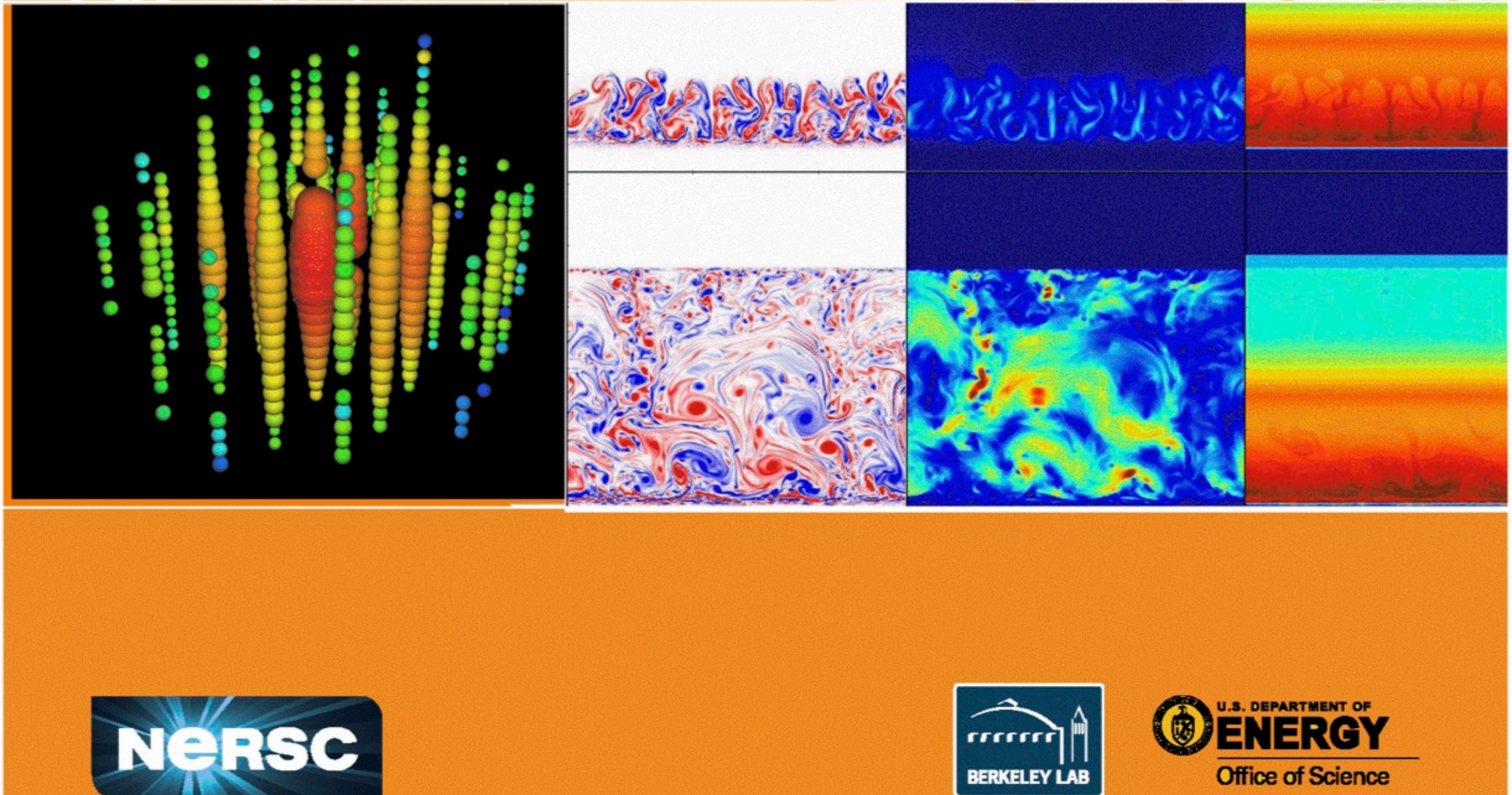
High Performance Computing and Storage Requirements for Nuclear Physics:Target 2017

Gerber, Richard
Wasserman, Harvey

Publication Date
2015-01-21

Large Scale Computing and Storage Requirements for Nuclear Physics

Report of the NERSC Requirements Review Conducted April 29-30, 2014



NERSC

BERKELEY LAB

U.S. DEPARTMENT OF ENERGY

Office of Science

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. David Goodwin is the NERSC Program Manager, and Ted Barnes serves as the Nuclear Physics (NP) allocation manager for NERSC.

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

This work was supported by the Directors of the Office of Science, Office of Nuclear Physics, and the Office of Advanced Scientific Computing Research, Facilities Division.

High Performance Computing and Storage Requirements for Nuclear Physics: Target 2017

Report of the HPC Requirements Review

Conducted April 28-29, 2014

Bethesda, Maryland

DOE Office of Science

Office of Nuclear Physics (NP)
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 5

2 Office of Science NP Mission 6

3 About NERSC 8

Meeting Background and Structure 9

4 Workshop Demographics 10

4.1 Participants 10

4.2 NERSC Projects Represented by Case Studies 12

5 Findings 13

5.1 Summary of Requirements 13

5.2 Computing and Storage Requirements 14

6 Lattice QCD Case Studies 17

6.1 Lattice QCD for Cold Nuclear Physics 17

6.2 QCD Thermodynamics at High Temperature 33

7 Nuclear Structure Case Studies 37

7.1 *ab initio* Nuclear Structure 37

7.2 *ab initio* Calculations of Nuclear Reactions and Exotic Nuclei 43

8 Nuclear Astrophysics Case Studies 50

8.1 Convection in X-Ray Bursts 50

8.2 Core Collapse Supernovae 59

9 Nuclear Science Data Case Studies 67

9.1 RHIC and LHC Heavy Ion Experiment Program Requirements 67

Appendix A. Attendee Biographies 75

Appendix B. Meeting Agenda 78

Appendix C. Abbreviations 80

Appendix D. About the Cover Images 82

1 Executive Summary

The National Energy Research Scientific Computing Center (NERSC) is the mission-science computing center for the DOE Office of Science, serving approximately 5,000 users working on some 700 projects that involve nearly 600 codes in a wide variety of scientific disciplines. In addition to large-scale computing and storage resources, NERSC provides support and expertise that help scientists make efficient use of its systems. NERSC is one of three supercomputing facilities funded by DOE's Office of Advanced Scientific Computing Research (ASCR).

In April 2014, NERSC, ASCR, and the DOE Office of Nuclear Physics (NP) held a review to characterize high performance computing (HPC) and storage requirements for NP research through 2017. This review is the 12th in a series of reviews held by NERSC and Office of Science program offices that began in 2009. It is the second for NP, and the final in the second round of reviews that covered the six Office of Science program offices. These reviews are vital for NERSC, ASCR, and the program offices in understanding future facility needs for research supported by the Office of Science.

This latest NP review revealed several key requirements, in addition to achieving its goal of characterizing NP computing and storage needs. High-level findings are:

1. Scientists will need access to significantly more computing and data resources to meet their research goals and those of the Office of Nuclear Physics.
2. NP researchers and their teams require assistance getting codes ready for Cori (NERSC's next-generation system, to be deployed in 2016) and subsequent advanced architecture systems.
3. Application teams need a supported software stack that executes well on next-generation architectures
4. NP teams need to run codes at the largest scale and also require the ability to run massive numbers of low to medium concurrency jobs.

This report expands upon these key points and adds others. The results are based upon representative samples, called "case studies," of the needs of selected group projects within NERSC. The case study topics, case study authors, and review attendees were selected by the NERSC meeting coordinators and NP program managers to represent the NP mission-science computing workload. Prepared by the NP workshop participants, the case studies contain a summary of science goals, methods of solution, current and future computing requirements, and special software and support needs. Also included are strategies for computing in the highly parallel "manycore" environment that is expected to dominate HPC architectures over the next few years.

The report from the earlier (2010) NERSC NP review is available at
http://www.nersc.gov/science/hpc-requirements-reviews/target-2014/.

2 Office of Science NP Mission

The mission of the Nuclear Physics (NP) program is to discover, explore, and understand all forms of nuclear matter. The fundamental particles that compose nuclear matter—quarks and gluons—are relatively well understood, but exactly how they fit together and interact to create different types of matter in the universe is still not fully explained. To solve this mystery, NP supports experimental and theoretical research—along with the development and operation of particle accelerators and advanced technologies—to create, detect, and describe the different forms and complexities of nuclear matter that can exist in the universe, including those that are no longer found naturally. [1]

Although nuclear physics originated as the study of atomic nuclei, this field has since broadened considerably and now encompasses the study of all forms of strongly interacting matter, as well as the use of strongly interacting systems such as nuclei to study other phenomena. The newer topics in nuclear physics include searches for new phases of nuclear matter, such as the quark-gluon plasma; aspects of nuclear astrophysics, including the formation of the heavier elements in supernovae and the properties of condensed objects, such as neutron stars; unusual, strongly interacting particles, such as hypothetical exotic mesons that contain the gluons that bind quarks; and the use of nuclei to search for evidence of new physics "beyond the standard model."

DOE supports an extensive program of experimental research in nuclear physics at the national laboratories, which is the largest component of the NP effort. This experimental work is closely coupled to a strong program of research in theoretical nuclear physics, which has the goal of interpreting the experimental results in terms of our current understanding of theory and planning future experiments and facilities to exploit advances in our understanding of the field.

In recent years, HPC has become remarkably important as a tool for promoting theoretical advances in nuclear physics and hence for interpreting the results of the experimental program. A unifying theoretical starting point for nuclear physics is provided by quantum chromodynamics (QCD); this quantum field theory describes the strong interaction of nuclear physics in terms of relatively simple basic interactions between the fundamental particles within nuclei, known as quarks and gluons. A complete solution of the equations of QCD would in principle answer most of the outstanding questions in nuclear physics. The areas of nuclear physics at the immediate frontier of applications of QCD are studies of strongly interacting particles themselves (medium energy nuclear physics) and studies of QCD phases (heavy ion collisions). Our current approach for extracting numerical predictions from this theory uses an elegant path integral method that relates QCD predictions to Monte Carlo integrals on a space-time lattice, known as "lattice QCD." Because the extreme computational requirements of this approach are a limiting factor, future progress in the direct solution of QCD will depend on the scientific community having access to appropriate computing facilities.

[1] DOE/NP Mission Statement, http://science.energy.gov/np/about

For the study of atomic nuclei, the current state of the art uses various computational methods for treating dynamical fermion systems. These range from direct quantum Monte Carlo studies for smaller nuclei through the traditional nuclear shell model for moderate sized nuclei to more phenomenological methods such as density functional theory for the largest nuclei. In addition to determining properties of nuclei, this area also contributes to "beyond the standard model" studies through predictions of transition strengths that may occur under various assumptions about fundamental particle properties. A direct connection between this traditional "low energy nuclear physics" area and the fundamental theory QCD may be achieved in the relatively near future through a realization of the Holy Grail of QCD-based calculations of nuclear forces.

Finally, research in nuclear astrophysics seeks to understand the nuclear processes that have shaped the cosmos—from the origin of the elements, the evolution of stars, and the detonation of supernovae to the structure of neutron stars and the nature of matter under extreme conditions. Furthering the DOE goal of understanding the natural world, computer simulations in these areas support DOE efforts in understanding the origin of the universe, the nature of dark energy and dark matter, and astrophysical production of exotic nuclei, as outlined in the 2011 DOE Strategic Plan.[2]

[2] U.S. Department of Energy Strategic Plan, May 2011, http://energy.gov/sites/prod/files/2011_DOE_Strategic_Plan_.pdf

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 5,000 scientists working on over 700 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 DOE's missions and scientific goals, research 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 cited 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[3]) and ASCR Leadership Computing Challenge[4] 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 HPC 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.

This report illustrates NERSC's alignment with, and responsiveness to, DOE program office needs; in this case, the needs of the Office of Nuclear Physics. 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. However, as demonstrated by its users' scientific productivity, the combination of effectively managed resources, and excellent user support services, NERSC continues its 40-year history as a world leader in advancing computational science across a wide range of disciplines.

NERSC provides an important computational resource for NP scientists. During the 2013 allocation year, about 54 NP projects computed at NERSC. These NP projects consumed approximately 215 million hours, about 10% of the total 2013 DOE-allocated time at NERSC. Additionally, NERSC's PDSF cluster, NERSC Global Filesystem, networking, and science gateway resources continue to play a vital role in NP experimental science, continuing a tradition dating back to at least 1996

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

[3] http://www.scidac.gov

[4] http://science.energy.gov/~/media/ascr/pdf/incite/docs/Allocation_process.pdf

Meeting Background and Structure

In support of its mission to provide world-class HPC systems and services for DOE Office of Science research, NERSC regularly gathers user requirements. In addition to the requirements reviews, NERSC collects information through the Energy Research Computing Allocations Process (ERCAP), workload analyses, an annual user survey, and discussions with DOE program managers and scientists who use the facility.

In April 2014, ASCR, NP, and NERSC held a review to gather HPC requirements for current and future science programs supported by NP. This report is the result of that meeting.

This document presents a number of findings based upon a representative sample of projects conducting research supported by NP. The case studies were chosen by the DOE Program Office Managers and NERSC staff to provide broad coverage in both established and incipient NP research areas. Most of the domain scientists at the review were associated with an existing NERSC project, or "repository" (abbreviated later in this document as "repo").

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 future supercomputer architectures are expected to contain multiprocessors with 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, "manycore" environment.

Requirements presented in this document will serve as input to the NERSC planning process for systems and services and 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 Nuclear Physics.

NERSC and ASCR have been conducting individual requirements reviews for each of the six DOE Office of Sciences program offices that allocate time at NERSC. A first round of meetings was conducted between May 2009 and May 2011 for requirements with a target of 2014. This second round of meetings targets needs for 2017. Reports from the previous reviews are available online at http://www.nersc.gov.

Specific findings from the NP review held in 2014 follow**.**

4 Workshop Demographics

4.1 Participants

4.1.1 DOE / NERSC Participants and Organizers

Name	Institution	Role/Area of Interest
Ted Barnes	DOE / NP	Program Manager, Nuclear Data and Nuclear Theory Computing, Office of Nuclear Physics
Sudip Dosanjh	NERSC	NERSC Director
Katie Antypas	NERSC	NERSC Services Department Head, NERSC Exascale Application Readiness Program
Richard Gerber	NERSC	NERSC Senior Science Advisor, User Services Group Lead, Meeting Organizer
Dave Goodwin	DOE / ASCR	NERSC Program Manager
Lisa Gerhardt	NERSC	User Services Group Consultant for High Energy & Nuclear Physics
Richard Carlson	DOE / ASCR	Collaboratories and middleware
Randall Laviolette	DOE / ASCR	SciDAC Partnerships
Harvey Wasserman	NERSC	NERSC Exascale Application Readiness Program, Meeting Organizer

4.1.2 Domain Scientists

Name	Institution	Area of Interest	NERSC Repo(s)
Robert Edwards	Jefferson Lab	Hadronic structure in Lattice QCD	m1383
Graham Heyes	Jefferson Lab	Experimental Nuclear Physics Computing	N/A
Raphael Hix	ORNL	Nuclear Astrophysics	m1373
Lisa Gerhardt	NERSC	NERSC High Energy & Nuclear Physics Consultant; IceCube neutrino experiment	icecube
Peter Petreczky	Brookhaven National Lab	Hot QCD	m1416
Jeff Porter	NERSC / LBNL	ALICE experiment	gc5, m1094, alice
Sofia Quaglioni	LLNL	*ab initio* calculations of light nuclei	Some overlap with m94
Martin Savage	University of Washington	Lattice QCD	m747
Sergey Syritsyn	RIKEN BNL Research Center	Lattice QCD	mp133
James Vary	Iowa State University	*ab initio* calculations of light nuclei	m308, m94
Michael Zingale	SUNY Stony Brook	Nuclear Astrophysics	m1938, m106

4.1.3 Observers

Name	Institution	Area of Interest
Hal Finkel	ALCF	Catalyst, Assistant Computational Scientist
Hai Ah Nam	OLCF	Scientific Computing Group; computational nuclear physics
James Osborn	ALCF	Nuclear & High Energy Physics, Sparse Linear Algebra
Jack Wells	OLCF	Director of Science for the NCCS at ORNL
George Fai	DOE / NP	Nuclear Theory Program Manager

4.2 NERSC Projects Represented by Case Studies

NERSC projects represented by case studies are listed in the table below, along with the number of NERSC hours they used in 2013. Note: NP resources at NERSC in 2013 included a science category "Accelerator Science" that provided allocations for six projects that used 2.4 million hours that are not represented at this review. Also not represented is a project allocated from the NERSC Director Reserve under the NP nuclear structure category for 14 million hours.

NERSC Project ID (Repo)	NERSC Project Title	Principal Investigator	Workshop Speaker	Hours Used at NERSC in 2013 (M)	Archival Data at NERSC 2013 (TB)	Shared Data on Disk (TB)
Lattice Quantum Chromodynamics (LQCD)						
m747	*Hadron-Hadron Interactions with Lattice QCD*	Martin Savage	Martin Savage	32.4	140	0.41
mp133	*Exploration of Hadron Structure using Lattice QCD*	John Negele	Sergey Syritsyn	41.6	54	0.0
mp7	*Lattice QCD Monte Carlo Calculation of Hadronic Structure and Spectroscopy*	Keh-Fei Liu	Robert Edwards	14.2	437	26.5
m1416	*QCD Thermodynamics at High Temperature*	Alexei Bazavov	Peter Petreczky	6.0	0	0.0
Nuclear Structure						
m94	*ab initio Nuclear Structure*	James Vary	James Vary / Sofia Quaglioni	47.5	28	2.2
Nuclear Astrophysics						
m106	*Core-Collapse Supernova Simulations*	Stan Woosley	Michael Zingale	4.4	173	0.82
m1938	*Convection in X-ray Bursts*	Michael Zingale	Michael Zingale	0[5]	0[5]	0[5]
m1373	*Developing an Understanding of Core-Collapse Supernova Explosion Systematics using 2D CHIMERA Simulations*	Raphael Hix	Raphael Hix	2.1	76	5.5
Data Analysis						
alice	*Data Analysis and Simulations for the ALICE Experiment at the LHC*	Jeff Porter	Jeff Porter	0.15	4	400
gc5 m1094	*STAR Detector Simulations and Data Analysis*	Jeff Porter	Jeff Porter	0.34	1,490	125
Total Represented by All Case Studies				**148 M**	**2,403 TB**	**560 TB**
All NP at NERSC in 2013 (54 projects)				**216 M**	**4,287 TB**	**1,367 TB**
Percent of NERSC NP 2013 Allocation Represented by Case Studies				**69%**	**56%**	**41%**

[5] m1938 is a new repo for 2014. The work being undertaken in m1938 was part of m106 in 2013 and earlier.

5 Findings

5.1 Summary of Requirements

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

5.1.1 Scientists will need access to significantly more computing and data resources to meet their research goals and those of the Office of Nuclear Physics.

- Science teams anticipate needing more than 5 billion hours of computing time at NERSC to conduct NP research in 2017. This is about 25 times more than they used in 2013 and consistent with the historical usage growth rate.
- NP scientists expect to store 32 PB of data in the NERSC HPSS archival store system in 2017. This need, 7.5 times what NP had archived in 2013, is growing faster than the historical trend for NP, driven by requirements for saving both simulated and experimental data.
- Research teams in NP need 6 PB of storage in the NERSC /project file system in 2017 or in a similar shared location for community data and code.

5.1.2 NP researchers and their teams require assistance getting codes ready for Cori and subsequent advanced architecture systems.

- While some teams are using GPUs for portions of their work, most of the codes in use have not been ported to run well—or at all—on manycore systems like the Intel Xeon Phi that will comprise the NERSC Cori system, which is scheduled to be deployed in 2016.
- Scientists want NERSC to provide expertise and manpower to assist the porting effort. This includes documentation, training, and consulting, in addition to dedicated coding help and/or liaisons.
- Researchers in NP want DOE to provide expert direct assistance to help port to new architectures.
- Developers need technical information about new architectures and access to early hardware and simulators.

5.1.3 Application teams need a supported software stack that executes well on next-generation architectures.

- Scientists want supported software—libraries, tools, solvers—that will run efficiently on Cori and request that DOE support multidisciplinary teams along the lines of what is available through SciDAC.
- NP codes rely heavily on basic math libraries—LAPACK, SCALAPACK, BLAS, ARPACK, PARPACK, MKL—and researchers need them to work well on Cori and future systems. Some teams are planning to begin using HDF5 for I/O.

5.1.4 NP teams need to run codes at the largest scale and also require the ability to run massive numbers of low- to medium-concurrency jobs.

- Some problems—such as gauge field generation in Lattice QCD—require full-machine scale parallelism.
- Other codes and different stages of analysis within a single code framework need to run massive numbers of jobs at lower concurrency.

