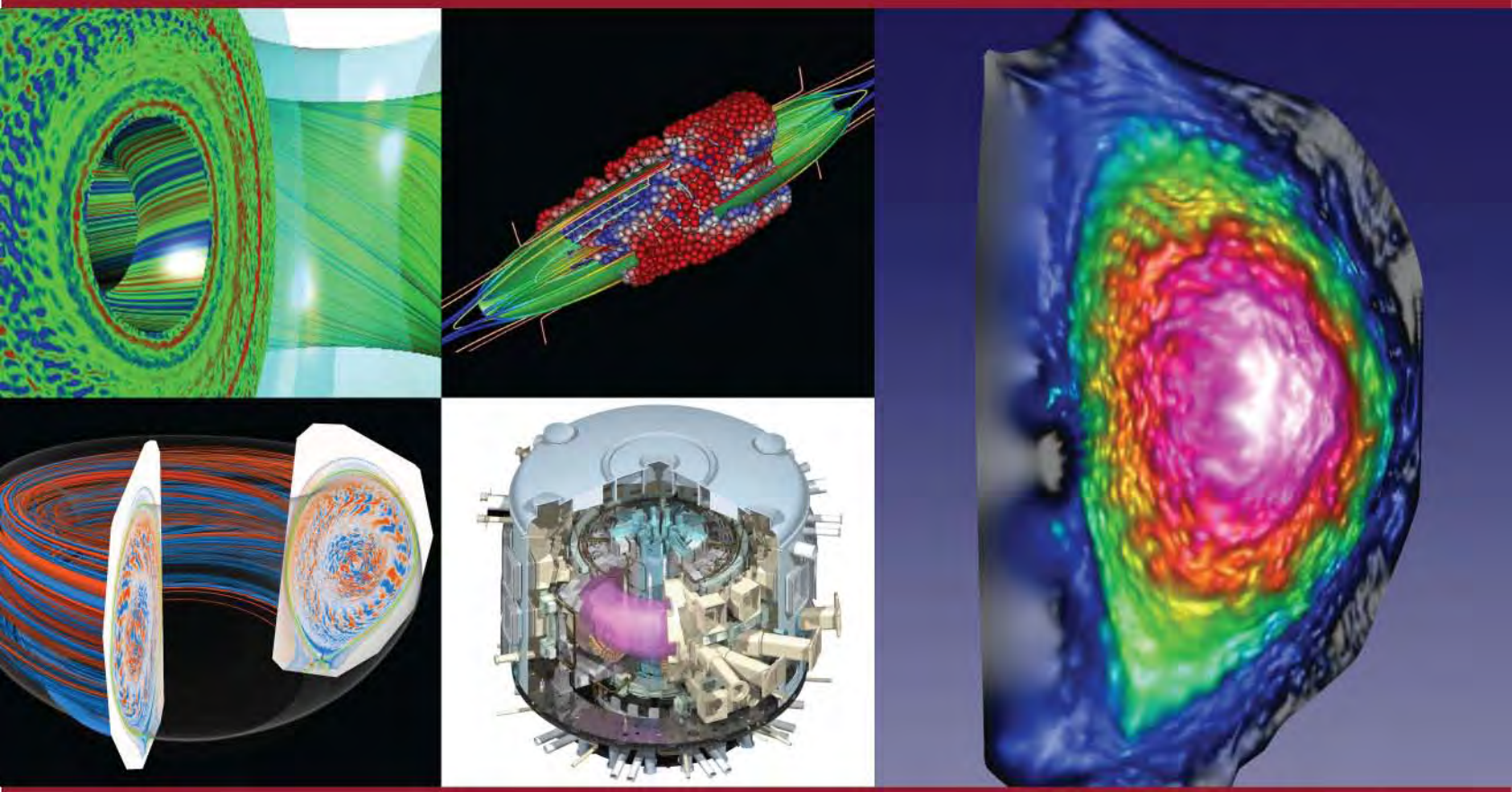
LARGE SCALE COMPUTING AND STORAGE REQUIREMENTS



Fusion Energy Sciences

Report of the NERSC / FES / ASCR
Requirements Workshop
August 3 and 4, 2010

DISCLAIMER

This report was prepared as an account of a workshop sponsored by the U.S. Department of Energy. Neither the United States Government nor any agency thereof, nor any of their employees or officers, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of document authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Copyrights to portions of this report (including graphics) are reserved by original copyright holders or their assignees, and are used by the Government's license and by permission. Requests to use any images must be made to the provider identified in the image credits.

Ernest Orlando Lawrence Berkeley National Laboratory is an equal opportunity employer.

Ernest Orlando Lawrence Berkeley National Laboratory

University of California

Berkeley, California 94720 U.S.A.

NERSC is funded by the United States Department of Energy, Office of Science, Advanced Scientific Computing Research (ASCR) program. Yukiko Sekine is the NERSC Program Manager and John Mandrekas serves as the FES allocation manager for NERSC.

NERSC is located at the Lawrence Berkeley National Laboratory, which is operated by the University of California for the US Department of Energy under contract DE-AC02-05CH11231.

This work was supported by the Associate Directors of the Office of Science for Fusion Energy Sciences and for Advanced Scientific Computing Research, Facilities Division.

This is LBNL report LBNL-5247E published December, 2011.

Large Scale Computing and Storage Requirements for Fusion Energy Sciences Research

Workshop Report

Workshop Conducted August 3-4, 2010

Rockville, MD

DOE Office of Science

Office of Fusion Energy Sciences (FES)

Office of Advanced Scientific Computing Research (ASCR)

National Energy Research Scientific Computing Center (NERSC)

Editors

Richard A. Gerber, NERSC

Harvey J. Wasserman, NERSC

Table of Contents

1 Executive Summary 1

2 DOE FES Mission 2

3 About NERSC 5

4 Workshop Background and Structure 7

5 Workshop Demographics 8
- **5.1 Participants 8**
- **5.2 NERSC Projects Represented by Case Studies 9**

6 Findings 10
- **6.1 Summary of Requirements 10**
  - 6.1.1 Scientists in FES need large systems and large allocations of computational resources to meet their research goals. 10
  - 6.1.2 Codes used by the fusion research community require support for a variety of complex software libraries and tools. 10
  - 6.1.3 Data storage systems must support high-bandwidth and large-capacity runtime I/O and permanent storage for archiving and sharing. 11
  - 6.1.4 Computational jobs will require robust, highly available systems with high throughput, long run times, and short queue waits. 11
- **6.2 Other Significant Observations 11**
- **6.3 Computing Requirements 13**

7 NERSC Initiatives and Plans 14
- **7.1 Compute Resources 14**
- **7.2 Software 16**
- **7.3 Data 16**
- **7.4 Job Turnaround and Throughput 17**

8 Plasma Turbulence & Transport 18
- **8.1 Plasma Turbulence and Transport Overview 18**
- **8.2 Plasma Turbulence Case Studies 20**
  - 8.2.1 The Center for Plasma Edge Simulation (CPES) 20
  - 8.2.2 Magnetic Fusion Plasma Microturbulence 26
  - 8.2.3 Gyrokinetic Simulation of Energetic Particle Turbulence and Transport (The SciDAC GSEP Center) 31
  - 8.2.4 Global Gyrokinetic PIC Simulations of Plasma Microturbulence 36

9 Magnetohydrodynamics 42
- **9.1 Magnetohydrodynamics Modeling and Simulation Overview 42**
  - 9.1.1 Extended Magnetohydrodynamics Simulation of Tokamak Plasmas 42

10 Fusion Simulation Program 46
- **10.1 The Fusion Simulation Program 46**
  - 10.1.1 Summary and Scientific Objectives 46

11 Innovative Confinement Concepts 50

11.1 Overview 50
11.2 The Plasma Science and Innovation (PSI) Center 50

12 Wave-Plasma Interactions 53
12.1 Center for Simulation of Wave-Plasma Interactions (CSWPI) 53

13 Heavy-Ion Fusion Science and Inertial Fusion Energy 57
13.1 Simulation of Intense Beams and Targets for Heavy-Ion-Fusion Science 57
13.2 Large-Scale Particle-in-Cell Simulations of Laser-Plasma Interactions Relevant to Inertial Fusion Energy 67

14 General Plasma Science 71
14.1 Center for Integrated Analysis and Computation of Reconnection and Turbulence 71

15 Atomic Physics 78
15.1 Atomic and Molecular Collision Physics for Controlled Fusion Energy 78

16 Plasma Materials Interactions 82
16.1 Modeling Plasma Materials Interactions 82

Appendix A. Attendee Biographies 87

Appendix B. Workshop Agenda 92

Appendix C. Abbreviations and Acronyms 93

Appendix D. About the Cover 95

1 Executive Summary

The National Energy Research Scientific Computing Center (NERSC) is the leading scientific computing facility for the Department of Energy's Office of Science, providing high-performance computing (HPC) resources to more than 4,000 researchers working on over 550 projects. In addition to large-scale computing resources NERSC provides critical staff support and expertise to help scientists make the most efficient use of these resources to advance the scientific mission of the Office of Science.

In August 2010, NERSC, DOE's Office of Advanced Scientific Computing Research (ASCR) and DOE's Office of Fusion Energy Sciences (FES) held a workshop to characterize HPC requirements for FES research over the next three to five years. The effort is part of NERSC's continuing involvement in anticipating future user needs and deploying necessary resources to meet these demands.

The workshop revealed several key points, in addition to achieving its goal of collecting and characterizing computing requirements. Key requirements for scientists conducting research in FES include:

1. Larger allocations of computational resources at NERSC;
2. Continued support for a complex ecosystem of software libraries and tools;
3. Data storage systems that can support high-volume/high-throughput I/O; and
4. Robust, highly available computational systems with high throughput, long run times, and short queue waits.

This report expands upon these key points and adds others. The results are based upon representative case studies that illustrate the needs of science teams within FES. The case studies were prepared by FES workshop participants and contain a summary of science goals, methods of solution, current and future computing requirements, and special software and support needs. Participants were also asked to describe their strategy for computing in the highly parallel, "multi-core" environment that is expected to dominate HPC architectures over the next few years.

The report also includes a section with NERSC responses to the workshop findings. NERSC has many initiatives already underway that address key workshop findings and all of the action items are aligned with NERSC strategic plans.

2 DOE FES Mission

The Fusion Energy Sciences (FES) mission is to expand the fundamental understanding of matter at very high temperature and density and to build the scientific foundation needed to develop a fusion energy source. This is accomplished by studying plasma and its interactions with its surroundings across wide ranges of temperature and density, developing advanced diagnostics to make detailed measurements of its properties and dynamics, and creating theoretical and computational models to resolve the essential physics principles.

FES has four strategic goals:

- Advance the fundamental science of magnetically confined plasmas to develop the predictive capability needed for a sustainable fusion energy source;
- Support the development of the scientific understanding required to design and deploy the materials needed to support a burning plasma environment;
- Pursue scientific opportunities and grand challenges in high energy density plasma science to explore the feasibility of the inertial confinement approach as a fusion energy source, to better understand our universe, and to enhance national security and economic competitiveness, and;
- Increase the fundamental understanding of basic plasma science, including both burning plasma and low temperature plasma science and engineering, to enhance economic competiveness and to create opportunities for a broader range of science-based applications.

FES research is directed towards developing a predictive understanding of plasma properties, dynamics, and interactions with surrounding materials. The greatest emphasis is presently weighted towards understanding the plasma state relevant to stable magnetically confined fusion systems, but increasing emphasis is expected in the areas of plasma-material interaction physics and the materials science associated with the high heat and neutron fluxes that will be encountered in a burning plasma environment. FES programs also encompass research in high energy density laboratory plasma physics and general plasma science.

In the last two decades, progress in our understanding of plasma systems and their control requirements has enabled the fusion community to move to the edge of a new era, the age of self-sustaining “burning” plasmas. For both magnetic and inertial fusion, new experimental plans are being developed to make historic first studies of fusion systems where the energy produced in the fusion process is substantially greater than the energy applied externally to heat and control the plasma. In a burning plasma, energy confinement, heating, and stability affect each other in ways we need to predict, and the scientific issues associated with creating and sustaining power-producing plasma can be

explored directly. The flagship program of this new era is the ITER project, an international fusion research project being constructed in Cadarache, France, that will realize magnetically confined burning plasmas for the first time.

From the days of the National Magnetic Fusion Energy Computer Center (MFECC) in the mid-1970's—the predecessor of NERSC—high performance computing (HPC) and NERSC have played a significant role in fusion energy research. Advanced simulations are critical for advancing the FES mission and achieving its goals, especially the development of a predictive capability needed for a sustainable fusion energy source. While most FES program areas have been taking advantage of the opportunities afforded by high performance computing, advanced simulations enabled by access to NERSC resources are especially important in the following areas:

- ***Theory***: The FES theory program focuses on advancing the scientific understanding of the fundamental physical processes governing the behavior of magnetically confined plasmas. In addition to its scientific discovery mission, the theory program is also responsible for providing the firm scientific grounding and establishing limitations and ranges of applicability of the underlying physics models implemented into large scale simulation codes. Strong synergies and connections with other program elements such as General Plasma Science exist, from the crosscutting science of magnetic reconnection and plasma turbulence, to the study of fusion-relevant atomic and molecular processes in plasmas.
- ***SciDAC***: FES has invested significantly in the Office of Science's (SC) Scientific Discovery through Advanced Computing (SciDAC) program. The FES SciDAC portfolio is aimed at advancing scientific discovery in fusion plasma science by exploiting leadership class computational systems and associated advances in computational science. Current areas of focus include magnetohydrodynamics, plasma turbulence and transport, wave-plasma interactions, and energetic particle effects. A computational materials project is expected to be added in FY 2012. The FES SciDAC projects are among the most demanding users of NERSC resources, being responsible for using more than 55% of the FES annual allocation.
- ***FSP***: Encouraged by the success of SciDAC, FES is currently considering the initiation of the Fusion Simulation Program (FSP), a multi-year major computational effort aimed at the development of an experimentally validated integrated simulation capability for ITER and burning plasmas.

Advanced simulations are expected to become increasingly important in other areas of the FES program which have been historically modest users of NERSC resources, such as materials science, experimental validation enabled by targeted validation platforms, and high energy density plasma science.

3 About NERSC

The National Energy Research Scientific Computing (NERSC) Center, which is supported by the U.S. Department of Energy's Office of Advanced Scientific Computing Research (ASCR), serves more than 4,000 scientists working on over 550 projects of national importance. Operated by Lawrence Berkeley National Laboratory (LBNL), NERSC is the primary high-performance computing facility for scientists in all of the research programs supported by the Department of Energy's Office of Science. These scientists, working remotely from DOE national laboratories; universities; other federal agencies; and industry, use NERSC resources and services to further the research mission of the Office of Science (SC). While focused on research that supports DOE's missions and scientific goals, computational science conducted at NERSC spans a range of scientific disciplines, including physics, materials science, energy research, climate change, and the life sciences. This large and diverse user community runs hundreds of different application codes. Results obtained using NERSC facilities are citied in about 1,500 peer reviewed scientific papers per year. NERSC activities and scientific results are also described in the center's annual reports, newsletter articles, technical reports, and extensive online documentation. In addition to providing computational support for projects funded by the Office of Science program offices (ASCR, BER, BES, FES, HEP and NP), NERSC directly supports the Scientific Discovery through Advanced Computing (SciDAC[1]) and ASCR Leadership Computing Challenge[2] Programs, as well as several international collaborations in which DOE is engaged. In short, NERSC supports the computational needs of the entire spectrum of DOE open science research.

The DOE Office of Science supports three major High Performance Computing Centers: NERSC and the Leadership Computing Facilities at Oak Ridge and Argonne National Laboratories. NERSC has the unique role of being solely responsible for providing HPC resources to all open scientific research areas sponsored by the Office of Science. The Leadership Computing Facilities support a more limited number of select projects, whose research areas may not span all Office of Science objectives and are not restricted to mission-relevant investigations.

This report illustrates NERSC's alignment with, and responsiveness to, DOE program office needs, in this case the needs of the Office of Fusion Energy Sciences. The large number of projects supported by NERSC, the diversity of application codes, and its role as an incubator for scalable application codes present unique challenges to the center. As demonstrated by the overall scientific productivity by NERSC users, however, the combination of effectively managed resources and excellent user support services, the NERSC Center continues its 35-year history as a world leader in advancing computational science across a wide range of disciplines.

[1] http://www.scidac.gov
[2] http://science.energy.gov/~/media/ascr/pdf/incite/docs/Allocation_process.pdf

For more information about NERSC visit the web site at http://www.nersc.gov.

4 Workshop Background and Structure

In support of its mission and to maintain its reputation as one of the most productive scientific computing facilities in the world, NERSC regularly collects user requirements from a variety of sources. Methods include scrutiny of the NERSC Energy Research Computing Allocations Process (ERCAP) allocation requests to DOE; workload analyses; and discussions with DOE program managers and scientist customers who use the facility.

In August 2010, the DOE Office of Advanced Scientific Computing Research (ASCR, which manages NERSC), the DOE Office of Fusion Energy Science (FES), and NERSC held a workshop to gather HPC requirements for current and future science programs funded by FES. This report is the result.

This document presents a number of consensus findings. The findings are based upon a selection of case studies that serve as representative samples of NERSC research supported by FES. The case studies were chosen by the DOE Program Office Manager and NERSC personnel to provide broad coverage in both established and incipient FES research areas. Since FES supports many research endeavors in these fields the case studies presented here do not represent the entirety of FES research.

Each case study contains a description of scientific goals for today and for the future, a brief description of computational methods used, and a description of current and expected future computing needs. Since supercomputer architectures are trending toward systems with chip multiprocessors containing hundreds or thousands of cores per socket and perhaps millions of cores per system, participants were asked to describe their strategy for computing in such a highly parallel, “multi-core” environment.

Requirements presented in this document will serve as input to the NERSC planning process for systems and services, and will help ensure that NERSC continues to provide world-class resources for scientific discovery to scientists and their collaborators in support of the DOE Office of Science, Office of Fusion Energy Sciences.

NERSC and ASCR have been conducting requirements workshops for each of the six DOE Office of Sciences offices that allocate time at NERSC (ASCR, BER, BES, FES, HEP, and NP). The process began in May 2009 (with BER) and will conclude in May 2011 (with NP). The target for science goals and computing requirements has been approximately 2013 for each workshop.

Specific findings from the workshop follow**.**

5 Workshop Demographics

5.1 Participants

Name	Institution	Area of Interest	NERSC Repo(s)
John Mandrekas	DOE FES	FES Program Manager Theory & Advanced Fusion Simulations; FES HPC Allocations	
Yukiko Sekine	DOE ASCR	NERSC Program Manager	
Lee Berry	ORNL	Magnetic Fusion Energy	m77, m876
Jeff Candy	General Atomics	Magnetic Fusion Energy	m681, mp94, gc3
CS Chang	NYU[3]	Magnetic Fusion Energy	m499, mp338, m77
Stephane Ethier	Princeton Plasma Physics Laboratory	Magnetic Fusion Energy	mp19, m912, m499
Alex Friedman	LLNL & LBNL	High Energy Density Laboratory Plasmas / Inertial Fusion Energy	mp42
Kai Germaschewski	University of New Hampshire	General Plasma Science	m148
Martin Greenwald	MIT	Magnetic Fusion Energy	
Stephen Jardin	Princeton Plasma Physics Laboratory	Magnetic Fusion Energy	m876, mp288
Charlson C. Kim	University of Washington	Magnetic Fusion Energy	m489, mp21, mp200
Zhihong Lin	UC Irvine	Magnetic Fusion Energy	m499, m808, m92
John Ludlow	Auburn University	General Plasma Science	m41, mp48
Doug McCune	Princeton Plasma Physics Laboratory	Magnetic Fusion Energy	m681
Linda Sugiyama	MIT	Magnetic Fusion Energy	m499, m224, mp288
William Tang	Princeton Plasma Physics Laboratory	Magnetic Fusion Energy	
Xianzhu Tang	LANL	Materials	mp204, m139
Frank Tsung	UCLA	High Energy Density Laboratory Plasmas / Inertial Fusion Energy,	m1110
Brian Wirth	UC Berkeley[4]	Materials	m916
Alice Koniges	NERSC	Magnetic Fusion/HEDP	
Richard Gerber	NERSC	Workshop Facilitator	
Harvey Wasserman Kathy Yelick	NERSC NERSC	Workshop Facilitator NERSC Director	

[3] Moved the Princeton Plasma Physics Laboratory (PPPL) in mid-2011
[4] Currently at the University of Tennessee

5.2 NERSC Projects Represented by Case Studies

The NERSC projects represented by the workshop case studies are listed in the table below, along with the number of NERSC hours used by those projects in 2010. The workshop attendees represented a large fraction of the FES research performed at NERSC, with 91% of the FES time allocated to a project for which one of the workshop attendees was the Principle Investigator or a senior researcher. There is also one case study representing new work, for which there is no existing project at NERSC (see Chapter 9).

NERSC Project ID (Repo)	NERSC Project Title	Principal Investigator	Workshop Speaker	Hours Used at NERSC
m77	*Center for Simulation of Wave-Plasma Interactions: SciDAC Project*	Paul Bonoli, MIT	Lee Berry	4.9 M
m499	*Center for Plasma Edge Simulation: SciDAC FSP Prototype Center*	C.S. Chang (NYU)	C. S. Chang	10.2 M
mp19	*Turbulent Transport and Multiscale Gyrokinetic Simulation*	Wei-li Lee (PPPL)	C. S. Chang	13 M
gc3	*Magnetic Fusion Plasma Microturbulence*	Bruce Cohen, LLNL	C. S. Chang (and Jeff Candy)	1.5 M
m808	*SciDAC GSEP: Gyrokinetic Simulation of Energetic Particle Turbulence and Transport*	Zhihong Lin (UC Irvine)	C. S. Chang	1.6 M
mp288	*3D Extended MHD simulation of fusion plasmas*	Stephen Jardin (PPPL)	Stephen Jardin / Linda Sugiyama	1.0 M
m148	*Center for Integrated Computation and Analysis of Reconnection and Turbulence*	A. Bhattacharjee (U. New Hampshire)	Kai Germaschewski	2.4 M
m489	*The Plasma Science and Innovation Center*	Brian Nelson (U. Washington)	Charlson Kim	0.8 M
m916	*Ab-initio Modeling of the Energetics and Structure of Nanoscale Y-Ti-O Cluster Precipitates in Ferritic Alloys*	Brian Wirth (UC Berkeley)	Brian Wirth	0.6 M
m1110	*Large scale particle-in-cell simulations of laser-plasma interactions relevant to Inertial Fusion Energy*	Frank Tsung (UCLA)	Frank Tsung	1.5 M
mp42	*Large scale Particle-in-Cell simulations of laser-plasma interactions relevant to Inertial Fusion Energy*	Alex Friedman (LBNL)	Alex Friedman	0.4 M
m41	*Computational Atomic Physics for Fusion Energy*	Michael Pindzola (Auburn U.)	John Ludlow	9.3 M
			Total Represented by Case Studies	**47 M**
			All FES at NERSC in 2010	**51 M**
			Percent of NERSC FES Represented by Case Studies	**91 %**

6 Findings

6.1 Summary of Requirements

The following is a summary of consensus requirements derived from the case studies. Note that many requirements are stated individually but are in fact closely related to and dependent upon others.

6.1.1 Scientists in FES need large systems and large allocations of computational resources to meet their research goals.

a) Researchers in FES anticipate needing 1.4 billion hours of computing at NERSC to support their research in 2013, about forty times more than they used in 2010.

b) Realistic simulations using full-function codes for fusion plasmas in devices such as ITER will need to use more than 10^6 cores for the largest runs and 10^4 to 10^5 cores for hundreds of routine runs.

c) Integrated transport-MHD modeling will likely require HPC resources at the Exascale to address the problem of electromagnetic turbulence relevant to ITER.

6.1.2 Codes used by the fusion research community require support for a variety of complex software libraries and tools.

a) Examples include the PETSc and SLEPc solver libraries and the packages supported through those, as well as standard math libraries LAPACK, ScaLAPACK, BLAS, and FFTW and special function libraries.

b) Support for optimized IO libraries is crucial; these include HDF5, MPI-IO, netCDF, and ADIOS (Adaptable I/O System).

c) Robust visualization and analysis tools are needed, e.g. VisIt.

d) Tools in common use by the scientific community must be supported. This includes Python (and its derivatives SciPy and NumPy), which requires an operating system that has broad support for standard LINUX/UNIX system calls and dynamic shared-object libraries.

6.1.3 Data storage systems must support high-bandwidth and large-capacity runtime I/O and permanent storage for archiving and sharing.

a) Simulations running on full- or nearly-full systems may write large portions of supercomputer memory to persistent storage for checkpoint/restart. This will require petabytes of online storage.

b) Archival storage will also consume multi-petabytes of space.

c) There is a need for efficient high-level parallel libraries/data models such as Parallel netCDF, parallel HDF5, and ADIOS.

6.1.4 Computational jobs will require robust, highly available systems with high throughput, long run times, and short queue waits.

a) Integrated whole-device modeling of large devices — as required for ITER, DIII-D, and other experiments — will require simulations that run for long periods of time and faster turnaround is required for research progress.

b) Interactive use of supercomputers is needed for code development, analysis, and optimization to allow scaling to large parallel concurrency.

c) Many long ensemble runs that explore parameter space are needed using a large integrated machine.

d) Short wait times for medium core counts are required to make sustained progress.

e) Large suites of verification and validation runs are necessary for building a predictive capability.

