National Computational Infrastructure for Lattice Gauge Theory: Accomplishments and Opportunities

Lattice QCD Executive Committee

R. Brower, (Boston U.) N. Christ (Columbia U.), M. Creutz (BNL),
P. Mackenzie (Fermilab), J. Negele (MIT), C. Rebbi (Boston U.),
S. Sharpe (U. Washington), R. Sugar (UCSB) and W. Watson, III (JLab)

I. Introduction

The objective of the SciDAC Project *National Computational Infrastructure for Lattice Gauge Theory* is to construct the computational infrastructure needed to study quantum chromodynamics (QCD), the fundamental theory of the strong forces of subatomic physics. Nearly all high energy and nuclear physicists in the United States working on the numerical study of QCD are involved in this project, and the infrastructure created will be available to all. The project includes the development of community software for the effective use of terascale computers, and research and development of specialized computers for the study of QCD.

The long term goals of high energy and nuclear physicists are to identify the fundamental building blocks of matter, and to determine the interactions among them that lead to the physical world we observe. The Department of Energy supports major experimental, theoretical and computational programs aimed at reaching these goals. Remarkable progress has been made through the development of the Standard Model of High Energy and Nuclear Physics, which provides fundamental theories of the strong, electromagnetic and weak interactions. This progress has been recognized through the award of Nobel Prizes in Physics for the development of each of the components of the Standard Model: the unified theory of weak and electromagnetic interactions in 1979, and quantum chromodynamics, the theory of the strong interactions, in 1999 and 2004. However, our understanding of the Standard Model is incomplete because it has proven extremely difficult to determine many of the most interesting predictions of QCD, those that involve the strong coupling regime of the theory. To do so requires large scale numerical simulations within the framework of lattice gauge theory.

The scientific objectives of lattice QCD simulations are to understand the physical phenomena encompassed by QCD, and to make precise calculations of the theory's predictions. Lattice QCD simulations are necessary to solve fundamental problems in high energy and nuclear physics that are at the heart of the Department of Energy's large experimental efforts in these fields. Major goals of the experimental programs in high energy and nuclear physics on which lattice QCD simulations can have an important impact are to: 1) verify the Standard Model or discover its limits; 2) understand the internal structure of nucleons and other strongly interacting particles and 3) determine the properties of strongly interacting matter under extreme conditions, such as those that existed immediately after the "Big Bang" and are produced today in relativistic heavy–ion experiments. Lattice QCD calculations are essential to research in all of these areas.

The numerical study of QCD requires very large computational resources, and has been recognized as one of the grand challenges of computational science. The advent of terascale computing,

coupled with recent improvements in the formulation of QCD on the lattice, provide an unprecedented opportunity to make major advances in QCD calculations. The infrastructure created under this SciDAC grant will play a critical role in enabling the U.S. lattice QCD community to take advantage of these opportunities. In particular, the hardware research and development work provides the groundwork for the construction of dedicated computers for the study of QCD which the DOE's High Energy and Nuclear Physics Programs have indicated they will fund over the next four years, and the community software will enable highly efficient use of these computers and the custom designed 12,228 node QCDOC computer recently constructed at Brookhaven National Laboratory (BNL).

II. Achievements

In this section we briefly review the major achievements under our current SciDAC grant.

A. Software Development

The objective of our software effort has been to create a unified programming environment, the QCD Application Programming Interface (QCD API), that will enable members of the U.S. lattice gauge theory community to achieve high efficiency on terascale computers, including the QCDOC, commodity clusters optimized for QCD, and commercial supercomputers. Among the design goals have been to enable users to quickly adapt codes to new architectures, easily develop new applications, and preserve their large investment in existing codes.

The QCD API is an example of an application specific code base serving a national research community. It exploits the special features of QCD calculations that make them particularly well suited to massively parallel computers. These include regular grids or lattices, which allow for near perfect load balancing, and regular and predictable communication patterns, which enable latency to be hidden by overlapping data movement with local computation.

The QCD API has a three level structure:

- Level 3: Highly optimized, computationally intensive subroutines.
- Level 2: Data Parallel language to enable rapid code development
- Level 1: QCD specific Message Passing and Linear Algebra routines

All the fundamental components have been implemented and are in use on the U.S. QCDOC hardware at BNL, and on both the switched and mesh architecture Pentium 4 clusters at Fermi National Accelerator Laboratory (FNAL) and Thomas Jefferson National Accelerator Facility (JLab). The software code and documentation can be found at links from **http://www.usqcd.org**. Here we briefly describe each component of the QCD API.