5.2 Computing and Storage Requirements

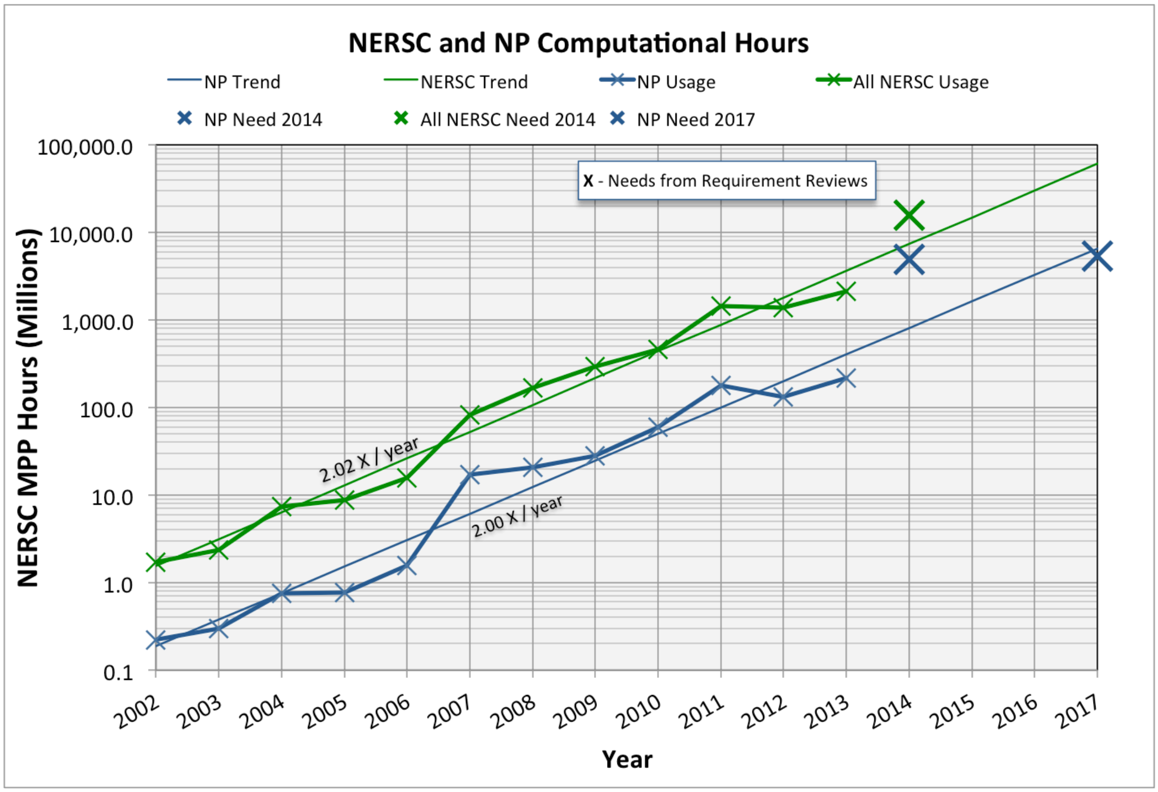
The following two tables list, respectively, the 2017 computational hours and storage needed at NERSC for research represented by the case studies in this report. "Total Scaled Requirement" at the end of each table represents the hours needed by all 2013 NP NERSC projects if increased by the same factor as that needed by the projects represented by the case studies.

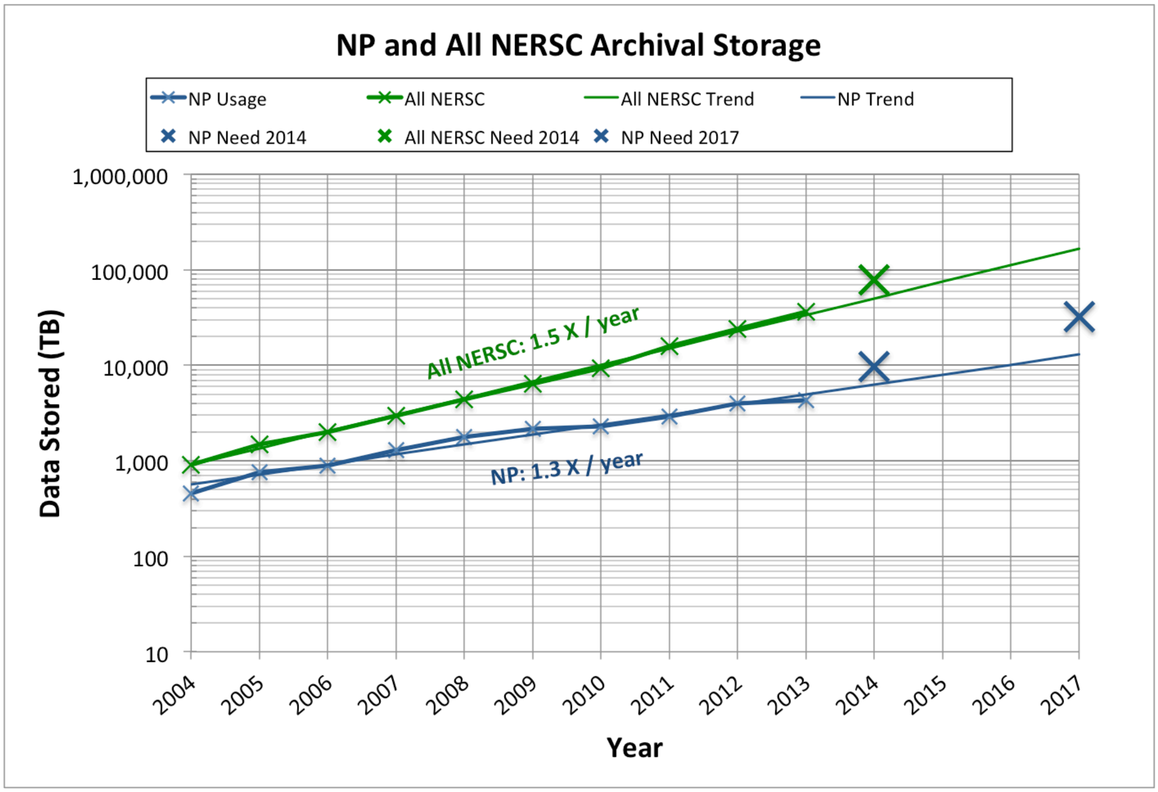
5.2.1 Computing Requirements

Case Study Title	**NERSC Repo(s)**	**Principal Investigator**	**Compute Resources Needed in 2017**	
			Million Hours	**Factor Increase Over 2013**
Hadron-Hadron Interactions with Lattice QCD	m747	Martin Savage	3,000	93
Exploration of Hadron Structure using Lattice QCD	mp133	John Negele		
Lattice QCD Monte Carlo Calculation of Hadronic Structure and Spectroscopy	mp7	Keh-Fei Liu		
QCD Thermodynamics at High Temperature	m1416	Alexei Bazavov	100	17
ab initio Nuclear Structure	m94	James Vary	246	5.2
Core-Collapse Supernova Simulations, Convection in X-ray Bursts	m106 m1938	Stan Woosley, Michael Zingale	100	23
Developing an Understanding of Core-Collapse Supernova Explosion Systematics using 2D CHIMERA Simulations	m1373	Raphael Hix	200	94
Data Analysis and Simulations for the ALICE Experiment at the LHC	alice	Jeff Porter	0.7	2
STAR Detector Simulations and Data Analysis	gc5 m1094	Jeff Porter	N/A	N/A
Total Represented by Case Studies			**3,647**	**24.5**
Percent of NERSC 2013 NP Allocations Represented by Case Studies			**68.8 %**	
All NP at NERSC Total Scaled Requirement			**5,302**	**24.5**

5.2.2 Storage Requirements

Case Study Title	PI	Repo(s)	Archival Data Storage Needed in 2017		Shared Online Data Storage Needed in 2017	
			TB	Factor Increase	TB	Factor Increase
Hadron-Hadron Interactions with Lattice QCD	Martin Savage	m747	10,000	16	1,000	37
Exploration of Hadron Structure using Lattice QCD	John Negele	mp133				
Lattice QCD Monte Carlo Calculation of Hadronic Structure and Spectroscopy	Keh-Fei Liu	mp7				
QCD Thermodynamics at High Temperature	Alexei Bazavov	m1416	200	286	10	-
ab initio Nuclear Structure	James Vary	m94	280	10	16	7
Core-Collapse Supernova Simulations, Convection in X-ray Bursts	Stan Woosley, Michael Zingale	m106 m1938	500	3	25	30
Developing an Understanding of Core-Collapse Supernova Explosion Systematics using 2D CHIMERA simulations	Raphael Hix	m1373	2,000	26	400	73
Data analysis and simulations for the ALICE experiment at the LHC	Jeff Porter	alice	14	3.4	800	2
STAR Detector Simulations and Data Analysis	Jeff Porter	gc5, m1094	5,000	3.4	250	2
Total Represented by Case Studies			**2,404**	**7.5**	**560**	**4.5**
Percent of NERSC 2013 NP Allocations Represented by Case Studies			**56 %**		**41%**	
All NP at NERSC Total Scaled Requirement			**32,000**	**7.5**	**6,000**	**4.5**


NERSC and NP Computational Hours
NP Trend
NERSC Trend
NP Usage
All NERSC Usage
NP Need 2014
All NERSC Need 2014
NP Need 2017
X - Needs from Requirement Reviews
NERSC MPP Hours (Millions)
100,000.0
10,000.0
1,000.0
100.0
10.0
1.0
0.1
2.02 X / year
2.00 X / year
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
Year


NP and All NERSC Archival Storage
NP Usage
All NERSC
All NERSC Trend
NP Trend
NP Need 2014
All NERSC Need 2014
NP Need 2017
Data Stored (TB)
1,000,000
100,000
10,000
1,000
100
10
All NERSC: 1.5 X / year
NP: 1.3 X / year
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
Year

6 Lattice QCD Case Studies

6.1 Lattice QCD for Cold Nuclear Physics

Principal Investigator: Robert Edwards, Martin Savage, Sergey Syritsyn
NERSC Repositories: m747 (nplqcd), mp133 (Negele), mp7 (Liu)

6.1.1 Project Description

6.1.1.1 Overview and Context

The structure of the proton and neutron, and the forces between them, originates from an underlying quantum field theory known as quantum chromodynamics (QCD). This theory governs the interactions of quarks and gluons that are basic constituents of the observable matter in our surrounding environment. QCD has been thoroughly tested by experiments at high energies, giving us insight into nature's workings over distances that are smaller than the size of nucleons (the term used for both protons and neutrons). However, at low energies or larger distances, the theory becomes formidable and efforts to theoretically determine fundamental nuclear physics phenomena directly from QCD have been met with less success. A long-standing effort of the DOE's Nuclear Physics program is to determine how QCD in this low-energy regime manifests itself into the observed spectrum of hadrons and the observed nuclear phenomena, and to use QCD to make reliable predictions for processes that cannot be experimentally accessed. These theoretical efforts provide critical support to the DOE's nuclear experimental projects, in particular those being executed at the Thomas Jefferson National Accelerator Laboratory (JLAB), Brookhaven's Relativistic Heavy Ion Collider (RHIC), and Michigan State's Facility for Radioactive Beams (FRIB), as well as a planned Electron-Ion Collider (EIC).

The specific goals of lattice QCD in cold nuclear physics are to:

- **Determine the spectrum of QCD.** In addition to the excited meson and baryon spectrum, this includes the search for exotic states that may exist. This is tied closely to the experimental program at JLAB, including the GlueX experiment to be performed with the 12 GeV upgrade to JLAB.
- **Determine how QCD makes hadrons and quantify their structure.** This is tied closely to a number of experimental nuclear physics programs, including the spin-structure efforts at RHIC, JLab, and a future EIC, and also impacts the high-energy physics program at the LHC.
- **Determine how nuclei and their interactions emerge from QCD, and develop technology to calculate these quantities with quantifiable uncertainties.** This will allow for reliable predictions of nuclear reactions and structure that are difficult to measure in the laboratory but play a role in extreme conditions in the cosmos or in neutron-rich environments. This includes a precise determination of the two-nucleon, three-nucleon and multi-nucleon forces. This effort will compliment the experimental program at FRIB and is crucial to refining the chiral nuclear forces.
- **Quantify the connection between the underlying fundamental symmetries of nature and experimental observables.** For a number of observables that probe fundamental aspects of nature, the strong interactions provide large, and presently unquantified, modifications to the underlying interactions. Such modifications can

be precisely determined from Lattice QCD calculations. These theoretical efforts are critical to the interpretation of precision experiments being performed at Los Alamos National Laboratory and Oak Ridge National Laboratory, which are precisely measuring the properties of neutrons in an effort to discover new physics.

6.1.1.2 Scientific Objectives for 2017

The objective of the nuclear physics program at NERSC for the 2014-2017 period is to perform the suite of calculations that address the science goals stated above at light-quark masses that produce a pion near and at the physical mass, but without strong isospin breaking and without the complete electromagnetic interactions. Ensembles of gauge-field configurations will be generated with a few lattice spacings and a few lattice volumes. These, and ensembles produced elsewhere, will be used to produce the first realistic calculations of the hadron spectrum, the structure of the hadrons, and the forces between two and three hadrons. Specifically, with the HPC resources provided through 2017, we anticipate calculations of the following:

- Resonance determination of the light quark meson spectrum, including possible exotic mesons
- Photo-couplings of light and charmonium mesons
- N* and strange quark baryons—resonance determination of low lying spectrum
- Isovector and isoscalar form factors of the nucleon and generalized parton distributions
- Individual contributions of the up, down, and strange quarks to hadronic structure and nucleon spin
- Precise determination of low-energy nucleon structure observables
- Meson-meson interactions with precision
- Nucleon-nucleon, hyperon-nucleon, hyperon-hyperon interactions
- Two-body bound-states
- Lowest-lying states in light nuclei with A<8
- Neutron electric dipole moment
- Constraints on physics beyond the standard model (when combined with planned experiments)

This list of calculations can be roughly grouped together under three specific goals (discussed above). At the physical pion mass, but in the isospin-limit and without electromagnetic interactions, the NERSC HPC resources will result in calculations of the following:

- The spectrum and structure of light hadrons from QCD
- Nuclear forces and the interactions between hadrons from QCD
- The manifestation of fundamental interactions, and possible modifications to the standard model, in the light hadrons

Estimates of the computational resources that are required to complete these calculations and calculations of other important quantities are shown in Figure 1, Figure 2, and Figure 3.

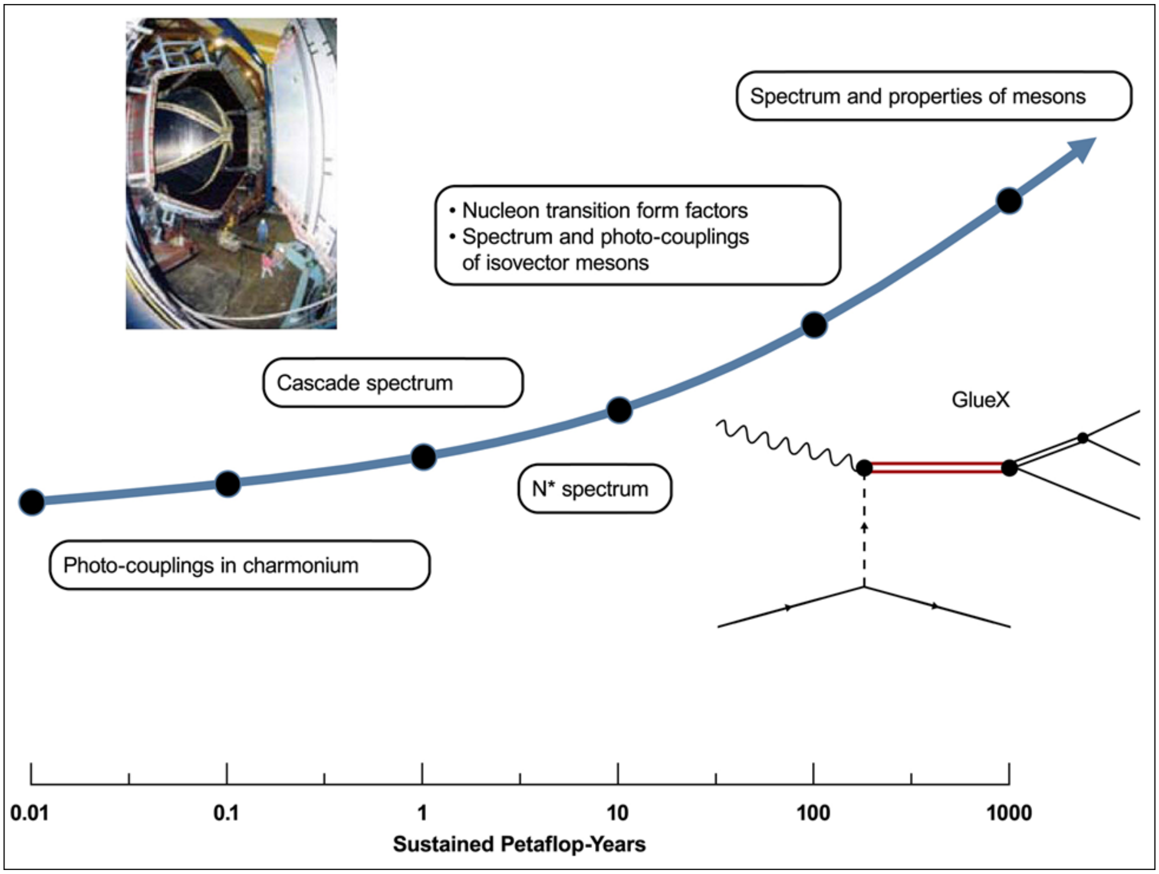


Figure 1. Estimate of the computational resources required to determine the spectrum and structure of hadrons. [Figure is reproduced from the Scientific Grand Challenges: Forefront Questions in Nuclear Science and the Role of Computing at the Extreme Scale workshop held in 2009]

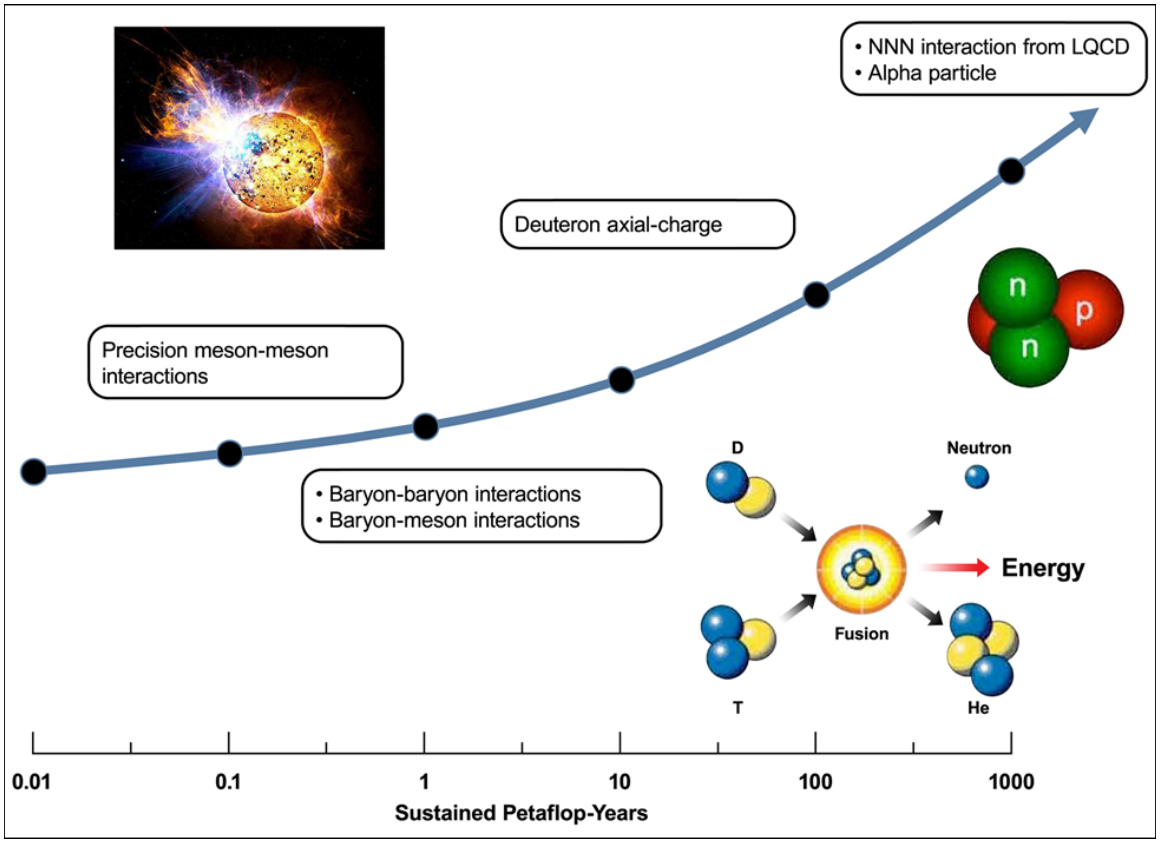


Figure 2. Estimates of the computational resources required to calculate the two and three-body nuclear forces. [Figure is reproduced from the Scientific Grand Challenges: Forefront Questions in Nuclear Science and the Role of Computing at the Extreme Scale workshop held in 2009]

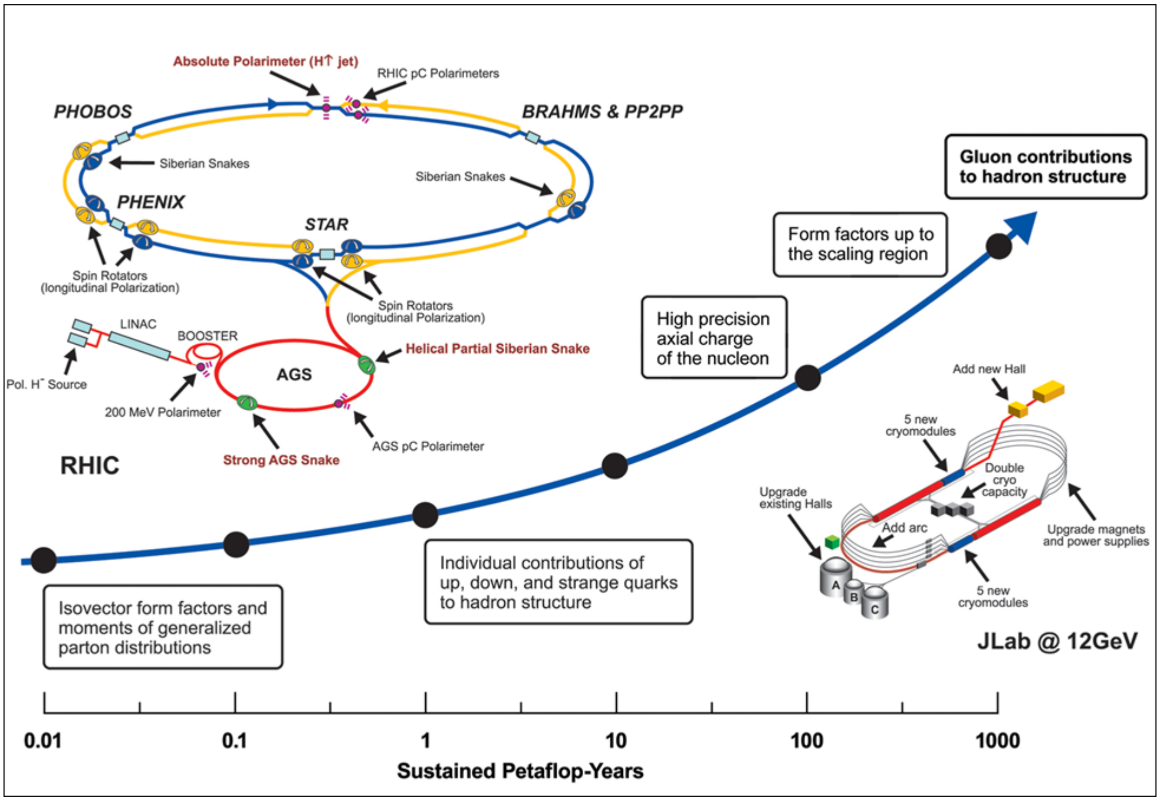


Figure 3. Estimates of the computational resources required to calculate the structure of the nucleon. [Figure is reproduced from the Scientific Grand Challenges: Forefront Questions in Nuclear Science and the Role of Computing at the Extreme Scale workshop held in 2009]

This period (2014-2017) will also see the evolution of the currently employed Lattice QCD codes toward deployment and production on exascale facilities (for instance, workflow). The resources provided by NERSC will be used coherently with other HPC resources available to the nuclear physics community, such as the capacity resources provided by USQCD (SciDAC project) at its HPC centers at the Brookhaven National Laboratory (BNL), JLab, and FermiLab, the capability resources provided by USQCD via INCITE awards, and awards from the NSF HPC centers (XSEDE and Bluewaters).