6.2 Other Significant Observations

- In the five-year time frame ITER data analysis will pose some serious computational, networking, and data challenges. The Fusion community may be facing a major change in the way data are dealt with, perhaps moving from the current “data file paradigm” to a “data streaming paradigm” to accommodate much larger data sets. Realistic development of such capabilities represents a major challenge.

- At the time this workshop took place realistic simulations using full-function codes for fusion plasmas in devices such as ITER were out of the reach of computational systems installed at NERSC. Although the Hopper system will provide much-needed additional resources there remains a need to employ codes

with greater physics fidelity that can be *orders of magnitude slower* than the simpler codes.

- There is some overlap in research goals and needs with NNSA, such as in the areas of IFE and materials modeling.

- Applications span a wide range of numerical methods: FFTs, dense and sparse linear algebra, particle methods, structured grids with AMR, and unstructured grids. There is a need to develop new methods for better scalability and memory efficiency.

- HPC requirements in FES are closely tied to well-defined computational demands that arise from specific research goals. Some of these include the need to perform whole-device modeling of larger physical systems (e.g., ITER vs. DIII-D); to include more accurate physics (e.g., full- instead of simplified-velocity electrons); to utilize higher resolution grids (e.g., to capture beam-in-plasma effects); and to run for longer simulation times (to span different stages of plasma discharge).

- Particle in Cell methods represent one of the most important algorithms employed within FES. Codes implementing this method generally require a low interconnect latency and fast gather-scatter to memory. A low-latency, high-bandwidth interconnect is also required for extended magnetohydrodynamics simulations.

- The size of simulation data sets is beginning to challenge existing dedicated visualization computers such as NERSC's Euclid system.

- Although impressive performance gains have been demonstrated for some CPU-intensive kernels in FES codes there is no full mainstream plasma simulation code that is currently ready to migrate to GPU technology. There is concern in the community about the programming effort required to port codes.

6.3 Computing Requirements

The following table lists the computational hours required by research projects represented by case studies in this report. "Total Scaled Requirement" at the end of the table represents the hours needed by all 2010 FES NERSC projects if increased by the same factor as that needed by the projects represented by case studies in this report.

NERSC Project Title	Principal Investigator	Hours Needed in 2013	Increase Over 2010 NERSC Use
Center for Plasma Edge Simulation	*C. S. Chang*	300 M	38
Magnetic Fusion Plasma Microturbulence	*Cohen / Candy*	2.4 M	2
Gyrokinetic Simulation of Energetic Particle Turbulence and Transport (SciDAC GSEP)	*Zhihong Lin*	100 M	67
Global Gyrokinetic PIC Simulations of Plasma Microturbulence	*Lee / Ethier / Wang*	130 M	10
Extended MHD Simulation of Tokamak Plasmas	*Stephen Jardin*	256 M	32
Fusion Simulation Program	*Tang*	N/A	N/A
Plasma Science and Innovation (PSI) Center	*Brian Nelson*	30 M	38
Center for Simulation of Wave-Plasma Interactions	*Paul Bonoli*	50 M	33
Simulation of Intense Beams and Targets for Heavy-Ion-Fusion Science	*Alex Friedman*	60 M	100
Large-Scale Particle-in-Cell Simulations of Laser-Plasma Interactions Relevant to Inertial Fusion Energy	*Frank Tsung*	40 M	27
Center for Integrated Analysis and Computation of Reconnection and Turbulence	*Bhattacharjee*	50 M	10
Computational Atomic Physics for Fusion Energy	*Pindzola*	60 M	6
Plasma – Materials Interactions	*Brian Wirth*	60 M	83
Fusion Simulation Program	*William Tang*	136 M	N/A
Total Represented by Case Studies		1252 M	41
All FES at NERSC Total Scaled Requirement		1375 M	41

7 NERSC Initiatives and Plans

NERSC has initiatives already underway and long-term strategic plans that address some requirements presented in this report. A brief summary of these initiatives and plans is presented in this section.

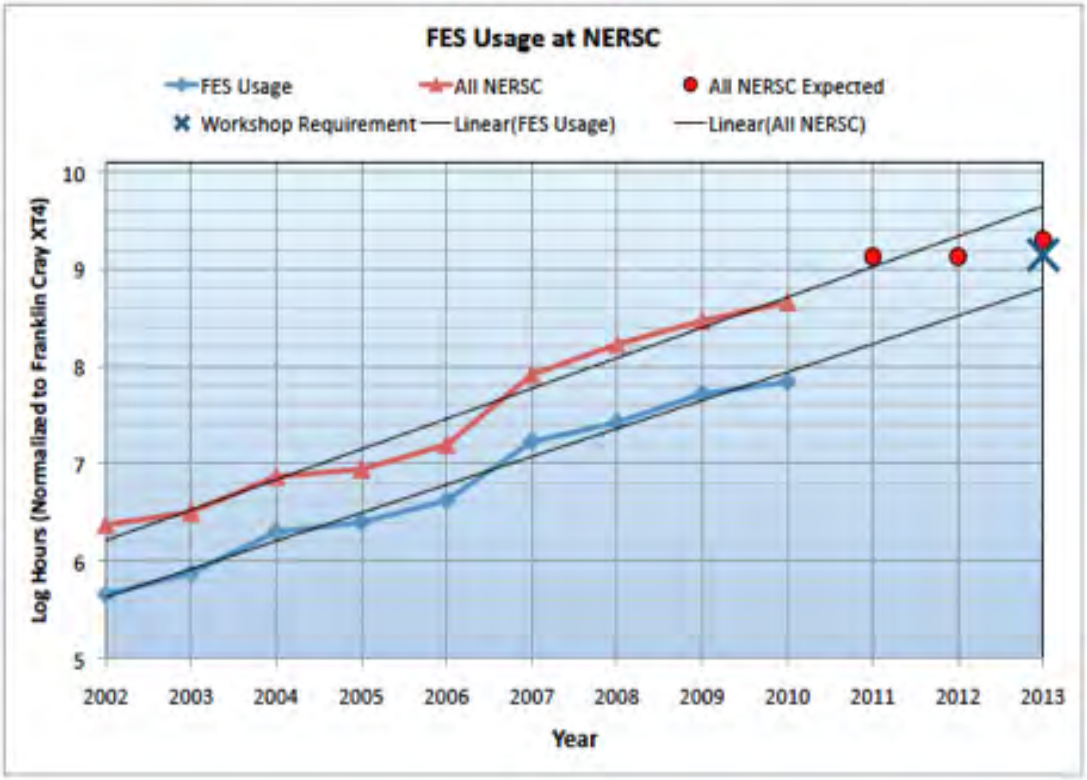
7.1 Compute Resources

The NERSC Hopper system, a Cray XE6 with 1.3 PF/s of peak performance and 120 TF/s performance on a set of representative applications, was installed shortly after this workshop. Hopper represented a 4-fold increase in aggregate application performance over the quad-core Franklin system that went into production in mid-2009. Hopper (NERSC-6) entered full production in May 2011. Total allocations in 2011, 2012, and 2013 are all expected to be about 1.1B hours (a factor of 3.5 over 2010) with no significant growth until AY2014 when we expect that NERSC-7 will be available by mid-year.

The NERSC-7 system will replace Franklin but between the decommissioning of Franklin and the installation of NERSC-7 there will be a gap in which only Hopper will be available as NERSC's large-scale resource. This gap may last at least six months to allow for the installation and acceptance of NERSC-7 and may be longer if there are substantial site preparation or budget pressures.

Current technology trends, along with the estimated NERSC funding profile, suggest that NERSC-7 will be only about two times the performance of Hopper, which would triple NERSC aggregate computing power. By 2014, with NERSC-7 expected to be in production in the second half of the year, NERSC expects a seven-fold increase in allocations over 2010, and in 2015, with NERSC-7 in production the entire year, an 11-fold increase over 2010. This is substantially less than the 40-fold increase needed for FES researchers to accomplish their research goals in 2013 and substantially below the historical growth in NERSC computing of 10x every four years. A 40-fold increase in computer resources for FES at NERSC is very unlikely. Meeting the 40-fold increase would require a larger budget and facility upgrades to support the new machines.

The figure below, showing the historical growth of FES and overall usage at NERSC, indicates that the need for computational hours to support FES in 2013 exceeds that expected by the historical trend (lower black line) and approaches the total number of hours expected at NERSC, based on current plans and funding profiles.



There are many examples of computing challenges in FES that require exascale computing resources and NERSC is preparing for systems of that scale. NERSC personnel have been active participants in DOE-wide and interagency discussions on exascale computer planning. Exascale systems are not expected until 2018 – 2020, but NERSC believes that many of the issues associated with both exascale hardware and software will appear in systems acquired well before formal exascale systems, possibly as early as NERSC-7. A promising technology on the road to exascale is the use of GPUs as accelerators to conventional CPUs. NERSC has a 48-node GPU testbed system and plans to continue exploring using GPUs in HPC.

Power is the most significant limitation to exascale computing and while future NERSC systems are expected to realize significant improvements in efficiency measured in delivered FLOPs per watt, there are significant hardware and software challenges on the path to exascale. In particular, the necessary power efficiency improvement will require new low-power processor architectures that are likely to be heterogeneous to satisfy the full workload and will likely affect programming. One possible path is a GPU-style approach that would be available in the NERSC-7 timeframe.

NERSC has provided users with the Dirac testbed for exploring this architectural path, and while there has been considerable activity in studying the applicability of GPU-based hardware solutions with impressive performance gains in some CPU-intensive FES kernels, our conclusion is that this technology and current tools are not appropriate for the broad workload. NERSC will be closely monitoring these and other low power

processor and memory technologies, communicating with vendors to help them understand the needs of the full workload, and working with the FES community to better understand ways of adapting their algorithms and software to future systems.

7.2 Software

NERSC recognizes the importance of providing scientific software libraries and is committed to encouraging vendors to install and support these popular libraries and ensure that they're optimized for the production platforms. Cray currently provides optimized versions of PETSc, Global Arrays, Trilinos, HDF5, netCDF, FFTW, and a large suite of mathematical libraries. The DOE ACTS tools group also provides optimized versions of important software and works with NERSC to provide user support when needed.

The HDF5 library is one of the most commonly used I/O libraries at NERSC and DOE. For this reason, NERSC partnered with the nonprofit Hierarchical Data Format (HDF) Group to optimize the performance of the HDF5 library on modern HPC platforms. This effort is continuing through UCSD. NERSC is also actively engaged with the HPC community in improving I/O performance. Efforts have been focused on MPI-IO, file caching/prefetching/aggregation, and other areas.

7.3 Data

NERSC plans to continue a constant investment in storage each year; at the planned budget levels this would result in a four-fold increase in disk capacity in the next four years. However, some FES research activities require at least a ten-fold increase in disk capacity in the next three to five years, and it is not likely that NERSC will have enough online storage to provide petabyte level resources to individual projects.

NERSC is also investing heavily in improving both capacity and bandwidth for the HPSS archival storage system. In 2011, NERSC added increased bandwidth to achieve 10 percent of the fastest file system's aggregate bandwidth. NERSC is also adding a tape library to increase its archival storage capacity. The NERSC HPSS system is designed to handle 50% growth per year in amount of IO and total data stored. The system handled over 4 PB of IO in 2010 and grew by 2.2 PB. Additional capacity is expected to grow by a factor of two each year for the next three to five years. This is in line with conventional bandwidth guidelines at other centers. NERSC is also working to significantly improve data movement between HPSS and NGF.

NERSC is also working closely with EsNet to implement the Advanced Networking Initiative-based 100Gb networks that will bring about significant improvement – 10X – in data movement capability. NERSC also supports GlobusOnline capability and a dedicated set of data transfer servers for rapid file transfers. For the foreseeable future (~five years) these capabilities are likely to represent the primary methods for data transfer between sites.

7.4 Job Turnaround and Throughput

NERSC staff and management continually monitor machine load, average wait times, and machine efficiency and set queue policies and machine resources to meet the needs of both interactive and batch demand. We will continue to do this in support of both capability and ensemble workflows. There are practical considerations that limit maximum job lengths set within the batch queues, such as machine MTBF and the need to occasionally remove the machine from service for maintenance. As machines stabilize we increase queue run time limits and reduce the NERSC impact on MTBI. We are also working on acquiring a technology that will allow NERSC staff to simulate queue structure and machine efficiency.

Several groups at NERSC are presently working to develop software tools to address the needs of complex scientific workflows that go beyond and/or do not require policy changes to HPC batch schedulers. This work is designed to improve usability for workloads that involve submitting, and require high throughput for, very large numbers of jobs to a highly parallel machine such as Hopper. The genesis for these tools is the rapidly growing genomics portion of the NERSC workload but there is no reason why the benefits of using these tools cannot accrue to more traditional supercomputing users such as FES as well. A few examples of the innovations we have developed include a task farmer, tools to create a private virtual cluster, and file caching utilities for data intensive applications. NERSC will communicate the details and availability of these tools to the broad NERSC user community via the usual channels such as the NERSC User Group.

NERSC is also having discussions with developers of the Berkeley Lab Checkpoint/Restart (BLCR) software to determine how this can be integrated with existing schedulers to allow system-level checkpoint/restart.

8 Plasma Turbulence & Transport

Contributors: C.S. Chang (NYU); Zhihong Lin (U.C. Irvine); Stephane Ethier (PPPL); Scott Parker, Yang Chen (U. Colorado Boulder); Jeff Candy (General Atomics); S. Ku (NYU), W. Wang (PPPL)

8.1 Plasma Turbulence and Transport Overview

In the absence of large and fast MHD-type instabilities, particle and plasma energy loss in tokamaks is largely driven by microturbulence that arises from plasma interactions with background "neoclassical" processes. Here, "neoclassical processes" refer to plasma transport resulting from the guiding-center orbital dynamics of the charged particles in an inhomogeneous toroidal magnetic field, including effects such as "banana" orbits, and Coulomb collisions. To achieve high fusion yield it is therefore critical to understand microturbulence and to control the transport losses it drives. Due to the inherently nonlinear and complex nature of turbulent transport in tokamak plasmas, well-coordinated strong computational programs are essential.

Micro turbulence in a strongly magnetized tokamak plasma is best studied by gyrokinetic simulations that solve first-principles gyrokinetic equations obtained after reducing the 6 dimensional Vlasov equation (3-D in position and velocity space) into 5-D (with 2-D in velocity space) by using the fact that the gyro motions are faster and of smaller scale than the corresponding microturbulence scales. Although reduction of the 6-D Vlasov system to 5-D gyrokinetic systems offers enormous savings in computing time, a full-scale 5-D gyrokinetic simulation of background and turbulence in realistic device geometry has historically been impossible even on the world's largest supercomputers. In order to decrease simulation time, a perturbative delta-*f* method was developed that evaluates only perturbed kinetic dynamics on a fixed background plasma assuming a closed thermodynamic system. The delta-*f* simulations have been used to obtain a significant understanding of turbulent transport physics over the last two decades and are still widely used in magnetic fusion studies due to their robustness and savings in computing time.

Recent high performance computers (HPCs) have made the original five-dimensional, full-function (full-*f*) gyrokinetic simulation possible, in which the self-organizing multiscale interactions of the kinetic background neoclassical plasma and the turbulence are simulated together. However, considerably more computing power is required to achieve a full 6-D Vlasov simulation in a tokamak plasma. Among the U.S. gyrokinetic codes to be described here, GYRO, GTC, GTS and GEM use delta-*f* methods, while the new XGC1 code uses full-*f* method. A full-*f* simulation capability is being developed for GTC and GTS. GYRO uses the TGYRO transport framework to evaluate the self-organizing evolution between the background fluid plasma and the turbulence.

There are two general techniques for solving the gyrokinetic equations. The first is the

Particle-in-Cell technique, in which a microscopic distribution function is statistically sampled by marker particles while the macroscopic physics is interpolated to the 3-D spatial grid. The elliptic electromagnetic field equations are then solved on the 3-D spatial grid, and the electromagnetic field is interpolated back to the particle positions to advance the marker particles to the next 5-D positions. The second technique uses a continuum formulation, in which both the microscopic and the macroscopic quantities are solved on 5-D position-velocity space grid. These two techniques are complementary in numerical technique, physics, and Cluster/Cloud/HPC usage. Most of the particle codes are compute intensive and need large numbers of particles to achieve a statistically meaningful number of particles distributed among many processor cores. The continuum technique is memory-intensive since the PDEs are solved on a high-dimensional (5-D) position-velocity grid. Among the U.S. gyrokinetic codes, GTC, XGC1, GTS, and GEM use the Particle-in-Cell technique, while GYRO is a continuum code. Each plays a different role in fusion research with some overlap in functionality to allow cross-verification.

In five years, these codes are expected to push the limit of available NERSC and OLCF computing on heterogeneous processor platforms, using either GPUs or other architectures. For some of the particle codes that already require extreme computing (XGC1, GTS, GTC) work is underway to aggressively modify data layouts to effectively use multicore and many-core systems and to cope with less memory per core. There is also research on using heterogeneous architectures, perhaps using GPUs, to achieve higher fidelity simulation in less wall-clock time. These codes use high-performance libraries such as ADIOS, which is quickly gaining in popularity for fast and efficient parallel I/O.

With computing hardware and software becoming more complex, a closer collaboration between physicists and applied mathematics, computer scientists, and performance engineers is becoming a necessity for fusion particle codes. Systematic and aggressive funding by DOE may be needed to make this happen, along the lines of the SciDAC approach, which includes all the necessary cross-disciplinary teams.

In the following sections, representative scientific study cases are briefly described for several important gyrokinetic codes, followed by a summary of current and future computational needs and a short concluding remark.

8.2 Plasma Turbulence Case Studies

8.2.1 The Center for Plasma Edge Simulation (CPES)

PI: C. S. Chang, Courant Institute of Mathematical Sciences, New York University
NERSC Repository m499

8.2.1.1 Summary and Scientific Objectives

Study of the edge plasma and its influence on the core plasma performance is one the highest priority items in ITER physics research. Core fusion and wall erosion are critically dependent on edge physics. Plasmas in the edge region of large tokamaks are in a multi-scale kinetic regime with complex geometry and large variations of collisionality. Difficult issues include (i) a large plasma pedestal pressure gradient in a trans-collisionality regime (weakly collisional in the pedestal top and collisional in the bottom); (ii) the magnetic separatrix and material wall; (iii) the importance of neoclassical physics (diffusion) together with the turbulence physics; (iv) the importance of the kinetic neutral penetration; and (v) the global edge localized mode crash of the plasma pedestal.

The SciDAC Center for Plasma Edge Simulation (CPES) (http://www.scidac.gov/FES/FES_CPES.html) has been established to address these scientific challenges. This SciDAC Fusion Simulation Prototype (FSP) program has the following key goals:

i. Develop large-scale massively parallel kinetic codes for higher fidelity simulation of the electromagnetic multi-scale edge physics in realistic tokamak edge geometries;

ii. Build an integrated predictive plasma edge simulation package applicable to existing magnetic fusion facilities and next-generation burning plasma experiments such as ITER; and

iii. Perform integrated simulations using kinetic, MHD, neutral particles, and atomic physics methods for higher fidelity understanding of the multi-scale edge physics and the wall heat load.

To fulfill goal (i.), above, we have been developing two new extreme-scale edge kinetic codes, XGC0 and XGC1. Until full development is complete we also use well-established electromagnetic core turbulence codes such as GEM and GTC in an edge-like pedestal plasma in a core geometry (without the magnetic separatrix) to study electromagnetic turbulence in a steep gradient plasma. Our project also has a code-coupling mission as a proto-FSP. For this reason we investigate coupling between the XGC0 code and GEM, GTC, Elite and M3D codes. This will enable us to perform the necessary coupled simulation for the ELM cycle (crash and reformation of edge pressure gradients due to Edge Localized Mode effects) and the electromagnetic turbulent transport.

This case study describes science and methods associated with the XGC suite of codes and GEM. During 2010 we estimated that roughly 80% of our NERSC ERCAP allocation would be for XGC0/XGC1 and roughly 5% (~.5M hours) would be for GEM.

Significant progress has already been obtained with XGC1 at NERSC. XGC1 has obtained, for the first time, tokamak edge turbulence simulation in realistic geometry with magnetic separatrix and material wall. This was a whole-core plasma simulation from the plasma center (magnetic axis) to the core-edge boundary to study the core-edge coupling effect on the edge and core turbulences. These simulations used realistic DIII-D tokamak geometry without scale separation between the background and turbulence dynamics, unlike what was usually done in other gyrokinetic codes that used perturbation method on a fixed background. This study showed that the pedestal must be treated globally, not locally and that there is rich turbulence physics compressed into the edge of a tokamak plasma, possibly non-locally coupled to the core turbulence dynamics. The experimentally observed inward momentum transport in the edge plasma at C-Mod and other tokamaks could also be related to this nonlocal edge ITG turbulence. In order to make the simulation more complete, electromagnetic edge turbulence simulation is planned in the near future.

We have also made significant progress performing coupled simulation with XGC0 and GEM at NERSC. We have found new phenomena in edge electromagnetic turbulence transport are needed to explain the experimental observations.

8.2.1.2 Methods of Solution

Our current primary codes are the XGC0 and XGC1 Particle-in-Cell codes. The full-function gyrokinetic particle code XGC1, the newest member of the U.S. gyrokinetic code family, is the flagship HPC code in CPES. It was developed in the Proto-FSP SciDAC Center for Plasma Edge Simulation with close collaboration among physicists, applied mathematicians, and computer scientists.

XGC1 is different from other gyrokinetic codes in that it handles the diverted edge magnetic geometry and the grounded (zero-potential) material wall boundary condition in a cylindrical coordinate system using an unstructured triangular grid. Other gyrokinetic codes are designed to handle only the core plasma inside the magnetic separatrix. A flux-coordinate system, in which the closed magnetic flux surface labels represent the minor radius of the torus, is used in those codes for highly efficient gyrokinetic simulation. However, a flux coordinate system becomes invalid near the magnetic separatrix surface. Thus, the core gyrokinetic codes using a flux coordinate system cannot approach the magnetic separatrix surface.

Due to the complicated edge plasma property and geometry, XGC1 has a few unusual features that demand more extreme scale computing than existing gyrokinetic codes written for core plasmas. These include a) the mixture of the cylindrical coordinate

system for particle pushing and the magnetic field-line aligned coordinate system for perturbed field solvers, b) the use of an unstructured triangular mesh system, and c) the utilization of the full function (full-*f*) particles for multiscale kinetic physics without scale separation, demanded by the importance of the orbit loss, neoclassical physics (an MHD diffusion effect) and its coupling to turbulence physics. Currently XGC1 includes full-*f* ions and adiabatic electrons for ITG (ion temperature gradient) turbulence, and both the full-*f* ions and full-*f* electrons for neoclassical physics in realistic diverter geometry.

Particle motion is described by a Lagrangian equation of motion in 3-D cylindrical coordinate system on realistic toroidal geometry with magnetic separatrix. Either Runge-Kutta or predictor-corrector methods are used. The field quantities are evaluated on grid mesh using linear multigrid PETSc solvers. The scale of the simulation is characterized by grid size, which then determines the number of particles. The parallelism is expressed by hybrid MPI/OpenMP.

In an effort to improve the electron kinetic physics in turbulence simulations, we have implemented two complementary models: the delta-*f* hybrid electron model (as is done at the University of California, Irvine) and the split-weight delta-*f* electron model (as is done at the University of Colorado). The next version of XGC1 will also include an electromagnetic turbulence capability using these kinetic electron models and a new electromagnetic gyrokinetic formalism for steep gradients.

XGC0 is a simplified version of XGC1 with reduced kinetics for experimental time-scale kinetic transport modeling. XGC0 performs the important pedestal buildup physics, 3-D magnetic field effect on pedestal dynamics, divertor heat load, neutral particle and atomic physics, and other functions.

GEM (Gyrokinetic ElectroMagnetic) is a comprehensive gyrokinetic delta-*f* Particle-In-Cell code developed for the study of core tokamak plasma microturbulence and associated transport by the Center for Integrated Plasma Studies at the University of Colorado. GEM uses generalized field-aligned magnetic coordinates parameterized by the Miller equilibrium model. Kinetic electrons and electromagnetic perturbations are included using the parallel canonical momentum formalism. An adjustable split-weight scheme is used to enhance the time step otherwise limited by the fast electron motion along the magnetic field. A finite-β Ampere solver is used to allow accurate simulation of Alfven waves or finite-β modification of Ion Temperature Gradient (ITG) modes. GEM is radially global, with multiple ion species, arbitrary density and temperature profiles and equilibrium radial electric field (E_r) profile. For long-time simulation, a Coarse-Graining Procedure (CGP) has been developed to control the secular growth of the mean square of the particle weights. Such secular growth of the particle weights is a necessary consequence of entropy production in a steady-state turbulence with non-zero transport, but can be suppressed using CGP without changing quantities of physical interest such as the particle and energy fluxes. For parallelization, GEM currently uses a 1-D domain decomposition along the field line direction, with domain cloning in the perpendicular plane. GEM has been extensively benchmarked both linearly and nonlinearly with the continuum codes GS2 and GYRO on ITG, TEM, and ETG and the

kinetic ballooning mode, e-i collisional effects and flux surface shape effects.

