic potentials that can be used in large molecular dynamics (MD) simulations with several million atoms. These simulations, in turn, have made it possible to predict, for the first time, the anisotropy of the interface energy and mobility. Moreover, new mesoscale simulation methods such as phase-field and level set have emerged that incorporate these interfacial properties and thermodynamic data to simulate complex microstructures, which appear nearly indistinguishable from experimental micrographs.

This integration of new techniques holds much promise to guide the optimization of microstructures so as to cut down the current 10-15 years required to commercialize a new material to just a few years, and, even more ambitiously, to guide the search of new materials. However, realizing this promise still requires extension these techniques to multi-component alloys (*e.g.*, 12 components for super alloys), to experimentally relevant length and time scales, and to three dimensions. A two- to three-order of magnitude increase in computing power will provide a unique opportunity to achieve these goals by, for example, extending large MD simulations to predict interface mobility over the full range of driving force relevant for microstructural evolution and by enable three-dimensional mesoscale simulations to reach the large system sizes relevant for materials processing. A key target is to model a cubic millimeter of material where the predictions of mesoscale models can be meaningfully interfaced with macroscale industrial codes.

Magnets of the Future: Predictive Modeling of Switching and Hysteresis

Predictive modeling of the technical properties of magnets – energy product, coercivity, remenance – which requires modeling of the dynamics of magnetic moments and how these are reversed or switched – is the central scientific challenge in magnetic materials. It is also one where computational approaches can prove decisive thereby having a profound impact on a wide range of technologies from energy production and utilization (generators, transformers, and motors) to transportation (sensors and motors) and computers (magnetic storage and memory).

While the underlying mechanism for materials magnetism involves electronic interactions at the atomic level, long range, magnetostatic, interactions and large-scale features (*e.g.*, domain walls and their interaction with microstructure) are crucial for determining bulk magnetic properties in real materials. Consequently, magnetism is an intrinsically multiscale problem. A problem that is, however, greatly simplified by the observation that the basic equation describing the dynamics of magnetic moments at the

different length scales is believed to have the same form, the Landau-Lifshitz-Gilbert (LLG) equation (Fig. 4). Albeit that the description of the magnetic moments changes from length scale to length scale – first principles electronic structure methods at the smallest length scales, spin models at intermediate length scales, and continuum micromagnetics models with empirical parameters at the device level. Thus, new challenge/opportunity is to develop rigorous approaches to extending and bridging the models that describe the different length scales and to apply these capabilities to discover and design new magnetic materials.

A hundred-fold increase in computer power would allow the exploration and understanding of the structure of domain walls and their interaction with and pinning by defects – key steps towards understanding magnetization reversal. A thousand-fold increase would enable detailed parameter free modeling of the dynamics of magnetization reversal in small magnetic bits. Further advances coupled with concomitant advances in microstructure modeling could lead to science based modeling of nanomagnets.

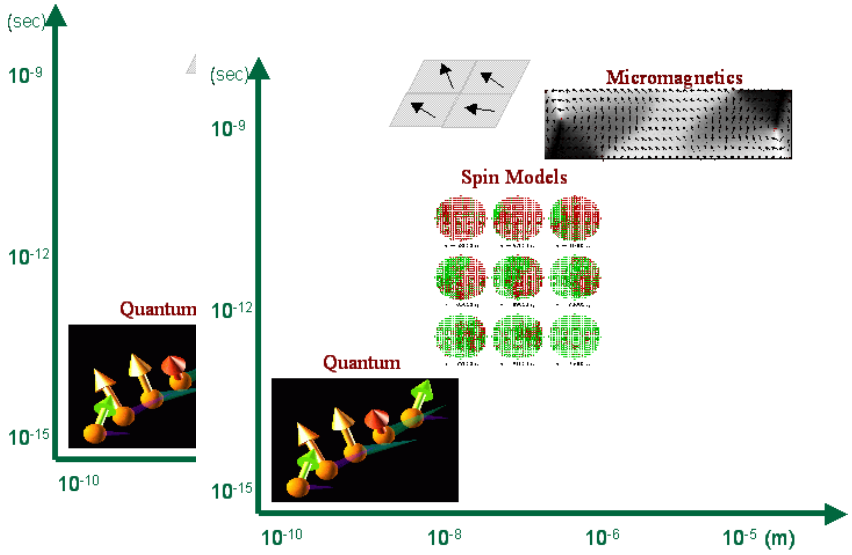


Figure 4. Illustration of the methods used magnetic materials modeling at different length and time scales: atomic scale – first principles; nanoscale – extended Heisenberg spin model; and micron/device scale – micromagnetics. Ideally each scale needs to be enhanced by 1-3 orders of magnitude.

Advancing Modeling of the Fundamental Interactions in Complex Systems

Increases in computing power have a very large impact on first-principles quantum simulation methods to predict structural and electronic properties of complex materials. First-principles simulations are extremely computationally demanding but are essential to understand the properties of complex materials in detail. Electronic properties are modeled using various first-principles methods depending on the accuracy needed.

The Quantum Monte-Carlo (QMC) method is the most accurate and expensive, while Density Functional Theory (DFT) has been so widely used to model electronic properties in the past decades that it has been called the "Standard Model of Condensed Matter". Because it involves independent statistical sampling QMC is uniquely suited to take advantage of future generations of computers, readily utilizing parallel computation on machines with tens of thousands of nodes. Recently an O(*N*) algorithm has been developed and applied to the prediction of the optical gap in seminar conductor nanostructures consisting of a thousand atoms. One of the most successful methods developed in the past fifteen years is First Principles Molecular Dynamics (FPMD) due to Car and Parrinello, which unifies molecular dynamics and DFT. FPMD is an example of a very powerful simulation tool whose development was accelerated by the large computing power brought about by the first Cray vector computers in the 1980's.

New levels of computational power coupled with concomitant advances in theory, algorithms, and software engineering, will vastly expand the domain of applicability of first principles methods making them applicable to spintronics, super-hard materials, catalytic reactions, and hosts of other applications, as well as expanding the role of first principles modeling as the foundation upon which multiscale modeling is built.

Research Issues

Research challenges can be broken into three broad classes. First, developing and extending the length and time scales covered by the models used at each scale. Second, coupling models across different scales to produce robust and predictive multi-scale modeling capabilities. Third, formal theoretical advances to allow modeling and simulation to address many outstanding problems – formal theory of spin dynamics, origins of pairing in high T_C superconductors, and so forth.

Extending models can be achieved through improvements in algorithms (*e.g.*, changing from algorithms that scale as N^3 to one that scales linearly in N, where N is the number of atoms in the simulation), better use of computational resources, and parallelization. Larger length scales can generally be achieved through parallelization and domain decomposition. Here, a major goal would be the development of QMC and DFT electronic structure methods that scale linearly with system size to 10,000 to 100,000 processors.

Research into extending the time scale is an overarching need at all length scales and is one of the most challenging problems in materials science. Here parallel computers have no obvious advantage since time is intrinsically serial. However, advances can have a profound impact on the exploration of new physical phenomena (*e.g.*, growth mechanisms, rare events).

Although development of seamlessly coupled multiscale methods is a *Holy Grail* of materials science, lack of computational resources is generally not the limiting factor, although large simulations are often necessary to validate multiscale models. Addressing this area necessitates researchers with expertise in many different fields, building the

teams of materials scientists, mathematicians, and computer scientists will require major changes in the way materials research is traditionally performed. Achieving reliable and robust techniques for coupling/mapping ab initio electronic structure with/onto atomistic molecular dynamics or Monte Carlo simulations and integration of mesoscale models with existing thermodynamic databases for quantitative modeling of multi-component alloy would be major steps towards the overall goal of multiscale modeling.

Technology Barriers

The diversity of applications in Computational Materials Science makes it difficult to make general statements about the current technological barriers to research. The following are general concepts that are perceived to be barriers by members of the community.

Hardware Barriers

While there is a consensus that a large increase in computing power is desirable, the way in which this increased power should be realized is less clear – a large number of moderately powerful processors versus a moderate number of very powerful processors. From an application development standpoint, it is generally preferable to deal with fewer, more powerful processors. However, since it is easier/cheaper to build a supercomputer by assembling a large number of moderately powerful processors, it is important to assess the usability of such a computer in the context of Materials Science simulations. Two realistic examples are 1) a 100 Tflop/s computer built from 100,000 processors of 1Gflop/s each, or 2) a 100 Tflop/s computer built from 10,000 processors of 10 Gflop/s each. It should be noted that for both 1) and 2), the number of processors far exceeds that of currently available computers, so that our conclusions are, to some extent, speculative and are further complicated when (unknown) considerations of bandwidth and latency of the interconnect are taken into account.

For DFT/FPMD — which has $O(N^3)$ complexity — it is reasonable to expect scaling to 10,000 processors within one to two years given adequate software development investments, while scaling to 100,000 processors is a longer term goal. For QMC, classical MD, and continuum models of $O(N)$ complexity, the situation is more favorable since these methods can maintain a reasonable communication/computation ratio by increasing the size of the system studied and therefore the amount of work performed by each processor – so called weak scaling.

In general it is expected that the cost advantage of using a finer granularity (i.e., a large number of small processors) may be offset by the increased cost of application software development.

Software Barriers

Massively parallel scalability. Scalability of some applications to a few thousand CPUs has been demonstrated (see Fig. 5). However it is important to note that this scaling is typically only obtained after considerable investment in software development and that

efforts made to obtain scaling to 2000 CPUs may not be reusable when targeting 10,000 CPUs.

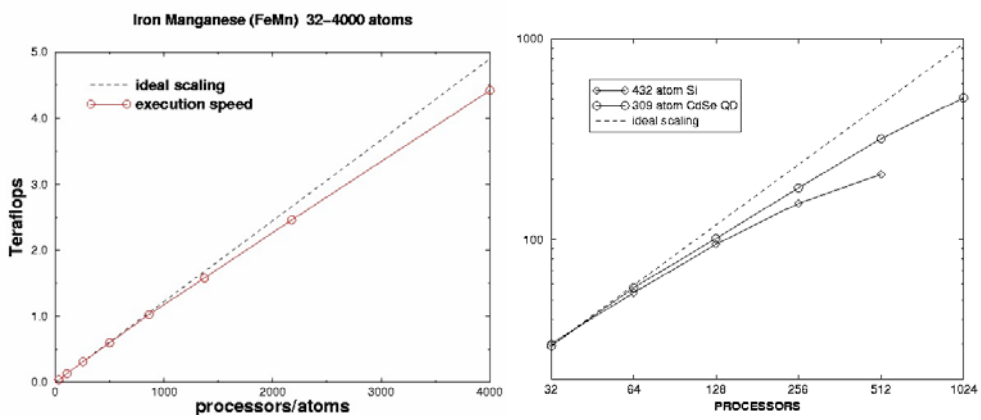


Figure 5. Scaling behavior of the first principles electronic codes LSMS (left) and PARATEC (right).

It is generally difficult to *predict* scalability of an algorithm for processor counts beyond currently available since it often involves trial and error and unpleasant surprises (*e.g.*, lack of scalability of MPI collect operations on IBM SP3's beyond 1024 tasks). Consequently, improved performance models are an important priority since they would facilitate the development of high-performance application software before a new platform is built.

System reliability. Ideally, system reliability on 100,000-processor platforms should be dealt with by the operating system. Failing this, most (likely all) materials applications will require additional software development to address fault tolerance, given that long runs (days or weeks of wall-clock time) are the norm.

Support of libraries. The availability of communications (e.g. MPI) and mathematical (*e.g.*, ScaLAPACK) libraries is an important ingredient in the development of scalable application codes.

Software engineering issues. Materials simulation codes must often be rapidly modified to address ever changing physical models. The cost effectiveness of good software engineering and design practices is slowly being recognized in the community, together with the fact that simulation software typically has a much longer lifetime than most hardware platforms. The cost of maintaining, rewriting or modernizing legacy codes remains an obstacle to research, since this activity is often not recognized as research and thus not funded as such. Several groups have started efforts aiming at improving code reuse within groups, and ultimately throughout the Materials Science community.

Algorithm Barriers

Some simulation methods are naturally suited to parallel computing. QMC is currently only limited by access to sufficient computational resources, and relies on an algorithm that scales extremely well to very large numbers of processors (*i.e.*, is "embarrassingly parallel"). Classical molecular dynamics and most methods based on continuum models are also well positioned to exploit future large platforms using domain decomposition.

Simulation methods relying on more complex, O(N^2) or O(N^3) algorithms would benefit greatly from larger computing power, although with a less spectacular increase in the length scales that can be described. For instance, DFT simulations, an O(N^3) algorithm, of 256 atoms are routinely carried out on 0.5-1.0 Tflop/s computers. An eight-fold increase in the number of atoms, *i.e.*, a two-fold increase in linear dimension, would require a 512-fold increase in computing power, *i.e.*, a 256-512 Tflop/s platform. Furthermore, larger systems usually involve longer simulation and equilibration times, which would further increase the size of computer required. This shows that algorithmic developments that reduce the computational complexity of DFT to O(N) or O(Nlog$N$) are a priority, and must be considered as important as the construction of larger supercomputers.

In addition to the above overarching algorithmic considerations, advances in specific lower level mathematical algorithms would benefit materials applications generally. Particularly import are scalable algorithms for large complex matrices that are either dense or sparse with a know sparsity pattern and portable adaptive meshing and multigrid methods for interface tracking, phase-field and level set.

Resources Required

The resources required fall naturally fall into two categories. Firstly, state of the art capability and capacity computing. Secondly, people – materials, applied mathematics, and computer science researchers – to support software development and maintenance of methods and software used in cutting edge research.

Computational Resources

Present estimates of annual high performance computing resources used at DOE facilities by materials science is approximately 2.7 Tflop/s-years. Historically, approximately 18% of the computer time available at NERSC is utilized by materials science projects. With current NERSC hardware the annual usage is about 1.8 Tflop/s-years. Additional materials projects are serviced by the Center for Computational Sciences (CCS) at ORNL (about 0.9 Tflop/s-years annually). Materials scientists are also major users at the NSF supported Pittsburgh Supercomputer Center (1.2 Tflop/s-years), giving an overall "materials" usage of 3.9 Tflop/s-years at the three centers. Various scaling laws, from linear to cubic in the number of atoms or electronic orbitals, from whence to extrapolate from this base, are given in under "Algorithmic Barriers" above.

We illustrate with the Locally Self-consistent Multiple Scattering (LSMS) electronic structure code whose measured performance on up to 4,000 processors of the Alpha Cluster Lemieux at Pittsburgh is given in the left panel of Figure 5. LSMS runs at approximately 75% of peak. For large systems, hundreds to thousands of timesteps are required to achieve the ground state configuration. Each timestep typically requires 3-8 self-consistent field (SCF) iterations to achieve consistent forcing fields. Overall wall clock time is linearly proportional to the number of timesteps, the number of SCF iterations, and the time of each SCF iteration. (The latter scales linearly in the number of atoms, and we allocate one atom per processor, scaling up the problem in a weak sense, so the time for an SCF iteration is independent of the problem size.) Whereas runs of 1,000 atoms for 1,000 timesteps are now routine, and fit within an 8-hour turnaround window, we would like to regularly run systems of 10,000 atoms for 10,000 timesteps. Given a machine one hundred times more powerful, consisting of 10,000 processors, each of which is 10 times faster than current processors of approximately 1.5 Gflop/s peak (*e.g.*, at NERSC), and given a fixed percentage of peak of the current algorithm on the faster processors, the expanded run should complete within the same turnaround window. These estimates do not rely on any algorithmic improvement. While it is difficult to contemplate complexity scaling better than today's linear scaling, if history is a guide, we may expect further algorithmic reductions in the coefficient coming from the number of SCF iterations and timesteps.

Human Resources

In order to exploit fully the capability of high performance computing it is necessary to adopt a new approach to accessing and utilizing high-end computational resources. This is necessitated by a number of generic characteristics of computational materials science. Most important is the recognition that, in terms of overall advances in performance – algorithmic efficiency – gains arising from the intellect and ingenuity of the researcher are larger than those from improved hardware, impressive though Moore's Law is (see Fig. 6). When one adds to this, the multiscale nature of the materials science, the lack a single computer code, or even a small set of codes, which could then be used by the whole community, the need to rapidly respond to the discovery of new phenomena, and continuously tune codes to the most advanced computer architecture, it is clear that a community wide response is needed.

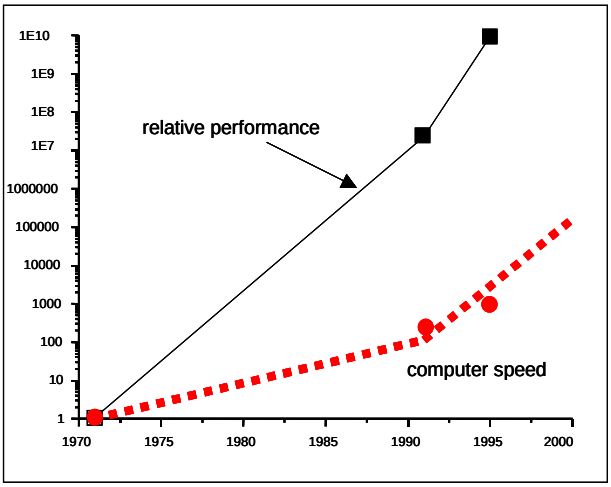


Figure 6. Relative performance increase of Ising model simulations (squares) compared the normalized speed of the computers (circles) the simulations were executed on. The dashed line is a schematic of the increase in peak performance of the fastest supercomputers since 1972.

Metrics of Success

A measure of success common to all areas of Materials Science is a reduction in the time spent between the discovery of a new phenomenon and its use in a technological application. A reduction of half to a third of the time of the current years to commercialize discoveries through computing appears to be a distinct possibility.

Ultimately, the combination of large computational resources, cutting-edge software, and numerical methods will realize the goal of predicting materials properties accurately without recourse to experimental input, and possibly discovering new phenomena and materials by numerical simulation.

Finally, direct simulation of experimental quantities can be used to take maximum advantage of experiments performed at the nation's advanced characterization facilities, thereby greatly reducing the demand on these expensive facilities.

Nanoscience: Building a Better World one Atom at a Time

Seeking to understand the natural world atom-by-atom, nanoscience has been defined as research at the atomic, molecular or macromolecular levels, at length scales of approximately 1-100 nanometers[3]. A nanometer, equal to one billionth of a meter, is approximately equal to the length of three water molecules laid end to end; a human hair is about 100,000 nanometers in width. The goal of nanoscale research is to provide a fundamental understanding of phenomena and materials at the nanoscale and to create and use structures, devices and systems that have novel properties and functions because of their small size.

Impact on Science and Society

The promise of nanoscience – and the related field of nanotechnology, which exploits advances in nanoscience to create fundamentally new and useful systems – has resulted in the National Nanotechnology Initiative (NNI) becoming one of the U.S. federal government's highest priorities for funding, now approaching $1 billion annually. Nanotechnology is expected to be a $1 trillion/year business by 2015. Nanoscience and nanotechnology are leading to unprecedented understanding and control over the fundamental building blocks of all things physical, chemical, and biological. This is likely to change the way almost everything – from computers to drug delivery systems to consumer products to objects not yet imagined – is designed and made.

One particularly relevant example of nanoscience and nanotechnology is the sub-field of molecular electronics, in which the goal is the development of self-assembled chemical computers – computers made up of billions of individual molecules, each of which plays the role of a circuit element in today's integrated circuits (*i.e.*, gates, switches, memory, *etc*.). Instead of lithographically etching these circuit elements in silicon, as is done today, self-assembled chemical computers will be created by dipping substrates into mixtures of chemicals. If the mixture is correctly formulated and precisely controlled, wires and switches will chemically assemble themselves onto the substrate from these materials. As lithography-based methods of preparing silicon-based integrated circuits are expected to hit fundamental barriers about a decade from now, molecular electronics is the most likely route to computing in the post-silicon era.

Nature is the ultimate nanotechnologist, *e.g.*, a human being is made up of a relatively small number of rather mundane chemicals, self-assembled into a nanostructured material of incredible functionality; bacteria such as *E. coli* move by rotating flagella powered by what looks very much like a nanoscale electric motor, whose power output is the equivalent in a human being of being able to swim through rapidly setting cement; biological materials are capable of self-healing, sensing, growth, and adaptation. Many

[3] This is essentially the definition adopted in the National Nanotechnology Initiative, http://www.nano.gov.

nanoscience and nanotechnology research directions involve biomimetics – creating materials, structures and systems that exhibit the properties of biological systems.

Nanoscience and nanotechnology are enabled by two fundamental scientific advances: the advent of experimental capabilities to image and manipulate materials at the single molecule level, and the extraordinary growth in computational and theoretical capabilities for predicting and understanding phenomena at the nanoscale. Enormous investments are underway by the DOE in experimental capabilities for nanoscience, such as the $1.4 billion Spallation Neutron Source at Oak Ridge National; Laboratory and the experimental equipment going into the new DOE nanoscience centers. A concomitant investment in computational and theoretical nanoscience is crucial to explore new nanomaterials and processes, to aid in interpretation of experimental measurements, to design new experimental instruments, to develop targets for new materials, and to enable the transition to large-scale production. In the race to explore and exploit nanoscience, the U.S. is facing stiff competition from Japan: Japan is conceded to be ahead of the U.S. in nano-synthesis, and is spending more annually on nanoscience than the U.S.[4] With the advent of new instruments, the U.S. has the edge in characterization. The U.S. is on a par and perhaps ahead in computational nanoscience; however, that situation is now threatened, since, as we will see below, access to the most capable computers in the world is essential for computational nanoscience.

Scientific Opportunities

Because nanoscience and nanotechnology involve manipulation at the molecular level, variables come into play that are not relevant to macroscale bulk materials. For example, the properties of a nanostructured material can depend strongly on the number of atoms making up the nanostructure. Quantum effects can be very important or even dominant. Small changes in the conditions under which self-assembly is performed can change radically the final product. Because of the unique features of nanomaterials, the results of many nanoscale experiments cannot be understood in the absence of theory. The large-scale manufacturability of nanostructured devices will require an extraordinarily detailed and predictive understanding of how the manufacturing conditions impact the desired product. As a result, theory, modeling and simulation (TMS) have long been recognized as playing a fundamentally important role in nanoscience and nanotechnology, and this is reflected in the prominent role given to TMS in the early planning of the NNI.[5]

Computational nanoscience – the large-scale computational solution of theoretically derived equations to perform simulations of the structure and/or dynamics of nanostructures and devices – is the crucial unifying element in TMS. Computational nanoscience enables experiments to be understood, properties of nanostructures to be

[4] http://www.nanotechfoundation.org/worldwide.html

[5] D. M. Dixon, P. T. Cummings and K. Hess, "Investigative Tools: Theory, Modeling and Simulation," In *Nanotechnology Research Directions: IWGN Workshop Report Vision for Nanotechnology in the Next Decade*; M. C. Roco; S. Williams and P. Alivisatos, Ed.; Kluwer Academic Publishers: Dordrecht, 2000.

predicted, and new nanostructured materials designed. For instance, computational nanoscience makes it possible to answer the question "What if we had a quantum computer? What could we do with it that we could not do with conventional computers? What would need to be achieved experimentally to create a quantum computer?" These are important questions to answer before engaging in the high-cost experimental pursuit of a quantum computer.

With appropriate investments in theory, modeling, and simulation, computational nanoscience has the potential to be an equal partner with experiment, and the most crucial tool in the design of manufacturing processes for devices based on nanoscale structure and function.

In a recent DOE report on computational nanoscience[6], some of the remarkable advances of the past fifteen years that have revolutionized computational nanoscience were identified:

Continuation of Moore's law, and beyond-Moore's-law increases in computing power enabled by advancing chip technologies and massive parallelization. One measure of the combined impact of Moore's law and parallelization is to look at winners of the Gordon Bell Prize, given each year at the Supercomputing conference to the application demonstrating the highest sustained performance. In 1988, the prize was given to an application achieving 1 Gflop/s in 1988; in 2003, the winner attained 35 Tflop/s. This is a 35,000-fold increase in just 15 years; Moore's law alone would predict ten doublings for a thousand-fold increase if all of the power of the extra transistor density went directly into floating point rate.

Explosion in application and utility of density functional theory. DFT provides a very cost effective means of including some of the effects of electron correlation, which is important in non-metallic nanosystems.

Molecular dynamics on as many as billions of atoms. Large-scale MD calculations have been enabled by new software designed and written for massively parallel computers.

Revolution in Monte Carlo methods. Examples include the Gibbs ensemble method for direct simulation of phase equilibria, continuum configurational bias methods for simulating long-chain systems, parallel tempering methods, and the Wang-Landau density of states algorithm. These methods permit extraordinarily fast equilibration of systems with long relaxation times.

New mesoscale methods. These include dissipative particle dynamics and field-theoretic polymer simulation. Such methods enable the study of systems with long relaxation times and large spatial scales.

Coupled Cluster and Quantum Monte Carlo methods for nearly exact descriptions of the electronic structures of molecules. These methods provide a level of benchmarking to establish the validity of more approximate methods that has been missing heretofore.

---

6 C. W. McCurdy, E. Stechel, P. T. Cummings, B. Hendrickson and D. Keyes "Theory and Modeling in Nanoscience: Report of the May 10–11, 2002, Workshop Conducted by the Basic Energy Sciences and Advanced Scientific Computing Advisory Committees to the Office of Science, Department of Energy," 2002.

Car-Parrinello and related methods for molecular dynamics with force fields calculated on-the-fly from density functional theory. These first-principles molecular dynamics methods have made it possible to study complex systems in which chemical reactions (bond breaking and making) take place. The primary disadvantage is the high computational cost, which has limited these methods to small systems (about 100 atoms) and short times (about 10 ps).

The importance of theoretical insights and algorithms. *Progress in computational materials science and nanoscience has traditionally been advanced at least as much, and usually more, by breakthroughs in algorithms than by increases in computing power as David Landau has pointed out. Improvements in computer speed (Moore's law) alone would account for three orders of magnitude increase in performance over the period 1970-1995, whereas theoretical insights and algorithmic advances have resulted in an additional seven orders of magnitude increase in the speed of Monte Carlo algorithms for simulating*

These and other advances have revolutionized the capabilities that computational nanoscience can bring to bear on the outstanding problems in nanoscience and nanotechnology, leaving the field poised for major accomplishments. It is important to recognize that advances in the theory and algorithms underlying computational nanoscience account for as much or more of the advancements of the field as do increases in computing power. Recognizing this, the DOE recently established a joint program to fund research involving equally nanoscience modelers and mathematicians/ computational scientists to develop novel approaches to outstanding problems in nanoscience. One of the major challenges to computational nanoscience is multiscale modeling (see Fig. 1), and one of the goals of DOE's computational nanoscience research program is to investigate whether some of the multiscale modeling tools developed within other application areas (*e.g.*, global climate) might have applicability within computational nanoscience. Clearly, because they involve widely varying time and or spatial scales, solution of many of the interesting problems in computational nanoscience will only be achieved when the multiscale modeling problem, with its related theoretical and algorithmic issues, is substantially solved.

However, there are several problems that have been solved, or systems whose properties have been substantially understood, even with today's theory, algorithms and level of computing capability. Many of the properties of one class of nanomaterials – carbon nanotubes – are now largely understood and predictable on the basis of electronic structure calculations. We are certainly in the situation where our ability to predict the properties of specific single-wall carbon nanotubes far outstrips our ability to synthesize them with such specificity. On the other hand, modeling and optimizing the processes for catalytically producing carbon nanotubes is well beyond current algorithmic and computational capabilities, as it is an example of a multiscale process involving wide ranges of time and length. Computationally we can today characterize the adsorption properties of several regular (*i.e.*, crystalline) nanoporous materials, but not yet simulate the process to make them (usually, templated directed self-assembly) – that is, we can tell

an experimentalist what constitutes a desirable material, but not the path to synthesize it. We can calculate the conductance of a single molecule, but not within the complex

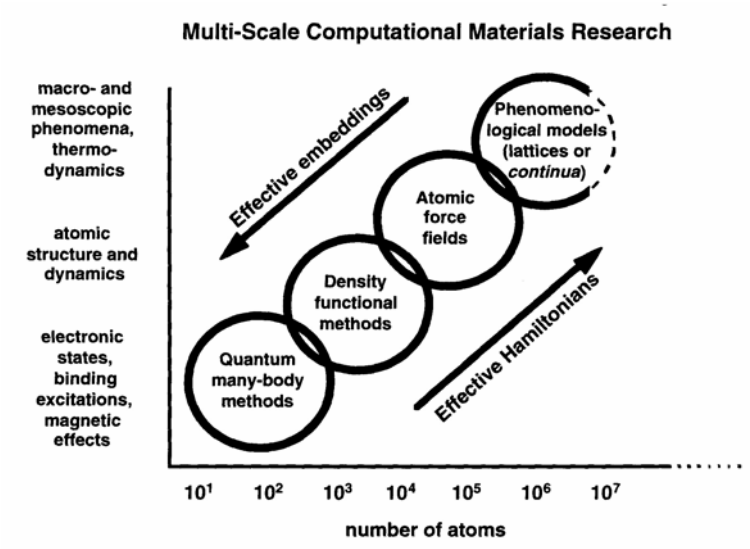


Figure 1. True multiscale modeling occurs when information can pass seamlessly between various levels of description, both when up-scaling from shorter time and/or length scales to longer time and/or length scales and vice versa (down-scaling). While up-scaling (or coarse-graining) is reasonably well understood, its counterpart (down-scaling) remains an unsolved problem in general and the focus of many research efforts.

environment in which it is measured experimentally (which usually involves processes of self-assembly, reaction, and in some cases the motion of an atomic force microscope tip across a surface). Finally, computational nanoscience has led to understanding of some aspects of nanotribology, the science of friction and wear between two surfaces separated by nanoscale distances. These are just a few examples of current successes in computational nanoscience; additional examples are documented in the recent DOE report[5].

A number of outstanding problems in computational and theoretical nanoscience could be addressed by a thousand-fold increase in computing power. The problems that would benefit most are those for which such an increase would result in a thousand-fold increase in the time scale or system size studied, thus bringing important new phenomena into the realm of feasible simulation. For example, for atomistic simulations involving short-range forces, molecular dynamics calculations scale as N, so that a thousand-fold increase in computing power would result in a thousand-fold increase in the numbers of atoms simulated. On the other hand, most current electronic structure methods scale as N^3 or worse, where N is the number of atoms in the system. For these calculations, a thousand-fold increase in computing power will only translate into a factor of 10 in system size as measured by atom number. This illustrates why investments must be made both in computer hardware and in the theory and algorithms that underlie the methods. The

development of *O*(*N*) methods for electronic structure calculations, currently the focus of several research groups, would dramatically increase the impact of a thousand-fold increase in computational power. Increases in computing resources by thousand-fold also have impact in the quality of results obtained: One could average a thousand calculations in the time it currently takes to do a single one, so the *precision* of calculations that rely on ensemble averages (*e.g.*, averaging over different initial states) would be enhanced dramatically by a thousand-fold increase in computing resources. Among the simulations ready to scale up are: the switching of magnetic nanoparticles, the design of optical band gap nanocrystals, molecular electronics, organic-inorganic interfaces in solution, nanoadhesives, superstrong structural materials, nanomotors, and nanocrystal growth.

As an example, consider the switching of magnetic nanoparticles. Current magnetic storage technology is based on the giant magnetoresistance effect (GMR), which was discovered in 1988 and commercialized by 1998, and yields gigabit/in^2 data storage. The rapid (10-year) transition from laboratory discovery to commercial product was driven to a large degree by computational and theoretical modeling.[5] GMR-based drives are at the heart of high-capacity disk drives for laptops, cameras, *etc*. (In fact, in disk storage, the equivalent of Moore's law is that storage capacity, enabled by GMR, increases by a factor of two every twelve months.) The next generation of storage devices, corresponding to terabit/in^2 storage capacity, will likely be based on nanoparticles. Storing a bit as the spin of a FePt nanoparticle with 5nm diameter yields a storage capacity of 10-100 Tb/in^2. However, while FePt particles in this size range can be synthesized, virtually nothing is known about such particles. In particular, the dynamics of flipping the spin (*i.e.*, writing) are unknown, and extraordinarily difficult to study experimentally. With a thousand-fold increase in computing power, it should be possible to study and characterize, using all-electron *ab initio* methods, the spin dynamics of a 3nm particle (about 1000 atoms), not quite as large as the target particle, but sufficient to provide considerable insight.

Research Issues

The primary scientific barriers to further advances in computational nanoscience have been identified as follows:[5]

- Bridging length and time scales from electronic through macroscopic
- Calculating electron transport mechanisms at nanoscale
- Developing theoretical and simulation approaches to nano-interfaces
- Calculating optical properties of nanoscale structures and opto-electronic devices
- Understanding and predicting complex nanostructures involving "soft" biologically or organically based structures and "hard" inorganic ones as well as nano-interfaces between hard and soft matter
- Understanding and predicting self-assembly and directed self-assembly
- Developing theoretical and simulation approaches to quantum coherence, decoherence, and spintronics

- Developing self-validating and benchmarking methods

Most of these are self-explanatory. The last item is perhaps less so. Since the nanoscale is such difficult experimental territory, we often do not have the luxury (as we do at the macroscale) of validating calculations against experimental measurements. For example, in developing classical force fields for atomistic simulation, at the macroscale it is typical to use experimental measurements (*e.g.*, phase equilibria) to refine the force fields; such validation methods are not available at the nanoscale. Self-validating methods means that a calculation at any level of description must be validated by more fundamental calculations at a lower level of description. As a result, for validation if nothing else, computational nanoscience will always demand extreme levels of computing, since only by going to more fundamental methods can one be assured of the validity of the assumptions and approximations.

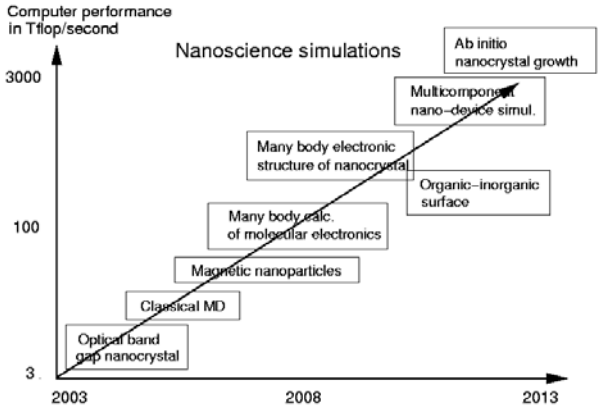


Figure 2. One projection for the type of problems that might be addressable in computational nanoscience in the future as tera- and peta-scale computational capabilities become available.

Technology Barriers

Computational nanoscience covers a wide variety of methods and techniques, so that the computer hardware requirements are rather general. Many of the calculations are compute-bound for long periods of time – the input to the calculation is small, and the output of the calculation is small. Checkpointing and visualization files can be large. However, other kinds of calculations (*e.g.*, quantum chemistry calculations on large numbers of atoms with large basis sets) can result in enormous linear algebra calculations (*e.g.*, the inversion of a matrix of dimension one million) requiring large scratch data spaces. Nevertheless, as a whole, nanoscience calculations are largely dominated by computational performance issues. Hence, the issues in nanoscience are:

- Interconnect speed, bandwidth, latency
- CPU memory bandwidth
- CPU speed

The faster the individual processor and the faster the interconnect, the more productive

nanoscience calculations can be, so long as the memory can keep up with the processor. The important point is that most algorithms relevant to computational nanoscience have global communication requirements that put a premium on the network. There are some techniques in computational nanoscience (such as parallel tempering methods) that are ideally suited to a low latency interconnect computing systems (such as clusters). These are the exception, rather than the rule.

We note that there are several efforts to build hardware specific to the algorithms used in computational nanoscience, particularly for molecular dynamics. Examples include GRAPE and the IBM Blue Gene project.

On the algorithmic side, the three main issues that dominate computational nanoscience are:

- Multiscale methods (auto up- and down-scaling).
- Guaranteed accuracy linear scaling , *i.e.*, $O(N)$, for electronic structure calculations.
- Overcoming time barriers, *i.e.*, developing methods that speed up time stepping.

Another issue is the manipulation of the data in density of states (DOS) calculations (*i.e.*, calculations in which the probability of states of the system is calculated). Terabytes of data will be generated for large DOS calculations and the manipulation of this data (presumably distributed across many processors) is a significant issue. There is still fundamental science to be addressed – such as electron transport (*i.e.*, solving the time-dependent Schrödinger equation) and methods for heavy atoms in which relativistic effects are important. These problems require theoretical breakthroughs, not brute-force extension of existing methods. However, the requisite theoretical insight may be driven by very large scale, heroic calculations.

Resources Required

Computational nanoscience is capable of utilizing all computational resources made available to it. Productions runs today on more than 1024 processors are rare, due to queuing policies. Any given problem may be simulated at a more fundamental level, one which involves fewer assumptions and approximations but typically requires orders of magnitude more computation. For example:

- In excess of 10 Tflop/s sustained on specialized molecular dynamics hardware (MDGRAPE-2 and WINE-2) exhibiting near 100% parallel scaling efficiency
- In excess of 1 Tflop/s sustained for the public domain biophysical MD program NAMD, exhibiting parallel scaling efficiency of 70% on 2250 processors of Lemieux

The computational complexity scaling behavior of calculations in computational nanoscience depends upon the type of algorithms employed. Atomistic molecular dynamics for atoms with sort-ranged interactions scales as $O(N)$. Atomistic molecular dynamics for atoms with long-ranged interactions scales as $O(N (\log N)^{3/2})$ using PPPM or Particle-Mesh Ewald. Density functional theory, used in electronic structure calculations and ab initio molecular dynamics scales as $O(N^3)$.

For any given problem, there is always a larger spatial or time scale relevant to study. Hence, it is somewhat arbitrary to estimate the amount of computing capability needed. We estimate current demand as follows: The annual current capacity of NERSC is 10 Tflop/s-yrs, of which we estimate about 10% of the available resources are expended on computational nanoscience, or 1 Tflop/s-yr. Extrapolating from this, within the collection of all of the supercomputing centers in the US (DOE, NSF, DOD, *etc.*) we estimate that about 6 Tflop/s-yrs are devoted to computational nanoscience. Conservatively, we estimate that this represents 5% of the total demand, the remainder being met by workstation clusters and parallel supercomputers within universities and industry. Hence, we estimate that the total demand in 2004 for computational nanoscience is around 60 Tflop/s-yrs. In estimating future growth in demand, we make several additional assumptions. First, demand will increase at least as fast as Moore's law, since as computers become faster, the size of calculations performed grows to meet the available capacity. Second, many of the significant calculations in computational nanoscience will be performed on leading-edge high performance computers, for which the annual growth in capacity exceeds Moore's law due to increasing parallelization. Third, the field of computational nanoscience is growing, as a result of general investments in nanoscience (purely theoretical/computational projects constitute about 10% of those funded in the NNI) as well as specific investments (*e.g.*, the DOE computational nanoscience program). Since the national investment in nanoscience is growing rapidly, we can expect the same to be true of computational nanoscience. Moore's law alone would lead to a factor of 10 increase in computational nanoscience every 5 years; adding the effect of massive parallelization and increases in the size of the field, we estimate that the demand for computational nanoscience will increase by a factor of 20 every 5 years. Hence, from its current base of 60 Tflop/s-yrs in 2004, we estimate a demand for 1.2 Pflop/s-yrs in 2009 and 24 Pflop/s-yrs in 2014.

In terms of human resources, we expect the demand for computational nanoscience to grow at a rate at least as large as the nanoscience initiative, and there are reasons to expect it to grow at a larger rate than this. While computational and theoretical nanoscience currently represents 10% of the investment in the NNI, we expect the reliance on theory, modeling and simulation to grow relative to the overall investment as the complexity of experiments demands more from simulation and as the transition begins to products, where design becomes an issue. Thus, we expect this percentage to increase in the future. As the recent report on computational nanoscience demonstrated,[5] there is tremendous opportunity for mathematicians and computational scientists to impact computational nanoscience. For example, methods for scale-spanning that have been developed in other contexts (such as climate modeling) and software development/maintenance practices for large-scale complex community-based codes are two obvious capabilities of immediate interest to the computational nanoscience community. The DOE computational nanoscience program reinforces this development, since it explicitly calls for teams composed equally of computational nanoscientists and applied mathematicians/computer scientists to address the outstanding problems in nanoscience. However, it is clear that the current funding level ($6 million/yr) falls well short of the demand (only 4 of 34 proposals were funded, after many pre-proposals were

rejected to limit the full proposal pool). Quadrupling the computational nanoscience program budget in the short term, and growing the program to meet the expanding demand for computational and theoretical nanoscience, in concert with large increases in available computational resources, would allow computational nanoscience to fulfill its potential as a powerful new scientific "instrument" for research at the nanoscale.

Metrics of Success

In computational nanoscience, the metric of success is simply stated: prediction, not "post-diction," or explaining an experiment or observation after the fact. When computational scientists can reliably predict the outcome of an experiment before it is performed, or design a directed self-assembly process before it is tested, then we have reached the goal of computational nanoscience as a truly predictive tool. Reaching this goal requires large investments to advance theory, algorithms, and computational resources, and the people that produce these advances and make them useful.

Plasma Science: Taming a Star

Plasmas are very hot gases in which the individual atoms have broken up into a collection of electrons and atomic nuclei (or ions). Sometimes called the fourth state of matter, plasmas exhibit a rich variety of complex, collective phenomena. The sun and the stars are predominantly plasma, as is the composition of over 99% of the visible universe.

Plasmas interact strongly with magnetic fields. These fields can be imposed externally or be created by electrical currents flowing in the plasma itself. A plasma configuration can become unstable if the electrical currents, the temperatures, the densities, or their gradients exceed critical values relative to one another and to the strength and geometry of the magnetic field. As a result of these instabilities, the plasma will rearrange itself to seek a stable configuration where these criteria are satisfied. This rearrangement can occur on a global scale (Magnetohydrodynamic or MHD stability), or on a fine scale (plasma micro-turbulence).

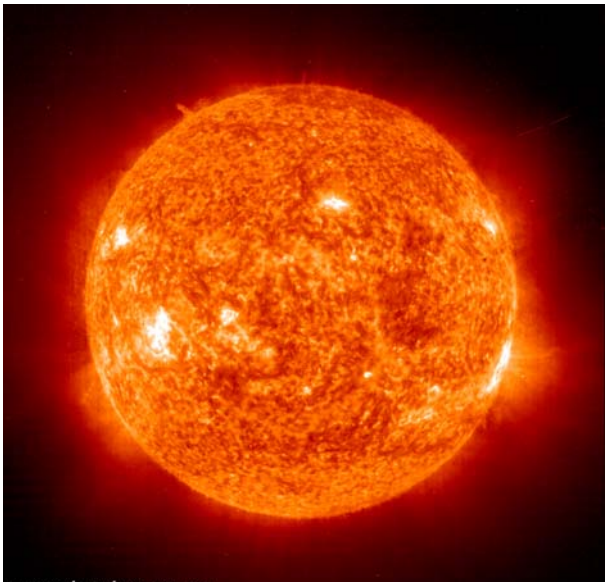


Figure 1: View of the sun from the SOHO satellite. Over 99% of the visible universe is plasma.

Radio-frequency (RF) electromagnetic waves interact strongly with plasmas. They are reflected, absorbed and transmitted in plasma depending on their frequency, the plasma temperature and density, and the orientation of the wave fields relative to the background magnetic field.

Plasma science seeks to understand the physical processes that underlie these phenomena.