6.1.2 Computational Strategies (now and in 2017)

6.1.2.1 Approach

Lattice QCD is the numerical technique in which space-time is discretized and the path integral that dictates the quantum dynamics of the quark and gluon fields is evaluated by a combination of Monte Carlo techniques and sparse-matrix inversions. It is currently the only known method to rigorously solve QCD. Realistic calculations with uncertainties that can be systematically removed require highly optimized algorithms and cutting-edge HPC systems that currently exist, and will exist, at NERSC in the near future. The current "mode of operation'" in Lattice QCD is to split the requisite tasks into two or three distinct subtasks:

1. **Generating one or more ensembles of gauge-field configurations (lattices).** A number of ensembles with different lattice spacings and volumes are required at each given set of light-quark masses to perform the necessary extrapolation to the continuum and to infinite volume. This process has significant resource requirements, which increase with decreasing lattice spacing, with increasing volume, and with decreasing light-quark masses. Each configuration is saved to disk and archived to be used in (multiple) subsequent calculations. The multiple volumes and lattice spacings at a fixed quark mass allow for the reduction in the systematic error(s) introduced by the non-zero lattice spacing and the non-infinite volume of the gauge-fields. Gauge-field generation typically occurs over a period of months. The generation of gauge fields requires HPC capability facilities.
2. **Producing light-quark propagators on each gauge-field configuration.** In general, a large number of light-quark propagators, and also potentially low eigenvectors, need to be determined on each gauge-field configuration. Some intermediate objects may or may not be required for subsequent calculations produced at the time of propagator generation (for example, intermediate-stage hadronic blocks). These are either saved to disk and possibly archived for subsequent use or used immediately without being written to disk. Parts of these calculations can place significant demands on memory and I/O bandwidth. The generation of propagators presently requires HPC capacity facilities but can be performed equally well on capability resources.
3. **Correlation functions of quantities that will yield the desired physics are produced from the propagators.** They are produced immediately after propagator production or from propagators that have been saved to disk or archived. The correlation functions are written to disk, archived, and moved off-site for subsequent analysis that generally can be performed on workstations or small local clusters. The generation of correlation functions require HPC capacity facilities but can be performed equally well on capability resources.

6.1.2.2 Codes and Algorithms

In 2013 the Chroma lattice field theory code suite was ranked as the third code in terms of allocated cycles at NERSC. This code suite, used for lattice QCD calculations, runs on almost all parallel machines. It was designed from the ground up as part of the DOE SciDAC effort within the USQCD initiative. The Chroma software system is built over a C++ data parallel API and implementation called QDP++. This package provides an architectural independent programming API along with I/O support. The Chroma suite is designed following modern software engineering programming practices with regression tests and nightly builds. Significant effort has gone into optimizing architecturally specific, time-critical routines.

Lattice QCD uses a four-dimensional rectangular grid to represent space and time. The grid is divided into equal domains for each processing node. The grid spacing and size of the grid are parameters in the theory. The quarks are site variables, while the “gluons” —the field configuration—are situated on the links. For a given set of input parameters of the theory—namely, the lattice spacing, volume, and quark masses—an observable is computed by averaging over all possible values of the field configurations. A Monte Carlo method is used to produce the most important set of configurations contributing to this average. These importance-sampling methods allow for a systematically improvable calculation of an observable.

In broad terms, there are two main classes of computations: the generation of the gauge field configurations requiring large-scale capability resources, and the analysis of these configurations requiring either capability or capacity resources. The Chroma code suite is actually a library, with parts of the code supporting both of these types of calculations.

Field configurations are generated by a molecular dynamics evolution of the gauge field through an artificial simulation time. The time-consuming part of the calculation is the computation of the force coming from the dynamical quark fields, which is a non-local force involving the gluon fields. Computations of this force require the solution of large sparse linear systems of equations; namely, the solution of a Hermitian positive-definite sparse matrix problem.

As a part of the USQCD SciDAC initiative, significant effort has gone into better mathematical formulations of the gauge updating (a Hamiltonian integrator). The formulation of QCD, and the demands that the integrator satisfy reversibility and be area-preserving, limit the extent to which improved mathematical formulations can be borrowed from standard ODE-s. Nevertheless, over the last few years, improved methods have resulted in about a factor of 10 increase in the performance of the integrator.

Once the gauge-field configurations are computed and stored, subsequent calculations of observables will typically be dominated in cost by repeated solution of these matrix problems. Some calculations may involve the intermediate storage of large files scaling with the lattice volume. These files are later staged back in and processed into numerous, but smaller, files.

Lattice grid sizes in use now range up to 72x72x72x256. The gauge field variables on each link are 3x3 complex matrices, and the site variables—the quarks—are represented as 3x4 complex matrices.

The Chroma codes can be compiled in several architectural modes supporting a hybrid communications and threads model. The parallel version is built over a communications package called QMP that was developed as a part of the USQCD SciDAC effort. This communication package is implemented in MPI as well as other architecturally specific hardware communication variants. The thread model is implemented in either OpenMP or a package called QMT that was also developed as a part of SciDAC. The QMT package has been implemented in Pthreads with architecturally specific thread semaphore routines that are found to outperform OpenMP on some platforms. The choice of computational model is usually suggested by the hardware architecture.

Recently, the underlying component of the Chroma code (QDP++) has been implemented using a just-in-time compilation framework that can generate and execute code on GPUs. This code is in production now utilizing 4,000 GPU nodes, or 120,000 equivalent cores, in the generation of gauge fields on 72x72x72x256 lattice grid sizes on the Titan system at ORNL. The code uses a hybrid MPI/OpenMP model, where the OpenMP directives are generated automatically from within the QDP++ code. The code is linked against the QUDA library that provides high performance matrix linear system solvers for GPU systems. The just-in-time compilation system has been extended to use the LLVM compiler framework, allowing for gauge generation on Cray XE systems as well as the Bluegene/Q systems.

The analysis phase of the calculations is typically composed of a large number of matrix linear system solutions, followed by the contraction of the solution vectors to produce correlation functions—arrays of complex numbers. The deployment of large GPU-based systems has had a profound impact on these analysis phases. In particular, the high-performance linear system solvers available within the QUDA package have given more than a 10x reduction in the time-to-solution in these critical parts of the calculations. Recently, the deployment of an algebraic multigrid code linear solver in the QOPQDP package has also shown a 10x reduction in the time for matrix solutions for CPU based systems. Typical job sizes for solution of a single matrix are a few (10s) of nodes, but larger numbers of solutions can be obtained on larger partition sizes, such as 10,000 to 20,000 cores. The increased capacity afforded by these improved methods has resulted in a significant increase in I/O demands in both intermediate and long-term storage. Intermediate datasets are approaching 1 TB. Parallel and global file systems available to hold longer term datasets are being used. A common such system is Lustre, but other HPSS systems can be used.

Another technique employed in the analysis phase is acceleration of the linear system solutions by deflation of low-mode eigenvectors. The algorithm allows for the initial construction and successive refinement of the eigenmodes while determining the matrix solutions for each of the many required right-hand sides. Crucially, the low eigenmodes can be employed in the analysis phase in a method call all-mode averaging (AMA) and allows for a dramatic increase in the statistical precision of the final correlation functions by a factor of 10x to 20x. Combined with deflation, the overall computation is accelerated by a factor of 100x.

The computation of the eigenvectors is done with ARPACK, and the deflation and analysis methods used are implemented in the Qlua software package. The large number of eigenmodes places large demands on memory and I/O throughput. For a single gauge configuration, the intermediate storage can reach 20 to 35 TB in size. Thus, high-performance I/O becomes crucial on the large problem sizes.

6.1.3 HPC Resources Used Today

6.1.3.1 Computational Hours

At NERSC, we have in MPP units 30 million (NPLQCD), 40 million (LHP), and 18 million (Spectrum), for a total in the vicinity of 90M MPP units. Outside of NERSC, we have allocations on USQCD, INCITE and XSEDE resources in 2014.

NPLQCD:

USQCD: 41M CPU core-hours and 320K GPU node-hours

XSEDE: 25M CPU core-hours

LHP:

USQCD: 20M CPU core-hours

Spectrum:

USQCD: 34M CPU core-hours and 3,500K GPU node-hours

INCITE: 58M ORNL core-hours

If we consider one USQCD core-hour to be about the same as a NERSC MPP unit and add the INCITE allocation, we have a total of about 211 million MPP node-hours. The USQCD GPU node-hour allocation is 3.8 million.

6.1.3.2 Parallelism

Gauge generation typically uses about 40,000 to 60,000 cores on NERSC systems. Further analysis jobs sizes range from a very small number of cores to in excess of 64,000 cores. Analysis calculations with AMA/deflation require 512 nodes of Edison and 1,024 nodes on Hopper.

The maximum limits are determined by the smallest local volume per compute element, which is considered to be 2x2x2x2. Thus, for the 72x72x72x256 lattice gauge generation, the maximum number of cores that can be used is about 3 million. For the analysis portion, the same maximum holds.

Severe strong scaling sets in at the largest possible core counts. For this reason, most of our running is on partitions that are smaller than the maximum possible. This is because some of the subprocesses in the production work flow run efficiently on smaller partitions. More reasonable scaling is found for a local volume of 4x4x4x4, where the calculations could be scaled to 370,000 cores for the largest problem size. For the NERSC systems, problem sizes such as 48x48x48x96 are in use, resulting in 40,000 partition sizes.

Both strong and weak scaling are important to our research programs. Given the type and size of the problems we have been recently running at NERSC and what we expect to obtain from INCITE resources on leadership facilities, we anticipate requiring better weak scaling properties compared with strong scaling.

6.1.3.3 Scratch Data

We require 100 TB of temporary disk space for our current production, given our allocation(s).

6.1.3.4 Shared Data

NPLQCD used 27 TB of data in the following directory in 2013:

/project/projectdirs/nplqcd

This directory is used for sharing data among users and for storing configurations.

6.1.3.5 Archival Data Storage

We stored 632 TB of data in the NERSC HPSS archival storage system in 2013.

6.1.4 HPC Requirements in 2017

6.1.4.1 Computational Hours Needed

In the 2011 NERSC case studies, we estimated that we required about 1 billion hours per year by 2014 to accomplish our stated goals by 2014. In retrospect, these estimates were quite accurate, and consequently, we are approximately one year behind schedule in delivering the described science. To accomplish the goals we have set out for 2017, we estimate that 3 billion hours will be required per year by 2017. This represents essentially a Moore's Law growth of resources used at NERSC in 2011, which has been the underlying assumption in projecting resource requirements but represents a significant increase over the allocations obtained between 2011 and 2014.

We expect to obtain resources from USQCD and also INCITE for this research direction. Historically, the USQCD collaboration has received allocations that are greater than 3% of the available combined INCITE resources of ALCF and OLCF.

We will certainly require more compute hours in 2017 than in 2014. This is driven by the need to reduce the mass of the light quarks to their physical values to produce quantities that can be directly compared to nature and to make predictions for extreme environments. In turn, the associated reduction in the pion mass requires larger lattice volumes for the calculated systems to be well-contained within the lattice volume.

It is important to note that the resource requirements we have presented for 2017 are insufficient to fully quantify all of the uncertainties associated with these lattice QCD calculations at the physical light-quark masses. We estimate that more than *100 billion* hours are required to reliably estimate the lattice spacing uncertainties at the physical point, and precision calculations with a complete quantification of all uncertainties is currently estimated at a sustained 100 Pflop years.

6.1.4.2 Parallelism

We expect to run jobs from small core counts to partitions on the order of 500,000 cores. The maximum we could use is > 3 million cores.

6.1.4.3 I/O

Our applications do not have built-in checkpoint/restart, but a similar effect is achieved by the nature of the workflow, namely task-level restarts. For the analysis phase, the basic unit of work is a configuration, usually a collection of matrix solutions. If these collections fail, then they can be re-evaluated.

The requirements for the eigenvector computations are more stringent though, going through third-party packages, and overall the codes do not allow for checkpointing.

For gauge generation on 96x96x96x256 lattices, we need:

150 GB per configuration and about 1,000 configurations, leading to ~150 TB.

For the propagator and analysis portion:

150 GB (read per configuration)

100 x 1.5 TB (intermediate write per configuration for ~ 100 propagators)

300 GB (final write per configuration, and need 1000 configurations)

Checkpoint: gauge generation and analysis ~ 300 TB.

For the gauge generation portion, except for an initial load, the code is writing only 150 GB about every 20 minutes. The code blocks on this write.

For the propagator portion, the code blocks on a 150 GB read (done once) and then must write the 1.5 TB every minute or so.

Whenever deflation is used, the code will have to load 35 TB at the start of the job. To avoid idling, peak available bandwidth (as high as possible) will be required at the start. Less than about 10% of the runtime would be acceptable for the I/O portion of the code.

At the current peak SCRATCH I/O bandwidth on Edison (~150 GB/sec), reading 35 TB will take ~4 min. Lattice QCD jobs are typically longer than 1 hour, so we expect the I/O to take less than 7%.

6.1.4.4 Future Data Needs

In 2017, we expect to need <1,000 TB of temporary scratch disk space, <1,000 TB of NERSC project space (globally accessible shared data), and <10,000 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to increasing lattice volume, reduced lattice spacing, and reduced light-quark masses.

6.1.4.5 Memory Required

New algorithms that are being employed, such as all mode averaging, are memory intensive, currently requiring ~3 GB/core on 16,000 core partitions. While this algorithm is not required, it is desirable for some calculations, and therefore it would be desirable to have more than 3 GB/core on partitions > 50,000 cores, and hence > 50 TB of memory per run.

6.1.4.6 Emerging Technologies and Programming Models

Our codes make extensive use of CUDA directives. The Chroma code, using the Just-In-Time compilation framework, can generate Nvidia PTX code and, using the Just-In-Time compiler, can dynamically produce and execute code on the GPUs. All lattice grid based operations are executed on the GPUs, and data is transferred to the front end as necessary, such as for MPI communications. The code is linked against the QUDA QCD library, which is implemented directly in CUDA. At runtime, the QUDA code dynamically tunes the block sizes that will give optimal performance on the particular GPU system.

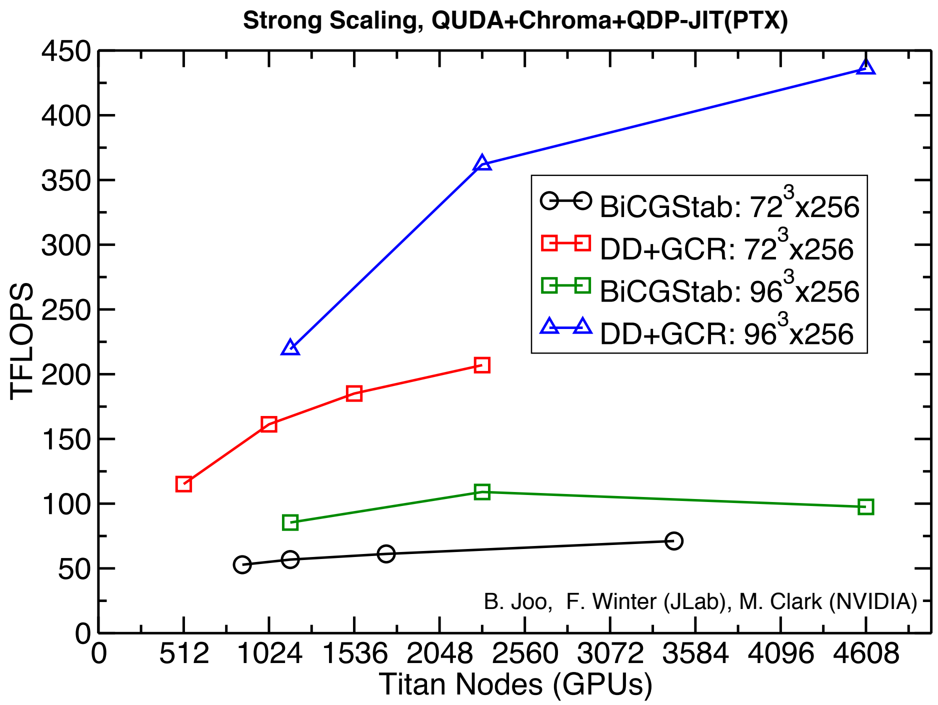


Figure 4. Sustained strong scaling performance in Gflops of a Wilson-clover mixed precision BiCGstab, and DD-GCR, a GCR solver with a domain decomposed preconditioner that minimizes communications.

The development of domain-based formulations of the gauge field integrators and inverters is precisely aimed at these heterogeneous architectures. Crucial to these methods is the importance of the characteristic length scale within QCD. This length scale, basically the distance for confinement of quarks within QCD, indicates the minimal size by which domains of space-time can be decomposed and isolated from other domains. These domains of space can be allocated to a computational node with fast local communications (say, through memory) but relatively slow off-node communications. This situation is quite characteristic of GPU embedded accelerators.

A generalized conjugate residual with a Schwarz-based decomposed preconditioner is in production now, and strong scaling results are shown in Figure 3. A sustained performance of ~200 Tflops is achievable using the domain-based preconditioner on the target lattice size of current calculations, and higher performances (~450 Tflops) are achievable on larger lattice sizes. These results are a strong indication that such decomposition techniques are appropriate. More refined decomposition methods involving Algebraic based multi-grid methods are now implemented and deployed, but so far only on CPU-based systems. The performance per socket is roughly comparable between the current GPU-based systems using a more conventional BiCGstab based Krylov solver and the newer multigrid systems. Current work is directed toward implementing the multigrid systems directly onto GPUs.

The development and implementation of these domain-based methods is crucial to achieve high performance and lower time to solution on future large HPC systems with their distinct memory hierarchies. Further refinement is expected in the methods used for spare matrix solves in the analysis phase of the calculations, resulting in greater throughput. First results have already shown that such domain-based methods are very effective in the integrator portion of gauge generation, and it is expected that such techniques will be incorporated soon into USQCD codes, including the Chroma code.

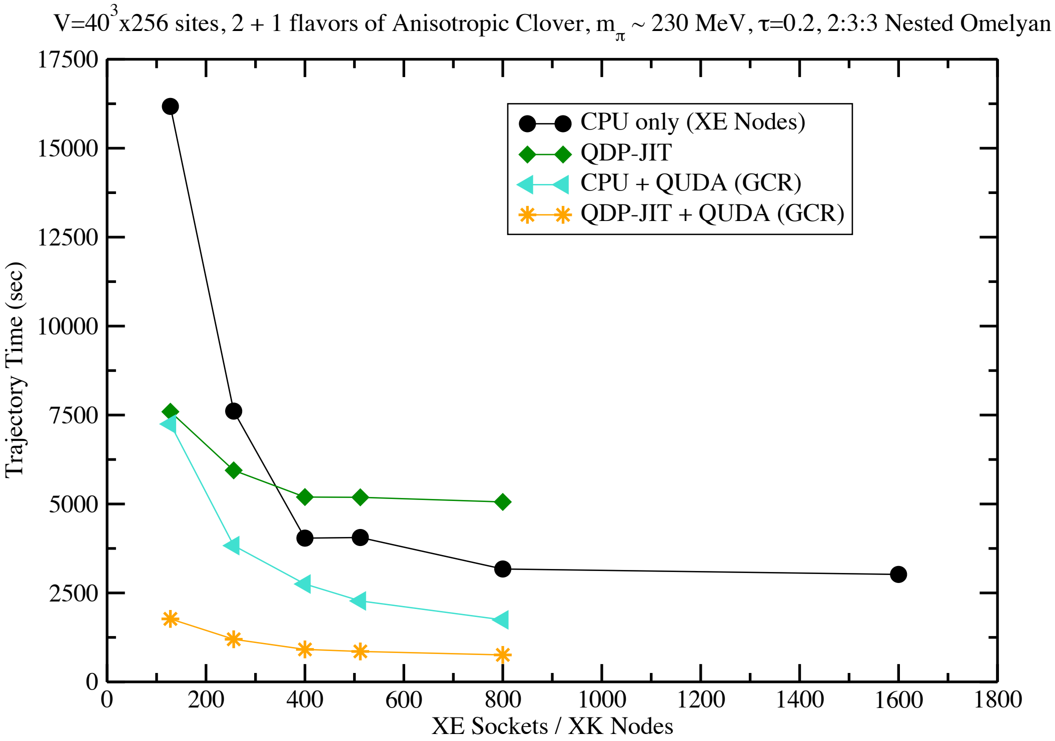


Figure 5. Strong scaling for the time to solution of one gauge configuration trajectory. Shown are the times with only CPUs, the new QDP-JIT compilation framework for GPUs which dynamically compiles, offloads and executes all lattice based operations directly onto the GPUs, the time with the code only including the QUDA solver, and both the QDP-JIT and QUDA solver. Significant speedups can be obtained by combining methods, demonstrating the efficiency of the methods at mitigating Amdahl's law. [Figure reproduced from "A Framework for Lattice QCD Calculations on GPUs," F.T. Winter, M.A. Clark, R.G. Edwards, B. Joo, to appear in IPDPS'14.]