Message Passing: The message passing API, QMP, defines a uniform subset of MPI-like functions with extensions that (1) partition the QCD space–time lattice and map it onto the geometry of the hardware network, providing a convenient abstraction for the Level 2 data parallel API (QDP); (2) contain specialized routines designed to access the full hardware capabilities of the QCDOC network and to aid optimization of low level protocols on networks in use and under development

on clusters. Release 2.0 of the QMP message passing API is published on **www.usqcd.org/usqcd-software**, along with complete documentation. It includes: (1) a message passing library design and binding for both C and C++, (2) code implementing QMP atop MPI for portability and (3) an implementation atop VIA and gigabit ethernet to support the new gigabit ethernet mesh clusters at JLab. The implementation for the QCDOC has important hardware functionality, such as the ability to start twenty-four different communications with a single cpu instruction, and persistent storage in the communications hardware of the data pattern for repeated communications transfers. There is a basic test suite to verify each implementation.

Linear Algebra: The linear algebra routines, QLA, can be used in combination with QMP to develop complex data parallel operations in QDP or in existing C or C++ code. The C implementation has on order 24,000 functions generated in Perl, with a full suite of test scripts. The number of functions in the C++ implementation of QLA is considerably reduced by making extensive use of the language's class structure and of operator overloading. On the QCDOC, assembly language coding for a small set of QLA routines has also led to a substantial boost in performance for C code. For the C code, the most heavily used QLA routines have been optimized for Pentium 4 clusters using SSE instructions. For the new C++ code base, optimized QLA routines have been generated using an assembler tool called BAGEL written by Peter Boyle of the UKQCD group in Edinburgh. We anticipate a similar strategy for other processors, such as the Altivec instruction set, which is available on the PowerPC(G5), if they should supplant the the Pentium 4 in future clusters.

Data Parallel Interface: Level 2 (QDP) contains data parallel operations that are built on QMP and QLA. The C implementation is being used to improve performance of the MILC code, a large, publicly available suite of applications. Despite the fact that the MILC code has been carefully optimized over its fifteen year lifetime, rewriting computationally intensive subroutines in QDP makes a significant improvement in its performance. Chroma, an entirely new application code base, has been written *di nuovo* in the C++ implementation of QDP. QDP allows extensive overlapping of communication and computation in a single line of code. By making use of the QMP and QLA layers, the details of communications buffers, synchronization barriers, vectorization over multiple sites on each node, etc. are hidden from the user.

Level 3 Subroutines: A very large fraction of the resources in any lattice QCD simulation go into a few computationally intensive subroutines, most notably the repeated inversion of the Dirac operator, a large sparse matrix. To obtain the level of efficiency at which we aim, it is necessary to optimize these subroutines for each architecture. For example, the performance of the assembly coded inverter for the QCDOC is 42% of peak for the Domain Wall quark action and 45% of peak for the Asqtad action. These are the two quark formulations that will be used in the initial work on the QCDOC. For the QCDOC running at 400 MHz these numbers correspond to a total sustained performance of 4.13 and 4.42 teraflop/s respectively for the full 12,288 processor machine. These results are for very small local volumes volumes: 4^4 lattice sites per processor for DWF and $5^3 \times 4$ for Asqtad. Small local volumes are desirable because they enable large numbers of processors to be applied to a single calculation; however, they place great strain on the inter-processor communications system. The DWF result is in double precision, while the Asqtad one is from a new version of the Asqtad inverter that uses single precision and allows lattice dimensions on individual processors to be odd. Both properties are important for the initial runs planned for the QCDOC. Level 3 codes written with SSE2 instructions achieve 1.4 to 2.0 gigaflop/s per processor for clus-

ters built at FNAL and JLab during the past year with 2.8 GHz Pentium 4 processors and 800 MHz front side buses. However, optimal performance is achieved for larger local volumes than with the QCDOC.

The basic components are now in place to perform highly efficient production calculations on the target architectures: QCDOC and optimized clusters. Since all QCD API code can be run in a C or C++ environment with MPI, it can be used without change on commercial supercomputers and desktop workstations.

Finally, we note that considerable work has been done under the BNL component of our project on the QCDOC operating system, which is now in use. This effort is, of course, critical to the use of that machine.

B. Hardware Research and Development

The second major activity under the Lattice QCD SciDAC grant has been the design, construction and operation of commodity clusters optimized for the study of QCD. This work has taken place at FNAL and JLab. The objective has been to provide computing platforms to test the QCD API, and to determine optimal configurations for the terascale clusters planned for FY 2006 and beyond. The clusters that have been constructed are being used to carry out important research in QCD.