8.2.1.3 HPC Requirements

Plasmas across the magnetic separatrix surface are not in thermodynamic equilibrium due to intersection of the particle guiding center orbits with the material wall. Thus, unlike in other gyrokinetic codes, XGC1 must use the full-*f* gyrokinetic particles. This means that XGC1 must use large-scale computational systems to conduct a realistic device study. At the same time XGC1 has an advantage because the background and turbulence behaviors are simulated together, with the multiscale physics described by a single gyrokinetic set of equations, and because the particle noise does not grow in time. An XGC1 ion turbulence simulation in the whole volume DIII-D requires about 20 hours when run on about 120,000 processor cores (~1 Petaflop per sec), or equivalently two million processor hours, until a nonlinearly saturated quasi-steady turbulence state is achieved. A similar simulation for an ITER plasma in realistic geometry would take about 10 days. For this reason XGC1 simulations have been out of the reach of NERSC systems such as Franklin. (We have used NERSC computers for two FES 2010 Milestone activities in which several key discoveries were made resulting in journal and conference papers.)

Estimates of runs we would like to do on Hopper provide an indication of XGC science requirements. A single heroic ITG turbulence with full-*f* ion particles and adiabatic electrons for an ITER core-edge confinement study would require 200 hours on 100,000 Hopper cores (20M hours); two 20-hour XGC1 runs on 100,000 cores (4M hours) would be required for core-edge simulation of a DIII-D tokamak; two 20-hour XGC1 runs on 100,000 cores (4M hours) would also be required for core-edge simulation of NSTX tokamak; one 60-hour XGC1 run on 100,000 cores (8M hours) would be required for core-edge simulation of a C-Mod tokamak (which has a much smaller size grid than other existing devices); and one 120-hour run using 100,000 Hopper cores (12 M hours) is required for simulation of kinetic electron turbulence in a DIII-D plasma. This is 44 M hours needed in 2011 alone.

Obviously, scalability is an important aspect of XGC1. Excellent scalability to the maximal available number of cores on the present Jaguarpf (at the National Center for Computational Sciences) has been established in collaboration with the SciDAC PERI project. These studies also showed that a 6-way, hybrid MPI/OpenMP decomposition performs best. The MPI-only method does not not scale as well, and is never competitive when using 6,144 or more cores.

A current limitation we are trying to overcome is in the algorithm that enables the fully electromagnetic turbulence simulation within the 5-D gyrokinetic formalism. This is related to the time-step resolution of the fast electron motions, hence to the computing power. Presently, a 5-D ion full-*f* simulation of the DIII-D edge plasma requires 20 hours of simulation on 100,000 processor cores. Fluid-kinetic or split-weight electron simplification techniques can significantly reduce the computing requirement. The electromagnetic simulation of DIII-D edge plasma would require 200K processor cores of a Franklin-like machine to complete in one day. With more computing power, the

electrons can be simulated using a full-velocity function, instead of the simplified velocity space function. An ITER simulation will require about 1 million processor cores for a 20 hour run.

In the future our demand for computing time will increase in part because once kinetic electron turbulence capabilities become operable in the code then we'll need to take shorter time steps to follow the fast-moving electrons.

Resources for GEM are not shown in the table. It runs typically on 512-2048 processor cores, and uses about 1GB of memory per core and 100GB of total disk space, falling between GRYO and GTS for the parallelization requirement.

The most noticeable difference between now and 5 years hence is the number of cores used per run in particle codes, while the memory size per core must be reduced (see the XGC1 column). The checkpoint and read/write file size will increase significantly, demanding fast and efficient I/O. The off-line storage demand will go up accordingly.

8.2.1.4 Computational and Storage Requirements Summary (XGC1 Only)

	Current (2010)	**In 2013**
Computational Hours	8 M	300 M
Parallel Concurrency	512 - 223 K	1 M
Wall Hours per Run	1-5 days	1-5 days
Aggregate Memory	10-30 TB	100 TB
Memory per Core	0.3 GB	0.3 GB
Archival Storage	5 TB	50 TB

8.2.1.5 Support Services and Software

Since XGC1 is anticipated to be running at or very near the maximal capacity of the new machines over the next 5 years, efficient fault tolerance services will be needed. Of course, the ability of the batch system to efficiently handle very large jobs is required, too.

Libraries required include HDF5, Hypre, NETCDF, PAPI, PETSc, SuperLU, PSPLINE, and other software for high-speed file I/O. The GEM code has dependences on FFTW, LAPACK, and IDL.

8.2.1.6 Emerging HPC Architectures and Programming Models

We have been quite successful in adapting to the current level of multi-core architecture by using a hybrid MPI/OpenMP programming model. We find that using only OpenMP on each node is not the optimal solution on a 12-core Cray XT5. Instead, the best solution was to use two MPI processes per node with six OpenMP threads each. As the core

numbers increase per node, our low level strategy is to find the best mixture between OpenMP and MPI per node. At a higher level, our 3-5 years strategy is to develop asynchronous algorithms for effective utilization of many heterogeneous cores. A run-time scheduler such as StarPU will be used to coordinate and map threads to computational resources. Another approach is the incorporation of partitioned global address space (PGAS) languages to offer means for expressing locality of data. This approach has not been used in the XGC1 code yet since data localization is achieved using more conventional models, but it is under consideration for the future.

We are also looking into the GPUs. Sparse matrix-vector multiply has been demonstrated on GPGPU and is supported by an optimized library from NVDIA. Similarly, multi-grid has been demonstrated to have efficient implementation on GPGPU. These are expected to be useful for the XGC particle-in-cell code.

For GEM, no exotic programming models are currently used or expected in the near future.

8.2.2 Magnetic Fusion Plasma Microturbulence

NERSC Repository PI: Bruce Cohen, Lawrence Livermore National Laboratory
NERSC Repository gc3

Contributors: Jeff Candy, General Atomics

8.2.2.1 Summary and Scientific Objectives

This project provides general support for gyrokinetic simulation of plasma microturbulence, including the Eulerian codes GYRO and GS2 (USA), GENE (Germany) and the Particle-in-Cell codes GEM and PG3EQ (USA). This group of codes has produced the most ambitious turbulent-transport benchmarking exercise ever carried out in the community, and has a track record of extensive data sharing and collaboration.

This case study focuses on the GYRO code. On the five-year time horizon, we have two related goals with respect to GYRO modeling: first, carry out optimizations relevant to multi-scale simulations (simulations which simultaneously resolve ion space/time scales and electron space/time scales) so that GYRO can be *routinely* applied to these cases. Second, improve GYRO performance and its integration into TGYRO, to make profile-prediction a routine practice as well.

Here we give a brief summary of the computational requirements for a typical stand-alone GYRO analysis (no feedback via TGYRO) of a DIII-D plasma. These runs are taken from recent work (July 2010) from Chris Holland. Four cases were studied:

- 4 gyrotron source inner heating;
- 4 gyrotron source inner heating;
- 6 gyrotron source inner heating;
- 6 gyrotron source inner heating;
-

The global simulations had the following characteristics:

- 40 complex toroidal modes (spectral accuracy);
- 500 radial grid points (3rd order upwind derivative, spectral gyroaverage)
- 128 velocity space grid points (spectral accuracy);
- 10 poloidal arc points (3rd order upwind derivative);
- 3 kinetic species (deuterium, carbon, electron);
- transverse electromagnetic fluctuations;
- 76,800,000 total distribution function complex "unknowns";
- 240,000 total field complex "unknowns";
- $\Delta x / \rho_s = 0.3$, and $0 \leq k_\theta \rho_s \leq 2.5$

8.2.2.2 Methods of Solution

GYRO is a nonlinear tokamak microturbulence package designed to run on nearly all modern computing platforms, from an ultraportable laptop to the world's largest CRAY XT4/XT5 and IBM Blue Gene systems. Developed at General Atomics, GYRO uses a fixed (Eulerian) grid to solve the 5-D gyrokinetic-Maxwell equations. Operation is flexible, with the capability to treat the following:

- a local (flux-tube) or global radial domain, in a full or partial torus;
- general (shaped) or simple circular plasmas;
- adiabatic, drift-kinetic or gyrokinetic electrons;
- electrostatic, transverse and compressional electromagnetic fluctuations;
- experimental or user-defined physical parameters.

TGYRO is a parallel transport manager with the ability to call multiple instances of GYRO and the kinetic neoclassical[5] code NEO. The TGYRO code can include turbulent fluxes from GYRO, TGLF or the simple IFS-PPPL model and neoclassical fluxes from NEO, or from simple Hinton-Hazeltine theory.

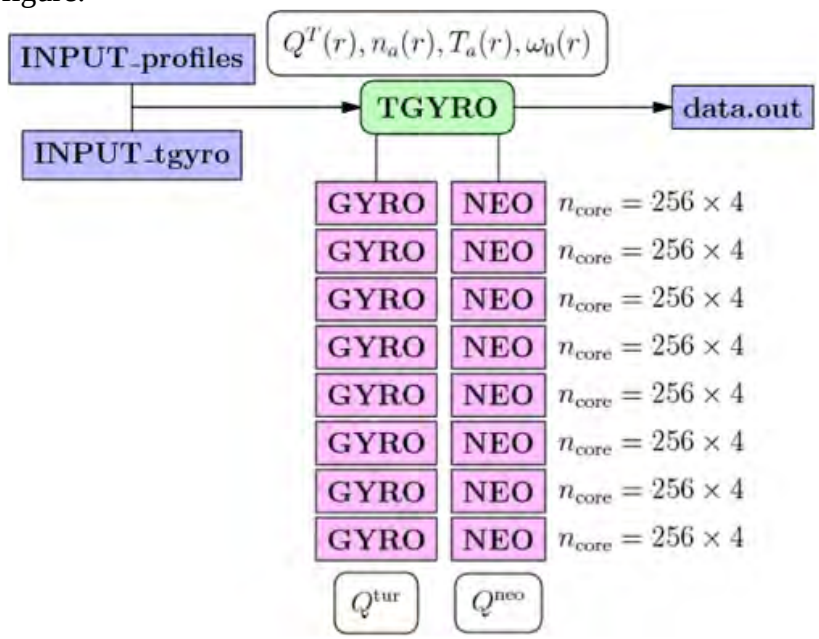
Temporal discretization in GYRO is described in Chapter 5 of the GYRO Technical Guide[6]. Briefly, if all species are gryokinetic explicit forth-order Runge-Kutta (RK4) is used and if electrons are drift-kinetic they are treated implicitly for improved efficiency, using a second-order IMEX-RK (implicit-explicit) method. The spatial discretization is described in Chapter 4 of the Technical Guide. Briefly, a mixture of finite-difference, finite-element, pseudo-spectral and spectral methods are used. Parallelization is accomplished by pure MPI code, although we are geared-up for a complete reworking of the parallelization scheme to target multi-scale wavenumber resolution on multi-core architectures. One feature of GYRO (and of Eulerian codes in general) that has been borne out by a decade of code results is that the grid resolution can be significantly less that in competing Particle-in-Cell codes due to high-order discretization methods and clever choice of coordinates. In GYRO, for a typical (not multi-scale) production run, we need about 180 radial gridpoints (5th order or higher method), 16 complex toroidal modes (spectral), 10 poloidal arc points per orbit (3rd order), and 128 velocity gridpoints (spectral). For multi-scale runs, the resolution requirements increase significantly with maximum resolved wavenumber and these challenging cases are becoming the norm for GYRO users.

A pictorial view of how GYRO, TGYRO, and NEO interact is given in the following

[5] Neoclassical refers to the transport, or diffusion, properties of charged particles (ions and electrons) in a tokamak magnetic field. Because of the toroidal geometry and the way the magnetic coils are wrapped around, the magnetic field is the strongest in the middle of the doughnut hole and decreases as you go away from that middle. Depending on the velocity of the charged particle, it can get "trapped" in the magnetic field and have a back-and-forth trajectory called a "banana" orbit. The width of the banana orbit, which has a significant radial extent, determines the diffusion properties in a "quiet" tokamak where there is no turbulence. This type of diffusion is called neoclassical transport.

[6] https://fusion.gat.com/THEORY/images/e/ea/Gyro_technical_guide.pdf

figure.



8.2.2.3 HPC Requirements

The GYRO study above required 178,600 MPP-hours on a Cray XT5. Post-run data storage requirements were in the range of 1-20 GB. Checkpoints were written to two alternating files, less than 1 GB each, approximately once/hour.

TGYRO manages hundreds to thousands of GYRO and NEO simulations in order to obtain a profile prediction given measured (in the case of post-experiment analysis) or computed (in the case of reactor studies) input power. For a typical TGYRO DIII-D case one might use eight radial simulation zones, with four simultaneous GYRO runs per zone (for a parallel Jacobian calculation), and 256 cores per GYRO instance. This uses 8 x 256 x 4 = 8192 cores in total. For the TGYRO study shown above, case, 9 simulation radii (9 instances of GYRO) were used. Depending on the number of iterations, the computing time can be as short as 6 hours but as GYRO wavenumber resolution increases, this time can increase to 24 hours or more. Currently, we do not store GYRO data for each iteration, so the data-storage requirements are not huge. For the eight-zone case, TGYRO requires about eight times the single-case GYRO storage amount: no more than about 8-32 GB for low-resolution simulations, although this can rise rapidly for simulations including higher wavenumbers. To summarize, typical numbers might be:

- 8 x 256 x 4 = 8,192 cores for 12 hours (low resolution);
- 10 x 512 x 5 = 25,600 cores for 48 hours (to resolve $k_y\rho_i > 1$);
- post-simulation storage: minimum 8GB, reaching 100GB or more for higher-resolution runs;

- checkpoint storage: minimum 2GB, reaching 16GB or more for higher-resolution runs.

8.2.2.4 Computational and Storage Requirements Summary

	Current (2010)	In 2013
Computational Hours	1.2 M	2.4 M
Parallel Concurrency	512	1,024
Wall Hours per Run	12	24
Aggregate Memory	0.5 TB	1 TB
Memory per Core	1 GB	2 GB
Online Storage	20 GB	40 GB
Archival Storage	100 GB	100 GB

8.2.2.5 Support Services and Software

The GYRO/NEO/TGYRO suite of codes uses FFTW, LAPACK, MUMPS, PETSc/SLEPc, NetCDF, python, and IDL, although much of the IDL routines are being rewritten to use Python and its associated scientific modules: SciPy, numPy, etc..

For the FACETS codes: Through the individual components, we depend on FFTW (TORIC), NetCDF4 (BOUT++), NetCDF3 (PlasmaState, TORIC), HDF5 (UEDGE, FACETS::Core), PETSc (UEDGE, FACETS::Core), Babel (UEDGE), BLAS/LAPACK. Also, PETSc in itself has several sub-dependencies. We routinely use SUNDIALS (a suite of nonlinear differential algebraic equation solvers), MUMPS, SuperLU, and HYPRE. Total workflow includes numPy, matplotlib, SciPy, tables, and VisIt.

Assistance with code profiling and performance tuning is more critical than ever. Our mathematical methods and overall code base are proven. However, significant re-optimization is needed now that architectures have evolved to multi-core and users are increasingly focusing on multi-scale simulations to more accurately capture the full electron energy transport physics. GYRO was heavily optimized for single-core and vector (Cray X1) systems, but we know that the performance of GYRO is not optimal on multi-core architectures, especially in this dramatically more challenging regime, and this is not specific to NERSC. The CSPM SciDAC project includes significant plans for performance analysis and re-optimization on multi-core and for multi- scale cases.

The evolution of HPC resources has not been in an optimal direction from our perspective. Centers like the OLCF have imposed a paradigm in which codes are supposed to run with more and more cores in a fixed amount of time (2-12 hours). However, in reality, increasing spatial resolution can bring in shorter timescales, so that even if codes scale well with increased spatial resolution, the total number of time steps, and thus total wallclock time, must increase. Thus, rather than very large core counts for the standard 2-12 hours, we need modestly increased core counts for significantly longer times (24-28 hours). Generally speaking, good science almost always requires large ensembles of runs (hundreds) that explore parameter space, never a few runs that approach the full machine size. So, as always, we think dedicated access with short wait-time to moderate core counts is the most useful service a computing center can provide. We find that compute capacity is more important (perhaps much more important) for the progress of science than ultimate compute capability.

8.2.2.6 Emerging HPC Architectures and Programming Models

Currently there is no use of GPUs and multicore optimizations are minimal for GYRO/NEO/TGYRO. However, over the next 5-year funding cycle our budget includes provisions for an intensive effort implementing multi-core optimization as well as GPU exploration. We are moving towards factorizing GYRO into discrete components (computational kernels) that can be optimized or otherwise rewritten on a per-architecture basis. We have avoided this in the past because of the difficulty of maintaining multiple sources, but in the future this may be unavoidable.

A separate NERSC allocation (m681) exists for the Fusion Application for Core-Edge Transport Simulations (FACETS) that also uses the GYRO (and GEM) codes. Because the framework uses very little overhead, we expect that most of the efforts, especially in the near term, will be in the exploration of the needs for the individual components.

8.2.3 Gyrokinetic Simulation of Energetic Particle Turbulence and Transport (The SciDAC GSEP Center)

PI: Zhihong Lin (UC Irvine)
NERSC Repository: m808

8.2.3.1 Summary and Scientific Objectives

The SciDAC GSEP project will further extend first-principles global gyrokinetic simulations to study new physics in energetic particle turbulence and transport. The ultimate goal is to build the predictive capability for energetic particle turbulence and transport in ITER burning plasmas through self-consistent gyrokinetic simulations of a full burst cycle of energetic particle turbulence. These simulations will give us the required understanding of the nonlinear physics of energetic particle instability and the ability to predict energetic particle transport given a fixed energetic particle drive.

The confinement of energetic particles is a critical issue for ITER because ignition relies on self-heating by the energetic α-particles. In this work, the diffusion of energetic particles by microscopic ion temperature gradient (ITG) turbulence is studied in large-scale simulations using the global gyrokinetic toroidal code (GTC). The radial particle diffusivity D is found to decrease drastically for high-energy particles due to the averaging effects of the large gyroradius and drift-orbit width, and the fast wave-particle decorrelation. Consistent with gyrokinetic theory, GTC simulations find that D scales with energy as $D \propto E^{-1}$ for energetic passing particle transport due to drift-orbit averaging and wave-particle decorrelation of parallel resonance, and $D \propto E^{-2}$ for trapped particle transport due to gyroaveraging, banana-orbit averaging and wave-particle decorrelation of drift-bounce resonance.

The gyrokinetic simulation and theory may have important implications for burning plasmas including ITER. GTC simulations suggest that the transport of energetic particles ($E >> 10Te$) is negligible and the transport of low energy α-particles ($E \leq 10Te$) is relatively strong, which are good for particle confinement and helium ash removal, respectively. These gyrokinetic simulations and theory also verify the conventional concept that energetic-particle transport is reduced by gyroaveraging, drift-orbit averaging, and wave-particle decorrelation.

In order to develop the predictive capability for assessing the effects of energetic particles on the performance of the burning plasmas, extensive validation of the GTC code has been performed using experimental data from existing tokamaks. As an initial step, results from GTC simulations have been used successfully to explain the transport of fast ions during neutral-beam injection experiments in the DIII-D tokamak for discharges dominated by ITG turbulence. For the neutrons and the neutral-beam current drive (NBCD) measurements, the discrepancy between classical theory (neglecting fast ion transport by microturbulence) and experimental data is eliminated when fast ion transport by microturbulence is considered. For the fast-ion Dα (FIDA) signal, the predicted

spectra and radial profiles still differ from experiment but the discrepancy is reduced. Thus, the expected transport by microturbulence is the correct order of magnitude to explain the observations.

8.2.3.2 Methods of Solution

The primary codes used are GTC and GYRO. Here, we describe GTC. GYRO has been described above in section 7.2.2.1.

In GTC simulations, the phase attribute of the fast gyration (or cyclotron) motion of the charged particles around the magnetic field lines is averaged away, reducing the dimensionality of the system from 6D to 5D. This gyrokinetic method removes the fast cyclotron motion, which has a much higher frequency than the characteristic waves of plasma microturbulence. The Particle-in-Cell method consists of moving particles along the characteristics of the gyrokinetic equation. The electrostatic potential and field are obtained by solving the Poisson equation on a spatial mesh after gathering the charge density on the grids. The electrostatic forces are subsequently scattered back to the particle positions for advancing the particle orbits. The use of spatial grids and the procedure of gyroaveraging reduce the intensity of small-scale fluctuations (particle noise). Particle collisions can be recovered as a "subgrid" phenomenon via Monte Carlo methods. The particle noise is further reduced using a perturbative simulation method where only the perturbed distribution function is calculated in simulation. Numerical properties of the electron dynamics are improved by an electrostatic fluid-kinetic hybrid electron mode based on an expansion of the electron response using the electron–ion mass ratio as a small parameter. Electron response is adiabatic in the lowest order and nonadiabatic response is taken into account in the higher order equations.

GTC employs the magnetic coordinates, which provide the most general coordinate system for any magnetic configuration possessing nested surfaces. General geometry with strong shaping has been implemented in GTC using a Poisson solver in real space and a spline fit of the equilibrium data from an MHD code such as EFIT. The property of straight field lines in the magnetic coordinates is most suitable for describing the instability with field aligned eigenmodes and enables the implementation of an efficient global field aligned mesh for the quasi-2D structure of the plasma turbulence in the toroidal geometry. The global field-aligned mesh provides the highest possible computational efficiency without any simplification in terms of physics models or simulation geometry. The magnetic coordinates are also desirable for efficiently integrating the particle orbits, which move predominantly along the magnetic field line direction. The equation of motion can be derived from a Hamiltonian formulation that conserves phase space volume and is best for integrating particle orbits for a long period.

8.2.3.3 HPC Requirements

We need to further extend the first-principles global simulations to study new physics in energetic particle turbulence and transport, including the micro-scale kinetic effects of

thermal particles on the meso-scale energetic particle instability, the nonlinear saturation of meso-scale Alfven waves and the transport of energetic particles, and the coupling between meso-scale and micro-scale turbulence. Our ultimate goal is to build the predictive capability for energetic particle turbulence and transport in the ITER burning plasmas, which requires understanding nonlinear physics of energetic particle instability, predicting energetic particle transport given a fixed energetic particle drive, and self-consistent gyrokinetic simulations of a full burst cycle of the energetic particle turbulence. We need to study the nonlinear physics of the meso-scale AE/EPMs with a transition from a single coherent mode to many interacting modes, and the regime where thermal plasma nonlinear response is competing on the same footing with energetic particle nonlinear dynamics, relevant near marginal stability. We will also study the influence of the three dimensional equilibrium on the energetic particle instability and transport, and investigate the low-n asymptotic limits of the nonlinear gyrokinetic equation and derive the corresponding set of Maxwell's equations valid over a broad range of toroidal mode numbers, *n*.

A new direction will be the construction of a first-principles energetic particle transport model including quasilinear energetic particle profile relaxation and nonlinear saturation of meso-scale energetic particle instabilities interacting with the micro-scale turbulence. To ensure timely deployment of critical computational tools developed in various SciDAC projects, computational partnership with applied mathematicians and computational scientists has been built during the current GSEP project and will be enhanced in the proposed project in the area of statistical data analysis, end-to-end system, and optimization and parallelization on existing petascale computers and the emerging exascale computer.

With the remarkable progress in first-principles simulations, verification and validation (V&V) are the next critical steps for building a predictive capability of energetic particle confinement in ITER. The V&V will continue the ongoing benchmarks among global codes GTC, GYRO, HMGC, and TAEFL. Full-physics simulations will be required for comparison with experimental measurements through synthetic diagnostics including eigenmode properties, instability threshold, spectral intensity, density and temperature fluctuations, and energetic particle transport in real and velocity space. Experimental measurements have been carried out on DIII-D experiments dedicated to GSEP validation and more are planned.

8.2.3.4 Computational and Storage Requirements Summary

	Current (2010)	In 2013
Computational Hours	1.5 M	100 M
Parallel Concurrency	3k – 40k	200 K
Wall Hours per Run	20	20
Aggregate Memory	5 – 50 TB	100 TB
Memory per Core	1 GB	0.5 GB
I/O per Run Needed	1 TB – 10 TB	20 TB
On-Line Storage Needed	1 TB, 100 files	20 TB
Data Transfer Needed	1 GB / month	10 GB /month

8.2.3.5 Support Services and Software

When GTC uses hundreds of thousands of cores, having each node create an individual netCDF restart file causes unacceptable delay in the file system due to the large number of simultaneous file creation requests. To remove this bottleneck, the IO was rewritten to use HDF-5 collectives, and an abstraction layer called ADIOS (ADaptable IO System) [Lofstead2007] was developed and implemented in GTC. ADIOS provides a simple API that can select automatically the best techniques for each different grouping of data as specified by an entry in an external XML configuration file without touching the science part of the code.

GTC uses the PETSc library for its linear solvers. A hybrid MPI/OpenMP PETSc would be very useful for GTC.