The Chroma code is now in use in gauge generation production on the Titan system. Strong scaling results are shown in Figure 4. When only the improved matrix solver in QUDA is used, performance increases are observed for small partitions, but strong scaling is curtailed due to Amdahl's law. The new JIT compilation system lowers these performance penalties but offloads more code to the GPUs and leaves data within the GPU memory.

Reasonable performance is obtained up to 400 nodes for the smaller problem size (40x40x40x256) used in these calculations. For the target lattice size now (72x72x72x256), we expect reasonable performance to about 4,000 nodes.

OpenMP directives are used extensively in the code, and the hybrid MPI/threading model is crucial to obtaining good performance. Our software runs in production now on Mira using threading. Extensive work is under way to improve the performance through an LLVM back-end code generator.

We have a significant partnership with Intel Parallel Computing labs to implement QCD codes on the MIC architectures. Several publications have already appeared. Figure 5 shows comparisons of performance of a highly optimized conjugate gradient iterative linear system solver on various lattice volumes and systems, including Xeon (Sandy Bridge) and Xeon Phi systems, as well as an optimized solver for the NVIDIA K20m. The results show that comparable levels of performance are achievable on a Phi and Nvidia GPU system. Our partnership with Intel is moving toward deploying a suite of optimized multi-node linear system solvers suitable for Knights Landing based systems, such as NERSC-8 (Cori). Current work, as presented in the ISC'13 paper, have already shown strong scaling results for Knights Corner based systems such as Stampede.

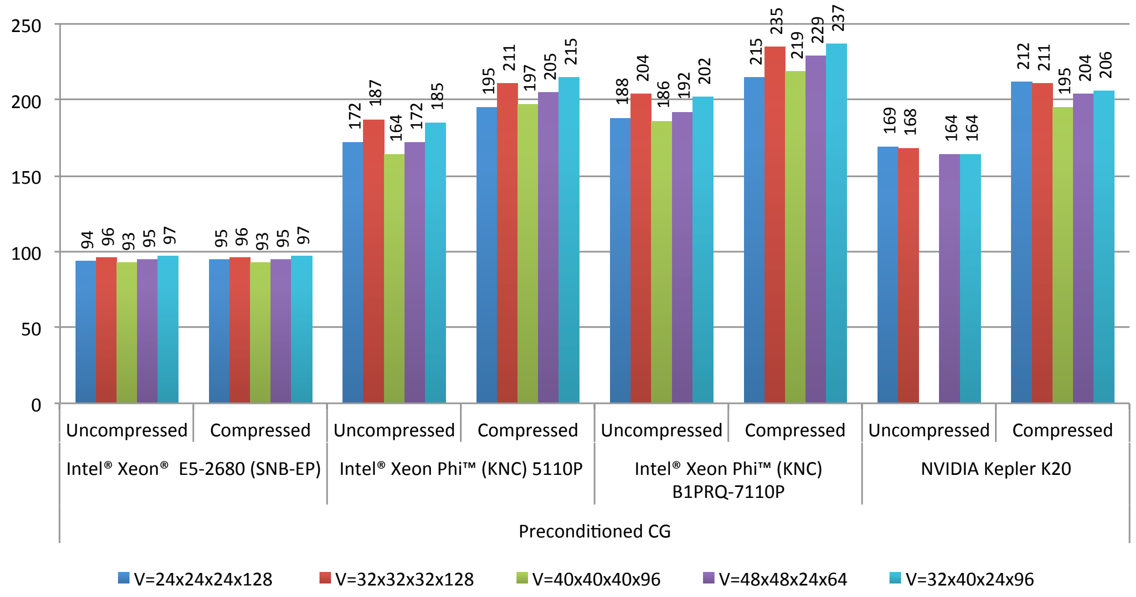


Figure 6. Performance of a conjugate gradient iterative solver on various lattice volumes for Xeon E5 (Sandy Bridge), XeonPhi 5110P, Xeon Phi 7110P, and NVIDIA K20m. The vertical axis shows performance in Gflops. [Figure reproduced from "Lattice QCD on Intel Xeon Phi Processors," B. Joo, P. Dubey, K. Vaidyanathan, M. Smelyanskiy, K. Pamnany, V. Lee, P. Dubey, W. Watson, in ISC'13.]

Regarding the question of plans for other funded groups or researchers engaged to help with these activities, a component of the current Nuclear Physics Lattice QCD SciDAC-3 software project is devoted to porting and optimizing codes to GPUs and MIC architectures. There has been significant progress during the last five years in migrating code to GPUs, and a concerted effort is currently under way for the MIC architecture. These efforts have included a collaboration with the SciDAC FastMath and Super Institutes. In addition, collaborations with Nvidia and Intel are ongoing.

We believe that NERSC should support the activities related to the transition to these architectures by providing fulltime people, physically located at the PI's institution, until the porting is complete. Our SciDAC-3 software project was not supported at the requested level, and as a result we have less than an optimal number of FTEs dedicated to translating our codes to GPU and MIC architectures.

DOE, NP, and/or ASCR should also provide more resources to support people to help with the porting to these architectures. Transitioning to new architectures is costly in human resources. The reduced support of the SciDAC-3 grant means that not all of our codes are ready to run on all architectures as they become available, delaying production running significantly. A further diversification of hardware will require additional FTEs for our effort to remain competitive.

6.1.4.7 Software Applications and Tools

There is increasing dependence of USQCD codes on software libraries, but it tends to be at the level of BLAS, LAPACK, and ARPACK. Useful services can extend to training workshops or NDA access to new emerging architectural systems.

On-line disk storage and long-term tape storage demands are increasing with more ambitious analysis campaigns. Access to global file systems is a benefit.

With the increasing demand for high I/O throughput, we believe that HDF5 will be the data format of choice. Performance of HDF5 may vary, and we will need assistance in tuning its performance.

6.1.4.8 HPC Services

With the demand for high I/O throughput, we believe that HDF5 will be the data format of choice. Performance of HDF5 may vary, and we will need assistance in tuning its performance.

6.1.4.9 Additional Data-Intensive Needs

There is an increasing need for data transfer. Currently, Globus is regularly used. Some assistance with tuning to optimize data transfers may be helpful. In addition, data transfers directly from the tape systems will become more essential.

We currently have a data management plan, but it becomes impractical to store and transfer large datasets to USQCD facilities. Thus we will need more access to local tape facilities.

6.1.4.10 Additional Data-Intensive Needs: Burst Buffer

This technology might be helpful in moving large intermediate datasets to local storage, but this will require more investigation as to the efficacy of such methods.

6.1.4.11 Requirements Summary Worksheet

NERSC repos m747, mp133, mp7	**Used at NERSC in 2013**	**Needed at NERSC in 2017**
Computational hours (millions)	90	3,000
Typical number of cores* used for production runs	16-64K	100-200K
Maximum number of cores* that can be used for production runs	>1M	> 1M
Data read and written per run	0.1 TB	1 TB

Maximum I/O bandwidth	10 GB/sec	10 GB/sec
Percent of runtime for I/O	10	10
Scratch file system space	100 TB	1,000 TB
Shared file system space	27 TB	1,000 TB
Archival data	632 TB	10,000 TB
Memory per node	64 GB	100 GB
Aggregate memory	50 TB	> 300 TB

* Conventional cores

6.2 QCD Thermodynamics at High Temperature

Principal Investigator: Alexei Bazavov (Brookhaven National Laboratory)
Worksheet Author: Peter Petreczky (Brookhaven National Laboratory)
NERSC Repository: m1416

6.2.1 Project Description

6.2.1.1 Overview and Context

This project aims to improve our understanding of the thermodynamics of QCD, the theory of strong interactions. QCD possesses a low-temperature confined phase, where states like the proton, pion, etc. are observed, and a high-temperature deconfined phase, where quarks and gluons are the fundamental degrees of freedom (quark-gluon plasma, QGP phase). Understanding the properties of QGP is important for understanding fundamental interactions as well as the early evolution of the universe. Experimentally, QGP can be achieved by colliding heavy nuclei at high energies. Such experiments are ongoing in RHIC at BNL and LHC at CERN. The physics of the deconfinement transition is non-perturbative, accessible to lattice QCD techniques, while at very high temperatures conventional weak-coupling techniques are expected to work due to asymptotic freedom.

6.2.1.2 Objectives for 2017

To date we have learned a lot about bulk properties of quark-gluon plasma at zero net baryon density (chemical potential). The goal is to extend these studies to non-zero baryon density using a Taylor expansion approach. In particular, we would like to calculate the equation of state, the transition temperature, and fluctuations of conserved charges at non-zero baryon density. The calculations of the Taylor expansion coefficients require calculations of the product of the inverse of the fermion matrix and a vector multiple times. At very high temperatures, such as T >500 MeV, charm quarks will start to play a significant goal in QCD thermodynamics, in particular in the equation of state. The effect of the charm quarks will need to be included in the calculations of the equation of state.

Another goal is to gain a better understanding of dynamical properties of the quark-gluon plasma, such as meson spectral functions and certain transport coefficients (for example, the heavy quark diffusion constant or electric conductivity). So far, most of such studies have been performed ignoring the effects of dynamical quarks.

6.2.2 Computational Strategies (now and in 2017)

6.2.2.1 Approach

We calculate quantum statistical averages using importance sampling. To do so we use the MILC code. Using molecular dynamics algorithms, the code realizes the Markov process that samples the phase space of 4D field theory in discretized Euclidean space-time (imaginary time formalism). A molecular dynamics (MD) algorithm produces typical (most probable) configurations of fields that give dominant contribution into the QCD path integral. Physical observables are calculated as averages on these configurations. The main ingredients of the code are the calculation of the force, guiding the MD evolution, and inversion of the sparse fermion matrix with a conjugate gradient algorithm. For fractional powers of the fermion matrix (related to staggered fermion discretization scheme), a rational hybrid Monte Carlo

(RHMC) algorithm is used. Combined with a multi-mass inverter, RHMC is very efficient.

6.2.2.2 Codes and Algorithms

The MILC code is written in C with MPI and runs on a variety of platforms such as BlueGene/L, P and Q, and Cray XT5/XE6 with platform-dependent low-level optimization based on USQCD SciDAC libraries. It also has a version that can run on GPU-based systems using CUDA.

6.2.3 HPC Resources Used Today

6.2.3.1 Computational Hours

Six million core hours were used by our project at NERSC in 2013. In addition, 18 million BG/Q core hours and 50 million BG/L core hours were used at BNL; 15 million BG/Q hours were used from centers in Europe; 25 million BG/Q hours were used at ANL (via INCITE); and 40 million hours were used on USQCD clusters.

6.2.3.2 Parallelism

The code has been tested to run reasonably well up to 300,000 cores. For QCD thermodynamics jobs, a much smaller number of cores is used because the lattice sizes are smaller. Strong scaling is more important than weak scaling for our problems.

6.2.3.3 Scratch Data

We estimate needing less than 10 GB for temporary disk space.

6.2.3.4 Shared Data

We don't use a project directory at this time.

6.2.3.5 Archival Data Storage

We had less than 1 TB stored on the NERSC HPSS data archive in 2013.

6.2.4 HPC Requirements in 2017

6.2.4.1 Computational Hours Needed

We estimate needing 100 million core hours at NERSC in 2017.

Our NERSC allocation is a very small fraction of the cycles available for the study of high-temperature QCD. The increase at NERSC over 2013 usage reflects the expected increase in other resources and a healthier fraction of NERSC resources in the total mix.

6.2.4.2 Parallelism

We expect to be typically using about 4,096 cores in 2017.

6.2.4.3 I/O

We would ideally like to have less than 1 percent of our runtime devoted to I/O, although I/O bandwidth is generally not critical for our work. We typically write or read less than 10 GB for the runs and do not use checkpoint/restart.

6.2.4.4 Future Data Needs

In 2017, we expect to need about 0.5 TB of temporary scratch disk space, 10 TB of NERSC project space (globally accessible shared data), and 200 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to the fact that many more gauge configuration files will be generated in the future, and these are typically stored/archived for future use.

6.2.4.5 Memory Required

We estimate needing very little memory, probably less than 1GB per core.

6.2.4.6 Emerging Technologies and Programming Models

The code does use CUDA now to perform the inversion of the fermion matrix on GPUs. We have a code for the calculation of the Taylor expansion coefficients of the QCD pressure that primarily involves calculation of the inverse of the fermion matrix that now runs on Titan. **We believe that GPU- based architectures would be the most cost- effective way to provide the compute cycles needed for lattice QCD**.

The code does not presently use OpenMP directives, but there are plans to implement this. However, due to limited manpower it has not happened yet.

There are efforts to port codes, primarily the fermion matrix inverter, to MIC, but there is no production code. Vectorizing the code for QCD with a vector length of 16 appears to be a challenge.

All software development activities related to GPU and MIC reside with other groups we collaborate with; we do not have a dedicated person in our group working on this.

NERSC could play a role in the transition to these architectures. A NERSC liaison who deals with lattice QCD applications would be useful to us.

From DOE/ASCR, we believe there should be more funding to support code development for GPU and MIC architectures.

6.2.4.7 Requirements Summary Worksheet

(NERSC Repository m1416)	**Used at NERSC in 2013**	**Needed at NERSC in 2017**
Computational Hours (millions)	6	100
Typical number of cores* used for production runs	512-4,096	512-4,096
Maximum number of cores* that can be used for production runs	4,096	4,096
Data read and written per run	<0.01TB	<0.01TB

Percent of runtime for I/O	<1%	%1
Scratch file system space	<0.01TB	<0.01TB
Shared file system space	0 TB	10 TB
Archival data	0.7 TB	200TB
Memory per node	<1GB	<1GB
Aggregate memory	<0.1TB	<1TB

* Traditional cores

7 Nuclear Structure Case Studies

7.1 *ab initio* Nuclear Structure

Principal Investigator: James P. Vary
NERSC Repository: m94 (Split between this and the next case study in 2013)

7.1.1 Project Description

7.1.1.1 Overview and Context

Atomic nuclei are quantum systems that comprise 99.9% of the mass of the visible universe; yet, unlike the situation for atomic electrons, we lack precise knowledge of most of their properties. That is, we do not have a detailed microscopic understanding, based on first principles, of how their constituents—neutrons and protons—are bound together, how the nuclear modes of excitation are formed, or how nuclear reactions take place. This lack of precise knowledge limits our ability to efficiently use atomic nuclei for nuclear energy, both for next-generation fission reactors and for fusion reactors under development. We need *ab initio* (first principles) simulations of nuclear structure based on the underlying theory of the strong interactions to develop this knowledge and provide predictive tools for these applications.

Another driver for *ab initio* simulations of atomic nuclei is the quest to solve fundamental problems that will help uncover new laws of nature. A primary example is the search for the exotic process called neutrinoless double beta decay in nuclei, which violates one of the fundamental conservation laws accepted up to the present time: lepton number conservation. There are vigorous worldwide efforts by many groups searching for this rare process in experiments that require investments of tens of millions of dollars. Once this rare process is detected, the nuclear structure physics for the nuclei participating in the decay process will need to be calculated with high precision to determine exactly the new laws that violate this conservation law. This research area is dubbed "new physics" or "physics beyond the standard model," and it has major implications for our knowledge of the birth of the universe and the asymmetry between matter and antimatter in the universe. A recent National Academies of Sciences report identifies this as one of the leading science questions for the 21st century.

HPC enables simulations of the theory at sufficient precision, with quantified uncertainties, to compare with experimental results and determine new features of the theory, such as the low-energy constants of chiral effective field theory. With these determinations, one achieves a highly precise predictive tool capable of addressing practical problems related to nuclear energy, as well as forefront theoretical questions such as the origin of nuclear binding energy, nuclear collective motion, fundamental electroweak transition rates, and physics beyond the standard model.

7.1.1.2 Objectives for 2017

Our main objective is to achieve *ab initio* no core shell model developments enabling the calculation of nuclear double beta-decay in A=48 nuclei. This includes:

- Theory of the nuclear matrix element, including higher order corrections arising in effective field theory
- Code capability to solve for the structure of these nuclei, including coupling to the continuum, with chiral nucleon-nucleon plus three-nucleon interactions
- Solving for that structure using an eigensolver for large sparse Hermitian Hamiltonian matrices that is load-balanced and scalable to millions of cores
- Ability to work in a hybrid mode with partial recompute-on-the-fly of the many-body Hamiltonian matrix elements to save on storage and achieve larger basis spaces producing eigensolutions close to the converged, infinite matrix, limit
- Extensive production runs to verify and validate the theory and quantify its uncertainties

7.1.2 Computational Strategies (now and in 2017)

7.1.2.1 Approach

The *ab initio* nuclear many-body problem is defined within non-relativistic quantum mechanics to be the challenge of solving the Schroedinger equation for all nucleons of the nucleus interacting simultaneously via state-of-the-art, strong nucleon-nucleon plus three-nucleon interactions. Equivalently, in the matrix formulation of quantum mechanics, one seeks the solution of the many-body Hamiltonian matrix eigenvalue problem. This includes both stationary state solutions and solutions for states above breakup threshold that require coupling to the continuum. Our approach is based on the Lanczos algorithm.

Parallelism in all our codes is expressed using MPI and OpenMP. We have begun employing GPUs.

7.1.2.2 Codes and Algorithms

The primary code we have developed and continue to improve is “Many-Fermion Dynamics – nuclear” or “MFDn”. There are seven major stages of the calculations performed by MFDn:

1. Enumerate the many-body basis space according to user-defined criteria
2. Determine the location of non-zero many-body matrix elements in this basis space and hence the number of non-zeroes that must be evaluated for the full Hamiltonian
3. Read in the nucleon-nucleon plus three-nucleon interaction files that define the Hamiltonian
4. Construct and store (partially or fully) the many-body Hamiltonian matrix
5. Perform the Lanczos iterations until either a fixed number of iterations are achieved or a convergence criterium is met, and perform orthonormalization of the Lanczos basis vectors after each iteration
6. Transform the eigenvectors from the Lanczos basis back to the original basis
7. Use a selected set of the eigenvectors in the original basis to calculate a suite of experimental observables and 1-body density matrices that, optionally, may be stored for reuse later (the option to evaluate and store 2-body density matrices is in the planning stage)

7.1.3 HPC Resources Used Today

7.1.3.1 Computational Hours

We employed approximately 27 million hours at NERSC in 2013 for MFDn production runs aided by the resources available during the Edison acceptance phase. In addition, we employed 55 million hours in 2013 for MFDn runs under the INCITE award for nuclear structure and nuclear reactions (James Vary, PI). The total INCITE award was 155 million cpu hours in 2013, split about evenly between ORNL and ANL leadership class machines. A new INCITE award for 2014-2017 will provide 62 million hours for MFDn runs at ORNL and ANL facilities in 2014. In addition, MFDn is employed by collaborating groups around the world that collectively make MFDn runs that use an estimated 25 million hours/year outside the U.S.

7.1.3.2 Parallelism

We employ between a few percent and 100 percent of the available cores since MFDn is highly scalable and we frequently need results in the largest feasible basis spaces to achieve convergence and minimize uncertainties. For the largest possible basis spaces we need the maximum memory available on the machine.

We are not aware of a limit to the number of cores that MFDn may use since it has run successfully on the largest machines available (Edison, Titan, and Mira) using all of the cores.

Weak scaling is far more important since we need the problem size to become as large as possible to approach convergence with minimum uncertainty. This will enable us to reach larger nuclei (expanded scientific portfolio) and/or larger basis spaces (improved convergence). These are the needs defined by our goals outlined above.

7.1.3.3 Scratch Data

We typically employ up to a few hundred GB for a single run. This storage is required for the final eigenvectors, which we may wish to store for later post-processing to evaluate additional experimental observables. When this occurs, we move the output eigenvectors to HPSS for long-term storage.

7.1.3.4 Shared Data

We employ a project space (/project/projectdirs/m94) primarily for libraries of MFDn interaction input files, MFDn output files enabling post-analysis, and code repositories. We used 2.4 TB of this space in 2013.

7.1.3.5 Archival Data Storage

We employ about 5 TB of HPSS storage at NERSC.

7.1.4 HPC Requirements in 2017

7.1.4.1 Computational Hours Needed

Due to our drive toward heavier nuclei and larger basis spaces, we anticipate our need for NERSC resources to accelerate beyond our historical trend in utilization of hours at NERSC, which has approximately doubled each year. This was demonstrated in 2013 when we exploited "free" time during the Edison acceptance phase and used 27 million hours. To achieve our scientific objectives in 2017, we estimate that we will need 96 million hours.

Note that this estimate exceeds what we currently use in our INCITE award (62 million in 2014). We have been achieving growth of about 35% each year in our INCITE award over the past several years, and we expect that to continue. Thus for 2017 we project INCITE resources for MFDn production runs in the neighborhood of 155 million hours.

7.1.4.2 Parallelism

We expect to be in the multi-million-compute-core domain as soon as such resources become available. We continually work with computer scientists and applied mathematicians within our SciDAC award (NUCLEI) to refine our algorithms to exploit new and emerging technologies.

7.1.4.3 I/O

We have a limited restart capability. Analysis has shown that it is not practical to store the full Hamiltonian for restart as it is less costly in computational resources to regenerate it. However, it is cost-effective to store the intermediate Lanczos vectors during a run that may be used for a restart of the run at the point of the last successful Lanczos iteration.

We work to keep I/O limited, although a single run may output 300-500 GB to scratch in the form of converged eigenvectors that are saved for post processing. This is likely to grow to 600-800 GB in 2017.

We strive to limit I/O to less than 5% of the total runtime.

7.1.4.4 Future Data Needs

In 2017, we expect to need 3 TB of temporary scratch disk space, 8 TB of NERSC project space (globally accessible shared data), and 50 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to storing the converged eigenvectors of runs for larger nuclei and/or larger basis spaces. In addition, the growth in storage is for 1-body and 2-body density matrices produced in large runs.

7.1.4.5 Memory Required

We find it is more efficient to store the many-body Hamiltonian than to regenerate it on the fly, so we would prefer the maximum memory per core that is feasible. When it comes to tradeoffs between memory and communication bandwidth, we favor memory over bandwidth.

7.1.4.6 Emerging Technologies and Programming Models

MFDn runs on Titan at ORNL and employs the GPUs with CUDA directives. Some, but not all, of the most computationally intensive parts of MFDn have been upgraded for GPUs. Additional developments are under way to implement further use of the GPUs by other sections of the code.

MFDn is hybrid MPI/OpenMP. It runs on Mira with threading, Titan with GPUs, and a variety of other architectures.