Impact on Science and Society

Although plasmas play an important role in many aspects of everyday life, *e.g.*, neon signs, plasma video displays, spark plugs, and flames, a major focus of research in plasma science is the quest for harnessing fusion energy. The development of a secure and reliable energy system that is environmentally and economically sustainable is one of the most formidable scientific and technological challenges facing the world in the twenty-first century. The vast supplies of deuterium fuel in the oceans and the absence of long-term radiation, CO2 generation, and weapons proliferation concerns makes fusion the preferred choice for meeting the energy needs of future generations.

The DOE Office of Fusion Energy Sciences (OFES) supports an active research program in fusion energy science with three major U.S. Magnetic Fusion Energy (MFE) experiments underway, and is currently negotiating a role for the U.S. in the upcoming ITER burning plasma experiment. The U.S. also supports a large magnetic fusion theory effort, which has a long history of being at the cutting-edge of computational physics research. In fact, the present National Energy Research Scientific Computing Center (NERSC) is an outgrowth of the MFE computer center, MFECC, which was established in the late 1970s as the first national supercomputer center.

In MFE experiments, high-temperature (100 million degrees centigrade) plasmas are produced in the laboratory in order to create the conditions where hydrogen isotopes (deuterium and tritium) can undergo nuclear fusion and release energy (the same process that fuels our sun). Devices called tokamaks and stellarators are "magnetic bottles" that confine the hot plasma away from material walls, allowing fusion to occur. Unfortunately, confining the ultra-hot plasma is a daunting technical challenge. The level of micro-turbulence in the plasma determines the amount of time it takes for the plasma to "leak out" of the confinement region. Also, global stability considerations limit the amount of plasma a given magnetic configuration can confine and thus determines the maximum fusion rate and power output.

A complementary approach to MFE is called Inertial Fusion Energy (IFE). DOE's IFE program, also within OFES, is coordinated with, and gains leverage from, the much larger Inertial Confinement Fusion (ICF) program of the National Nuclear Security Administration (NNSA). In IFE, intense beams of particles (ion-beam fusion) or light (laser fusion) are focused on small targets that contain pellets of frozen heavy hydrogen. When these pellets are imploded sufficiently rapidly and symmetrically, the conditions for a small nuclear fusion "explosion" are created. These explosions release substantial energy, but are small enough that their energy can be confined within the fusion chamber, where it can be converted to a useful form. Plasma physics issues arise in the beam itself in the case of ion-beam fusion, in obtaining high compression ratios and maintaining symmetry in the target, and in an advanced concept known as fast-ignition.

Plasma science is also of great importance in understanding crucial interactions between the sun and the Earth. Plasma is always being emitted from the sun in the form of a supersonic wind called the "solar wind". In addition to the solar wind, plasma in the sun's outer atmosphere, called the "corona", can undergo sudden and violent activity in the form of "coronal mass ejections" and "solar flares", examples of which can be seen in Figure 1. As a result of these activities, billions of tons of matter and intense energetic particles can be thrown out of the solar corona into outer space, causing "storms" that can disturb significantly the near-Earth environment. All of the various phenomena that occur in the near-Earth environment whose behavior and interactions directly affect the planet and human technologies on and in orbit around it make up "space weather". Space weather can have significant effects for several Earth-based technologies such as satellites, communications and navigation systems, and radiation exposure in manned space missions.

Scientific Opportunities

Computational modeling currently plays an essential role in all aspects of plasma physics research. Perhaps nowhere is this as evident as it is in magnetic fusion energy (MFE) research where simulation models are actively being improved, tested and applied to the interpretation of data and to the design of new experiments. Improvements in the modeling comes in the form of both more complete models that include better descriptions of the physical processes and more efficient models that use advanced algorithms.

Present capability is such that we can apply our most complete computational models to realistically simulate both nonlinear macroscopic stability and microscopic turbulent transport in the smaller fusion experiments that exist today, at least for short times. Anticipated increases in both hardware and algorithms during the next 5-10+ years will enable application of even more advanced models to the largest present-day experiments and to the proposed burning plasma experiments such as ITER. (See Figure 2 and the discussion below).

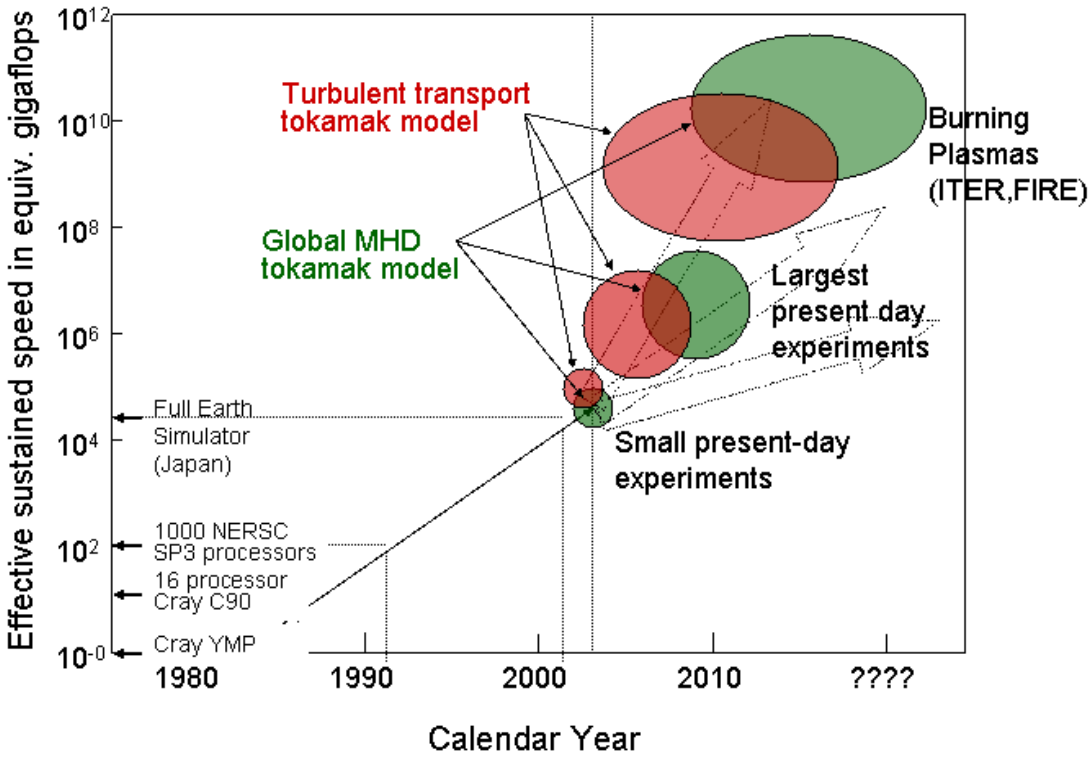


Figure 2. Factors of 100-10,000 in effective sustained speed are required to do complete modeling of proposed MFE burning plasma experiments.

A number of advances in the formulation and algorithms have complemented the increases in hardware speeds to provide vastly improved capability today than what was possible 30 years ago (see Figure 3). We expect this trend to continue into the future. This rate of increase of effective capability is essential to meet the anticipated modeling demands of fusion energy research as described below.

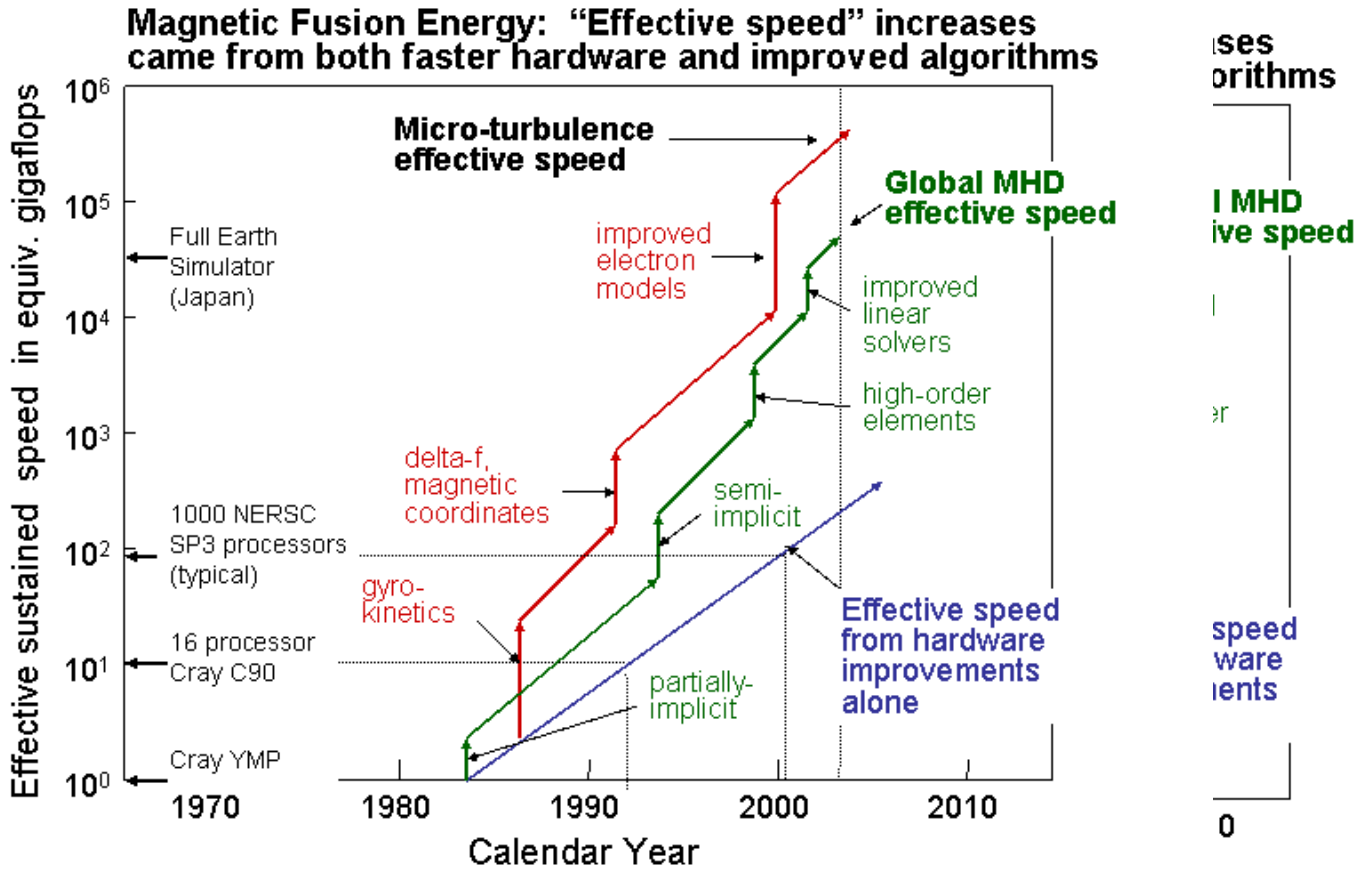


Figure 3. Magnetic Fusion Energy "effective speed" increases came from both faster hardware and improved algorithms.

The present thrust in computational plasma science is to merge together the now separate macroscopic and microscopic models, and to extend the physical realism of these by the inclusion of detailed models of such phenomena as RF heating and atomic and molecular physical processes (important in plasma-wall interactions), so as to provide a true integrated computational model of a fusion experiment. Such an integrated modeling capability will greatly facilitate the process whereby plasma scientists develop

understanding and insights into these amazingly complex systems that will be critical in realizing the long term goal of creating an environmentally and economically sustainable source of energy.

A number of external drivers are at work to make this time an especially opportune one for accelerating our capabilities in computational modeling of plasma. In MFE, the international ITER experiment is scheduled to begin its 10-year construction phase in 2006. There is a clear opportunity for the U.S. to take the lead in the computational modeling of this device, putting the U.S. in a strong position to influence the choice of diagnostic hardware installed and the operational planning of the experiments, and to take a lead in the subsequent phase of data interpretation. Furthermore, a comprehensive simulation model such as envisioned in the Fusion Simulation Project is felt to be essential in developing a demonstration fusion power plant, to follow ITER, by effectively synthesizing results obtained in ITER with those from other non-burning experiments which will be evaluating alternate MFE configurations during this same time period.

The IFE community expects that favorable results from the National Ignition Facility (NIF) in the next few years will further validate the models used in their target designs and will give an extra impetus to proceed with an Integrated Research Experiment in the 2015 time-frame. The space-weather community is anticipating an unprecedented influx of high-quality data from the NASA Magnetospheric Multi-scale Missions in 2009 and need to have computational predictions to be able to compare with these measurements.

Research Issues

The plasmas in modern magnetic fusion experiments are typically not quiescent, but exhibit macroscopic motions that can affect their performance, and in some cases can lead to catastrophic collapse of the discharge (see Fig. 4).

The modeling of such dynamics for realistic experimental parameters requires an integration of fluid and kinetic physics in a complex magnetic geometry as described by the extended-MHD equations. The magnetic field, required for confinement, imposes a large anisotropy to the problem. However, the key challenge in performing computations relevant to the hot plasmas of modern fusion experiments is to increase the dimensionless parameter

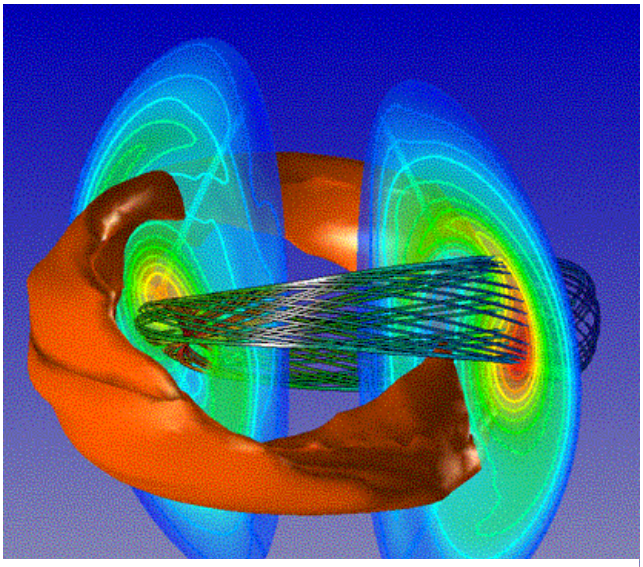


Figure 4: Extended-MHD calculation of an internal mode in NSTX.

characterizing inverse plasma collisionality, the Lundquist number, *S*. Present global MHD calculations are limited to Lundquist numbers $S < 10^7$ and problem times less than 1 msec. These values are adequate for modeling small low-temperature experiments, but are several orders of magnitude less than what are required to accurately simulate the largest of the existing fusion experiments. Several more orders of magnitude would be required to simulate "ITER-class" burning plasma experiments.

The confinement of energy and particles in fusion plasmas is often significantly degraded by turbulence associated with small spatial-scale plasma instabilities driven by gradients in the plasma pressure (see Fig. 5).

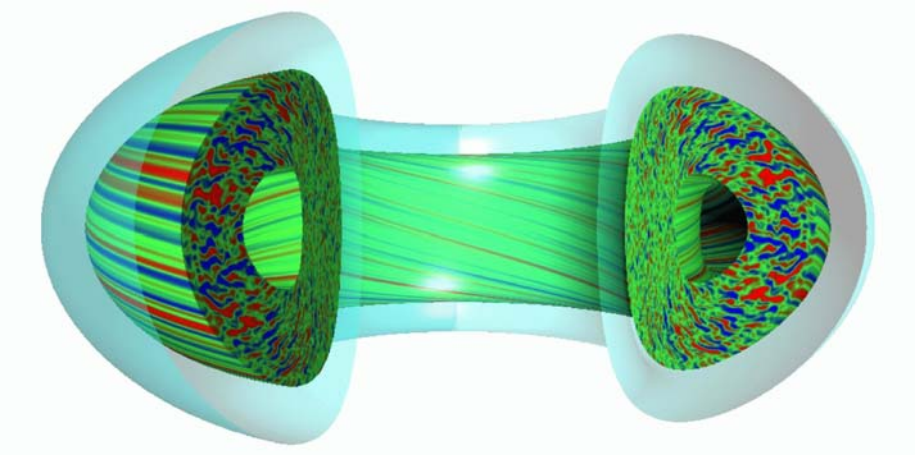


Figure 5: Electric potential structure associated with micro-turbulence in tokamaks.

The detailed physics of the growth and saturation of these instabilities, their impact on plasma confinement, and the development of an understanding of how such turbulence might be controlled remain unsolved problems for which we have only glimpses of understanding. At the present time roughly 10^{-3} s of a turbulent discharge can be modeled. This time needs to be increased by a factor of one to two orders of magnitude to address relevant time scales in the largest experiments and even more for ITER-class burning plasma experiments.

The bulk of the plasma turbulence results today have been obtained with an accurate model of ion dynamics (kinetic ions), which play a dominant role, but with a simplified (adiabatic or fluid) model for the electrons. This is not adequate for making quantitative predictions for real experiments. Early simulations of electron and electromagnetic effects reveal important dynamics on smaller and faster scales than what are encountered in electrostatic calculations. Simulations that can simultaneously resolve ion, electron, and electromagnetic-scale interactions necessitate an increase in computing resources of two orders of magnitude.

The scientific issues that arise when modeling a magnetic fusion experiment encompass a wide range of disciplines including those mentioned above, as well as others. However the dynamics of high temperature plasma does not respect these categorizations and an understanding of overall plasma performance requires combining all of these disciplines

in an integrated simulation that includes interactions between phenomena which were previously studied as essentially separate disciplinary problems. To achieve the ultimate goal of such an integrated approach, we must simulate the evolution of the 3D distribution of the plasma temperature, density, current and magnetic field on long time-scales in a way that includes all the relevant physical processes active at the shorter time-scales. While this is a long-term and ambitious goal, the program now stands ready to begin such cross-disciplinary studies and to increase the physics content of existing integrated codes. To accelerate this process the fusion community is engaged in a study laying the groundwork for a major initiative referred to as the Fusion Simulation Project. A major requirement for this endeavor is access to significantly increased computing power.

Progress in all key physics areas of Inertial Fusion Energy (IFE), including the "drivers" which impart the energy to the fusion fuel, the targets, and also the fast-ignition concept could be dramatically accelerated by increased computing resources. The principal IFE driver approach supported in the Office of Science consists of beams of heavy ions produced by linear induction accelerators. These intense beams are non-neutral plasmas that exhibit collective behaviors dominated by space-charge forces; demanding a self-consistent, integrated treatment from the source to the target.

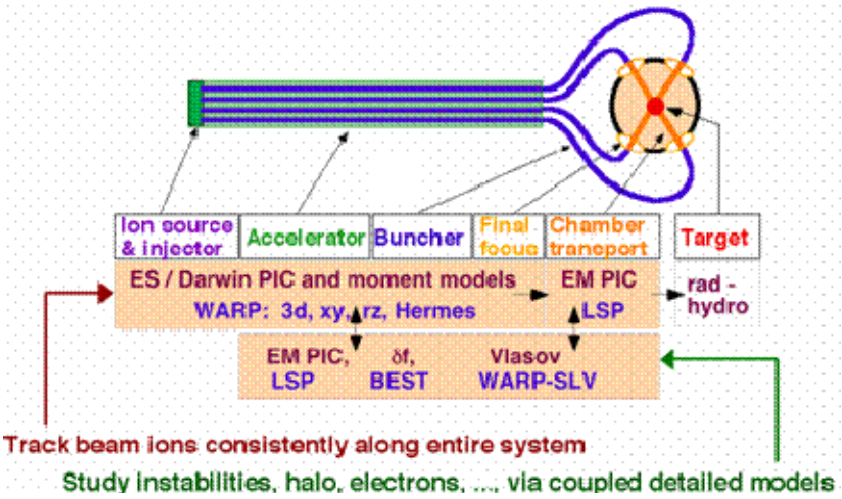


Figure 6: Heavy Ion driver simulation strategy.

Similarly, 3-D simulations of targets are required. Finally, there is broad international interest in fast ignition, which uses a separate short-pulse laser to ignite the compressed fuel, reducing the total required input energy.

Space weather simulations typically couple physical processes on the very large solar-terrestrial scales to small scales that are one-thousandth of the Earth's radius. It is necessary to carry out such calculations for several hours of real time in order to be able to predict even short-term space weather. (Such a simulation would require of the order of 10^{21} aggregate flops and run for several months at close to peak performance on the 40 Tflop/s Japanese Earth Simulator.) Furthermore, as in MFE research, our physical understanding of plasma macro- and micro-instabilities and their implications for plasma

stability and transport remain poorly understood, and the interplay of these effects in a complex integrated model can be done only by means of computer simulations.

Technology Barriers

The existing large-scale plasma science codes, both kinetic and (extended-MHD) fluid-based codes, typically exhibit relatively low ratios of sustained to peak performance (2% to 10%) on the current generation of IBM SP-type machines. They are limited by memory bandwidth and latency rather than by raw processor speed. The PIC kinetic codes would also benefit from computers with hardware support for gather-scatter and scatter-add operations. The codes typically do not scale well above 1000 processors (for strong scaling) on existing architectures, primarily due to latency in the interprocessor interconnects.

There are many algorithmic improvements that could still be made to the plasma science codes. The fluid codes are dominated by sparse-matrix solves, and improvements here would be of immediate benefit. There is a need for application-specific preconditioners for strong anisotropy and multi-scale phenomena. Initial results with high-order and spectral finite elements look very promising for efficiently representing multi-spatial scales and strong anisotropy. Nonlinear implicit techniques show promise for dealing with the multi-temporal scales. The hybrid particle/fluid description is a particularly efficient representation for many applications where a small component of high-energy particles affects the global stability properties of the plasma. This should be developed further. Advanced adaptive mesh refinement (AMR) algorithms also show promise but must be extended to handle implicit equations and hybrid calculations.

The large amounts of data generated by plasma science simulations are inconsistent with the traditional data-management and visualization tools now in use. Tools for large-scale data and meta-data management and to facilitate remote visualization are needed. Also needed are efficient means for check pointing and restarting large jobs and software tools that facilitate managing the complexity of large distributed software projects.

Resources Required

An increase of two to three orders of magnitude in computing power would qualitatively enhance each of the sub-fields mentioned above, allowing dramatic new capabilities that can move plasma science to the next level. The increase should take many forms, from desktop to flagship facility. Increased compute power is not only needed for increased resolution. It is also needed to enable integrating model components, increasing model fidelity, executing longer runs, increasing the number of runs in an ensemble to improve statistics or to investigate multi-dimensional parameter spaces, and increasing the number of applications of a given capability. We plot in Figure 7 the approximate ratios of compute power (in terms of aggregate Tflops for an entire calculation) versus memory requirement in Gigabytes for typical existing and anticipated calculations in Plasma Science.

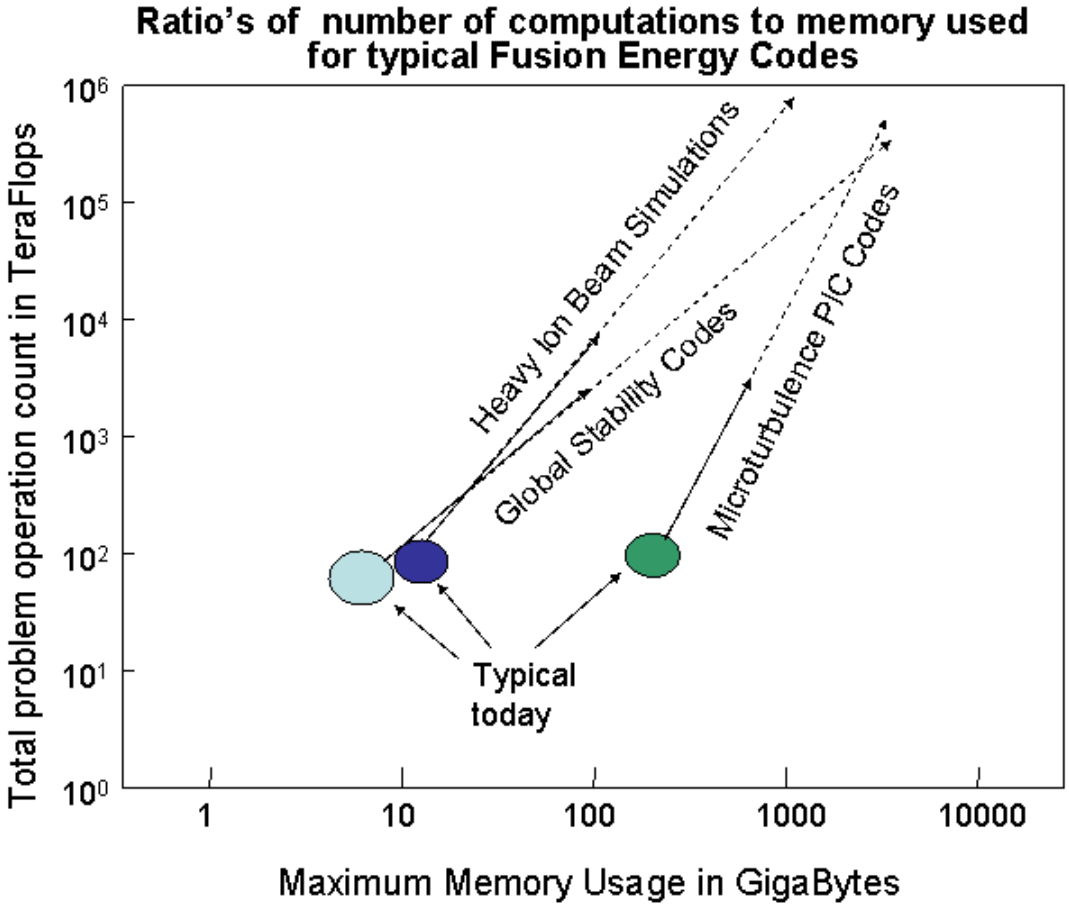


Figure 7. Total flops/calculation and total memory usage for typical existing and anticipated calculations in plasma science.

This should serve as a guide to the type of large-scale computing hardware required. To be effective, this hardware increase must also be accompanied by an increase in human resources in the form of teams consisting of computational physicists, applied mathematicians and computer scientists.

Future capabilities and MFE community computing requirements

	Today	5-years	10-years
Target capability	• resistive MHD, small experiments, short times ▪ turbulence in core of small experiments, ▪ turbulence in core of ITER for short times ▪ fluid edge turbulence in existing experiments	• resistive MHD in DIII-D ▪ 2-fluid MHD in small experiments ▪ routine turbulence simulation of existing experiments ▪ turbulence in ITER core ▪ kinetic edge turbulence in existing experiments	• resistive MHD in ITER ▪ 2-fluid MHD in DIII-D ▪ turbulence simulation of ITER, core to edge, including transport barriers, short times
Required flops	$5x10^{18}$	$5x10^{19}$	$5x10^{20}$
Aggressive capability		▪ resistive MHD in ITER ▪ 2-fluid MHD in DIII-D ▪ turbulence simulation of ITER, core to edge, including transport barriers, short times	• 2-fluid MHD in ITER ▪ turbulence in ITER, core to edge, long times ▪ unify microscopic and macroscopic scales
Required flops		$5x10^{20}$	$5x10^{22}$
Minimal capability		▪ resistive MHD, small experiments, longer times ▪ turbulence, small experiments, short times	▪ resistive MHD in DIII-D ▪ 2-fluid MHD in small experiments ▪ routine turbulence simulation of existing experiments ▪ turbulence in ITER core ▪ kinetic edge turbulence in existing experiments
Required flops		10^{19}	$5x10^{19}$

Note: flops = total (actual) floating point operations in 1 year by entire community

Figure 8. Resources required in Total flops/year for minimal, target, and aggressive capability in MFE computational physics research.

Today's largest calculations require about 3×10^{16} aggregate flops. They are run for about 80 hours on 1024 processors at about 100 Mflop/s per processor. This would take about 30 s on a computer that actually delivered 1 Pflop/s sustained performance. If we could run such jobs for 8 hours, this would be a factor of one thousand increase. These

simulations can calculate the turbulence due only to ions (with a simplified electrostatic, adiabatic approximation for the electron response) for a time period of about 1 millisecond, or isolated macroscopic stability events in some of the smallest experiments today using the actual parameters of those experiments.

In MFE, for example, we anticipate that an increase of two to three orders of magnitude in computing power will enable fusion researchers to quantitatively predict the onset conditions, strength, and nonlinear saturation mechanisms of both micro-scale and macro-scale instabilities, a major step in understanding how to control them. This same level of increase in computing power would allow researchers to begin developing fully integrated simulations of fusion systems that span the scales from micro to macro. Such an integrated simulation capability would dramatically enhance the utilization of a burning fusion device in particular and the optimization of fusion energy development in general, and would serve as an intellectual integrator of the broad range of physics phenomena occurring in advanced tokamaks.

In space weather, with a factor of two to three orders of magnitude increase in current computational speed, it would become feasible to carry out global space weather simulations that couple large solar-terrestrial scales to much smaller scales involving ion dynamics at Lundquist numbers $S < 10^6$. This cannot be done at the present time with existing supercomputers in the U.S., and yet is necessary if space weather models aspire to capture accurately the physics of collisionless space plasmas.

The biggest programmatic step enabled by petascale computing will be a comprehensive, integrated simulation bringing together all of the subdisciplines in fusion simulation and able to predict reliably the behavior of plasma discharges in a steroidal magnetic fusion device on all relevant time and spatial scales. This will require gains in algorithms for multiscale nonlinear problems, in addition to raw petaflop/s of computing power.

Besides the hardware costs, it is anticipated that a major new computing initiative directed at developing an integrated computational model of a MFE fusion reactor would require an influx of new funding of about $20 M/year for 5 to15 years to support an integrated team of physicists, mathematicians, and computer scientists to develop the large scientific application software required to provide high fidelity simulations of the reactor. Similar initiatives in IFE would require $5 M/year, and in space weather $3 M/year.

Metrics of Success

The first metric of success of this simulation effort can be measured by the degree to which simulation results agree with existing and anticipated high-resolution experimental measurements. There is a wide range of experimental results for comparison that are available now or will be available in the near future. New MFE imaging diagnostics will facilitate comparisons with turbulence and RF calculations. The proposed integrated beam experiment will validate IFE codes. Satellite measurements will be compared with the space-weather calculations. Successes in reproducing these measurements from

simulated results will be a dramatic validation of this field's capability and of computational physics in general.

However the true success of this endeavor will be measured by the degree that these simulation codes are relied upon for optimization of experimental operations and the design and optimization of the next generation of experiments and communication systems. In fusion energy, these would be the "Demo-class" power-producing fusion reactors. It is expected that the design capability offered by these simulation codes will give the U.S. a significant competitive edge in the design and manufacture of commercial fusion energy power plants.

Simulating Quarks and Gluons with Quantum Chromodynamics

The goals of high energy and nuclear physicists are to identify the fundamental building blocks of matter and to determine the interactions among them that lead to the physical world we observe. Major progress has been made towards these goals through the development of the Standard Model of high energy and nuclear physics. The Standard Model consists of two quantum field theories. One, Quantum Electrodynamics, provides a unified theory of the weak and electromagnetic interactions, and the other, Quantum Chromodynamics (QCD), provides a theory of the strong, or nuclear, interactions. The Standard Model identifies quarks as the fundamental building blocks of strongly interacting matter, and gluons as the carriers of the strong forces. QCD explains how quarks and gluons interact to form the particles we directly observe, such as protons, neutrons and the host of short-lived particles produced in accelerator collisions. It also describes how they interact to form atomic nuclei.

The Standard Model has been enormously successful. It correctly describes a vast amount of data produced in high energy and nuclear physics accelerator experiments, and in cosmic ray experiments. It has passed every experimental test to which it has been put. However, our knowledge of the Standard Model is incomplete because it has been difficult to extract many of the most interesting predictions of QCD. The only existing method for doing so from first principles and with controlled systematic errors is through large-scale numerical simulations within the framework of lattice gauge theory.

QCD simulations are needed to determine a number of the basic parameters of the Standard Model, to make precision tests of it, and to obtain a quantitative understanding of the physical phenomena controlled by the strong interactions. Despite the many successes of the Standard Model, it is expected that by probing to sufficiently short distances or high energies, one will find a more encompassing theory. It is not that the Standard Model is expected to be proven wrong, it is simply expected to have a limited range of applicability, just as classical mechanics does. A central objective of the experimental programs in high energy and nuclear physics is to search for a breakdown in the Standard Model and new physics beyond it. However, one cannot determine whether a theory breaks down without knowing what it predicts, thus QCD simulations are an integral part of this effort. In a significant number of cases, uncertainty arising from lattice resolution is the major present impediment to progress. Thus, numerical studies of QCD play an important role in efforts to obtain a deeper understanding of high energy and nuclear physics.

Impact on Science

High energy and nuclear physicists seek to understand matter at the smallest distance scales or largest energy scales. However, these fields impact science at all scales, including the largest probed by astrophysics. Because of the inherent interest and

scientific importance of high energy and nuclear physics, the United States, the European Community, and Japan all support very large experimental programs in these fields. Major goals of the experimental programs are to: 1) verify the Standard Model of High Energy Physics, or discover its limits, 2) determine the properties of strongly interacting matter under extreme conditions, and 3) understand the internal structure of nucleons and other strongly interacting particles. Lattice QCD simulations are essential to research in all of these areas.

Most strongly interacting particles decay via the weak interactions. In many cases an experimental measurement of the decay properties, coupled with a lattice QCD calculation, will provide a direct measurement of one of the fundamental parameters of the Standard Model. By determining the same parameter from different experiments and lattice QCD calculations, one can check the consistency of the Standard Model. A major, international effort is in progress to perform experiments required to determine the Standard Model parameters associated with heavy quarks (charm, bottom, and top). These are among the least well-known parameters in the Standard model, and their precise determination would provide a very important test of the theory. As Figure 1 illustrates, lattice QCD calculations are a vital part of this effort. Each solid colored band is the allowed region of the Standard Model parameters ρ and η from a particle experiment and corresponding lattice calculation. For the Standard model to be correct the solid bands must overlap, and ρ and η must lie in the region of overlap. The figure on the left shows the constraints as they exist today. The figure on the right shows the constraints as they would exist if the errors in the lattice QCD calculations were reduced to 3% with no improvement in the experiments. Thus, in this case, as in many other tests of the Standard Model, it is essential that improvements in lattice QCD calculations keep pace with those in experiments in order to reap the full return on the very large investments being made in the experiments.

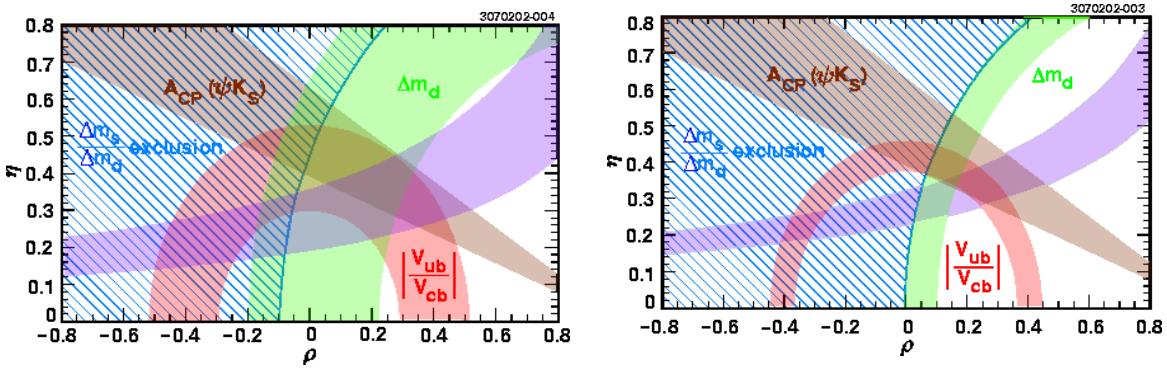


Figure 1. Constraints on the Standard Model parameters ρ and η (one sigma confidence level). For the Standard Model to be correct, they must be restricted to the region of overlap of the solidly colored bands. The figure on the left shows the constraints as they exist today. The figure on the right shows the constraints as they would exist with no improvement in the experimental errors, but with lattice QCD uncertainties reduced to 3%.

At ordinary temperatures and densities quarks and gluons are confined in elementary particles. However, at more than 100 million degrees Kelvin, as occurred in the first moments in the expansion of the Universe, or at nuclear matter densities exceeding one billion grams per cc, as may occur in the cores of compact stars, quarks and gluons are liberated and form a quark-gluon plasma, an entirely novel form of matter. Creating and studying the early Universe in microcosm is one of the principal and exciting goals of experiments at the Relativistic Heavy Ion Collider (RHIC) at the Brookhaven National Laboratory. Since the plasma would exist only for a fleeting moment before decaying into ordinary particles, it cannot be directly observed. Confirming its existence depends on model calculations that rest on results from first principles numerical simulations. Results are urgently needed to support the experimental effort.

The following observables, which can all be computed on the lattice, are required: 1) the temperature and order of the ordinary-matter/plasma phase transition; 2) the equation of state (energy density and pressure *vs.* temperature); 3) the strange quark excess and strangeness fluctuations; 4) the weakening of quark binding as the temperature and/or density increases; 5) the plasma modes of oscillation; and 6) phase diagram and equation of state at non-zero baryon density.

Knowing the temperature of the phase transition and equation of state tells us whether the energy of a heavy ion collision is sufficient to produce the plasma. If the transition is strongly first order, supercooling and phase separation could occur with dramatic consequences for the subsequent evolution of the final state. Excess strangeness production and strangeness fluctuations have been proposed as a signature for plasma production, but a first principles quantification is needed. The weakening of binding in heavy quark-antiquark systems is another signature. Plasma modes may also have a measurable effect in the production of muon-antimuon pairs. Finally, the phase diagram at nonzero density is central to a characterization and possible indirect detection of quark-matter cores inside dense stars.

Protons, neutrons, and other strongly interacting particles, which comprise most of the known mass of the universe, are dramatically different from any other quantum systems that have ever been explored. The quark and gluon constituents of a proton are absolutely confined within it at ordinary temperatures and densities, and cannot be removed, unlike electrons that can be removed from an atom, or nucleons that can be removed from a nucleus. Unlike the photons, which carry the electromagnetic forces in atoms, the gluons, which carry the strong force, are essential constituents of the proton, accounting for half of its momentum and undergoing significant excitations. Whereas most of the mass in atoms and nuclei comes from the masses of the constituents, most of the mass of the proton arises from the gluon interactions. Indeed if the quarks became massless, the mass of the proton would hardly decrease. Because quarks and gluons interact so strongly and nonlinearly, the only known way to calculate the remarkable properties of the strongly interacting particles they form is by numerical studies of QCD.

In the three decades since the discovery of quarks in the nucleon, a tremendous experimental effort has been devoted to exploring the quark and gluon structure of

hadrons at MIT-Bates, Brookhaven National Laboratory (BNL), Cornell, Fermi National Accelerator Laboratory (FNAL), Jefferson National Accelerator Facility (JLab), the Stanford National Accelerator Laboratory (SLAC), and international accelerators. The distributions of quarks and gluons in nucleons have now been measured in exquisite detail. Surprising puzzles have arisen from these experiments, such as the discovery that hardly any of the spin of the proton comes from the spin of valence quarks, in striking contrast to atoms and nuclei, whose spin arises primarily from the spin of valence fermions. Other experiments measure the strange quark content of the nucleon and seek evidence for hadrons with exotic quantum numbers that would expose novel QCD phenomena. Lattice calculations are essential to quantitatively understand these experiments from first principles, and to obtain physical insight into how QCD works.

Scientific Opportunities

There is a wide range of scientific opportunities in lattice QCD. As in most areas of computational science, the problems that can be addressed depend critically on the computational resources available.

The U.S. lattice QCD community currently sustains just under 1 Tflop/s. That is, it has continuous use of computers that together sustain this rate. No one machine realizes this performance. These resources come from allocations at DOE and NSF supercomputer centers, special purpose computers located at Columbia University and BNL, and optimized clusters at FNAL and JLab. They have allowed the determination of a limited number of key quantities to an accuracy of a few percent. Among these quantities are the strong coupling constant, the masses of the *c* and *b* quarks, and the decay constants of the π and *K* mesons. These resources have also enabled the development and testing of new formulations of lattice QCD that will significantly improve the accuracy of future calculations. As might be expected, these more sophisticated formulations require significantly more computer power than the simpler ones they replace.

In order to provide timely support to the experimental programs in high energy and nuclear physics, the U.S. lattice QCD community will need to sustain 100 Tflop/s within the next few years, and multiple Pflop/s within the next ten years. Resources at this level will enable a wealth of exciting calculations with existing algorithms and codes. Here we mention a few examples.

The study of the decays of mesons with one light and one heavy quark via the weak interactions is a key component in the program to make precision tests of the Standard Model. Two examples are the *B* and *D* mesons, which contain a heavy bottom and charm quark, respectively. Their decays to final states that do not contain strongly interacting particles are characterized by decay constants. Calculations currently in progress are expected to determine these decay constants to an accuracy of approximately 5%. However, an accuracy of 1% is needed to resolve Standard Model tests. With current algorithms this improvement will require a hundred-fold increase in computing resources. The Cleo-C experimental program in progress at Cornell will measure the *D* meson decay constant to high accuracy, thereby providing a validation of the lattice calculation.

However, the *B* meson decay constant will be extremely difficult to measure, so the lattice determination of it will be critical. With computing resources of this magnitude, there are a host of other processes that can also be calculated with accuracy sufficient to have an important impact on our understanding of the Standard Model. These include the decays of *B* and *D* mesons to final states containing strongly interacting particles, and the mixing of neutral *K* and *B* mesons with their anti--particles. Computational resources at the Pflop/s level would, for example, enable a high accuracy evaluation of ε'/ε. This quantity characterizes the violation of charge conjugation-parity (CP) symmetry in the *K* meson system. There has recently been a very accurate experimental measurement of this quantity, and an equally accurate lattice determination would provide an important test of the Standard Model. More generally Pflop/s scale resources would allow use of the most sophisticated of the recently developed lattice formulations with realistic quark masses. Such simulations would significantly improve the accuracy of all lattice QCD calculations.

Turning to the study of high temperature QCD, current calculations predict the temperature of the phase transition at zero baryon density with an accuracy of 15%, but have not yet established whether it is first order, or merely a crossover, at realistic quark masses. The equation of state and strangeness fluctuations have been determined only on coarse lattices where discretization artifacts are not completely under control. Simulations have confirmed the weakening of quark binding between very heavy quarks, but for light quarks the methodology has not been adequately developed. It is possible to follow the phase boundary slightly away from zero baryon number density, but unsolved sign fluctuation problems prevent ready simulation at high baryon number density.

With recently developed improved lattice formulations, and resources sustaining100 Tflop/s it will be possible to determine the temperature and order of the phase transition, the equation of state of the quark-gluon plasma, the strange quark excess and binding, and observe the weakening of quark binding, all at a resolution sufficient for modeling heavy ion collisions. Pflop/s resources will be required to study the properties of strongly interacting matter at high densities. There have been fascinating proposals regarding the possibility of superconducting phases at very high densities, which can be studied from first principles only with lattice methods. These studies have potential impact on our understanding of the early development of the universe and on the cores of neutron stars.

Resources at the 100 Tflop/s level will lead to major advances in our understanding of the internal structure of the nucleon. For example, they will enable a precision calculation of electromagnetic form factors characterizing the distribution of charge and current in the nucleon. This calculation is necessary to understand recent precise measurements of proton and neutron form factors at JLab and measurements at Bates and JLab of the contribution of strange quarks to these form factors. They will also make possible calculation of the quark structure of the nucleon at a level that will provide fundamental understanding of experiments at JLab using 6-12 GeV electron beams and HERMES and RHIC-spin high-energy experiments. In addition, they will enable calculation of transitions to excited nucleon states revealing the degree of deformation present in these

states and allow calculation of the potential between selected hadrons to explore the QCD origin of nuclear forces. Pflop/s resources would enable study of the gluon structure of the nucleon, in addition to its quark structure. They would also allow precision calculation of the spectroscopy of strongly interacting particles with unconventional quantum numbers, guiding experimental searches for states with novel quark and gluon structure.