8.2.3.6 Emerging HPC Architectures and Programming Models

GTC employs three levels of parallelism. The original parallel scheme implemented in GTC was a 1-D domain decomposition in the symmetric, toroidal direction (the long way around the torus) using the Message Passing Interface (MPI). Each MPI process was in charge of a toroidal domain with both particles and fields. Particles moved from one domain to another while they traveled around the torus. All communications was one-way traffic to avoid congestion. A second level of parallelism has now been implemented to increase the concurrency. Within each toroidal domain, we now divide the particles between several MPI processes, but each process keeps a copy of all the fields on a single toroidal plane. A “particle-domain” communicator links the MPI processes within a toroidal domain of the original 1D domain decomposition, while a “toroidal-domain” communicator links in a ring-like fashion all the MPI processes with the same intra-domain rank. To take advantage of the shared memory capability of multi-core nodes, a third level of parallelism is also implemented at the loop level using OpenMP compiler directives. These three levels of parallelism using mixed-mode MPI-OpenMP enable

GTC to scale to very large number of processors and use a very large number of particles, which results in a very high phase space resolution and a low statistical noise. The weak scaling of the GTC computing power is almost a linear function of the number of cores up to 100,000 of cores on Cray XT5 supercomputer. GTC is portable and optimized for various scalar and vector supercomputers.

An effort also currently exists for modifying GTC for GPU acceleration.

8.2.4 Global Gyrokinetic PIC Simulations of Plasma Microturbulence

NERSC PI: Wei-li Lee (Princeton Plasma Physics Laboratory)
Contributors: Stephane Ethier, Weixing Wang (PPPL)
NERSC Repository: mp19

8.2.4.1 Summary and Scientific Objectives

We use global, gyrokinetic Particle-in-Cell simulations to study all aspects of plasma micro-turbulence in the core of tokamak fusion devices. Our highly scalable GTS code takes as input the parameters of real experiments to carry out self-consistent simulations of particles, energy, and momentum transport due to micro-turbulence. One of our objectives is to continue our on-going study of momentum transport under different conditions and for several existing tokamaks. This study is particularly relevant to ITER so we plan on carrying out predictive simulations of ITER to determine its capacity to generate intrinsic rotation. This will include a study of the impact of kinetic electrons, mainly though trapped-electron modes.

If ready, we will also start the study of finite-beta (ratio of plasma pressure to magnetic pressure) effects in current tokamaks, mainly in NSTX where these effects are believed to be very important.

An example of the applicability of GTS is in the study of toroidal momentum generation and transport, which was carried out on 8k-100k processor cores. It has been known in Alcator C-Mod and other experiments that the plasma can self-generate toroidal rotation with an explicit external momentum input. Toroidal plasma rotation is known to improve stability and confinement of a tokamak plasma, and is important for ITER, which cannot have much external momentum source. Understanding this phenomenon requires a complex numerical simulation that must include anomalous, non-diffusive and non-local momentum transport. The figure below shows a GTS finding that for collisionless trapped electron mode (CTEM) turbulence, the Nonlinear Residual Stress (left) can drive the toroidal momentum (right) efficiently in the whole turbulence region via momentum transfer from waves to particles. The magnitude of the generated momentum is about 5% of the ion thermal speed at the end of the simulation, in the co-current direction, consistent with the experimentally observed direction.

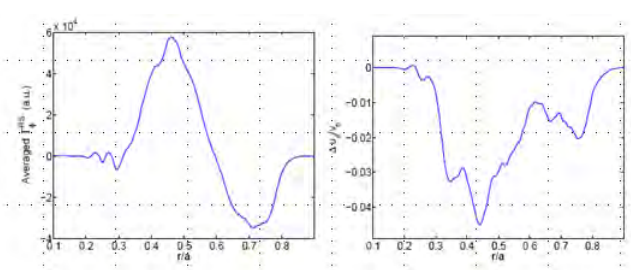


Figure 2 Residual stress from CTEM turbulence (left) and the resulting momentum generation (right)

CTEM turbulence also reproduces the experimental flow pinch effect found by Yoshida, et al in 2008 (see the figure below). This simulation illuminates the underlying physical dynamics governing the radial penetration of modulated flows observed in the experiment.

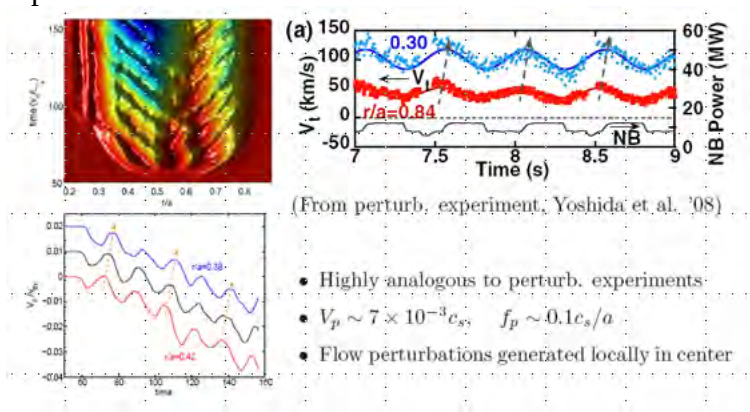


Figure 1 Flow pinch phenomenon found in CTEM turbulence reproduces experiments

8.2.4.2 Methods of Solution

Our production code, GTS (Gyrokinetic Tokamak Simulation), uses the highly scalable Particle-in-Cell (PIC) method in which simulation particles are moved along the characteristics in phase space. This reduces the complex gyro-averaged Vlasov equation, a 5-dimensional partial differential equation, to a simple system of ordinary differential equations. Straight-field-line magnetic coordinates in toroidal geometry are employed since they are the natural coordinates for describing the complex tokamak magnetic equilibrium field and lead to very accurate time-stepping -- even when a relatively low order method, such as second-order Runge-Kutta, is employed. In the PIC method, a grid replaces the direct binary interaction between particles by accumulating the charge of those particles on the grid at every time step and solving for the electromagnetic field, which is then gathered back to the particles' positions. The associated grid is built according to the profiles determined from the experimental data of the tokamak shots under investigation. This ensures a uniform coverage of phase space in terms of resolution. The grid-based field is solved using the PETSc parallel solver library. With the combination of inner and outer iterations, and the fast multi-grid solver in PETSc, the gyrokinetic Poisson equation of integral form, which contains multi-temporal and multi-spatial scale dynamics, is solved accurately in real space. Fully kinetic electron physics is included using very few approximations while achieving reduced noise. A fully conserving (energy and momentum) Fokker-Planck collision operator is implemented in the code using a Monte Carlo algorithm.

The GTS code is a full geometry, delta-*f* PIC code [1]. It is based on a generalized gyrokinetic simulation model and the use of realistic magnetic configurations. The GTS code is targeted at simulating the core plasma turbulence and transport in practical fusion experiments. It is highly robust at treating globally consistent, shaped cross-section tokamaks inside the magnetic separatrix surface by directly importing plasma profiles of

temperature, density and rotation, from the TRANSP [2] experimental database, along with the related numerical MHD equilibria reconstructed by MHD codes.

GTS employs straight-field-line magnetic coordinates and a field-line-following mesh. The particle guiding center motion is calculated by Lagrangian equations [3,4] in the flux coordinates, which allows for very accurate particle orbit integration even with a relatively low order method due to the separation between fast parallel motion and slow perpendicular drifts. The field-line-following mesh can best represent the nature of the quasi-2D mode structures of drift wave turbulence in toroidal systems, and hence offers the most efficient spatial resolution for highly anisotropic fluctuations.

The GTS code solves the gyrokinetic Poisson equation in configuration space for the turbulence potential at the particle coordinates, using the field line-following mesh. Unlike for flux-tube or wedge simulations, the real space, global Poisson solver, in principle, retains all toroidal modes from (m/n = 0/0) all the way up to a limit that is set by grid resolution, and therefore retains full-channel nonlinear energy couplings. A proper treatment of the nonlinear toroidal mode couplings is essential. With the combination of inner and outer iterations, a generalized field solver is developed to approach the gyrokinetic Poisson equation in integral form [5].

Fully-kinetic electron physics is included in GTS in order to simulate electron turbulence and ion turbulence with non-adiabatic electron physics [5,6]. For ITG and TEM turbulence with k_perp\rho_e<<1, we use a drift kinetic description for electrons, neglecting the finite gyroradius effects. However, for electron gyroradius scale turbulence, such as ETG driven turbulence, electrons are treated as fully gyrokinetic. Kinetic electrons are treated via two different approaches which use minimum approximations while achieving reduced noise: a) a delta-*f* method which solves for the total perturbed electron distribution function corresponding to turbulence fluctuations, and thus treats both adiabatic and non-adiabatic electrons kinetically; b) a split weight method [7] which simulates only non-adiabatic electrons kinetically. One highlighted feature distinct from many other gyrokinetic simulations is that both trapped and untrapped electrons are included in the non-adiabatic response.

Coulomb collisions between like particles are implemented via a linearized Fokker-Plank operator with particle, momentum and energy conservation. Electron-ion collisions are simulated by the Lorentz operator.

GTS is coupled with neoclassical physics via an interface with the GTC-NEO code [8] that calculates the neoclassical equilibrium, transport and electric field with finite orbit effects[7].

[7] Neoclassical refers to the transport, or diffusion, properties of charged particles (ions and electrons) in a tokamak magnetic field. Because of the toroidal geometry and the way the magnetic coils are wrapped around, the magnetic field is the strongest in the middle of the doughnut hole and decreases as you go away from that middle. Depending on the velocity of the charged particle, it can get "trapped" in the magnetic field and have a back-and-forth trajectory called a "banana" orbit. The width of the banana orbit, which has a significant radial extent, determines the diffusion properties in a "quiet" tokamak where there is no turbulence. This type of diffusion is called neoclassical transport.

GTS has 3 levels of parallelism: a one-dimensional domain decomposition in the toroidal direction (long way around the torus), dividing both grid and particles, a particle distribution within each domain, which further partitions the particles among processors, and a loop-level multi-threading method, which can be use to further divide the computational work within a multi-core node. The domain decomposition and particle distribution are implemented with MPI, while the loop-level multi-threading is implemented with OpenMP directives. The latter is very useful to reduce bandwidth contention among multiple cores within a node, which could be important on Hopper. Overall, the three levels of parallelism make for a highly scalable code that can run on hundreds of thousand of processor cores.

The simulation size is determined primarily by the number of particles, although the number of grid points can also be important when simulating large devices such as ITER.

8.2.4.3 HPC Requirements

The implementation of a fully electromagnetic model in the code will put more emphasis on the multi-grid solver. We expect the time spent in the solver to increase fivefold compared to the current code running a simulation with the same number of grid points. The time spent in the charge deposition and gather-push phase will double in each case since we will now deposit the electric current as well as the charge, and gather the magnetic force as well as the electrostatic force.

We also plan to run a full-f version of GTS, where the particles will describe the complete particle distribution function. The number of particles will need to increase by about 100 times, making those simulations very expensive.

Impurity species will also be added to the code, further increasing the computational cost.

We intend to implement a new dimension of domain decomposition in GTS in order to decrease the memory requirements per MPI process and improve strong scaling.

A low interconnect latency and fast gather-scatter to memory are the most important hardware features for our 3-D PIC codes.

8.2.4.4 Computational and Storage Requirements Summary

	Current (2010)	In 2013
Computational Hours	13 M	130 M
Parallel Concurrency	8K – 100K	32 – 130K
Wall Hours per Run	72	72
Aggregate Memory	16TB – 100 TB	32 – 160TB
Memory per Core	1 GB	1 GB
I/O per Run Needed	2.5 TB	8 TB
On-Line Storage Needed	4 TB / 10 K files	8 TB / 10 K files
Data Transfer Needed	5 GB / run	10 GB / run

8.2.4.5 Support Services and Software

Both GTC and GTS make use of freely available scientific software, such as the PETSc library for the solver part of the codes, and the ADIOS high-performance I/O library for data output. The PETSc library has given us some problems for time to time, mainly in the way it interacts with other libraries. This has to do mainly with the installation procedure. Additionally, GTS makes use of the SPRNG 2.0 portable random number generator, the PSPLINE fast splines library, and the CMLIB mathematical library, which replaced the NAG routines used in the code when we moved to Franklin. We rely on NERSC user services to maintain working versions of all these software libraries. Debugging and optimization tools (DDT, CrayPat, IPM), as well as analysis and visualization software (MATLAB, VisIt), are also very important. We currently do not expect the software required by GTS/GTC to change much during the next five years, at least not in the production versions of the codes.

In terms of support services, current NERSC services are very well attuned to users' needs. HPC consulting, software maintenance, training (by web documentation and workshops), and visualization support, are all important for ensuring a smooth computing work environment. Training and optimization advices from the HPC consulting group will be particularly critical this year and during the next five years as we transition to the new Cray XE6 system and other highly multi-threaded architectures.

8.2.4.6 Emerging HPC Architectures and Programming Models

We have several ongoing projects with NERSC and Berkeley Lab computer scientists looking into potential optimizations for our gyrokinetic PIC codes on emerging HPC architectures. For example, an ARRA-funded NERSC postdoc is looking into the advantages of using Fortran co-arrays in GTS to speed up the particle communications between processes on the Cray XE6. We also have a collaboration with the Future Technologies Group of the Computer Research Division at LBNL to look into optimizations of the gather-scatter operations in GTC/GTS on multi-core architectures and GPUs. We believe that it is important that NERSC continue to allow the consulting staff to participate in and supervise such research activities on advanced architectures prior to its mainstream adoption. Our group understands the importance of continuously

improving our codes in order to take full advantage of the current and future computing systems at NERSC.

9 Magnetohydrodynamics

9.1 Magnetohydrodynamics Modeling and Simulation Overview

While the gyrokinetic equations and associated codes used in the study of plasma turbulence and transport, as discussed in Section 7, represent a near first-principles approach to describing the properties of magnetically confined plasmas, solving these equations in the temporal and spatial scales characterizing large scale instabilities is not practical. A reduced dimensionality approach, based on velocity moments of the kinetic equations coupled to the Maxwell equations and often referred to as the magnetohydrodynamics or MHD approach, is generally used to study the macroscopic (device-scale) equilibrium and stability properties of magnetically confined systems. Simulation models range from ideal MHD, which neglects dissipative effects, to more comprehensive higher physics-fidelity models, collectively known as "extended MHD", which are based on two-fluid equations and implement advanced closures.

9.1.1 Extended Magnetohydrodynamics Simulation of Tokamak Plasmas

PI: Stephen C. Jardin (PPPL)
Contributors: J. Breslau, J. Chen, G. Fu (PPPL), N Ferraro (General Atomics), S. Kruger (TechX), D. Schnack, C. Sovinec (U. Wisconsin), H. Strauss (HRS Fusion), L. Sugiyama (MIT)
NERSC Repositories: mp288 (mp200, m912, m681, m224, m499)

9.1.1.1 Summary and Scientific Objectives

Our codes solve the extended magnetohydrodynamics (MHD) equations in magnetic fusion confinement geometries[8]. The primary objective is to study device-scale instabilities in the tokamak. These include sawtooth oscillations, neoclassical tearing modes (NTMs) and the interaction of coupled island chains, edge localized modes (ELMs), disruption forces, runaway electrons and heat loads during a disruption, mechanisms for causing plasma disruptions, mass redistribution after pellet injection, and the interaction of high-energy particles with global MHD modes. The goal is to validate our simulations with existing tokamak experiments and to apply them to make predictions for ITER. The validated models will be extended to propose control techniques that can be further tested on today's experiments and used in ITER.

[8] see Jardin, S. C., Computational Methods in Plasma Physics. Taylor and Francis (Boca Raton), 2010.

9.1.1.2 Methods of Solution

The primary codes used in our center are NIMROD, M3D, and M3D-C1, an extension of M3D that uses high-order finite elements and is fully implicit. These codes are each in the range 200,000 – 300,000 lines of code using a combination of FORTRAN and C. All the codes are implicit to some degree. They differ in whether they solve a single (big) matrix equation or several smaller equations at each time step. They have the option of employing a linearly implicit method, a partially implicit method, or of using Newton-Krylov techniques to solve the non-linear implicit equations. All of these result in large, sparse matrix equations. Typically, we have on the order of 10^7 to 10^8 degrees of freedom (DOF) per problem today. This results from having 8-9 variables per element (or mesh point) for conventional MHD, 104-105 element DOF per variable per toroidal plane, and typically 102 toroidal planes (or half as many Fourier modes). Additional variables appear in the extended MHD models.

Certain theoretical questions about the extended MHD solutions may influence the simulation methods and HPC requirements in the next five years. The codes presently have the capability to begin to investigate MHD turbulence in existing fusion plasma experiments, e.g., 40 or more toroidal harmonics and radial grid sizes somewhat smaller than the average ion gyro-radius, although full turbulence resolution of large plasmas such as ITER require additional capabilities. Certain extensions to the MHD model that are of interest to fusion plasma dynamics will enhance turbulent effects, such as allowing an anisotropic temperature along and across the magnetic field instead of a scalar temperature (pressure). The limits on the smoothness of turbulent solutions are not understood for standard fluids, while MHD, where the magnetic field is also chaotic and the plasma current density plays the role of a second vorticity, is even more complex. It is likely that HPC capability and policies, such as batch queue wait times as a function of job size and length will influence the direction of code development in this area.

9.1.1.3 HPC Requirements

We need the ability to process significantly larger data sets with tightly coupled parallelization so that we can increase the resolution and physics content in our calculations. A factor of two greater resolution in each direction results in almost an order of magnitude increase in the number of processors required, but the running time will stay the same or increase somewhat. The next generation ITER fusion burning experiment will be a factor of five larger in terms of ion gyro-radius per toroidal radius, leading to a minimum factor 53 increase in job size. Most of the computational time comes from data movement and implicit solves, so that a low-latency high-bandwidth system is required. We find that we are able to make good use of approximately a factor of two increase in computational capability every year, and expect this trend to continue for at least the next five years.

In addition to the fluid-like equations, we are increasingly solving some kinetic (particle-based) equations to close our set of equations. This is because a high-temperature magnetized plasma does not generally have a Maxwellian distribution function, and the

difference between the actual distribution function and a Maxwellian can be important for some applications. We have several efforts aimed at representing the non-Maxwellian part of the distribution function as either particles or as continuum phase-space distribution functions. These closures are very compute intensive, but they also tend to improve the parallel scaling of the calculation. In addition, we are involved in an ongoing collaboration to directly couple large-scale particle codes to the MHD codes on time-step scales, through the SciDAC CPES (Center for Plasma Edge Simulation).

Most scientific runs will be carried out at medium, rather than maximum, size. The capability to run medium size jobs (several hundred to several thousand processors) for long wall clock times (100 to several hundred hours) is necessary to study extended MHD plasma phenomena. A typical M3D nonlinear edge instability simulation last year used 432 processors for 200 hours on Franklin. Longer runs and runs at 2-3 times the size should be carried out, but are currently difficult due to long wall clock turnaround. Long jobs are best controlled through multiple short runs, so good batch queue turnaround is important. (In general, a one-day turnaround for a six–eight-hour job works well.) Alternatively, methods of job monitoring that also allow user control of running jobs would work. These may be worth developing, in particular, for jobs run on multi-core architectures where data communication to outside files may be less efficient.

The ability to run small jobs using the large simulation codes and smaller supporting codes is important for analyzing and developing the codes. Several smaller codes or smaller runs are used to develop the optimal mesh size, packing, and shape for a large run, as well as to translate experimental data or other input equilibria for the large codes. Support by NERSC for multiple related jobs run simultaneously in batch as a set would greatly improve efficiency and scientific productivity. In general, scientific studies require parameter scans of at least 3 parameter values, while most plasma problems contain multiple unknown parameters, each of which should be scanned. This is useful for smaller short jobs (e.g., to calculate linear growth rates for each of a set of toroidal harmonics) or for larger, longer nonlinear simulations (e.g., to compare MHD and two-fluid effects). The ability to run medium size jobs is critical to successful scaling to large size nonlinear simulations.

9.1.1.4 Computational and Storage Requirements Summary

	Current (2010)	**In 2013**
Computational Hours	8 M	64 M – 256 M
Parallel Concurrency	500	4 K – 10 K
Wall Hours per Run	100-1000	100-1000
Aggregate Memory	250 GB	4 TB – 16 TB
Memory per Core	0.5 GB	0.5 GB
I/O per Run Needed	30 GB	240 GB – 1 TB
On-Line Storage Needed	30 GB	240 GB – 1 TB
Data Transfer Needed	1 GB/day	8 – 32 GB / day

9.1.1.5 Support Services and Software

We make extensive use of third-party software such as PETSc for solvers and the SCOREC routines from RPI for grid and adaptation software. We also use HDF5 for some output in M3D. It is imperative that the routines from these libraries are compatible and do not require excess data-movement when used together.

Visualization of time-dependent data, including 3-D analysis, is crucial to understanding the simulations. We rely on visualization support from NERSC, using AVS/Express and VisIt. Simulation sizes are already becoming large enough that it is difficult to move data to local computers for analysis. The size of simulation data sets is beginning to challenge the existing dedicated visualization computers (e.g., Euclid), and the analysis software on the parallel systems, such as Franklin, is not as up to date.

9.1.1.6 Emerging HPC Architectures and Programming Models

Since we rely on PETSc for the linear solvers, it is essential that PETSc continue to have optimized solvers on new HPC architectures.

The high order C^1 continuous (i.e., continuous first derivatives) basis functions used in M3D-C^1 require on the order of 10^6 multiplications to perform a volume integral over a single element. Since this is all operating on local data, it may be possible to use GPUs effectively in performing these integrations. This is something that we plan to explore.

The M3D code also has an active OpenMP version that is used for development work on the MPI code and for smaller scientific runs. The Fortran routines encode the physics (the terms in the MHD equations) and call a set of C subroutines for the differential operators, global operations such as max, min, and volume integrals, and the Poisson-type matrix solvers for the quasi-implicit terms. The subroutines can be overloaded according to the choice of MPI PETSc or OpenMP method. This structure could be used to develop a hybrid MPI/OpenMP code for multi-core computation.

10 Fusion Simulation Program

10.1 The Fusion Simulation Program

PI: William Tang (PPPL)
Contributors: Doug McCune (PPPL); Martin Greenwald (MIT)
NERSC Repository: None

10.1.1 Summary and Scientific Objectives

The Fusion Simulation Program (FSP) mission will be to provide predictive capability for the behavior of magnetic confinement devices via science-based simulations of nonlinear, coupled phenomena on time and space scales required for fusion energy production. This will require multi-scale, multi-physics integration well beyond current capabilities. The mission will be accomplished through improvements and innovation in physics formulation, numerics and algorithms along with the use of increasingly powerful computer architectures. A rigorous verification and validation program will be an integral part of the FSP, requiring significant computational resources on its own. Productions services, with a large user base are also planned. The FSP is currently in the middle of a two-year planning exercise whose main goal is to define a compelling and detailed program plan. The results of the planning study will be evaluated by FES in FY2012. If fully funded, the FSP will roughly double the scope of computing in the MFE program.

10.1.1.1 Methods of Solution

Future Integrated Modeling Tools will target realistic simulations of fusion and energy systems with unprecedented physics fidelity. This involves delivering shorter-term opportunistic HPC software tools (built largely from modestly improved existing tools); and a longer-term development emphasizing new, more rigorous, more engineered .codes with high-performance capabilities

10.1.1.2 HPC Requirements

Simulations

FSP is envisioned to roughly double the scale and scope of the current Magnetic Fusion Energy computational program. We envision needing

- 10s of large jobs using in aggregate >1M cores (~200 M hours)
- 100s of medium scale runs using 10,000s of cores (~ 20 M hours)
- 10,000s of small runs using 1000s of cores (~200 M Hours)
- Memory requirements from 0.1 GB/core for largest jobs to 2 GB/core for small and medium runs

Applied to NERSC allocations, all MFE researchers used 68 M hours in 2010. A doubling of that number yields an estimate of 136 M hours needed to support FSP at NERSC in 2013.

As with computation we can only make rough estimates of storage requirements at this point.

- Aggregate archival storage is likely to be in the multi-PB range in 1000s to 10,000s of files per year
- Temporary storage needed by jobs during runs is also predicted to go into the PB range
- As noted in the previous slide, we are planning to catalog all FSP runs across all platforms regardless of physical location
- UAL (universal access layer) planned for location independent data access

Data Analysis

In the five-year time frame ITER data analysis will pose some serious computational, networking, and data challenges. Fusion experiments demand real-time data processing. ITER anticipates collecting ~106 named data items with kHz to MHz sample rates and will require a between-pulse analysis. This is not a batch analysis; rapid near-real-time analysis of many measurements is needed and future large data challenges are going to be formidable. This analysis will enable better informed decisions, which will result in better experiment planning.

Current estimates of data size is roughly 40 TB per shot for long-pulse shots of 400 seconds, which would demand 100 GB/sec bandwidth from France to the U.S. We will likely need to be able to parallelize at least a significant fraction of this data for streaming. Current estimates of the time between shots is roughly 1600 seconds -- a rather limited period of time. I/O will be very stressed for (i) reading even a fraction of this amount of data from memory into CPUs and then writing back to disk and (ii) displaying the information. Realistic development of such capabilities is a major challenge.

We may be facing a major change in the way we deal with data, perhaps moving from the current “data file paradigm” to a “data streaming paradigm” to accommodate much larger data sets. This is analogous to looking at various frames of a movie while the movie is still being generated. Advanced image processing capabilities could enable end-users/physicists to examine/analyze information while a shot in progress.

Most present-day computer systems do not have the memory (50 TB or so) needed to deal with large data collection, which might lead to an approach of examining one stream at a time or possibly processing one stream on one machine while simultaneously moving another stream.