The bottleneck in QCD calculations on clusters, as on commercial supercomputers, is data movement, not cpu performance. QCD calculations take place on four-dimensional space-time grids or lattices. To update variables on a lattice site, one only needs data from that site and a few neighboring ones. The standard strategy is to assign identical sub-lattices to each processor. Then, one can update lattice points on the interior of the sub-lattices, for which all the relevant neighbors are on the same processor, while data is being collected from a few neighboring processors to update lattice sites on the sub-lattice boundaries. This strategy leads to perfect load balancing among the processors, and, if the computer and code are properly tuned, to overlap of computation and communications. If the sub-lattices are small enough so that they fit into the processor's cache, then single processor performance can be very fast. However, single processor performance falls off significantly if the sub-lattice does not fit into cache since data must then be repeatedly moved between cache and main memory. On the other hand, as the size of the sub-lattices is decreased, the strain on the inter-processor communications is increased. Thus, a careful balance between processor performance, memory bandwidth and latency, and inter-processor bandwidth and latency is needed to optimize performance.

Under this grant a wide range of processors and communications systems has been evaluated. At present, nodes containing a single Pentium 4 processor provide the best price/performance for QCD applications, but that can easily change. So, constant attention to market developments is necessary. Both switched and mesh communications systems have been studied. Myrinet and Infiniband fabrics have been tested for switched clusters, and gigabit ethernet has been used for the mesh ones. A total of seven clusters of various sizes have been built with an eighth under construction. The most recent gigabit Ethernet cluster was built at JLab. It has 384 2.8 GHz Xeon processors, and sustains approximately 500 gigaflop/s. The latest switched architecture cluster, which is currently under construction at FNAL, has 260 3.2 MHz P 640 nodes and an Infiniband

network. It is expected to sustain approximately 350 gigaflop/s. It will be doubled in size and performance in late FY 2005. The experience gained with these prototype clusters will enable us to build highly cost effective terascale production clusters in the future.

The prototype clusters are used to carry out state of the art physics projects by members of the U.S. lattice gauge theory community. They are expected to sustain a total of approximately 1.7 teraflop/s by the end of the grant period.

A 12,288 QCD on a Chip (QCDOC) computer has recently been constructed for our community with funding from the DOE's Advanced Scientific Computing (ASCR), High Energy Physics (HEP) and Nuclear Physics (NP) Programs. The QCDOC was designed by a group of lattice gauge theorists centered at Columbia University in collaboration with colleagues at IBM. The key component is an Applications Specific Integrated Circuit (ASIC) that incorporates cpu, memory and communication on a single chip. The 50 million transistor ASIC was designed with the aid of IBM technology, and manufactured by IBM. The QCDOC is expected to sustain between 3.0 and 5.0 teraflop/s depending on the specific project and lattice QCD formulation.

Although the design and construction of the QCDOC was funded outside the lattice QCD SciDAC grant, a significant portion of the SciDAC software effort has been directed towards it. A question often asked regarding costum designed computers is whether diverse user groups not involved in their design can make effective use of them. In this instance, the QCD API has enabled users to port existing codes to the QCDOC and to easily generate new codes for it, achieving high efficiency in both cases.

III. Opportunities

Work done under the SciDAC Program has provided major opportunities for research in lattice QCD. Below we indicate a number of ways in which this work could very profitably be built upon and enhanced under an extension of the Program. First, we will make some general comments that we believe apply to most, if not all of the application areas being supported by SciDAC.

The SciDAC Program has been enormously successful in producing community codes for terascale computing. It has done so by providing the support needed to enable applications scientists to work on such codes, and by encouraging collaborations among application scientists, applied mathematicians and computer scientists in their development. We strongly recommend that this very successful approach be continued. If the DOE is to fully capitalize on the large investments in human and financial resources that have gone into them, it is important that these codes continue to evolve, since once codes become stagnant, they quickly become obsolete. It is also important that codes be properly maintained and ported to new platforms, and that appropriate support be provided for the code users.

The very successful SciDAC software cannot enable the science for which it was created without terascale computers to run on. We therefore strongly recommend that the DOE provide hardware with the capability and capacity to meet the needs of the SciDAC application areas, either as part of an extension of the SciDAC Program, or as part of a separate program. We recognize that the bulk of these computing resources for areas other than lattice QCD will be provided by

commercial supercomputers located at DOE centers. However, we have clearly demonstrated that for our field designing hardware and software that specifically takes into account the structure of the computation is highly advantageous. We are delighted that the HEP and NP Programs intend to fund dedicated hardware for lattice QCD which will enable us to continue to demonstrate the advantages of this approach while accelerating our scientific work. We expect this approach to be useful in some other fields, and we urge that work in this direction by us and by others be supported in an extended SciDAC Program. Indeed, as was recognized in a recent review, the success of the new project to build and operate dedicated hardware for lattice QCD depends on the continuation of our software development and hardware research and development efforts.