The calculations listed above would significantly deepen our understanding of the Standard Model, and therefore of the basic laws of physics.

Research Issues

QCD is formulated in the four-dimensional space-time continuum. In order to carry out numerical studies it is necessary to reformulate the theory on four-dimensional space-time lattices. Physical observables are expressed in terms of Feynman path integrals, which can be evaluated numerically in the lattice formulation. In current simulations these integrals involve hundreds of millions of variables. The fact that the quarks are fermions severely complicates the evaluation of these multi-dimensional integrals by giving rise to nonlocal couplings among the integrands. Nevertheless, well tested algorithms exist to evaluate these integrals accurately. These algorithms are hybrids of molecular dynamics and Monte Carlo methods.

Once simulations have been performed on the lattice, two major challenges remain in order to obtain results for the continuum theory. The first is to perform extrapolations to the limit of zero lattice spacing. In recent years improved formulations of QCD on the lattice have been developed that sharply reduce finite lattice spacing artifacts, thereby significantly reducing errors arising from the extrapolation to the continuum limit. It is possible to match lattice results to underlying physical results through analytical calculations. To obtain accurate physical results, one must perform calculations for a range of (small) lattice spacings. The advantage of being able to work at increased lattice spacing can be seen from the fact that the computational resources required for a simulation grow approximately as the seventh power of the inverse of the lattice spacing. A second challenge arises from the fact that the up and down quarks have masses that are much smaller than those of the other quarks, and of the typical energy scale in QCD. Because the required computational resources grow approximately as the inverse of the quark mass to the power 2.5 to 3, it has not been possible to perform simulations at the physical masses of the up and down quarks. Instead one performs simulations at a range of unphysically large values of these masses, and performs extrapolations to the physical ones. These extrapolations are guided by theory based on a property of QCD known as chiral symmetry. The improved lattice formulations also reduce chiral symmetry violating lattice artifacts, bringing these extrapolations under significantly better control. Thus, the improved lattice formulations of QCD are having a dramatic impact on our ability to perform accurate calculations. Indeed, in the last several years, the advances arising from improved lattice formulations have far outweighed those from increases in computing power. Continued work in this area will be a very important component of

research in QCD in the coming years. However, it should be emphasized that in order to implement these more sophisticated formulations of lattice QCD much more computing power is required than to implement the older ones. Indeed, it seems likely that it will not be possible to use the most promising of them with realistic quark masses until one has computing resources capable of sustaining multiple Pflop/s.

Between 70% and 90% of the computer resources in lattice QCD calculations are spent on repeated inversions of the Dirac operator for the quarks in a background field produced by the gluons. On the lattice, the Dirac operator is a sparse matrix, and iterative techniques, such as the conjugate gradient algorithm, are typically used to perform this inversion. To date the rather random nature of the nonzero elements of the Dirac operator have hampered the application of standard multilevel solver techniques to accelerate the inversion. This is a problem on which a successful collaboration between application physicists and applied mathematicians could reap large benefits.

Going beyond the standard approaches, several algorithmic obstacles hamper the application of lattice methods to a variety of tantalizing new problems. For non-zero baryon densities the measure of the Feynman path integrals is not positive definite, making importance sampling techniques extremely expensive, if applicable at all. Recent developments appear to enable simulations at low baryon density, in the region relevant to the study of heavy-ion collisions. However, new ideas are needed to carry out studies of QCD at large densities. A related issue is the inability of current algorithms to deal with time dependent problems, such as the calculation of scattering cross-sections.

A more fundamental issue arises for lattice theories of the Standard Model when parity violation is included. No known lattice formulation of left-handed neutrinos has been found that does not introduce mirror right-handed neutrinos, which do not exist in nature. This is troubling as the lattice is the most precise way to define a quantum field theory non-perturbatively. Nevertheless this is not a practical problem for experimental predictions since weak interactions in most cases can be handled perturbatively.

Technology Barriers

Lattice QCD is a mature field with an extensive code base. Three major codes, CPS, MILC, and SZIN are publicly available. Each of these packages contains tools to carry out simulations with a variety of lattice formulations, and to study a broad range of physics problems. Each contains over 100,000 lines of code. The MILC and SZIN codes run on virtually all commercial machines, as well as on clusters, while the CPS code runs on special purpose computers designed at Columbia University, the most recent of which is the QCD on a Chip (QCDOC).

A major effort is in progress under the DOE SciDAC Program to develop a QCD Application Programming Interface (API), which will enable the U.S. lattice community to use special purpose computers, clusters and commercial machines with high efficiency, while preserving the large investment in existing codes. This project includes the creation of standards for communication interfaces, optimized low level algebraic kernels, optimized high level operators, and run-time functions. Existing application

codes can easily be ported to the QCD API by linking to libraries that are being created for each of the targeted architectures.

A number of major benefits are expected from the SciDAC effort. It has already been demonstrated that the QCD API will enable existing codes to be ported to, and obtain excellent performance on, the QCDOC with a minimum of effort. The high level structure of the QCD API should allow new applications to be developed easily, enabling the physicists to focus on QCD, rather than the code. In addition, with the existence of a true community code young scientists entering the field will be able to move to new collaborations, as they progress from graduate student to postdoc to junior faculty, without having to learn new software packages.

This SciDAC Software Project involves nearly all the senior lattice gauge theorists in the U.S., as well as a number of computer scientists and computer engineers located at universities and national laboratories. The effort currently supports approximately 10 FTEs, and has the participation of many more on a voluntary basis. As the community moves to computing platforms sustaining hundreds of Tflop/s and then Pflop/s, this software effort will need to grow by at least 50% in order to maintain and enhance the code that is currently being developed.

Resources Required

There are numerous exciting scientific opportunities in lattice QCD, and it is essential that the U.S. lattice community take advantage of them expeditiously. Important experiments that depend on lattice QCD calculations for their interpretation are in progress or are planned for the near future, so we must move quickly to support the experimental program in a timely fashion. Recent advances in algorithms, particularly new formulations of QCD on the lattice, now enable calculations of unprecedented accuracy, provided the required computational resources are available. Furthermore, we must act now to recover U.S. computational leadership in lattice QCD. Lattice gauge theory was invented in the United States, and U.S. physicists have traditionally been intellectual leaders in the field. However, for the last five years greater investments in computational resources have been made in Europe and Japan. Within the next year European lattice theorists will have dedicated resources sustaining over 10 Tflop/s. If U.S. physicists are to regain leadership of the field, and be able to attract outstanding young scientists, we must act now to obtain comparable resources. Within the next five years the U.S. lattice community needs to sustain hundreds of Tflop/s, and by the end of the decade multiple Pflop/s. Special purpose hardware provides a pathway for doing so, thanks to the relatively simple data structures and regular features of QCD simulations.

The underlying homogeneous and local character of Quantum Chromodynamics implies that the corresponding numerical methods are very well suited to parallel computing and can efficiently be mounted on a massively parallel computer with a torus communication network. This offers important opportunities for highly cost effective and very large scale computing since mesh machines can easily scale to tens of thousands of processors. Today 512 to 1024 processors would be typical of production jobs. Jobs as large as

12,288 processors have been run on the special purpose QCDSP computer at Brookhaven, and jobs using several thousand processors are routinely run on this machine and on a sister machine of 8192 processors at Columbia University. These machines were available much earlier than Pittsburgh's Lemieux and NERSC's Seaborg, and are still unrivaled by the latter for practical purposes, sustaining hundreds of Gflop/s in production mode.

Perhaps the longest single QCD job is being carried out at the Pittsburgh Supercomputer Center. It uses 1024 processors of the Compact AlphaServer, Lemieux. The code sustains 300 Mflop/s per processor, so the total sustained throughput is approximately 300 Gflop/s. Lemieux's processors have a peak speed of approximately 2 Gflop/s, so the code is running about 15% of peak. Approximately 6400 processor-hours are required for each time unit of the simulation and 3000 time units will be needed for the full simulation. This calculation therefore requires approximately 0.65 Tflop/s-years. Neither Lemieux nor NERSC's IBM SP Seaborg have sufficient capacity to allow completion of the calculation in a scientifically reasonable turnaround time, given the other load on them. In order to produce results in time to be useful for the experimental programs in high energy and nuclear physics, the typical turn-around time for a major QCD calculation should be about a year or less.

The percentage of peak obtained by a good QCD code varies with the balance of resources of the computer being used. For most commercial cache-based machines, it is 10-15%. For computers specially designed for QCD, it falls in the range of 35-50%, depending upon the type of problem.

A major challenge in the study of lattice QCD is to bring very large computing resources to bear on lattices of moderate size. For a fixed space-time volume the computational effort required for lattice QCD calculations grows with a high power of the number of lattice points. As a result, substantial benefits can be obtained from improved algorithms that permit increasingly accurate answers to be obtained without significantly increasing the number of lattice points (decreasing the lattice spacing), even though these algorithms increase the number of floating point operations per lattice point. Furthermore, it is vitally important to decrease the masses of the up and down quarks towards their physical values. This requires even more significant increases in computation per lattice site. Thus, computers with increasing numbers of processors are likely to be applied to lattices of limited size. These aspects of lattice QCD calculations permit a relatively small amount of memory to be used on each processor, presenting an opportunity to increase the cost effectiveness of customized machines.

The scaling of the computational complexity of the QCD calculations we would like to reach can be estimated from a few simple considerations. First, QCD calculations are large-scale Monte Carlo simulations. A large fraction of the computer time is spent generating independent configurations, snapshots of the system being studied. These configurations can be used to obtain a wide variety of physical results. To improve the accuracy of these results, one has to perform simulations on finer grids and with smaller quark masses. If a is the lattice spacing, the cost of generating configurations scales as

a^{-7} if all other parameters are held fixed. The cost of generating configurations also scales as $m_l^{-2.5}$, where m_l is the mass of the two lightest quarks (up and down).

An important consideration in designing customized computers for QCD is the relationship between data movement and floating point operations. The basic operation in QCD simulations is the multiplication of a three component vector, whose elements are complex numbers, by a 3×3 matrix, whose elements are also complex numbers. In performing this operation an average of 1.45 bytes of input data is required for each floating point operation, and an average of 0.36 bytes of output data is produced. In current commercial processors, data movement cannot keep up with floating point calculations unless the problem fits entirely into cache. However, restricting the amount of data on a processor to that which will fit into cache will be counter-productive unless the inter-processor communications system is sufficiently robust. One typically decomposes the four-dimensional lattice into domains of L^4 lattice sites, each domain being assigned to one processor. Since the underlying interactions are short range, in order to update variables at a lattice site, one needs information from only a few neighboring ones. Thus, one can update those sites needing only information stored on a processor, while the data needed to update sites on the edges of the domain is collected from neighboring processors. The sustained interprocessor bandwidth (in Megabytes/s) needed to completely overlap communication with floating point calculations interior to an L^4 domain is approximately 0.364 MF/L, where MF is the sustained floating point performance in Mflop/s. Thus, careful attention must be paid to the balance among memory bandwidth and latency, cache size, floating point performance and interprocessor bandwidth and latency, in order to obtain high performance.

These specific characteristics of lattice QCD can effectively be exploited using two strategies. With current technology, highly effective machines capable of sustaining hundreds of Gflop/s can be constructed from workstation clusters, a speed that will grow to 1 Tflop/s in a year to two. By careful selection of systems meeting the high memory and inter-processor bandwidth requirements of QCD and the use of mesh communication networks, efforts at both FNAL and JLab have resulted in very successful, dedicated machines constructed from commodity components. This strategy requires little design time and can respond quickly to advances in widely available and highly cost effective commercial technology.

A second strategy, exploits the well understood characteristics of lattice QCD algorithms to design and construct specifically targeted machines. Here considerably more design effort and time are required. However, highly scalable and very powerful machines result. The present QCDOC project, part of the U.S. SciDAC effort targeting machines sustaining tens of Tflop/s, is a significant example.

Metrics of Success

There is a wealth of data from high energy and nuclear physics experiments that can be used to validate lattice QCD calculations. This includes knowledge of the masses of strongly interacting particles, their decay properties and their internal structure. True success will be the generation of new results with accuracies sufficient to advance our

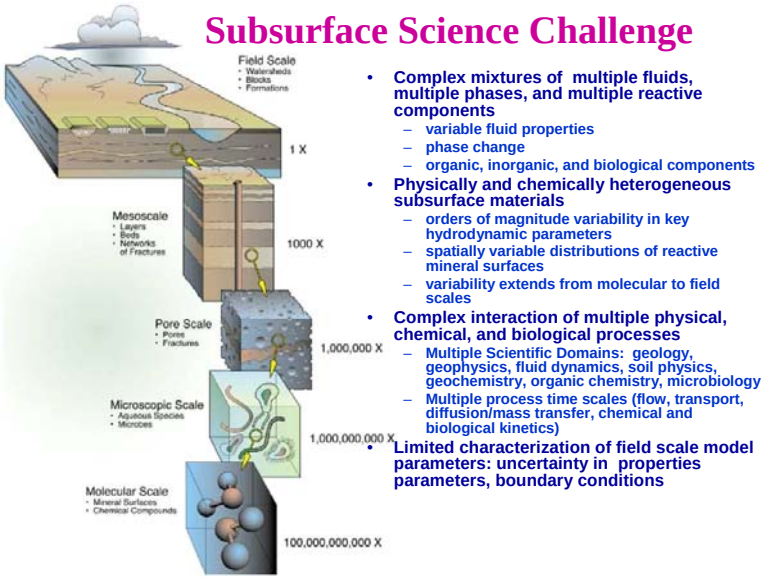
understanding of fundamental physics. A successful program in lattice QCD would lead to calculations of the decay properties of strongly interacting particles to an accuracy needed to make precise tests of the Standard Model. It would enable the determination of the phase diagram and equation of state of the quark-gluon plasma at high temperatures and at baryon densities relevant to heavy-ion collision experiments, and it would provide a quantitative understanding of the internal structure of strongly interacting particles. Finally, a successful lattice QCD program would develop the tools needed to study strong coupling field theories in general; so, when a more encompassing theory than the Standard Model is developed, high energy and nuclear physicists will be able to extract its predictions. Success in all of these areas is possible in the next decade provided the computational resources outlined above become available.

Subsurface Transport and Fate: Environmental Stewardship and Health

Liquids and gases move through rock and sediment with a rich variety of complex behaviors that are dependent on site-specific physical, chemical, and biological conditions. Examples include the infiltration of snowmelt into vegetated soils, the flow of groundwater to drinking water wells, the formation and extraction of oil and gas deposits, and the biodegradation of contaminants. In each case, the migration and mobility of the components of interest are strongly controlled by interactions with spatially variable subsurface materials. The study of subsurface transport and fate seeks to understand these interactions in terms of multiple physical, chemical, and biological processes.

Impact on Science and Society

Exposure to contaminated soil and groundwater is a real concern for many communities in the United States. An accurate assessment of these threats and the design of effective remedies are critical to the protection of human health and the environment. However, the inherent complexity of the subsurface, coupled with a limited ability to observe processes and interactions as they occur, has proven to be a formidable obstacle. In these situations, predictive simulation of field-scale contaminant behavior is the only viable methodology for assessing risk and engineered remedies in the context of the hydrological, geochemical, and microbial processes operating in the subsurface. Difficulties associated with subsurface complexity are not unique to environmental remediation. The engineered repository for commercial nuclear waste from civilian energy production at Yucca Mountain, the geologic sequestration of carbon to reduce the buildup of atmospheric carbon dioxide, and the characterization and extraction of oil and gas deposits, are notable examples of subsurface applications that require an understanding of multiple physical and chemical processes operating on complex mixtures in strongly heterogeneous subsurface materials (see figure). A critical need for all these application areas, is the reliable prediction of field scale subsurface behavior that will be the basis for decisions on environmental stewardship and the protection of human health with long-term implications for U.S. environmental and energy security. This has been the driver for DOE's leadership in state-of-the-art subsurface science and simulation.



Computational modeling and simulation can provide the impetus for transformational improvements in our understanding and prediction of subsurface behaviors that will enable the development of more efficient, cost-effective engineering strategies. Advanced subsurface simulation technology can accommodate new knowledge without compromising the number and detail of coupled processes, spatial resolution, or the simulation duration. High computational performance and large memories can simultaneously address these requirements in the context of 1) multiscale variability in material properties, and 2) uncertainties in conceptual process models and parameters. Advanced computing is, thus, a necessary component in the development of reliable, scientifically defensible predictions of engineering performance and risk in the subsurface.

Comprehensively detailed simulators, universally used by the various application areas in subsurface science to understand and predict subsurface behaviors, are heavily dependent on observed data. Historically, however, field observations to determine spatially distributed model parameters have been sparse and inadequate due to limitations in the ability to characterize large domains of heterogeneous subsurface materials and the mismatch in small–scale observations and their representation at larger modeling scales. This scenario often resulted in large uncertainty in predictions of contaminant and nutrient fate and transport, carbon cycling, climate change, and risk assessment. This long-standing issue in subsurface science is in contrast to other disciplines that rely less on characterization data because 1) models are based on first principles (e.g., molecular dynamics, high-energy physics, astrophysics), or 2) characteristics of the model domain are well-understood and/or well-controlled (e.g., hydraulics, combustion).

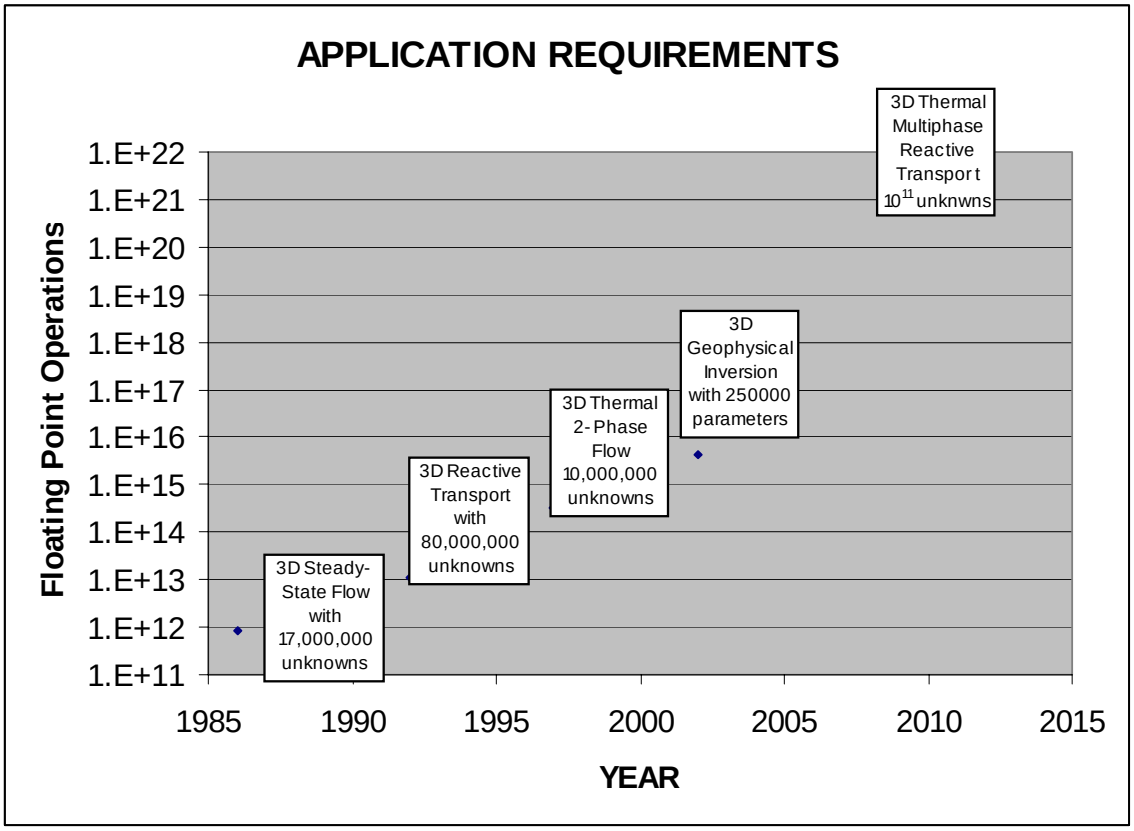
In response to the limited availability of data, the role of computational simulation in the subsurface goes well beyond predicting natural and engineered behaviors. Simulation is essential to the characterization of model parameters, interpretation of experimental observations from the laboratory and the field, testing of conceptual process models and hypotheses, and the identification of uncertainty in model predictions. In this respect, the concept of the three pillars of science – theory, experiment, simulation – is truly applicable in subsurface science.

Scientific Opportunities

Subsurface science is rapidly approaching a convergence of scientific and technological advancements that will significantly accelerate progress on long-standing difficulties with the determination of field-scale process models and parameterizations in highly heterogeneous geologic materials. The research efforts of many investigators on subsurface scaling issues is now being complemented by 1) increasingly comprehensive and sensitive measurement technologies for chemical, geophysical, and microbiological properties, and 2) increasingly powerful and robust computational and modeling technologies. This has bolstered a common theme among the subsurface application areas, which is the study of behaviors at more fundamental length scales with the goal of identifying mechanistically sound upscaling of process descriptions and parameterizations to the field scale. Ongoing research covers length scales from molecular to field:

- Molecular (10^{-12} to 10^{-9} m) – *e.g.*, chemical reactivity at the mineral - water interface
- Microscopic (10^{-9} to 10^{-6} m) – *e.g.*, microbiological immobilization of metals
- Pore (10^{-6} to 10^{-3} m) – *e.g.*, transport-limited mass transfer
- Meso (10^{-3} to 10^{0} m) – *e.g.*, formation of residual organic nonaqueous phases
- Field (10^{0} to 10^{3} m) – *e.g.*, multiphase, multicomponent flow and reactive transport; multiprocess inverse modeling; predictive uncertainty for risk assessment; multiphysics coupling to the regional hydrologic system

Each nominal length scale interval covers a range of approximately three orders of magnitude. With three spatial dimensions and one temporal dimension, this requires on the order of 10^{12} grid cell-time steps. Current high-end capability computing can achieve this level of resolution only if the physical models used in the simulations are highly simplified. As a consequence, compromises in resolution, dimensionality, and/or process detail must be made in current simulations. While the steady advance in high-end computing performance has enabled a progression of increasingly detailed simulations at each nominal length scale interval (see chart), our ability to bridge these intervals has



been limited by the resolution of the measurement technologies and the ability of the computational simulators to accommodate complex coupled process models in highly resolved heterogeneous subsurface materials.

Recent advances in measurement and characterization technology (*e.g.*, NMR, X-ray synchrotron, neutron scattering, electromagnetic and seismic geophysical methods), however, are providing unprecedented detail on structures and processes in the

subsurface environment. Within the next 5 to 10 years, it is expected that three (3) orders of magnitude of resolution in each spatial dimension will be routinely achieved in experimental studies for *all* fundamental length scale intervals (*i.e.*, molecular, micro, pore, and meso scales). Simultaneous advances in high end computing are enabling new lines of research based on comprehensively detailed process simulators that can use these high-resolution data sets as testbeds to develop new theories and process models that will ultimately lead to more reliable predictions of field scale behavior. Assuming a four-dimensional problem (3 spatial dimensions + 1 time dimension), a thousand-fold increase in computational performance equates roughly to a 5 to 6 times increase in grid resolution for each spatial dimension. For complex, coupled process simulations, this would result in grid resolution that is also approaching three orders of magnitude of resolution in each spatial direction.

The significance of the convergence of experimental and computational technologies is that three orders of magnitude of resolution in each spatial dimension (10^9 grid cells) represents a unique opportunity to begin to link and integrate research at different conceptual length scale domains (*e.g.*, molecular to microscopic, pore to meso, *etc*.). This is tantamount to a direct determination of how processes at one scale affect those at a higher scale ("upscaling") and will allow the development of reliable subgrid parameterizations for use at the larger length scales. Resolving the long-standing subsurface "upscaling" problem would be a major scientific breakthrough in subsurface science, the principal outcome of which would be a significant reduction in the uncertainty of predictive simulations.

A thousand-fold increase in computational simulation capability could also be used for scientific advances at a single scale. These would include:

- Simulating regional ecological impacts of climate change by coupling groundwater, vadose zone, watershed, river, meteorological, and ecological process models.
- Simulating long-term, large-scale, three-dimensional, high-resolution, three-phase, multifluid flow and multicomponent reactive transport.
- Estimation of parameters for use in large-scale, long-term, 3-D high-resolution, multicomponent, multiphase, multiphysics simulations.
- Stochastic simulation of conceptual models and realizations to quantify uncertainty in model predictions.
- Quantifiable inversion of 3-D, real-time, multisensor data.
- Pore-scale simulation of multiple domains in a 10 cm cube.

Research Issues

While high-end computing can advance scientific understanding and enable advanced subsurface simulators to increase the usefulness of experimental and observational data, increased investment in this area alone will not produce the desired breakthroughs. Advanced subsurface simulation must be an integral part of a larger, holistic science and technology agenda that develops new knowledge of the subsurface environment from:

- interdisciplinary laboratory and field investigations of subsurface processes at all fundamental length scales,

- incorporation of the development as well as the application of comprehensive, detailed simulators in research programs to address process and property complexity,
- high-resolution laboratory and field data sets to provide testbeds for developing and incorporating new knowledge and capabilities into advanced subsurface simulators, and
- multiscale experimental facilities for studies of process and property "upscaling" from fundamental length scales to the field.

In subsurface modeling and simulation, specifically, additional research is needed on representations and parameterizations of subsurface phenomena at one scale that can be reliably used at higher levels (*i.e.*, "upscaled"). In addition, new research is needed at the pore scale to identify the effect of evolving biogeochemical microenvironments associated with multi-region (multi-porosity and multi-permeability) domains on the mobility of reactive components. This will require the development of new capabilities for existing pore-scale simulators to address spatially detailed alternatives to the simulation of bulk surface processes (*e.g.*, adsorption and redox reactions).

In regard to the smaller fundamental scales of research in computational surface science, it will be important to identify which of the observed phenomena could potentially contribute to increasing the fidelity of field-scale models. Sensitivity and uncertainty analyses are needed to identify specific physical processes and associated parameterizations that contribute most to the uncertainty and variability in predictions at the field level. Multiphysics couplings such as the linkages between groundwater, surface water, and the atmosphere in the regional climate analyses are generally ad hoc, not robust and not scalable. Substantial efforts are needed to formulate and analyze the underlying mathematical models and corresponding numerical schemes.

Although most subsurface simulators running on multiprocessor computers show moderately good parallel scalability, they also obtain a low percent of peak performance. In addition, few have been tested on 500 or more processors. It is unclear how efficiently the Newton/Krylov or other solvers will perform on problems requiring 10,000 processors. Furthermore, the opportunity for load imbalances in some reactive transport schemes is also an open problem in simulations employing 10,000 or more processors.

Technology Barriers

High-end computing is underutilized by the subsurface science community. A perception is that it takes an inordinately large amount of time to develop, test, and debug a parallel program to solve a problem of interest. From this perspective, the most important technology considerations are software tools that allow scientists to focus on subsurface science, not computer science:

- Portable, parallel programming tools with higher levels of abstraction that insulate subsurface science software developers from low-level details of message passing as well as from changes in architectures, operating systems, and compilers.
- Frameworks that can accommodate accurate, robust, efficient, portable, scalable and interoperable component technologies (adaptive meshing and discretization, nonlinear

solvers, sparse linear solvers, transport algorithms, minimization/optimization algorithms, data partitioning, and visualization).

- Environments that allow loose or tight coupling between simulators, *e.g.*, multiphase flow coupled to reactive transport, or compositional models coupled to geomechanics models.
- Problem solving environments that make models accessible to researchers with varying levels of computing skills. Such an interface tool would allow users to easily run simulations on the high performance machines directly from a personal workstation, without having to manually login and enter commands. The interface should display results as the model is running and provide options for steering the simulation. Analysis and visualization tools should be directly accessible from the interface.
- Portable parallel debuggers and performance analysis tools.
- Useable, state-of-the-art subsurface simulation software.
- Tools for managing and analyzing large data sets.

Other considerations include:

- Stable and consistent operating systems and compilers.
- Environment and infrastructure for collaborative use of these resources.

The principal computational effort for most subsurface simulators is in the solution of the nonlinear system of equations that results from the implicit discretization of the governing partial differential equations. Thus, research to gain efficiencies in global nonlinear convergence, formation of Jacobians, and solution of large, sparse systems of linear equations is of paramount importance. Additional research is also needed on mathematical and statistical alternatives that use state-of-the-art uncertainty analysis methods such as smart sampling, response surface modeling, and sensitivity analysis (including direct sensitivity calculations using adjoint equations and automatic differentiation) to facilitate uncertainty quantification for both forward and inverse problems in subsurface science.

As demonstrated by the low percentage of peak performance, which is controlled by access to floating point operands, not their arithmetic processing, memory bandwidth and latency continues to be a substantial hardware barrier. Although it may be possible, with a significantly larger investment in research, to develop more cache-friendly algorithms for some parts of the subsurface simulations, other alternatives would be to 1) buy computers that take full advantage of the long vectors that are characteristic of the grid-based solution schemes (*e.g.*, the NEC SX-6 and the Cray X series), or 2) fund research into approaches that promise to increase memory bandwidth and decrease memory latency in commodity-based supercomputers.

Resources Required

One of the key observations from previous efforts to use high-end computers for subsurface simulations is that simply providing parallel computing cycles and tools will not automatically result in scientific progress. Subsurface scientists need to be driven by grand challenge research imperatives to become hands-on participants with mathematicians and computational scientists in the design, development, and application

of advanced subsurface simulation software. And, they need to be supported to do so. This type of interdisciplinary research will provide the foundation for a critical mass of next generation subsurface scientists to access, develop, and use advanced subsurface simulators for large-scale calculations.

Sustained funding is needed to maintain the operability, usability, and efficiency of developed parallel code – one-time investments cannot realistically be expected to result in end user support. Parallel subsurface simulation software must be made available on platforms where its performance has been optimized not only for efficiency, but also for usage. Furthermore, all component software, tools, and utilities must be useable and supported by knowledgeable staff. Users should expect to develop and stage parallel software applications on seamlessly compatible hardware platforms dedicated to large scale subsurface capacity computing. The capability computing described earlier for subsurface science must be supplemented and supported by considerable use of capacity computing cycles. Capacity computing cycles will be used to assure the quality of the input data and problem specification; develop, test, and debug new algorithms and process couplings; perform proof-of-principle calculations; stage preliminary calculations leading up to full capability simulations; and post-process results. The reality is that subsurface science's need for capacity computing resources will exceed its need for capability computing resources for sometime to come.

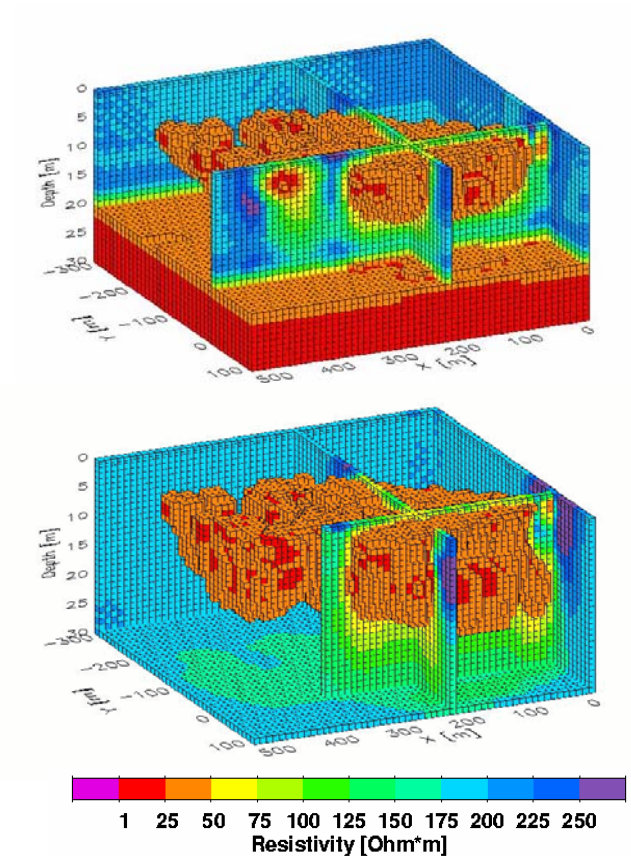
Metrics of Success

There are many ways to measure the successes of a major new investment in computational subsurface science. Some of the measures of success are advancements in computational subsurface science:

- Coupled process simulations with three orders of magnitude of resolution in every spatial dimension.
- High-resolution simulations of three-dimensional, three-phase, multi-fluid flow coupled with multicomponent reactive transport.
- Geophysical inversion of ~1 million parameter problems in less than a day.
- Some of the measures of success will be advances in subsurface science, including:
- New knowledge and insights from high-resolution simulations of three-dimensional, complex, coupled processes at scales ranging from the molecular to the field.
- New field-scale process models and parameterizations from research programs focused on the bridging of previously distinct spatial scales of research.
- More accurate and detailed characterization of subsurface properties through the incorporation of multiple data types at multiple length scales.
- And, finally, some of the measures of success will be direct benefits to society, including:
- More reliable predictions of the spread of underground contaminant plumes, assessments of environmental and human health hazards, and evaluation of cleanup strategies.
- Full multiphysics model coupling of groundwater, vadose zone, surface water, watershed, and atmospheric processes for the assessment of regional climate impacts.

Sidebar #1

Three-dimensional electromagnetic imaging/inversion has shown great potential in hydrological and hazardous waste site characterization, as well as in oil and gas exploration. Knowledge of the subsurface electrical conductivity, estimated through the inversion process, is extremely important, since it is directly related to subsurface fluid content and its spatial distribution. Thus maps of electrical conductivity can be utilized in hydrological investigations, oil and gas exploration, and more recently in reservoir and environmental site characterization. Inversion also has important applications in mineral and geothermal exploration and general geologic mapping. It has been used successfully to map subsurface transport pathways for contaminants, to delineate buried metallic waste, to define the extent of waste pits, and to determine the safety of proposed long-term waste disposal sites. Application to safe CO_2 sequestration in gas-depleted reservoirs also looks promising, and is a top priority with the U. S. Department of Energy for mitigating global warming. In spite of these successes, 3D inversion of data continues to be a challenging endeavor, requiring large-scale model parameterizations and data sets necessary for realistically imaging complex 3D geology. As a result, months of large-scale parallel computing time are needed to appraise solution quality and access solution non-uniqueness that arise from the inversion process.



As an example, the top image demonstrates 3D EM inversion methodology on a field data set acquired over a buried waste site in Cologne Germany. Over 4800 data points collected on multiple measurement profiles have been simultaneously inverted for more than a quarter of a million subsurface electrical parameters (resistivity, which is the reciprocal of conductivity). The resulting image clearly map the buried waste pit, and indicates the depth of the base of the pit to be between 12 to 15 m, which correlates with borehole logs taken at the site. This image could be interpreted that contamination leaking from the pit into deeper geological horizons has been minimal. However, when the known host background is altered to a half-space, the image of the base of the pit changes considerably (bottom image).

Now the base of the pit extends down towards, 20 m depth. Thus, without prior information (constraints), knowledge of the pit base would be difficult to ascertain in this example. To achieve the results presented, several months (wall clock time) were needed using 256 processors on the terascale ASCI-Red machine at Sandia National Laboratories. Multiple inversion models had to be evaluated using different starting models and stabilization parameters and assumptions on the data noise.

Faster solution times for 3D geophysical inversion will be necessary for the full potential of 3D subsurface imaging technologies to be realized. This translates into fewer and faster non-linear inversion iterations as well as a substantial reduction in the computational burden required for the solution of the forward modeling problem for computing predictive data, without sacrificing accuracy for speed. In the non-linear inverse problem we are considering, hundreds to thousands of forward-modeling applications typically arise, where each application involves the solution of a partial differential equation describing the physics of the measurement. As previously shown, many of theses solutions are required to fine tune stabilization parameters, quantify data noise and to access the unique properties of the solution through an appraisal process. Not only do significant improvements in solution speed translate into more reliable 3D images, they also advance the prospect of real-time 3D earth imaging in the not too distant future, given the anticipated improvements in parallel computer hardware and algorithms. An example of real time imaging that has significant potential is imaging ahead of the drill bit in the exploration for hydrocarbons. Here real time imaging could save the oil and gas industries the billions of dollars that are wasted due to well blowouts and bypassed oil.

Sidebar #2

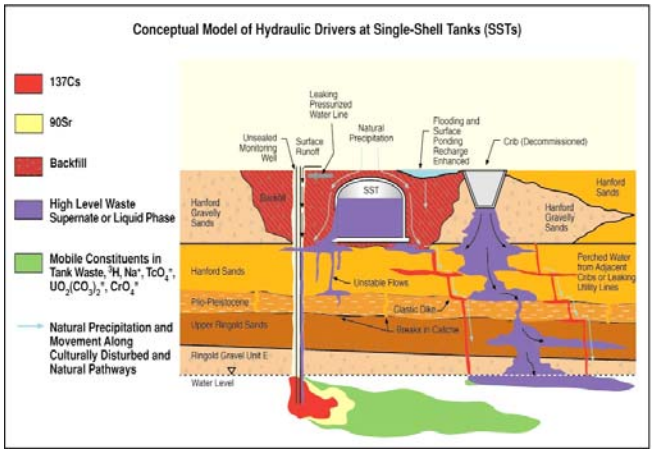
Migration of High-level Radioactive Waste from Leaking Subsurface Tanks

During the 1960s, million-gallon subsurface tanks in the SX Tank Farm of the Hanford Site held high-level radioactive waste (HLW) that was among the highest in temperature (up to 200 ºC), density (up to 1.6 g/ml), radioactivity (> 1.75 Ci/L), pH (>14), and ionic strength (e.g., Na = 20 M) in the nuclear weapons production complex. Gamma logging of boreholes around these underground storage tanks confirmed that HLW from historical leaks had entered the sediments 50 meters above the water table aquifer. The most radioactive sediment samples were measured near Tank SX-108 in the 200 West Area. Uncertainty about future groundwater contamination from this tank prompted a retrospective investigation of conditions in 1966 when the leak was thought to have occurred.

The scientific and computational challenge of the investigation was to understand the behavior of contaminants released to the subsurface environment in the context of:

- extreme, concentration-dependent physical and chemical properties of the waste mixture,
- thermally driven liquid and gas phase transport with evaporation and condensation,
- large-scale variability in layering and geometry of unconsolidated sediments accompanied by smaller scale textural heterogeneities, and
- dissolution of sediments by the caustic waste and the subsequent formation of secondary minerals.

Tight coupling of highly detailed process models is necessary to account for dynamically evolving subsurface conditions: temperature- and concentration-dependent fluid and solid properties, mineral dissolution- and precipitation-induced alterations to hydraulic and reactive porous media properties, and the impact of phase changes on saturation-dependent fluid flow.



The simulated subsurface system must resolve a large range of time scales to address leak events on the order of days as well as the regulatory time frame for long-term radionuclide risk of 10,000 years. Similarly, a large range of length scales are necessary to account for the sensitivity of unsaturated flow to centimeter-scale sediment features, while maintaining a sufficiently large three-dimensional domain to address the thermal impact of the adjacent tanks, and the lateral liquid and gas migration along geologic layers. Finally, the number of unknowns (*e.g.*, chemical species, minerals, phase pressures and saturations, sediment properties, liquid and gas properties) in each grid cell is typically large, exceeding 50 in some cases.

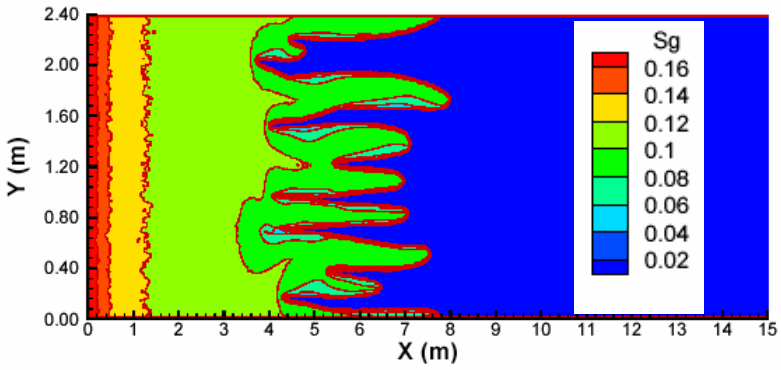
Two years ago, advanced subsurface simulators could not address the spatial, temporal, and process detail required to adequately reconstruct the leak event and predict the volume and distribution of contaminants. Consequently, only parts of the problem were simulated with significant simplifying assumptions. Even with simplifications, the hydrothermal multiphase simulation alone required over 4×10^{15} floating point operations to model a one million-grid cell depiction of the entire tank farm for the 50-year operational history. While this simulation is among the largest for a complex subsurface field problem, the average grid resolution was still over one meter. With current modeling technology, fully coupled flow, transport, and reactions for a 10,000-year predictive simulation at ten times finer spatial and temporal resolution would require over 10×10^{21} floating point operations. Advanced subsurface simulators need to be designed, developed and tested for problems of this magnitude.

Sidebar #3

Reduction of Global Warming through Geologic Sequestration of Carbon Dioxide

The geologic sequestration of carbon is a promising approach for the reduction of atmospheric accumulation of greenhouse gases generated by the combustion of fossil fuels. Carbon, primarily in the form of carbon dioxide (CO_2) is injected into subsurface formations instead of being released into the atmosphere. Porewater or brine displaced during CO_2 injection is subject to hydrodynamic instability (viscous fingering and gravity override) because CO_2 is always less dense than water at typical subsurface temperatures and pressures; it also has lower viscosity. Accordingly, the displacement front will be uneven, with CO_2 advancing in fingers while bypassing a considerable fraction of formation volume. Understanding the behavior of these fingering instabilities is critical to successful CO_2 injection design. If large-scale fingers are allowed to develop (diameter of meters to tens of meters), sweep efficiency and CO_2 storage capacity may be considerably reduced relative to what would be obtained for (nearly) uniform displacement. On the other hand, if characteristic finger widths are small, decimeters or less, then the adverse impact of the viscous instability on the displacement process would be effectively limited.

Mathematical models for the physical and chemical processes controlling subsurface CO_2 behavior have been implemented in advanced numerical simulators. The most complex simulations, coupling multiphase fluid flow, heat transfer, chemical interactions between rocks and fluids, and mechanical changes, are typically performed in two-dimensions to achieve higher resolution of the fingering phenomenon. In a recent simulation designed to determine the spatial scale of fingering for typical CO_2 injection conditions, CO_2 gas was injected into a two-dimensional subsurface formation represented by more than 200,000 grid blocks and a spatial resolution of 1.5 centimeters. The high-resolution grid was considerably finer than the width of the fingers to adequately capture the displacement behavior. The predicted density displacement front (see figure) demonstrates that finger widths of the order of 1 dm can be expected under the modeled condition and, more importantly, are not artifacts arising from the grid resolution. The one year simulation required 0.1 wall clock days using 32 processors.