The collaboratory nature of ITER also presents security challenges. Users can access parts of data per shot but are not allowed access to other associated information. Users need to add information/annotate shots and query off their own and other collaborators annotations.

To make this endeavor a success, efficient collaborative tools will be essential. The Fusion Grid System has been set up to address these issues. The current FGS uses Globus/SSL Certificate proxy-based authentication with a Globus gatekeeper and gridftp daemons. The original development was done through a SciDAC Collaboratory in 2000-2004. Briefly, the FGS has

- Distributed Authorization System (ROAM – Resource Oriented Authorization Mechanism)
- Systems (hardware) made available to support service
- Service software installation in service provider account
- Service execution in user service accounts
- Service provides configuration without requiring end user involvement
- Process remote user requests with tasks including: Job queuing, monitoring, data access, job cancellation
- No interactive access by end users
- Interactive access by maintainers for trouble shooting

10.1.1.3 Computational and Storage Requirements Summary

	Current (2010)	**In 2013[9]**
Computational Hours		136 M
Parallel Concurrency		1 M
Wall Hours per Run		10
Aggregate Memory		2-100 TB
Memory per Core		.1-2 GB/core
I/O per Run Needed		1 PB
On-Line Storage Needed		2 PB
Archival Storage		10 PB
Data Transfer Needed		

[9] The FSP is currently completing a two-year planning study. FES will evaluate the results of this planning in 2012 before deciding whether to launch this initiative in 2013 or beyond.

10.1.1.4 Support Services and Software

A previous (SciDAC-I) FES SciDAC Collaboratory Project (GA, MIT, PPPL, + CS Partners) successfully implemented useful new collaborative technology that addressed problems defined by fusion scientists. The FusionGrid provides services used to benefit daily FES research. FusionGrid proved successful by providing Simulation as a Service, optimizing the most expensive resource - people's time

To support the Fusion Systems Grid, the FSP project has a number of questions for NERSC

- Can systems be made available?
- Can GLOBUS be used?
- Can Service Provider Accounts be set up?
- Can end-user service accounts be established?
- Can expert access be allowed for troubleshooting?
- How could more flexible computing environments be implemented?
- Will storage/systems on FSP-administered devices reside on the NERSC LAN?

A possible Proto-type Project with NERSC would be to explore using the TRANSP/PTRANSP Auxiliary Heating (ICRF) Service as a near-term example for delivery of FSP production services to the user community.

10.1.1.5 Emerging HPC Architectures and Programming Models

The FSP is currently in a planning stage, but will require new, rigorous, well-engineered codes capable of running efficiently on future systems.

11 Innovative Confinement Concepts

11.1 Overview

This FES portfolio was reoriented in FY 2011 to address problems that are critical to the tokamak concept and tests the general validity of the relevant plasma physics and technology in a wider expanse of parameter regimes than those provided by the largest facilities. This program element emphasizes research on small- and medium-scale experiments that support toroidal configurations for future burning plasma experiments, such as the tokamak, the stellarator, and the spherical torus. Advanced simulations play a critical role in the validation mission of this program element.

11.2 The Plasma Science and Innovation (PSI) Center

NERSC PI: Brian Nelson (University of Washington)
Contributors: Charlson C. Kim
NERSC Repository: m489

11.2.1.1 Summary and Scientific Objectives

The Plasma Science and Innovation (PSI) Center is composed of researchers from the University of Washington, University of Wisconsin-Madison, and Utah State University. The PSI Center's main goal is to develop and provide practical and accurate, user-friendly codes to enable high-confidence predictive simulations of innovative confinement devices. Important tasks include adding sufficient physics modeling, boundary conditions, and geometric capabilities to benchmark results against experimental data.

The PSI Center has active collaborations with most of the ICC experiments, e.g., TCS-U University of Washington, MST (University of Wisconsin), and LDX (MIT). ICC experiments are usually smaller and cooler than their tokamak counterparts. Physical dimensionless parameters of these devices are often well within the fidelity regime of the PSI-Center codes. In addition, extended MHD effects, such as the Hall term, are often primary effects in the dynamics of interest unlike in tokamaks, where they are often smaller and isolated to narrow singular layers. These conditions provide the opportunity for more direct Validation and Verification (V&V) comparisons between simulations and experiments as well as test beds for developing extended models.

However, ICC simulations should not be viewed as easier because of their typically smaller magnetic Lundquist number. ICC simulations often involve formation of the plasma. These are usually strongly driven simulations that generate strong flows, steep gradients, regions of low density, and field nulls. These formation simulations are significantly different from the usual tokamak simulations and often involve more dynamic conditions.

11.2.1.2 Methods of Solution

The three main codes used by the PSI Center are NIMROD, HiFi, and PSI-Tet. Both NIMROD and HiFi are 3D, parallel, initial value, extended MHD codes using implicit time advances and finite element spatial discretization. NIMROD uses high order polynomial nodal finite elements in two of the spatial dimensions and a Fourier representation in the third periodic direction. The third Fourier direction provides computationally efficient representation of the periodic direction. HiFi uses 3D modal finite elements in all three directions. This provides HiFi with a high degree of geometric flexibility. NIMROD also has a hybrid kinetic-MHD PIC option and is developing a continuum method for advanced closures of the MHD equations. NIMROD is a prominent extended MHD code in the tokamak community. The parallelism for both codes is achieved through MPI. PSI-Tet is a 3D zero beta (no plasma pressure term) plasma equilibrium solver that uses tetrahedral elements to solve Taylor-like equilibria. PSI-Tet uses a hybrid OpenMP/MPI parallelism. All three codes rely on sparse scalable solvers.

11.2.1.3 HPC Requirements

Typical production simulations of ICC devices utilize many dozen to several hundred cores running tens of wallclock hours per run. Generally, the PSI-Center does not run enterprise-class computations using many thousands of processors. The typically smaller and colder plasmas modeled by the PSI-Center have modest computing needs compared to tokamaks. Higher throughput of modest sized, long-running jobs would be most beneficial to ICC simulations in the near future. This does not preclude the need for large-scale runs nor should it imply the absence of high performance plasmas (i.e. plasmas at high temperatures and densities close to the macroscopic stability limits), in the ICC community (e.g. MST). Indeed some of NIMROD's largest and best performing simulations have been to model the nonlinear dynamics of Reverse Field Pinch configuration similar to MST.

11.2.1.4 Computational and Storage Requirements Summary

	Current (2010)	**In 2013**
Computational Hours	0.8 M	30 M
Parallel Concurrency	300 – 1 K	4 K
Wall Hours per Run	24 – 48 Hr	24 – 48 Hr
Aggregate Memory	Many 100s GB to several TB	10+ TB
Memory per Core	1 – 2 GB	2 GB, more if possible
I/O per Run Needed	100 MB/hr – 10 GB/hr	1 GB/hr – 10s GB/hr
On-Line Storage Needed	1 TB	Many TBs
Data Transfer Needed	10s – 100 GB/wk	100s GB/wk

11.2.1.5 Support Services and Software

The PSI Center helps others run PSI-supported codes for their experiment of interest. NERSC could contribute greatly by providing end-user support, some of which may already exist. It would be useful if NERSC could make PSI Center codes (and perhaps all the enterprise class codes) available in the modules software environment. NERSC could also provide workflow tools to chain job submission, job tracking, on-the-fly data postprocessing, and job monitoring, (e.g, by posting on-the-fly post-processed plots onto a NERSC-maintained user's website). Another useful service would be a clickable website where one could specify files and directories and frequency for back up. Also helpful is a global file system across all machines to eliminate file transfers for post-processing. Continued and expanded visualization support is appreciated.

11.2.1.6 Emerging HPC Architectures and Programming Models

No specific plans have been devised to address new architectures. The PIC and continuum extensions to the NIMROD code represent the best candidates to take immediate advantage of any new architecture. Significant coordinated effort among computational physicists, applied mathematicians, and computer scientists — both hardware and software — will be required to advance and adapt sparse scalable solvers to new architectures. This need will be particularly acute if there is a significant paradigm shift, e.g. massively parallel hybrid architectures using GPU coprocessors.

12 Wave-Plasma Interactions

12.1 Center for Simulation of Wave-Plasma Interactions (CSWPI)

PI: Paul T. Bonoli (MIT)
Contributors: Lee A. Berry, David L. Green (ORNL), R. H. Harvey (CompX), Cynthia Phillips (PPPL), Myunghee Choi (General Atomics)
NERSC Repository: m77

12.1.1.1 Summary and Scientific Objectives

The over-arching goal of this project is to quantitatively understand how high power (tens of megawatts) radio frequency (RF) power in the ion cyclotron frequency range of frequencies (ICRF) and in the lower hybrid range of frequencies (LHRF) propagates from an external antenna and how it is subsequently absorbed in a tokamak plasma. This capability is needed to understand how to optimally use this power to heat, drive current, control plasma profiles, control plasma stability, and avoid parasitic losses in magnetically confined fusion plasmas, including ITER. This problem is computationally intensive because of its non-linear, 3-D, multiscale nature. Three-dimensional runs for linear and quasi-linear models of the plasma core can take 100-200k processor hours on today's machines. Non-linear 3-D runs that couple the core and edge (including the antenna) are expected to take 10-100 times more cycles. Specific scientific objectives include:

- Coupled core-to-edge simulations that lead to an increased understanding of parasitic losses in the boundary plasma between the RF antenna and the core plasma;
- Simulations of core interactions of RF power with energetic electrons and ions to understand how these species affect power flow in the confined plasma;
- RF effects on fast-particle driven instabilities to understand if these interactions increase (decrease) the instability drive that can lead to reduced fusion power.

To support these goals, we have to develop improved algorithms to achieve the needed physics, resolution, and/or statistics to address these issues and to efficiently utilize new computer architectures.

These objectives reflect the five-year goals of the CSWPI, as stated in the project's plans for future research.

12.1.1.2 Methods of Solution

Two classes of codes are used for addressing the CSWPI physics objectives. First, full-wave codes solve the integral-differential Maxwell's equations with a non-local, linear RF plasma conductivity to calculate the RF electric fields that are driven by external power sources. The All Orders Spectral Algorithm (AORSA) and the Toroidal Ion Cyclotron Code (TORIC) are the principal codes used by the project. The AORSA solver is fully spectral and employs a Cartesian coordinate system while the TORIC code is semi-spectral with Fourier modes in the poloidal and toroidal directions and finite element in the radial direction. The number of spectral modes required to achieve numerical convergence in these solvers is dictated by the shortest wavelength in the system that must be resolved. The plasma response can either be for thermal plasma components with a Maxwellian distribution function or non-thermal components with a specified distribution function. Typical problems require solving for 100K to 1 M unknowns for a single toroidal mode coupled by an ICRF launching structure. For lower hybrid heating and current drive calculations using TORIC, about 10 times more resolution is required, with about a 100-fold increase in the number of unknowns.

The RF fields can in turn modify the distribution function. The second class of codes model the evolution in time and phase space (velocity and configuration) of the distribution function as determined by quasilinear RF interactions. Explicit code coupling is valid for this problem because the RF time scale, nanoseconds, is much shorter that the distribution function evolution that takes place on transport/collisional time scales: 10s of ms to fractions of a second. Both continuum (CQL3D) and Monte Carlo (ORBIT-RF and sMC) techniques are employed. Although the interaction between the wave solvers and Fokker Planck / Monte Carlo codes is highly nonlinear, the problem is solved stably by iterating between the physics modules as time is evolved on the quasilinear time scale. Thus four different physical models are iteratively coupled: (1) the plasma conductivity for non-Maxwellian distribution functions, (2) a wave solver incorporating this non-Maxwellian conductivity, (3) the quasi-linear operator for the non-thermal distribution, and (4) a Fokker-Planck solver or Monte Carlo code.

For the full wave AORSA solver, a completely dense matrix results from the integral plasma conductivity relation and spectral basis set used in that code. A block tridiagonal system with completely dense blocks results from the algebraic conductivity (finite ion Larmor radius expansion) and spectral basis set used in the TORIC solver. Factoring these matrices is often the dominant computational effort, and we rely on dense solvers (ScaLAPACK or HPL) and either Thomas or cyclic reduction techniques for AORSA and TORIC respectively. Filling the matrices and post processing can take a significant fraction of the total computational time, especially for non-thermal distributions. Efficient integration/quadrature techniques and special function evaluations are critical for these steps.

Both of the full wave codes solve for one toroidal mode at a time, and must be run once for each of the 50-100 significant toroidal modes of the antenna. The final plasma

response is computed by linear superposition of the electric field solutions of the individual toroidal modes.

The Fokker-Planck solvers both rely on particle-based techniques (for generating the quasilinear operator for CQL3D) and for determining a Monte-Carlo solution for ORBIT-RF and sMC.

12.1.1.3 HPC Requirements

A single, typical, 2D run of the full wave solvers requires hundreds to thousands of processors and takes ~1 hour to complete. Higher resolution runs require 10k processors (or more) and take 5-10 hours to complete. Full 3-D solutions can only be obtained at modest resolution, and can also be completed on ~10k processors in 5-10 hours depending on details of the calculation. In both cases, ~100k processor hours are required. Key issues going forward include: implementing restart capability at the completion of the matrix fill and matrix factor steps in the full wave solvers; and coupling high-resolution edge solutions to the present core solvers,

Additional computational resources are required to reach our long-range goal of developing a quantitative understanding of how to reliably heat and drive current in a fusion plasma, including ITER. These needs, relative to present simulations, are driven by: routine 3-D simulations (50-100 times more work); the need for improved resolution in the edge plasma, including nonlinear wave-wave; and wave-particle interactions, and the need to model the time-dependent behavior of energetic ion species.

12.1.1.4 Computational and Storage Requirements Summary

	Current (2010)	**In 2013**
Computational Hours	1.5 M	50 M
Parallel Concurrency	10 K	200 K
Wall Hours per Run	8	12
Aggregate Memory	10 TB	20 TB
Memory per Core	1 GB	1 GB
I/O per Run Needed	100 MB	100 TB (for restart)
On-Line Storage Needed	0.1 TB	10 TB
Data Transfer Needed	1-5 GB / day	10-20 GB / day

12.1.1.5 Support Services and Software

A logical continuation/expansion of the present requirements would be reasonable, assuming that restart capability can be accommodated within anticipated increases in file system capability. However hardware and compiler support for PGAS languages such as Co-array FORTRAN and UPC would be desirable. At the present time we perform limited 3-D RF field reconstructions using the VisIt software suite. In the next 3-5 years we anticipate increased reliance on 3-D visualization software such as VisIt as we greatly

expand our studies of how ICRF and LHRF fields are coupled (both linearly and nonlinearly) by 3-D antenna launching structures and how they interact with the complicated edge geometry of the tokamak vessel.

12.1.1.6 Emerging HPC Architectures and Programming Models

Our first need is that efficient libraries be developed. FFTs, sparse, block, and dense matrix solves, and parallel function evaluation are prominent needs. Second, we have identified CPU-intensive kernels in our codes with local data needs that would significantly benefit from GPU technology. Examples include a double integral in the matrix fill for AORSA and a 4D sum in AORSA. Performance gains for the 4D summation have already been demonstrated on ORNL machines with GPUs. Some effort to realize this potential is planned within the present project, but additional resources will be required for a production implementation.

But even this type of effort depends what the hardware will be. A general strategy will be to separate inter-node communications from intra-node memory management. That is, use MPI only for inter-node communications. This should provide a good starting point for future development. However, progress on even this step will be limited by resources.

13 Heavy-Ion Fusion Science and Inertial Fusion Energy

13.1 Simulation of Intense Beams and Targets for Heavy-Ion-Fusion Science

PI: Alex Friedman, LLNL, LBNL, and Heavy Ion Fusion Science Virtual National Laboratory

Contributors: John Barnard, Ron Cohen, Mikhail Dorf, David Eder, Dave Grote, Steve Lund, Bill Sharp, LLNL; Enrique Henestroza, Ed Lee, Jean-Luc Vay, LBNL; Ron Davidson, Igor Kaganovich, Hong Qin, Ed Startsev, PPPL; Kirsten Fagnan, Alice Koniges, NERSC; Andrea Bertozzi, UCLA

NERSC Repository: mp42

13.1.1.1 Summary and Scientific Objectives

The US Heavy Ion Fusion Science Virtual National Laboratory (HIFS-VNL), a collaboration of LBNL, LLNL, and PPPL, conducts research on the science of intense heavy-ion beams, on high-energy-density physics (HEDP) and especially Warm Dense Matter (WDM) generated by ion beams, and on target physics for ion-beam-driven Inertial Fusion Energy. Ongoing experiments are therefore focused on generating, compressing, and focusing space-charge-dominated beams and using them to heat thin foils, the evolution and properties of which are then measured. To further this work, a new accelerator facility, the Neutralized Drift Compression Experiment-II (NDCX-II), is under construction at LBNL, with completion planned for 2012. Obtaining maximum benefit from these experiments is a key near-term goal of the simulation program. Recent simulation efforts in support of NDCX-II have concentrated on developing the physics and engineering design of the NDCX-II accelerator, on identifying favorable operating points, and on planning the WDM experiments to be done with its beam once operations ensue. As we transition to support of the actual experiments, the scope of our beam simulations must expand greatly, with primary emphasis on detailed simulations of the actual beam as realized and its interaction with the target. This will include extensive studies of the coupled dynamics of the beam and the neutralizing plasma (which allows the beam to be compressed to a compact volume in space and time), and routine transfer of the ion beam data at the target plane from our main beam physics code, Warp, into hydrodynamic simulations of the target behavior using the Hydra code (run at LLNL) and the new ALE-AMR code (run at NERSC).

Intense ion beams are non-neutral plasmas and exhibit collective, nonlinear dynamics that must be understood using the kinetic models of plasma physics. This physics is rich and

subtle: a wide range in spatial and temporal scales is involved, and effects associated with instabilities and non-ideal processes must be understood. In addition, multispecies effects must be modeled during the interaction of the beams with any stray electrons in the accelerator and with the neutralizing plasma in the target chamber. The models must account for the complex set of interactions among the various species, with the walls, and with the applied and self-fields. Finally, oscillations of the beam core and "mismatches" of the beam confinement can dilute the beam phase space and parametrically pump particles into a low-density outlying "halo" population; this physics imposes stringent requirements on numerical noise, requiring good particle statistics and mesh resolution. A blend of numerical techniques, centered around the PIC method are used in Warp to address these needs, including: electrostatic and electromagnetic solvers, adaptive mesh refinement, cut-cell boundaries, large time step particle pusher, and implicit field solvers and particle pushers. Ion beams have a long memory, and initialization of a simulation at mid-system with an idealized particle distribution is often unsatisfactory; thus, a key goal is to further develop and extensively exploit an integrated and detailed source-to-target beam simulation capability.

In order to determine material properties (such as equation of state) in the WDM regime, we must use simulation codes to help interpret experimental diagnostics. The WDM regime is at the boundaries of solid state physics and plasma physics, between non-degenerate material and degenerate, between ionized and neutral, between liquid and vapor. To understand and simulate the results of experiments that volumetrically heat material, many processes must be combined into a single simulation. These include phase changes, ionization processes, shock processes, spall processes, and droplet formation, to name a few. Our goal is to fully characterize the experiments that will be conducted on NDCX-II. This accelerator-based facility will compress ion beams and use them to heat material targets to temperatures of about 10,000 degrees K. The material is heated so rapidly that, although its temperature is well above the vaporization temperature, its inertia will keep it at solid density (at least for an inertial confinement time of order one ns). Experimental diagnostics will record the response of the material (e.g. temperature, density, velocity) to infer the equation of state and other properties. Synthetic diagnostics in the simulations will be essential for inter-comparison.

Hydrodynamic codes such as Hydra and ALE-AMR can be used to model ion deposition and the subsequent response of the target to bulk heating. The ALE-AMR code can also model the strength and failure of materials using various models, some of which include history variables and non-isotropic terms. We are implementing surface tension effects in ALE-AMR. The formation of droplets has been one of the dominant features of data from the predecessor to NDCX-II, NDCX-I, yet there are no available codes that can describe the interplay between surface tension, inertial, and van der Waals forces that form the droplets. The droplets are predicted to be on the 0.1-micron scale for NDCX-II, yet the gradient scale length for the ion deposition (i.e. the transverse focal spot size) is on the 500 micron scale. So our intent is to model the formation of droplets initially on a microscopic patch of the target, but eventually increase the computational domain of the simulation until the entire beam heated part of the foil is included in the simulation.

Within 3 to 5 years, we expect that end-to-end simulations of the beam from source through target, and of the detailed target response, will be routinely carried out at high fidelity. We will also be exploring extensions to NDCX-II, and/or a follow-on facility, requiring extensive use of ensembles of runs, as has been done for the baseline design of NDCX-II.

Our principal goals, and activities in support of those goals, over the next five years are as follows:

(1) Optimize the properties of the NDCX-II beam for each class of target experiments; achieve quantitative agreement with measurements; develop improved machine configurations and operating points. To accomplish these goals, we plan to use Warp to simulate NDCX-II from source to target, in full kinetic detail, including first-principles modeling of beam neutralization by plasma. The output from an ensemble of Warp runs (representing shot-to-shot variations) will be used as input to target simulations using ALE-AMR on NERSC, and other codes.

(2) Develop enhanced versions of NDCX-II (the machine is designed to be extensible and reconfigurable), and carry out studies to define a next-step ion beam facility. To accomplish these goals, much of the work will involve iterative optimization employing Warp runs that assume ideal beam neutralization downstream of the accelerator.

(3) Carry out detailed target simulations in the WDM regime using the ALE-AMR code, including surface tension effects, liquid-vapor coexistence, and accurate models of both the driving beam and the target geometry. For this we will need to make multiple runs (to capture shot-to-shot variations), and to both develop and employ synthetic diagnostics (to enable comparison with experiments). The new science that will be revealed is the physics of the transition from the liquid to vapor state of a volumetrically superheated material, wherein droplets are formed, and wherein phase transitions, surface tension and hydrodynamics all play significant roles in the dynamics. These simulations will enable calculations of equation of state and other material properties, and will also be of interest for their illumination of the science of droplet formation.

13.1.1.2 Methods of Solution

Our main ion-beam code, Warp, was originally developed to simulate space-charge-dominated beam dynamics in induction accelerators for heavy-ion fusion (HIF). In recent years, the physics models in the code have been generalized, so that Warp can model beam injection, complicated boundary conditions, denser plasmas, a wide variety of accelerator “lattice” components, and the non-ideal physics of beams interacting with walls and plasmas. The code now has an international user base and is being applied to projects both within and far removed from the HIF community.

Warp uses a flexible multi-species PIC model to describe beam dynamics and the electrostatic or electromagnetic fields in particle accelerators, particularly those driven by

induction modules. While the core routines of Warp solve finite-difference representations of Maxwell's equations and relativistic or non-relativistic motion equations, the code also uses a large collection of subordinate models to describe lattice elements and such physical processes as beam injection, desorption, and ionization. The representation of particles by a much smaller number of "macroparticles" can be derived from Boltzmann's equation, describing the evolution of a population of particles interacting by collisions and the collective fields.

Warp also includes a Warped-coordinate particle advance to treat particles in a curved beam pipe. Self-fields are obtained via Poisson equations for the scalar and vector potentials or full electromagnetic via Maxwell equations. Simplified models are available for the self- magnetic and inductive forces. Time-dependent applied external fields can be specified through the Python user interface. Warp also has 2-D models, using Cartesian or cylindrical geometry, as well as a module representing the beam with a 4-D Vlasov formulation and with low-order moment equations. Models are available for background gas, wall effects, stray electrons, space-charge-limited and source-limited emission, and atomic processes such as charge exchange. Elaborate initialization and run-time options allow realistic modeling of induction accelerators. A beam may be initialized with one of many analytic distributions or with a distribution synthesized from experimental data, or ions can be emitted from a flat or curved diode surface. Lattice-element fields may be represented by several options, from simple hard-edge analytic forms to first-principles 3-D calculations. Poisson's equation can be solved using several methods, including FFT, Multigrid, and AMR/Multigrid. The electromagnetic (EM) solver can also use MR. With multigrid, the Shortley-Weller method for the subgrid-resolution description of conductors allows the use of complicated boundary conditions. Warp consists of about 130,000 lines of Fortran, 100,000 lines of Python, and about 4,000 lines of C.

Parallelization of Warp is done using domain decomposition with MPI. Warp uses independent spatial decompositions for particles and field quantities, allowing the particle and field advances to be load-balanced independently; we recently added more general decompositions. In transverse-slice 2-D runs, the field solution is repeated on each node, but solved in parallel by processors within a node.

The size and duration of Warp jobs varies tremendously, depending on such factors as problem dimensionality, grid size, duration, particle count, and the physical processes being modeled. However, with our generalized decomposition, we do not foresee any limitation resulting from the code's architecture. For a 3-D test problem using 512x512x512 cells, we have demonstrated excellent parallel scaling of the electromagnetic PIC capability, up to nearly 35,000 processors.

Our Warp projects tend to use modest amounts of memory but require many time steps. We typically run (in 2-D or 3-D) with of order 100 grid cells along each axis. Higher resolution and large 3-D simulations typically have a mesh of order 100s by 100s by 1000s of grid cells. The data per cell is either a single point or a 3-D vector. Typically of order 1,000,000 particles are used, with 13 or more variables per particle. We currently

use 512 to 1024 processors for typical Franklin runs and 4096 for a few key runs with fine grids and an augmented number of particles.

