QCD and the BlueGene

To cite this article: Pavlos Vranas 2007 J. Phys.: Conf. Ser. 78 012080

View the article online for updates and enhancements.

Related content

- News Release
  Charlene Matthewman

- XXXVI Symposium on Nuclear Physics (Cocoyoc 2013)
  Libertad Barrón-Palos, Irving Morales-Agiss and Enrique Martínez-Quiroz

- Towards the petaflop for Lattice QCD simulations the PetaQCD project
  Jean-Christian Anglès d’Auriac, Denis Barthou, Damir Becirevic et al.
QCD and the BlueGene

Pavlos Vranas
Lawrence Livermore National Laboratory, 7000 East Ave, Livermore CA 94550
vranasp@llnl.gov

Abstract. Quantum Chromodynamics is the theory of nuclear and sub-nuclear physics. It is a celebrated theory and one of its inventors, F. Wilczek, has termed it as "... our most perfect physical theory". Part of this is related to the fact that QCD can be numerically simulated from first principles using the methods of lattice gauge theory. The computational demands of QCD are enormous and have not only played a role in the history of supercomputers but are also helping define their future. Here I will discuss the intimate relation of QCD and massively parallel supercomputers with focus on the Blue Gene supercomputer and QCD thermodynamics. I will present results on the performance of QCD on the Blue Gene as well as physics simulation results of QCD at temperatures high enough that sub-nuclear matter transitions to a plasma state of elementary particles, the quark gluon plasma. This state of matter is thought to have existed at around 10 microseconds after the big bang. Current heavy ion experiments are in the quest of reproducing it for the first time since then. And numerical simulations of QCD on the Blue Gene systems are calculating the theoretical values of fundamental parameters so that comparisons of experiment and theory can be made.

1. QCD, a new urgency.

Now is a most interesting time. Heavy ion experiments at the Relativistic Heavy Ion Collider (RHIC) at the Brookhaven National Laboratory (BNL) are unraveling the secrets of nature under extreme conditions [1]. By colliding heavy nuclei at enormous speeds the temperature and density of nuclear matter becomes high enough for a transition to occur from stable nuclear matter to a plasma state. The plasma is comprised of the constituents of nuclear particles, namely quarks and gluons, and is called the quark gluon plasma. This state of matter has not existed in the universe since about 10 microseconds after the big bang. RHIC is continuing to collect information while the Large Hadron Collider (LHC) at CERN, Geneva will also probe the quark gluon plasma soon. With this comes the nearly immediate need to be able to make firm contact between experiment and the underlying theory of sub-nuclear matter, Quantum Chromodynamics (QCD). It is important to verify that QCD describes the behavior of matter under these extreme conditions as seen by experiment. And it is equally important to be able to offer theoretical qualitative and quantitative insight to the properties of the quark gluon plasma as it is further probed by RHIC and LHC.

What makes the theoretical effort unusual is that the theory of QCD under these conditions becomes difficult to handle analytically. However, there is a way to perform many of the necessary calculations from QCD’s first principles. The method for doing so was first introduced in 1974 [2] and is called Lattice Gauge Theory. It provides for a way of simulating QCD from first principles on a
computer. The computational demand for this calculation is very large. Today multi-Teraflops scale supercomputers can perform some of these calculations with an error of about 10% while the need for multi-PetaFlops scale supercomputers is clear if one wishes to achieve errors at the few percent level and be able to explore the properties of QCD in more detail.

It is important to appreciate how fortunate it is that the supercomputers available today and in the near future are just about fast enough to address some of the needs of the experimental efforts at RHIC and LHC. And that the theoretical tools of lattice gauge theory that have been developed over the past 30 years are sharp enough to not only be able to address experiment but also to be able to use efficiently the largest supercomputers of our times. The advances in physics and supercomputing and their convergence provide for a unique opportunity both for science and for technology.

Here I will discuss QCD and Lattice Gauge Theory (Lattice QCD or LQCD for short), the BlueGene/L supercomputer and how LQCD works and performs on BG/L. As an example I will focus to one application of LQCD namely the physics relevant to the transition of nuclear matter to the quark gluon plasma as the temperature is increased to very high values. Of course, LQCD is also being used to address questions relating to a large variety of physical problems ranging from nuclear physics, to the physics beyond the standard model of elementary particles to QCD in extreme conditions just to name a few. For exposure to some of these problems please see [3], [5].