CO_2 sequestration raises other issues requiring far larger modeling studies than can be done with current computational capabilities. An important example is the onset and evolution of convective instabilities in the aqueous phase. When CO_2 dissolves in water, the small 1-2% increase in water density may give rise to convection in the aqueous phase, which would distribute CO_2 over the entire thickness of the permeable interval, as opposed to free-phase CO_2 just migrating near the top of the permeable interval. Preliminary calculations indicate that the onset and evolution of convective flow are sensitive to grid resolution. Obtaining grid-converged results for convective behavior in three dimensions will require on thousand times larger computational capabilities than are presently available. Even greater computational demands arise when the coupling of rock-fluid interaction and geomechanics to fluid flow involve multiple space and time scales. For example, hydrodynamic instabilities, mass transport, and rock-fluid interactions entail processes operating on small spatial scales, while pressurization of formation fluids and stress and strain changes occur on large space scales. Pressurization effects and mechanical deformation occur rapidly, while rock-fluid interactions may take hundreds of years to play out. Convective instabilities may have characteristic time scales of thousands of years. Modeling of multiscale processes and of coupling between fluid flow, chemistry, and geomechanics remains difficult and would be greatly enhanced with more powerful computational capabilities.

A quantitative understanding of the interplay between hydrodynamic instabilities and ever-present heterogeneities is needed for assessing (1) sweep efficiency and CO_2 storage capacity, (2) pressurization effects during CO_2 injection, and (3) constitutive relations (relative permeabilities) in the water- CO_2 system. The field is ready to undertake far bigger simulations to address these issues in the context of multiple space and time scales, and coupling between fluid flow, chemistry, and geomechanics. Key requirements are access to ultrascale computers, simulators that can effectively use 10,000 processors on these computers, and scientific support for the characterization of field scale processes and properties.

Mathematical Tools for Large-Scale Simulation

Mathematics is the bridge between physical reality and computer simulations of physical reality. The starting point for a computer simulation is a mathematical model, expressed in terms of a system of equations and based on scientific understanding of the problem being investigated, such as the knowledge of forces acting on a particle or a parcel of fluid. For such a model, three types of mathematical tools may be brought to bear in order to produce and use computer simulations.

Model Analysis. Although the mathematical model is motivated by science, it must stand on its own as an internally consistent mathematical object in order for its computer representation to be well defined. For that reason, a number of issues must be resolved regarding the mathematical structure of the model. Do the equations have a unique solution for all physically meaningful inputs? Do small changes in the inputs lead only to small changes in the solution? Or, as an indication of potential trouble, could a small uncertainty be magnified by the model into large variability in the outcome?

Approximation and Discretization. For many systems, the number of unknowns in the model is infinite, as when the solution is a function of continuum variables such as space and time. In such cases, one must approximate the infinite number of unknowns with a large, but finite number of unknowns in order to represent it on a computer. The mathematical issues for this process, called “discretization,” include the extent to which the finite approximation better agrees with the solution to the original equations as the number of computational unknowns increases, and the relationship between the choice of approximation and qualitative mathematical properties of the solution, such as singularities.

Solvers and Software. Once one has represented the physical problem as the solution to a finite number of equations for a finite number of unknowns, how does one best use the computational resources to calculate the solution to those equations? Issues at this stage include the development of optimally efficient algorithms and the mapping of computations onto a complex hierarchy of processors and memory systems.

While these three types of tools represent distinct mathematical disciplines, these disciplines are typically employed in concert to build complete simulations. Choice of appropriate mathematical tools can make or break a simulation code. For example, over a four-decade period of our brief simulation era, algorithms alone have brought a speedup in excess of a factor of a million to the task of computing the electrostatic potential induced by a charge distribution, typical of a computational kernel found a wide variety of scientific problems. This algorithmic speedup represents an improvement comparable to that obtained from the hardware speedup due to Moore’s law over the same time period. The series of algorithmic improvements producing this factor are all based on a fundamental mathematical property of the underlying model, namely, that the function describing electrostatic coupling between disjoint regions in space is very smooth. Expressed in the right way, this coupling can therefore be resolved accurately with little

computational effort. The various improvements in algorithms for solving this problem came about by successive redesigns of the discretization methods and solver algorithms to better exploit this mathematical property of the model.

Another trend in computational science is the steady increase in the intellectual complexity of virtually all aspects of the modeling process. Scientists contributing to this report identified and developed eight crosscutting areas of applied mathematics for which research and development work will be needed to accommodate the rapidly increasing complexity of state-of-the-art computational science. They fall into the following three categories.

Managing Model Complexity. Scientists want to use increasing computing capability to improve the fidelity of their models. For many problems, this means introducing models with more physical effects, more equations, and more unknowns. In *multiphysics modeling*, the goal is to develop a combination of analytical and numerical techniques to better represent problems with multiple physical processes. These techniques may range from analytical methods to determine how to break a problem up into weakly interacting components, to new numerical methods for exploiting such a decomposition of the problem to obtain efficient and accurate discretizations in time. A similar set of issues arises because many systems of interest have processes that operate on length and time scales that vary over many orders of magnitude. *Multiscale modeling* addresses the representation and interactions of behaviors on multiple scales so that results of interest are recovered without the (unaffordable) expense of representing all behaviors at uniformly fine scales. Approaches include the development of adaptive methods, namely, discretization methods that can represent directly many orders of magnitude in length scales that might appear in a single mathematical model, and hybrid methods for coupling radically different models (continuum versus discrete, or stochastic versus deterministic) each of which represents the behavior on a different scale. *Uncertainty quantification* addresses issues connected with mathematical models that involve fits to experimental data or are derived from heuristics that may not be directly connected to physical principles. Uncertainty quantification uses techniques from fields such as statistics and optimization to determine the sensitivity of models to inputs with errors, and to design models in order to minimize the effect of those errors.

Discretizations of Spatial Models. Many of the applications described in this document have as core components of their mathematical models the equations of fluid dynamics or of radiation transport, or both. *Computational fluid dynamics* and *transport and kinetic methods* have as their goal the development of the next generation of spatial discretization methods for these problems. Challenges include the development of discretization methods that are well suited for multiphysics applications, without loss of accuracy or robustness. *Meshing methods* specifically address the process of discretizing the computational domain, itself, into a union of simple elements. This process is usually a prerequisite for discretizing the equations defined over the domain. The area includes the management of complex geometrical objects arising in technological devices, as well as in some areas of science, such as biology.

Managing Computational Complexity. Once the mathematical model has been converted into a system of equations for a finite collection of unknowns, it is necessary to solve the equations. The goal of efforts in the area of *solvers and fast algorithms* is to develop algorithms for solving these systems of equations that balance computational efficiency on hierarchical multiprocessor systems, scalability (the ability to use effectively additional computational resources to solve increasingly larger problems), and robustness (insensitivity of the computational cost to details of the inputs). An algorithm is said to be "fast" if its cost grows roughly only proportionally to the size of the problem. This is an ideal algorithmic property that is being obtained for more and more types of equations. *Discrete mathematics and algorithms* make up a complementary set of tools for managing the computational complexity of the interactions of discrete objects. Such issues arise, for example, in traversing data structures for calculations on unstructured grids, in optimizing resource allocation on multiprocessor architectures, or in addressing bioinformatics problems that are posed directly as combinatorial problems.

Computational Fluid Dynamics

Computational fluid dynamics (CFD) is concerned with the development of discretization methods for fluid dynamics and related continuum physics problems, such as magnetohydrodynamics and plasma dynamics. Physically, such problems exhibit a variety of complex behaviors, such as advective and diffusive transport, complex constitutive properties, discontinuities and other singularities, multicomponent and multiphase behaviors, and coupling to electromagnetic fields. Mathematically, these problems are represented as solutions to nonlinear partial differential equations that are functions of time, and of physical space variables (up to three variables) or phase space variables (up to six variables).

The nonlinearity of the PDE makes the rigorous mathematical analysis of the stability and accuracy of such numerical methods for these problems very difficult to carry out. For that reason, the most successful approaches to the design of discretization methods have been through a combination of formal mathematical tools and physical reasoning. Techniques include the following:

- Formal mathematical methods for analyzing numerical methods, such as truncation error analysis for general systems and the stability analysis of simplified model problems
- Methods to reproduce physically important asymptotic behaviors, such as discrete traveling wave structures for shocks, local conservation, or symplectic dynamics for kinetic problems
- Methods that exploit the mathematical structure of the underlying PDE in the discretization to obtain efficient and stable methods. Examples include exploiting the locality for hyperbolic PDE and local regularity for elliptic and parabolic PDE to obtain scalable solution algorithms or the use of well-posedness of initial or boundary value problems to formulate adaptive mesh refinement algorithms

Beginning with the work of von Neumann in the 1940s, this combination of mathematical and physical reasoning has been highly successful at providing state-of-the-art algorithms for computational fluid dynamics. Innovations include high-resolution methods for hyperbolic conservation laws; projection methods, and artificial compressibility methods for low Mach number fluid flows; adaptive mesh methods; and a variety of Lagrangian particle methods for representing fluid and kinetic problems. These methods are widely used throughout the scientific community. For the most part, they have also been the result of collaborations between applied mathematicians and applications scientists in the context of solving specific scientific problems.

Impact on Applications

CFD comprises two broad classes of discretization methods. In grid-based methods, such as finite-difference or finite-element methods, the discretization is derived using a finite-dimensional approximation to the solution, such as local polynomial interpolants, to derive discrete equations for solution values defined on a grid. For particle methods, the finite-dimensional spatial approximation is given as a sum of smoothed δ-functions

whose centers move along Lagrangian trajectories. Typically, grid-based methods have been used for problems in physical space, while particle methods have been used for problems in both physical space and phase space. Within these methodologies, we have identified the following set of algorithmic design issues that constitute critical technology requirements in CFD for the applications discussed here.

Adaptive Mesh Refinement (AMR). In many applications, the spatial resolution required to maintain a given level of accuracy, is a function of space, time, and the solution itself. This observation has led to the development of adaptive methods for allowing run-time variation in the local density of computational degrees of freedom in order to maintain a given level of accuracy. Adaptive methods can be based on block-structured grids, in which space is discretized by a union of logically rectangular grids, and unstructured-grid methods, in which the discretization of space is given as a collection of general polyhedra. Both approaches have sufficient geometric flexibility to permit the necessary local variation in resolution. Almost all of the CFD applications described in this document require some form of AMR to resolve multiple spatial scales. This situation may occur because of localized flow features (thin fronts in astrophysics and combustion, sawtooth instabilities and reconnection zones in plasmas, localized particle beams in accelerators) or because of the need to resolve geometric features (mountain ranges in atmospheric flows, cell membranes in biology) or to resolve a cascade from large to small length scales over time (gravitational collapse in astrophysics). Hence, we view AMR as a fundamental context in which all of the other technology barriers must be resolved. However, the presence of adaptive grids makes considerable demands on the other technologies discussed here.

Implicit Temporal Discretizations. For time-dependent problems, the options range from the use of completely explicit methods, for which the solution at the new time is a local function of the values of the solution at the old time, to fully implicit methods, for which the solution at the new time is the solution to a general nonlinear system of equations coupling all of the new-time unknowns. In order to be stable, explicit methods require the use of a time step sufficiently small to resolve all of the dynamical processes representable on the grid. Implicit methods allow the use of much larger time steps and are appropriate where a subset of those dynamical processes have relaxed to a quasi-steady condition. While the cost per time step is much larger for a fully implicit method than for an explicit method, explicit methods often must take much smaller time steps than are necessary to resolve the temporal variation of the solution. An intermediate approach is to use semi-implicit methods, in which the evolution operator is broken down into substeps, each of which may be computed by using an implicit method or an explicit method. Such methods allow one to treat processes that lead to the most onerous restrictions on the time step to be treated implicitly, while treating other processes explicitly. Further, implicit coupling of the unknowns can arise from eliminating fast time scales by asymptotic analysis. The incompressible flow equations, which requires the solution of an elliptic PDE for the pressure at each time step, arises from the elimination of fast, weak sound waves in the limit of vanishing Mach number. New implicit methods are needed, ranging from fully implicit methods for tokomak modeling in order to resolve the slowly-varying evolution processes; implicit methods for Maxwell's equations arising

in kinetic models for beam dynamics in accelerators; and new semi-implicit methods for combustion in closed containers. New semi-implicit methods are required for spatial modeling of cells, which involve incompressible fluid flows with complex constitutive behavior.

Hybrid Particle and Grid Methods. Many particle methods are in fact hybrids of particle and grid methods. Particle-in-cell methods for electromagnetics involve the use of a particle representation of the distribution of charged particles, while using a grid representation for solving the electromagnetic field equations (Poisson or Maxwell), with a physical space forcing obtained from averaging the particle distributions over the velocity variables at each point in space. In order to compute the forces at the particle locations, the electromagnetic fields are interpolated from the grid. Several applications require new hybrid particle and grid methods. In plasma modeling, energetic nuclei such as α particles have mean free paths that are much larger than the fluid length scales and must be treated using kinetic equations, while other species, such as the electrons, are strongly collisional and can be treated by using a fluid representation in physical space. Similar considerations arise in biological cell modeling, in which some species are at sufficiently low concentrations that they cannot be represented as continua in physical space. It is also possible that the regions that must be treated by using kinetic models may be a function of space and time; in that case, adaptive hybridization of kinetic and fluid models may be appropriate.

Grid-based Discretizations of Kinetic Problems. Traditionally, kinetic problems in phase space, such as beam dynamics in accelerator modeling, and gyrokinetic turbulence in plasma physics, have been solved using Lagrangian particle methods. Such methods are naturally adaptive, in the sense that computational degrees of freedom are placed only in regions in phase space where the number density is nonzero. In addition, one can discretize the resulting ordinary differential equations for the particle trajectories to preserve the symplectic structure of the original system. However, particle methods are at a severe disadvantage in resolving effects such as beam halos in RF accelerators, in which the number density varies by several orders of magnitude. In that case, the initially uniform particle distribution sufficiently fine to resolve the halo is grossly over-resolved in the beam core. These limitations suggest the use of grid-based methods, combined with AMR, as a more accurate and efficient approach to such problems.

Irregular Boundaries, Anisotropy, and Fronts. Global geometric features of the PDE have an impact on discretization methods in three ways. To model RF accelerators and plasmas in fusion devices such as stellarators, one must impose boundary conditions on irregular surfaces, that is, BCs not corresponding to a constant coordinate value of a globally defined analytical coordinate system. In high-β plasmas, strong inertial forces and enhanced diffusive transport along magnetic field lines lead to strong directional biases in the large-scale dynamics. Thin fronts whose internal structure is orders of magnitude smaller than the other fluid scales of the problem arise in a variety of settings, from flame fronts in premixed combustion and supernova deflagrations to membranes of cells. Typically, such geometric features are represented by a combination of two approaches. One approach consists of specialized gridding techniques that align grids to the geometry, such as body-fitted grids, field-aligned and flux-surface grids, and

anisotropic grids. These techniques are discussed in detail in the gridding section. The second approach is the use of discretization methods that account properly for the boundary conditions at irregular boundaries or fronts, such as volume-of-fluid and cut-cell discretizations, or natural weak formulations on structured or unstructured boundary--fitted grids. For problems with strong anistropies, specialized discretizations are used in conjunction with specialized models that represent the appropriate cancellations analytically. For example, the fundamental fluid dynamics of climate as being driven by small perturbations of a hydrostatically stratified thin layer of fluid is represented by using equations that preserve those properties. Specialized discretizations on high aspect ratio grids are employed that also preserve important invariants of the flow. The algorithmic and software tools available for representing such geometric structures are generally very limited. The most mature technologies are for fixed boundaries, in particular unstructured grid methods. The tools available for moving boundaries and fronts are much more limited, with many of the fundamental algorithmic issues still areas of active research.

Research Issues

In the areas described above, a number of research issues exist. They vary from taking reasonable well-understood algorithmic technology and turning it into software tools, to requiring the resolution of fundamental questions in discretization design.

Implicit Methods with AMR. AMR is a mature and widely used technology for discretizations based on explicit methods for hyperbolic PDE. For methods requiring some degree of implicitness, however, far less software is available, and open algorithmic questions remain. For semi-implicit methods in which the implicit solution method is being applied to elliptic or parabolic PDE, the literature offers a variety of extensions; the principal barrier to developing these ideas further in science applications is the need for robust discretization and solver libraries. Other semi-implicit methods, based on splitting of fast and slow hyperbolic dynamics, need both fundamental algorithm development work and software development. For fully implicit time discretizations, fundamental questions remain regarding the design of AMR discretizations that are stable and accurate and lead to robust and scalable solvers for the resulting nonlinear PDE. Similarly fundamental questions arise for implicit field methods for Maxwell's equations in AMR-PIC calculations.

Multiscale Behavior in Anisotropic Physics. Most adaptive techniques have been developed in the context of isotropic physics, with little experience in the anisotropic cases represented by fusion reactors, climate, and some areas of astrophysics. In these cases fundamental issues remain to be resolved. Can one use isotropic refinement in an anisotropic system, or must the refinement respect the anisotropy? In both the climate and fusion cases, there are cases in which the small-scale behavior is more nearly isotropic, while the large-scale behavior remains anisotropic. Is it possible to change discretizations, or even models, as a function of the scales being resolved, in order to better represent this transition?

Algorithms and Software for Kinetic and Hybrid Formulations. A number of new algorithmic ideas need to be investigated and developed into software for problems with a nontrivial kinetic component. These include hybrid fluid/kinetic methods for nonlinear problems (astrophysics, plasma science, and biology); AMR for electromagnetic PIC (accelerator modeling, plasma science); and grid discretizations for kinetic problems (astrophysics, plasma science, and accelerator modeling).

Tools for Complex Geometries and Free Boundaries. There is a dearth of software tools for problems with complex boundaries and free surfaces. Such tools are a pacing item for problems involving fronts in combustion and astrophysics and arguably are the most important missing capability holding back the development of spatial models of cells in biology. The capabilities required include methods and solvers for elliptic and parabolic free boundary problems; consistent temporal discretization methods for deforming boundaries; and the capability to couple to continuous transport on the boundary, and discrete channels through the boundary.

Higher-Order Discretization Methods. Many of the CFD methods used in real applications are at most second-order accurate in space and time. Preliminary results with a variety of grid-based methods indicate there may be considerable advantages to developing higher-order (fourth-order or higher) methods for a number of the problems described here, including climate, fusion modeling, biological modeling, or for the grid methods in kinetics. To carry this out, one must reconsider a number of the algorithm development issues described above in the context of such higher-order methods. These issues include extension to complex moving geometries and free boundaries; treatment of conservation and adaptivity; robustness in the presence of discontinuities and marginally-resolved gradients; extension to multiphysics problems; and higher-order discretization in time for semi-implicit methods.

Metrics for Success

In this section, we have alluded to a large number of specific requirements in CFD needed in order to move forward in various areas of science. Success in CFD will be to develop the algorithmic and software capabilities to the point where new science is enabled. It would be premature to attempt to enumerate those capabilities as specific milestones. However, it is useful to divide them into four broad categories.

Adding New Algorithmic Capabilities to Existing Software. In astrophysics there is already a substantial algorithmic and software capability in adaptive mesh refinement methods for explicit methods for hyperbolic PDE (e.g. for inviscid compressible flow), with some specialized capabilities for coupling to particle methods. Augmenting those capabilities with the algorithms and software required to implement semi-implicit or fully implicit methods, a broader range of hybrid particle/mesh methods, and front tracking methods would constitute a substantial and measurable success. It would considerably expand the range of scientific problems accessible to this community, while building on their existing experience and software base.

Introducing New Techniques to Established Computational Science Efforts. Both plasma science and climate modeling have identified adaptive grid methods as part of their future needs. One metric of success would be the development of this new capability for those areas, based on the experience (and leveraging off of the software) developed in the application of adaptive grid methods in other fields.

Enabling New Uses of High-End Computing. Relatively few tools are available for spatial modeling of cells in biology, with limited capabilities relative to the richness of the geometric, mechanical, and biochemical processes in cells. The development of a high-performance CFD toolset aimed at cell modeling, in close collaboration with the cell-biology community, would enable the development of new models of spatial behavior of cells, leading to a new mode of scientific investigation in this field.

Development of Shared CFD Infrastructure. Considerable overlap exists in the needs of widely different scientific communities in CFD. Nevertheless, there is little sharing of common infrastructure across these communities. Indeed, for some areas, there is little sharing *within* the community, with each research group writing its own code from scratch. One metric of success for CFD would be the extent to which shared software libraries would be developed and widely used across multiple scientific enterprises.

Discrete Mathematics

We live in a discrete universe: People, computers, even fundamental particles come in integer numbers; and not surprising, computational models in science require the study of discrete objects. Discrete mathematics, or combinatorics, studies the arrangement, grouping, ordering, or selection of a finite number of discrete objects. Discrete algorithms (also called combinatorial algorithms) are computational methods for solving problems in discrete mathematics. In this report, we further restrict ourselves to algorithms that solve discrete problems arising in computational science; we refer to this area as *combinatorial scientific computing*, and in it we include all areas of discrete mathematics that play a role in solving the problems of scientific computing. Graph theoretic, matroidal, and geometric algorithms play important roles in the discretized mathematical models of many problems in scientific computing. The solution of large-scale, sparse systems of linear equations and eigen computations of sparse matrices involve combinatorial problems and algorithms in addition to linear algebra. Algorithms for linear and nonlinear programming, especially with some variables restricted to integer values, can be used to solve facility location and other design problems in scientific computing. String algorithms aid in the solution of sequence analysis problems.

Discrete algorithms are critical in the efficient solution of many large-scale problems in scientific computing because they reduce the time and storage needed to calculate a solution on a computer. Even when a scientific problem is modeled by differential equations and linear algebra, the languages of continuous mathematics, decomposing the problem into subproblems, mapping them on to processors in a parallel computer, scheduling the computations to satisfy precedence constraints, and organizing the data for the problem into efficient data structures that permit fast retrieval are all problems requiring discrete algorithms.

Impact on Applications

Many combinatorial problems arise in the computational infrastructure needed to solve scientific problems on a computer: in the design of computer architectures (e.g., interconnection networks in multiprocessors), in compilers (e.g., graph coloring for register allocation), and in operating systems (e.g., resource scheduling). They also arise in core computational strategies for parallel computing: problem decomposition and load balancing (graph and hypergraph partitioning), performance enhancement (ordering data accesses and computations to enhance spatial and temporal locality), external memory algorithms (scheduling disk reads to overlap computations and thereby mask the disk read latencies), and processor and resource allocation. Hence combinatorial algorithms play a pervasive role in all the application areas considered in this workshop.

Many applications considered in this workshop lead to irregular computational problems: in such problems, decomposition of the problem into a collection of computational subproblems leads to tasks whose time and space requirements vary widely. Examples of

irregular computations include unstructured mesh or sparse matrix computations. Concurrency is discovered by partitioning a task graph (whose vertices correspond to the tasks and edges to precedence constraints between the tasks) into subgraphs and mapping the subgraphs to the processors such that the computational work is balanced and the communication costs are reduced. Combinatorial, geometric, algebraic, and multilevel algorithms have been designed for this problem in the past decade, and many graph partitioning software packages have been produced as a result of this research. In many contexts, hypergraph models are superior to graph models in capturing the computational costs, and hence algorithms for hypergraph partitioning are being developed. Parallel graph and hypergraph algorithms for partitioning large-scale problems, when the task graph needs to be distributed among the processors, need to be developed in the future.

The vertices and edges of the task graphs can be ordered to enhance spatial and temporal locality on hierarchical memory machines. Orderings based on breadth-first search in the task graph, an eigenvector of the Laplacian matrix of the graph, and space-filling curves have been used to improve locality. Research into cache-oblivious algorithms has yielded new algorithms that deliver good performance independent of the cache parameters.

Graph coloring has been used to discover independent tasks in parallel computation, as well as to allocate registers in scheduling machine instructions by optimizing compilers. Edge-coloring algorithms can be used to schedule disk accesses in a multiprocessor when problems are too large within the combined core memories of the processors.

Job and task scheduling algorithms on multiprocessors enable jobs to be scheduled so as to increase throughput and reduce the waiting time for users. While much theoretical and practical work has been done on multiprocessor scheduling, scaling to tens or hundreds of thousands of processors for petaops machines will bring an entirely new set of algorithmic challenges.

Research Issues

Problems that require discrete mathematical techniques to model them abound in the applications areas considered here. The new biology is especially rich in problems where discrete objects need to be modeled. The challenge is to develop a combinatorial model for the given problem and then identify the combinatorial concepts and tools that can be used to solve the problem. Often, new combinatorial techniques need to be developed as well. Research in the past two decades has brought to light the many combinatorial subproblems that arise in solving sparse systems of linear and nonlinear equations. The problems of scale that arise in petaop computing will continue to bring new problems to light.

Computational problems in many application areas such as nanoscience and materials are currently solved by algorithms whose running times scale as $O(n^3)$ to $O(n^7)$, where n is a measure of the size of the input for the problem. For an algorithm with the time requirement proportional to n, a ten-million-fold increase in the computational power will only enable the solution of a problem whose input size is ten times bigger than before. Breakthroughs in such application areas will not come about primarily through improvements resulting from Moore's law; instead, the need here is for fundamentally

new algorithms. Discrete algorithms offer hope for such improvements, and these will be based on a deeper understanding of the structure of the problem. An example is the mutipole algorithm that reduced the cost of solving the n-body problem from n^2 to n. Nanoscience applications scientists report that currently computational speeds have improved modeling capabilities a hundredfold, while algorithmic improvements have improved them a hundred-million-fold.

In the past, researchers in algorithms have been satisfied with developing polynomial algorithms to solve computational problems. In the future, however, increasing attention will have to be paid to developing approximate algorithms that solve the problems in near-linear time in the input size of the problem because only such algorithms will scale on petaop computers.

The solution of large-scale sparse systems of linear equations or eigenvalue computations is a major computational task in every one of the twelve applications discussed at the Scales workshop. Combinatorial techniques play critical roles in the development of scalable solvers for these problems. Weighted matching algorithms move elements of large modulus to the diagonal and thereby enable convergence of Krylov solvers for numerically difficult problems. Other ordering algorithms that reduce the fill or move the nonzeros in the matrix closer to the diagonal also can be chosen to accelerate convergence. Incomplete factorization preconditioners and novel combinatorial preconditioners based on support theory are among the most robust techniques to accelerate convergence. Further research is needed to improve the efficacy while reducing the computational costs of these techniques. Approximation algorithms for maximum weighted independent sets provide efficient coarsening and refinement strategies for algebraic multigrid solvers. Sparse direct solvers continue to be useful when high accuracy eigencomputations need to rely on shift and invert techniques, and to solve coarse grid problems or subdomain problems in domain decomposition. Of course, domain decomposition relies on graph partitioning algorithms to obtain subdomains with desirable aspect ratios for faster convergence.

Combinatorial algorithms come to the fore in designing algorithms that exhibit high performance on the deep memory hierarchies on current teraop machines and on the deeper hierarchies expected on future petaop machines. Cache oblivious algorithms, developed in the last few years, hold the promise of delivering high performance for irregular problems while being insensitive to sizes of the multiple level caches. Meanwhile, structural analysis problems from industry currently demand the use of external memory algorithms since the memory on the multiprocessors is insufficient to hold the data, output, and the temporary storage needed. In this context, processors have to read a buffer of data from disk, exchange the data as needed, compute, and then write the output to disk. Scheduling the parallel disk reads and writes to avoid overflowing the memory on each processor has been modeled as weighted edge coloring problems. On petaop machines, fault tolerance calls for replication and makes for richer combinatorial models of these problems.

Next-generation petaop architectures are expected to have an order of magnitude more processors than current teraop processors. Scaling the number of processors by an order

of ten or more brings about entirely new algorithmic challenges: some algorithms will scale well, while others will not. New bottlenecks to performance will appear and will need to be addressed. Many of these issues cannot be anticipated before the new architectures are available for algorithm development and implementation. The new architectures might lead to the development of decentralized, self-organizing algorithms and new models of computation for large cellular architectures. These would represent a shift in the computational paradigm from the current single-program multiple-instruction style of programming. Formal specification and verification of algorithms would become increasingly important in the new regime.

Genomic and proteomic technologies are now capable of generating massive datasets (in the terabytes) in a single day's experimentation. Processing the raw data, organizing it, extracting knowledge from it, and storing relevant computed quantities, without having to store all of the instrument-generated data into a database, is a daunting challenge.

Massive problems abound in newly acquired sequence information of genomes and proteomes. Multiple alignment of the sequences of hundreds of bacterial genomes is a computational problem that can be attempted only with a new suite of efficient alignment algorithms on a petaop computer. Large-scale gene identification, annotation, and clustering expressed sequence tags (ESTs) are other large-scale computational problems in genomics. Currently 5 million human ESTs are available in databases that we cannot cluster effectively; meanwhile, this collection continues to grow. Clustering a set of 400,000 ESTs needs 2 hours on 60 processors of a Pentium PC cluster. These massive datasets will necessitate research into parallel and distributed data structures for organizing the data effectively.

Understanding the characteristics of protein interaction networks and protein-complex networks formed by all the proteins of an organism is another large computational problem. These networks are "small-world" networks, where the average distance between two vertices in the network is small relative to the number of vertices. Small-world networks also arise in electric power networks and semantic networks from homeland security and in models of the Web; understanding the nature of these networks, many with billions of vertices and trillions of edges, is critical to making them invulnerable to attacks. Enumeration of all the elementary cycles of a graph and circuits of a matroid help in characterizing gene regulatory networks.

Metrics of Success

The implementation of a new algorithm and its wide adoption by an applications community is the primary measure of success. The functionality of combinatorial algorithms is often delivered as open source customized software tools, for instance, in partitioning libraries and sparse solvers. When combinatorial algorithms and software enable modeling capabilities at scales larger by several orders of magnitude, applications scientists will have opportunities to optimize, predict, and control scientific phenomena.

Meshing Methods

Mesh generation is the discretization of the computational domain into a union of simple elements. It is often the first step in the numerical solution of partial differential equations. Meshes can take many forms ranging from structured collections of rectangular or curvilinear elements to unstructured collections of elements such as tetrahedra or hexahedra or combinations of the two (see Figure 1 for meshes generated for biology and accelerator applications). Mesh generation is a challenging and time-consuming endeavor and can often take weeks or months of an application scientist's time—time that would be better spent analyzing the results of scientific simulations.

Several factors complicate the mesh generation process. First, the boundaries of the mesh are often determined by physical objects with complex surfaces. These objects, called geometries, are often stored as solid models which can require considerable preprocessing to be made suitable for mesh generation. For example, creating watertight solid models and removing small, unimportant features can take considerable time and effort. Second, mesh quality can significantly impact solution accuracy and efficiency. Quality is influenced by element shape, orientation, and alignment with respect to the underlying physics problem to be solved, as well as interaction with the algorithm used in the equation solver. Ideally, if important physical features are known in advance, they can be accounted for in the initial mesh generation step, but this is often not the case. In these cases, adaptive mesh refinement (AMR) procedures, such as adding more elements to increase resolution and moving grid points to align them with solution features can be used to improve mesh quality as the solution proceeds. Also, in many physical phenomena, precisely tracking moving interfaces (e.g., between two different fluids or along a flame front) is critical in obtaining an accurate numerical solution. These interfaces can become complex and distorted, and several techniques, such as marker particle techniques, level set methods, and volume-of-fluid methods, exist to track them and couple their motion to the underlying physical phenomenon.

Impact on Applications

The effective use of these technologies can significantly affect the cost (in both human and computer resources) and accuracy of application solution procedures. In particular, mesh generation techniques for complex geometries and AMR procedures are often cited as critical in obtaining scientific results previously unattainable with less advanced methods. Many application domains have begun to take effective advantage of state-of-the-art meshing tools; however, a great deal of work remains to be done to extend the use of these tools in these domains, to insert existing technologies into new application domains, and to increase tool capabilities.

We highlight here several areas in which fruitful collaborations would lead to increased scientific capability.

Grid generation. Biological applications increasingly deal with complex geometrical data from imaging devices, such as magnetic resonance imaging or scanning electron microscopes. Focused research to develop robust algorithms and tools for grid generation from these models has the potential to significantly increase the role of computation in biological applications.

Multiscale meshes. Scientists concerned with accelerator design, plasma science, and astrophysics want to generate and use different meshes on the same computational domain, to specify vastly different mesh scales (e.g., smaller elements near geometry boundaries) or to generate meshes whose elements have a particular orientation (e.g., along magnetic field lines in tokomak simulations). These areas will benefit from development of new mesh generation algorithms, both structured and unstructured, that take into account multiple scales, multiple physics, and interesting physical features.

Mesh adaptivity. Adaptive meshing was widely cited as a pressing need in several application domains. In some cases, such as nanoscience, simulations are not currently adaptive, and the use of existing technologies would vastly decrease the computational cost of their simulations. The primary difficulty in these cases is redesigning and reimplementing the application software framework to be able to use adaptive mesh procedures. In other areas, such as astrophysics and plasma science, adaptive techniques are already used and new capabilities are being requested. In particular, efficient implicit AMR solvers, better boundary conditions for high-order adaptive methods, and techniques suitable for multiscale and multiphysics applications are needed for next-generation simulation codes.

Moving interfaces. A wide variety of physical phenomena—including problems in flame propagation, materials sciences, semiconductor manufacturing, crack propagation and fluid mixing—involve moving interfaces. In many of the most challenging situations, interfaces can become complex and distorted, existing over a variety of length scales. The ability to accurately track these interfaces and couple their motion to complex and sophisticated physics will have a major impact. As examples, in semiconductor manufacturing, thin layers of deposition substances exist in tandem with structures four orders of magnitude larger; careful modeling will provide new insights and recipes for building precise electronic components. Flow modeling in large geologic structures requires adequate tracking of fluid interfaces through materials of greatly varying structure and dimensions.

Interoperable meshes. Several new capabilities critical for the next generation of application simulation codes can be created only through interoperability of various meshing technologies and tools. For example, when using adaptive techniques on complex geometries, obtaining a highly accurate solution, particularly near the boundaries, requires a connection between the original geometric description of the computational domain and the mesh. Furthermore, hybrid solution techniques for the multiscale multiphysics problems currently of interest in many of the application areas often require more than one mesh and discretization type in the same computational domain. Creating these capabilities through interoperable meshing technologies will

enable highly accurate representations of the computational domain and greatly increased flexibility in the choice of solution technique.

Research Issues

A significant amount of research effort has been invested in exploring different mesh generation and adaptation algorithms. In some areas, such as tetrahedral mesh generation, the technology is reasonably mature and good-quality meshes can be produced for a wide variety of application areas with existing software. In other areas, such as hexahedral mesh generation and adaptation, the technology is less mature and active research is necessary before the technologies can be widely used. New research to improve application solution through advanced mesh generation and adaptivity on massively parallel computers is required in several areas.

Geometry Preprocessing: Robust tools are needed for transforming new kinds of data (such as MRI data) into a CAD model suitable for mesh generation. Increased automation of CAD model preparation for mesh generation is desirable, for example, automatic tools for small feature detection and removal, cleanup to create watertight geometries, and decomposition tools to create more easily meshable domains. Also needed are general techniques for incorporating geometry information into later steps of the simulation process.

Mesh Generation: Researchers need to reduce the time to generate a high-quality mesh from months to days or hours by increasing automation. The coupling between mesh generation techniques and discretization and solver technologies must be strengthened in order to increase efficiency and control approximation error. Also needed are algorithms appropriate for multiscale, multiphysics applications.

Mesh Quality Control: Systematic studies of error estimation must be carried out for different mesh types, physical systems, and solver algorithms. Solution accuracy must be improved through mesh quality improvement techniques driven by error estimators. Work is also needed on specialized algorithms suitable for specific application.

Adaptive Refinement: Research is needed on efficient implicit AMR solvers, better boundary conditions for high-order adaptive methods, and grid point movement schemes to align mesh points with solution features. The link to solver technologies should be strengthened and general methods developed that utilize hierarchical data structures. Also needed are new solution technologies that address spurious numerical effects at mesh interface boundaries. A critical need is development of efficient distributed-memory algorithms and dynamic load-balancing methods that scale to tens of thousands of processors.

Interface Tracking: Hybrid methods can capitalize on the strengths of existing methods. Techniques are needed to automatically align interfaces with small-scale structures in order to provide increased accuracy. Moreover, interface motion must be coupled to higher-order operators in accurate PDE schemes, as well as to subgrid physics through better mathematical modeling. A major need is an increase in the range of scales modeled by the interface through adaptive interface representation. In addition, advanced methods

must be devised to handle cut cells, implicit schemes, conforming/moving meshes, and fixed grid formulations of moving interfaces.

Interoperability: Common interfaces must be developed for mesh and geometry access, and compliant components must be made available to the research community. One approach is to use hybrid solution strategies that combine different mathematical models, discretization techniques, and solution algorithms. Researchers should also explore efficient and scalable methods of exchanging data between different mesh types for multiscale and multiphysics applications.

Metrics of Success

Success in this area can be measured in many ways, including the following:

- Generation of new scientific results through increased computational efficiency and accuracy delivered by advanced meshing techniques
- Improved reliability and robustness of solution methods obtained on higher-quality meshes, or better approximations to the computational domain
- Significant reductions in the total time to solution, which includes human costs as well as the computational cost of the simulation code
- Increased capabilities delivered to application domains from new research in advanced meshing methods and interoperable solution strategies
- Significant reduction of the amount of time and energy application scientists expend thinking about their meshes

Multiphysics Solution Techniques

Multiphysics solution techniques comprise time integration, nonlinear, and linear solver techniques that can be abstracted and generally applied to accurately, robustly, and efficiently solve-strongly coupled, highly nonlinear, multiphysics systems. These systems are characterized by myriad complex interacting physical mechanisms. The mechanisms can balance to produce steady-state behavior, they can nearly balance to evolve a solution on a dynamical time scale that is long relative to the component time scales, or they can be dominated by one or a few processes that drive a short dynamical timescale consistent with this dominating mode.

This is a relatively new focus area within computational science, and its impact will be broad and profound. The opportunity to begin to explore and develop these new methods now is due to a convergence of recent developments: new parallel supercomputing technology with significant single CPU memory resources and aggregate performance in the hundreds to thousands of teraops range, along with recent algorithmic advances in time integration, nonlinear solver, and linear solver technology. This new research effort is envisioned as a close collaboration between numerical mathematicians, computational scientists, and applications scientists. The components of such an effort are as follows:

- Development of a numerical analysis methodology to provide detailed analysis of inter-algorithmic numerical error and convergence in the current state-of-the-practice multiphysics solvers
- Development of application specific prototype-type systems to enable testing of modern nonlinear implicit algorithms and advanced splitting methods that will improve the accuracy, robustness, and efficiency of the time integration of multi-physics systems
- Development of a set of software tools for producing robust, accurate, and efficient solution methods.

Each of these is critically important to advance current simulation capability and enable predictive simulation of complex multiphysics applications.

To put this new research area in context, we consider an example of what is currently done for multiphysics time integration within most application areas (DOE and elsewhere). In practice, time integration is typically carried out by using a low-order expansion time step linearization or operator-splitting technique. These techniques linearize or split strongly coupled physics and do not fully converge the nonlinearities of the system in an attempt to reduce the cost and complexity of the transient solution at each time step. Inherent in a number of these techniques is a failure to accurately maintain the approximate balance of the competing time scales when integrating at the dynamical time scales of interest. Essentially these approaches evolved during an era when computation capability was much more limited in memory and computational performance. The desired advantage of these techniques is a reduction in the complexity of the component solves. For a given time-step size, all these methods clearly exchange

temporal accuracy and/or possibly numerical stability in the pursuit of computational efficiency. Additionally, a common result of this trade-off is a poorly understood time step limitation, which makes long time-scale integration computationally infeasible.

Impact on Applications

Problems dominated by a few processes with dynamic time scales comparable to that of the whole problem are amenable to explicit or operator-split time integration methods that resolve this driving mode. However, solution methods for problems where multiple processes balance in a way that leads to equilibrium or slowly varying behavior require the use of significant coupling in the nonlinear and linear solution techniques and a significant implicit character to the time integration methods. Examples of these first two cases are common in many DOE Office of Science application areas:

- Steady-state or long time-scale transient operation of fusion reactors, within the extended magnetohydrodynamic model, which balances plasma flow, wave propagation, and resistive dissipation (including both ion and electron physics)
- Simulation of operation of a solid oxide fuel cell which balances, fluid flow, heat and mass transfer, combustion chemistry, and electrostatics
- Simulation of subsurface contaminant transport in which there are balances between advection, diffusion, and reaction
- Simulation of protein-folding processes that span time scales of ten orders of magnitude, with significant coupling between fast vibrational frequencies of bond stretching contributing to the global folding process that occurs on scales of one second

Many multiphysics simulations involve the use of hybrid methods. A typical example arises when one wants to perform a simulation where some portion of the geometry could be represented by the Navier-Stokes equation and the remainder of the geometry required a Boltzmann equation representation. Additionally, these distinct subregions may need to be followed adaptively. Another example of a hybrid algorithm is a situation where over the entire geometry one wishes to use a deterministic model for some of the fields of interest, while the remaining fields are simulated statistically (it is assumed here that all fields are tightly coupled). The application of these hybrid methods could be broad and the impact of hybrid methods high.

We briefly discuss a few direct examples of impacts made in early efforts in the area of multiphysics solution techniques. These impacts have been in the areas of numerical analysis of errors, algorithm development and testing on prototype systems, and “first-of-a-kind” simulations enabled by modern multi-physics solution techniques. The examples indicate the types of impact one could expect from a healthy research effort in this enabling technology.

In plasma science, multiphysics solution methods have been used successfully in magnetohydrodynamics (MHD) simulations, which find application in magnetic

confinement fusion and space-weather modeling. The MHD problem takes on an increasing multiphysics nature when one goes to the Hall, or extended, MHD model. This model incorporates both electron and ion physics and is the global model that best represents many application problems of interest. A multiphysics time integration method has been developed that uses Newton-Krylov methods with physics-based preconditioning using multigrid methods. The simulations went deep into the nonlinear phase. Typical simulations required 10^5 time steps with the modern integrator. The modern integrator provided a factor of 30 speedup in simulation time as compared to an explicit method for the same level of accuracy. This speedup was possible because the modern multiphysics integration method could employ time-steps sizes that were a factor of 1000 over the explicit limit, while remaining highly accurate.

In analysis, modified equation analysis has been applied to linearized time steps and split time steps on simple model problems. This analysis has provided insight into the source and propagation of numerical error resulting from linearized time steps and split time steps. Moreover, results on model problems has enabled useful insight into the application of split and linearized time-step methods to more complicated systems. In related work, numerical experiments on coupled and split methods have been applied to reaction-diffusion problems, and the time steps at which each method reached its asymptotic convergence state was assessed. Extremely interesting (and concerning) behavior was uncovered on a higher-order accurate splitting method. In one example the higher-order splitting method performed well early in the simulation but developed significant (zeroth-order) errors later in the simulation. This type of error can be devastating to the goal of predictive simulation.

Clearly, the potential impact of multiphysics solution techniques is broad and profound. The type of impact a particular application observes will vary. The impact could be as grand as a truly first-of-a-kind simulation or simply a much better understanding of the errors present in the existing multi-physics solution method. Multiphysics solution techniques have demonstrated that they are indeed “enabling technologies” in a number of different application settings. The sidebar stories in this section depict the potential impacts that can be realized.