ALE-AMR is a relatively new code that combines Arbitrary Lagrangian Eulerian (ALE) hydrodynamics with Adaptive Mesh Refinement (AMR) to connect the continuum to micro-structural regimes. The code is unique in its ability to model both hot radiating plasmas and cold fragmenting solids. The hydrodynamics are done in a Lagrangian model (wherein material moves with the mesh), but the resulting mesh can be modified to prevent tangling or severe mesh distortions. If the entire mesh is restored to the mesh of the previous time-step after every step, the code is said to be run in Eulerian mode (fixed mesh). In general, this is not done, and we only modify a portion of the mesh during a fraction of the time steps. This ability to do selective remapping is the reason to use the word "arbitrary." We also employ the Hydra code, another 3-D radiation hydrodynamics ALE code; that code is run on LLNL computers, which are accessed from LBNL and LLNL by group members. A common feature of ALE codes is the ability to have multiple materials in a given computational zone. Such mixed zones are generally created during the advection phase of the advance, when material from the old mesh is transferred to the new mesh. The ALE-AMR code uses a volume-of-fluids approach to calculate the interface between different materials in a zone. Information from neighboring zones can be used to explicitly construct the interfaces if needed.

Our ALE-AMR code consists of about 130,000 lines of C++, 25,000 lines of Fortran, 6,000 lines of Python, and about 3,000 lines of other code (such as sh); this is excluding the SAMRAI package.

One key added capability of ALE-AMR, relative to other ALE codes such as Hydra, is the ability to dynamically add mesh elements (refinement) or remove mesh elements (coarsening) during the run. ALE-AMR refines by a factor of three along each dimension, so in 3D one zone becomes 27 zones. During refinement all material interfaces must be explicitly defined to place the correct amount of each material in the new zones. Numerical techniques were developed for many of the physics packages to work efficiency on a dynamically moving and adapting mesh. ALE-AMR also continues several features that allow for very long-time simulations, a unique fragmentation capability, and the ability to "shape-in" unusual objects.

Additional physics, beyond basic hydrodynamics, is implemented in ALE-AMR using operator splitting. For example, a flexible strength/failure framework allows "pluggable" material models to update the anisotropic stress tensor that is used in the hydro advance during the following step. The code also includes an ion deposition model for bulk heating of the material, and both heat conduction and radiation transport using the diffusion approximation. The hydro uses explicit time stepping but some packages, e.g., radiation transport, can do an implicit solve at each explicit time step.

The parallelism in ALE-AMR is currently MPI-only with the ability to do dynamic load balancing based on the computational requirements. The domain decomposition is zonal. During the ion deposition phase of the simulation, the regions with ion beams will have

smaller number of zones in the domain assigned to a given processor because of the additional computation work associated with beam energy deposition. There are various places in the code were additional levels of parallelism are possible and we are investigating hybrid models, e.g., OpenMP + MPI.

13.1.1.3 HPC Requirements

Our need for NERSC time is rapidly growing. In past years our requirements were modest. We concentrated on NDCX-II machine design using simplified tools such as a 1-D beam physics code (ASP), and on WDM simulations using both Hydra (at LLNL) and a specialized 1-D code (Dish). Recently, we began applying NERSC resources to iterative design calculations for the NDCX-II facility, and our usage rate increased roughly five-fold. A NISE allocation of 500,000 hours has been extremely valuable. Now, two developments compel us to carry out far more demanding simulations at NERSC: (1) the need to capture beam-in-plasma effects (requiring far more simulation particles, and a smaller time-step size and grid spacing); and (2) the introduction of the ALE-AMR code into our group (requiring considerable resources for realistic problems).

We begin with a discussion of recent Warp usage, which has emphasized iterative design and assessment (on NERSC and LBNL clusters) using ensembles of runs with random errors. For this task 256 cases (instances) are typically run in a single batch job, 8 at a time. This employs 128 cores, with less than 1 GB/core, using 60 GB total memory and 35 hours of wall-clock time. Much data processing is in-line, and I/O is only about 100 GB / batch job. This approach leads to very light traffic in and out of NERSC, with results stored at the Center.

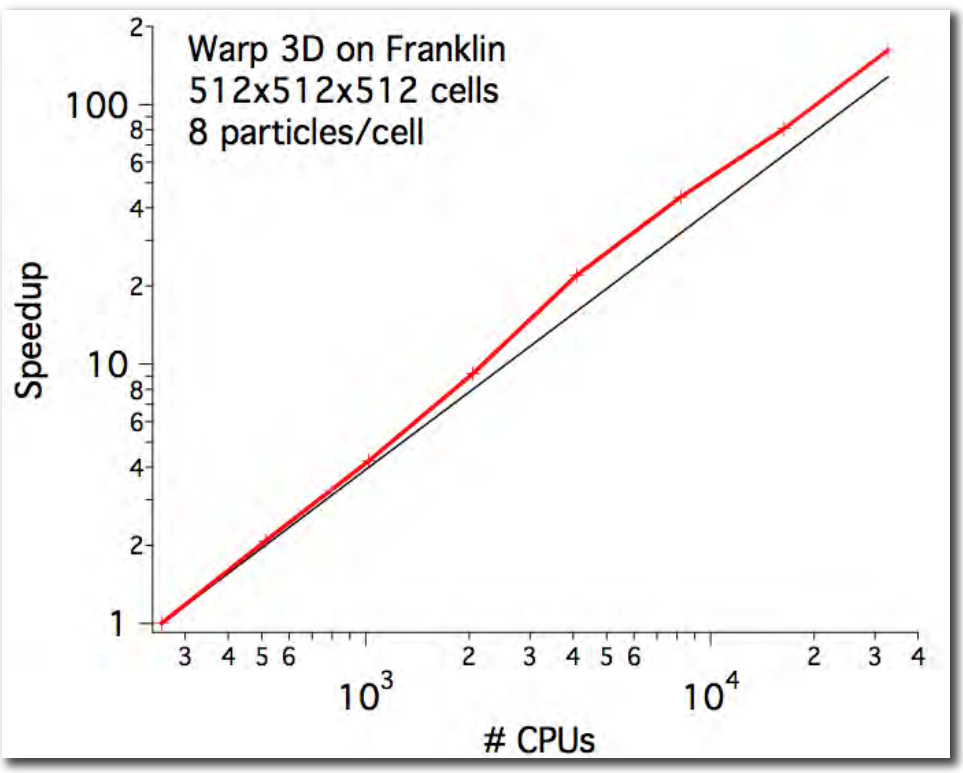
Another class of Warp runs models ion beams in plasmas. Current problems of this type use 100's x 100's x 1000's of cells, and millions of particles (with 13 or more variables per particle), with 512, 1024, and sometimes 4096 processors on Franklin. With the electromagnetic Maxwell (EM) field model, tests show good scaling at fixed problem size (5123 cells) to 35,000 processors (see Figure).

We project the need for four classes of Warp runs during the next five years: (1) Ensemble runs to optimize the output beam from the NDCX-II accelerator, for each class of target being shot. So far, we haven't used gradient methods for optimization because of particle noise; we hope to overcome this with larger runs. (2) Simulations of plasma injection into the drift-compression line and final-focus solenoid, which can be quite costly because the plasma flow is relatively slow (~10 ms) and it is necessary to operate on an electron timescale. Both EM and explicit electrostatic (ES) models are used; run times are comparable because the time-step size in the ES model, which is set by the need to resolve plasma oscillations, is near the Courant limit for light waves on the mesh. Also, the EM algorithm scales more readily to very large numbers of processors. (3) Integrated simulations of one or more beams compressing in a neutralizing plasma (with properties obtained from plasma-injection runs as described above, or via measurements). Such runs require less computer time than plasma injection runs, because the beam is in the system for < 1 ms; however, ensembles are typically needed. (4) Detailed simulations resolving

short time- and space-scales for, e.g., two-stream instability. Since the highest growth rates for a cold beam and plasma are for short wavelengths, while we seek to capture the overall system scale, such runs can be costly, even in axisymmetric (r,z) geometry.

A proper estimate of the computer resources required for an integrated, end-to-end kinetic simulation of beam(s) and plasma using Warp would assess the several regions of the system separately and would assume use of AMR, variable time-step sizes, and perhaps high-order differencing (to allow the cell size Δx to exceed the Debye shielding length λD by large factors). Such an estimate would itself require the development and use of a small computer program.

The computational requirements for target simulations using ALE-AMR are large because of the wide range of spatial scales involved and the need to include multiple detailed physics models. The width of the ion beam is generally of order 0.5 mm, which requires a computational mesh of a few mm in each dimension. However, droplets as small as 0.1 micron are expected, and it is desired to have a number of zones within each droplet. For a uniform 3D run, this would imply the need for an unacceptable number (~1012 - 1015) of zones. The ratio of the zone size at the coarsest level to the zone size at the finest level is greater than 107 for 6 levels of refinement. Even with this level of refinement, the number of zones in the coarse level would still be too large to model the entire domain. Only selected regions will be modeled at the highest resolution with the resulting information (e.g., the droplet distribution) being assumed to be relevant to other regions with similar hydrodynamic properties (i.e., pressure, density, temperature, stress, and strain rates).



13.1.1.4 Computational and Storage Requirements Summary

Total (both main codes)	**Current (2010)**	**In 2013**
Computational Hours	0.6 M	60 M

The "current usage" numbers above include about 200k hours / year on the Fusion and Lawrencium clusters, and an estimated 400k hours at NERSC (twice our usage during the first half of 2010). Our NERSC usage in 2009 was 74k hours. A NISE allocation of 500k hours has been extremely useful.

Warp code (ion beam simulation)	**Current (2010)**	**In 2013**
Computational Hours	0.5 M	40 M
Parallel Concurrency	128 - 32k	50k – 500k
Wall Hours per Run	2 – 35	10 – 100
Aggregate Memory	60 GB	3 – 30 TB
Memory per Core	1 GB	1 GB
I/O per Run Needed	0.1 TB	0.5 -5 TB
On-Line Storage Needed	1 TB	10 – 100 TB
Data Transfer Needed	0.1 TB / year	1 TB / year

ALE-AMR code (target simulation)	**Current (2010)**	**In 2013**
Computational Hours	0.1 M	20 M
Parallel Concurrency	128 – 10k	10k – 30k
Wall Hours per Run	6 – 50	20 – 150
Aggregate Memory	64 – 5,000 GB	5 – 15 TB
Memory per Core	1 GB	1 GB
I/O per Run Needed	0.2 TB	1 - 10 TB
On-Line Storage Needed	2 TB	20 – 200 TB
Data Transfer Needed	0.1 TB / year	1 TB / year

13.1.1.5 Support Services and Software

For some of our applications, long batch queue wait times and limitations on interactive use are impeding factors. Interactivity is useful for debugging and for diagnostics development, because Warp produces many of its diagnostics on-line (thereby avoiding massive data transfers, offloading mostly processed data). The Lawrencium and Fusion

clusters at LBNL have been convenient, especially for multi-week runs, but are limited in capacity, capability, and availability.

The Warp code is deeply tied to Python. Dynamic libraries are part of the core of Python and it would be useful for them to be supported. This would greatly ease code maintenance and preserve flexibility. Dynamic libraries are not actually essential, since Warp and Python can be built statically. However, static loading requires modification of the Python source, introduces a more complicated and fragile build process, and makes installing new and upgrading existing packages more difficult since they must be manually incorporated into the build system (when they could otherwise be installed independently). A secondary need, related to the startup time, is an efficient mechanism for Python on the nodes to read in the sizable number of startup scripts and dynamic objects (when supported).

Scaling studies of ALE-AMR showed excellent parallel scaling to >5000 cores, but uncovered memory access differences on Franklin during initialization. For example, when the large equation-of-state table was moved from /project to /home there was a 10X speedup, with further speedup observed when the table was read from $SCRATCH (this is the current mode of operation). These effects are still being studied, but they do not impact the performance of the code after initialization

13.1.1.6 Emerging HPC Architectures and Programming Models

Warp's core loop incorporates the PIC elements: field solve, gathering of field onto particles, particle advance, charge or current deposition onto grid. The algorithms that have been implemented to enable progressing toward high fidelity end-to-end calculations span the space between (a) "computationally intensive" massively-parallel methods using the lowest level of approximation with explicit solvers on uniform grids, small time steps, and large numbers of macroparticles, and (b) "algorithmically intensive" moderately-parallel methods using higher-level approximations involving implicit methods, global solvers, AMR grids, large time steps, and moderate numbers of macroparticles.

We will pursue both efficiency and flexibility so that code users may continue to employ varying levels of description, itself a key to insight. On a five-year timescale we must accommodate a significant increase in cores per node, a decrease in memory per core, and an increase in the computational cost of data movement. At this point it is unclear whether the NERSC-7 computer will employ a descendant of today's CPU/GPU architecture and programming model, a Blue-Gene-like configuration, or an intermediate model. It is very possible that one combination of algorithms will perform better on one machine while a different combination will be more adapted to another.

In any case, data locality will be important. The gathering of fields, particle advance, and charge or current deposition are local if properly implemented. A complication is the need for scatter-add deposition of source terms from the particles onto the mesh. Integrity

of these terms must be preserved as we optimize performance using (probably) MPI plus on-node parallelism, and AMR.

The explicit electromagnetic (Maxwell) field advance (Finite-Difference Time-Domain) is also local. However, the electrostatic field model is based on a global elliptic solver (multigrid is our main approach) is likely to be more challenging; for it we plan to pursue optimization of our own AMR/multigrid solver, since it interfaces nicely with Warp's data structures, but also will assess other available packages as they are developed. Much of our work to date has used the electrostatic field model, which is appropriate for many of our problems. However, we may shift emphasis to an explicit electromagnetic Maxwell-Vlasov formulation. Given the denser neutralizing plasmas anticipated in future experiments, the time-step size for EM is likely to be similar to that for ES.

We will partner with others in the PIC simulation community and the computer science community, to develop, test, and implement multi-level decompositions that make effective use of the new architectures that emerge. We plan to extract kernels of the main components and analyze their projected performance on those architectures, and select the most effective methods.

Initial efforts at porting PIC methods to GPU architectures have been encouraging. However, while our beams are non-neutral or neutralized plasmas, they are not homogeneous plasmas, and boundary condition handling can be nontrivial. We must also deal with collision events (between charged particles, and with neutrals), and must collect more extensive particle diagnostics than on most other PIC codes. These attributes will need to be accommodated.

For the ALE-AMR code, we will also carry out the necessary optimizations to enable efficient use of the NERSC-7 machine. At present the code employs the SAMRAI AMR package, but it could as readily employ the CHOMBO package should that framework be more efficient on NERSC-7. In any event, performance will depend heavily on that of the underlying package. Physics such as surface tension, important to understanding the formation of droplets in NDCX, may be implemented through operator splitting techniques, allowing some degree of independent optimization.

13.2 Large-Scale Particle-in-Cell Simulations of Laser-Plasma Interactions Relevant to Inertial Fusion Energy

NERSC PI: F. Tsung, UCLA

NERSC Repository: m1110

13.2.1.1 Summary and Scientific Objectives

Inertial fusion is a process where fusion reactions are initiated by heating and compressing a fuel target, typically in the form of a pellet that most often contains a mixture of deuterium and tritium, using high-energy beams of laser light, electrons or ions, The goal of this project is to use state-of-art PIC tools (such as OSIRIS and UPIC) to study parametric instabilities under conditions relevant to inertial fusion energy (IFE). These instabilities can absorb, deflect, or reflect the laser, and generate hot electrons which can degrade compression. However, in some exotic schemes the hot electrons can be beneficial, such as in shock ignition where the fast electrons create a shock that can produce an ignition, thereby enhancing gain. Therefore, a thorough understanding of these instabilities is crucial. Because of the highly nonlinear nature of these instabilities (which includes the interaction between waves and particles and waves and other waves), PIC codes that are based on first principles are well suited to study them.

The UCLA computer simulation group has a long history of expertise in PIC simulations as well as in parallel computing. In the past few years, we have applied this expertise to the study of laser plasma interactions. Some of our accomplishments include:

(i) Using the parallel PIC code OSIRIS to observe (for the first time) the high frequency hybrid instability (HFHI);

(ii) Indentifying the importance of convective modes in two-plasmon decay;

(iii) Showing the importance of plasma wave convections in the recurrence of Stimulated Raman Scattering (SRS); and

(iv) Finding that multi-dimensional plasma waves become localized due to wave-particle effects even in the absence of plasma wave self-focusing.

With the National Ignition Facility (NIF) coming online, this is the perfect time to apply both the expertise of the UCLA group and the HPC resources at NERSC to study the various laser-plasma interactions that can be occur under IFE relevant conditions. In the next 3-5 years, we plan to tackle the following problems at NERSC:

(i) 2-D simulations of SRS involving multiple speckles or multiple laser beams;

(ii) Effects of overlapping laser beams for two plasmon/HFHI instabilities near the quarter critical surface; and

(iii) Two-dimensional studies of SRS/2wp instability under shock ignition relevant conditions.

13.2.1.2 Methods of Solution

OSIRIS is a fully explicit, multi-dimensional, fully relativistic, parallelized PIC code. It is written in Fortran95 and takes advantage of advanced object-oriented programming techniques. This compartmentalization allows for a highly optimized core code and simplifies modifications while maintaining full parallelization done using domain decomposition with MPI. There are 1-D, 2-D, and 3-D versions that can be selected at compile time. In addition, one of OSIRIS's strongest attributes is the sophisticated array of diagnostic and visualization packages with interactive GUIs that can rapidly process large datasets (c.f. visualization section). These tools can also be used to analyze data generated from our PIC codes.

Recently, we have added dynamic load balancing, perfectly matched layers absorbing boundary conditions [vay:02], and an optimized version of the higher order particle shapes [esirkepov:01]. The use of higher order shape functions combined with current smoothing and compensation can dramatically reduce numerical heating and improve energy conservation without modifying the dispersion relation of plasma waves.

OSIRIS also has packages for including physics beyond the standard PIC algorithm. These include tunnel and impact ionization as well as a binary collision operator. There are two field ionization models, the ADK model and a simple barrier suppression model. These algorithms could also be used to model electron positron pair creation. Due to the presence of the grids (cells), particles in PIC codes have finite size and therefore collisions are modified from point particle collisions, especially when the impact parameter is comparable to the cell size, typically a Debye length. For smaller impact parameters, the effects of collisions are greatly reduced in PIC codes. In order to study the effects of collisions for absolute and not normalized plasma density and temperatures, it is also useful to explicitly add a Coulomb collision model into the PIC algorithm. We have implemented a binary collision module for OSIRIS using both the methods of T. Takizuka and H. Abe [takizuka:77] and Nanbu [nanbu:97]. We have generalized these methods for relativistic temperatures, and extended them to handle particles of different weights (useful, for instance, in a density gradient). The algorithm has been tested by comparing the relaxation times obtained from simulations of a two-species plasma out of equilibrium. The algorithm was also extensively tested to guarantee that the proper Jüttner distribution functions are reached in equilibrium for relativistic temperature.

The code is highly optimized on a single processor, scales very efficiently on massively parallel computers, and is very easily portable between different compilers and hardware architectures. To date, it has been ported on Intel, AMD, and IBM PowerPC, and

BlueGene processors running a large variety of operating systems (Mac OS X, AIX, Linux, among others). And for each of these platforms, the parallel scalability has been good regardless of the network configuration. On the Atlas machine at LLNL, 80% efficiency was achieved for 4,096 CPUs using a fixed size problem (strong scaling) with significant communication overhead (only 512x512x256 cells and only one billion particles were used). More recently, OSIRIS was ported to the IBM BlueGene Intrepid cluster (8,192 quad-core nodes, 32,768 processors). The code is 97% efficient on 32,768 CPUs with weak scaling (constant work per task) and 86% efficient with strong scaling. OSIRIS scales at better than 60% efficiency on more than 64k cores of the Cray XT5. Therefore there are no bottlenecks at this point, although due to the large memory requirement of future simulations, higher bandwidth for I/O and checkpointing will be needed.

Another code, UPIC, developed by Dr. Viktor Decyk of the UCLA simulation group, is being used as a testbed for the GPU platform. The UCLA Parallel PIC Framework (UPIC) is a unified environment for the rapid construction of new parallel PIC codes. It provides trusted components from UCLA's long history of PIC development, in an easily accessible form, as well as a number of sample main codes to illustrate how to build various kinds of codes. UPIC contains support for electrostatic, Darwin, and fully electromagnetic plasma models, as well as relativistic particles.

13.2.1.3 HPC Requirements

With an order of magnitude increase in computing power, we could study the effects of multiple (in this case, more than two) beams on the excitation of SRS/2wp instabilities in NIF relevant regimes. However, an increase of two orders of magnitude increase would be required before we could finally perform full 3-D simulations of parametric instabilities using parameters relevant to NIF. There are important questions that can only be answered by full 3D simulations, such as the effects of side loss in a three dimensional system where particles can move in and out of the packet due to particles' linear response to the laser's electric field. Some of the computational cost can be eliminated by computational tricks such as sub-cycling, where the ion orbits are calculated less frequently. Thus, 100 times more computing power would be required before we could include the necessary higher dimensional effects, such as side loss or wave front bending in full 3-D geometry. Estimates for a typical 3-D run are included in the table below.

13.2.1.4 Computational and Storage Requirements Summary

	Current (2010)	**In 2013**
Computational Hours	1.5 M	40 M
Typical Parallel Concurrency	1 K	100 K
Wall Hours per Run	100	400
Aggregate Memory	1 TB	600 TB
Memory per Core	0.6	6 GB
I/O per Run Needed	4 TB	500 TB

On-Line Storage Needed	15TB/run	600 TB
Off-line Storage Needed	1TB/year	n/a
Data Transfer Needed	n/a	n/a

13.2.1.5 Support Services and Software

For simulations that require more than 100TB (of memory?), it may be impossible to checkpoint and thus queuing policy will need to change. Also, as the size of simulation data increases, parallel visualization tools will be needed.

13.2.1.6 Emerging HPC Architectures and Programming Models

Viktor Decyk of our group has ported his code UPIC to the GPU using CUDA. This work, which relies on streaming of data, will also improve performance on other advanced architectures. A skeleton 2D electrostatic PIC code on the NVIDIA GTX 280 runs 17 times faster than a similar code running on the 2.66GHz Intel Nahalem chip. This work is being prepared for publication.

References:

[OSIRIS] R. G. Hemker, F. S. Tsung, V. K. Decyk, W. B. Mori, S. Lee, and T. Katsouleas, *Proceedings of the 1999 Particle Accelerator Conference*, New York, NY, edited by A. Luccio and W. Mackay (IEEE, Piscataway, NJ, 1999), Vol. 5, pp. 3672–3674, R. A. Fonseca, L. O. Silva, F. S. Tsung, V. K. Decyk, W. Lu, C. Ren, W. B. Mori, S. Deng, S. Lee, T. Katsouleas, and J. C. Adam, *Lecture Notes in Computer Science* (Springer, Heidelberg, 2002), Vol. 2331, pp. 342–351. R. A. Fonseca *et al*, "One-to-one direct modeling of experiments and astrophysical scenarios: pushing the envelope on kinetic plasma simulations", *Plas. Phys. Cont. Fus.*, **50**, 124034 (2008).

[esirkepov:01] T. Zh. Esirkepov, *Comp. Phys. Comm.*, **135**, 144 (2001).

[vay:02] J. L. Vay, *Jour. Comp. Phys.*, **183**, 367 (2002).

[Takizuka:77] T. Takizuka and H. Abe, *J. Comp. Phys.*, **25**, 105 (1977).

[nanbu:98] K. Nanbu and S. Yonemura, *J. Comp. Phys.*, **145**, 639 (1998).

14 General Plasma Science

14.1 Center for Integrated Analysis and Computation of Reconnection and Turbulence

NERSC PI: Amitava Bhattacharjee, University of New Hampshire
Contributors: Kai Germaschewski, Will Fox, Yi-Min Huang, Brian Sullivan, University of New Hampshire

NERSC Repository: m148

14.1.1.1 Summary and Scientific Objectives

The Center for Integrated Computation and Analysis of Reconnection and Turbulence (CICART) has recently been renewed for a second three-year period (2010-13) to perform research in the theory of magnetic reconnection and turbulence phenomena in fusion, space and astrophysical plasmas. CICART was originally established in July 2007 under the auspices of the Department of Energy's Experimental Program to Stimulate Competitive Research (DOE/EPSCoR). The purpose of the DoE/EPSCoR program is “to enhance the capabilities of designated states to conduct nationally-competitive energy-related research and to develop science and engineering human resources in energy-related areas to meet current and future needs.”

CICART is a partnership between the University of New Hampshire (UNH) and Dartmouth College. The principal participants of CICART constitute an interdisciplinary group, drawn from the communities of applied mathematics, astrophysics, computational physics, fluid dynamics, fusion, and space physics. It is a main premise of CICART that fundamental aspects of magnetic reconnection and turbulence in fusion devices, smaller-scale laboratory experiments, and space and astrophysical plasmas can be viewed from a common perspective, and that progress understanding any of these interconnected fields is likely to lead to progress in others. For example, fast and impulsive reconnection during sawtooth crashes in tokamaks, substorms in the Earth's magnetotail, and solar flares — problems that span a very broad range in plasma parameters and boundary conditions — can all be viewed within the framework of a generalized Ohm's law. A common theme that emerges from our recent reconnection studies is that the secondary instability of thin current sheets that mediate reconnection dynamics is key to a more complete understanding of the phenomena cited above in both laboratory and space plasmas.