There is no current effort to port to or optimize for MIC architecture.

With support from SciDAC/NUCLEI, we have an intensive research and development project to continually improve MFDn. We have collaborated with researchers at LBNL (Ng, Yang, Aktulga) and at Old Dominion University (Sosonkina) to produce major gains in efficiency and scalability. These research projects are ongoing and have produced many jointly authored publications.

NERSC can assist in transitioning to new and emerging architectures by providing experts who can devote significant time to the challenges posed by the evolving architectural landscapes. DOE can assist by continuing the successful SciDAC program that brings together multi-disciplinary teams needed to achieve efficient use of new facilities for discovery-level science.

7.1.4.7 Software Applications and Tools

We need BLAS, LAPACK, and SLAPACK for handling dense kernels. We also need PARPACK for some of the eigenvalue calculations.

7.1.4.8 HPC Services

We utilize NERSC consultants typically on a call-in basis, and this has worked very well for us. We anticipate this need to continue indefinitely. We also benefit greatly from the extensive web resources on everything from "getting started" to details on compiler options and information on libraries.

7.1.4.9 Additional Data-Intensive Needs

We do have a data plan for MFDn input libraries as well as MFDn outputs. Both of these data sets are stored in project space and backed up on HPSS.

7.1.4.10 Additional Data-Intensive Needs: Burst Buffer

Our data is probably not in the category of "intensive," although we would certainly benefit from improved I/O capabilities at NERSC.

7.1.4.11 Requirements Summary Worksheet

NERSC Repository m94. (Also see next case study.)	**Used at NERSC in 2013**	**Needed at NERSC in 2017**
Computational hours (millions)	27	96
Typical number of cores* used for production runs	From 5% to full machine	From 5% to full machine
Maximum number of cores* that can be used for production runs	Full machine	Full machine
Data read and written per run	< 1 TB	< 1 TB
Maximum I/O bandwidth	Not known	Not known

Percent of runtime for I/O	< 5%	< 5%
Scratch file system space	1 TB	3 TB
Shared file system space	1 TB	8 TB
Archival data	27.6 TB	250 TB
Memory per node	All available GB	Maximum possible GB
Aggregate memory	Full machine	Full machine

- Conventional cores

7.2 *ab initio* Calculations of Nuclear Reactions and Exotic Nuclei

Prepared by: Sofia Quaglioni, Lawrence Livermore National Laboratory (LLNL)

Contributors: G. Hupin, LLNL (present address: University of Notre Dame); C. Romero-Redondo, LLNL; P. Navrátil, TRIUMF; J. Langhammer, Technishe Universität Darmstadt; Robert Roth, Technishe Universität Darmstadt

Refers to the following HPC allocations:
NERSC Repository: m94 (Principal Investigator: J. P. Vary, Iowa State University)
INCITE: "Nuclear Structure and Nuclear Reactions" (Principal Investigator: J. P. Vary)
LLNL Institutional Computing Grand Challenge: "From Nucleons to Nuclei to Fusion Reactions" (Principal Investigator: S. Quaglioni, LLNL)

7.2.1 Project Description

7.2.1.1 Overview and Context

Our ultimate goal is to develop a fundamental theory and efficient computational tools to describe dynamic processes between nuclei and to use such tools toward supporting several DOE milestones. This includes:

- Performing predictive calculations of difficult to measure landmark reactions for nuclear astrophysics, such as those driving the neutrino signature of our sun
- Improving our understanding of the structure of nuclei near the neutron drip line, which will be the focus of the DOE's Facility for Rare Isotope Beams (FRIB) being constructed at Michigan State University
- Helping to reveal the true nature of the nuclear force.

Furthermore, these theoretical developments will support plasma diagnostic efforts at facilities dedicated to the development of terrestrial fusion energy.

Despite nearly 80 years of study, coming to a fundamental understanding of nuclei and their role in the universe remains a central goal of nuclear physics. The origin of this lack of understanding resides in two main challenges characterizing the nuclear many-body problem: the interaction between nucleons is complex and is not yet fully understood at a fundamental level; and a fully developed first-principles theory able to provide a unified treatment of a large range of nuclear phenomena (bound-state, scattering, and reaction observables) is still missing.

Our research program is devoted to advancing the state of the art in both these aspects. We apply HPC to solve the nuclear many-body problem in terms of constituent protons and neutrons interacting through the most fundamental interactions available nowadays: nucleon-nucleon (NN) and three-nucleon (3N) forces grounded in the fundamental theory of quantum chromodynamics (QCD) within the framework of chiral effective field theory. (The 3N force is a direct consequence of the complexity of the QCD theory.) At the same time, to achieve a robust *ab initio* description of nuclear properties we are developing a unified and computationally efficient approach to bound and scattering states in light nuclei.

7.2.1.2 Objectives for 2017

Until just five years ago, an *ab initio* treatment of light-nucleus fusion reactions was impossible. With our work we have demonstrated that it is possible to describe complex processes such as the ^{7}Be(p,γ)^{8}B radiative capture[6] (important for its influence on the Standard Solar Model) and the ^{3}H(d,n)^{4}He and ^{3}He(d,p)^{4}He fusion reactions[7] with first principle calculations based on realistic NN potentials. In addition, we delivered an evaluation of the (poorly known) n-^{3}H elastic cross section for 14 MeV neutrons,[8] important for plasma diagnostics in fusion experiments, with the required 5% accuracy.

More recently, we have taken initial steps to achieve a complete *ab initio* description of light-nucleus fusion and light exotic nuclei by incorporating the 3N force into our binary-reaction formalism[9] and extending our approach to the description of three-cluster dynamics.[10] These technical developments are highly nontrivial and individually—let alone combined—pose an unprecedented computational challenge. Therefore, initially we have been pursuing them separately. However, our objective for 2017 is to achieve the first *ab initio* description, complete with both 3N-force and three-cluster dynamic effects, of processes important for plasma diagnostic in fusion experiments and for the standard solar model, such as the ^{3}H(^{3}H,2n)^{4}He and ^{3}He(^{3}He,2p)^{4}He fusion reactions; and the spectroscopy of exotic nuclei such as ^{11}Li, a ternary system of two nucleons orbiting around a ^{9}Li core, whose components are not bound in pairs. In addition, with increased access to HPC resources in 2017, we will be able to use our present theory and codes to perform high-fidelity calculations of the scattering of nucleons on a variety of p-shell targets, using NN+3N forces, by accessing much larger model spaces than possible today.

To achieve these goals in 2017, we will need a significantly larger amount of computational resources compared with our present usage. The concomitant storage of sparse matrices of large dimensions associated with the description of three-cluster dynamics and the enormous number of 3N force matrix elements (~6 billion) will dramatically increase the required amount of memory per node. Algorithmic changes will likely be required to further improve the distribution of matrix elements across all available processors and systematically implement load-balancing techniques that have proven very efficient in the most recently implemented parts of our production code.

7.2.2 Computational Strategies (now and in 2017)

7.2.2.1 Approach

Similar to *ab initio* nuclear structure calculations, which aim at describing the wave functions of nucleons bound inside a nucleus, we deal with a strongly correlated non-relativistic quantum many-body problem. We view the nucleons as point-like fermions, all of which interact among each other through high-precision two- and three-nucleon forces. However, in *ab initio* nuclear reactions we work on the even harder problem of describing

[6] P. Navrátil, S. Quaglioni and R. Roth, Phys. Lett. **B704**, 379 (2011).

[7] P. Navrátil and S. Quaglioni, Phys. Rev. Lett. **108**, 042503 (2012).

[8] J. A. Frenje *et al.,* Phys. Rev. Lett. **107**, 122502 (2011); P. Navrátil *et al.*, LLNL-TR-423504 (2010); J. D. Anderson *et al.*, LLNL-TR-435981 (2010).

[9] G. Hupin, J. Langhammer, P. Navrátil, S. Quaglioni, A. Calci, And R. Roth, Phys. Rev. C **88**, 054622 (2013).

[10] S. Quaglioni, C. Romero-Redondo, P. Navrátil, Phys. Rev. C **88**, 034320 (2013); C. Romero-Redondo, S. Quaglioni, P. Navrátil, and G. Hupin, Phys. Rev. Lett. 113, 032503 (2014).

how nuclei—or clusters, as we like to call them—themselves interact with each other. To describe two-cluster (or three-cluster) dynamics, we use basis states made of pairs (or triplets) of nuclei in relative motion with respect to each other, in which each cluster of nucleons is described by its *ab initio* many-body wave function.

Mathematically our approach casts the many-body problem into one-dimensional (or two-dimensional, in the case of three-cluster dynamics) integral-differential coupled-channel equations for the cluster's relative motion. The main computational challenge is to construct the Hamiltonian and overlap (or norm) non-local couplings from the input two- and three-nucleon interactions and the many-body wave functions of the target and projectile nuclei. (The non-locality and the algebraic complexity of these couplings are a consequence of the Pauli exclusion principle, which is treated exactly in the formalism.) We use the R-matrix method on a Lagrange mesh to find the elements for the scattering matrix. At the end of the run we use the scattering matrix elements to calculate scattering and reaction observables to compare with available experimental data or to make needed predictions.

The Hamiltonian and the norm couplings can be obtained in second quantization in terms of matrix elements of creation and annihilation operators acting on the target nucleus. The number of nucleons affected (i.e., the rank of the operator involved) increases with the number of nucleons forming the projectile and going from two- to three-nucleon interactions. For example, while the description of a nucleon scattering on a nucleus with a two-nucleon interaction requires the evaluation of matrix elements of one- and two-body densities of the target, calculations of deuterium-nucleus scattering including a three-nucleon force involve up to four-body densities. Storing in memory many-body density matrices is computationally very demanding. We adopt two computational strategies, which we have used to benchmark our results. One, more recent, strategy takes advantage of the target's implicit expansion in Slater determinants of single-nucleon states to perform an efficient on the fly calculation of the density matrix elements. In the second, first implemented, strategy we first calculate and store in memory the reduced matrix elements of tensor operators, from which we can then reconstruct the desired density on the fly. With the exception of the treatment of isospin symmetry, the two implementations are formally equivalent. The latter method is feasible only for very light systems but is more efficient for reactions with different projectiles, while the former is ideally suited for addressing heavier targets.

7.2.2.2 Codes and Algorithms

The calculations described above are performed using the NCSM_RGM (no-core shell model – resonating group method) reaction code, with inputs from the auxiliary TRDENS (transition densities) preprocessing code. Both are HPC codes mostly dealing with sparse linear algebra but also including components featuring dense linear algebra. NCSM_RGM uses a hybrid MPI/OpenMP parallelization to take full advantage of all computing cores, plus MPI/IO to mitigate I/O costs associated with large input files. In the newest routines, we make use of dynamic load balancing techniques. The largest runs utilize 12,288 MPI tasks with eight OpenMP threads per MPI task (total size = 98,304 cores, 6,144 new Jaguar XK6 nodes). TRDENS is based on the MPI-2 protocol and was run with up to 900 cores on Sierra, a Dell supercomputing system with peak speed of 261 Tflops/s, 12 processors/node, and 24 GB memory per node.

Finally, for reactions involving heavier targets, we are shifting large portions of the calculations to an auxiliary sister code, which we will call here NCSM_RGM_SD, that precomputes the Hamiltonian and norm couplings within a Slater determinant single particle basis. NCSM_RGM_SD is also a hybrid code, adopting MPI parallelization between the individual compute nodes and OpenMP parallelization across the cores of each node. The largest runs have been performed on the NERSC Edison machine using 600 nodes, for a total of 14,400 cores.

7.2.3 HPC Resources Used Today

7.2.3.1 Computational Hours

The LLNL Institutional Grand Challenge project "From Nucleons to Nuclei to Fusion Reactions," led by S. Quaglioni, has a significant allocation on Sierra, a LLNL capability computer for moderate to large parallel jobs. Sierra is a Dell supercomputing system with peak speed of 261 Tflops/s, 12 processors/node, and 24 GB memory per node, for a total of 1,944 nodes. We also have access to CAB, an LLNL large capacity machine for small to moderate parallel jobs. CAB is a Linux cluster with 16 2.6-GHz processors/node and 32 GB memory per node for a total of 1,296 nodes. Our combined utilization in 2013 was over 10 million core hours.

About 23 million hours were also used in 2013 for reactions calculations on the ORNL Cray XK7 Titan machine as part of the INCITE project "Nuclear Structure and Nuclear Reactions" led by J. Vary.ß

Finally, in 2013 we made use of about 10 million CPU hours on the Edison machine at NERSC (20 million NERSC MPP hours) to run the NCSM_RGM_SD code and calculate nucleon-nucleus scattering on p-shell nuclei, including the 3N force. This work was performed under NERSC project m94, "*ab initio* Nuclear Structure," also led by J. Vary.

7.2.3.2 Parallelism

As of today, the NCSM_RGM code has been run with as many as 98,304 cores (composed of 6,144 nodes, with 12,288 MPI tasks and eight OpenMP threads per MPI task).

The TRDENS code was run with up to 900 cores on Sierra.

The NCSM_RGM_SD code has been run on the Edison machine using between 200 and 600 nodes, for a total of 4,800 to 14,400 cores. Runs using twice the number of cores are likely feasible, given the good scaling performance presented by the code so far.

Utilization for typical NCSM_RGM, TRDENS, and NCSM_RGM_SD runs varies depending on the system under study (projectile/target mass, two- or three-nucleon Hamiltonian, or size of the model space). Due to limits on the size of runs per user, typical NCSM_RGM runs on LLNL machines are performed with 12,000 to 19,200 cores and are restarted multiple times.

Our project requires strong scaling. The size of our problems is such that calculations would simply not be possible without parallel computing. We rely on scaling to complete our calculations, or major stages of it, within the allowed runtime.

7.2.3.3 Scratch Data

The required amount of temporary space varies somewhat with the problem under investigation. We require on the order of 1 TB per major run.

7.2.3.4 Shared Data

We have used the project folder within project m94 for the common use of certain matrix-element files for collaborations.

7.2.3.5 Archival Data Storage

We have not used the data archive.

7.2.4 HPC Requirements in 2017

7.2.4.1 Computational Hours Needed

To reach the scientific goals listed in section 8.2.1.2, as well as other scientific achievements that will be made possible by the sustained development of our *ab initio* reaction approach and codes, we forecast that in 2017 our project will require on the order of 75 million hours on a machine such as Edison (150 NERSC MPP hours).

We have not secured any significant allocation for 2017, although we expect that we will compete for HPC allocations on LLNL and ORNL systems.

The primary factor driving the need for more core hours is the increased challenge of more complex reaction calculations, including the 3N force on larger target and projectile nuclei.

7.2.4.2 Parallelism

In 2017 we expect that the NCSM_RGM code will typically use on the order of 28,400 to 60,000 conventional cores per run, with larger runs of up to 100,000 cores.

With further improvements on the MPI scaling, we expect that the NCSM_RGM code could be run with using the full Edison machine.

7.2.4.3 I/O

Our application has several built-in restart procedures that allow us to perform fairly large calculations even on computers with a modest number of nodes, such as Sierra.

Our data requirement will grow in 2017, particularly in output. We expect that input should not exceed 100 GB, while we estimate that we will reach on the order of half a TB of output per run in our calculations of three-cluster dynamics. These outputs, including Hamiltonian and norm couplings, scattering phase shifts, scattering wave functions, and elements of the scattering matrix at various energy steps, can be moved to long-term storage and utilized for later calculations of reaction observables.

We use MPI I/O to mitigate costs associated with large input files. We would like to keep the time devoted to I/O to less than 5% of the total runtime.

7.2.4.4 Future Data Needs

In 2017, we expect to need 4 TB of temporary scratch disk space, 8 TB of NERSC project space (globally accessible shared data), and 30 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to the increased scale of the scientific problem under investigation. In particular, we will have to store multiple instances of input three-nucleon force matrix elements, converged eigenvectors for the target, projectile and compound nuclei, and transition densities, as well as converged scattering phase shifts, scattering wave functions, and elements of the scattering matrix at various energy steps.

7.2.4.5 Memory Required

We will need a minimum of 64 GB of memory per node but would benefit from the maximum feasible memory per node, which would allow us to increase the model-space size and hence the fidelity of our calculations. We can use all available memory per node.

7.2.4.6 Emerging Technologies and Programming Models

For the time being we have not explored the use of GPUs, mainly because for our computational problem it is more straightforward to use OpenMP capabilities, and in part due to the lack of manpower and readily available knowhow to explore these new technologies. We envision that GPUs may be advantageous during the solution of the two- and three-body dynamical equations, where the calculation of the scattering matrix requires dense linear algebra (matrix multiplications and inversions, eigenvalue problems) at each energy step.

Our software currently runs in production on Titan, but for the time being we do not take advantage of the GPUs. Rather, our NCSM_RGM software uses a hybrid MPI/OpenMP parallelization to take full advantage of all computing cores and memory in the nodes.

Currently we are not porting to, nor optimizing our codes for, MIC architecture. Our strategy for exploiting GPUs and other new technologies will be to seek the help of experts in computational science, perhaps through the SciDAC NUCLEI collaboration, which has helped improve the *ab initio* nuclear structure code MFDn.

In transitioning to new and emerging architectures, the assistance of NERSC experts on the new architectures will be fundamental. Ideally, large allocations on NERSC machines should be accompanied by dedicated support from an expert on the computer architecture to help nuclear scientists port their codes and use the machine in the optimal way, maximizing the scientific impact of their calculations. This type of assistance should be accompanied by DOE programs such as SciDAC that allow the close collaboration of nuclear physicists with computational scientists to solve compelling nuclear science problems. In addition, it would be desirable to create positions at the intersection between computational science and nuclear physics that could foster the growth of a new generation of nuclear computational scientists.

7.2.4.7 Software Applications and Tools

We need to link our codes to the multithreaded Intel Math Kernel Library (MKL). Our compiler of preference is the Intel Fortran compiler. We make use of Lustre file system software to stripe large input files onto multiple hard drives and speed up MPI I/O. At the moment we have implemented our own binary file formatting through MPI I/O, but we may transition to HDF5 in the next few years.

7.2.4.8 HPC Services

We would benefit from consulting and account support, data analytics, training, and collaboration tools.

7.2.4.9 Additional Data-Intensive Needs

We would need to be able to transfer data among collaborators in the USA, Canada, and Europe.

Currently we do not have a data management plan.

7.2.4.10 Additional Data-Intensive Needs: Burst Buffer

Technology aimed at improving I/O would be useful for our work, although our data needs may be not as intensive as in other projects.

7.2.4.11 Requirements Summary Worksheet

NERSC Repository m94 (See also previous case study)	**Used at NERSC in 2013**	**Needed at NERSC in 2017**
Computational hours (millions)	20	150
Typical number of cores* used for production runs	4,800 to 14,400 (3% to 10% of machine)	28,800 to 60,000 (20% to 45% of machine)
Maximum number of cores* that can be used for production runs	28,800 (20% of machine)	Full machine
Data read and written per run	<1TB	< 1TB
Maximum I/O bandwidth	Not known	Not known
Percent of runtime for I/O	<5%	<5%
Scratch file system space	1 TB	4 TB
Shared file system space	1 TB	8 TB
Archival data	Not used	30 TB
Memory per node	All available memory	All available memory
Aggregate memory	<38 TB	Typical: 160 TB

*Conventional cores

8 Nuclear Astrophysics Case Studies

8.1 Convection in X-Ray Bursts

Principal Investigator: Michael Zingale (SUNY Stony Brook)
NERSC Repositories: m1938, m106

8.1.1 Project Description

8.1.1.1 Overview and Context

X-ray bursts (XRBs) are the thermonuclear runaway of a thin layer of hydrogen and/or helium on the surface of a neutron star. This fuel layer accretes from a binary companion star, and the immense gravitational acceleration on the surface of the neutron star compresses it, increasing the temperature and density to the point of explosion. One-dimensional hydrodynamic studies have been able to reproduce many of the observable features of XRBs, such as burst energies (~10^{39} erg), rise times (seconds), durations (10s – 100s of seconds), and recurrence times (hours to days) (see Strohmayer & Bildsten 2006 for an overview of XRBs[11]). By construction, however, one-dimensional models assume that the fuel is burned uniformly over the surface of the star, which is unlikely if the accretion is not spherically symmetric (Shara 1982[12]). Furthermore, the Rossi X-ray Timing Explorer satellite has observed coherent oscillations in the light curves of >20 outbursts from LMXB systems (first by Strohmayer et al. 1996[13]; more recently by Altamirano et al. 2010[14] and references therein). The asymptotic evolution of the frequency of such oscillations suggests that they are modulated by the neutron star spin frequency (Muno et al. 2002[15]) and are therefore indicative of a spreading burning front being brought in and out of view by stellar rotation.

Before the actual outburst, the burning at the base of the ignition column will drive convection throughout the overlying layers and set the state of the material in which the burning front will propagate. One-dimensional simulations of XRBs usually attempt to parameterize the convective overturn and mixing using astrophysical mixing-length theory or through various diffusive processes (e.g., Heger et al. 2000[16]). A proper treatment of the convection in these extreme conditions, free from parameterizations, requires multi-dimensional simulations. One of the major open questions is how does the burning begin in a localized fashion. We know that rotation is important (Spitkovsky et al. 2002[17]), but the convective transport of heat is also important. Detailed models of the convective burning, resolving the burning layer, are our initial targets.

[11] Strohmayer & Bildsten 2006, in Compact Stellar X-Ray Sources, Cambridge Univ. Press, 113

[12] Shara, 1982, ApJ, 261, 649

[13] Strohmayer et al. 1996, ApJL, 469, L9

[14] Altamirano et al. 2010, MNRAS, 409, 3, 1136

[15] Muno et al. 2002, ApJ, 580, 1048

[16] Heger et al. 2000, ApJ, 528, 368

[17] Spitkovsky et al. 2002, ApJ, 566, 1018

Our simulation code, Maestro (Nonaka et al. 2010[18]), is designed to efficiently model subsonic convective flows by filtering sound waves out of the equations of hydrodynamics, but maintaining changes in compressibility due to stratification and local heat release. With this tool, we completed the most detailed study of convection in pure helium bursts to date (Malone et al. 2011[19]), and with NERSC resources this past year (2013) we finished a two-dimensional study of mixed H/He bursts (Malone et al. 2014[20]; see Figure 1) using a 10-isotope network that captures hot-CNO, 3-alpha, and rp-process reactions. Both of these studies were two-dimensional, and we found converged dynamics when run with a resolution of 3 or 6 cm—although this extremely fine resolution limits the ability to model these events over a significant fraction of the star (owing to memory requirements). Our current studies can follow the rise in temperature of the H layer to above 10^9 K, evolving for timescales of 0.1 s and bringing us close to the point where rp-process breakout reactions will occur.