Research Issues

This new effort is envisioned as an algorithmic development, analysis, and evaluation effort with a companion effort of multiphysics-related software tools development. The research effort requires a close collaboration between mathematics, computation, and scientific applications researchers. For some of the software component solver requirements there exists traditional solver software for time integration, nonlinear, and linear solution methods. The main aspects of this research are as follows:

- Development of a numerical analysis methodology to provide detailed analysis of inter-algorithmic numerical error and convergence of multi-physics solvers
- Development of prototype reduced system models for the various scientific areas that approximately span the application space and can be used in algorithm exploration and demonstration

- Analysis of prototype scientific applications to provide hard quantitative comparisons of error and convergence characteristics of decoupled/coupled solvers and linearized, split, semi-implicit, and fully-implicit methods
- Development of robust and efficient advanced (higher order) strong coupling approaches (nonlinear / linear) for multi-physics applications, including the development of advanced robust and efficient (higher-order) splitting methods
- Development of robust and efficient solution methods for non-PDE-based scientific applications, including parameter continuation, bifurcation methods, and efficient long-time integration techniques
- A companion software development effort at developing physics-based preconditioners (approximate block factorizations), and other nontraditional preconditioning approaches (recursive projection method for fixed point solvers), for specific scientific applications. This effort will build on current solver efforts such as the SciDAC TOPS effort, PETSc, and the Trilinos Solver Framework effort.

Metrics for Success

The following are specific metrics for success that can be used for this new area:

- Quantification of multiphysics solver issues for prototypes of interest to scientific applications, via analysis and numerical experiments
- Relaxed component time scale restrictions, resulting in more robust, accurate, and efficient solutions for scientific applications
- Initiation of a robust algorithmic effort for long-time scale time integration for non-PDE based systems and hybrid methods
- Enabling of long-time scale time integration of complex non-PDE based scientific simulations
- Application of extensible, useful tools to create accurate, robust, and efficient multiphysics solution methods that can be prototyped and employed in a number of scientific application areas

The main metric, however, is the success of the multiphysics solution methods effort in enabling first-of-a-kind simulations that result in discovery and a deeper understanding in an application area through large-scale simulation.

Multiscale Simulation

Physical processes at exceedingly small scales of time and space are becoming increasingly well understood. Technologies for engineering systems at the micro and nano scales are rapidly emerging. Yet in general we have no means of translating fundamental, detailed scientific knowledge at small scales into its effects on the macroscopic world in which we live. Without the capability to "bridge the scales", important scientific and engineering problems will remain out of reach for the foreseeable future.

Mathematical modeling and computational simulation have reached the point, following 30 years of exponential advances in modeling, algorithms and computer hardware, where simulation of most physical processes over relatively narrow ranges of scales has become an essential tool for both scientific discovery and engineering design. But currently we do not have a mathematical framework and software infrastructure for the integration of heterogeneous models and data over the wide range of scales present in most physical problems. Fundamentally new mathematics and considerable development of computational methods and software will be required to address the challenges of multiscale simulation. Three major approaches to the multiscale problem have been identified: *multiresolution discretization methods,* which resolve multiple scales within a single model system by dynamically adjusting the resolution as a function of space, time, and data; *hybrid methods,* which couple different models and numerical representations that represent different scales; and *closure methods,* which provide analytical representations for the effect of smaller, unresolved scales on larger scales in a numerical simulation that might resolve only the larger scales.

Impact on Applications

Multiresolution numerical methods include adaptive timestep methods for stiff ordinary differential equations, differential-algebraic systems, and stochastic differential equations; adaptive mesh refinement (AMR) and front-tracking methods for partial differential equations; and adaptive analysis-based methods for integral equations. The following are examples of science opportunities involving the development of new multiresolution discretization methods.

RF modeling in fusion problems. Radio-frequency analysis codes calculate the details of heating the plasma as it is exposed to a strong electromagnetic field. Calculations indicate that the regions of high gradients are strongly localized in space; furthermore, the underlying solvers for the integral equations are based on dense matrix representations that do not take advantage of locality. The application of adaptive analysis-based methods to this problem could speed its solution by 2 to 3 orders of magnitude and thus decrease the time spent in this phase of the design cycle for fusion reactors by months or years.

Supernova simulations. The extension of the combination of AMR, front-tracking, and low-Mach number models to the case of nuclear burning in supernovas would enable computation of the large-scale, long-time dynamics of those processes which lead to the explosion of a type 1A supernova and are believed to determine its later evolution. For both type 1A and core-collapse supernovae, transport of photons and other particles is an essential component of the physics: it is the mechanism by which we observe and interact strongly with the multiscale structure of the system. However, the development of AMR for radiation is far less mature than it is for fluid dynamics, and new ideas will be required.

Stochastic dynamics of biochemical reactions. In microscopic systems formed by living cells, small numbers of reactant molecules can result in dynamical behavior that is discrete and stochastic rather than continuous and deterministic. Accurate description of such behavior is impossible to obtain through deterministic continuous modeling (e.g. ODEs). Stochastic simulation has been widely used to treat these problems; however, because this procedure simulates every reaction event, it is prohibitively inefficient for most realistic problems. Recently, a coarser-grained approximate method called tau-leaping has been proposed for accelerated discrete stochastic simulation. A theoretical and computational framework for such accelerated methods is needed, along with reliable and efficient means to partition the system so that each reaction and species is modeled at the appropriate level.

Analysis-based methods in quantum chemistry. A standard approach to the calculation of ground states and transitions to excited states is to begin with a Hartree-Fock wave function (a tensor product of single-particle wave functions) and to compute coupled-cluster corrections that represent the effect of inter-particle interactions. Current methods for computing these corrections scale like N^6, where N is the number of particles. The introduction of real-space hierarchical representations of these corrections that represent the smooth nonlocal coupling by an appropriately small number of computational degrees of freedom could lead to a computational method that scales like N. This would enormously increase the range of problems that could be computed to chemical accuracy with such approaches.

The starting point for the development of *hybrid methods* is typically an analysis of a general mathematical model that describes the system at all relevant scales. This analysis yields either a hierarchy of models each describing the behavior on a particular spatial scale with some overlap in the range of validity, or a splitting of the unknowns into components corresponding to slow and fast dynamics. Examples include the derivation of fluid equations as phase space averages of a more fundamental kinetic description and deterministic chemical reaction models as averages over many discrete collisions. In hybrid numerical representations, the averaged dynamics, which are less costly to represent numerically, are used in regions where the deviations from those dynamics are small, while the more complete description is used in regions where the deviations from the averaging hypothesis-large mean-free paths in the kinetic/fluids example or sufficiently low concentrations of reactants in the chemistry case-are sufficiently large to have a substantial effect on the macroscopic dynamics. The assumption is that there is

sufficient advantage to hybridizing two or more models, as opposed to using a single model that is more generally applicable.

The following are examples of science opportunities involving the development of new hybrid methods.

Reaction and diffusion processes in cells. A multiscale computational framework for the numerical simulation of chemically reacting systems, where each reaction will be treated at the appropriate scale, is clearly needed for the simulation of biochemical systems such as cell regulatory systems. The framework can be based on a sequence of approximations ranging from stochastic simulation at the smallest scales to the familiar reaction rate equations (ODEs) at the coarsest scales. The coupling is not straightforward, however, and must be done dynamically. Many technical issues are involved in ensuring that the system is properly partitioned, that the models chosen at each scale are sufficient to approximate those processes, and that the hybrid method itself is stable and accurate. Further complicating this problem is the need to incorporate spatial dependence, leading to the coupling of stochastic simulation with PDE models that will eventually need to distinguish and model the highly heterogeneous structures within a cell.

Macroscopic stability in tokamaks. The appropriate description of the macroscopic dynamics of a burning plasma is a spatially heterogeneous combination of fluid and kinetic models, which operate on time scales ranging from nanoseconds to minutes. To perform predictive simulations for such problems will require the development of a variety of hybrid simulation capabilities. Examples include state-space hybridization of a kinetic description of weakly collisional energetic particles produced by fusion with a fluid description of other species and spatial hybridization of two-fluid and kinetic treatments of localized plasma instabilities with large-scale fluid models.

Catalytic surface reactions and the synthesis and oxidation of particulates. Chemical reactions at a gas /solid interface are not well modeled by continuum equations. Such reactions must be modeled at the atomistic level; and integrating their treatment into a continuum simulation will require hybrid discretizations that couple atomistic and continuum scales. The central issue in developing these hybrids is determining statistical distributions from the atomistic scale that are needed to express the coupling between atomistic and continuum scales.

Hybrid models for climate modeling. The large-scale motions of the Earth's atmosphere and oceans are well described by the hydrostatic approximation, in which the vertical momentum equation is replaced by a hydrostatic balance law. However, as it becomes necessary to resolve ever-larger ranges of scales in climate models, the use of the hydrostatic approximation at the smallest scales becomes physically invalid. Hybrid models must be developed in which the hydrostatic approximation is used for large scales, while a non-hydrostatic model is used to simulate localized small-scale behavior. Some of the components of such an approach include asymptotic analysis of the various fluid-dynamical processes (e.g., compressive/thermodynamic, gravity-wave, vortical) operating at different scales; a systematic understanding of the well-posedness of initial-boundary value problems for hydrostatic and nonhydrostatic models; and carefully

designed numerical methods to implement well-posed couplings between such pairs of models

An open question for a large class of problems is the derivation of macroscopic models from more-detailed microscopic models, often referred to as *closures*. Such problems include those that lack a strong separation of scales, rare-event problems, and problems involving the reduction of high-dimensional state spaces to a small number of degrees of freedom.

The following are examples of science opportunities involving the development of new closure methods. **Turbulent mixing of a multifluid medium.** In turbulent mixing problems that arise in high energy-density physics, it is necessary to simulate the transition between various regimes: distorted sharp interfaces, macroscopic breakup ("chunk mixing"), and atomic-scale mixing. In addition, it is necessary to develop models of the interaction of the turbulent medium with other physical processes such as radiation transport and nuclear burning.

Problems with a small number of degrees of freedom. In materials science and biology, only a few features of a large molecule may determine its function. This situation can occur, for example, when an associated potential has a few well-delineated minima, with much of the motion confined to the neighborhoods of these minima and an occasional jump from one neighborhood to another. Also in this class of problems are rare events, such as ion exchange and nucleation of defects. In all these cases, the number of significant aggregate degrees of freedom is orders of magnitude smaller than that of the microscopic description, as are the time scales over which the aggregate degrees of freedom change relative to those of the microscopic description.

Interaction of turbulence and chemistry in combustion. Current numerical methodology is able to model the interplay between turbulence and chemistry in laboratory-scale systems, but the resolution requirements for such simulations make them intractable for more realistic flames. Existing methodologies for representing reaction kinetics in a turbulent flame explicitly separate turbulence scales from chemical scales. In most situations, however, turbulent transport and reaction scales are strongly coupled; and new approaches are required to represent the reaction process that respect this coupling.

Large-scale subsurface flows. Simulations in environmental remediation rely on ad hoc closure schemes that have little relation to the microphysics. New closure methods will enable the development of models that systematically represent the effect on the macroscale dynamics of pore-scale physics such as wettability, dynamic relations among fluid saturations, and the behavior of disconnected phases. Simulation of subsurface flow is complicated, however, by the fact that the subsurface medium itself is heterogeneous, with fluctuations on scales two orders of magnitude smaller than those of large-scale flow scales, but whose presence nevertheless has a substantial impact on transport at those scales. In addition, these fluctuations can be characterized only statistically. Such problems need to be attacked with new closure methods combined with new hybrid stochastic and deterministic models to represent those closures.

Hierarchical models. In materials science, there exists an elaborate hierarchy of models for various length scales: macroscale continuum mechanics, molecular-scale models based on classical mechanics, and various techniques for representing quantum-mechanical effects. While some methods are emerging for coupling these scales, they are not yet on a firm theoretical foundation, and they are not applicable to all systems of interest. Work must be done to establish these foundations and to extend the range of applicability of these methods.

Research Issues

Such methods are the subject of ongoing development in the mathematics community and have been used successfully on specific multiscale problems. Furthermore, these methods form, in large measure, the fundamental components from which more elaborate multiscale simulations will be built. Open questions in this area include the extension of AMR and front-tracking methods to a broader range of multiscale problems involving complex combinations of multiple physical processes; the development of new multiresolution-in-time algorithms for stochastic differential equations, particularly for the case of pure jump processes; and fast adaptive analysis-based algorithms for integral equations for problems with spatially varying coefficients. With such advances, we will be able to provide new capabilities for solving a variety of timely and important science problems for DOE.

Several areas of mathematical research arise in the development of hybrid methods. First, we are often confronted with models that were not designed for coupling with models on other scales or were not intended for use in simulations at all. In these cases, we will need to develop mathematically well-posed versions of the individual models and of the coupling between models on different scales. Second, we will need to develop stable and accurate discretizations for the individual models, and for the coupling between scales and models. This activity will almost certainly require the development of new numerical methods. For example, for many models, such as the kinetic models cited above, the finer-scale behavior is often represented by sampling a stochastic process (i.e., a Monte Carlo method). The coupling of such a method to a high-order accurate difference approximation can lead to a catastrophic loss of stability or accuracy, even though the component methods by themselves are well behaved. We also expect that hybrid methods will use the multiresolution methods described above as a source of components, and the need to hybridize them will lead to the development of new classes of such methods. We must build our understanding of where and how small-scale fluctuations affect large-scale dynamics and of how ensembles of simulations might best be used to quantify uncertainty in chaotic or stochastic systems. A variety of new analytical techniques have been suggested that, combined with large-scale simulation, provide new tools for attacking the problem of deriving closures. One such technique is the use of concepts from nonequilibrium statistical mechanics for representing the dynamics of a coarse-grained system in terms of the unresolved degrees of freedom. This technique is less susceptible to realizability problems than are traditional moment closure methods. Markov-chain Monte Carlo methods are also promising for finding near-invariant sets and the transition probabilities between them and projective integration methods for self-

consistently determining macroscopic degrees of freedom and their effective dynamics. These methods provide a starting point for designing new closure models, particularly when validated against direct simulation with all of the degrees of freedom resolved. The resulting multiscale models can then be incorporated into large-scale simulations to, for example, further reduce the size of regions using the costly detailed model in a hybrid method of the type described above.

Metrics for Success

In the near-term we will see the application of existing techniques to new multiscale problems, and the development of new algorithms for stochastic models.

Development of existing multiresolution techniques for new applications. In some areas, mature models are already capable of correctly representing the important multiscale behavior. Extensions of existing multiresolution numerical methods will be developed to accurately, efficiently, and stably simulate that behavior.

Development of multiresolution and hybrid stochastic numerical methods. Stochastic models are central to many multiscale science problems. Early development of these new fundamental algorithms will provide a foundation for the later development of the entire program.

Mathematical and numerical analysis of the coupling between scales. A number of areas, such as the coupling between quantum and classical molecular dynamics in materials, plasma turbulence and transport scales in fusion, and hydrostatic and non-hydrostatic scales in atmospheric fluid dynamics, are susceptible to analysis using current state of the art analytical tools, particularly when combined with a robust numerical experimentation capability. Success in analyzing such applications will provide the basis for new multiscale algorithm development, as well as providing new extensions to the mathematical tools.

In the medium-term we will develop entirely new techniques both in analysis and simulation for multiscale problems, as well as the availability of major components of software infrastructure.

Prototype simulations using multiresolution and hybrid methods. This would involve application of the methods described above to scientifically important problems arising in multiscale science.

New methods for analyzing multiscale behavior. New methodologies for closure should be developed and used to derive multiscale models for some of the "difficult" cases in multiscale science, e.g. problems without strong scale separation, rare event problems, or reduction of high-dimensional state spaces to a small number of degrees of freedom. Also needed are algorithms and software for multiscale sensitivity and uncertainty analysis and software for core components of multiscale algorithms.

In the long-term we expect to see the impact of the program on DOE science applications by means of a new generation of multiscale simulation techniques.

Comprehensive scientific simulations using new multiscale techniques. This would involve the integration of the multiscale methods described above into comprehensive multiphysics simulations.

Application of new closure models. The new models will be used in high-performance simulations to solve some of the outstanding hard problems in multiscale science, e.g., fluid turbulence, protein folding.

A new generation of multiscale software. Robust and adaptive mathematical software will be developed, implementing a new generation of multiscale algorithms for simulation and analysis.

Solvers and Fast Algorithms

Like the engine hidden beneath the hood of a car, the solver is an unglamorous but critical component of a scientific code, upon which the function of the whole critically depends. As an engine needs to be properly matched to avoid overheating and failure when the vehicle's performance requirements are pushed, so a solver appropriate to the simulation at hand is required as the computational burden gets heavier with new physics or as the distance across the data structures increases with enhanced resolution. Solvers may otherwise fail in various modes, such as expansion of fill-in beyond available memory or a catastrophically small pivot for a direct solver (e.g., a form of Gaussian elimination), or a convergence plateau or a putative solution containing a large component of low frequency or nullspace error for an iterative solver (e.g., a form of relaxation). Even if a solver usually avoids failure, it is of little use if it bogs down in convergence rate or scalability as a simulation expands to fill available machine capacity. Classical methods that make up the majority of solvers in scientific and engineering codes have floating point operation complexity requirements that grow superlinearly in the storage size of the discrete system. The storage size is typically the product of number of fields represented times the number of elements in the mesh, though it may also be measured in the number of particles or basis functions. Solvers whose work grows superlinearly in a code whose other computational phases (as is typical) grow linearly in the storage size eventually consume 90%, then 99%, 99.9% etc., of the execution time as the discrete size is expanded, eventually bottlenecking scalability at a point that depends upon the specific complexity term coefficients and the budget and patience of the customer. Furthermore, the solver may have the most stringent communication requirements of any phase of the code, imposing another limit of Amdahl type on scalability.

The term "solver" is most commonly associated with a method that, given a vector b, and some means of applying the action of operator A, returns x in $Ax=b$. The Department of Energy and its antecedent organizations have been investing in algorithms for generic and special cases of this problem, which occupied such greats as Householder, Von Neumann, and Wilkinson, for over fifty years. A linear algebraic solver is a common inner kernel in scientific and engineering codes, and therefore a useful archetype. However, a linear algebraic solver is but the *base* of a solver toolchain, on top of which can be built many other solver classes. Nonlinear algebraic methods ("rootfinders") typically linearize a nonlinear operator at a succession of points that eventually approach the root and solve (approximately) a system of the form $Ax=b$ at each linearization, where A is an approximation to the Jacobian matrix. Often these systems are dominated by a Poisson operator, whose conditioning, which is related to the modes with the largest and smallest eigenvalues, degrades as the number of representable modes approaches the infinite number of the continuum problem, through mesh resolution. Similarly, eigenanalysis techniques, such as "shift-and-invert," require many solutions with a matrix pencil whose condition number becomes arbitrarily poor as the eigenvalue is approached. A Krylov method, such as the method of conjugate gradients or restarted GMRES, preconditioned by some simpler iterative scheme, such as an incomplete factorization, is

a common linear solver combination for large-scale linear systems, since it can work within a constant multiple of the memory requirements for representing the discrete system. However, this combination, commonly found in DOE's simulation codes, does not scale indefinitely.

Besides solvers for algebraic problems, there are stiff integrators for method-of-lines formulations of multi-rate partial differential equations (PDE). Stiff integrators require that a nonlinear algebraic problem be solved across the domain on each timestep, which, in turn leads to a succession of linear problems. Finally, optimization methods for PDE-constrained problems in parameter identification, control, and design, lead to inner loops of large nonlinear problems whose Jacobians are saddle-point, or Karush-Kuhn-Tucker matrices.

Under the rubric of solvers, we also include the so-called "fast solvers" of multigrid and fast transform type. These methods are somewhat more fragile than the general-purpose methods above, requiring certain problem structure to work effectively or even to work at all. Their importance stems from their optimal complexity – their asymptotic operation count grows linearly in the problem size N, or only slightly faster than linearly (such as $O(N \log N)$). Hence, these methods do not suffer from the poor scaling problems of most other solvers in the limit of highly resolved PDE-based simulations.

Other important solvers exist that are based on formulations of a physical problem other than a PDE discretized on a mesh. For instance N-body problems of gravitational and electrostatic potential type can be solved with hierarchical algorithms in the fast multipole family, which have asymptotic complexity as low as $O(N)$. From their invention and development in the last two decades, these methods have spread slowly to date, due to the lack of freely available software and the large coefficient of the linear complexity term. However, as problem size increases with the availability of terascale computers, fast solvers of all types are receiving increased and deserved attention.

Impact on Applications

Solvers lie at the core of a computer simulation of any equilibrium problem and of any multirate problem in which the timescales of the fastest phenomena are much smaller than the timescale of integration required to march out the physical results of interest, yet the fast phenomena impose an unacceptably small stability restriction on the timestep of explicit integration. In such a case, in which the accuracy restriction on the timestep is much looser than the stability restriction, an implicit method is called for to put the fast phenomena in quasi-equilibrium. Multirate problems such as these arise across the applications spectrum of computational science. In global climate simulations spanning centuries, acoustic and gravity waves need to be filtered. Magnetic fusion energy simulations spanning seconds need to step over Alfvenic and magnetosonic waves. Protein folding simulations spanning milliseconds cannot afford to resolve quantum fluctuations of the bonding orbitals. Depending upon context, combustion simulations may benefit from partitioning reactions into fast and slow and handling the fast reactions implicitly. Determination of the phenomena that can safely be stepped over with an implicit solve is problem-specific and requires expert judgment taking into account the

energy contained in different scales and energy transfers between the scales. This determination is not something that software can be expected to do automatically. However, application domain experts have decades of experience in formulating their systems to analytically filter out unwanted phenomena and they usually require the solution of an implicit subproblem in the implementation. These implicit solves are frequently the most fundamental barrier to large-scale parallelization of the resulting simulation. An infrastructural investment in solvers therefore has impact across the entire range of computational science.

Nearly every direction in which a simulation code is extended has the potential to be obstructed by the solver. Increasing spatial resolution tends to increase the condition number of Poisson and more general equilibrium problems discretized on a mesh. Increasing the duration of transient simulations of a fixed-size problem soon exhausts the potential of spatial domain decomposition as a parallelization paradigm and requires solvers that can parallelize over the causal, temporal dimension. Introducing more physics adds both more unknowns and more modes of coupling between unknowns that may shorten the length of a "safe" update step in nonlinear rootfinders. Higher order spatial discretizations lead to coefficient structures that lack the convenient "M-matrix" positivity property of lower order schemes, which challenges the heuristics behind coarsening algorithms in multilevel solvers.

Besides these challenges to solvers that arise within a single simulation as it is scaled up, there is the fundamental issue of how multiple simulations are used to answer a single scientific or engineering question. A scientist employing simulation to discover mechanisms and an engineer employing simulation in design need sensitivity information along with the result of a simulation. It is not sufficient to map a single input to a single output, such as a temperature boundary condition to a heat flux across a surface. A much more interesting result is a partial derivative of the output with respect to the input. In design, control, or parameter identification applications, such sensitivity information can be employed to improve, in some sense, the input data, and drive the system toward some objective. In scientific discovery, sensitivity estimates can be used to bound the effects of uncertainty and to refine the search for controlling phenomena. Sensitivity and optimization require solving large nonlinear and linear algebraic systems of which the Jacobian of the original PDE (or its adjoint) is just a component block. Unfortunately, many contemporary simulation codes do not employ sophisticated implicit solvers and may not even form or estimate the Jacobian matrix. Nevertheless, a sophisticated implicit solver infrastructure for a simulation is a key stepping stone for the advanced uses of the simulation that get at the underlying scientific or engineering questions.

There are also questions of a statistical nature that simulation is called upon to answer, leading to a need for ensembles of simulations. Contemporary ensemble computations typically launch independent instances of the simulation. In practice, this often means that tens to thousands of instances of simulations, each of which may spend 90% or more of its cycles in the solver, are setting up and solving closely related linear and nonlinear systems, without exploiting any opportunities for reuse of solver information between the simulations. Opportunities include reuse of hierarchical representations, graph orderings,

approximate factorizations, Krylov subspaces, initial iterates, and dynamically selected algorithmic tuning parameters.

Research Issues

The most obvious and seemingly pressing issues in solver research flow directly from the enumeration of barriers in the previous section. The optimality of fast solvers must be extended to less ideal systems. Opportunities for parallelism must be explored beyond spatial domain decomposition. However, the most fundamental progress may occur by backing up several steps before the solver library is called. The mathematical and structural properties of the matrices and nonlinear residual functions fed to a solver are inherited from decisions made upstream in the formulation of the continuous problem and in its discretization. Analytical decisions to "filter out" various phenomena give rise to implicit systems or subsystems. These include for instance, the choice between primitive variables and decompositions of those primitive fields into potentials and streamfunctions, leading to the need to invert Poisson-type problems in an inner loop. Even more fundamental is the decision to formulate a problem as a system of PDE or as some equivalent combination of PDE and integral equations. Given the poor asymptotic conditioning of the discrete representation of differential operators, whether as finite element stiffness matrices or spectral element differentiation matrices, the highly spatially resolved simulations of the future will require either higher and higher precision arithmetic or reformulation, such as via integral equations. A second-kind integral equation has ideal spectral properties for Krylov iterative methods – a compact perturbation of an identity operator. Integral equation kernels of increasing generality are expressible in multiresolution bases, which means that their action can be applied at high (guaranteed) accuracy at relatively low cost. Such representations are ideal for Krylov methods on parallel computers, in which most of the work is in the form of matrix-vector multiplications.

Another research interaction is between solvers and meshing methods. The premium on mesh generation is well understood with respect to accuracy of representation of a discrete solution. There is also a premium with respect to conditioning of the discrete operator. For instance, anisotropy is a bugbear of the simplest multilevel methods based on pointwise relaxation. It must be dealt with either by sophisticated coarsening strategies or relaxation methods with larger implicit aggregates, or both. As Joe Thompson, a pioneer of generalized gridding in fluid dynamical applications, is wont to say, "when the discretization is right, the solution is easy." While oversimplified for rhetorical purposes, there are several mathematical manifestations of this folk wisdom. There is a strong incentive for PDE infrastructure that combines meshing and solution strategies, rather than limiting their interaction to a narrow interface defined by a stiffness matrix that is constructed on one side and solved on the other, without any of the geometric or physical information on the formation of the matrix available to the solver.

A promising theme in solvers for large-scale applications is "physics-based preconditioning," in which an approximate solution procedure from an existing code (or freshly derived from operator splitting or asymptotics) is called by a Krylov-iterative solver as a preconditioner. Nearly every simulation code defines a map between a

nonlinear residual and a correction to the solution within each major iteration for an equilibrium problem or timestep for an evolution problem. This map effectively approximates the application of the inverse of the Jacobian matrix to the residual, which has the same role, namely production of a correction to the current solution, in Newton's method. Since the product of the Jacobian with an arbitrary vector can be approximated by a finite difference of two residual vectors, an effective Jacobian-free Newton-Krylov iteration can be composed from any residual-to-correction map and finite differences of residuals. A more traditional Newton-Krylov approach requires computation of a Jacobian matrix of the full system, which can be used to precondition the system through incomplete factorization or some other purely algebraic matrix splitting. Physics-based preconditioning derived from operator splitting has the potential to express the preconditioning through a series of simple, scalar operators rather than one global system. The simpler scalar operators can be derived from lower order discretizations than the discretization in which the solution is sought, often leading to a series of M-matrix operators, each of which has an optimal inversion procedure via multigrid.

Since optimal solvers are now known for some important operators, a natural question arises on what to do after O(N)? Because O(N) is the information-theoretic lower bound for a solver (it takes this much work just to print the answer), it might appear that research on solvers is nearly complete. What remains, of course, is to extend optimal complexity to many other operators with properties like anisotropy, inhomogeneity, indefiniteness, strong skew-symmetry, and other features that still defy optimality (or carry huge constants) for fast solvers like multigrid and multipole today. Within multilevel methods, two key algorithmic decisions are the definition of the coarse problem and construction of the operator for the coarse problem. Numerous types of adaptive multigrid schemes are currently being explored, which automatically infer these crucial algorithmic elements from observing slowly converging modes when multigrid is applied with default choices.

We have considered so far research issues that attempt to improve solvers given only a limited interface (namely the stiffness matrix) with upstream processes and we have looked upstream of the solver, to problem formulation and discretization to exploit a richer interface. There are also many research issues in solvers that look downstream, to implementation in software and execution on high-performance architectures.

Solvers occupy a high percentage of the execution time of the application, so there is a strong premium on making them perform at a high percentage of the hardware peak. When measured against the peak of the limiting hardware resource, for a given algorithm/hardware combination, today's solvers perform rather well. For PDE-based problems with large sparse Jacobians running on cache-based microprocessors, the limiting resource is usually memory bandwidth, as measured, e.g., by the Stream benchmark. The most efficient of today's solvers deliver a high percentage of the Stream benchmark; however, this may be only a low percentage of floating point peak. There is therefore a vigorous research program to rewrite solvers in both arithmetically neutral ways (e.g., reordering and reblocking the same operations) and in algorithmically different ways (e.g., multi-seeded Krylov spaces) to obtain more reuse of cached matrix elements.

Today's state of the art solvers involve numerous tuning parameters, the selection of which can be critical to obtaining optimal performance. Examples of such parameters are: the ratio of coarsening between levels in a multilevel scheme, the means of constructing coarsening operators and interlevel solution transfer operators, the type and number of smoothings on each level, the size of Krylov subspaces, inner iteration tolerances in inexact Newton methods, Newton robustification parameters (e.g., for line search and trust region methods), continuation parameters, and so forth. The list can run into dozens when combined with parallelization parameters, such as subdomain overlaps, or parameters that govern the dropping of coupling in a preconditioner where full coupling would lead to communication bottlenecking. The number of algorithmic tuning parameters is overwhelming to users, who must already manipulate a host of discretization parameters and physical parameters and would prefer to regard the solver as a black box. For this reason, there is an active interest in self-tuning solvers. Many approaches seem possible and are under investigation, from theoretical models for convergence tolerances in inner-outer iterations, to machine learning methods that are free of numerical analysis theory altogether. An even more basic type of algorithmic "tuning" is algorithm selection, based on problem and machine parameters, such as symmetry, sparsity, definiteness, the availability of an explicit matrix representation, the availability of sufficient memory, the processor granularity, the topology of the communication network, and so forth.

In order to guide research in solvers, research is also needed into effective and convenient means of profiling codes and visualization of performance, which poses different problems than the classic subject of scientific visualization, particularly when a simulation is hosted on thousands of processors. Since the solver may be the least scalable phase of a simulation, in terms of stressing the communication network, the memory bandwidth, the vector length or register set size, or the ratio of load-store units to floating point units, there is a profitable interaction of solver researchers with hardware architects, as well. Solvers provide an important target for architects, because solver infrastructure is common to numerous scientific codes.

Other solver research issues include the use of high-level algorithmic description languages and porting tools that make it possible to deliver a single solver library for calling by the host of languages employed in scientific computing (Fortran 77/90/95, C/C++, Python, Java, Matlab, and so forth).

Metrics of Success

There are numerous *desiderata* for solvers, and for each one metrics can be devised that enable progress in solver software to be quantified in some way. The most important property of a successful solver, as emphasized above, is optimality. The complexity of a solver should ultimately be brought down to the same order as the complexity of other dominant phases of a simulation, such as residual evaluation over the grid (often called one "work unit") or the cost of doing a particle-push. The ultimate requirement on a

solver is linear growth of work in the input size. Any superlinear terms (beyond a logarithmic factor) should be in quantities much smaller than the input size, such as the number of unknowns on the coarsest grid in a hierarchical algorithm or the number of unknowns in a subdomain. Thus, there will always be a role for an efficient sparse direct solver, as a component of a hierarchical algorithm.

One would also like a solver to have guaranteed robustness within a well-defined class of inputs, and to fail gracefully when those inputs are violated. For instance, one would like to be able to back off to a more conservative though perhaps less efficient solver, or to terminate with a clear identification of the problem, so that the user can repair algorithm tuning, choose a replacement algorithm, or reformulate the problem.

A solver must have not only optimality in the mathematical (convergence) sense, but implementation scalability on thousands of processors, at least in the sense of weak scaling, in which problem size and processor number are scaled in direct proportion. (There are ambiguities in "weak scaling," such as whether to count the number of equations per processors or the number of Jacobian nonzeros per processor, and this notion therefore interacts with other algorithmic choices, such as local adaptivity, coarsening ratios, fill limits, etc.)

A solver should be efficient, in the sense of delivering a high percentage of peak of the limiting architectural resource, and ideally in terms of its use of any expensive resource.

A solver library should make efficient use of a user's time, in the sense of allowing a user to relatively rapidly prototype to find a solver that works, and then of allowing the user to systematically tune the solver for performance. A good solver library allows the user to understand important tradeoffs that are available, such as working set size versus execution time, number of inner iterations versus number of outer iterations, robustness versus efficiency, etc.

Given the diversity of solvers and the absence of a "one-size-fits-all" approach, solver libraries should interoperate as much as possible at a component level, for "plug and play" modularity, reuse, extensibility, and composability. For instance, it should be possible to make one solver a preconditioner for another solver, and to register user-developed solver components with the library.

Ideally, solver technology should not simply follow advances in simulation and respond to them; solver technology can be a pathfinder for some aspects of large-scale simulation. For instance, a trend in many areas of computational physics, from climate modeling to magnetically confined fusion, from astrophysics to combustion, is the need to combine multiple types of physics, previously investigated separately, into a single code, so that two-way feedback mechanisms can be studied. The coupling of models has natural expression in terms of the blocking of matrices. Classical physical filtering schemes have analogues in Schur complementation. The language and knowledge of the solver community can enrich numerous other modeling communities.

To more effectively deliver solvers to applications, standard interfaces are a desirable. The ideal interface is multilayered, allowing a naïve user to specify minimum information and accept defaults while allowing a sophisticated user to tune the algorithm

in great detail. The unavoidable complexity of optimal algorithms means that their insertion into demanding applications is best achieved through active collaborations between scientists, mathematicians, and computer scientists. Better clearinghouses of information, such as custom short courses and meetings of principal investigators are important.

The ultimate measure of success is not an exponent of complexity, a percentage of peak, or a robustness guarantee, but the freedom of computational scientists to move ahead without bumping into the solver at every turn. As succinctly put by accelerator designer Kwok Ko of SLAC at the SCaLeS workshop, “Success is a moving target - as long as the science is moving forward, we’re successful.”

Transport Methods

In a wide variety of applications, a significant fraction of the momentum and energy present in a physical problem is carried by the transport of particles. Depending on the circumstances, the types of particles might involve some or all of photons, neutrinos, charged particles, or neutrons. In application areas that use transport, the computational time is usually dominated by the transport calculation. Therefore, there is a potential for great synergy; progress in transport algorithms could help quicken the time to solution for many applications.

The complexity, and hence expense, involved in solving the transport problem can be understood by realizing that the general solution to the Boltzmann transport equation is seven dimensional: 3 spatial coordinates, 2 angles, 1 time, and 1 for speed or energy. Low-order approximations to the transport equation are frequently used due in part to physical justification but many times simply because a solution to the full transport problem is too computationally expensive. An example is the diffusion equation, which effectively drops the two angles in phase space by assuming that a linear representation in angle is adequate. Another approximation is the grey approximation, which drops the energy variable by averaging over it. If the grey approximation is applied to the diffusion equation, the expense of solving what amounts to the simplest possible description of transport is roughly equal to the cost of implicit computational fluid dynamics. It is clear therefore, that for those application areas needing some form of transport, fast, accurate and robust transport algorithms can lead to an increase in overall code performance and a decrease in time to solution.

The seven-dimensional nature of transport means that factors of 100 or 1000 improvement in computer speed or memory are quickly absorbed in slightly higher resolution in space, angle, and energy. Therefore, the biggest advances in the last few years and in the next several years will be driven by algorithms. Because transport is an implicit problem requiring iteration, the biggest gains are to be made in finding faster techniques for acceleration to convergence. Some of these acceleration methods are very application specific because they are physics based; others are very general because they address the mathematics of the transport equation. Funding more research in the latter area could have a large impact on many physics applications. Usually it is a collaboration of someone with a tough problem to solve and someone with a new idea that makes the big advances. More heads are needed to continue the progress of the last few years.

Unfortunately, transport as a discipline is not taught in many graduate schools. Students and researchers too often pick up transport theory in pieces on an *ad hoc* basis. Therefore, they don't know the published literature and existing techniques. Knowledge of advances in one application area often takes years to propagate to other application areas.

Impact on Applications

Applications abound where transport is required. In astrophysics, the life cycle of the stars, their formation, evolution, and death all require transport. In star formation and

evolution for example, the problem is a multi-physics one involving MHD (magneto-hydrodynamics), self-gravity, chemistry, radiation transport, and a host of other phenomena. Supernova core collapse is an example where 3D, multi-group, multi-angle photon and neutrino transport are important in order to model the explosion mechanism. The spectra and light curves generated from a supernova have generated a wealth of data. In order to make a connection between simulation data and observational data and in order to remove systematic errors in supernova standard candle determinations of cosmological parameters, 3D, multi-group, multi-angle radiation transport is required.

The simulation of nuclear reactor science poses a similar set of challenges. In order to move beyond the current state-of-the-art for such calculations, several requirements must be met:

- A description based on explicit heterogeneous geometry instead of homogenized assemblies.
- Dozens of energy groups instead of two.
- The use of 3D high-order transport instead of diffusion.

These requirements will allow for accurate real-time simulations of new reactor operating characteristics, creating a virtual nuclear reactor test bed. Such a virtual reactor would enable assessments of the impact of new fuel cycles on issues like proliferation and waste repositories. With a 1000-times increase in computer power, accurate virtual reactors could reduce the need to build expensive prototype reactors.

In the broad area of plasma physics, ICF (Inertial Confinement Fusion) and to a lesser extent MFE (Magnetic Fusion Energy) require the accurate modeling of photon and charged particle transport. For ICF, whether one is dealing with direct drive through photon or ion beams or dealing with indirect drive via thermal photons in a hohlraum, the accurate transport of energy around and into tiny capsules requires high-order transport solutions for photons and electrons. For direct drive experiments, simple radiation treatments suffice (i.e. laser ray tracing with multi-group diffusion). Although the radiation treatment can be rather crude, direct drive experiments require sophisticated models of electron transport. In indirect drive such as at NIF, laser energy is converted into thermal x-rays via a hohlraum, which in turn is used to drive some target. In order to accurately treat the radiation drive in the hohlraum and its attendant asymmetries will require a radiation transport model with NLTE opacities for the hohlraum. The ability to generate NLTE is a tremendous computational challenge. Currently, calculating such opacities in-line comes at a great cost. Typically, the difference between an LTE transport and NLTE transport calculation is a factor of 5. This fact has sparked research into alternatives such as tabulating steady state NLTE opacities or by simplifying the electron population rate equations so that their calculation is fast. However, all of the alternatives suffer from drawbacks that inhibit their widespread use.

In the coming years, simulations of NIF (National Ignition Facility) experiments will be crucial in attaining the goal of ignition. The simulations need to be predictive rather than

after-experiment fits; therefore, high-order transport coupled self-consistently to other nonlinear physics is a requirement. With a 1000-fold increase in computer power, these types of simulations are feasible.

Infrared transport plays a major role in internal combustion engines, under-hood cooling for automobile engine compartments, turbine design, wildfire models, and combustion in general. One of the challenging aspects of infrared transport is that infrared radiation is often propagated across nearly transparent regions. The traditional solution method for these types of applications is the view-factor method that uses ray tracing between different surfaces in a problem. Unfortunately, this class of methods scale as N^2 rather than N, where N is the total number of cell surfaces in the calculation. As one refines a calculation, the time-to-solution quickly becomes unacceptably large even with a perfect parallel implementation of the algorithm. Therefore, new algorithms that scale as N need to be developed for these important commercial applications. Adaptation will be an essential element of such algorithms.

There exist a myriad of other applications requiring some form of transport. We close this section by just listing some of them: atmospheric physics, medical diagnostics and treatment, plasma diagnostics, combustion, and non-destructive testing.

Research Issues

As mentioned in the introduction to this chapter, the biggest payoff in algorithm improvements is probably in the area of acceleration of nonlinear iterations to convergence. Pre-conditioners for iterations can be physics-based (diffusion accelerating high-order transport) or mathematics-based (Krylov wrappers around existing iteration techniques). Much progress has been made recently, but much more needs to be accomplished.

Since transport is seven-dimensional, an obvious approach to reducing memory requirements is to use adaptive meshing in all dimensions. In spatial dimensions, much progress has been made with AMR (Adaptive Mesh Refinement) techniques. Currently, most applications use the same spatial mesh for all physics components. In the future, each component may have a mesh optimized for its own needs. Applying these techniques to the angular and energy dimensions of phase space is an area of research that is still in its infancy. The benefits are still to be achieved.

Some of the other important areas of research: Newton-Krylov techniques for tightly coupled self-consistent multi-physics simulations (see the chapter on Multiphysics Techniques), hybrid deterministic and stochastic transport techniques, higher-order time integration, subgrid models for stochastic media simulations, etc.

Metrics for Success

Currently, most of the applications mentioned above do not use full transport but rather some approximation. The approximations currently used are grey or multi-group diffusion and sometimes diffusion with variable Eddington factors. In extreme cases, transport effects are not included at all. These situations arise not because people are ignorant of the importance of transport, but rather its high cost forces code developers to

seek cheaper alternatives. Success in this field would be the enabling of high-order transport in applications where it is currently ignored or approximated.

With the removal of severe physics approximations from the codes, a consequence of the above success is the use of virtual simulations as partners with or even replacements for experiments.

Another mark of success is cultural. Success in this area implies an increase in young students doing transport. The ability to attract bright young people to this area is desperately needed if the barriers are to be overcome. Attracting more people to transport can begin by increasing cross-fertilization of transport ideas across disciplines. A measure of success on this front would be the increase in communication amongst the atmospheric, astrophysics, and nuclear engineering communities for example.

Uncertainty Quantification

Uncertainty quantification (UQ) is the science of combining imperfect information from multiple sources to reach conclusions and to assess the validity of these conclusions. UQ is thus concerned with the transformation from data to knowledge to decisions. It is fundamental to all three endeavors, theory, simulation and experiment, that define modern science. UQ holds the promise of integration across all of them. It defines the link between science and the decision process.

Uncertainty quantification starts with the identification and characterization of error or uncertainties from all steps in the sequence of approximations that leads to a computational model prediction. This includes data error (input data, equation parameters, boundary data, etc), numerical discretization error, and physical modeling error. The characterization is usually a probability model (often expressed as a statistical distribution or error bars in simple cases). These sources of error and their influence on the resulting conclusions must be carefully quantified. The result of the UQ analysis is a probability description of the possible future outcomes, depending on uncertain knowledge of the state of the world, the scientific laws which govern it and the solution procedure for solving these laws.

Given this information, it is possible to manage uncertainty, through assessment of risks associated with decisions, and through prioritization of future research to reduce uncertainties. Uncertainty quantification, thus defined, combines elements of statistics, probability, numerical analysis, computer science, and application science in a close working relationship.

Numerical simulation is fundamentally different from theory and experiment with regard to accuracy and validity. Experiments study the real world while simulations only study a man-made model or abstraction of nature, not nature itself. Theories too are models of nature, but theories are transparent. Every assumption and every step of a theory is generally accessible to the scientific community. Simulations are generally not accessible to anyone but the author and a few collaborators. The assumptions, the details of the model and the solution of the model are opaque to the general community. Most simulations are sufficiently complex that it is not practical to try to reproduce the simulation. Establishing the accuracy and fidelity of simulations is therefore an essential part of any simulation activity.

Indeed, because a simulation is only an opaque model of reality, computational scientists may be asked why anyone should believe a computer simulation? The solution of the model problems likely contain errors, and even if the model problems are solved correctly, we don't know if the simulation model is the right model or includes all the important effects. To prove that simulations model reality with some fidelity, a strong verification and validation program is an essential part of every modeling activity.