There are two specific directions that the lattice QCD effort would pursue under an extension of the SciDAC Program.

A. Future Software Development

The basic QCD API is set, and high performance code built upon it is ready for use on the QCDOC and clusters. To take full advantage of the API additional emphasis must now be placed on applications codes. Chroma, the first large body of code written in QDP, must be completed and optimized; additional components of the MILC code should be ported to QDP; and new applications need to be written. The development of new formulations of QCD on the lattice has played an important role in advancing our field in recent years, and additional ones will surely be introduced in the near future. Each new formulation will require a new Level 3 inverter. In addition, for some formulations, subroutines other than the inverter use significant numbers of cycles, so Level 3 code may need to be written for them as well. Growing use of the QCD API by members of our community will require additional user support, documentation, and revision control.

As new hardware comes into use, it will be necessary to optimize the QMP and QLA libraries for it, and to develop new Level 3 routines. One example for the near future is the IBM BlueGene/L, which appears to be a very promising platform for the study of QCD. QDP++ and Chroma are already running on it, and we plan to optimize the performance of QMP and write Level 3 routines for this platform. To assist in our optimization efforts, our computer science colleagues in Dan Reed's group at North Carolina have developed a high level performance analysis toolkit for the MILC code. We plan to extend this tool to other major codes.

Dedicated hardware for lattice QCD will be located at BNL, FNAL and JLab. We plan to build a unified user's environment, presenting to the users identical batch environments, identical commands for interacting with disk and tape resources, and identical development environments. As this effort matures, the BNL, FNAL and JLab facilities will be operated as a single meta-facility, including data grid capabilities, and virtual batch queues for job submission. In this effort we will make use of the grid tools developed within the SciDAC Program by groups such as the Particle Physics Data Grid.

A very large fraction of the computing resources used in lattice QCD go into Monte Carlo simulations that generate representative configurations of the QCD ground state. The same configurations can be used to calculate a wide variety of physical quantities. Because of the large resources needed to generate configurations, the U.S. lattice community has agreed to share all of those that are generated with DOE resources. To enable this sharing we have created standards for file formats, and

built into the QCD API I/O routines that adhere to them. Thus, all members of our community will be able to access the large data sets we plan to create and archive. We are charter members of the International Lattice Data Grid (ILDG), which seeks to share QCD data internationally. To do so will require the adoption of common standards for data files, agreements on sharing data, and the establishment of an international grid to facilitate transfer of the data. There is significant work yet to be done, but the payoff will be very large, enabling all workers in lattice QCD to greatly enhance their research.

If past history is a guide, new algorithms will be as important as faster hardware in advancing research in lattice QCD. Consequently the software infrastructure must be flexible enough to accommodate the evolution of QCD algorithms. For many years the central algorithmic problem faced by our field has been the inversion of the Dirac operator, a very large sparse matrix. We need both improved algorithms for existing applications, and radically new approaches for problems outside the reach of current methods, such as simulations at finite chemical potential. These problems pose fundamental mathematical challenges with strong relations to analogous problems in other areas of science and applied mathematics. SciDAC offers an ideal setting for this type of algorithmic research by encouraging interdisciplinary collaborations.

The use of multi–grid methods to accelerate the inversion of the Dirac operator has been explored extensively in the past without significant success. However, members of our group have recently begun working with applied mathematicians from the TOPS multi–grid algorithm team on this problem, and have obtained impressive preliminary results. It is very important to continue this work, as even modest gains in performance would have a major impact on the science. In addition, Lüscher has recently introduced a blocking method based on the "Schwarz alternating procedure" that shows promise. This approach too warrants further exploration.

B. Future Hardware Research and Development

Under the current SciDAC grant we have carried out research and development of commodity clusters optimized for QCD. This work has put us in a position to construct highly cost effective terascale clusters in FY 2006 and beyond, which we expect to be funded outside the SciDAC Program. It is important to continue cluster research within an extension of the SciDAC Program in support of the construction project. This work would continue and enhance the very successful program of testing components and building prototype clusters initiated under our current grant.

Another major hardware research and development effort we foresee for an extension of the SciDAC Program is the design of a successor to the QCDOC. The architects of the QCDOC have the very ambitious goal of producing a petascale production machine in FY 2009 at a cost of approximately $0.01 per sustained megaflop/s. Whether or not that goal can be reached, their track record over many generations of machines warrants support for a serious research and development effort.