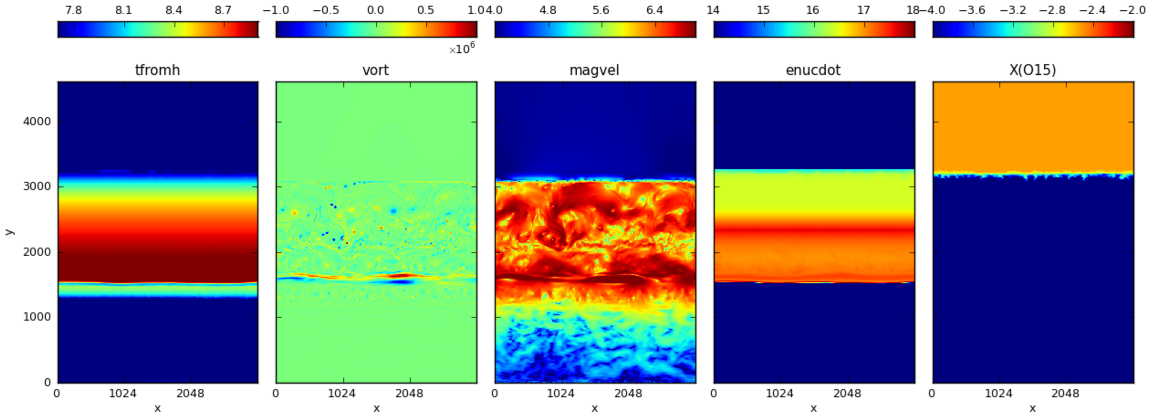


Figure 1. Convective flow in a two-dimensional mixed H/He X-ray burst simulated with Maestro at NERSC.

[18] Nonaka et al. 2010, ApJS, 188, 358

[19] Malone et al. 2011, ApJ, 728, 118

[20] Malone et al. 2014, submitted to ApJ

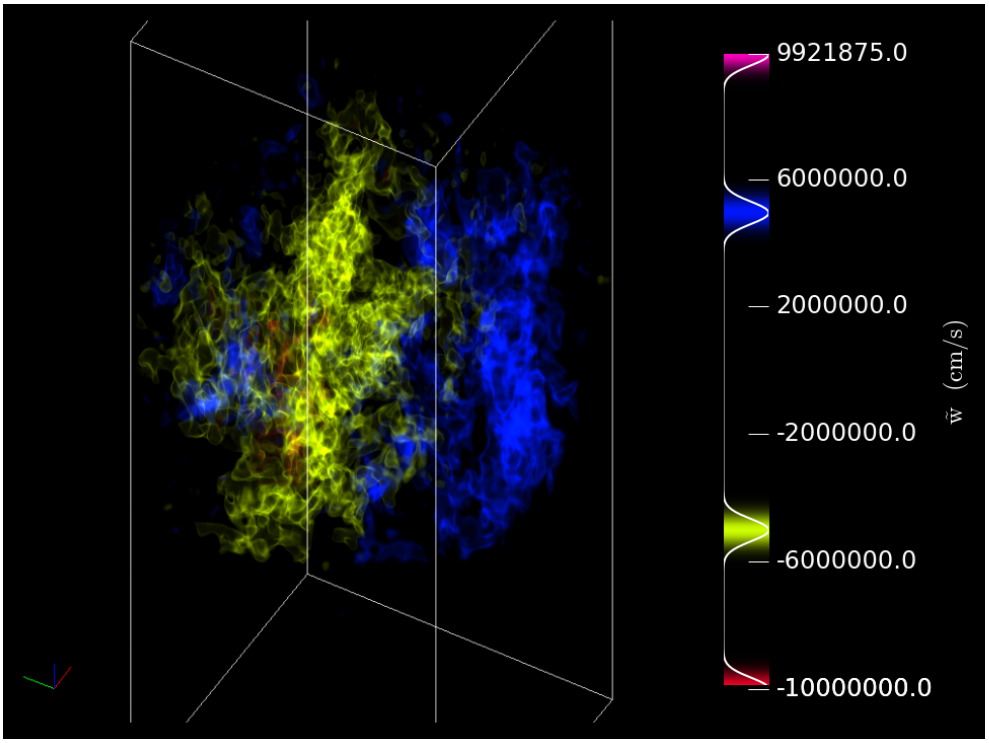


Figure 2.

Most recently, we switched to three-dimensional studies (see Figure 2). At the moment, these are low-resolution, small-domain proof-of-concept runs, but we expect to use the remainder of our NERSC allocation exclusively for larger three-dimensional studies.

8.1.1.2 Scientific Objectives for 2017

The next major advancements for our XRB models are better/larger reaction networks, larger three-dimensional domains, and subgrid-scale models for the burning. These will require both algorithmic and computational developments. We believe that we can meet these demands in the 2017 time frame.

Large reaction networks are computationally expensive, both because we need to carry the additional species around and advect them with the flow and because the integration of the network itself, done implicitly, requires the solution to larger linear algebra problems. To allow us to model larger networks, our plan is to move the reaction part of our codes to the accelerators (GPUs or Intel MICs) found on new supercomputer platforms. To maximize the investiment of the code changes, we insist on using open standards, and therefore will explore using OpenACC to start. We have a plan to begin this porting this year (2014).

Convection and turbulence are three-dimensional phenomena. Running in a small domain artificially confines the convection and changes its dynamics, so we need to run as large of a region as possible. At present we are modeling just a 15 m x 15 m area on the surface of the neutron star. We will need to push toward kilometer scales to see the effects of rotation on the convection, but there is no way we can do this while maintaining a 6 cm resolution. Our goal is therefore to attack this on two fronts. First, we will expand the domain as large as possible while maintaining our resolution requirements to see how the convective behavior changes with an increased domain size. At the same time, we will work on developing a subgrid model that allows us to describe the unresolved burning if we simulate with a coarser grid. These models are common in simulations of Type Ia supernovae, but the scales involved in X-ray bursts are different (in particular, the relative size of the burning zone thickness to a pressure scale height), so new methods will need to be devised. We have begun some initial attempts and will continue to do so as we approach the 2017 time frame.

Finally, if the flow approaches sonic, we can transition the problem from Maestro into our sister code, Castro, which solves the fully compressible equations of hydrodynamics using a similar discretization strategy and microphysics. We will likely explore this transition for some of our simulations in the time frame leading up to 2017.

8.1.2 Computational Strategies (now and in 2017)

8.1.2.1 Approach

We solve the equations of hydrodynamics under the assumption of a low Mach number. This is expressed by requiring that the pressure everywhere in our domain remain close to the background hydrostatic pressure of the star we are modeling. Mathematically, this constraint manifests itself as an elliptic constraint on the velocity field; physically, the effect is to filter sound waves from the system. Together with a hyperbolic system of PDEs describing conservation of mass, momentum, and energy, we can efficiently model convective flows in a stratified background. We use a finite-volume discretization to solve the equations, advancing the state in each time step using a fraction step method consisting of reactions, advection, and projecting the velocity field to satisfy the constraint. A key benefit of our method is that by filtering sound waves from the system, we can take much longer time steps that correspond with fully compressible hydrodynamics code (we are limited by the time it take the fluid to cross a zone instead of the more restrictive time it takes a sound wave to cross a zone).

Our simulation code, Maestro, uses the BoxLib library (developed at LBNL) to provide an adaptively refining grid. Maestro can operate with two geometries: a full, spherical star, and a plane-parallel region of an atmosphere. The latter is appropriate for XRBs; we model a small patch of the surface and include reactions that describe hydrogen burning through the hot-CNO cycle and helium burning and proton captures to some heavier nuclei. This energy release heats the accreted layer on the neutron star and drives convection. In the Maestro algorithm, we describe the background hydrostatic state of the atmosphere as a one-dimensional base state that can evolve due to the large-scale heat release and carry the two- or three-dimensional departures from this hydrostatic state of a Cartesian grid. An equation of state appropriate to stellar interiors is used to complete the system. For the XRB, the gravitational acceleration is assumed to be constant.

8.1.2.2 Codes and Algorithms

Maestro uses a Godunov-type second-order finite-volume method to predict the fluid state on the interfaces through the zones. The velocity on these interfaces is required to satisfy the elliptic constraint in our system, and this is enforced by solving a variable-coefficient Poisson problem using multigrid. The fluxes through the interfaces can then be computed to update the fluid state in the zones to the new time. Again, the new velocity must satisfy the velocity constraint, so another multigrid solve of a variable-coefficient Poisson problem is done. These two solves differ in the centering of the terms; in the first, the quantity being solved for is cell-centered, while in the latter it is node centered—BoxLib provides the two multigrid solvers needed for this problem. Reactions are coupled in through Strang splitting using the VODE ODE integration library to advance the reaction system.

Importantly, Maestro is publicly available (http://bender.astro.sunysb.edu/Maestro/), allowing for reproducibility within the community.

8.1.3 HPC Resources Used Today

8.1.3.1 Computational Hours

We used 4.4 million MPP hours this year for XRB studies at NERSC. While we have time at Titan and Blue Waters, none of that time is for the XRB studies.

8.1.3.2 Parallelism

For our initial, low-resolution three-dimensional XRB runs, we ran with MPI only at 192 cores (a bug in the Cray compiler prevented us from doing OpenMP, but gfortran is fine at NERSC). Assuming that the OpenMP bugs are fixed (they were at OLCF, but we haven't tried the latest compilers at NERSC yet), the low resolution run could be done at 1,536 processors. Our next jump will be twice the resolution (eight times the number of zones), which would put us on 12,288 cores. We could probably go a few times larger than that and still scale reasonably well. Weak scaling of our code showed good results to O(10^5) cores, although that was for a different problem.

We also take the time spent in the queue into account when planning the job parameters. Time to completing the science is our most important metric. Many factors can influence this, including the problem size and the queue sizes. We generally pick a problem size that we believe will give converged results. We then typically pick the number of processors somewhere between the smallest number of processors that can fit that problem and the maximum we can strongly scale to and run at that intermediate number. The rationale is to minimize the time spent in the queue.

An additional consideration not captured by scaling metrics is that science often requires a parameter study, so multiple medium-sized runs are more interesting (and scientifically relevant) than a single "hero calculation."

8.1.3.3 Scratch Data

Single files are 10–100 GB, and over the course of a queue window we output dozens, so a minimum of 1 TB of scratch is needed. We do our best to migrate this data to HPSS as soon as possible, but often we will want multiple files around to do some exploratory analysis while the simulation is still running.

8.1.3.4 Shared Data

Project directories (under the NERSC repo m1400) are invaluable. Science is done by collaboration, so often we need other members of our group to be able to explore the data as the run is progressing to help with the analysis and to help plan the next simulations.

8.1.3.5 Archival Data Storage

Repo m106 had 173 TB stored in HPSS at NERSC in 2013. Occasionally we delete some old runs and test runs, but we want to keep the majority of the scientific output from simulations that have been published for as long as possible (some agencies are asking for data management plans, and without HPSS we could not store the data).

8.1.4 HPC Requirements in 2017

8.1.4.1 Computational Hours Needed

A big three-dimensional run today with a bigger reaction network and larger surface region modeled will take over 1 million core hours. That is the base point for our 2017 plans. We need to do multiple runs to assess the robustness of the results. In the 2017 time frame, we can imagine trying to go bigger still, putting individual runs at 10–20 M core hours, and again wanting to do 10s of runs to understand the sensitivity of the results.

We typically work on several different science applications with the same codes and seek resources for these different projects through INCITE, Blue Waters, and XSEDE. We will continue to do so, but we expect the XRB work to remain our target for NERSC.

We want to run bigger problems. We are modeling only a small fraction of the star currently, and to explore the effects of rotation and understand the distribution of burning regions we need to model a much bigger region of the star.

8.1.4.2 Parallelism

We should comfortably run on 10,000s cores, but again the queue structure and the desire to run many jobs in a parameter study will determine the optimal size.

For a true science run, if we have a lot of local physics (for example, a big reaction network), we could use 100,000 cores.

8.1.4.3 I/O

We use checkpoint/restart in our application. For large 3-D simulations, a single checkpoint file can be hundreds of GB. Our total output for a big 3-D simulation is ~10 TB. We expect these numbers to scale with the problem size, so files 10x the current size are to be expected, giving 100 TB for a typical big simulation.

For a simulation on Jaguar at OLCF (the precursor to Titan) done in July 2012, we got a data rate of 1.9 GB/s writing a 5.2 GB plotfile out from 1,024 processors (this was a moderate-sized job). For a more recent job on Titan (done in the past few months), we got a data rate of 8.5 GB/s writing an 85 GB plotfile out from 2,048 processors (this run had four levels in the adaptive mesh grid hierarchy). These numbers are typical and result from real science runs on the crowded machine.

Less than 5% of runtime is acceptable for I/O. If I/O becomes too slow, we will work on doing *in situ* analysis instead.

8.1.4.4 Future Data Needs

In 2017, we expect to need 10 TB of temporary scratch disk space, 100 TB of NERSC project space (globally accessible shared data), and 500 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to growth in the size of the problems we wish to run.

8.1.4.5 Memory Required

The "sweet spot" for our code at the moment is 2 GB/core. By doing OpenMP on nodes, some of the necessary metadata for the grid hierarchy is shared across the cores on the nodes, allowing for a more efficient use of the memory. Aggregate memory can grow toward 1 TB or more depending on how big of a job we seek to run.

8.1.4.6 Emerging Technologies and Programming Models

Our goal is to use OpenACC, as it appears to be the more portable standard for accessing GPUs. At the moment we do not use this, but we have plans to put some of our microphysics (starting with the reaction networks) on the GPUs this year.

While we run on Titan frequently, we do not use the GPUs there (yet).

Our code makes extensive use of OpenMP for communication within a node. We split the outermost loop across zones in a grid patch across the OpenMP threads when there is enough work in the loop. This is a significant performance win. More tuning of the OpenMP for the XRB simulations proposed here will take place this year.

We do not currently run on either Sequoia or Mira.

We have not started porting to MIC, but some of our collaborators at LBNL have reported good success at porting BoxLib codes to the Intel MIC. We will rely on their experiences and expertise if it becomes necessary for Maestro. A caveat is that utilizing the MICs may require us to change our gridding strategy to allow for more uniformly sized grids. For the XRB simulations, this is not likely a problem since we typically fully refine a rectangular region that encompasses the convection, so we have the flexibility to grid uniformly there.

We collaborate closely with the CCSE group at LBNL that develops the BoxLIb library for AMR, which is the core library we use in Maestro. We inherit any performance gains they implement and work closely with them on the code development and science. They are separately funded.

The dream, of course, is that one sends their code off and it gets sent back a short while later completely optimized for the new machine. We know, of course, that this is impossible, and as scientists we expect the code to remain functional during any changes and that testing show that the code still gives the correct answer when ported. While there are some tools to help with the migration process—in particular, identifying bottlenecks—it would be useful for these to be made more well known (with online training provided) and, ideally, for NERSC consultants to assist in doing the initial profiling of a code on the new architecture and consult with the developers on some ideas and easy targets for optimization.

Having computational scientists (preferably postdocs) directly funded as part of the science project is the optimal way to address new architectures. Supplemental funds to projects that are heavily invested in computational nuclear (astro)physics to help porting to these codes would work best. Given that our simulation codes are publicly available, the community will benefit from any optimizations. Having a dedicated NERSC science liaison can also be useful, but this person would really need to be able to spend the time coming up to speed on the science needs and the intricacies of the code for the effort to be worth it. In effect, they would be embedded in the science group.

The main concern is that whatever technology is adopted, it should be accessible through open standard programming models (i.e., OpenACC instead of CUDA). We don't want to be in a situation where we need to customize our code, rewriting large swaths, for each machine out there.

8.1.4.7 Software Applications and Tools

We are moving toward doing most of our visualization and analysis with the yt package (http://yt-project.org/), which is written in python (and Cython) and provides a very powerful way to script analysis and visualization. Having yt installed as a module on the NERSC machines would be optimal.

Maestro is written in Fortran (95+) using MPI and OpenMP. Some C is used in the BoxLib library and some python is used in our build system. Mainly, we need a robust Fortran 200x compiler with support for OpenMP and (soon) OpenACC.

8.1.4.8 HPC Services

Training of graduate students and postdocs is a plus—ideally through a regular summer school that teaches the students some of the more popular algorithmic techniques (maybe domain-specific parallel sessions?) along with MPI/OpenMP/OpenACC.

8.1.4.9 Additional Data-Intensive Needs

At the moment, we are comfortable with our data handling.

We cannot bring all the data we generate back to our local facilities (the network speeds are too slow), so we store all the needed data on HPSS at NERSC. Our submission scripts are instrumented to automatically store the necessary files as the job runs, virtually eliminating the possibility of data loss.

All of our code is in version control, allowing us to go back, if necessary, and recreate old simulations. We also store a host of meta-data (the machine we are running on, build directories, output directories, compiler versions and flags, git hashes for all of the source repos we used, values of every runtime parameter, etc.) in all of our output files. This greatly helps with reproducibility; all of the information needed to reconstruct the simulation code and inputs is part of the output itself.

8.1.4.10 Requirements Summary Worksheet

NERSC Repos m1938, m106	**Used at NERSC in 2013**	**Needed at NERSC in 2017**

Computational hours (millions)	4.4	100
Typical number of cores* used for production runs	1,536 for small jobs; 12,288 for large	10,000s
Maximum number of cores* that can be used for production runs	10,000s	100,000
Data read and written per run	10 – 100 TB	100s TB
Maximum I/O bandwidth	8 GB/sec	10s GB/sec
Percent of runtime for I/O	1%	< 5%
Scratch file system space	1 TB	10 TB
Shared file system space	0.82 TB	25 TB
Archival data	173 TB	500 TB
Memory per node	2 GB	2 GB
Aggregate memory	0.1s TB	few TB

*Conventional cores

8.2 Core Collapse Supernovae

Principal Investigator: W. Raphael Hix (ORNL)
Case Study Authors: Eric Lentz (our INCITE PI) and Bronson Messer
NERSC Repository: m1373

8.2.1 Project Description

8.2.1.1 Overview and Context

The deaths of massive stars (M > 8-10 solar masses) as core-collapse supernovae are an important link in our chain of origins from the Big Bang to the present. They are the dominant source of elements in the periodic table between oxygen and iron and potentially are responsible for producing half the elements heavier than iron. Core-collapse supernovae serve both to disperse elements synthesized in massive stars during their lifetimes and to synthesize and disperse new elements themselves. As the name suggests, core-collapse supernovae are initiated by the collapse of the cores of massive stars at the end of their lives. The center of a massive star as it nears its demise is composed of iron, nickel, and similar elements, the end products of stellar nucleosynthesis. Above this iron core lie concentric layers of successively lighter elements, recapitulating the sequence of nuclear burning that occurred in the core. Unlike prior burning stages, where the ash of one stage became the fuel for its successor, no additional nuclear energy can be released by further fusion in the iron core. No longer can nuclear energy production stave off the inexorable attraction of gravity. When the iron core grows too massive to be supported by electron degeneracy pressure, the core collapses. The collapse proceeds to ultrahigh densities, in excess of the densities of nucleons in the nucleus of an atom ("super-nuclear" densities). The inner core becomes incompressible under these extremes, bounces, and, acting like a piston, launches a shock wave into the outer stellar core. This shock wave will ultimately propagate out of the iron core and through the stellar layers beyond the core (silicon, oxygen, etc.) and disrupt the star in a supernova explosion. However, the shock stalls in the outer core, losing energy as it plows through. Exactly how the shock is revived is uncertain and remains the central question pursued by our simulations.

After core bounce, 10^{46} Joules of energy in the form of neutrinos and antineutrinos of all three flavors (electron, muon, and tau) are released from the newly formed proto-neutron star (PNS) at the center of the explosion, over a period of tens of seconds. The kinetic energy observed in supernova explosions is 10^{44} Joules, 100 times smaller than the available energy in neutrinos. Past simulations have demonstrated that energy in the form of neutrinos emerging from the PNS can be deposited behind the shock and potentially revive it. While a prodigious amount of neutrino energy emerges from the PNS, the neutrinos are weakly coupled to the material directly below the shock. In fact, because the neutrino heating is very sensitive to the distribution of neutrinos in energy and direction of propagation, realism in modeling core-collapse supernovae requires at least spectral neutrino transport. Convection directly beneath the shock fundamentally alters the nature of neutrino shock reheating, allowing simultaneous downflows that fuel the neutrino luminosities and upflows that bring energy to the shock. And the instability of the shock wave itself—the stationary accretion shock instability (SASI)—can dramatically alter the shock and explosion dynamics. The combination of spectral neutrino radiation transport with multi-dimensional hydrodynamics and complex microscopic physics renders simulations of core-collapse supernovae that require the use of HPC resources to attack this problem.

8.2.1.2 Objectives for 2017

Our objectives for the coming years are a series of 2D and 3D simulations toward two goals: to explore the neutrino-driven core-collapse supernova mechanism, and to predict observable consequences of these explosions that may be compared to observations. These simulations will be pursued across a range of computing platforms through programs such as ERCAP, INCITE, PRAC, etc. Ultimately, our understanding of the core-collapse supernova mechanism and our ability to use core-collapse supernova simulations to directly confront observations will require an ensemble of numerical experiments to address the variety of stars that undergo core-collapse (varying in stellar mass, rotation rates, metallicity, etc.) and to examine uncertainties in our understanding of the core-collapse mechanism and the progenitor stars.

Extreme computational cost—100 million of core-hours for a single simulation—precludes pursuing these investigations solely in three dimensions. Instead, a range of 2D simulations run at NERSC and similar centers will be checked against a smaller set of 3D simulations run using resources provided by programs such as INCITE.