Verification is the determination that the model is solved correctly, that the model does not contain coding errors, mathematical errors or other mistakes. *Validation* is the determination that the simulation models capture enough of the critical effects that the simulation accurately describes nature.

While verification is simple in concept, for large-scale, complex codes that incorporate many different phenomena, complete verification is usually impossible. First, simple analytic solutions usually do not exist for complicated, non-linear, interacting models. Correct convergence rates are often difficult or impossible to determine from the model due to non-linear coefficients, tabular data, or non-analytic coefficients and algorithms. The usual practice is to verify each piece of the model as thoroughly as possible, and then verify the integrated code with limiting cases. Validation is done by simulating experiments or observations, and then comparing the calculated results with measured data. Discrepancies between the observations and the calculations point to deficiencies or errors in the model when verification has been performed to reduce the probability of numerical error. The model is then improved (and the simulation verified again) until the experimental results can be reproduced.

For simulation to be a successful partner with theoretical and experimental science, it will have to work hand in hand with theory and experiment. An essential element of that collaboration will be a strong verification and validation program. V&V as a discipline is still in its infancy for simulations of the type envisaged for SCALES, so it will important to define a V&V strategy for each science area tailored to the different characteristics and needs of that area.

UQ starts with, or assumes, that a V&V strategy has been employed. Going beyond V&V, UQ assesses the magnitude of errors in the prediction process. These will comprise simulation errors, physical modeling errors and, experimental or observational data errors. The V&V program does not show these errors to be zero, of course, because they can never be zero. UQ is the science of assessing errors in prediction resulting from these (residual, or post V&V) errors in the prediction process.

Impact on Applications

Simulation science in all application areas will become an indispensable partner in major economic decisions if it provides credible uncertainty quantification of its predictions. Further, the development of UQ tools and UQ culture will speed up progress in any application by strategic allocation of resources in areas that generate the most uncertainty.

A UQ framework of statistical, mathematical, and computational collaboration and tools, going beyond the intuitive methods that served the practicing computational scientist in the past, will facilitate a new culture in computational science applications. For example, an investment decision between experimental physics research and adaptive grid generation research can be guided by the relative contributions to uncertainty by the physics model and by the discretization. UQ enables the common business practice of applying limited resources where they can have the most impact.

The recognition that UQ requires input from statistical sciences enlarges the computational science collaborative effort that so far has included applied mathematics, computer science, and application science. This inclusion will have a positive effect that reaches far beyond a narrow view of UQ to a larger context that includes the entire

scientific method cycle of computational experiments. We emphasize that it is investment in the collaborative effort of the disciplines that provides as much progress in computational science applications as does investment in novel and more powerful computer hardware.

Consider an example from the environmental sciences, where a legal requirement mandates an assessment whether some contaminant will remain within a storage area for extended periods of time under a variety of different scenarios. The scenarios may involve mechanical failure of containment systems, geological uncertainties, changes in precipitation patterns and changes in human activity. Each scenario requires a separate simulation study. Environmental investment decisions, if guided by simulations that include credible UQ, will typically prevent far greater future investments in environmental cleanup due to a containment failure.

Weather forecasting and climate prediction involve a wide range of uncertainties, from under resolved simulations and subgrid models of uncertain accuracy to data uncertainties involving initial conditions. The variability of the weather and the unpredictability of the climate introduce stochastic elements that complicate the enterprise. Major decisions ranging from specifying evacuation areas in an approaching hurricane path to international treaties for environmental protection can be made more effective with predictions that include UQ.

Many astrophysics problems are complicated by multiphysics and multiscale issues, so that the simulations are under resolved, and probably will be for some time. Careful quantification of various sources of error in the simulations will guide the most effective investment of resources to issues that provide the greatest value in uncertainty reduction. UQ will also aid in drawing scientific conclusions and in distinguishing between differing physical explanations and mechanisms for observed phenomena.

Research Issues

Bayesian and other methods of statistical inference depend on probability distributions for solution and observational error to assess and refine the uncertainties in the model formulation. These uncertainties and those of the forward simulation itself combine to define the uncertainty of the outcome, i.e. the prediction. Thus error analysis is central to prediction. But it is very expensive. If pursued naively, large ensembles (to assess statistics of error) of super accurate simulations (to give a comparison to define the error) are needed. A major research issue is to find efficient ways to model and understand error, as it is introduced into the prediction and transmitted from input to output. For multiphysics or multiscale simulations, these issues are acute, as there are many stages to be simulated between input and output, and the possible accumulation of error across stages or its propagation between them is a key research issue. Many applications have very stringent requirements of the allowed errors in the predictions. This fact leads to the analysis of the tails of the error probability distributions, which are likely not to be modeled well by Gaussian probabilities.

Large-scale simulations are necessarily high-dimensional, particularly with respect to the numbers of input parameters and outputs. Resultant problems include search over high-dimensional spaces (for model tuning, optimization or sensitivity analysis), links among multiple scales and reduction of model output to useful information (using data mining or pattern analysis, for example). One participant estimated that only 5% of output is ever examined; this estimate seems optimistic even currently, let alone for future simulations. Just as the simulations and observations are high dimensional, so are their errors. But reduction to a low dimensional model of error will improve the robustness of the analysis and reduce the information needed to characterize these models.

The statistical sciences, computer science and applied mathematics bring multiple, complementary approaches to the "curse of dimensionality." From statistics, these include design of experiment, Latin hypercube sampling, and emerging methods for model evaluation that address both systematic and random error. From computer science these include automatic differentiation to aid sensitivity studies and genetic algorithms for efficient Monte Carlo sampling. Those from applied mathematics techniques include upscaling, homogenization and asymptotic analysis. Some approaches, such as data mining and sensitivity analysis, span multiple infrastructure disciplines. The thread of dimensional reduction runs through all these ideas, and is a central technology developed as part of UQ.

The research needs are significant. Despite recent progress, almost all of these approaches lack the scalability necessary to cope with ultra-scale simulations. Lower-dimensional approximations of models, derived from a combination of domain knowledge, applied mathematics and statistics, are a promising path. Other ways to merge contributions from multiple disciplines are necessary.

The simulation effort to determine model parameters (i.e. the inverse problem) from observations of the physical system is at least as important and as time-consuming as are forward simulations to obtain predictions for system behavior. Going beyond simple maximum likelihood ideas, the inverse problem solution has a range of possible values, i.e. a measure of uncertainty. From this point of view, the inverse problem becomes stochastic and has much in common with optimization in the presence of uncertainty, or stochastic optimization. Many of the application areas have a need for inverse problems considered stochastically and for optimization with UQ. We mention plasma science (optimize the design of a burning plasma), materials science (optimize accurate trial functions), environmental remediation (inversion of 3D multisensor data), chemistry (optimize with a complex energy surface) and astrophysics (adaptation to parameter drift).

Uncertainty quantification depends on data. Just as the simulations depend on data, uncertainty quantification will normally require knowledge of the errors, uncertainties and variabilities in this data. This key issue should be understood clearly. Probabilities are studied in computer experiments by the generation of ensembles of multiple related simulations. Errors are identified by comparison to a more accurate simulation, generally involving a larger numerical effort (finer grids) or more accurate physics. Both the increase in accuracy to assess errors and the ensembles of simulations to generate

statistics for the description of error require a large increase in the computational effort. This effort will involve cycles and personnel.

Often the data is assembled and interpreted through an independent simulation effort (called the inverse problem), and the uncertainty assessment of this step is an input to the uncertainty quantification of the (forward) simulation. In addition, UQ depends on assessments of error in the numerical discretization of the equations to be solved and in the physical model or conceptual scenario that leads to these equations. Assessment often takes the form of a probability, either described by its mean and variance, or as a full distribution. UQ is dependent on these probability assessments. It is thus dependent on the application scientists to provide them, or to provide the simulation codes so that the UQ program can work jointly with the application scientists to generate these probabilities. The UQ program thus depends on the availability of the necessary cycles and the computational science manpower. Cycles will also be needed to test methods for extraction of probability models from codes. As with the simulation itself, the output of UQ (probability assessments) requires verification and validation, and the necessary cycles and personnel to carry this out.

Metrics for Success

The primary measure of success is for UQ to become an accepted part of the infrastructure of computational science. As such, UQ will become a metric of success for other investment decisions in computational science. To achieve this goal, it must be capable of specifying meaningful and credible bounds on the uncertainties associated with predictions, within a feasible computational and scientific effort. Further, it must be capable of partitioning the uncertainties and assigning them to the various sources within the simulation process.

Intermediate measures of success include the new scientific research:

- The creation of software tools, both generic and domain specific, suitable for large-scale problems, solved on massively parallel computers.
- Efficient methods to assess, model, and understand the various errors in the simulation process will be needed, including those generated by the difficult multiphysics and multiscale applications. Efficient methods going beyond Gaussian statistics to study higher moments, rare events and full probability distributions for UQ and error statistics will be needed.
- Methods to mitigate the large ensemble sizes needed to study the statistics of error and its propagation through the simulation and prediction process will be required.
- Methods for inverse problems and optimization with UQ.

Computer Science for Large-scale Simulation

The role of computer science in ultrascale simulation is to provide tools that address the issues of **complexity**, **performance**, and **understanding**. These issues cut across the computer science disciplines. An integrated approach is required in order solve the problems faced by applications.

One of the major challenges for applications is the complexity of turning a mathematical model into an effective simulation. There are many reasons for this. It is still too hard to turn algorithms into practice because of the complexity of the design and the realization of the code. Current programming models and frameworks do not provide sufficient support to shield domain experts from the details of parallelism. Even after an application code produces scientific results, it is too hard to get performance. Code tuning efforts needed to match algorithms to current computer architectures require lengthy analysis and experimentation.

Once an application runs effectively, the next hurdle is often saving, accessing, and sharing data. And once the data is stored, since ultrascale simulations often produce ultrascale-sized datasets, it is too difficult to process, investigate, and visualize the data in order to accomplish the purpose of the simulation---to advance science. These difficulties are compounded by the problems faced in sharing resources, both human and computer hardware.

Despite this grim picture, prospects for placing usable computing environments into the hands of scientific domain experts are improving. In the last few years, there has been a growing understanding of the problems of managing complexity in computer science, and therefore of their potential solutions. For example, there is a deeper understanding of how to make programming languages expressive and easy to use without sacrificing high performance on the sophisticated, adaptive algorithms.

Another example is the success of component-oriented software in some application domains; such “components” have allowed computational scientists to focus their own expertise on their problems while exploiting the newest algorithmic developments. Many groups in high-performance computing have tackled these issues, with significant leadership from the Department of Energy. Fully integrated efforts are required to produce the qualitative change in the way application groups cope with the complexity of designing, building, and using ultrascale simulation codes.

One of the drivers of software complexity is the premium on performance. The most obvious aspect of the performance problem is the performance of the computer hardware. While there have been astounding gains in arithmetic processing rates over the last five decades, users often receive only a small fraction of the theoretical peak of processing performance. Further, there is a perception that this fraction is declining. This perception is correct in some respects.

For many applications, the reason for declining percentage of peak performance is the relative imbalance in the performance of the subsystems of high-end computers. While raw performance of commodity processors has followed ``Moore's Law'' and doubled every 18 months, the performance of other critical parts of the system, such as memory and interconnect, have improved much less rapidly, leading to less balanced systems. Solving this problem requires attention to system-scale architectural issues.

As with code complexity issues, there are multiple on-going efforts to address hardware architecture issues. Different architectural solutions may be required for different algorithms and applications. A single architectural convergence point, such as that occupied by current commodity-based terascale systems, may not be the most cost-effective solution for all. A comprehensive simulation program requires that several candidate architectural approaches receive sustained support to explore their promise.

Performance is a crosscutting issue, and computer science offers automated approaches developing codes in ways that allow computational scientists to concentrate on their science. For example, techniques that allow a programmer to automatically generate code for an application that is tuned to a specific computer architecture addresses both the issues of managing the complexity of highly-tuned code and the problem of providing effective portability between high-performance computing platforms. Such techniques begin with separate analyses of the “signature” of the application (e.g., the patterns of local memory accesses and inter-processor communication) and the parameters of the hardware (e.g., cache sizes, latencies, bandwidths). There is usually lots of algorithmic freedom in scheduling and ordering operations and exchanges while still preserving correctness. This freedom should not be arbitrarily limited by particular expression in a low-level language, but chosen to best match a given application-architecture pair. Similarly, the performance of I/O and dataset operations can be improved significantly through the use of well-designed and adaptive algorithms.

Computer science also addresses the issue of understanding the results of a computation. Ultrascale datasets are too large to be grasped directly. Applications that generate such sets already today rely on a variety of tools to attempt to extract patterns and features from the data. Computer science offers techniques in data management and understanding that can be used to explore data sets, searching for particular patterns. Visualization techniques help scientists explore their data, taking advantage of the unique human visual cortex and visually stimulated human insight. Current efforts in this area are often limited by the lack of resources, both in terms of staffing and hardware.

Understanding requires harnessing the skills of many scientists. Collaboration technologies help scientists at different institutions work together. Grid technologies that simplify data sharing and provide access to both experimental and ultrascale computing facilities allow computational scientists to work together to solve the most difficult problems facing the nation today. But while these technologies have been demonstrated, much work remains to make them a part of every scientist’s toolbox. Key challenges are in scheduling multiple resources and in data security.

Table 1summarizes some of the critical computer science technologies needed by many applications. These reflect some of the issues considered by the following eight working

groups. **Visual Data Exploration and Analysis** considers the understanding the results of computation through visualization. **Computer Architecture and Ultrascale Simulation** looks at the systems on which ultrascale simulations run and for which programming environments and algorithms (described in the section on computational mathematics) must provide tools.

Programming Models and Component Technology for High End Computing seeks better techniques to turn algorithms into efficient, maintainable programs. **Access and Resource Sharing** looks at how to both make resources available to the entire research community and to promoting collaboration. **Software Engineering and Management** discusses the techniques and tools used to manage the complexity of code development. **Data Management and Analysis** looks at managing and understanding the increasing large volumes of data produced by ultrascale simulations. **Performance** seeks an understanding the art of achieving high performance with the goal of making it a science. **System Software** considers the basic tools that support all other software.

These interrelated issues point to the necessary synergy between the applications, algorithms, and computer science. Only an integrated effort can ensure that the right problems, on the right scale, with the right techniques, are addressed.

Table 1: Some critical computer science technologies required by many applications

Collaborative code development Scalable I/O Data understanding Memory bandwidth and latency Parallel I/O Remote access & data sharing	Software design, development, and support Metadata management Data migration Interconnect performance Component model for parallel computing Programming environments and languages for HPC

Though these chapters cover a broad range of topics in computer science, a few common themes emerge. All focus on the computer science needs of applications. Consequently, one common theme is the importance of sustained support for software maintenance, documentation, training, and evolution if the best practices and newest techniques are to be adopted for use in computational science applications. Many applications have lifetimes measured in decades and must be confident that any tools that they depend on will continue to be available. Sustained support for these efforts is also necessary to ensure a continued supply of practitioners—many working groups were concerned about the small number of University graduate programs in computational science and the numbers of graduating students with strong backgrounds in computational science.

Another theme is computer science at large scale — large data sets, complex programs, large teams of programmers and computational scientists, as well as the need for enormous amounts of computing power. The impressive achievements in commodity computing are often applicable to high performance scientific computing. But because of the scale requirements of HPC applications, not all needs can be met with commodity

software and hardware. Instead, solutions are needed that leverage the accomplishments in commodity computing but that address the unique needs of high-performance computing.

The computer landscape of today is very different from that of ten years ago. This has been achieved primarily through evolution rather than revolution: faster processors, denser memory, better interconnects, and better programming models. As we look at the barriers and goals for the next decade, we must remember that no single approach, whether it is hardware or software, will satisfy the needs of every important application. The following chapters present directions for short-, mid-, and long-term research that are needed to ensure continued and rapid advancement of computational science.

Visual Data Exploration and Analysis

Scientific visualization is the transformation of abstract information into images. It plays an integral role in the scientific process by facilitating insight through analysis into observed or simulated phenomena. Visualization as a discipline spans many research areas, from computer science cognitive psychology and even art. Yet the most successful visualization applications are created when close synergistic interactions between visualization experts and domain scientists are part of the algorithmic design and implementation process, leading to visual representations with clear scientific meaning. Visualization is used to explore, debug, present, analyze, and gain understanding. Visualization is literally everywhere. Images are present in this report, on television, on the Web, and in books, journals, and magazines. The common theme is the ability to present information visually that is rapidly assimilated by human observers, and transformed into understanding or insight.

Impact on Applications

As an indispensable part a modern science laboratory, visualization is akin to the biologist's microscope or the electrical engineer's oscilloscope. Whereas the microscope is limited to small specimens or use of optics to focus light, the power of scientific visualization is virtually limitless. Visualization provides the means to examine data that can be at galactic or atomic scales or at any size in between. Moreover, unlike the traditional scientific tools for visual inspection, scientific visualization offers the means to create visual representations of abstract concepts that are otherwise unseeable. Trends in demographics or changes in levels of atmospheric CO_2 as a function of greenhouse gas emissions are familiar examples of such unseeable phenomena.

Over time, visualization techniques evolve in response to scientific need. Each scientific discipline has its own language, verbal and visual, for communication. The visual language for depicting electrical circuits is very different from the visual language for depicting theoretical molecules or trends in the stock market. No single visualization tool can serve for all science disciplines. Instead, visualization researchers work hand in hand with domain scientists as part of the scientific research process to define, create, adapt, and refine software that to incorporate domain knowledge and therefore "speak the visual language" of each scientific domain.

Research Issues

The visualization research topics presented in this section are a blend of computer science technologies for realizing needed growth in visualization capacity and capability and new visualization technologies that address specific science needs. The challenges posed by modern computational science performed on large-scale computer systems are acute: not only is the amount of data becoming larger, but the complexity of the data itself is growing. Because of their fundamental design, visualization tools from earlier periods simply do not exhibit the capacity to process large scientific data sets. Similarly, the

capabilities of earlier tools are not adequate to effectively present the meaningful information inherent in large, multidimensional data.

The topics discussed in this section take aim at known challenges posed by modern computational scientific research. Among these challenges are the fact that data sizes grow ever larger as computing capacity increases. With larger and more detailed simulation comes the need for more sophisticated visual analysis techniques, as well as the need for visualization infrastructure that provides the ability to simultaneously perform scalable data analysis that spans multiple data sets. Complicating matters even more is the case when multiple data sets are distributed across multiple sites. Even dramatic improvements in network technology cannot accommodate the "MxN" data movement required to aggregate data in this situation.

The most promising avenue for taking on large and distributed data visualization problems is parallel processing; task parallelism allocates specific tasks to processors in assembly-line fashion, and data parallelism spreads the workload for a single dataset across multiple processors. Both forms of parallelism require careful attention to design and implementation. Another challenge is the fact that as computing technology at large centers becomes more accessible to the research community, the remote user population will grow and will expect more support for scalable tools that provide the ability to perform scientific research from remote locations: data must be analyzed where it was created without incurring the cost of large-scale data movement across the wide area network. Design and implementation of "traditional visualization software," as well as most commercial visualization products, have not taken into account these challenges, which in fact are tomorrow's requirements.

High-Capacity Visualization

One of the most significant challenges facing visualization is the need to process and display very large scientific datasets. Significant early advances by the visualization community in this area have identified areas requiring research to meet the needs of emerging computational science programs. One such area is data models and algorithms for processing and visualizing time-varying data, which add complexity to the large-data visualization challenge. Technologies that are already used to accelerate static data processing (such as multiresolution representations) can be applied with some degree of success to the access and processing of independent time steps of dynamic, time-varying data. However, new algorithms that effectively accelerate visualization of time-varying data, particularly out-of-core methods, are sorely needed. New, related visualization technology that focuses on effective visual display of time-varying data will enable better scientific understanding of complex dynamic phenomena. Achieving these objectives requires careful attention to the architecture of pipelined and parallel visualization processing tools, along with effective use of high-resolution displays.

We can draw a parallel between gains reasonably anticipated through improved processing of time-varying data and the gains realized through the same multiresolution techniques used in simulation codes themselves. Figure 1 depicts a multiresolution technique known as adaptive mesh refinement (AMR). In AMR, a computational grid is locally refined to higher resolution in "regions of interest." A reactive chemistry

combustion simulation would refine the computational grids in regions of substantial amount of chemical reactivity, such as along a flame front. The primary benefit of AMR is that one can achieve very high spatial and temporal resolutions that are local in scale; the cost of local refinement does not propagate to the full computational grid. Another example where substantial efficiency gains are realized through AMR is astrophysics simulations. In these codes, the range of spatial resolution in the computational domain varies from the cosmological or interstellar level down to planetary scales, where most of the volume in between is empty. The exact amount of efficiency gain is difficult to generally quantify because the refinement is a function of the particular phenomenon being modeled as well as parameters that specify the maximum amount of permissible error.

AMR codes typically realize somewhere between one and two orders of magnitude in efficiency gain compared to non-adaptive approaches. The gain in efficiency spans the entire processing pipeline, starting with storage of data on disk, continuing through data movement and downstream CPU cycles for analysis, and including visualization. Use of AMR approaches for efficient representation of time-varying data is not a new idea, but its potential has not been realized because of lack of development. AMR is just one possible approach.

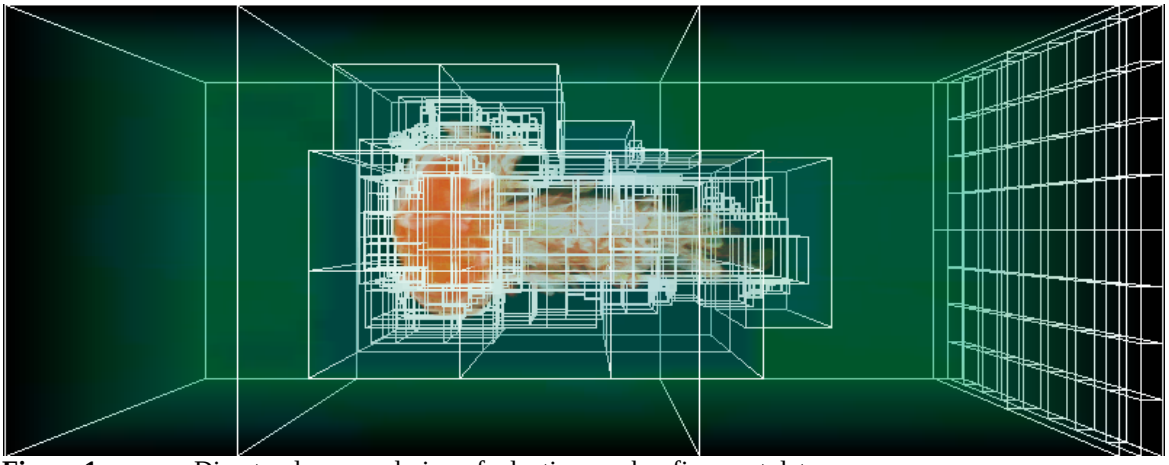


Figure 1. Direct volume rendering of adaptive mesh refinement data.

Remote Visualization

Remote visualization is an integral part of all our lives. When we watch the weather forecast on television, we are viewing a presentation of data assembled from a number of remotely located sources: satellite imagery, regional ground-based stations, weather balloon observations, and computer simulations that predict tomorrow's weather. This same metaphor applies to modern computational science, where large datasets are generated on supercomputers and are analyzed or viewed by remotely located researchers. The trend toward consolidated centers that provide extreme computing capabilities as centralized resources, combined with the increased size of generated data, produce an acute need for remote visualization capabilities. As research teams are increasingly composed of geographically distributed scientists, interactive and collaborative remote visualization technologies can help to accelerate scientific discovery

while reducing the costs associated with travel (see Figure 2). There is an overlap between the needs of remote visualization and the objectives of other DOE research areas. A user should be required to authenticate only once in order to use a vast web of distributed resources, and they should expect that their data streams are adequately secure. As remote and distributed applications evolve, suitable Grid infrastructure that supports single sign-on authentication and secure transmission of data streams must be developed and uniformly deployed across DOE facilities. These issues of remote data access are discussed in more detail in the Access and Resource Sharing.

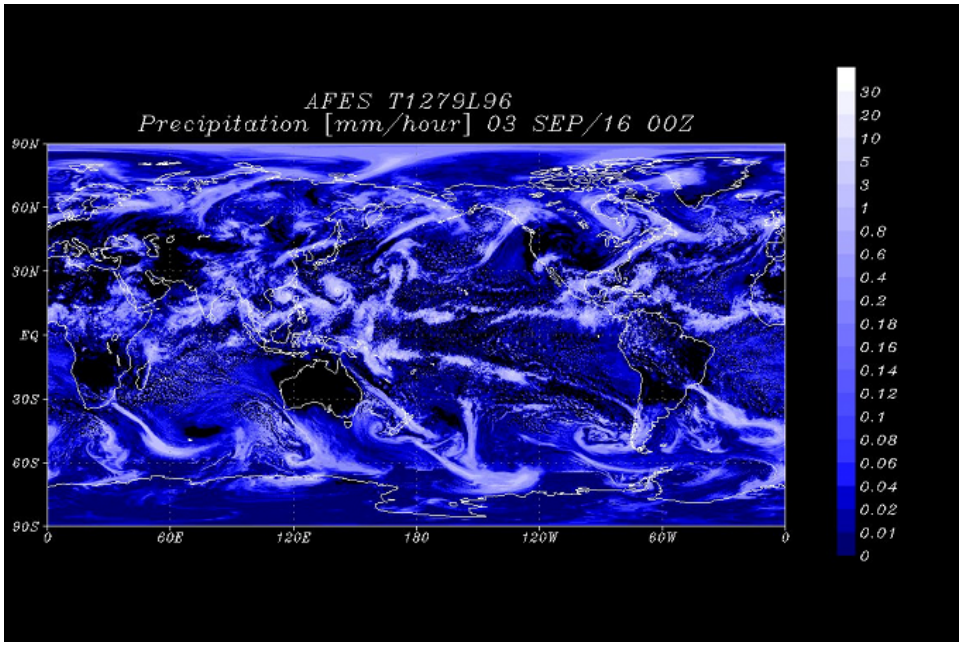


Figure 2. High-resolution datasets are computed at centralized facilities but are viewed by remotely located researchers. Remote visualization techniques help scientists make effective use of centralized facilities.

Multiresolution Methods

One avenue for addressing the problems posed by remote visualization is to enable the researcher to examine data at different resolutions. A quick examination of a low-resolution model or a statistical summary might reveal that no further inspection is necessary, thereby resulting in a significant time and resource savings. Alternatively, a low-resolution model can provide a visual roadmap for high-resolution exploration, allowing a researcher to select small, high-resolution subsets of a dataset for more thorough analysis. Creating such multiresolution representations for specific scientific domains is a research area unto itself. However, creating effective methods for visually presenting such multiresolution representations and enabling the interactive transition between visual depictions are both active areas of visualization research.

Advances in data modeling technology will help to create statistically valid or bounded-error representations of fields that are more compact than the original. Such

multiresolution techniques are important so that remote users may quickly examine simulation results, and have the option to “drill into” the raw, full-resolution data if desired. If possible, it is desirable to use techniques similar to, if not the same as, those used by the simulation itself. Adaptive mesh refinement is particularly attractive because it provides multiple levels of resolution that are scientifically significant.

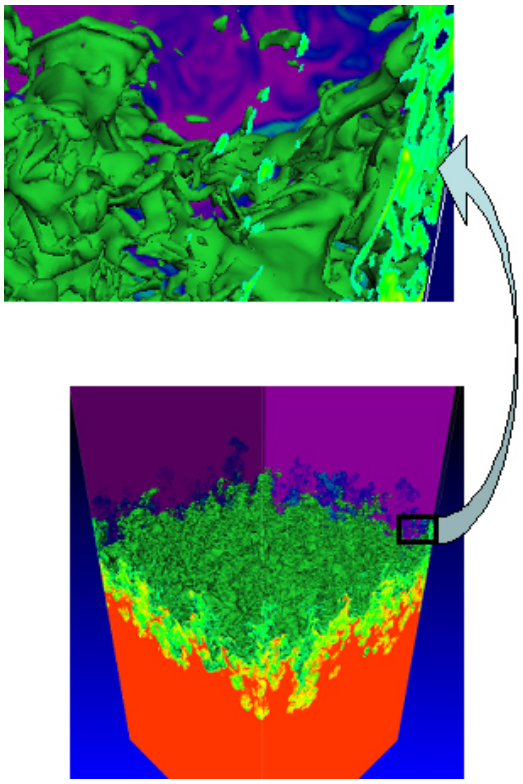


Figure 3. Multiresolution visualization requires specialized data models.

Multidimensional and Multivariate Visualization

Scientific computing has evolved to simulate phenomena at ever-increasing levels of fidelity and accuracy. Accurate modeling of phenomena often requires solving for more and more unknown variables. In order to facilitate scientific advances and provide insight into these complex systems, visualization technology is needed that can effectively display many variables simultaneously. The visualization challenge is compounded by the scientific need for comparative analysis of experimental and simulation data, as well as data obtained or computed over a period of time.

One approach to multivariate and multidimensional visual data analysis is based on the idea of “data mining.” In data mining, a user navigates between different datasets, or different resolutions of a dataset, based on observations that in turn raise questions or spark ideas. Another scenario would leverage off-line analysis to locate or track domain-

specific features in the data to assist in data navigation. This area of research not only spans visual data analysis but also includes the science of data summarization, along with efficient storage and retrieval of large and diverse data.

Coupling Analysis, Visualization, and Data Management

At the core of the visualization processing pipeline is technology for accessing, manipulating, and processing data. As data models and data management systems evolve to accommodate ever-increasing dataset sizes and locations, there is a corresponding need for visualization tools to take advantage of these emerging technologies that store, retrieve, characterize, and analyze data. Statistical analysis forms an integral part of data understanding; yet few techniques exist for visualizing error, uncertainty, and other statistical features. Identification and characterization of interesting features are highly domain-specific. Automatic detection and display of such features is a blend of statistical analysis, data management, and domain-specific visualization techniques. Through advances in visualization technology that include closer ties to data management technology (e.g., processing and display of statistical information), computational science programs benefit from increased visual data analysis capacity and capability.

"Behavioral" Visualization

As computer simulations increase in complexity, there is a growing need for visual representations of complex processes. One example is the behavior of optimization calculations in combinatorial algorithms. Visualization of algorithmic behavior, decision trees, and related "behavioral processes" provides insight into the operation and improvement of complex scientific software. A good example is how the search space in protein conformation is pruned to identify minimal energy conformations in complex molecules. Another example is the visual display of chemical pathways in combustion simulations, or metabolic pathways in cells (see Figure 4). The evolution of simulation programs requires new visualization techniques to facilitate scientific insight.

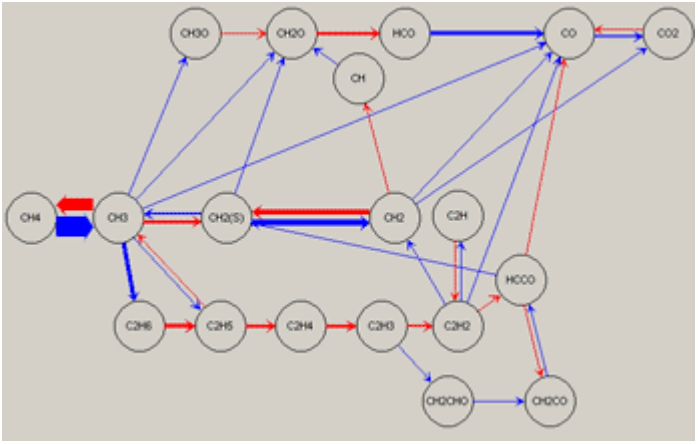


Figure 4. Chemical pathway visualization. The nodes represent species, and the edges represent flow of a conserved quantity, such as transfer of a particular element.

Delivering Visualization Technology to Application Scientists

Application scientists have indicated that the best software tools are those specifically tailored for their domain. Such tools provide results in a familiar "language" that is readily comprehensible and applicable to scientific research (see Figure 5). To develop such tools, visualization researchers must be part of the multidisciplinary science team performing the research. Even though each discipline needs tailored software tools, careful general-purpose software design and implementation will result in a "toolbox" of compatible components that can be combined in various ways to provide domain-specific solutions. Such components, with supporting data models, provide the "standards" to which disparate teams of visualization and science researchers can create compatible software tools. The evolution of a community-defined and supported software technology base will accelerate the growth of visualization research and its application to scientific domains through reduced duplication of effort and software engineering practices that promote reuse.

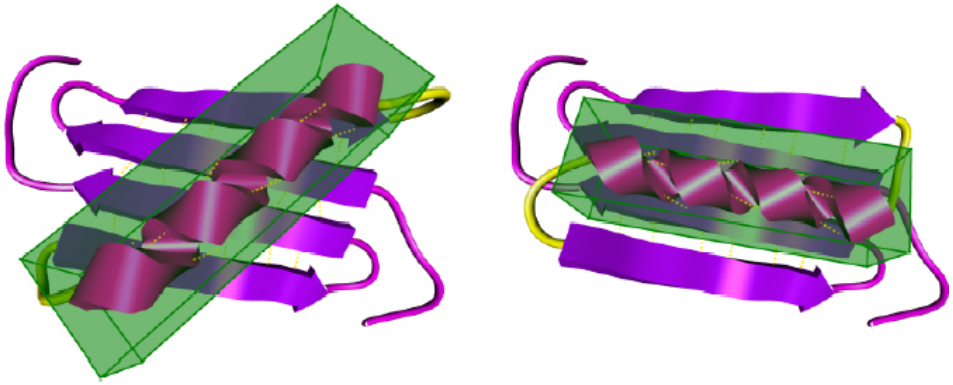


Figure 5. Visualization and manipulation of protein molecules is performed using "units" familiar to computational biologists – alpha helices and beta sheets.

Resources Required

The current model for funding visualization research and development tends to emphasize technology demonstrations. In contrast, science researchers need stable, production-quality software. The cost of ongoing software maintenance, documentation, training, and evolution far exceeds the cost of initial research and development. However, no funding mechanism exists to sustain these crucial activities. The traditional economic model of technology transfer from research into commercial products does not apply to scientific software, particularly visualization. The primary economic factor that results in a successful software commercialization—a large market that makes it possible to realize economies of scale—simply does not exist in the high performance computing world. Compared to traditional consumer markets for desktop publishing, photo editing, and so forth, the size of the market for high-performance visualization software is very small. As

such, software companies would be forced to charge a substantial fee for high-performance visualization software. Additionally, vendors of commercial visualization software are faced with the inordinate task of porting and supporting their software products on an ever-changing array of computer hardware and software. Given the small market, the explicit conundrum between the commercial need to charge substantial fees and the research need to minimize expenditures, and the difficulty of maintaining a commercial product on a wide variety of platforms, commercial support for high performance visualization software is simply unattractive to industry. The most successful "commercial" visualization operations are those that produce an open source product, that invite community involvement in development, and that receive funding for ongoing development that targets the current needs of the research community. However, best-effort support often adds burden to visualization projects that depend on open source projects.

Scientific visualization also places extreme demands on computing infrastructure. All aspects of the computing pipeline are subject to significant demands for multi-terabyte datasets: storage systems that serve as repositories; CPU and memory systems that process the data; networks that transport the data, and graphics systems that display it. The same maladies that plague the general scientific computing hardware market are present in the high-performance graphics and visualization world: the needs of the scientific visualization community are largely ignored by graphics hardware manufacturers. Those vendors are primarily drive by the needs of the computer gaming industry, which uses benchmarks that measure the number of frames per second generated when playing one of several different computer games. These ratings do not correlate to scientific visualization needs.

Given the central role of the remote visualization metaphor in modern scientific computing, there is an alarming lack of networking capacity to connect remote users with centralized facilities. Large-scale computer systems provide massive computational capacity but are often linked to the outside world via networks of inadequate capacity. Commodity Gigabit Ethernet hardware for desktop platforms is very inexpensive, yet the networks connecting major sites typically can support only two and a half such users operating at full capacity. Beyond the trunk lines themselves is the acute need for hardware that connects sites to the network. Effective use of centralized facilities requires high-speed network connectivity to deliver results to remotely located researchers. A difficult question is how much networking capacity is required? Like many of the questions raised throughout this document, the answer is multidimensional and highly dependent on how the technology is to be used.

In one view, the purpose of network backbones in a Grid computing environment is to connect multiple, diverse resources so they all appear as one resource to the researcher. In this view, it is reasonable to say that the network should perform at a rate commensurate with the computational resources it connects. An approximate performance metric in this scenario calls for network performance that is in the range of tens of gigabits per second. Such networks are starting to come into existence, as evidenced by the National Science Foundation's TeraGrid. Not only are fast networks needed, but the computational science requires that these many networks—commercial and those sponsored by advanced

federal research and development—be interconnected. Researchers need access to their data and computational resources, regardless of their location. A single network may provide adequate performance among a small number of sites, but researchers are realistically more dispersed and may not be able to perform their work at one of the few sites endowed with adequate networking capacity. In other words, all federally funded networks should be "peered" so that a researcher at any federal research organization has outstanding network connectivity (OC-192 or 10Gb/s) to any other site. Funding streams from different organizations have inadvertently produced "islands" of network capacity.

When designing large-scale platforms, the needs of computational science research programs are taken into account by considering grid resolution, number of unknowns, number of time steps, and related variables to estimate the approximate amount of computing power required for a given class of algorithms. On the other hand, visualization processing is typically delegated to relatively small computing platforms that have nowhere near enough computing power. A disparity of several orders of magnitude in computing power is typical: simulations are run on platforms that can reach tens of teraflops, yet visualization is delegated to machines that are capable of only a few gigaflops. A substantial increase in funding for visualization computing platforms is critical to "impedance match" the capacity of simulation and analysis platforms. Similarly, an increase in visualization research staffing is needed to support projected growth trends to meet the needs of science research programs. In its early planning stages, the ASCI program carefully defined visualization metrics that would be required to meet user needs given projected levels of computing capacity. Other sites and programs should adopt similar guidelines for future purchases. Otherwise, we can find ourselves in a situation similar to the Earth Simulator when the machine had to be idled so that storage and data analysis tasks were given an opportunity to catch up.

Metrics of Success

Visualization success can be characterized by using several metrics. First and foremost is the degree to which visualization helps advance science as an enabling technology. The most obvious metric is the number of scientific discoveries facilitated by visualization. However, achieving these discoveries requires close coupling between visualization and scientific researchers so that visual data analysis tools are effectively designed and applied. Therefore, a practical programmatic objective would be to aim for an increase in the number of multidisciplinary teams where visualization is included. While such presence doesn't guarantee scientific discovery, it does create the potential for increased synergy as part of the scientific research process. Achieving such an increase of visualization in science can be implemented at the institutional level or at the individual project level.

Another metric is longevity, or the temporal lifetime of visualization technology. The current visualization funding model encourages exploration of ideas but does not provide for the critical ongoing maintenance and lifecycle support activities needed to ensure that today's research prototypes form the basis for tomorrow's staple software tools. Increasing the lifetime of visualization technology will have long-term payoff in the form of reducing duplication of effort between visualization efforts. It will also simplify use of

software tools since researchers will not be frequently required to surmount a steep learning curve associated with a new technology.

Still another metric is the degree to which visualization, analysis, and data management are interoperable. Future research programs in visualization must include interoperability as a central theme to promote both longevity and widespread use by a large population.

Computer Architecture for High-end Computing

The role of computer architecture is to define the mechanisms used to realize a computational task. In high-performance computing, the emphasis has traditionally been on floating-point computation. As both algorithms and architecture have evolved, however, operations other than just floating point have become important. Many algorithms are adaptive, applying the computational work where it is most needed. These algorithms need more than just floating-point operations; they make heavy use of other operations such as memory accesses and conditional branches. Thus, key features of the architecture include the memory system, interconnect between processors, and the I/O system to persistent storage such as disks. Even in single processors, the performance of many calculations is bounded by the performance of the memory rather than the performance of the CPU.

It is important to realize that the performance of the fastest computers is already limited by basic physics. A commodity CPU now operates at over 2 GHz. In a single clock cycle, light (in a vacuum) travels only about 15 cm. Signals travel more slowly in wires; a modest-sized cluster is now over one hundred cycles across. This fact makes it impossible to construct a single ultrafast processor that can access memory within a single clock cycle. Algorithms and programming models must be developed that take this fact into account.

Difficult engineering problems complicate the problem. While memory latency (the time that it takes to access a memory location) is a matter of physics, memory bandwidth (the rate at which data can be moved) is primarily a matter of engineering. Bandwidth can be provided to applications, but at a cost. Other issues, such as heat dissipation, further challenge computer architects.

Impact on Applications

Computer architecture strongly controls the effectiveness and availability of computational power for applications. Algorithms connect the mathematics of approximation and models with what the hardware can implement. Assumptions about relative speeds of operations strongly influence the choice of algorithm and implementation, in turn strongly affecting delivered performance. Many applications, for example, require memory bandwidth more than any other resource; doubling memory bandwidth may double achieved performance for those applications. Others require lower effective memory latency. Still others require faster parallel collective operations.

Research Issues

Four areas were identified where further research and development are required. These are memory performance (both bandwidth and latency), interconnect performance (both bandwidth and latency), I/O performance, and fault tolerance.

The reason for the emphasis on memory performance can be seen in

Figure 9. While raw CPU performance has increased rapidly over the past several decades, memory system performance has fallen far behind. In fact, for many applications, it is the memory system performance that controls the performance of the application, even (or especially) for parallel applications.

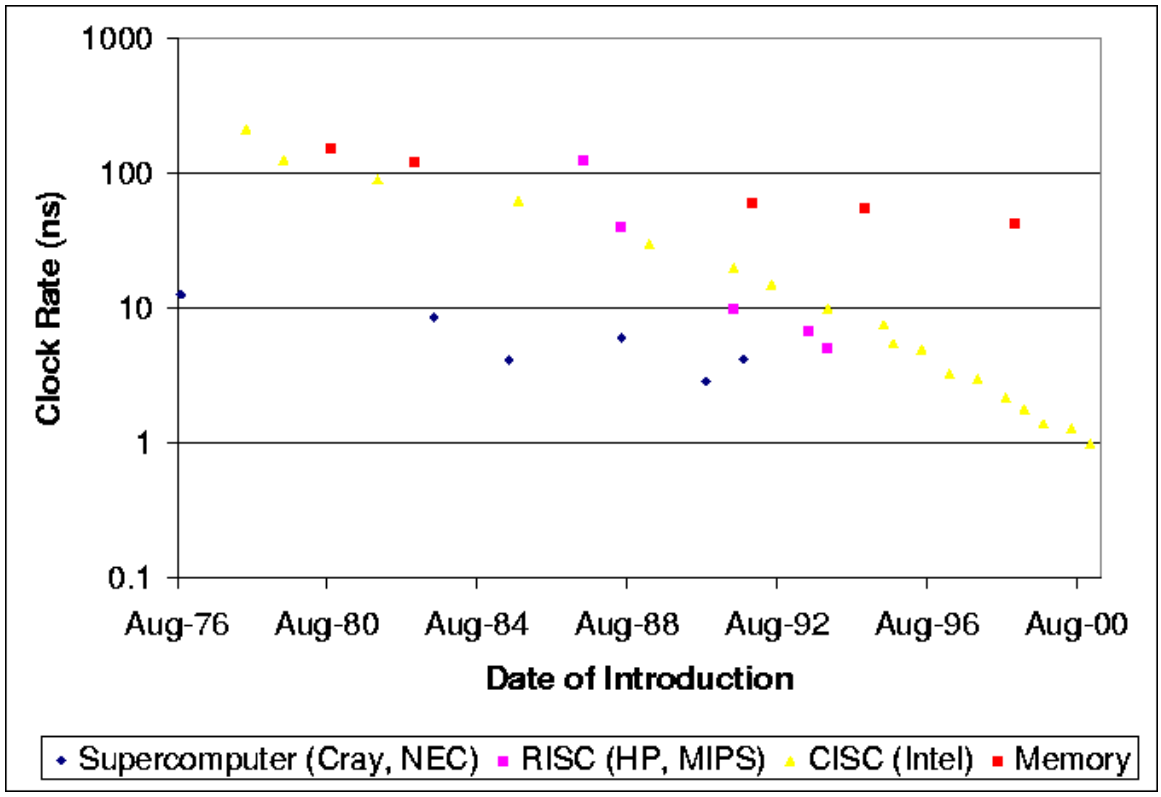


Figure 9: Improvement in CPU and memory performance over time.