This project has a dual mission: it seeks fundamental advances in physical understanding, and proposes to achieve these advances by means of innovations in computer simulation methods and theoretical models, and validation by comparison with laboratory experiments and space observations. CICART’s research program, developed in consultation with its Advisory Committee, has two elements: niche areas in the physics of magnetic reconnection and turbulence which build on the accomplishments of the

CICART group and to which the group is well-positioned to contribute, and high-performance computing tools needed to address these topics. In the following, two focus areas are presented that are particularly computationally intensive:

Reconnection and secondary instabilities in large, high-Lundquist-number plasmas

The problem of fast reconnection in high-Lundquist-number[10] plasmas has attracted a great deal of attention since the inception of the classical Sweet-Parker [*Sweet* 1958, *Parker* 1957] and Petschek [1964] models, both of which are based on resistive MHD. In the Sweet-Parker model, the reconnection layer forms an elongated current sheet and the steady-state reconnection time scale is given as $\tau_{SP} = S^{1/2}\tau_A$, which increases rapidly with *S*. (S is the Lundquist number and τ_A is the Alfvén wave crossing time) For weakly collisional systems such as the solar corona, the Lundquist number is typically very large and hence, the time scale τ_{SP} is of the order of years, which is much too long to account for fast events such as flares. A similar difficulty appears when the Sweet-Parker model (equivalently, the Kadomtsev model [*Kadomtsev* 1975]) is applied to the problem of sawtooth crashes in high-temperature tokamaks, where the predictions of the model for the sawtooth crash time is significantly larger than that observed. This inadequacy of the Sweet-Parker or Kadomtsev models has led to the belief that within the framework of the resistive MHD model, it is not possible to realize fast reconnection for high-*S* plasmas.

It has been demonstrated recently that this widely accepted point of view is not correct for large systems, characterized typically by a high Lundquist number ($S \geq 3\times10^4$) [*Bhattacharjee et al.* 2009, *Daughton et al.* 2009]. In such systems, if a critical value of *S* is exceeded, there appears to be three qualitatively distinct phases during the nonlinear evolution of reconnection. In the first phase, the system evolves to a quasi-steady but transient state with the characteristics of Sweet-Parker reconnection -- an extended thin current sheet in which reconnection occurs on a rapid, super-Alfvénic secondary instability occurs and produces plasmoids (magnetic islands) copiously. It is only relatively recently that the precise scaling properties of the plasmoid instability of extended current sheets has been established [*Louriero et al.* 2007, *Ni et al.* 2008, *Samtaney et al.* 2009]. The plasmoid instability is super-Alfvénic, spontaneously produces a large number of plasmoids, and leads to the third phase in which the system exhibits rapid and impulsive reconnection mediated by a hierarchy of current sheets. When averaged over time, reconnection in this third phase proceeds at a rate much faster than Sweet-Parker, and appears to have a weak dependence on *S*. We will address the following questions:

(i) *Scaling of the reconnection rate.* Does the nonlinear reconnection rate, and hence, the reconnection electric field, become independent of the Lundquist number *S* after the system exceeds the critical Lundquist number for the plasmoid instability? Does the maximum reconnection rate depend on the system size and initial and/or boundary conditions? What is the range of values for the maximum reconnection rate? Is the plasmoid instability operative during the *m=1* tearing instability in

[10] The Lundquist number is the dimensionless ratio of the Alfvén wave crossing timescale to a resistive diffusion timescale.

tokamaks and reversed-field pinches (RFPs)? If so, what would be the experimental signatures? We will address this question by means of analysis and Magnetic Reconnection Code (MRC) simulations.

(ii) *Role of collisionless effects in the evolution of the plasmoid instability.* As the Sweet-Parker layer breaks up into copious current sheets and plasmoids through multiple levels of cascading, the width of the thin current sheets will eventually attain kinetic scales. Under these circumstances, the generalized Ohm's law must replace the resistive MHD Ohm's law. As the thin current sheets become localized and intense, and their width δ_{SP} falls into the collisionless range $\delta_{SP} \leq d_i$ (or $\delta_{SP} \leq \rho s$ in the presence of a guide or toroidal field), there will be a sudden transition from a regime of slow Sweet-Parker reconnection to a regime of fast reconnection, dominated by Hall MHD effects. This criterion has been tested and verified experimentally in the laboratory [*Yamada et al.* 2006, *Egedal et al.* 2007]. The main question that we investigate is how this criterion is modified in the presence of the plasmoid instability of extended current sheets. We will study this onset problem using the MRC as well as the PSC (Particle Simulation Code), delineating carefully the regimes of dominance of resistive MHD and fully kinetic dynamics in the context of tokamaks, RFPs, and coronal plasmas.

Fast magnetic reconnection in laser-produced plasma bubbles

CICART investigates recent experimental observations of fast magnetic reconnection between the high-energy-density plasma bubbles created by focusing terawatt-class lasers *(~kJ/ns*) down to sub-millimeter-scale spots on a plastic or metal foil. The foil is ionized into hemispherical bubbles that expand supersonically off the surface of the foil. Each bubble is found to self-generate a strong magnetic field of order megagauss, which forms a toroidal ribbon wrapping around the bubble. If multiple bubbles are created at small separation, the bubbles expand into one another, and the opposing magnetic fields are squeezed together and seen to reconnect. The rates of reconnection are observed to be fast, and unexplained by classical Sweet-Parker theory.

It is of great interest to bring these results in line with what is already known about reconnection. As mentioned above, there are a number of new features in these laser-driven experiments, such as the high energy density in the plasma and magnetic field. Perhaps their most notable feature is the very strong reconnection drive: the opposing magnetic fields are driven together by the expanding bubbles at sonic and super- Alfvénic velocities. In contrast, in most other systems the reconnection inflows typically remain sub- Alfvénic, whether reconnection is driven externally or results from plasma instability. Therefore, in addition to providing a new and complementary set of experimental observations for the reconnection problem, these experiments also present a new, strongly driven reconnection regime.

The focus of our work for the next years is to reproduce and understand in more detail the experimental observations, and eventually to develop predictive capabilities using our simulations. In particular, we have the following goals: (i) Include initial conditions

generated from radiation hydro codes (which to date have modeled single-bubble expansion, but not reconnection of multiple bubbles) rather than our simplified analytic initial conditions. (ii) Benchmark and study the inclusion of a binary collision operator on the reconnection process. The experiments are in a semi-collisional regime, where multiple collisions occur over the course of the experiment, but that the current sheets in the reconnection layer are predicted to be less than a mean-free-path wide. Therefore, including the collision operator is necessary for quantitative modeling of the experiments. (ii) Cover a suite of bubble sizes up to the largest sizes encountered in the experiments (roughly $L/d_i \sim 100$, where d_i is the ion skin depth), using realistic physical parameters obtained.

14.1.1.2 Methods of Solution

Fluid models

CICART employs a number of codes for solving plasma fluid models. HMHD is written in Fortran and solves the resistive and Hall MHD equations in simple geometries. The equations are discretized using finite difference stencils and time integration is explicit. HMHD has been shown to scale up to 8192 processors on NERSC's Franklin Cray XT4. However, algorithmic scalability is limited by the dispersive waves in the system and the corresponding CFL condition for stability.

The Magnetic Reconnection Code (MRC) integrates the fully compressible 3D extended MHD (XMHD) equations in arbitrary curvilinear geometry. A generalized Ohm's Law includes the Hall term and electron pressure gradient, giving rise to dispersive Whistler / kinetic Alfven waves. The equations are discretized using finite-volume / finite-difference on a multi-block structured computational grid. The MRC uses PETSc for time integration and hence has access to a large variety of implicit linear and nonlinear solvers. The MRC employs automatic code generation – the discretized equations are specified at a high level and the code generator then generates C functions for evaluating not only the r.h.s. but also the Jacobian of the r.h.s. function. Time integration typically uses the Crank-Nicholson method. The resulting nonlinear equations are solved using the Newton-linesearch algorithm. Inside the Newton iteration, linear problems have to be solved. One option is to use preconditioned Krylov accelerators, however achieving good iterative preconditioning for our system of equations and geometries remains challenging. As an alternative, the MRC can build the actual sparse matrix and use parallel SuperLU or MUMPS to factorize the matrix. The factorization is reused for subsequent preconditioning steps until the system has moved too far from the previous state (inexact Newton). This approach works well for problems which can be reduced to 2D by symmetry (e.g., helical symmetry in tokamaks/RFPs), but a combined iterative / direct solver approach is necessary for larger 3D problems. The MRC has ~210,000 lines of code, out which 150,000 are automatically generated, though.

Kinetic models

For kinetic modeling, CICART uses the Particle Simulation Code (PSC). PSC is a fully electromagnetic, massively parallel 3-D PIC code that includes a binary collision operator. It has a variety of options for boundary conditions (periodic/open/wall/perfectly matched layers). The code uses a second-order leapfrog type scheme for time integration. Particles representing the distribution functions use second-order shape function, and current deposition onto the grid is handled consistently and conservatively, satisfying the charge continuity equation to round-off. The electromagnetic fields are updated on the Yee grid using Faraday's Law and Ampere's Law. The PSC has been modularized and many of the computational kernels exist in a number of variations, for example Fortran 90 and a C version using SSE2 intrinsics. PSC has shown higher than 90% parallel efficiency on a moderately sized 2D problem run on Franklin in strong scaling mode up to 8192 cores. A larger 3D problem showed near-perfect scalability. The PSC has about 40,000 lines of Fortran, C and CUDA code at the time of writing.

All of our codes are currently parallelized using an MPI-only programming model. Since parallelization is based on domain decomposition, there appears to be little opportunity to reduce memory requirements for redundant data, as basically the only shared data are ghost point values, and those are small by the surface-to-volume ratio.

14.1.1.3 HPC Requirements

Parallel concurrency, wall hours, aggregate memory, memory per core, and I/O in the table below reflect resources for a typical large production run. In addition, many smaller runs are needed both for parameter studies and setting up the larger runs, as well as other problems that CICART is working on.

14.1.1.4 Computational and Storage Requirements Summary

	Current (2010)	**In 2013**
Computational Hours	.5 M	50 M
Parallel Concurrency in typical production run	2048	20k to 200k
Wall Hours per Run	20	24
Aggregate Memory	100 GB	10TB to 55TB
Memory per Core	up to 0.5 GB	up to 0.5 GB
I/O per Run Needed	1 TB	5 TB to 50 TB
On-Line Storage Needed	2 TB	100 TB
Data Transfer Needed	5 TB	10 TB

14.1.1.5 Support Services and Software

Visualization and data I/O are increasingly bottlenecks as we scale our runs to larger numbers of degrees of freedom as well as larger number of cores. In our opinion, this is an area where a computing center like NERSC could provide more guidance and support.

For large datasets, parallel data analysis and visualization becomes essential. In our center, we have started to switch from custom I/O and analysis software to leverage existing packages, in particular HDF5 and ParaView; Visit would fall into the same category, though currently it appears that ParaView can do all we need. Writing out data efficiently in parallel requires tedious code development and testing and tuning, in particular since it is not at all clear a priori what would give best performance on a particular machine. It would be useful to publish some "Best Practices" as a guide for how to do I/O on a given machine, including performance results, and potentially providing a library that would solve the I/O problem for standard situations, e.g., for codes that use domain decomposition to spatially distribute a logically Cartesian grid.

Along with this would be support to read and visualize the data files both with existing software like ParaView and Visit, as well as providing a library that allows custom analysis codes to easily read the data. Resolving this area might be as simple as testing and endorsing an existing library that addresses these issues, e.g., SILO or XDMF and provide documentation on how to best use it in application codes running on NERSC.

Our code also use standard libraries like FFTW and PETSc, continued support of these libraries and the work needed to make them scale with the increasingly large machines is very important.

14.1.1.6 Emerging HPC Architectures and Programming Models

Some of CICART codes have been ported to specific architectures: OpenGGCM runs at very high performance on IBM's PowerXCell 8i processor, and the Particle-in-Cell code PSC now has a 2D particle pusher that is ported into NVIDIA's CUDA environment for GPGPUs.

We have an experimental version of a complete 2-D particle pusher running on NVIDIA's GPGPUs using the CUDA programming model. Whereas in our opinion GPUs are not as ideally suited for PIC as, e.g., the Cell architecture (due to the high concurrency inherent to the current deposition), we have still achieved a speedup of 16 over a single Intel Nehalem core using an NVIDIA Tesla card.

In general, these new heterogeneous architectures are promising for higher performance at lower power usage, however the porting effort is substantial. In some cases, like Particle-in-Cell, the computational kernels are very specific to the particular code in question, so there is little that can be done to alleviate the need to write custom code for each application and accelerator architecture. However, in other cases, generic tools that ease the programmers' burden could and should be developed. Clearly, for problems which are based on dense linear algebra, much of the computational effort is spent in basic linear algebra algorithms and solvers, and providing those algorithms tailored to a specific architecture may provide large gains with little application porting effort.

A second example are finite-difference / finite-volume based codes on structured grids. While the equations may vary substantially between different fields of science, the fundamental computational algorithms are very similar, basically consisting of multi-dimensional loops updating fields. In some of CICART's codes, we have been very successful using code generation that takes as input the computation to be performed, and as output generates code that evaluates the computation specifically tailored for a given architecture (e.g., SSE2 / Cell / CUDA).

In general, as the HPC world grows to large number of cores and higher complexity, it will be crucial to change the programming model from a "write everything from scratch in Fortran + MPI" approach to using modern tools and libraries that hide away most of the complexity that is involved in creating highly parallel asynchronous codes or program specific heterogeneous hardware.

15 Atomic Physics

15.1 Atomic and Molecular Collision Physics for Controlled Fusion Energy

NERSC PI: M. S. Pindzola, Auburn University; D. R. Schultz, Oak Ridge National Laboratory
Contributors:
S. D. Loch, C. P. Ballance, J. A. Ludlow, and T. G. Lee, Auburn University;
P. H. Worley, Oak Ridge National Laboratory;
J. P. Colgan and C. J. Fontes, Los Alamos National Laboratory;
M. G. O'Mullane, N. R. Badnell, and H. P. Summers,
University of Strathclyde, Glasgow, UK
C. Hudson, C. Ramsbottom, and P. H. Norrington,
Queen's University, Belfast, UK

NERSC Repository: m41

15.1.1.1 Summary and Scientific Objectives

Accurate atomic and molecular collision rates are required for modeling and designing controlled fusion devices. Over the next five years we will continue our current modeling efforts, while developing new theoretical and computational methods to run on the world's latest advanced computing platforms.

Physical effects we are studying include electron-impact excitation and ionization of atoms and diatomic molecules and their ions, dielectronic recombination of atomic ions, and heavy particle impact excitation, ionization, and charge transfer with atoms and diatomic molecules. We will develop our collision codes to gradually progress to heavy elements of relevance for fusion, such as Mo, Xe, and W. We will continue to employ a variety of perturbative distorted-wave and non-perturbative close-coupling methods depending on the complexity and the accuracy required for the atomic or molecular system under consideration. For the computationally intensive calculations we currently have production allocations at NERSC and NICS; code testing allocations at NCCS and ALCF; and pending production allocations at NCCS, HECToR, and PRACE in Julich, Germany.

Our main interface with the controlled fusion community is through the Atomic Data and Analysis Structure (ADAS) Project, with 30 members including General Atomics, ITER, JET, MPI Garching, ORNL, PPPL, Auburn U, and U Wisconsin. Our large scale fundamental atomic and molecular data is processed through the ADAS generalized collisional-radiative (GCR) codes to generate easily managed derived data in a form useful for the modeling of magnetic fusion plasmas. For example, Li GCR data was used

at DIII-D in impurity transport studies, Be and C GCR data will be used at JET in support of ITER wall studies, B GCR data will be used at ASDEX-Upgrade and Alcator C-mod to study impurity transport, Al CGR data will be used at Wisconsin's HSX, Ne and Ar GCR data is being used at DIII-D and Alcator C-mod in support of divertor cooling and disruption mitigation studies, and future heavy element GCR data, including W, will be used at JET and ASDEX-Upgrade in support of ITER wall and divertor facing studies.

We are supported by grants from the US Department of Energy, the European Atomic Energy Community, and the International Atomic Energy Agency.

15.1.1.2 Methods of Solution

Our most computationally demanding codes use the R-Matrix with Pseudo-States (RMPS) and Time-Dependent Close-Coupling (TDCC) methods to solve the Schrodinger and Dirac equations for a range of collisional processes involving atoms and molecules present in fusion plasmas.

The RMPS method is a basis set approach to solving the time-independent Schrodinger and Dirac equations. Based upon accurate structure calculations, the method requires the formation and diagonalization of large symmetric Hermitian matrices for all eigenvalues and eigenvectors. Current problems require up to 80,000 processors. The recent electron-impact high n-shell ionization of Boron required over 1,000 terms in the close coupling expansion to accurately model the ionization from the high angular momentum n=4 and n=5 shells. The disk space required for the resulting Hamiltonian matrices exceeded 2 TB. For light species, current development is underway to address the problem of ionization from the highly excited open p shell configurations of Carbon. For heavy species, the scale of the problem is an order of magnitude larger and shall push future computer architectures to their limits. Preliminary electron-impact excitation/ionization studies for Mo+ involving 6,000-9,000 close-coupled channels shall be used as a test case for the iso-electronic tungsten studies to follow. We can envision employing 500,000 processors in a 3-5 year time frame.

The TDCC method is a lattice approach to solving the time-dependent Schrodinger and Dirac equations. The method employs a mixture of explicit and implicit propagation techniques on multi-dimensional grids related to the number of active electrons and nuclei in the problem. For example, electron-impact single ionization of an atom on a 400 $\times$ 400 point lattice using a core for each 20 $\times$ 20 zone needs 400 processors, while electron-impact double ionization of an atom on a 400 $\times$ 400 $\times$ 400 point lattice using a core for each 20 $\times$ 20 $\times$ 20 zone needs 8,000 processors. A recently developed code for α + He collisions tracking double charge transfer on a 10^6-point lattice could easily make use of 10^6 processors. We note that classical mechanics calculations of flow dynamics are local in nature and can make use of adaptive mesh refinement strategies. On the other hand, quantum mechanics calculations of particle dynamics are non-local in nature and not easily adapted to local grid changes.

The RMPS codes are a large suite of codes, with codes for non-relativistic, semi-relativistic and fully relativistic scattering. The length of the suite is of the order of 100,000 lines. The TDCC codes are smaller with each TDCC code < 10,000 lines.

15.1.1.3 HPC Requirements

The suite of R-matrix codes is computationally demanding in a variety of ways, but matrix diagonalization and resolving the fine resonance structure inherent in electron-impact cross sections are the two most demanding. For the former, we have implemented a scheme that concurrently formulates and diagonalizes multiple Hermitian matrices across 10-80K processors. Because this scheme is very efficient at distributing matrices over large numbers of processors, we can envision using 100–500K processors in the near future as the size of the problem escalates. In terms of resolving the fine resonance structure of excitation cross sections we have implemented two degrees of parallelism in calculating the cross sections, one over energy and another over the partial wave. This is a simple parallel problem, with no communication between processors and therefore we are confident of this scaling also to 200-400 K processors. The R-matrix method requires significant amounts of disk space to store the resulting cross sections. For example, an ongoing electron-impact ionization study on Carbon involves 1,436 terms in the close coupling expansion, which results in over one million possible transitions that must be stored.

Recent TDCC calculations [Ludlow et al., PRA 79, 032715 (2009)] for the electron-impact single ionization of the ground state Mg atom used a 384×384 point lattice partitioned over 64 processors to calculate partial cross sections for 160 different incident energies and LS partial waves. Future TDCC calculations for single ionization of the excited states of atoms will need a much larger lattice with up to 1,000 processors. Recent TDCC calculations [Pindzola et al., JPB 43, 105204 (2010)] for the electron-impact double ionization of the ground state Be atom used a $192 \times 192 \times 192$ point lattice partitioned over 13,824 processors to calculate partial cross sections for 30 different incident energies and LS partial waves. Future TCCC calculations for double ionization of atomic ions will need a much larger lattice with up to 100,000 processors. Recent TDCC calculations [Pindzola et al., PRA 73, 052706 (2006)] for the electron-impact single ionization of the ground state H_2 molecule used a $192 \times 16 \times 192 \times 16$ point lattice partitioned over 1024 processors to calculate partial cross sections for 108 different incident energies and lMS partial waves. Future TDCC calculations for single ionization of Li_2 will need a much larger lattice with up to 10,000 processors. An additional simple parallelization of the electron-impact ionization codes over incident energies and/or partial waves could easily make use of machines with up to 1,000,000 processors.

15.1.1.4 Computational and Storage Requirements Summary

	Current (2010)	In 2013
Computational Hours	9 M	60 M
Parallel Concurrency in typical production run	50K (large) / 2K (small)	1M (large) / 10K (small)
Wall Hours per Run	1 to 24	1 to 24
Aggregate Memory	100 GB	100 TB / 10 TB
Memory per Core	2 GB	2 GB
I/O per Run Needed	10 to 100 GB	10 TB
On-Line Storage Needed	2 TB	20 TB
Off-line Storage Needed	20 TB	100 TB
Data Transfer Needed	1 to 10 TB per month	10 to 100 TB per month

15.1.1.5 Support Services and Software

Both the RMPS and TDCC suite of codes are written in Fortran using MPI to implement parallelization. The codes are developed within our group and utilize externally written library routines from LAPACK, ScaLAPACK, NAG, and parallel NAG. Visualization employs library routines from NCAR and GNUPLOT, and MPEG movies for the time-dependent codes.

We need ScaLAPACK routines for matrix diagonalisation of non-symmetric real and complex matrices. We would also like to have ScaLAPACK FFT routines to supplement the current parallel NAG FFT routines.

Both suites of code would benefit from improved communication between processors and larger amounts of memory per core.

15.1.1.6 Emerging HPC Architectures and Programming Models

We shall continue to experiment with OpenMP to assess whether it helps with nested Fortran DO loops for small numbers of cores. For larger numbers of cores and for GPUs, we will begin experiments with the CUDA/Fortran approach to nested Fortran DO loops.

16 Plasma Materials Interactions

16.1 Modeling Plasma Materials Interactions

PI: Brian D. Wirth, University of Tennessee
Contributors: X. Tang, LANL; K. Nordlund, University of Helsinki, D. Whyte, Massachusetts Institute of Technology
NERSC Repository: m1200

16.1.1.1 Summary and Scientific Objectives

Plasma-material interactions pose an immense scientific challenge and are one of the most critical issues in magnetic confinement fusion research. The demands on plasma-facing materials in a steady-state fusion device include extreme particle and thermal fluxes. These energetic fluxes have pronounced impacts on the topology and chemistry of the near-surface region of the material, which influence the plasma sheath potentials and subsequent incident particle flux spectra. These evolutions are also inherently multiscale in time and are likely controlled by diffusional phenomena that are influenced by the high heat loads and subsequent thermal (and stress) gradients in the material, as well as by defect micro/nanostructures induced by both the ion and neutron particle irradiation. Further complexity is introduced by the strong coupling between the plasma and material surface, each of which has a vastly different physical scale: ~10^{-9} m for surfaces and ~10^{-3} m for plasma processes.

For example, the high probability (> 90%) of prompt local ionization and re-deposition for sputtered material atoms means that surface material in contact with the plasma is itself a plasma-deposited surface, not the original ordered material and an unintended alloy at that. Likewise, the recycling of hydrogenic plasma fuel is self-regulated through recycling processes involving the near-surface fuel transport in the material and the ionization sink action of the plasma. Another very serious aspect of the plasma-wall radiation damage in fusion reactors is tritium retention. If any carbon is present in the reactor, it will erode in the form of small CH radicals and molecules, which in turn tend to stick in other parts of the reactor, forming both soft and hard carbon films. These can have very large T contents in a form, which is hard to remove, and this has in fact been the major reason why the revised ITER design projects that carbon-based materials will not be used in the first wall during D+T operation. Furthermore, the intense radiation environment (ions, neutrons, photons) experienced by materials exposed to the fusion plasma environment ensures that the material properties are modified and dynamically coupled to the plasma materials surface interaction processes. Some of the most critical plasma materials interaction issues include: i) the net erosion of plasma-facing surfaces; ii) net tritium fuel retention in surfaces; iii) H isotope and material mixing in the wall; and iv) the minimization of core plasma impurities. Furthermore, the plasma-material surface boundary plays a central role in determining the fusion performance of the core

plasma. However, while it is widely accepted that the plasma-surface interface sets a critical boundary condition for the fusion plasma, predictive capabilities for PSI remain highly inadequate.