8.2.2 Computational Strategies (now and in 2017)

8.2.2.1 Approach

The core-collapse supernova problem is a coupled multi-physics problem that requires sophisticated multi-physics codes linking the important ingredients with minimal approximation and close attention to the conservation of energy. There is also a wide variety in initial conditions (from stars of different masses, rotations, etc.). The contributors to a supernova and supernova simulation include:

- Self-gravity with some general relativistic elements
- Multidimensional fluid dynamics
- A nuclear equation of state (pressure as a function of temperature
- Mass density and composition) to close the hydrodynamic equations
- A dynamic thermonuclear reaction network for low density/temperature regions that are not in thermodynamic equilibrium
- A necessary set of neutrino-matter interactions
- A neutrino transport algorithm to transport neutrinos and deposit their energy in the matter accurately.

A core-collapse supernova lacks any inherent symmetry, so full understanding requires well-resolved 3D simulations.

8.2.2.2 Codes and Algorithms

Our code, CHIMERA, is a fusion of several codes that handle the required physics. We use a spherical polar grid with extensive domain decomposition in axisymmetry (2D) and in full 3D.

Gravity: We use a multi-pole method to solve the gravitational Poisson equation, with a correction in the monopole term for general relativity to get the structure of the neutron star correct.

Hydrodynamics: We use a finite-volume Riemann solver with PPM reconstruction in a dimensionally split Lagrangian + Remap (PPMLR) scheme based on the astrophysics VH1 code.

Equations of state: We use two nuclear equations of state loaded onto localized tables with thermodynamic variables and derivatives computed from the tables.

Nuclear network: We solve the thermonuclear kinetic equations, where needed, with an operator-split, implicit solver utilizing temporal sub-cycling (relative to the hydrodynamic step). It is a fully integrated copy of the XNet astrophysical thermonuclear reaction network code. For most CHIMERA simulations, the composition is limited to 17 isotopic species, but we are now exploring simulations that evolve 150 species.

Neutrino transport: We use a flavor-coupled, spectral, flux-limited diffusion algorithm with the flux limiter tuned to match full transport methods solved in spherical symmetry independently for each solid angle element (dθ, dϕ) using an implicit, multi-variable Newton-Raphson method. The independence of the transport solutions allows each "ray" to be solved on a separate MPI task.

Neutrino opacities: These are computed as needed using various approximations and/or codes from the literature and stored in a local table that is shared for all points on each MPI task. The table is interpolated numerically to get the values and derivatives needed to build the Jacobian for each iteration of the transport solve.

The transport (opacity interpolations, Jacobian building, linear solver) and the nuclear network (for larger networks) are the most computationally demanding parts and are thoroughly threaded (OpenMP).

8.2.3 HPC Resources Used Today

8.2.3.1 Computational Hours

NERSC: 2.1 million in 2013, 20 million in 2014, allowing for many 2D simulations on Edison and Hopper. We also have used Darter and Kraken in the past. For 3D we have used Titan (70 million hours in 2013, 85 million in 2014) and Mira in the past.

INCITE: 60 million in 2013, 85 million in 2014, allowing for one to two 3D simulations.

8.2.3.2 Parallelism

Our MPI parallelism is limited to assigning one transport solve on each MPI task, which is also our typical running condition. In 2D, the number of transport solves is the number of grid points in latitude, θ. In 3D, this is multiplied by the number of grid points in longitude, ϕ.

Our typical 2D runs at NERSC include O(250) MPI tasks, affording sub-degree resolution. With the current state of our OpenMP implementation, we can use two to four cores per

MPI task with reasonable efficiency for the small 17-species composition we use for most runs. With the larger 150-species network, we can use twice as many OpenMP threads (and cores) with similar efficiency.

Our current 3D run at OLCF uses 180x180 (32,400) MPI tasks with two threads. To match the resolution of our 2D simulations, we would eventually like to reach about 100,000 (250 × 500) MPI tasks. The OpenMP scaling in 3D is similar to 2D.

We have tested at up to 131,072 cores in 3D without OpenMP and could conceivably do so with several cores for each MPI task at that level.

Our simulations require many—O(10^6)—time steps computed in sequence with synchronization required within each time step, to follow the supernova over the 1-2 seconds when the explosion develops. As a result, they consume 1,000 or more wall-clock hours per MPI-task. Higher angular resolution reduces the size of the time steps (requiring more to complete). As a result, fully resolved 3D simulations would cost hundred of millions of core hours, placing them outside the cost profile of current allocation sizes.

Both strong and weak scaling are important. We have spent much effort to ensure that our weak scaling will allow us enough angular resolution to make reasonably (though not ideally) resolved 3D simulations with 10,000-30,000 solid-angle elements on the same number of cores. More recently, especially the last year, we have concentrated on implementing OpenMP for additional strong scaling, using more than one core per solid-angle element (transport and nuclear network solution domain). This has also enabled greater flexibility to exchange MPI tasks for OpenMP threads within a node.

8.2.3.3 Scratch Data

For 2D models, roughly 0.1 TB per model of scratch data is required. With as many as 10 models running simultaneously, this requires 1 TB of scratch space in total.

For 3D models, roughly 40 TB per model of scratch data is required. With typically only 1 model running at any given time, the total is 40 TB.

8.2.3.4 Shared Data

Our communal analysis efforts can require the entirety of several models to be available in the shared storage area at any one time. This would require roughly 2 TB for five 2D models and 200 TB for a single 3D model. In 2013 we stored 5.5 TB in the NERSC /project file system.

8.2.3.5 Archival Data Storage

Our current usage in NERSC HPSS includes the four models that were run in 2012 and 2013. This represents roughly 1.5 TB. Our allocation in 2014 will allow us to add perhaps another dozen models, raising the total to 16 models and roughly 6 TB. Combined with all our other needs for archival data storage, we had 76 TB stored in HPSS at NERSC in 2013. This had increased to about 100 TB by mid-2014.

Our current 3D models are archived at OLCF, consuming approximately 200 TB for a completed model.

8.2.4 HPC Requirements in 2017

8.2.4.1 Computational Hours Needed

We expect to need 20-40 million hours for 2D runs to support the continuation of the scientific program currently running at NERSC.

For a science program based on 3D models, three to four 1-degree resolution simulations per year would require 200-300 million hours (conventional) from all large allocation sources (including those listed below.)

We expect to compete for allocations at the level listed above for our 3D science program through INCITE and NSF PRAC; however, we have no guarantee of success despite our history of such.

For the 2D program, the increase in hours needed is due to computing a larger number of models to investigate the wide range of uncertainties in the supernova problem and including more accurate (and expensive) physics, most notably, realistic nuclear networks.

Any time allocated for 3D simulations in the future would be entirely new to our NERSC allocations.

8.2.4.2 Parallelism

For 2D, we expect something similar to today (up to 1,000 cores per run).

For 3D, we would expect 60,000 “transport rays” with two to four cores each, for a total of 120-240,000 cores, distributed on 60,000 or fewer MPI tasks.

Provided the allocations are large enough to allow 1,000 hours per core, we could reasonably use up to about 250,000 cores to achieve resolutions in latitude and longitude comparable to our current 2D simulations.

8.2.4.3 I/O

We record 5-10 restart/checkpoint files every hour of operation using a checkpoint facility built into the code. For 2D, the data stored is about 400 GB per simulation. This grows to 200 TB per simulation in 3D. For 3D, to meet the I/O time percentage below and rate above, the 80 GB restart file would need to be written in 10-20 seconds, or about 4-8 GB/s.

We try to keep the I/O cost to less than 5% by balancing the restart output frequency with the temporal resolution required to capture the dynamical features in the simulation during analysis.

Our restart I/O is more parallel than serial and is written from several hundred MPI tasks to each of tens of files, with the number of files tuned to maximize performance.

8.2.4.4 Future Data Needs

In 2017, we expect to need 80 TB of temporary scratch disk space, 400 TB of NERSC project space (globally accessible shared data), and 2,000 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to the increasing number of models and improved resolution in 3D.

8.2.4.5 Memory Required

We have expended some effort in recent years to reduce CHIMERA's memory footprint. CHIMERA currently uses somewhat less than 1 GB per "transport ray" in 2D or 3D. Using multiple cores per ray has minor impacts on the memory usage, and pooling multiple rays per MPI task can provide some savings. For 2D, the aggregate memory footprint is about 200 GB, typically spread over 240 MPI tasks. For 3D, the total memory footprint is < 50 TB for models of the size we anticipate running.

8.2.4.6 Emerging Technologies and Programming Models

We have thus far explored utilization of accelerators via library calls (CuBLAS) and OpenACC directives. This seems to be adequate to the tasks we wish to implement on the accelerators.

We run in production on Titan but are currently not using the GPUs. Recent progress should allow us to move matrix solutions within the nuclear network and neutrino transport to the GPU in the next couple of months. GPUs are currently difficult to use because of the small matrix size on each CPU, at least until CUDA5.

We have OpenMP directives in our most computationally intensive sections (transport, network) and are working to expand the coverage throughout CHIMERA. We use up to four-way OpenMP.

We do run on Mira, but our performance was limited by memory constraints.

Porting to, and optimizing for, the Intel MIC architecture is not presently under way or planned.

We do not collaborate with other groups on these issues. We are exploring these technologies ourselves, though we have had conversations with Dongarra's group at the University of Tennessee and with the OLCF scientific computing group. Messer, a member of the latter group, is a part of the CHIMERA team.

Our hope is to rely on directives and libraries, which seem to be sufficient at this time, but time will tell.

Empirically gained expert advice will be crucial for the effective use of near-future architectures; this is a role that NERSC could play. Furthermore (as we note in the next answer), NERSC is perhaps one of the best sources of expertise and interest with respect to software development and porting, especially for those tasks that are not immediately on our production requirements list.

Lack of support for software development and maintenance has long been a problem with respect to DOE (NP, ASCR, and other program offices) funding. However, the significant code changes that will have to be undertaken to exploit near-future architectures over the next decade or so make this problem far worse than it has, perhaps, ever been. Without support for software development and maintenance, our scientific output will diminish over time, regardless of the computational resources made available to us.

8.2.4.7 Software Applications and Tools

We utilize HDF5-parallel for I/O and have used VisIt for visualization with Silo to get data into VisIt. We use LAPACK (and some GPU libraries as well) for linear algebra. We like to compile using a variety of Fortran compilers. Currently these include Intel, Cray, GCC, and PGI. We've written our own analytics.

8.2.4.8 HPC Services

Our in-house data and code workflow system (Bellerophon) would benefit substantially from federated authentication services to facilitate data flow. Any data and analytics services that could be used via this mechanism could be incorporated into Bellerophon via its modular architecture.

8.2.4.9 Data Management Plan

Our data management plan is to store completed models in long-term storage at the computational facilities where the data as generated. In 2D, at 0.4 TB per model, it is feasible to store completed models on hard drives or tape maintained locally (at ORNL). For 3D, network bandwidth and data volume preclude local storage.

8.2.4.10 Additional Data Intensive Needs: Burst Buffer

Given our necessarily very long runtimes, we find little use for a checkpoint/restart staging function via an NVRAM burst buffer. We do (and always will) formulate our own checkpoints and handle their writing and reading. Because the burst buffer will have to be purged between batch submissions, and since we do not foresee an increased rate of checkpointing data (i.e., the FLOP/s we will perform will increase faster than the total amount of data we write to checkpoints), we do not see a benefit in the decreased latency to disk.

The use of a burst buffer as a faster "out-of-core" workspace (for example, for the storage of equation-of-state tables) and the possibility of using the NVRAM as a staging area for intermediate analysis are use cases we would like to explore.

8.2.4.11 What Else?

One of our biggest concerns is babysitting runs that take ~years to finish.

8.2.4.12 Requirements Summary Worksheet

NERSC repo m1373	**Used at NERSC in 2013**	**Needed at NERSC in 2017**
Computational hours (millions)	2.1	200
Typical number of cores* used for production runs	250-6,000	1,000-240,000
Maximum number of cores* that can be used for production runs	120,000	480,000
Data read and written per run	0.4-200 TB	0.4-200 TB

Maximum I/O bandwidth	10 GB/sec	10 GB/sec
Percent of runtime for I/O	<5%	<5%
Scratch file system space	40 TB	160 TB
Shared file system space	5 TB	400 TB
Archival data	76.5 TB	2,000 TB
Memory per node	1 GB per MPI rank	1 GB per MPI rank
Aggregate memory	0.2-50 TB	0.2-50 TB

*Conventional cores

9 Nuclear Science Data Case Studies

9.1 RHIC and LHC Heavy Ion Experiment Program Requirements

Principal Investigator: R. Jeff Porter
NERSC Repositories: rhstar, alice, gc5, m1094

9.1.1 Project Description

9.1.1.1 Overview and Context

Heavy Ion Physics research is a subprogram of the Nuclear Physics program in the DOE Office of Science, organized to study the high temperature phase of QCD matter. Experimental research in the field carried out at the U.S. Relativistic Heavy Ion Collider (RHIC) facility pioneered discoveries of a new state of thermalized matter known as the quark gluon plasma (QGP). The STAR experiment, one of the four original RHIC experiments, provided pioneering measurements during the QGP discovery and in the initial exploration of its properties. The STAR experimental program continues at RHIC providing new measurements with new detector capabilities, higher statistics, and at different beam energies to further map out the properties of the matter and determine threshold energy densities for the onset of QGP formation. The ALICE (A Large Ion Collider Experiment) collaboration constructed and operates a heavy-ion detector designed to exploit the unique physics potential of proton-proton and nucleus-nucleus interactions at even larger collision energies accessible at the Large Hadron Collider (LHC). The science program with ALICE extends the work done at RHIC with a principal goal of studying the physics of this new phase of strongly interacting matter produced at these high temperatures.

The physics of the QGP is studied by exposing properties of heavy ion collisions from the reaction products using a multitude of signals extracted from each collision. These signals include global event properties such as produced particle multiplicities and event-type characterizations, identified particle distributions in transverse energy and momentum, particle correlations, and rare or heavy particle yields. The measurements are repeated for different colliding systems in terms of the beam or collision energy and beam species. The science is conducted by groups of physicists working on one or more of dozens of targeted investigations, each analyzing the common event ensemble to explore different aspects of the collisions and infer information about the QGP and its formation.

Three broad categories of computing activities are required to extract physics from the experimental data. The first is event reconstruction, which takes the pure detector signals from each event and, through a series of pattern recognition algorithms, produces highly refined data in the form of the set of particle trajectories tagged with characteristics (energy, momentum, and mass) emerging from the collision. Event reconstruction is done at specific facilities that have access to the raw detector data and carried out by a team of scientists and engineers on behalf of the entire collaboration for subsequent use in further data analyses. A second category is very similar to, and includes, event reconstruction but is done with simulated rather than real data. Full collisions are simulated (or specific particle simulations are generated and mixed with real event data) to produce simulated detector signals that are then passed through the same event reconstruction algorithms as in the first category. Since these simulations do not require (much) raw data, they are done on a broader set of facilities but are carried out by a team on behalf of the collaboration. The final category is the actual data analysis, in which individual scientists extract the various signals noted above from the real data and evaluate the efficiency of that extraction by doing the same analysis on simulated data sets. In all three cases, the processing is generally "pleasantly parallel," done independent event by independent event. As such, HPC resources have largely been unnecessary for carrying out these computing tasks; however, storage and network resources are critical to the work as individual data sets are hundreds of TB in size, and full analyses may need to be spread over several runs to access PB scales of data.

9.1.1.2 Objectives for 2017

The goals for the next three years in both the STAR and ALICE experiments are a continuation and expansion of their original physics mission to study QCD through exploring the unique phases of nuclear matter formed during these heavy ion collisions. Both experiments expect to see modest increases in annual data rates and volumes over previous years (factors of 2-3x) due to improved accelerator luminosities and new detector capabilities. The ALICE experiment also expects the LHC to deliver PbPb collisions at a top energy of 5.5. TeV/c^2, more than twice as large as in previous PbPb runs during the initial LHC Run 1 period.[21] In addition, the focus in both experiments will shift to take advantage of new detector capabilities recently installed in these experiment systems.

In 2014, STAR installed two new detectors systems—the Heavy Flavor Tracker (HFT) and the Muon Telescope Detector (MTD)—to study the characteristics of charm production in heavy ion collisions. The dynamics of charm production are predicted to depend strongly on the formation and evolution of the QGP during the collision. As a result, new high-precision measurements from STAR of charmed hadron production should uniquely probe QCD physics associated with QGP formation.

During the current LHC long shutdown period, Long Shutdown 1 (LS1), ALICE installed a new patch of the electromagnetic calorimeter detector (EMCal) at the opposite azimuthal angle as the original patch. The full calorimeter now provides the capability to explore back-to-back jet production using the two patches as a single di-jet calorimeter (DCal). By comparing known dijet phenomenology in proton-proton collisions with those observed in PbPb and proton-Pb collisions, the group will be able to make new tomographic studies of the matter produced in the collision.

The new detector capabilities described above will be operational in 2014 (STAR) and 2015 (ALICE). To take full advantage of these new capabilities, new algorithms and analysis techniques will be developed and deployed. The goal by 2017 is to have fresh new insights into the dynamics of QCD matter during the formation and evolution of the QGP.

[21] The LHC schedule includes several multi-year running periods broken up by long shutdown periods of a year or more in length. Run 1 lasted from 2010 to 2013, Run 2 will go from 2015 through 2017, and Run 3 is expected to start around 2019.

9.1.2 Computational Strategies (now and in 2017)

9.1.2.1 Approach

In collider-based experiments like STAR and ALICE, collisions are registered within the detector as independent events. A computing task is the processing of an event collection, done as a set of independent jobs with each job assigned a subset of the collection. This type of processing as noted earlier is "pleasantly parallel" and easily distributed onto clusters of off-the-shelf hardware. Many of the specific investigations that the scientists pursue require very large data samples to make statistically significant measurements. As such, both STAR and ALICE have yearly data acquisition targets upward of 10^9 events corresponding to petabyte-scale annual data volumes.

The common features of STAR and ALICE workflows—high throughput processing of large data volumes on commodity hardware—have led these groups to invest in the NERSC/PDSF system. The PDSF model is different than other NERSC systems in that the cluster hardware is purchased by the scientific groups but operated and located at NERSC, which gives users access to other important NERSC resources, such as HPSS, NGF, or other data-intensive CPU systems like NERSC's Carver system. Use of PDSF by both STAR and ALICE are fully integrated into their distributed computing models. For STAR this includes some grid-based simulation processing as well as giving users with direct access to STAR analysis-ready data copied from its primary facility at BNL. For ALICE, PDSF is fully integrated with the ALICE Grid facility, which is built from approximately 80 facilities around the world and transparently operates as a single batch system to the ALICE user.

9.1.2.2 Codes and Algorithms

The large data sets and large number of physicists accessing the experiment data require use of common software infrastructure. Both STAR and ALICE have built their infrastructures upon the ROOT data analysis framework,[22] an organized set of C++ class libraries with utilities to handle data I/O, data collection management, histograms and plotting, functional fits to data, and many other useful tools. Experiment-specific software is written in C++ and consists of several hundred thousand to more than a million lines of codes with explicit reliance on the ROOT framework. In addition, simulations used for studying experiment efficiencies and systematic effects rely on the GEANT[23] (or GEANT4[24]) Detector Description and Simulation Toolkit to describe the geometry and materials of the detector and their support structures and to simulate the passage of particles through the material. Maintaining such large infrastructures of multiple platforms can be a challenge. STAR software, for example, is validated on a very limited set of platforms, currently versions of Scientific Linux. The ALICE computing team has been able to focus more on code portability, packaging external dependencies within the software releases. ALICE software can and is run on a large variety of modern Linux systems.

STAR software is manually deployed on PDSF and NERSC systems by downloading source codes from the STAR repository and building the full suite of software. A full rebuild is done with each major system change such as OS or ROOT version (typically once a year), while

[22] http://root.cern.ch/drupal
[23] http://wwwasd.web.cern.ch/wwwasd/geant/
[24] http://geant4.web.cern.ch/geant4/

STAR-specific codes are rebuilt more routinely (typically once or twice a month). All ALICE software is maintained and built centrally and distributed as binary executable files through the CVMFS[25] toolset. Local installations of the ALICE software are done by individual users for special runs on the local system, disconnected to the ALICE Grid facility.

9.1.3 HPC Resources Used Today

9.1.3.1 Computational Hours

On PDSF at NERSC, STAR and ALICE have an allocation based on their investment into the system. Those allocations translate to about 7 million (STAR) and 9 million (ALICE) hours for 2014. The PDSF allocations are a fraction of the overall computing in these experiments: about 10% of STAR computing and about 3% for ALICE. STAR computing is carried out primarily at BNL with additional resources at KISTI,[26] while ALICE computing is spread over 80 facilities worldwide. Both groups also used a modest 340,000 hours on the NERSC Carver system to develop production and analysis workflows for other NERSC resources.

9.1.3.2 Parallelism

As mentioned before, almost all of our workflow is perfectly suited for serial processing in which each job uses just one core. Some analyses could be sped up by strategic use of parallel processing, but the overhead in implementing such strategies has generally been considered too high to pursue. There are special uses of parallel processing in the high-level trigger (online) systems that support data-taking operations at the experiment sites; however, those processing codes have not been deployed in carrying out the distributed (offline) computing work done at NERSC and elsewhere. As will be described later, that separation of online and offline processing is expected to disappear for ALICE in the 2018 time frame, making techniques used in parallel processing accessible to the workflow.

These codes are entirely serial today, using only a single core.

Our eventual use of parallel processing will be primarily focused on strong scaling to make up for the loss of increasing per core processor power as clock speeds remain fixed, and to handle the expected fast throughput demands that will exist for data filtering during data taking operations.

9.1.3.3 Scratch Data

Regarding temporary disk space, a typical analysis task will process 50 TB of input data, split into about 1000 individual jobs. The output data is modest by comparison, generally <1% of the input sample. Depending on the workflow, data can be staged to local scratch requiring about 10-20 GB/job. For simulation work, the input data is modest but the output data can be 10-50 GB per job. Thus a simulation task of 1,000 cores may require 50 TB of output scratch. This I/O is basically serial, but with many serial jobs appears parallel.

9.1.3.4 Shared Data

Both STAR and ALICE have project directory space. In addition to the 40 TB allocated from NERSC resources, the group has purchased space on project. STAR has two directories, "star"

[25] http://cernvm.cern.ch/portal/filesystem
[26] http://en.kisti.re.kr/

and “starprod,” which had a combined 250 TB stored in 2013. The “star” space is of general use for software deployment, web access, and user files, while the “starprod” space is specifically restricted for holding STAR data used in analyses. ALICE has one directory, “alice” at 150 TB, which is used by the ALICE-USA group for interactive analysis that complement the grid-based work. Star uses an additional 125 TB of project space using the “gc5” repository.