These four areas of research need to be considered in the context of the three major directions in computer architecture research. Broadly speaking, in the context of high performance computing, architectures can be considered either

- Commodity — exploiting the mass market for computing, using processors designed for desktops or servers,
- Custom — optimizing the design for scientific workloads, typified by vector machines such as the Earth Simulator or the Cray X1
- Commodity Process — a middle ground that exploits the production processes used in commodity processors (including making use of basic processor and instruction set design), but with components designed for the scientific workload.

Each of these directions offers advantages and disadvantages in the design and production of an effective high-performance computer. All will benefit from research into the four areas mentioned above.

External Dependencies

Achieving high performance from the computer hardware requires algorithms and implementation frameworks (compilers, languages, and middleware) that make good use of the resources provided by the hardware. Rapid advancements in CPU speeds put increasing emphasis on algorithms that can tolerate long memory access delays (latency-tolerant algorithms), that make good use of memory locality (cache-friendly algorithms), or that can take advantage of special hardware capabilities (such as vector or stream instructions, or processor-in-memory). Neither the architecture nor the algorithms can be developed independently. This interaction has become increasingly important with the development of modern, optimal algorithms, such as sparse-matrix linear system solvers and adaptive mesh refinement techniques, which place great demands on the memory system.

An equally important area of interaction is between the architecture and the system software, and particularly with the programming models supported on the hardware. There is a strong consensus that current programming models do not adequately support the computational scientist. What should replace or augment the current models is quite controversial, but architectural support is likely to be required for any approach that offers a significant improvement in productivity. In addition, there is no consensus on what features should be part of an ideal computer architecture; different algorithms place different demands on both the programming models and the computer architecture. Understanding how the needs of a wide-variety of applications are reflected in the choice of computer architecture, particularly as new algorithms are developed, requires a balanced approach involving many areas of computer science and mathematics.

Delivering Effective Computer Architectures to the Applications

Current computer architectures provide significant computing power to applications. A number of barriers remain, however, to making more effective use of these machines. One concern raised by many applications is the need for better support of ease-of-use features in programming applications and in the system software. While enhancements to the computer architecture can address these issues (e.g., by providing special support for a global address space), applications developers must have confidence—before they can commit to developing code—that systems that provide these new features will be available and supported. This situation argues for a goal of ubiquitous support of any new programming model, either through hardware or through sophisticated software. In addition, effective community-wide access to experimental architectures will help stimulate experimentation and help create increase community understanding of the needs of applications and the solutions computer architects can offer.

Also needed is high-quality, fine-grained performance data. Applications often cannot determine the reason for performance problems. Access to good data would increase our understanding of performance issues and help develop needed improvements.

Resources Required

The ultrascale machines of the future will be massively parallel machines, utilizing at least 10,000 processors (and perhaps far more). The message-passing programming model, and in particular the MPI model, will continue to be an important programming paradigm, both because many applications already exist that use this model and because it has proven effective (many applications use MPI on a wide range of platforms, from laptops to the world's fastest computer, the Japanese Earth Simulator). To ensure that applications are ready for a machine of this scale, further research is required to address issues in the following areas:

- Algorithmic complexity of applications interacting with architectures
- Uncertainty of scalability of some applications
- Reliability and fault-tolerance
- Memory hierarchy
- Processor and node interconnect
- Parallel I/O

One area requiring greater study is that of application scalability properties. Because of the complexity of current systems, it is often difficult to accurately predict the scalability of applications without performing experiments to validate the scalability model. Performing these tests, and developing new algorithms and solution techniques to the problems that will be uncovered, requires a testbed capable of simulating over 10,000 processing elements.

Because of the severe physical constraints, it is important to support research into innovative architectural concepts. New ideas are needed to deal with the trends shown in

Figure 9. To ensure that this work addresses the real needs of applications, support is needed for ``cross-cultural" studies, involving teams of researchers from the areas of algorithms, applications, and architecture. It is vital that these studies be carried out with representative applications, not small example programs. Many scientific applications involve hundreds of thousands of lines of code, and there is a real danger that small example programs will not include key features of real applications. This does not mean that smaller, representative codes cannot be developed to help computer architects design computers more effective for scientific applications. Rather, it means that the process of developing such examples requires the involvement of a cross-disciplinary team with the breadth of expertise necessary to identify the important features.

As parallel systems grow larger, there is an increasing need for cost-effective interconnects that scale to large numbers of processors. Further, these interconnects must maintain low latency and high bandwidth. After a long period during which the performance of commodity interconnects improved slowly, the commodity market has placed an emphasis on improving bandwidth. Interconnects technologies such as Infiniband offer significantly greater bandwidth than was available just a few years ago. Reducing latency is more challenging and will require both careful engineering of the interconnect as well as matching the hardware operations to the needs of the software (the

mismatch is a significant source of extra latency). In addition, the operations provided by these networks have not been designed to fit the needs of scalable parallel programming models, particularly in the case of low-latency operations.

Support is also needed for more sophisticated techniques to address the issues of memory latency and bandwidth. For example, hardware support for memory operations such as scatter, gather, and scatter-add can help the hardware cooperate with the algorithm developer in limiting the impact of memory latency on application performance. Other techniques for hiding memory and interconnect latency should be pursued.

There is renewed interest in the need to support computer architectures for high performance computing. The DARPA HPCS (High Productivity Computing Systems) is also looking at ways to improve productivity in computing through a combination of advances, including ones in computer architecture. Many of the issues and recommendations in this Chapter overlap with those discussed at the HECRTF (High End Computing Revitalization Task Force) workshop. In addition, Office of Science applications can play a key role in guiding the development of architecture by providing a range of important computational science applications, spanning the range from mature, large scale applications using well-established and regular data structures to applications exploiting the newest techniques in adaptive algorithms and sparse matrices.

Metrics of Success

Because of the critical role of the applications, the most important metrics of success relate to ensuring that the targeted applications can make significant improvement in solving their problems and that the software is ready when a new computer system is delivered.

The working group unanimously found that “percentage of peak” is not a useful metric. This metric is often quoted because it has one desirable property of a metric—it is (relatively) easy to compute. However, it has many flaws; for example, it contains no measure of cost, and there is no relationship between effectiveness on a computation and percentage of peak performance. For example, an architecture that made extensive use of low-cost (but thanks to commodity CPU technology, relatively fast) floating-point units to achieve higher application performance would be penalized under a percentage-of-peak metric, even though it might provide the most cost-effective solution. Furthermore, a significant cost in today's supercomputers is the memory, and clearly percentage-of-peak does not measure the effectiveness of application's or architecture's use of memory.

The working group recommends the creation of integrated architecture-application development groups. These should coordinate with the algorithm and software communities to ensure that all groups are aware of the constraints and opportunities. In addition, it is critical that there be a persistence of investment in any architectural developments. The lifetimes of most of the application codes used by researchers in the Office of Science are measured in decades. Computers, on the other hand, are obsolete in three to five years. For an application group to commit to a new architecture requires a commitment to provide sustained support for that architecture. It also requires planning and supporting the development of system software (including compilers and

middleware) to ensure that all critical software is ready as soon as the hardware is available. Because of the relatively short lifetimes of hardware, the development of the system software cannot wait until the machine is available.

Programming Models and Component Technology for High-End Computing

One of the major limitations to broader use of HEC is the difficulty of programming the machines. With proposals to significantly increase the number of processors, and physical as well as practical reasons to build hierarchical machines, the problem is likely to remain. This section describes some of the research problems in addressing this issue through improved software technology. It could be subtitled "taming and harnessing software complexity."

Component frameworks and programming models

One can separate the HEC software challenge into two subproblems: the difficulty of writing parallel high performance software, and the difficulty of leveraging the software investments by others through code re-use.

A *programming model* is a programming language or a class of languages that supports programming in a particular style. For example, message passing and threaded shared memory programming are two models, each of which is realized in several different languages augmented with a library of parallelism primitives. A given programming model may simplify some programming tasks while making others more difficult. For example, in most shared-memory programming models it is difficult to control the layout of data structures in physical memory, which can affect performance. However, it is relatively easy to load balance computation using a shared queue of pointers to task descriptions. Conversely, in a message passing model data layout across processes is entirely under programmer control, but load balancing requires that a task description be explicitly packaged and sent to other processors.

A *component* is a stand-alone unit of software that can be composed with other components to create an application. This composition mechanism, or *component model*, dictates the software mechanisms to connect components together, but does not specify what the component does, numerically or scientifically. Component-based design can help to manage the complexity of high-performance simulations by combining object-oriented design with the powerful features of well-defined abstract interfaces, programming language interoperability, and dynamic composability. A scientist creating a component must adhere to some rules that govern the interoperability of his/her component with other components, and there may be some supporting software that will aid the component developer in this task.

A set of components that can be grouped together as peers is called a *peer component framework*, which we refer to simply as a *framework*. Frameworks in this sense encompass the concept of *framing* software modules or components in an environment that orchestrates component composition and supports their execution.

Given that each person can manage only a limited amount of complexity, either the complexity can be hidden by the programming model and its compiler and runtime, or the complexity can be partitioned among various scientists and their disciplines.

Programming models and components approach the complexity of high end computing in orthogonal and complimentary ways:

- Programming models seek to reduce the complexity of HEC by trimming the degree of detail the programmer must master to use exotic computers. Understanding the cache structure or the network interface is left to the computer scientists who design the programming models, leaving application programmers free to concentrate on their particular domain of science.
- Components partition complexity into individual units of software that are developed for use by others. These "black boxes" encapsulate the complexity that a small group of people can handle by themselves. Components export well-known interfaces that are easier to master than the underlying complexity inside the "black box."

Components can use programming models as the implementation of the code within the black box. Both reduce the workload on application scientists, allowing them to concentrate more effort on their own application domain.

Impact on Applications

In short, programming models partition the complexity of HEC between programmer and the "system" software/hardware. Component frameworks partition the complexity between one programmer and another. Programming models make exotic computers *usable* and components make large, integrated multi-physics simulations *possible*.

Better programming models for HEC offer the potential for greater programmer productivity, better performance, and increased portability and maintainability. Programming parallel machines is hard, and programming them well requires a good understanding of architectures, compilers, and numerical algorithms, and expertise in the application domain. Abstractions elevate the physical details of the machine to a higher level reducing the quantity of application code and simultaneously increasing portability.

The promise of HEC component frameworks is a common code base from which applications are built. Components, rather than stand-alone applications, will become output of developers. Applications, pieced together from this code base, would be used to construct HEC simulations, using the combined expertise of 10's or 100's of computational scientists. Applications that increasingly cohere with detailed natural phenomena they simulate will necessarily become more complex. Given that the capacity of individual investigators to support such complexity is finite, components enable increasingly comprehensive and accurate simulations.

Research Issues in HEC Programming Models

The challenge in designing programming models for HEC environments is the competing goals of providing high programmer productivity and maintaining high performance. Productivity includes both the initial programming task and the substantial work of maintenance and porting to new machines. The techniques that improve productivity typically raise the level of abstraction to hide machine details, thereby reducing the total amount of code and improving portability. The techniques to ensure high performance

conversely expose some of the underlying machine structure, giving programmers control over data placement, scheduling, and communication, making performance more transparent, and enabling significant user-level performance tuning. The emphasis on performance is much stronger in the scientific world than in the business world, so solutions unique to HEC programming are required.

Programming Model Overview

Parallel programming models can be roughly categorized based on two criteria: how parallelism is expressed, and how parallel tasks communicate with one another. The communication mode is orthogonal to the expression of parallelism, which falls into one of several categories:

- Implicit parallelism uses an automatic parallelizing or vectorizing compiler to discover parallelism in a serial program.
- Data-parallel languages express parallelism at the level of individual data elements, such as across the elements of an array. The compiler is responsible for mapping this fine-grained parallelism into a smaller number of physical processors.
- A single program multiple data (SPMD) model uses a static parallelism model with a fixed number of parallel processes determined at program startup time. The most popular programming model for HEC machines today is SPMD parallelism with message passing communication, specifically using MPI.

Barriers to HEC Programming Models

The design of programming models is challenged on the one hand by a desire to support increasingly sophisticated physical models and mathematical algorithms, and on the other hand by the scale and complexity of the underlying parallel machines. Some relevant research issues are:

- **Supporting irregularity in applications.** In an effort to solve problems with increasing spatial scales and resolutions, many applications are moving toward sparse and adaptive algorithms that place more data values and computation at the locations of most interest within a physical domain. The more efficient algorithms also tend to have a lower ratio of computation to communication, and hence lower data reuse. These algorithms use more frequent memory accesses and communication as well as noncontiguous data access patterns, all of which perform poorly on hierarchical machines that depend on spatial and temporal locality. From the programming model perspective these computations involve pointer-based data structures or arrays with indirect memory accesses, both of which are difficult for compilers to analyze and optimize. The programming model challenge is to ease programming by providing the right set of abstractions for irregular computations without sacrificing performance. These may be built-in data types such as sparse matrices or adaptive meshes, or data movement operations, such as scatter/gather or strided memory moves.
- **Abstractions for HEC machine complexity.** Driving the need for new programming models from architectural direction are machines with multiple

levels of parallelism and deep memory hierarchies, often with two of three levels of cache, local memory, and remote memory with orders of magnitude difference in cost and performance between the levels. These problems are likely to increase with petaflop scale systems, as hardware designers address the limits of automatically discovered instruction-level parallelism. To maximize performance on current machines, programmers sometimes resort to writing two different parallel programs, mixing a message-passing model at the outer level with a shared memory threads to run on a single node. These hybrid programs are currently an ad hoc mixture of models, not well supported by any single programming model in use today. The result is programs that are difficult to implement, tune, and maintain. Moreover, such programs often work, or work well, only on a specific machine configuration. The challenge in this case is to either automate enough of the performance tuning process that programmers need not be aware of the many levels of parallelism and memory, or to provide machine-independent abstractions for expressing the parallelism and locality properties of the application and implementations that efficiently map that information to future machines.

Strategy for Overcoming Barriers to HEC Programming Models

Here we review five of the most promising avenues of research: shared-memory abstractions, visualization of processors, hierarchical parallel languages, automatic performance tuning, and domain-specific languages.

Supporting shared-memory abstractions. In order to address the problems of irregular applications, a shared-memory abstraction may simplify programming and even gain performance. The traditional two-sided message passing model is reasonable for many regular computations that can be implemented in a bulk-synchronous style with equally balanced computation phases separated by similarly balanced communication phases. However, highly optimized message-passing programs, especially for more irregular problems, are often written with “preposted receives” and asynchronous message passing, which expose some of the underlying message buffer management to the application program. Many open research problems remain, starting with the identification of the right set of shared-memory primitives that are both expressive and implementable. Several problems also remain at the implementation level in terms of mapping these abstractions across a wide range of shared memory and networked hardware. For the language-based models, many open questions remain about the effectiveness of compile-time and runtime optimizations and the extent to which user-level performance tuning can be automated.

Virtualization of processors for portability and flexibility. Simulations that involve a fixed amount of work on each mesh point or array element are easily programmed by using an SPMD model, but search algorithms and highly adaptive problems many involve an unpredictable amount of parallelism. The SPMD model is easily mapped to machines in which there is a single level of coarse-grained parallelism, but it increases the programming burden for these more dynamic problems, adding load-balancing algorithms and task migration to the application. New models should be developed to

virtualize the set of processors, allowing programmers to express divide and-conquer-style parallelism. The challenge is to do so in a general way, while keeping the cost of parallel task creation low and giving either the programmer or the runtime system sufficient information to optimize for locality.

Multi-paradigm hierarchical parallel languages. The need for more expressive models of parallelism is also driven by the trend in architectures towards multiple levels of parallelism. To support portability across current and future architectures, which may have vector processing, SIMD processor extensions, multithreading, and clustering, the programming model should support a rich set of parallelism constructs. Programmers should be able to express the maximum parallelism available in the problem, all within a single programming model, and allow the model implementation to make use of whichever levels are appropriate for a given machine.

Incorporating automatic performance tuning. The complexity of modern architectures makes performance tuning difficult, especially within a single processing node where the programmer has little direct control over data placement within caches, virtual memory pages, and memory banks. Yet most algorithms are highly sensitive to the memory system structure and performance. Automatic tuning has been demonstrated for specific algorithmic kernels such as the Atlas system for dense matrix algorithms or FFTW for fast Fourier transforms. Next generation programming models should incorporate optimizations that can be performed automatically on such algorithmic kernels so that the cost of hand-tuning a code for each new machine can be avoided.

The emergence of domain-specific languages. A more ambitious line of research is to develop new high-level languages that abstract away even more of the execution details. Such a domain-specific programming model may be based on a combination of a new programming language and a set of libraries or tools for the domain. High performance would depend on the implementation and might take advantage of static compiler optimizations, runtime optimizations, and domain-specific optimizations. The implementation could take advantage of years of effort spent developing hand optimizations for such codes.

Research Issues in HEC Component Frameworks

Component technology has been successfully used in the business community, but the requirements for high performance and scalability in HEC applications make many of the component tools developed for business applications inappropriate for HEC applications. The HEC community needs its own component model that supports performance tuning and parallel computing, while still allowing for the software productivity and cost advantages that components provide. Developing a viable peer component model for the HEC domain remains an outstanding research problem.

Barriers to HEC Components

The goal of HEC components is to allow plug-and-play simulations using and reusing mathematics and computer science components along with application-specific interchangeable software. The means to achieving this interoperability is the adoption of

a common standard for all components in the program. Many sociological and technical issues impede this goal.

Viable component model for parallel computing. All existing component approaches look at parallel components as compositions of SPMD programs. In the CCA vernacular this is called SCMD: single component multiple data. It is not clear that this is the correct design pattern; but even if it proves to be so, it is abundantly clear that the SCMD paradigm by itself is insufficient. While most parallel applications are largely SPMD, some aspect almost always breaks this pattern. For example, in data collection and visualization, all the processors must communicate to a single node or file. For another example, in climate applications the entire simulation is based on a design pattern other than SPMD, one that is similar to multiple SPMD applications running simultaneously.

Support for mixed models. The intermixing of numerical components, some of which use MPI and others of which use shared memory, is not trivial. Such difficulties can be masked when considering only very high level component interfaces (such at the level of whole applications) but are serious issues when developing lower level interfaces.

High cost of participation. To some degree a standard must be imposed on software designed to be interoperable. For the promise of pluggable components in HEC to be realized, a standard for the interfaces and bindings for the "plugs" must be adopted. However, complex environments limit the application design possibilities. In addition, requiring an entire programming language or parallel environment to which all compatible applications must be (re)written puts a great onus on applications scientists—one that they will correctly resist.

A single framework. While having interoperable software is important, imposing a single framework on applications scientists is difficult. Ignoring the formidable sociological obstacles, this approach requires the developers of this single framework to effectively take care of the entire world. This makes for a single point of failure—just a few people on which all applications rely—an inherently nonscalable situation.

Strategy for Overcoming Barriers

Although programmatic organization can help with the sociological factors in the adoption of component concepts, most of these issues need computer science research to solve.

Design patterns for parallel component computing. Components in the high-performance arena must be by nature different from their counterparts in the commercial sector. Computer science research must be done to develop viable answers to the real needs of parallel componentized applications. The answer may be to generalize the SCMD pattern, or a separate paradigm may be developed that augments it. In either case the needs of barriers (3 and 4) above must be met with simplicity and ease of adoption.

Develop simple, powerful, and unconstraining HEC component models. The most important factor in user adoption of component models is the mental and software investment that its use requires. While some standard has to be put in place for the sake

of interoperability, the development and support of the most minimal standard that still accomplishes this goal are necessary to avoid situation 3 above. Simplicity, while still fulfilling its interoperability mission, is hard to achieve from the computer science perspective. It requires theoretical, practical, and intense application involvement on the part of the computer science researchers. Templates and a plethora of example components help lower the pain threshold, and these must be developed as educational materials for domain scientists whose first concern is not components and computer science.

Encourage multiple implementations to suit the application. Whatever the component system turns out to be, it must begin as a specification, without relying on any particular implementation. To avoid the resistance encountered in item 4 above, users must have the most flexibility in componentizing existing code, and they must be allowed to "do it their way" if they wish. Frameworks may be written as application-specific as deemed necessary and as generic as possible. There is no harm, and considerable benefit, to having more than one framework to do the job. The framework should be there to enable and encourage componentry in application domains; frameworks are not important, the scientific importance and value are in the components. So long as frameworks adhere to the specification, the components from one will work with components from the other.

External Dependencies

Component systems depend on what would be considered normal operating system facilities commonly available. Unfortunately, HEC operating systems often may lack many common services such as threads, sockets, or shared memory. For this reason most HEC component models (e.g., CCA and Cactus) do not rely on these to accomplish their work; but to the extent that they are available, they will make the user/programmer's life easier. Programming models are in much the same situation. Much can be done with threads and operating system support for everyday services that we have on desktop platforms, but absent those, the utility of programming models is still in place.

Resources Required

At least two computer science researchers, familiar with component concepts, per application area would be required. Given that multiple applications will be developed, a group effort among the computational science researchers is necessary to define and standardize component models and interfaces between the application areas. It is important that the twin goals of developing the applications *and* interoperability between all participants be balanced. That is, roughly equal time must be allowed for developing the computer science these standards require and for ensuring active hands-on involvement in applications so that they get used.

Programming models do not need to be as universally applied and are orthogonal to the interoperability afforded by the component work. These language/environments are a conscious choice of the developers of applications and could be made application specific. Here again computer scientists will work directly with applications scientists

using the programming model, but this collaboration can be done relatively independently of other programming models for other applications.

Metrics of Success

The only real metric of success is the willing participation of applications scientists in the programming environments, tools, and component frameworks that are developed as a result of this work. Just as important is the achievement of pluggable interoperability between the resulting applications, numerical methods components, and performance tools derived from the program. This metric is rather difficult to quantify and is realized only at the end of the program. There are, however, software engineering metrics that quantify lines of code versus time and other tools, and these are valuable for determining how useful a programming model/environment is in making life easier.

Harder to quantify is the degree to which, for example, a particular implementation of a component framework promotes interoperability. A certain amount of forensic proof can be obtained after the fact by counting the number of times components are reused (that is, code did not have to be rewritten) in the various applications.

Because nature is complex, HEC applications must reflect that complexity. Since human beings are able to individually deal with only a finite amount of complexity, we will need to harness tens or hundreds of researchers on the same application code. The best way currently known to accomplish this task is the use of components and programming models. Because the machines and code are exotic, the components and programming models approach will necessarily be different from anything seen in the commercial sector.

The future vision is one where all scientific applications are assembled from components, and each component is created by a domain expert. These components are available to whatever need is pressing at the moment. In this sense the national component base will reflect the national science knowledge base as it is now. Knowledge is drawn from it, ramified, and synthesized and then contributed back to that same science base. Similarly, components reflect and encapsulate the expertise and insight of domain scientists and successively more complex and realistic applications will be drawn from them. These, in turn, will be contributed back to the component base as new components. This will take a significant investment in computer science to accomplish, but the potential result is nothing less than reshaping the way computational simulations are done in this nation.

Access and Resource Sharing

An ultrascale computer is a unique tool for large-scale science and engineering that enables scientists and engineers to conduct numerical simulations of unprecedented fidelity, detect patterns within enormous quantities of data, and use scientific apparatus in entirely new ways. This large-scale science is, to an ever greater extent, distributed and multidisciplinary, for the simple reason that the complex, multi-component systems that it seeks to understand—for example, the many biogeoterrestrial subsystems that drive and influence climate, or the multiple components that constitute a modern accelerator—engage specialized scientific groups that build their own models and collect and refine their own data to drive those simulations. In other words, ultrascale computers are not large personal computers but are national resources that must be knit into the national-scale (and frequently international-scale) integrated science environment of physical resources, collaborating scientists, and information flows that underlies the practice of large-scale science.

For these reasons, this report on ultrascale computing includes a section on access and resource sharing. In this section, we address the enabling technologies that must be developed and deployed to maximize the utility of an ultrascale computer through integration with the national-scale science environment that is typical of large-scale, multidisciplinary science. We explain how appropriate support for remote access to, and sharing of, an ultrascale facility can both enhance significantly the utility of that facility and enable new approaches to scientific investigation that are not possible today. We identify both immediate and long-term requirements for investment in research, development, and deployment in two key areas, namely, physical network infrastructure and the middleware software that enables secure, reliable, high-performance remote access to and sharing of ultrascale resources.

In developing this material, we have drawn heavily on the results of the DOE Office of Science High Performance Network Planning Workshop, August 13 to 15, 2002, which reviewed network infrastructure and technology requirements for major DOE application domains.

Impact on Applications

We structure our discussion of science requirements for remote access and sharing technologies in terms of six major classes of usage scenario: remote access to computation and data; distributed data integration; the coupling of simulation and experiment; the execution of multidisciplinary simulations; the orchestration of multidisciplinary, multiscale, end-to-end science process; and collaboration in support of all these activities.

Each class represents a significant opportunity for large-scale science within several of the science disciplines considered within this report. Each also poses challenging requirements for physical and software infrastructure.

Remote Access

Our first usage scenario concerns access to ultrascale computing facility resources from the scientist's desktop. This requirement arises in essentially every driving application domain (e.g., see astrophysics, combustion, climate), for the simple reason that the specialized nature of ultrascale computers makes them vital resources for large number of users, many of whom must necessarily be located remotely. Thus, technologies are needed that allow remote access to computational codes (e.g., from "portals" or "problem solving environments') and to data produced by those codes.

Key requirements include connecting high data-rate ultrascale systems to low bandwidth desktop systems; achieving remote and interactive exploration and visualization of petascale data; and computational steering for human interaction with running simulations.

Data Integration

Many fields of modern science, such as genomics, cosmology, and environment, have been transformed over the past 20 years because of the volume and quality of data available from modern instrument systems. Scientific progress depends increasingly on the scientist's ability to integrate data from multiple instruments and archives maintained by different discipline specialists. For example, in climate science, we have observational data collected by Earth observing satellites, simulation data produced by large numerical models, and many regional archives of biological and geophysical data; an understanding of the processes of global change can require the integration of data from these different sources so that, for example, historical sea surface temperature and land-use data can be compared with global change simulation scenarios. Astrophysicists must integrate digital sky surveys in many different wavelengths in order to study signals of the very beginnings of the universe—the cosmic microwave background radiation; and searches for the type of supernova that are used to probe the cosmological history of the universe involve large-scale and numerical simulations during the discovery process.

Data integration becomes increasingly important (and challenging) in the ultra-scale context for two reasons. First, large-scale numerical simulations can produce datasets of unprecedented size, fidelity, and value. Second, large-scale multidisciplinary models require multidisciplinary data from many archives.

Key requirements include bandwidth between archives and computers, the federated management of diverse archives, and consistent cataloging and metadata management.

Coupling Simulation and Experiment

A less common usage modality for large-scale computation, but nevertheless a vitally important one for some, concerns integrating simulations with the operation of an instrument that is involved in an experiment that is the physical manifestation of some aspects of the simulations. This requirement arises, for example, within the context of online model-driven control of magnetic fusion reactors and future accelerators and for instrument tuning in ultra-high resolution electron microscopy.

This usage modality becomes increasingly significant in the ultrascale context because, with the advent of extremely large computers we are able to run simulations fast enough to allow for model-driven control or real time interaction with microscopes.

Key requirements include network quality of service to ensure adequate communication between simulation and instrument, and computational scheduling/resource reservation so that the simulation can run on an ultrascale computer at the same time as the experiment is operating.

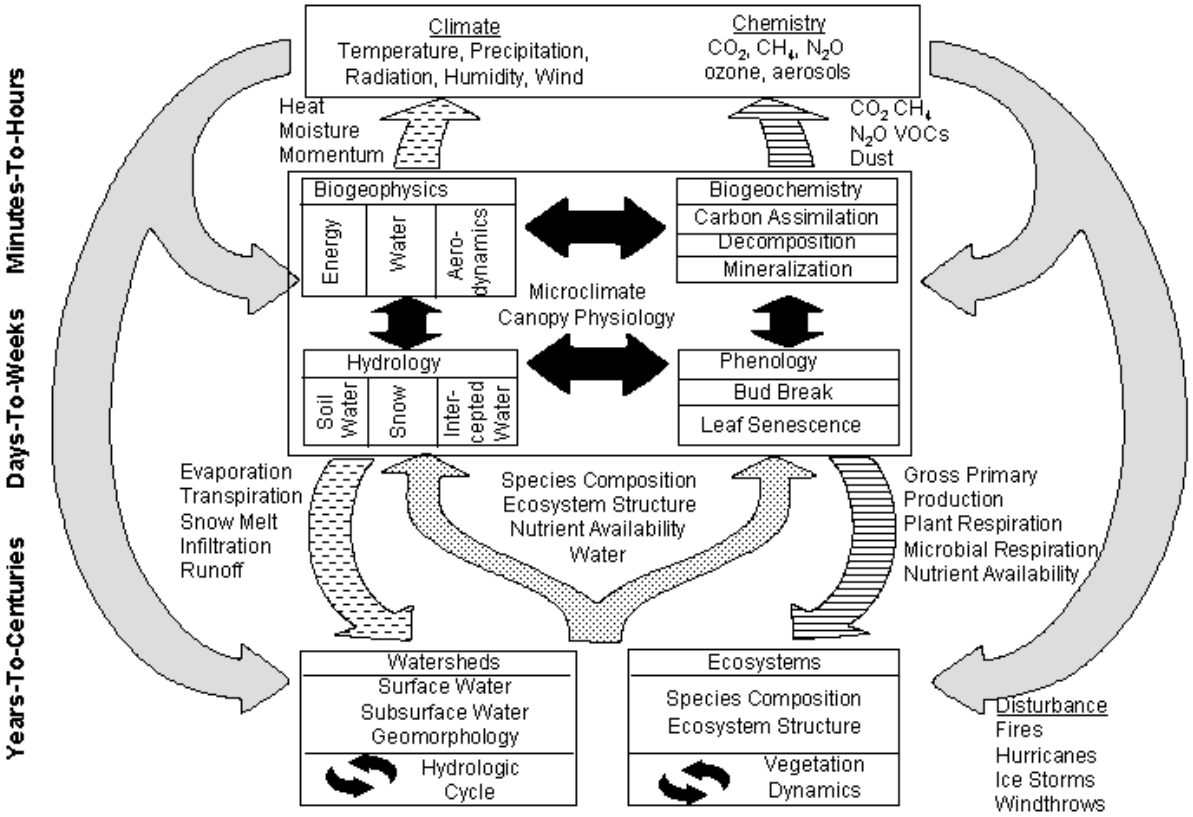


Figure 1. The Complexity of a "Complete" Approach to Climate Modeling – Involves the Many Interacting Processes and Data of Terrestrial Biogeoscience

Multidisciplinary Simulation

The fourth usage modality is also somewhat specialized, although again vitally important for certain groups. It involves large-scale multidisciplinary simulation, where a whole

system is simulated by the interaction of many subsystems—as is the case in nature. Typically, different components are developed by independent groups in specialized environments, and thus the whole system simulation involves integrating the subsystem simulations over networks.

This requirement arises in most disciplines in which macroscopic modeling is the goal, for example, climate, environment, and astrophysics. It is particularly significant in the ultrascale context because the high-fidelity subsystem simulations that are possible on ultrascale computers motivate multidisciplinary simulations, and it is not unusual that the computational requiremets of a few subsystems dominate.

Key requirements include bandwidth reservation between computers, co-scheduling of computers and other resources, standardized data exchange formats and protocols, and software environments and tools such as XCAT (distributed version of the DOE Common Component Architecture) as an Open Grid Services Architecture (OGSA) service that facilitates coupling distributed subsystem simulations.

Orchestration of Science Process

Our fifth usage modality is a broad and important one and concerns the orchestration of complex science processes. The complex systems that are often addressed within large-scale science typically involve complex workflows comprising distinct steps for such purposes as data access, data reduction, data integration, simulation, data analysis, steering, monitoring, and troubleshooting. The effective operation of such workflows requires tools that can allow their use by nonspecialists.

This requirement arises in many disciplines, such as accelerator modeling, climate, and environmental science. It becomes increasingly significant in the ultra- context because ultra-scale computers enable very large-scale science, which involves complex systems.

Key requirements include integration of ultrascale resources with the larger science environment and knowledge and workflow management tools for component orchestration.

Collaboration

Our sixth usage modality introduces issues relating to how distributed multidisciplinary teams function as they collaborate on design, execution, and analysis of ultrascale simulations and related activities.

The need for collaboration is a broadly crosscutting issue and becomes particularly important in the ultrascale context precisely because the complex science enabled by ultrascale computing is frequently multidisciplinary and thus highly collaborative.

Key requirements include security, access control, authorization, resource discovery, and resource management within dynamic virtual organizations. Equally important are collaboration tools and infrastructure that integrate computing, data, and instrument resources with human environments and knit together the human aspects of large-scale science.

Research Issues

Much work is required in order to incorporate ultrascale computers into the large-scale science environments that we discuss here. We use examples to illustrate the range of technology challenges.

Portals as the scientist's workbench. Componentized portal toolkits should provide for the dynamic assembly of interfaces to Grid Web services and applications, adaptation (e.g. to type of display and available bandwidth), preservation of the state of a scientist's workbench, and on-the-fly sharing of graphical and other simulation output with colleagues. Further, all of this should be available from Web clients wherever the scientist is located. Some promising work is coming out of projects like the EU's GridSphere and NSF's NMI Open Grid Computing Environment, but a great deal remains to be done.

Generalized metadata management of federating data archives and incorporating them into componentized simulation systems. The advent of XML and its enhancements in RDF, etc., have to potential of integrating data into the Grid Web Services environment of the Open Grid Services Architecture (OGSA). However work is required in the description of the semantics of data so that various XML schemas may be dynamically combined to federate related, but independent data archives that may be needed in concert to support large-scale simulation.

Scheduling. Many of the application scenarios introduced above require the co-scheduling of multiple resources, including people, networks, storage systems, and computers. These resources require reservation support, and scheduling mechanisms are required for organizing co-scheduling.

Event distribution and management. When looking across the space of application uses and system needs for events to facilitate everything from instrument state changes, to workflow synchronization, to application triggers of interesting features in data to faults, the small number of current event models must be greatly expanded and made easily accessible for many different uses.

Tools and approached for componentized systems. Componentizing software as in CCA/XCAT has the potential to realize a vision of flexibly reusing software. While prototypes like XCAT are proving useful, a lot of work remains to be done in accommodating the high speed, binary data flows of science applications, describing workflows and processing those descriptions in Grid environments, handling dynamic / data driven workflow, etc.

Knowledge management. A non-specialist should be able to formulate quantitative or qualitative, declarative or constraint based queries in problem solving environments that involve multiple related data and simulations operating in several discipline models. Semantic models and tools should generate correctly structured sets of operations – sequencing and parameterizations – and also manage acquiring or generating appropriate data that is input to the analysis and simulations that will resolve the query. This should be possible across multiple domain models, e.g. topography, hydrology, and climate illustrated in the Terrestrial Biogeoscience example above. The general data and

simulation services structure must be automatically mapped onto appropriate compute and data resources using Grid resource brokering and planning services – appropriate being determined by effective use, services that are located on specific / unique resources. All of this involves integrating and extending the integration of AI techniques and tools with Grid Web services technology to produce a Semantic Grid.

Authorization and access control for systems and groups. High value systems must be protected and access rules enforced, but at the same time the authorization process should not impede science access. In collaboration, the same thing applies, but now you have the added complexity of defining and managing access control for groups of participants that change both deliberately and through various system and communication faults. This also must be provided in a transparent way, techniques for which are just now emerging.

Security and intrusion detection. Remote access to ultrascale resources demands significant improvements in security technologies to enable robust security and intrusion detection commensurate with the value of ultra-scale systems.

Sharing of petascale data. Access to petascale data by distributed communities requires significant progress both in metadata standards and in distributed data management technologies.

End-to-end performance. High, reliable, end-to-end performance demands monitoring, troubleshooting, and configuration, and self-healing (autonomic) systems.

External Dependencies

Physical and Services Infrastructure

The effective integration of an ultrascale computational facility with other elements of a national large-scale science infrastructure requires the deployment of effective network communication and data management infrastructure.

End-to-end high-bandwidth networks must link DOE laboratories with each other and with key university and other laboratory partners such as NCAR. The speed required for these networks must be studied carefully to balance cost vs. benefit, but astrophysicists and high energy physicists estimate that >100 Gb/sec bandwidth links are required for all wide area and local area networks, in order to support the movement of large quantities of data for remote access, data integration, and collaborative scenarios. High reliability is also vital, particularly when coupling simulation and experiment; this requirement implies a need for independent redundant paths. We must also develop the ability to schedule end-to-end paths between, for example, executing simulations that are coupled with operating experiments, and we must solve “last mile” issues relating to high-speed access to computers, storage, experiment facilities, visualization, and collaboration systems

Another related requirement is for data-intensive infrastructure “embedded in the network.” Applications and communities that depend on the ability to share large quantities of data can benefit significantly from computing and caching facilities distributed around the network that can then be used for data management purposes.

All of these capabilities rely on persistent and managed services, such as identity credential management for cyber authentication, directory and database servers, and network services for allocating and managing the use of premium service (e.g., reserved bandwidth).

Policy Challenges

The effective integration of ultrascale computing resources with other elements of a DOE-wide integrated science environment raises a number of difficult policy challenges.

Security policies can make it difficult to obtain high-speed access (indeed, any remote access) to DOE facilities. The need to protect DOE resources is real, but so is the need for information to flow efficiently and painlessly among participants in distributed scientific collaborations. Careful thought must be given to security policies, access and authentication policies, and new technologies that may allow science goals to be achieved without compromising on security. New approaches to firewalls are likely to be required.

A second important area in which policies need to evolve concerns allocations and scheduling policies. Usage modalities that involve interactive access and/or co-allocation of multiple resources along end-to-end paths have important implications for site scheduling policies. Uniform allocation policies across sites and resources are also important.

Resource Requirements

The resources needed depend on the goals, scope, and timing of an ultra-scale computing program. The major categories are as follows:

- Physical infrastructure
    - End-to-end networks
    - Embedded cache/compute
    - Instrument interfaces
- Middleware research and development
- Operations and engineering

Metrics of Success

We suggest specific metrics of success that focus on the pragmatic issues relating to the creation and operation of an integrated science environment that incorporates ultrascale computing resources. For example,

- Deployment of 100 Gb/sec network among a core set of DOE laboratories,
- Transparent use of collaboration technologies by the multi-disciplinary teams recommended in other parts of this document.
- Remote access to ultrascale resources by 1000 remote users,

Data sharing at 1 TB/day among DOE and collaborator institutions Interpersonal collaboration of 100 users/day via the Access Grid and related technologies.

Software Engineering and Management

Scientific application codes vary widely in terms of the number and geographical dispersion of their developers, the lifetime of the software, and the number and sophistication of their users. Codes may be developed by an individual, by small collocated teams, or by large teams spread among many sites. Some codes last only long enough to solve a specific problem and are then discarded; others persist for years, if not decades. Some programs are used only by their developers, while others have a worldwide base of users conducting scientific research using source code they have never seen. Figure 1 depicts the diverse nature of scientific software as a three-dimensional space with number of developers, number of users, and code lifetime serving as the three axes. Most scientific disciplines or application areas harbor at least some codes that are evolving in the most challenging direction—shown by the red vector in this figure—along all three of the axes: They are structured and managed to accept contributions from a broad developer community; they plan to be of use to a wide community of scientists in their respective fields; and they will be in use for a decade or more. In addition to engineering a code to meet these three challenges, these codes are expected to run efficiently on many different high-performance computer systems and last through several generations of computer architecture developments. Software engineering and management will have the greatest effect on codes in the scientific software space located closest to the circled region in Figure 1.

This trend is apparent in both Office of Science and NNSA application areas. Some areas (chemistry and climate, for example) are farther along in this evolution than others (biology, for example), but all scientific application areas are sensing the need for community-wide software. Satisfying this need will require modern approaches to software engineering and management.

The commercial sector, where developing complex software systems is a driving force, employs software engineering and management practices that are not routinely used in scientific research environments. These commercial practices must be evaluated in order to determine the degree to which they are applicable to the development of high-performance scientific application codes.

What Is Software Engineering and Management?

Software engineering and management consists of a collection of *practices* and a set of *tools*. Although much has been written about software engineering in general, the specific requirements and development environments of scientific software have received minimal attention. As application software for terascale and petascale machines evolves to serve the needs of a growing user community, it is likely to adopt some of the acknowledged best practices from commercial software, while also creating some of its own. Areas in which investment in known best practices can result in more science-effective and more cost-effective application software include

- collaborative development;
- code architecture;
- source code management;
- application configuration management;
- portability in space and time;
- testing, debugging, and repair; and
- distribution management and user support.

In each of these areas, software exists that can help incorporate good practices into the software development and management process. Some of these tools are mature and widely used, while others remain research and development projects in their own right. Some commercial tools provide cost-effective solutions to specific problems; freely available open-source tools are useful in many areas but do not cover the entire spectrum. Application development efforts can be improved by awareness and utilization of helpful tools in the following areas:

- Source code management – CVS, Bitkeeper, Source Forge, and so forth
- Configuration management and portability – autoconf, other GNU tools
- Development and debugging tools – TotalView, Ensure++, and so forth
- Performance tools – PAPI, HPCView, and so forth
- Testing harnesses – Many project-specific tools that could be shared

Impact on Applications

Application barriers typically involve managing code complexity, the desire to flexibly incorporate new algorithms and physics over the lifetime of the code, and portability across computer architectures from different vendors as well as from one generation to the next of architectures from the same vendor, all without sacrificing performance. The use of software engineering and management practices and tools can benefit scientific applications by

- shortening development time by allowing for collaborative development and software reuse,
- enabling more science through incorporation of new algorithms and physics modules,
- lengthening software lifetimes,
- insulating application scientists from architecture and vendor specifics,
- leveraging community contributions, and
- widening the user base.

Research Issues

There are two types of research issues in software engineering and management. The first is the identification of which *known* practices and tools are the most appropriate for adoption by the scientific application community. This community has both requirements

and development environments different from those of either the commercial world or the systems software world, where many of the practices and tools have been developed. The second is the development of *new* practices and tools unique to scientific applications.

The best approach to initiating such research is the *adoption* of existing, known software engineering and management practices and tools early in the development process of scientific application codes. Some application areas are already far along this path and are even in a position to contribute their experience and some of their application-specific software for adoption by other application groups. Other areas are still in early stages, with software characterized as local, or even personal, as opposed to "community."

External Dependencies

Software engineering and management for scientific applications is expected to develop some of its own tools but will remain dependent on other communities, both commercial and otherwise, for software research and tools to support code development. At least three dependencies can be easily identified:

On open source community projects — A number of basic tools have been developed by the open source community and are in wide use. Despite gaps and imperfections, these tools are worthy of consideration. Examples include CVS and BitKeeper for source code management and Gnu `autoconf` for enabling portable software configurations scripts.