Gaining understanding and predictive capabilities in this critical area will require addressing simultaneously complex and diverse physics occurring over a wide range of lengths (angstroms to meters) and times (femtoseconds to days and beyond to operating lifetimes). The lower time and length scales correspond to individual ion implantation and sputtering, which occurs at or near the material surface, in addition to a range of ionization and recombination processes of the sputtered neutrals and ions in the near surface sheath. At intermediate length and time scales, a wealth of physical processes are initiated, including diffusion of the now implanted ionic/neutral species, the possibility of chemical sputtering processes at the surface, the formation of gas bubbles, surface diffusion driving surface topology changes and phonon scattering by radiation defects that reduces the thermal conductivity of the material. At longer length and time scales, additional phenomena such as long-range material transport in the plasma, re-deposition of initially sputtered surface atoms, amorphous film growth and hydrogenic species diffusion into the bulk material and permeation become important. This broad palette of physical phenomena will require development not only of detailed physics models and computational strategies at each of these scales, but algorithms and methods to strongly couple them in a way that can be robustly validated. Furthermore, efficient and effective modeling techniques and frameworks need to be developed to couple the materials surface evolution and eroded material source terms into models of the sheath, scrape of layer and edge plasma physics. While present research is confined to each of these scales, or pioneering ways to couple two or more of them, the current approaches already push the state-of-the-art in technique and available computational power. Therefore, simulations spanning multiple scales needed for ITER, DEMO, etc., will require extreme-scale computing platforms and integrated physics and computer science advances.

16.1.1.2 Methods of Solution

We currently use several methods and codes to simulate plasma – materials interactions, including:

- Ab initio electronic structure calculations – Vienna Ab-initio Simulation Package (VASP): Basic alloy thermodynamic and defect properties (formation, binding and migration energies for the interaction of deposited elements with hydrogen, helium, point defects and small defect clusters) will be obtained for mixed materials (W, C, Be) from ab-initio methods involving plane wave pseudopotential codes using the generalized gradient (GGA) and local density (LDA) approximations. We predominately use VASP, which employs Vanderbilt ultrasoft pseudopotentials within the generalized gradient approximation using spin polarization.

- Molecular Dynamics (LAMMPS) codes - parallelism by MPI - massively parallel for solving Newtonian dynamics for known, short-range interatomic potentials - easily handles 100's of millions of atoms, velocity Verlet/leap-frog or predictor corrector time integration. Limited by femtosecond time steps to hundreds of nanoseconds of simulation time.

- Accelerated Molecular Dynamics/Parallel Replica Dynamcs (AMDF) – massively parallel for extended time scale molecular dynamics simulations

- Kinetic Monte Carlo codes - mostly serial, but parallelized through replicas - a variety of codes used, none is standard

- ParaSpace - parallel cluster dynamics with spatial dependence, which involves a parallel, large sparse-matrix linear solver (PARDISO) - parallelism by OpenMP - backward difference time integration - easily treats 1E7 degrees of freedom - will need to extend to $>10^{12}$

16.1.1.3 HPC Requirements

The observations of novel structural response of tungsten surfaces to mixed helium and hydrogen plasmas demonstrate the complexity involved, and the difficult in extrapolating laboratory-based experiments to fusion device performance. We anticipate that expanded computational resources will enable use to address a number of key questions that are required to predict plasma – materials interactions of tungsten surfaces in fusion energy devices. Key questions that will be resolved by high performance computing include:

i) What are the controlling kinetic processes (e.g., defect and impurity concentrations, surface diffusion, etc.) responsible for the formation of a nanoscale ‘fuzz’ on tungsten surfaces subject to high temperature He plasmas?

ii) What exposure conditions (e.g., a phase boundary map of temperature, dose, dose rate, impurities) lead to nanoscale ‘fuzz’ or other detrimental surface evolution?

iii) How much tungsten mass loss occurs into the plasma as a result of nanoscale ‘fuzz’ formation? And, finally, how can this surface evolution be mitigated?

iv) What are the controlling He–defect and hydrogen/deuterium/tritium interaction mechanisms that influence hydrogen permeation and retention?

v) What plasma impurities increase sputtering yields of tungsten? What mitigation measures are possible to reduce tungsten mass loss?

In the future we will need to treat much larger, mixed material surfaces using molecular dynamics and accelerated molecular dynamics techniques hundreds of thousands of cores with extended timescales to the millisecond and beyond, as well as developing kinetic Monte Carlo models and spatially-dependent reaction-diffusion models with highly coupled systems of differential equation involving degrees of freedom in parallel, large sparse matrices.

16.1.1.4 Computational and Storage Requirements Summary

	Current (2010)	In 2013
Computational Hours	0.6 M	50 M
Parallel Concurrency	64 (ParaSpace) 256 (VASP) 120,000 (AMDF)	5000 (VASP, ParaSpace) 500k (Large Accelerated MD)
Wall Hours per Run	96	120
Aggregate Memory	96 GB (VASP) 256 GB (Paraspace) 12 TB (AMDF)	100 TB
Memory per Core	6 GB	6 GB
I/O per Run Needed	10 GB (VASP) 100 GB (AMDF)	1 TB (VASP, ParaSpace) 25 TB (AMDF)
On-Line Storage Needed	30 GB (VASP) 1 TB (AMDF)	50 TB
Data Transfer Needed	100 GB	10 TB

16.1.1.5 Support Services and Software

The development of efficient, parallel, large sparse-matrix linear solvers, similar to PARDISO with an OpenMP parallelism and implicit, backward difference time integration will be required that can extend today's 10^7 degrees of freedom to better than 10^{12}.

Furthermore, visualization services will continue to need to be extended to make it efficient to analyze and visualize large datasets as well as evaluate the effects of parameter sensitivity studies. Currently, VisIt and IDL are used fairly extensively for visualization, but parallel visualization tools must be further developed.

16.1.1.6 Emerging HPC Architectures and Programming Models

The emergence of GPU computing will likely continue to change the programming landscape for both accelerated molecular dynamics and parallel, sparse matrix reaction-diffusion based coupled differential equations that can be designed to utilize relatively small amounts of memory per core. For example, one such recent application of the AMDF code performed parallel replica dynamics on 1000 atoms using 12,000 replicas on the Roadrunner platform at Los Alamos National Laboratory and achieved a petaflop utilization. In this particular case, 2.8 million core-hours were used per run with 120,000 cores and a wall clock time of 24 hours. The AMDF code used 12 TB of total memory. Utilizing the Cell microprocessors on Roadrunner, the AMDF code and parallel replica approach was able to utilize just 0.1 GB of memory per core, but had fairly sizeable data read & write I/O needs with approximately 100 GB of data transferred during the course

of a 24 hour simulation, although the individual checkpoint files were quite modestly sized at 0.1 GB.

Appendix A. Attendee Biographies

Jeff Candy is a principal scientist in the Fusion Theory Group at General Atomics, and lead developer of the GYRO code. His primary research foci are gyrokinetic and neoclassical theory/simulation, predictive transport modeling, and algorithms for parallel computing. Dr. Candy maintains a general but active interest in numerical methods, in particular those with application to plasma kinetic theory. Dr. Candy received his Ph.D. in 1994 under the supervision of M.N. Rosenbluth, was visiting scientist at the JET Joint Undertaking in the United Kingdom, and has been at General Atomics since 1998. He has received various Canadian awards, including 4 NSERC postgraduate fellowships, 2 NSERC postdoctoral scholarships, and the Sir James Lougheed Award of Distinction. In 2003, he was the inaugural recipient of the Rosenbluth award for fusion theory. He was 2008 Jubileum Professor at Chalmers University in Sweden, and was elected fellow of the American Physical Society in 2009. Dr. Candy is the author of approximately 80 refereed journal articles.

Choong-Seock (C-S) Chang is the head of the SciDAC Fusion Simulation Prototype Center for Plasma Edge Simulation (CPES). C.S. Chang is a Principal Research Physicist at the Princeton University Plasma Physics Laboratory (PPPL), and a Joint Professor of Physics at the Korea Advanced Institute of Science and Technology (KAIST). C.S. Chang is a Fellow of the American Physical Society. He serves in numerous national and international scientific committees, including the Council of the US Burning Plasma Organization (USBPO), Executive Committee for U.S. Transport Task Force (TTF), Theory Coordinating Committee (TCC) for DOE FES, Users' Council for National Center for Computational Sciences (NCCS), International Tokamak Physics Activity (ITPA), and others.

Stephane Ethier is a Computational Physicist in CPPG at the Princeton Plasma Physics Laboratory. Previously, he was a postdoctoral researcher in the Applied Physics group of the Mechanical and Aerospace Engineering Department of Princeton University, a computer consultant at INRS-Energie et Materiaux, and a research assistant at McGill University in Montreal, Canada. He received his Ph.D. from the Department of Energy and Materials Institut National de la Recherche Scientifique (INRS), Montreal, Canada. He has been the recipient of numerous awards including two postdoctoral fellowships from the Fonds pour la Formation de Chercheurs et Aide a la Recherche, the Lumonics Student Paper Competion Award, High Performance Computing Centre Award, National Sciences and the Engineering Research Council Award for Summer Research in both plasma physics and biophysics. He has published several articles in refereed journals in addition to being a contributor to international conferences.

Alex Friedman received his Ph.D. in Applied Physics from Cornell University, and then carried out post-doctoral research at U.C. Berkeley. In 1980 he joined the staff of Lawrence Livermore National Laboratory, where he is an Associate Program Leader within the Fusion Energy Program. His research interests include heavy-ion beam-driven inertial fusion energy; computational plasma physics and particle-beam physics;

computational dynamics; and numerical analysis. He is a Fellow of the American Physical Society and a recipient of the LLNL Physics Department's Distinguished Achievement Award. He currently serves as the Simulations and Theory Group Leader of the Virtual National Laboratory for Heavy Ion Fusion Science, a collaboration of Lawrence Berkeley National Laboratory (where he maintains his principal office), Lawrence Livermore National Laboratory, and the Princeton Plasma Physics Laboratory.

Kai Germaschewski is an Assistant Professor at the Space Science Center of the Institute for the Study of Earth, Oceans and Space, University of New Hampshire. Kai Germaschewski received his Ph.D. in Computational Plasma Physics from the University of Duesseldorf, Germany in 2001. His advisor was Rainer Grauer and his thesis is titled "Pulse propagation in media with anistroptic dispersion". After working one year as a post-doc at the Ruhr-University Bochum, Germany he joined Amitava Bhattacharjee's Center for Magnetic Reconnection Studies (CMRS) at the University of Iowa in 2002, working on Hall-MHD simulation codes employing Adaptive Mesh Refinement. He moved together with the group to the University of New Hampshire in summer 2003. His research aims to gain a better understanding of fast reconnection processes in two-fluid systems, applicable to laboratory as well as space plasmas. The focus of his work is on sophisticated, high performance, massively parallel numerical methods, in particular block-structured adaptive mesh refinement to efficiently resolve a large range of spatial scales and implicit Newton-Krylov-Schwarz based methods to overcome stability limitations present in explicit numerical schemes.

Stephen Jardin is the Theory Department Facilitator for MHD. He is also co-head of the Computational Plasma Physics Group and head of Physics within the Next Step Option design effort. Jardin, a Principal Research Physicist, has been on the Plasma Physics Faculty of Princeton University with rank of Professor since 1986. He currently teaches a graduate course in Computational Methods in Plasma Physics. He has been a member of the National Energy Research Supercomputer User Group Executive Committee since 1992 and became Chairperson of its Program Advisory Committee in 1999. He is also a member of the ESnet Steering Committee and Chairman of the National Transport Code Collaboration Program Advisory Committee. He is a Fellow of the American Physical Society. Jardin is the author of more than 150 refereed papers, holds four U.S. patents, and has played a key role in the development of several large magnetohydromagnetic (MHD) computer programs now widely used in magnetic fusion research. He led the MHD design effort for the Burning Plasma Experiment (BPX) and Tokamak Physics Experiment (TPX) experimental proposals. He is the leader of the physics unit of the ARIES studies. Jardin received a bachelor's degree in engineering physics with highest honors from the University of California, Berkeley, in 1970, where he was elected a member of the Phi Beta Kappa. After receiving a National Science Foundation graduate fellowship, he received a master's in physics and a master's in nuclear engineering from the Massachusetts Institute of Technology in 1973. In 1976, he received a Ph.D. in Astrophysical Sciences, Plasma Physics Section, from Princeton University.

Charlson C. Kim is a Research Scientist in Tom Jarboe's laboratory at the University of Washington. His research interests are in full orbit PIC studies of FLR effects in ICC

devices such as RFPs, FRCs, and spheromaks. He also carries out NIMROD simulations of fast kink formation and dynamics of coplanar coaxial gun in support of P.Bellan at Caltech; he supports NIMROD simulation activity at the PSI Center; and supports drift kinetic-MHD tokamak simulations of tearing modes and kink/sawtooths. He earned a Ph.D in Physics from the University of Colorado in 2003 and was a Research Associate at University of Wisconsin, Madison, where he worked with Carl Sovinec.

Dr. Zhihong Lin is a Professor of Physics & Astronomy in the School of Physical Sciences at University of California, Irvine. In 2000, he received the Presidential Early Career Award for Scientists and Engineers. He was also the recipient of the Kaul Foundation Prize for Excellence in Plasma Physics research and Technology Development in 1999 for performing advanced simulations with unprecedented realism and resolution leading to results demonstrating the positive impact of modern massively parallel computers and for outstanding contributions to understanding the physics of sheared zonal flows. He is the author of 26 refereed journal publications (10 first author articles) and a contributor to many conference proceedings. Prior to his current position, he was a a staff research physicist in the Theory Department at PPPL and a DOE Fusion Energy Postdoctoal Fellow. Dr. Lin received his Ph.D. in Plasma Physics from Princeton University - Department of Astrophysical Sciences - Program in Plasma Physics in 1996. He received a B.S. in physics from Beijing University (China) in 1989.

John Mandrekas is a Program Manager at the DOE Office of Science's Fusion Energy Sciences program, where he manages the theory and advanced simulation programs and the High Performance Computing resources. Previously, he was a Senior Research Scientist at Georgia Tech in Atlanta, Georgia, where he did research in theoretical and computational fusion plasma science and taught undergraduate and graduate level courses in plasma physics and fusion. He received a Ph.D. in Nuclear Engineering from the University of Illinois at Urbana-Champaign in 1987.

Doug McCune is the current co-head of the PPPL Computational Plasma Physics Group (CPPG) and a computational scientist with over 20 years of experience. Since the 1970's, he has been the primary developer of TRANSP, a transport code and the world's premier software for time dependent analysis of measured data from tokamak fusion experiments. The TRANSP code includes a Monte Carlo fast ion package which hahs proved effective in predicting fusion rates in TFTR and JET DD and DT experiments. He continues to play a lead role in the development and modernization of TRANSP physics packages, which will be made available for use in other codes via the National Transport Code Collaboration Modules Library Project. In addition, Doug plays a lead role in the development of scientific software by the CPPG, pushing for adhereance to standards of portability, documentation, and reusability of codes.

Linda E. Sugiyama is a member of he Research Laboratory of Electronics (RLE) at the Massachusetts Institute of Technology High Energy Plasma Physics Group. She received a BS in applied mathematics from the University of Wisconsin, Madison in 1975 and a PhD in mathematics from MIT in 1980. She became a postdoctoral associate in RLE in 1980, joining the RLE research staff in 1983. Sugiyama's research has centered on a

continuing interest in many-body interacting systems, with a focus on the physics of plasmas in magnetic fields and on the development of magnetically confined plasmas for thermonuclear fusion. Because of the complexity of these many-body systems, a great deal of Sugiyama's work has been directed toward advancing and using computational models for simulating their behavior. A major recent direction for her research has been efforts to simulate a confined plasma's time evolution of a two-fluid magnetohydrodynamic model, in which the electronics and ions are treated separately, rather than the more conventional single-fluid approach. Sugiyama, widely known and respected in the field of plasma physics, is also active in professional organizations, such as the American Physical Society for which she serves as a member of the Committee on Women in Physics.

William Tang is the Chief Scientist at the Princeton Plasma Physics Laboratory (PPPL), the national laboratory for fusion research. He is also the Associate Director for the Princeton Institute for Computational Science and Engineering (PICSciE) which was recently established at Princeton University to stimulate progress in innovative computational science via interdisciplinary alliances involving computer science, applied mathematics, and prominent applications areas in the physical sciences and engineering disciplines. After receiving a PhD. in Physics from the University of California, Davis in 1972 with dissertation research carried out at the Lawrence Livermore National Laboratory, he advanced to the Principal Research Physicist rank at PPPL and Lecturer with Rank of Professor in the Department of Astrophysical Sciences by 1979 and became a Fellow of the American Physical Society at that time. He successfully served as Head of the PPPL Theory Department from 1992 through 2004. He is currently the Director of the Plasma Science Advanced Computing Institute for DOE's Fusion Energy Sciences Program.

Frank S. Tsung is a researcher in the Department of Physics & Astronomy at the University of California at Los Angeles. His research interests include particle-in-cell simulations of wave particle interactions, including laser wakefield accelerators, laser plasma interactions in inertial confinement fusion plasmas, and wave particle interactions in space and laboratory plasmas. He received his B.S. from UC Berkeley and his Ph.D. from UCLA under the supervision of John M. Dawson. He is a principal architect is one of the code OSIRIS and is a current member of NUGEX, the NERSC User Group’s Executive Committee.

Brian D. Wirth joined the University of Tennessee Nuclear Engineering Department in July 2010 as the ninth University of Tennessee-Oak Ridge National Laboratory Governor’s Chair. Wirth was previously an associate professor at the University of California, Berkeley, which he joined in 2002 following several years as a materials scientist at Lawrence Livermore National Laboratory. Wirth leads a number of research projects funded by various U.S. Department of Energy offices to investigate the performance of nuclear fuels and structural materials in nuclear environments. The research is planned to lead to improved predictions of the longevity of nuclear reactor components and ultimately the development of high-performance, radiation resistant materials for advanced nuclear fission and fusion energy applications. Wirth received a

B.S. in Nuclear Engineering from the Georgia Institute of Technology in 1992 and a Ph.D. in Mechanical Engineering from the University of California, Santa Barbara in 1998, where he was a Department of Energy Nuclear Engineering Graduate Fellow.

Appendix B. Workshop Agenda

Tuesday, August 3		
Time	Topic	Presenter
8:00am	Arrive, informal discussions	
8:30	Welcome, introductions, workshop goals, charge to committee	Yukiko Sekine, DOE-SC/ASCR
8:50	Workshop outline, logistics, format, procedures	Harvey Wasserman, NERSC
9:00	FES Program Office Research Directions	John Mandrekas , DOE / FES
9:40	NERSC Role in Fusion Energy Sciences Research	Kathy Yelick, NERSC Director
10:40	Break	
11:00	Case Study: Plasma Turbulence & Transport	C. S. Chang
12:00pm	Case Study: MHD	Stephen Jardin, Linda Sugiyama
12:45	Working Lunch	
1:10	Case Study: Fusion Simulation Program	William Tang
2:00	Case Study: Innovative / Alternative Confinement Concepts	Charlson Kim
2:35	Break	
2:40	Case Study: Wave-Plasma Interaction	Lee Berry
3:20	Case Study: Materials	Brian Wirth
4:00	Case Study: HEDLP / Intertial Fusion Energy	Alex Friedman, Frank Tsung
4:40	Break	
5:00	Case Study: General Plasma Science	Kai Germaschewski
5:40	General Discussions	
6:00	Adjourn for the day	

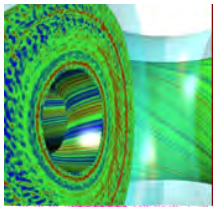
Wednesday, August 4		
8:00am	Arrive, informal discussions	
8:30	Case Study: Atomic Physics	John Ludlow
9:10	Break	
9:30	NERSC Initial Summary	Richard Gerber, NERSC
10:30	Case study format review; sample case study	Harvey Wasserman, NERSC
10:45	Report schedule and process	Richard Gerber, NERSC
11:00	Q&A, general discussions, breakout sessions, and lunch	
1:00pm	Adjourn	

Appendix C. Abbreviations and Acronyms

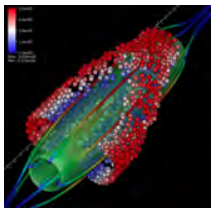
AE/EPM	Alfvén Eigenmode / Energetic Particle Mode
ALCF	Argonne Leadership Computing Facility
ALE	Arbitrary Lagrangian Eulerian
AMR	Adaptive Mesh Refinement
API	Application Programming Interface
ARRA	American Recovery and Reinvestment Act of 2009
ASCR	Advanced Scientific Computing Research
CGP	Course Graining Procedure
CICART	Center for Integrated Analysis and Computation of Reconnection and Turbulence
CSPM	Center for the Study of Plasma Microturbulence
CSWPI	Center for Simulation of Wave–Plasma Interactions
CTEM	Collisionless Trapped Electron Mode turbulence
CUDA	Compute Unified Device Architecture
DOF	Degrees of Freedom
ELM	Edge Localized Mode
EM	ElectroMagnetic
ESnet	DOE's Energy Sciences Network
FACETS	Fusion Application for Core-Edge Transport Simulations
FES	Fusion Energy Sciences Office
FFT	Fast Fourier Transform
FLOP	Floating-point Operation
FNAL	FermiLab National Accelerator Laboratory
FSP	Fusion Simulation Program
GCR	Generalized Collisional-Radiative
GPGPU	General Purpose Graphical Processing Unit
GPU	Graphical Processing Unit
GSEP	Center for Gyrokinetic Simulation of Energetic Particle Turbulence and Transport
HDF	Hierarchical Data Format
HEDP	High Energy Density Physics
HEDLP	High Energy Density Laboratory Plasmas
HFHI	High Frequency Hybrid Instability
HIF	Heavy Ion Fusion
HPC	High-Performance Computing
HPSS	High Performance Storage System
I/O	input output
ICC	Innovative Confinement Concepts
ICRF	Ion Cyclotron Frequency range of Frequencies
IDL	Interactive Data Language visualization software
IMEX-RK	Implicit-Explicit Runge-Kutta Method

INCITE	Innovative and Novel Computational Impact on Theory and Experiment
IFE	Inertial Fusion Energy
ITER	an international fusion research project being constructed in Cadarache, France
ITG	Ion Temperature Gradient
LANL	Los Alamos National Laboratory
LBNL	Lawrence Berkeley National Laboratory
LHRF	Lower Hybrid Range of Frequency
LLNL	Lawrence Livermore National Laboratory
MHD	Magnetohydrodynamics
MPI	Message Passing Interface
MTBI	Mean Time Between Interrupts
MTTF	Mean Time To Failure
NAG	Numerical Algorithms Group
NERSC	National Energy Research Scientific Computing Center
NetCDF	Network Common Data Format
NFA	Nanostructured Ferritic Alloys
NGF	NERSC Global Filesystem
NICS	National Institute for Computational Sciences
NIF	National Ignition Facility
NIMROD	Non-Ideal Magnetohydrodynamics with Rotation and Open Discussion Code
NSTX	National Spherical Torus Experiment
NTM	Neoclassical Tearing Mode
OLCF	Oak Ridge Leadership Computing Facility
ORNL	Oak Ridge National Laboratory
OS	operating system
PDE	Partial Differential Equation
PERI	SciDAC Performance Engineering Research Institute
PETSc	Portable, Extensible Toolkit for Scientific Computation
PIC	Particle-In-Cell
PPPL	Princeton Plasma Physics Laboratory
PSI	Plasma Science and Innovation Center
RF	radio frequency
SC	DOE's Office of Science
SciDAC	Scientific Discovery through Advanced Computing
SLAC	SLAC National Accelerator Laboratory
SRS	Stimulated Raman Scattering
TDCC	Time-Dependent Close-Coupling
TEM	Trapped Electron Mode
V&V	Verification and Validation
WDM	Warm Dense Matter
XML	Extensible Markup Language

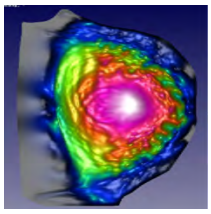
Appendix D. About the Cover



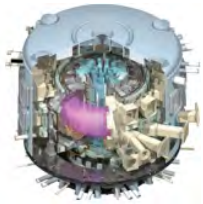
Turbulence in a torus-shaped plasma manifests itself as electron density fluctuations. Dr. Ron Waltz and Dr. Jeff Candy used the General Atomics GYRO code to compute gyrokinetic turbulence, shown in this advanced simulation.



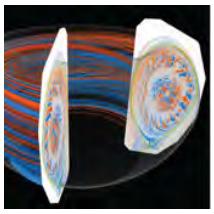
An example of a single equilibrium trajectory of an energetic ion in a Field Reverse Configuration (FRC) magnetic field showing confined and structured orbits well beyond the magnetic separatrix (denoted by green surface). Each colored sphere, colored by azimuthal velocity, represents an instant in time in the ion trajectory. Image courtesy of Charlson Kim, University of Washington



Density contours from the late stage of the crash of an Edge Localized Mode at the edge of the DIII-D tokamak plasma, showing plasma ejected into the surrounding region, under the influence of the perturbed magnetic tangle created by the instability. Substantial amounts reach the walls at top and bottom along the magnetic field lines. Linda Sugiyama, Massachusetts Institute of Technology.



A drawing of ITER, the international fusion research project being constructed in Cadarache, France, that will realize sustained burning plasmas for the first time.



Electric potential perturbation by Ion-Temperature-Gradient (ITG). The XGC1 multiscale gyrokinetic code was used to simulate the whole plasma volume including the magnetic axis in the plasma center, and the magnetic separatrix and grounded-potential material wall at the edge. The light green curves on the 2D poloidal planes show the magnetic separatrix. Visualization by Kwan-Liu Ma, UC Davis and SciDAC Institute for Ultrascale Visualization.

NERSC

U.S. DEPARTMENT OF ENERGY
Office of Science

BERKELEY LAB
Lawrence Berkeley National Laboratory