9.1.3.5 Archival Data Storage

NERSC HPSS has been a critical component to STAR use of PDSF, supporting both user backups and copies of analysis-ready data from BNL. The group managing copies into HPSS have not been able to keep up with data production. Thus, while there is currently on the order of 1.5 PB of STAR data on HPSS, there should be another 2 PB of data stored there. The ALICE-USA project has concluded that formally supporting archival storage for ALICE as a WLCG Tier 1 facility is beyond the scope of the project. As a result, ALICE does not rely on NERSC HPSS storage beyond the tens of TBs needed for archival by individual users.

9.1.4 HPC Requirements in 2017

9.1.4.1 Computational Hours Needed

The overall computing capacities of the STAR and ALICE experiments at all sites are roughly 400 million CPU-hours combined. In 2017, we expect that this will be a least doubled (more than doubled for STAR) to keep pace with the data rates of these experiments. This does not take into account an increase in compute capacities needed to handle new analyses being developed for the new detector systems. It is clear that NERSC in general and PDSF in particular will remain as important resources to the experiments.

A majority of both experiments’ computing needs is currently met at resources outside of NERSC. This will continue into 2017, with NERSC providing 5%-10% of the required resources. These external resources are provided to the experiments from participating institutions around the world. For STAR, the majority of those resources are at BNL. For U.S. operations in ALICE, the other primary site will be at the ORNL CADES facility. Additional resources at TACC and OSC (Ohio State) will likely also contribute to ALICE computing in the U.S.

Based on projections for data increases and analysis goals noted above, we expect our computing needs will approximately double in 2017 from what we have today. So while we are currently using about 15 million hours annually, that number will be at least 30 million in 2017.

9.1.4.2 Parallelism

The primary workflow will likely remain serial in nature, thus well suited for single core processing. However, the processor technology will be changing in such a way that the experiments will need to develop multicore processing schemes. Those efforts are under way, but it is too early to know the mapping of typical cores per job.

One goal in ALICE is to develop a whole-node processing model by 2017 such that a single job can occupy all the cores on a node. In that case, we expect to be able to use on the order of dozens of cores per job.

9.1.4.3 I/O

These applications do not employ checkpoint/restart.

For the following three issues, I will separately address two tasks: simulation (CPU-bound with little input) vs. analysis (I/O-bound with large input requirements). The estimates should be considered with uncertainties (and variations) of factors of 2 or more.

Total data that will be read and write per run in 2017:

Simulation: Output of tens of GBs per core per 1,000 core run, order of 50 TB per run

Analysis: Input of hundreds of GBs per core per 1,000 core run, order 100 TB per run

I/O bandwidth requirement (bandwidth = data read or written / time to read or write):

Simulation: Output of tens of GBs per core per 10 hours, order of 1 MB/sec per core

Analysis: Input of tens of GBs per core per minutes, order 100 MB/sec per core

Maximum percentage of total runtime that should be devoted to I/O:

Simulation: We expect CPU/wall to remain over 90%

Analysis: Our experience has CPU/wall at 70% as typical for our workflow

9.1.4.4 Future Data Needs

In 2017, we expect to need about 100 TB of temporary scratch disk space, 1,000 TB of NERSC project space (globally accessible shared data), and 5,000 TB of storage on NERSC HPSS. The growth in these requirements relative to 2013 is due primarily to steady increase in data volumes during the next three years and, for HPSS, the archival plans for STAR.

9.1.4.5 Memory Required

STAR and ALICE have somewhat different memory footprints, with STAR typically using 1 GB/core while ALICE often requires more than 4 GB/core. The goal for both experiments will be to effectively occupy and use all cores in a given node. ALICE will have more of a challenge to reach that goal but is actively working on a project to leverage concurrency during processing. Both experiments will likely require footprints of 1 GB/core.

9.1.4.6 Emerging Technologies and Programming Models

In this section I will focus almost exclusively on ALICE because that experiment is facing a significant challenge in the period just beyond the 2017 time frame that is motivating an active project to prepare for and leverage next-generation architectures. The challenge will occur during the LHC Run 3 period scheduled to begin in 2019, with planned upgrades to both the ALICE detector and LHC luminosity. These changes will increase the data coming off the detector by a factor of 100 relative to those observed in Run 1, reaching over 1 TB/s or almost 100 PB/day. Such rates far exceed the expected storage capacity for the experiment, which is estimated to remain on the order of tens of GB/s.

The typical solution in our field for handling such bandwidth limitations is to only store events that satisfy some trigger signal; however, this will be at odds with the ALICE physics program goal to focus on obtaining the largest minimum-biased event sample possible. Thus to reduce the stored data volume, the plan is to run full event reconstruction in near real time and store only the reduced data. For example, as is already done in ALICE, the raw pad signals from the TPC detector are used to reconstruct TPC space points in real time. The pad signals are discarded, thereby reducing the data volume by a factor of 5. Other strategies, such as keeping only those space points associated with tracks, are being considered. To accomplish this goal requires fast real-time event reconstruction that leverages new technologies and parallel programming techniques with a focus on strong scaling to keep up with the data rates. At the same time, event reconstruction must be of the same quality as what is currently done later at a more leisurely pace on the distributed offline systems. Essentially, the goal is to restructure ALICE software to be able to effectively run offline quality processing on the complex new architectures required for real time online processing. The ALICE project that is working toward this goal is referred to as the O^2 project, representing an online/offline software merger. Satisfying this goal will make emerging technologies accessible to ALICE software in the 2018 time frame or perhaps earlier.

Our codes do not currently have CUDA/OpenCL directives and do not currently run on GPU systems. The codes also do not currently have OpenMP directives and do not currently run on IBM BG systems. However, porting to and optimizing for the Intel MIC architecture is planned within the scope of the ALICE O^2 project. The ALICE O^2 project is composed of several working groups and includes participation from ORNL staff working with code modifications to support ALICE on Titan. We will try to leverage NERSC resources and user support to help in this multi-year development process.

9.1.4.7 Software Applications and Tools

We expect to rely on much of the same tool base that we currently use, updated to support multi-threading and C++11. The current mapped-out strategy includes a reliance on IPC via 0MQ tools.

9.1.4.8 HPC Services

Integration of any ALICE workflow at NERSC into the ALICE Grid model greatly simplifies use of the facility. To do this without inventing new services requires that we have outgoing connection from the compute nodes to the outside world (such as CERN and other ALICE Grid sites).

9.1.4.9 Additional Data-Intensive Needs

The outgoing network connectivity mentioned above is very important and allows us to support the full range of technologies needed for data and software access that we currently use. We already have a data management plan for our project.

9.1.4.10 Additional Data-Intensive Needs: Burst Buffer

One possible use of such a burst buffer is to construct a temporary PROOF facility for data analysis, such as is done with the PROOF On Demand toolset.[27]

[27] http://pod.gsi.de/

9.1.4.11 Requirements Summary Worksheet

I am filling this out using a combined ALICE and STAR usage and expectation. The processing is expected to happen on a PDSF-like facility where the projects buy into both CPU and storage hardware.

NERSC Repos alice, rhstar, gc5, m1094	Used at NERSC in 2013	Needed at NERSC in 2017
Computational hours (millions)	15 (PDSF)	30 (PDSF)
Typical number of cores* used for production runs	500	1,000
Maximum number of cores* that can be used for production runs	5,000	10,000
Data read and written per run	50 TB	100 TB
Maximum I/O bandwidth ** (note I multiply the per-core rate by nominal # of cores)	50 GB/sec	100 GB/sec
Percent of runtime for I/O	20	20
Scratch file system space	50 TB	100 TB
Shared file system space	525 TB	1,050 TB
Archival data	1,500 TB	5,000 TB
Memory per node (I assume 16-core node)	64 GB	32 GB
Aggregate memory	32 TB	16 TB

*Conventional cores

Appendix A. Attendee Biographies

Ted Barnes is the DOE Program Manager for Nuclear Data and Nuclear Theory Computing in the Office of Science, Office of Nuclear Physics (NP). His responsibilities include supporting computationally intensive research in nuclear physics, for example through the NERSC and SciDAC programs. He has also served as Acting Program Manager for Medium Energy Nuclear Physics, and as a Detailee managed the NP Nuclear Theory program. Before joining DOE he held a joint appointment in the ORNL Physics Division and the University of Tennessee Department of Physics and Astronomy and specialized in research on theoretical hadron spectroscopy and computational condensed matter physics. Barnes is an APS Fellow and holds a Ph.D. in Theoretical Physics from Caltech.

Robert Edwards received a B.S. in Physics and a B.S. in Mathematics in 1984 from the University of Texas at Austin. He obtained his Ph.D. in Physics in 1989 from New York University. He was a postdoc and later a Staff Scientist at the Supercomputer Computations Research Institute (SCRI), Florida State University, from 1989 to 1999, where he was a Co-PI on the Theoretical High Energy Physics grant (DOE). Research included development of new algorithms for calculations within spin systems and for dynamical fermion calculations within lattice QCD as part of the HEMCGC grand challenge project. Also, new methods were developed for calculations with chiral fermion actions that eludicated the role of topology within QCD. In 1998 he shared the Gordon Bell Prize for the development of the QCDSP supercomputer. Since 1999 he has been a Staff Scientist in the Theory Group at Jefferson Lab (JLab). During this time, he jointly started the lattice group within JLab and has led the Lab's effort in developing the infrastructure for the DOE SciDAC program, in particular for the USQCD collaboration, of which he is a member of the Software Committee and the Program Allocations Committee. Current research involves the determination of the highly excited state spectrum of hadrons within QCD as part of the Hadron Spectroscopy Collaboration. This work is using the DOE INCITE computing facilities at ORNL and ANL, as well as NSF, LANL, and NERSC facilities.

Lisa Gerhardt supports the computational needs of the high energy and nuclear physics communities on the PDSF cluster at NERSC. She completed her Ph.D. in Physics from the University of California, Irvine in 2007. Prior to coming to NERSC, Lisa worked in neutrino astrophysics with the IceCube group at Lawrence Berkeley National Laboratory.

Grapham Heyes is the Data Acquisition Group Lead at Thomas Jefferson National Accelerator Laboratory.

Raphael Hix is a research staff member in the Physics Division at Oak Ridge National Laboratory and a Joint Faculty Associate Professor in the Department of Physics and Astronomy at the University of Tennessee.

Péter Petreczky is a research scientist in the Physics Division at Brookhaven National Laboratory. He holds a PhD degree in Physics from Eötvös University, Budapest, Hungary. His interests include finite temperature field theory, Lattice QCD, physics of quark gluon plasma.

R. Jefferson ("Jeff") Porter currently works to support computing needs of experimental High Energy Nuclear Physics (HENP) groups at NERSC and is leading a project to build an ALICE production grid facility in the U.S., co-located at NERSC/PDSF and Lawrence Livermore National Laboratory. Porter earned his Ph.D. in Nuclear Physics from the University of California, Davis and has been a member of several HENP experiments: DLS at LBNL, NA49 and ALICE at CERN and STAR at BNL's Relativistic Heavy Ion Collider (RHIC). As the first liaison to NERSC for the Nuclear Science Division (NSD), he supported initial operation of the NERSC PDSF cluster and participated in the HENP Grand Challenge Collaboration, developing data-mining tools to access large data sets from the RHIC experiments. Porter spent several years as Database and Deputy Computing Leader for the STAR experiment at BNL before returning to the west coast with the STAR group at the University of Washington. He rejoined LBNL in 2006 to work with the Open Science Grid (OSG), operating testbeds for OSG and supporting deployment of grid middleware on NERSC systems. In 2009, he began to directly support ALICE and STAR computing operations at NERSC for the NSD.

Sofia Quaglioni has been a Scientific Staff Member in the Physics Division of the Physical and Life Sciences Directorate at Lawrence Livermore National Laboratory since 2009. She received a Ph.D. in Physics from the University of Trento, Italy in 2005 and earned an Early Career Award from the DOE Office of Science in 2011. Her research is dedicated to the theoretical and computational development of new approaches aimed at advancing the state of the art in the first-principles description of light-nucleus fusion reactions, which are important for astrophysics modeling and fusion energy applications.

Martin Savage is a Professor of Physics in the Nuclear Theory Group at the University of Washington. His current research interests center around using the numerical technique of Lattice QCD to calculate the properties and interactions of nuclei directly from quantum chromodynamics. He earned is B.Sc. (1983) and M.Sc. (1984) at the University of Auckland in New Zealand, and his Ph.D. from Caltech (1990). After postdoctoral positions at Rutgers University (1990-1991) and the University of California, San Diego (1991-1993), he became an Assistant Professor at Carnegie Mellon University (1993-1996) and then moved to the University of Washington in 1996. Savage has held a SSC Fellowship and a DOE Outstanding Junior Investigator award. He has published papers in a number of areas of subatomic physics, including experimental nuclear physics, theoretical particle physics, theoretical nuclear physics, and lattice QCD and is a founding member of the NPLQCD (Nuclear Physics with Lattice QCD) collaboration.

Sergey Syritsyn is a postdoctoral fellow in theoretical physics at Lawrence Berkeley National Laboratory.

James Vary is Professor of Physics at Iowa State University, specializing in theoretical nuclear physics with an emphasis on *ab initio* solutions of quantum many-particle systems and quantum field theory. He received his graduate degree in Nuclear Theory from Yale University and has held positions at MIT and Brookhaven National Laboratory before joining the faculty at Iowa State University. He has also held Visiting Professorships at Caltech, University of Heidelberg, The Ohio State University, and Stanford University. He held the position of Distinguished Visiting Professor at The Ohio State University in 1987-1988 and served as Director of the International Institute of Theoretical and Applied Physics (IITAP) at Iowa State University from 1993-2000, which carried out programs with joint sponsorship from UNESCO, foundations, corporations, and Iowa State University. Vary is the author of more than 380 refereed publications and editor of 4 conference proceedings books. He has delivered more than 500 invited lectures in more than 50 countries and has mentored 16 Ph.D. students and six Masters of Science students. He leads a 10-member nuclear theory group supported by the DOE and NSF. As part of his research activities, he is the Principal Investigator of a DOE Office of Science grant, a DOE SciDAC (NUCLEI) grant, and an NSF Peta-apps grant. Vary is the Principal Investigator of an INCITE award for 204 million CPU hours in 2014 on Titan at ORNL and Mira at Argonne. He has held several elected offices in physics, including election as Member of the Executive Board of the American Physical Society and is a Fellow of the American Physical Society and Recipient of an Alexander von Humboldt Senior Scientist Award, among other honors.

Michael Zingale is an Associate Professor of Physics and Astronomy at Stony Brook University. Michael earned a B.S. in Physics and Astronomy (1996) from the University of Rochester and a Ph.D. in Astronomy and Astrophysics (2000) from the University of Chicago. He was part of the Flash Code development team that won a Gordon Bell Prize in 2000, and received a Presidential Early Career Award for Scientists and Engineers (PECASE) through DOE NNSA in 2005, and an Outstanding Junior Investigator award for the DOE Office of Nuclear Physics in 2006. Michael's research involves the development of new algorithms for efficiently modeling convection in stellar interiors. Together with computational scientists at the Center for Computational Sciences and Engineering, he co-developed and publicly released the Maestro low Mach number hydrodynamics code for stellar convective flows. He applies the Maestro to studies of early phases of Type Ia supernovae, novae, and X-ray bursts.

NERSC Editors

Richard Gerber is NERSC Senior Science Advisor and User Services Group Lead and, with Harvey Wasserman, organizes the NERSC High Performance Computing and Storage Requirements Reviews for Science and edits the reports. He holds a Ph.D. in physics from the University of Illinois at Urbana-Champaign, specializing in computational astrophysics; he held a National Research Council postdoctoral fellowship at NASA-Ames Research Center 1993-1996; and has been on staff at NERSC since 1996.

Harvey Wasserman is a member of the NERSC User Services Group and helps to organize the NERSC High Performance Computing and Storage Requirements Reviews.

Appendix B. Meeting Agenda

Tuesday, April 29		
Time	Topic	
8:00 AM	Informal discussions	
8:30 AM	Welcome, Overview of Requirements Reviews	Richard Gerber, NERSC
8:45 AM	The View from ASCR	Barbara Helland, Dave Goodwin, ASCR
9:00 AM	NP Program Office Research Directions	Ted Barnes, NP
9:30 AM	NERSC Ten-Year Plan	Sudip Dosanjh, NERSC Director
10:00 AM	AM Break	
	Lattice QCD Case Studies	
10:15 AM	Lattice QCD for Cold Nuclear Physics	Martin Savage, University of Washington
10:45 AM	Hadron Spectroscopy with Lattice QCD	Robert Edwards, JLab
11:15 AM	Nucleon Structure on a Lattice	Sergey Syritsyn, BNL
11:45 PM	QCD Thermodynamics at High Temperature	Peter Petreczky, BNL
12:15 PM	Group Photo	
12:45 PM	Working Lunch Presentation. "Transitioning to NERSC-8 and Beyond: The NERSC Application Readiness Effort"	Harvey Wasserman, NERSC
	Nuclear Structure Case Studies	
1:15 PM	ab initio Nuclear Structure	James Vary, Iowa State
1:45 PM	ab initio Calculations of Nuclear Reactions and Exotic Nuclei	Sofia Quaglioni, LLNL
	Nuclear Science Data Requirements	
2:15 PM	ASCR Data Activities	Richard Carlson, ASCR
2:45 PM	Computing and Storage Requirements and Plans for Experimental Nuclear Physics at Jefferson Lab	Graham Heyes, JLAB
3:15 PM	PM Break	
3:30 PM	RHIC/LHC Heavy Ion Program Requirements	Jeff Porter, LBNL
4:00 PM	PDSF @ NERSC Update	Lisa Gerhardt, NERSC
4:15 PM	NERSC's Data Services Plan	Katie Antypas, NERSC
4:30 PM	Cross-cutting issues for Nuclear Science Data	All
5:00 PM	Adjourn	

Wednesday, April 30		
8:00 AM	Informal discussions	
	Nuclear Astrophysics Case Studies	
8:30 AM	Convection in X-ray Bursts	Michael Zingale, SUNYSB
9:00 AM	Core Collapse Supernovae	Raph Hix, ORNL

9:30 AM	AM Break	
9:45 AM	High-level findings discussion	Richard and Harvey
10:45 AM	Report schedule and contents	All
11:00 AM	Case Study Report Refinement and Discussions	All
12:00 PM	Working Lunch: Case Study Breakout Sessions	All
1:00 PM	Adjourn	

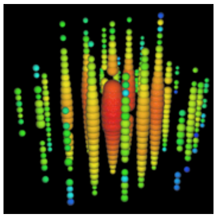
Appendix C. Abbreviations

ALCF	Argonne Leadership Computing Facility
AMR	Adaptive Mesh Refinement
API	Application Programming Interface
ASCR	Advanced Scientific Computing Research
AY	Allocation Year
CUDA	Compute Unified Device Architecture
EIC	Electron Ion Collider
ESnet	DOE's Energy Sciences Network
FFT	Fast Fourier Transform
FNAL	FermiLab National Accelerator Laboratory
FRIB	Facility for Rare Isotope Beams
GCR	Generalized Collisional-Radiative
GPU	Graphical Processing Unit
GPGPU	General Purpose Graphical Processing Unit
GPU	Graphical Processing Unit
HDF	Hierarchical Data Format
HPC	high-performance computing
HPSS	High Performance Storage System
I/O	input output
IDL	Interactive Data Language visualization software
INCITE	Innovative and Novel Computational Impact on Theory and Experiment
LANL	Los Alamos National Laboratory
LBNL	Lawrence Berkeley National Laboratory
LHC	Large Hadron Colider
LLNL	Lawrence Livermore National Laboratory
MD	Molecular Dynamics
MIC	(Intel) Many Integrated Core architecture
MKL	(Intel) Math Kernel Library
MPI	Message Passing Interface
NERSC	National Energy Research Scientific Computing Center
NetCDF	Network Common Data Format
NGF	NERSC Global Filesystem

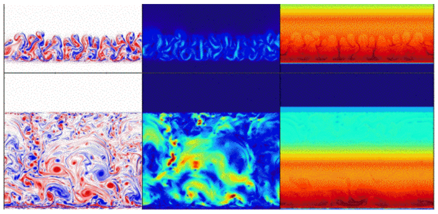
OLCF	Oak Ridge Leadership Computing Facility
ORNL	Oak Ridge National Laboratory
OS	operating system
PDE	Partial Differential Equation
PDSF	NERSC’s Parallel Distributed Systems Facility
QCD	Quantum Chromodynamics
RHIC	Relativistic Heavy Ion Collider
SC	DOE's Office of Science
SciDAC	Scientific Discovery through Advanced Computing
SLAC	SLAC National Accelerator Laboratory
SN	supernova
XRB	X-Ray Burst

Appendix D. About the Cover Images

Image showing a portion of NERSC's "Hopper" system, a Cray XE6 installed during 2010. Hopper is NERSC's first peta-FLOP resource, with a peak performance of 1.28 PetaFLOPs/sec, 153,216 compute cores, 212 Terabytes of memory, and 2 Petabytes of disk. Hopper placed number five on the November 2010 Top500 Supercomputer list.



Schematic image showing one of the two ultra-high-energy neutrino events detected by the IceCube experiment in 2013. The colored spots indicate where the optical modules detected a particle; red is where the particle interaction began. The size of the colored spot refers to the intensity of the Cherenkov light that was emitted. For more about the IceCube Collaboration, see http://icecube.wisc.edu.



Montage depicting evolution of the magnitude of vorticity (left column), Mach number (middle column), and specific energy generation rate (right column) from a simulation of two-dimensional convection in a mixed H/He accretor by Malone, Zingale, Nonaka, Almgren, and Bell using the Maestro code. Two times are shown: 6 × 10−4 s at the bottom and a portion of the result from 3 × 10−4 s above that. From "Multidimensional Modeling of Type I X-ray Bursts. II. Two-Dimensional Convection in a Mixed H/He Accretor," The Astrophysical Journal, 788:115 (12pp), 2014 June 20.

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal 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 its 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, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.