On commercial packages — Certain code development aids are so useful to the broad software development community that a commercial market exists for them. Examples are portable parallel debuggers and memory usage trackers. Despite their costs, such tools can be cost-effective.

On other research software — In recent years robust software has been developed by the scientific software community itself. This is particularly true of parallel mathematical libraries such as ScaLAPACK and PETSc and performance measurement and analysis software such as PAPI, Jumpshot, and TAO.

Delivering Software Engineering and Management to Applications

The most important mechanism for delivering software engineering and management to applications is the incorporation into scientific application development projects of people trained in the best practices and familiar with the best tools. Existing, successful medium- and large-scale projects have as a primary member of the development team a senior software engineer, perhaps supported by a small staff, who has responsibility for the application code *as software*, independent of the science incorporated into it. The senior software engineer is responsible not for writing application code but for managing the software development process, including designing the overall structure of the code for modularity and portability. This person is familiar with current software engineering practices and tools and can incorporate software engineering into the development plan

from the beginning of the project, even at the proposal stage. Documentation (both for contributors and users) needs to be planned (and budgeted for) from the beginning.

Resources Required

Simulation is one of the three fundamental pillars for research; accordingly, we must ensure that scientific software development efforts are funded to include appropriate levels of software engineering so that these efforts can meet the same standards of practice as do design and planning of laboratory instrumentation and field or lab experiments. Furthermore, these projects require ongoing support commensurate with the initial investment made during development and at an appropriate level for its location in the three-dimensional space of scientific software shown in Figure 1.

The first category of resources needed is for the software engineers to be incorporated into application projects as fully funded team members, senior participants in the development of the scientific applications codes of the future. These resources are additional resources needed in application development projects. The exact ratio of software engineers to the size of the project is application specific, but experience has shown that as few as from one to four software engineers per medium-sized project can have an enormous effect. The software engineering literature provides a number of more detailed quantitative studies.

The next category of resources needed is for research and development of software engineering tools that currently do not exist but that will be required by future applications. A program in software engineering for scientific applications at a level that could support three or four group efforts in their areas, selected by competitive proposals, would be likely to produce tools of broad applicability to scientific applications. Funding should be made available for workshops specifically targeted to software engineering and management for science would be a cost-effective way to foster the growth and visibility of both best practices and tools in the scientific application community.

The final category of resources needed is for the maintenance and evolution of the simulation software. This includes producing documentation for both developers and users, distributing software, systematic testing, and tracking and resolving bugs. This category of support is often neglected or overlooked. It is typically easier to garner funds for the next new project than to fund maintenance of existing codes.

Metrics of Success

Two families of metrics suggest themselves. The first is simply the quality of the scientific codes that are produced and of the code development processes that produce them. Code adhering to best software engineering and management practices and developed with the best tools should

- be easy to modify by incorporating new mathematical algorithms and new physics modules,

- be portable across current computer architectures and to next-generation architectures,
- leverage the work of others by incorporating common components,
- be systematically and continuously tested,
- provide a mechanism by which bugs are rapidly identified and repaired,
- have predictable performance on a wide range of architectures,
- be easy for new users to adopt in order to pursue new scientific results in the application area, and
- be maintained over its lifetime for improved capability, portability, and performance in a cost-effective way.

The second metric is the level of activity within the DOE scientific community devoted to software engineering and management issues. Such activity would include sharing of information on best practices and tools and development of specialized tools for software engineering in the scientific domain. Included also would be the expectation by program managers, reviewers, and proposal writers that software engineering and management are critical components of any large scientific application software effort.

Data Management and Analysis: Keeping Ahead of the Data Avalanche

Consider a typical scientific exploration process that involves large-scale simulations. It usually has three phases: simulation runs, postprocessing, and analysis. The *simulation* phase consists of intensive computations that generate large quantities of data, currently in the order of terabytes per simulation, and expected to grow to petabytes. The data must be saved as quickly as it is generated so that the computation is not slowed by I/O bottlenecks. In some cases, the simulation can benefit from dynamic steering by quickly analyzing intermediate results. A subsequent phase usually requires the *postprocessing* of the simulation data. This may include summarization of the data or transformation of the data from one format to another. In this phase, a large volume of data has to be read efficiently, and a large volume of data may be generated as well. In the *analysis* phase, relevant subsets of the data need to be selected based on properties of the data, and the subsets efficiently extracted. The analysis phase may require methods that discover specific patterns and relationships in the data.

All of these phases involve data management issues, including efficient writing and reading of data from disk caches and archives, selecting subsets of the data, moving data between computer systems, data reformatting, and keeping track of the conditions under which the data was generated. The technology that provides this capability in a well-organized and efficient manner is referred to as *scientific data management and analysis.* The goal of this technology is to minimize the amount of time scientists spend in managing their data, as well as improving the efficiency of the computational systems.

Impact on Applications

The main impact of data management stems from the increasing volume of the data as the computational power grows, since the volume of the data is expected to grow *linearly* with the computational speed. As an example of a typical application area, let us examine the field of computational astrophysics. Applications range in scale from Fermi acceleration of protons at shock fronts to gravitational clustering of galaxies. One specific application, the Terascale Supernova Initiative (TSI), aims to understand the mechanism driving core-collapse supernovae. The current focus of TSI is modeling the hydrodynamics (only) of the first second after core collapse using 3D simulations. The typical mode of operation is to run one large simulation that generates an order of 1 TB of data and then laboriously postprocess, analyze, and visualize this mountain of data. Running the model is easy; much effort has gone into scaling simulation codes for large numbers of processors. The data management, analysis, and visualization are the stumbling blocks. The usual methods and software do not scale well with the size of the data.

Currently, about five runs per year are generated, each run generating about 1 TB. In addition, there are about 20 postprocessing runs per year, each generating about 1 TB, for a total of 20 TB/year. This number is expected to double by next year. A hundredfold to thousandfold increase in the volume of data will amount to 2–20 PB. This huge increase in data volume requires extremely efficient and innovative data management techniques. Another example of expected growth of data volume is climate modeling. Currently, a so-called T42 CCSM (resolution 280 km) generates 7.5GB/yr. Thus, a typical simulation run of 100 years generates .75 TB. With the availability of terascale computing, the future generation runs, called T170 CCSM, will be four times as accurate per dimension (70 km) and will generate 110 GB/yr, or 11 TB/100 year simulation. Running hundreds of simulations per year or increasing the resolution further will bring the total volume into several petabytes per year.

Six main data management barriers were identified that imply the need for data management technologies to support them.

1. Scalable I/O (fast writing of data to disk during a simulation run, and fast reading during analysis). During simulation runs, applications may be generating checkpoint data and other intermediate files that must be written to disk for the purpose of restart/recovery and/or other usage. Speed of writing is often critical in these cases; typically, these files are written to a local I/O system and are often overwritten by newer checkpoint files.

2. Scalable archiving. Archival data is the output of a simulation run, postprocessing, or analysis that requires long-term storage. Archival data is used for repeated analysis or comparisons with the observed data and for possible sharing among researchers. The efficient access of subsets of data for the analysis is critical. This requires scalable indexing methods as well as fast reading from relatively slow media such as tape robots.

3. Metadata generation and management. Running multiple simulations (many runs) requires precise metadata to be collected in order to be able to find the relevant data for subsequent analysis. Similarly, data generated by postprocessing or analysis activities needs to have metadata that capture the nature of processing and the lineage (provenance) of the data. Tools for automating metadata generation and management can facilitate the searching and retrieving of relevant data.

4. Data searching (indexing, filtering, and retrieving). The archival data generated by the different phases of the simulation may have a complex hierarchical structure (objects) and in some cases multiple attributes of different data types (Boolean, numeric, string, etc.). Researchers are often interested in retrieving small subsets of the data based on some conditions on the attributes such as numeric ranges, Boolean values, or string matching. Fast location and reading of such subsets may require reorganization of the data, vertical partitioning (storing values for different attributes in separate files), construction of indexes, and other data-searching methods.

5. Data migration (between distant sites). In many cases, simulation data must be migrated to one or more machines different from the generating machine for further postprocessing and analysis. If data has to be moved out of the facility where it is generated, the rate of migration from the large simulation machine must match the rate of generation of the data, on the order of gigabytes per second. If the data is stored at the simulation site, we still need to migrate subsets of the data for analysis. Migrating a terabyte of data in a reasonable time (1000 seconds) requires a sustained gigabyte per second of reading, moving over the network, and writing the data. Such migration is common in large-scale collaborations where data is also replicated across multiple sites.

6. Data mining and data reduction. Data mining refers to a set of technologies that search the data for some specific patterns and relationships and/or try to discover hidden facts and infer rules from the data. Data reduction refers to techniques that produce smaller data sets from the original data while still retaining all or most of the useful information for further analysis. These techniques may remove redundant or irrelevant data; reduce the dimensionality of the data; or perform cluster analysis, grouping, summing, averaging and other operations on the data.

Research Issues

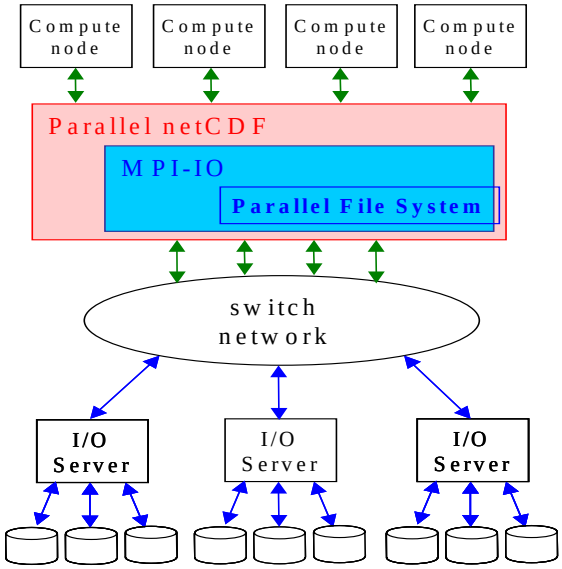
Terascale simulation codes are expected to produce large volumes of highly complex data. We describe here research problems in two main categories: I/O performance and data management and analysis. *Performance issues* refer to technologies for writing, reading, and moving large volumes of data in a cost-effective manner. Table 2 shows that the I/O bottleneck will dominate other computational bottlenecks (shown in red) in the coming years because performance improvements in I/O technologies lag behind that of the other components.

Table 2: Bottlenecks in data management

Balance vs. Bottlenecks				
Processors	Kiloflops	Megaflops	Gigaflops	Teraflops
I/O	Megabytes	Gigabytes	Terabytes	Petabytes
Network/sec	Kilobits	Megabits	Gigabits	Terabits
Memory Size	Kilobytes	Megabytes	Gigabytes	Terabytes
	1970's	1980's	1990's	2000's

Data management and analysis research deals with technologies for effectively describing, storing, and searching the output generated by the simulation program for post processing and analysis by scientific collaboration teams. Effective scientific data management increases the productivity of scientists by allowing fast identification of subsets of interest in the data.

Figure 10 Parallel Libraries and File Systems



An important research area is achieving fast parallel writes of both archival and checkpoint types of data to the local disks. Because the ability to read data efficiently depends on the layout of the data during the data generation phase, a related research area is to match the layout of the data to the expected usage patterns. Several research problems in this area are discussed below:

1. Collective I/O—Coordination among the processors is essential during the output phase of a simulation program in order to avoid multiple random writes of small amounts of data. Collective I/O provides a global picture of these individual I/O requests and merges them into larger contiguous blocks of data. Many parallel I/O libraries and file systems (see Figure 1) are being developed to provide effective implementations of collective I/O. This approach needs to be generalized and applied to various data formats being used by the scientific community.

2. **Overlapping I/O and computation**—Parallel computer architectures may have two types of processor nodes: compute nodes and dedicated I/O nodes. Other architectures allow each processor to perform both computation and I/O by allocating one or more I/O threads on each compute processor. More research and experimentation with multiple applications are needed to determine optimal allocation of compute and I/O power.

3. **Migration of archival data**—The goal is to overlap the I/O and the data migration operations with the computation, in order to avoid a situation where the simulation run needs to wait for the "evacuation" of data from the local disks. Also, the migration process should be automatic and transparent to the user. A promising approach, currently being pursued, is to build "migration engines" that are coupled with parallel I/O libraries. Effective tools for data migration to archival storage or over the network to other computational engines are essential for a smooth operation of terascale computing systems.

4. Data compression—The migration process can also benefit from high-performance data transfer protocols, multiple parallel client streams, and compressed data transfer. Because compression is compute intensive, algorithms must be developed that operate in parallel with the computation to avoid bottlenecks. If compression is used, decompression must be performed on the other end to allow processing of the data. Because of the large volume of data, more innovative approaches to data compression need to be pursued.

5. Fast parallel reads—Layout of archival data on parallel disks is important in order to support multiple access patterns during the post processing and the analysis phase of the simulation. The goal is to achieve load balancing among the disks by intelligent chunking, reorganization, and declustering of the data. Other important research issues involve caching, intelligent prefetching, and generation of "hints" for optimal data layouts. Reading data efficiently is also useful for analyzing data in a "streaming mode." Many analysis programs can process the data as it streams out of the computation engine. These techniques are especially useful for performing analysis in real-time applications, such as simulation or experiment dynamic steering.

We describe next several data management research issues dealing with the effective use, sharing, and registration of various types of data produced during large simulation runs.

Searching and indexing—Index structures of various types are widely used to organize and access large datasets. The choice of appropriate index structures depends on the characteristics of the data (dimensionality, sparseness, skewness, etc.) as well as the access patterns. Although much research has been performed on index structures, more research is needed in this area in topics such as parallel index structures, indexing of streaming data, and high-dimensional data.

Data reduction—Data reduction techniques allow one to decrease the volume of data stored and transmitted while maintaining sufficient accuracy. Data reduction technologies currently used or being researched include principal component analysis, wavelets, sampling, independent component analysis, histograms, and clustering. Because of the high volume of data, research in effective parallel data reduction techniques should continue.

Data transformation and conversion issues—Data transformation is an important and sometimes costly activity that allows reorganization or reshaping of the data between successive phases of the simulation. Many parallel algorithms require extensive data

reorganization even between successive steps of the algorithm. The complexity of the problem is exacerbated by the fact that the transformation sometimes must be done while the data is streaming continuously. Research is needed to study the data transformation requirements for various simulation applications and development of algorithms and software libraries that optimize data transformation and conversion.

Metadata—Metadata refers to descriptive data necessary to express the content, origin, and structure of the scientific data. For example, the conditions applied to the generation of a simulation dataset, the model used, the program used, and the machine where it was generated are all part of the dataset metadata. Another aspect of metadata is data provenance, that is, the lineage of the data production. This is important when the data generated is the result of postprocessing or analysis; in addition, scientific data is often replicated, and the origin of the data as well as data coherence must be maintained. Still another problem is integrating the metadata for datasets generated by multiple heterogeneous sources. As the volume of data increases, and as the number of scientists that share the data increases, tools must be available to facilitate the automatic collection and management of metadata.

External Dependencies

Data management and analysis depend on two key factors, one relating to the quantity of data and the other relating to the movement of data.

Data Quantity Issues. We assume that, for most if not all application areas, data quantity will grow linearly with the computing capability available. We believe it to be fairly likely that storage capacity will scale sufficiently; and indeed, for the twelve years, storage capacity has scaled faster than computational capability. However, history has shown that data transfer rates to disk and tape devices have not kept pace with computational capacity. While capacity scales with areal density (track bit density times track density), transfer rates scale with track bit density. Transfer rates also scale with the speed of the media relative to the read/write heads, but those speeds have not improved much; for instance, disk rotation rates have only quadrupled in over thirty years. Taking disk drives as elements of a storage system, and including cache memories, I/O ordering, RAID configurations, and the like helps but does not solve the fundamental problem. Figure 11 shows that storage system performance has increased by a factor of only 100 in the past thirty years, less than 20% annual compound growth rate (CGR). Tape devices and systems have done no better.

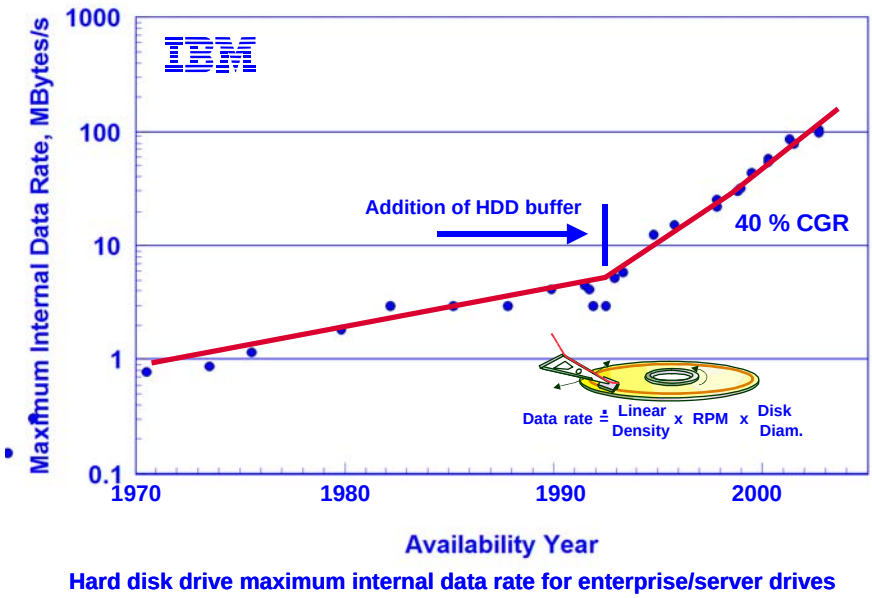


Figure 11: Improvements in storage system performance, 1970 to 2000

In order for transfer rates to keep pace, more and more parallel paths will be required, and hence more and more devices. Procurement, maintenance, and system administrator costs scale with the number of devices. So does the likelihood of media damage and device failure.

Moving Data between Computers. Wide area network (WAN) transfers of data cannot use the entire bandwidth available between end nodes. Effective throughput drops as distance and congestion increase; throughput of 1/15 to 1/10 of available raw network bandwidth is typical. And as the world continues to increase its network usage, congestion will also increase. Only if raw network bandwidth increases at least as fast as computational capability increases will WAN transfers be supportable. The need for network bandwidth is discussed in more detail in the chapter on Access.

Delivering Data Management Technology to Applications

The key to delivering data management technology is to simplify the tasks that the application scientist has to perform in order to take advantage of this technology. Three principles can simplify the process: abstraction of the functionality, interoperation of components, and automation of optimized data structures.

Abstraction of the Functionality. The functionality of a data management component should describe only what the component does, not how it does that. Another aspect of

functionality abstraction is that the components should be of general use, not tailored to a specific application. As a result of this approach, user interfaces and function invocation are simple to understand and use. This principle implies the separation of the logical data structures and functions from the physical data structures and indexes that implement the structures.

Interoperation of Components. *Consider a simulation that generates data in a particular format, such as HDF5 or NetCDF. If the next step requires that the dataset be indexed for an efficient search, then the index generator must be able to read (digest) that format. This task of making components interoperable cannot be left to each individual scientist to perform for a given application. The principle to adhere to is that components made available as part of the data management toolkit must interoperate or efficient data format translators must be provided.*

Automation of Optimized Data Structures. In the scientific domain as well as in the data management community, the way the data is stored and indexed determines to a large extent the ability to search and access the data efficiently. Thus, the challenge is one of matching the physical data storage layout and indexes to the anticipated access patterns. The most effective approach is to get access pattern "hints" from the user at the time the data are generated. These hints can be used to determine how to store the data, such as the granularity and algorithm for striping data on disks. Components should be designed to automatically take advantage of such hints. Another aspect of automatic optimization of data structures and indexes is the ability to infer access patterns. In many cases, access pattern hints are not known ahead of time or may change over time. The principle that helps simplifying the scientist's task in using data management technology is that the data structures should be optimized automatically according to observed access patterns.

Commercial database systems today are mainly based on the relational data model. This implies organizing and storing the data in tabular form, which is not well suited for most scientific data management needs. For example, scientific data are typically organized as specialized multidimensional arrays (such as mesh data) and are stored in specialized file formats (such as netCDF or HDF). Other applications, such as high-energy physics, do not use multidimensional array formats but have their own specialized data formats. In addition, commercial relational database systems are designed for transaction processing, and therefore are inefficient in supporting large volumes of data. On the other hand, relational database systems are well suited to support the information about the scientific datasets, referred to as metadata. We expect such systems to continue to be used for such purposes even when the volume of metadata increases, since it is relatively small compared to the scientific data generated by simulations.

Traditional file systems as provided by commercial vendors are designed to store data on disks or RAID systems in a sequential fashion. This solution does not scale. In order to achieve a scalable solution, nontraditional file systems based on clusters of computers, each supporting multiple disks, are now being developed (e.g., Lustre, GPFS, and PVFS). We expect that these will eventually be routinely supported by commercial vendors, but research to achieve high efficiency of storage and access based on the data structure and access patterns needs to be continued.

Resources Required

Research into data management technologies requires two kinds of resource: hardware and people. Hardware resources are needed to establish prototyping and testing facilities. We estimate that 100 terabytes of disk storage and 5 petabytes of archive capacity, together with the supporting satellite nodes and infrastructure, would be sufficient. We believe that meaningful scientific data management research needs to be performed in a partnership between computer scientists and application scientists; a typical group would contain several researchers plus adequate support staff.

Metrics of Success

The most important metric of the success of scientific data management technology is *user productivity* when dealing with large volumes of data. In other words, success is measured by providing technologies that minimize the data management work needed by the application scientist. Our task, then, is to define metrics for the success of those technologies.

Scalable I/O and *Scalable Archiving.* The obvious metric is to achieve *linear scaling* as the volume of data written by a simulation or read by an analysis program grows. Linear scaling in speed will require superlinear numbers of disk and tape drives and will involve more cost than might be supportable. The challenge, then, is to perform data selection and reduction. Success will be shown if applications do not need to stop computation while data are being transported.

Metadata Management. There is a challenge in automating the generation and organization of the metadata and in developing efficient search tools. A reasonable metric is to have metadata tools that continue to provide on-line performance as the metadata volume grows.

Data Searching. Efficient ad hoc on-line searching is needed for selecting subsets of data during analysis. A metric of the success of efficient data searching is the ability to search over several properties of billions of data objects in less than 1 minute.

Data Moving. Data will always need to be moved, but routinely moving petabytes of data will be prohibitive. Assuming that data searching and filtering methods will greatly reduce the amount of data that need to be moved, a good metric is the ability to move a 10 TB subset in one hour. That will require improvements in the software ensuring robust (lossless) transfers. Note that this requires 2.8 GB/s, or three times OC192 capability.

Data Mining and *Data Reduction.* In order to scale with the volume of data, parallel data mining technology must scale linearly with the number of processors. A good metric is *close-to-linear scaling* with the number of processors.

Interoperability. To make the metric of user productivity a reality, we need to have a framework for *interoperability* of these data management technologies. The aggregate of data management technologies should be packaged, simple to use, and robust. The goal is to empower individual researchers to *find* the data they are interested *in minutes*, to *select* the data they wish to get *in minutes*, and to *get* the data they wish to analyze *in less than an hour*. When data are used to implement dynamic steering of a simulation, such steering should be carried out expeditiously enough that simulation time is not wasted.

Performance Science

By “performance science,” we mean

1. Using benchmarks to assess and compare performance of high-end systems
2. Developing tools to instrument and gather performance data
3. Using performance tools to analyze performance phenomena
4. Using performance modeling to project future performance on specific platforms
5. Using simulations to better understand computer systems performance
6. Improving performance based on information learned from the above methods.

With regard to performance benchmarks, the single most relevant metric of high-end system performance is time to solution for the scientific applications of interest. Time to solution comprises a number of factors, including (1) time devoted to programming and tuning; (2) problem set-up time and grid generation; (3) time spent in batch queues; (4) execution time; (5) I/O time; (6) time lost because of job scheduling inefficiencies, downtime, and handling of system background interrupts; and (7) job postprocessing, including visualization and data analysis. In this paper we focus on execution time, but we emphasize that the others are also significant and should not be ignored. Indeed, no one benchmark or criterion can encompass all aspects of scientific computing, and no one figure of merit should be used to judge the suitability of a high-end system for a target mission.

Effective tools to instrument performance and gather performance data are essential to help scientists diagnose less-than-optimum performance and ultimately improve performance. Such tools are needed because modern systems are highly complex, with numerous underlying subsystems that often act in difficult-to-understand ways when executing a large-scale scientific computer program. Ongoing work in this arena includes the standardization and enhancement of hardware performance–monitoring facilities; the definition and enhancement of low-level microbenchmarks; the management of large volumes of performance data; and the development of intelligent, visualization-based facilities to locate and diagnose performance anomalies.

The objective of performance modeling is to develop a set of low-level system metrics, plus a simple-to-use methodology, for accurately projecting the performance of a specific high-level application program on a specific high-end system. Many sectors of the high-performance computing community stand to benefit from such metrics and methodology, including researchers exploring new system architectures, vendors designing future systems, computing centers procuring such systems, and even individual scientists, who are beginning to use simple performance models to dynamically control the execution path for optimal performance.

At the High-Speed Computing Conference in April 2003, one speaker observed that computational scientists have become quite expert in using high-end computers to model everything except the systems they run on. This is not quite true. A small community of researchers has been employing simulations to study system performance features for some time. Also, computer vendors often develop and use near-cycle-accurate simulators as part of their system development efforts. Simulations were used, for example, in the design of the ASIC device in the "QCD on a chip" system in development at Columbia University and Brookhaven National Laboratory. But these simulations have generally been rather limited in scope because of the very long run times required. Fortunately, a convergence of highly parallel computing technology and research in the parallel discrete event simulation (PDES) field now makes it possible to perform these simulations much faster. One particularly compelling application of such simulations is to study the behavior of various designs for large-scale interprocessor networks. At present, computer system architects and vendors have very little quantitative information on how future networks will behave (or even how present-day networks behave), yet the design of a powerful network is key to reasonable scalability in future high-end systems. Simulations may provide a way to understand these phenomena.

One final aspect of performance science that will we mention here is some work in automatic, or self-tuning, software. Recently, there have been some remarkable successes in efforts to develop highly efficient, self-tuning libraries, notably FFTW, Atlas, and LAPACK for Clusters, as well as frameworks for constructing self-tuning software. These successes suggest that such techniques can be implemented in other scientific libraries. Looking further into the future, we can envision the day when these techniques can be embedded directly into a user's program by the compiler (or a preprocessor), so that many of the requisite decisions can be made at execution time, without need of a pre-execution test run. Considerable work remains, however, in order to realize this vision.

Impact on Applications

We believe all high-end scientific computing applications stand to benefit from advances in performance science across all phases of the application and platform lifetime: design, integration, procurement, installation, and optimization. The following classes of applications appear most ripe for benefit:

- New applications, where performance analysis and tuning can be integrated with the software development process.
- Existing applications that are being rewritten.
- Applications with multiscale, dynamic, or adaptive algorithms, as well as those employing sparse matrix methods. These often achieve less-than-optimal performance levels on high-end systems because of their inherent complexity.
- Time-critical applications, where any reasonable time spent tuning can quickly be recouped in increased value for the researcher or customer.
- Applications that are likely to have a greater than average lifetime.

In addition to its obvious importance in accelerating high-priority scientific applications, performance science also has a unique opportunity to provide valuable quantitative information to architects, vendors, and system procurers, helping them deliver to

scientists the systems that best match the application workload. For example, in practice, DoD and DOE use performance science to drive their procurement analysis and integration debugging. We also note that it is often difficult to reveal the success of performance modeling and analysis in procurement activities because it often deals with sensitive and proprietary information, that is tightly restricted by vendors.

Research Issues

Key challenges include the following:

- Lack of effective, user-friendly tools. Some existing performance tools, although fairly simple in concept, require many thousands of lines of code to implement, and potentially days of effort to become effective in their use. Yet additional investments are needed to improve their effectiveness and user friendliness.
- The advanced level of expertise or computer time required to effectively use present-day performance modeling methodologies. Simpler, easier-to-use schemes need to be developed. A variety of modeling schemes is required, ranging from sophisticated schemes, requiring advanced tools and computer runs, that deliver highly accurate projections, to simple schemes that give good approximations to delivered performance, which can be incorporated into preprocessor tools and compilers.
- Lack of widely accepted simulation tools for analyzing processor, memory system, interconnect, and mass storage performance, to help guide the analysis, procurement and design of high-end systems.
- The need for the very frequent, low-level synchronization required to run performance simulation programs on highly parallel systems. Some techniques are known in the parallel discrete event simulation community, but these have not been implemented in tools aimed at high-end system analysis.
- The relatively weak scalability of present-day performance tools, in terms of both the size of systems that they can effectively analyze and the modest levels of systems that such tools can run on.
- Need for keeping performance tools abreast with changes in software, hardware, programming languages and programming models. For example, significant resources will be required to port existing performance tools to new languages such as UPC, yet without such support these innovative languages may not enjoy a very broad user base.
- Lack of good performance advisor tools. Performance tools need to do more than measure and identify performance phenomena. If these tools are to gain more widespread use, at some point they need to advise the user specifically how to change the code to improve performance.
- Lack of a common, accurate, and practical performance calculus. The performance science community needs a widely used, accurate, and practical calculus to motivate the discussion, manipulation, simulation, measurement, and modeling of today's complex applications. Yet the complexity and scale of today's platforms and applications makes this goal increasingly elusive.

External Dependencies

Most of the performance research work does not rely on advanced numerical algorithms or other sophisticated computer science technology. There is some reliance on effective practices of software engineering, so as to ensure that large-scale performance software tools are both functional and long-lived. This is of particular importance for software that is developed by a large number of people in separate, individual efforts, extending over several years. However, we do not see that fundamental new software engineering technology is required here. Instead, the focus needs to be on using and enforcing existing standards of good programming practice and software project management. Recent work in the field has highlighted the need for a set of standards and interface protocols to be defined and followed in high-end software and hardware. This calls for greater collaboration between computer vendors on the one hand and academic and laboratory scientists on the other. Thus, some thought and discussions are needed to make sure that intellectual property issues can be dealt with in a reasonable manner.

One key area of performance work that is emerging is that of large-scale data management. As we observed above, future systems with tens of thousands of CPUs will compound these challenges. Thus, advanced techniques such as automated instrumentation, on-the-fly performance data reduction and intelligent analysis of performance data will be essential for optimizing performance of scientific applications on future high-end systems.

Delivering Performance Science to Applications

Clearly a key challenge in performance science is encouraging more scientists to use the available tools in their day-to-day program development work. We have mentioned above the fundamental difficulties in devising tools that are both effective yet easy to use.
One important structural obstacle, which a number of us have observed, is that performance tuning often has low priority compared with other scientific program development tasks. In many cases, once the code is basically working and has produced a believable set of results, the inevitable schedule and resource priorities dictate that the scientist move on to other assignments. Yet the code developed in this process is typically used for many production runs; indeed, in some cases these codes may be in use for a far longer time than the developer envisioned. Moreover, the gap between potential and achieved performance often grows with time, because of changes in the architectures of the systems these applications are running on. The result is a suboptimal program, not using the high-end computing resource very well, but with no one available to do any tuning.
In short, it is essential that application development projects include time and people (both application scientists and computer scientists trained in performance tools) in the

schedule for performance tuning and optimization. Without this commitment, business-as-usual will prevail. Along this line, improved analysis and modeling tools would help here, by reducing the time and resources required to do effective tuning.

Resources Required

We decompose the resources required for accelerating the performance science into two major areas: strengthening existing tools for production, large-scale environments, and researching innovative techniques for key challenges, such as scalable data management and the development of a comprehensive performance calculus. Clearly, additional personnel could accelerate the strengthening of existing tools. Additional research funding for research issues would garner the attention of high-caliber research groups that may make tractable progress in this area.

With regard to computational resources, most performance science computing consists of relatively short test runs, rather than the 10,000+ CPU-hour runs typical of large-scale production scientific computing. On the other hand, large-scale system simulation will require significant production resources. Indeed, truly valid simulations, accurate enough that system architects and computer vendors can rely on them to improve their designs, are likely to be quite expensive. We believe that both performance modeling and system simulation are inadequately funded at the present time. System simulation, in particular, receives no specific support from the Department of Energy, although we are aware of two or three small-scale activities funded indirectly by DOE.

With regard to human resources, from our experience the best way to improve the effectiveness of performance tools, as well as to ensure that they are widely used for mission applications, is to forge collaborations between tool developers and application scientists. Computer scientists are often unable by themselves to understand a 100,000-line application code. Application scientists typically do not have the time or resources to become highly skilled in using performance tools on their own (and in some cases they do not have the support for tuning). But as a collaborative team, both parties can make progress, and application scientists can provide feedback to tool developers on how to make tools more usable. Along this line, it is essential that high-end computing centers commit to providing regular upgrades and required maintenance in support of some of the widely used performance tools. Without this commitment, scientists may be unwilling to invest the time to learn them.

One underlying structural problem in this arena, as in much of the high-performance computing world, is that the number of academic researchers pursuing these questions has diminished over the past few years. Perhaps more important, the number of bright graduate and undergraduate students who are electing to pursue research work in this area is depleted. What is needed here is more than just dollars. These students need to be convinced that there is great potential and excitement in this field; otherwise they will find other avenues for their talents.

Along this line, we need more universities to offer coursework in large-scale scientific computing. At present it is quite possible to graduate with a Ph.D. in computer science, for example, without having taken a single course (or part of a course) in numerical methods for scientific computing, much less parallel computing or performance science.

Engineering and physical science departments often do better in training students in basic numerical methods, but seldom do they offer any curriculum in parallel computing or performance science. From experience with several universities offer good coursework in this area, we observe that the most successful programs are interdisciplinary programs that often join forces with large HEC centers, such as the DOE laboratories or the NSF Supercomputing Centers.

Metrics of Success

The principal metric of success, one that is both meaningful and technically valid, is the following: Research in performance science is deemed successful to the extent that it reduces *time to solution* for key scientific applications.
Related metrics include the following:

- Dollar value of computing resources saved.
- Number of scientists actively using performance tools to analyze and improve the performance of their codes (this is a metric for the effectiveness and user-friendliness of performance tools).
- Ratio of performance improvement to tool use time. A tool or methodology that achieves a 30% performance improvement after several weeks of effort is clearly inferior, by this metric, to one that yields 20% (or even 10%) improvement after just a day or two.

Although these factors are quantifiable, they can be extremely intricate to measure in a practical environment and there have been few, if any, longitudinal studies of this topic. Recently, the DARPA High Productivity Computing Systems (HPCS) program has initiated a study of scientist productivity in high-end computing. In particular, this project is targeted toward understanding the effort required to develop, debug, and tune applications on high-end systems.

System Software and Large-scale Simulation

Systems software is a broad area that generally includes all the software needed to manage and maintain a large production computer center. Running ultrascale simulations at these computer centers will require systems software with scalability, adaptability, and fault tolerance beyond what exists today. The major categories of systems software are as follows:

Resource management takes care of scheduling jobs to most effectively utilize the supercomputers based on the application needs and priorities.

Accounting and user management provides access security to the supercomputers and keeps track of how much time each user has accrued.

Job management allows users to submit and query the status of their jobs. It is also used to interactively launch jobs.

System Monitoring monitors all the computers in the center, providing information on availability, status, and potential faults.

Configuration management is used to start up new computers and to reconfigure and reboot existing supercomputers as required by application needs as well as software upgrades.

Operating System (OS) provides the functions and runtime environment on the supercomputer. The OS overlaps with several other areas in this report including architecture, data management (parallel I/O), frameworks, and performance. Here we focus on those parts of the OS and systems software that are not covered by these other areas.

Impact on Applications

Beyond the obvious observation that no applications can run without systems software, there are many opportunities for systems software to reduce the time required by large science applications. Through resource management it is possible to ensure that large science applications get the majority of the time on the supercomputers and that smaller jobs are scheduled on other resources. When the applications are running, the OS can get out of the way. Such *low-impact* OS provides only those functions needed by the application and disables functions, such as sendmail, that might interrupt the computer during processing.

The mean time between failure (of some portion) of today's supercomputers is measured in days; thus, for applications that run for weeks or months, fault tolerance is already an important issue.

There are many types of faults that may affect ultrascale simulations. There are hardware failures, software failures, and failures inside the application itself. The most obvious and easiest to predict are hardware failures. The reliability of individual hardware

components is quite good but the shear number of components in an ultrascale computer, leads to the prediction that the computer will have to be able adapt to hardware component failures. The system software is an even more common cause of failure. Just considering the how often an OS will hang on a desktop, it is easy to see how a computer composed of thousands of interacting processors has many opportunities for software faults. Ultrascale systems software will need to be designed to adapt and reconfigure around such faults. Even if the hardware and software work fine, there is still the possibility of something going wrong in the application itself.

With so many opportunities for faults it has been predicted that for petascale computers the mean time to failure of one component could be as low as a few minutes. Such computers will require systems software that is tolerant of partial failures and is able to reconfigure the computer dynamically. The systems software will also be responsible for automatically migrating and restarting long-running applications.

The runtime environment provides the framework for the programming model used by the application. The most popular programming model today is message passing, but future programming models will have to be supported by the systems software if the applications require it. Debugging, validation, and system diagnostics were also identified as becoming important to ultrascale simulations.

Research Issues

The nation's premier scientific computing centers are facing a crisis. They all use incompatible, ad hoc sets of systems software, and this software is not designed to scale to the multi-teraflop systems that are being installed in these centers today. One solution would be for each computer center to rewrite its homegrown software to be scalable. But this would incur a tremendous duplication of effort and would delay the availability of the multi-teraflop computers for scientific discovery.

Standardizing interfaces in systems software is a catalyst for fundamentally changing the way future high-end systems software is developed and distributed. Research and development in this area will reduce facility management costs by reducing the need to support home-grown software, making higher-quality systems tools available, and enabling new machines to be up and running sooner. Standardization will also facilitate more effective use of machines by scientific applications by providing scalable job launch, standardized job monitoring and management software, and allocation tools for the cost effective management and utilization of the computational resources.

Besides the advantage of having standardized user interfaces to improve the usability of the production systems by the scientist, systems software research also addresses the following critical issues during the running of science applications.

Improved application performance by having an OS that does not get in the way of the application. Operating systems such as Linux have many programs running in the background that periodically interrupt an application's progress in order to check whether they should run. Even when all these extraneous programs are disabled in Linux, the Linux OS itself has problems scaling to thousands of processors. For these reasons there needs to be a high-performance OS research effort to complement ultrascale simulation

efforts. This OS research should be built on a flexible framework that allows the OS to be lightweight or heavyweight as required by the different application areas and be portable to different architectures. Eventually such an OS could adapt itself to have only those functions required for a given application; such an OS could even tune its parameters to improve an application's performance on a given architecture.

Support for parallel I/O and external networks required to get the data in and results out of the applications is a key CS research area. While Chapter [DATA] in this report discusses the issues of data management and parallel I/O, the OS is the interface between the application and the file system. Interaction between high-performance OS and parallel I/O research is crucial to meeting the data requirements of the applications.

System fault management today consists of restarting a job from its last checkpoint if a system fault occurs. As the size of computers grows, the time it takes to reload a job grows larger while the time between faults gets shorter. Some scientists have estimated that on a petascale computer, faults could occur faster than a job could be reloaded. Thus, next-generation computers may require a new approach to system fault management that detects, adapts, and reconfigures around faults to allow long-running applications to complete. For large computers, fault management research may be even more important than performance research because if an application fails before giving a result, it doesn't matter how fast the computer was running up to that point.

Fault management comprises two levels. First, the systems software itself needs to be able to survive a fault and to reconfigure the resources accordingly. Second, the applications need to be designed with fault tolerance in mind. Research is needed at both levels in order for long-running simulations to complete.

Scalable resource management is critical to efficiently schedule the applications to best utilize the available resources and meet the applications' deadlines. Most systems software today is unable to scale to the tens of thousands of processors that petascale computers will have. Significant scalability research is required across the entire system software suite. In particular, the resource management software will need to provide support for migration and dynamic job sizing so that applications can be dynamically configured around faults or moved to more efficiently utilize the overall computer resources, for example, to make room for another job to be started.

Programming model support for existing and future programming methodologies portably across all the top-end architectures is an important application need. The runtime environment is responsible for supporting the programming model used in the scientific application. While message passing, in particular MPI, is the predominant programming model in use today, it is not the only programming model. The runtime environments that are developed for the next generation supercomputers need to be flexible enough to support a variety of programming models. These issues are discussed in more detail in the chapters on architecture and programming models.

Resources Required

The software running the largest computers in the nation should not be a "student project" but rather, the result of professional software developers with experience with supercomputers. There should also be a conscious investment in the management and

maintenance of the developed systems software; otherwise the high-performance OS and scalable systems software will become obsolete and the development investment wasted. Before SciDAC there was no investment in scalable systems software. As a result all the major supercomputer centers today use ad hoc software to manage their systems. They are finding that their homegrown software does not work for the terascale computers. With an investment in standardization and maintenance of systems software, the nation's supercomputing centers will be able to leverage each other's system software efforts and avoid the expense of rewriting homegrown software.

Hardware testbeds are needed in order to test and debug OS and system software at scale before it can be moved to the nation's premier supercomputers. The production supercomputers cannot afford to have their science applications disrupted in order to debug OS software. Today system software is tested on small testbeds such as Chiba City at Argonne National Lab or XTORC at Oak Ridge National Lab. These testbeds are much too small for ultrascale development. What is required is a system software testbed with at least a thousand processors on which to harden new high performance OS and test fault management at scale.

Of particular concern is the fact that, in the past decade, the number of students doing research in high-performance computing has decreased markedly. Investment is needed in the education of the next generation of computer scientists who will carry on the research required in scalable systems software. Between 1985 and 1995 all the major computer science departments had courses and research efforts in high performance computing. The dotcom phenomena shifted most CS major's attentions to web servers and web applications. Even though the dotcom rush has faded, attention has not returned to high performance computing. Investment is needed to re-spark university interest in this area.

External Dependencies

Systems software must support the programming models chosen by the applications. It must work across multiple architectures chosen and designed by the vendors. The software must scale from small PC clusters to petascale supercomputers in order to cover all the resources chosen by large computer centers. The systems software must be able to handle a variety of usage models. Some applications may require thousands of ensemble runs, others may require the entire machine for a month for one run, and still others may desire to run constantly in the background. Flexibility to adapt to all these external dependencies will be an important aspect of the next-generation systems software.

Metrics of Success

The primary metric of success is increased scientific productivity. This can manifest in systems software in many ways, most of them completely transparent to the users. For example, next-generation machines could be available for production sooner because the systems software was designed to be portable to new architectures and scalable to tens of thousands of processors. In this case a new machine could go into production in a month instead of half a year. Another example would be the ability for the systems software to take care of any detection and fault management that may be required for a long-running

job to complete. In this case a scientist could submit a ground breaking calculation that will take 3 months to complete with complete confidence that the job will complete regardless of faults in the hardware or system software during those 3 months.

A less transparent metric is the perceived usability of the production systems. The ability to have common, portable job management and system monitoring interfaces regardless of what architectures or computer centers are being used will allow scientists to become familiar with the user environment and allow reuse of their submission scripts. Scientists should be focused on breakthrough science, not worrying about how to submit a job to the newest ultrascale computer. Standardized system software would help keep the focus on science. This could be measured by the number of calls to the computer center's help desk about systems software rather than improving their science application.

It is said that if the systems software is successful, one never hears about it. Success is having no complaints from the users.