2. QCD in general.
Nuclear particles such as protons and neutrons are made from smaller particles called quarks and gluons. They are held together by the strong nuclear force. This force is the strongest force in nature (the other three being electromagnetism, weak nuclear force and gravity). It acts only at very short distances and is mediated by gluons which carry "color" charge. Color can be thought of in a way similar to electromagnetic charge. Quarks and gluons are believed to be elementary particles; there is no experimental evidence that they are made up of smaller particles. The theory that describes the quarks and gluons and their interactions is Quantum Chromodynamics (QCD).

The quarks inside a nuclear particle are almost free of interaction - a phenomenon called asymptotic freedom. However, if one tries to "pull out a quark from a nuclear particle a strong force develops that keeps the quark from escaping. The force is mediated by the gluons. The gluons form "flux tubes" that connect the quark with its companions inside the nuclear particle. The force becomes stronger with increased separation. At some point it is so strong that there is enough energy in the vacuum to create a new quark-antiquark pair. The antiquark combines with the escaping quark while the new quark goes “back in” to the nuclear particle. As a result, a new nuclear particle is formed and there are no free quarks. This phenomenon is call confinement. No free quarks have been observed today. It is remarkable that a single theory can describe both of these rather different phenomena.

This picture holds at ordinary temperatures and is the reason why ordinary nuclear matter and by extension all matter around us is stable. However, if nuclear matter is heated to very high temperatures (such as generated in the early universe or at RHIC) a transition occurs. At about 200 MeV nuclear matter “melts” and undergoes a phase change to a plasma state of quarks and gluons. The theory of QCD predicts this dramatic phenomenon and LQCD provides some of the most accurate estimates of the transition parameters.

In an inspiring article by one of QCD’s inventors, F. Wilczek [4], a 2004 co-recipient of the Nobel Prize in physics, lists the important properties of QCD: 1) embodies deep and beautiful principles (it is a relativistic quantum field theory); 2) provides algorithms to answer questions (one such algorithm is LQCD); 3) has a wide scope (from nuclear physics to the genesis of the cosmos);
4) contains a wealth of phenomena (asymptotic freedom, confinement and many others);
5) has few parameters (and is therefore simple to describe);
6) is true (has been verified experimentally);
7) lacks flaws (it is fully described by its definition).
Based on these remarkable properties he termed QCD as “… our most perfect physical theory”.

![Figure 1: The lattice.](image)

3. Lattice Gauge Theory

Quarks and gluons are described by fields defined on the four-dimensional space-time continuum. Because space-time is continuous the fields are defined on an infinity of points, which presents a problem for a computer. In order to be able to define the theory in a way amenable to numerical simulation one can define the fields only on the sites and links of a discrete grid of points called the lattice. Typical lattices used today are rectangular. The distance between two adjacent lattice points is called the lattice spacing “a”. The size of the lattice should be large enough to contain the relevant physical processes and for most problems its linear dimension can be a few times the radius of the proton. In this case the number of points where the fields are defined is finite and the system is amenable to computer coding.

This description is deceptively straight-forward. The theory of QCD has a fundamental local symmetry called the gauge symmetry. The fields can be transformed at every point in a special way by different amounts of the symmetry operations and the whole theory remains unchanged. This is a potent symmetry that could have been mutilated by the discretization of space-time into a lattice. But if the fields are defined in a particular way the symmetry is preserved even on the discrete lattice. This is a powerful insight and is part of the reason for the launch of a whole new field of theoretical physics, Lattice Gauge Theory [2].

Furthermore, the discretization of the quark fields is problematic. It turns out that for each quark field that is discretized on the lattice one does not obtain the expected one type of quark. Instead, in four dimensions, one obtains $2^4 = 16$ quark types. This is called the lattice doubling problem. These
unwanted types can be removed at the expense of another symmetry, the chiral symmetry. This
symmetry is global and is a symmetry of the quark fields. One can perform the same symmetry
operation on the quark fields on all lattice sites and the theory will remain unchanged. The chiral
symmetry is very important to the physics of QCD since, among other things, it is responsible for the
masses of the lightest nuclear particles, the triplet of pions. But this symmetry must be sacrificed on
the lattice to cure the doubling problem. Typically this is done in a way that the symmetry is gradually
recovered as the lattice spacing is reduced to zero (the continuum limit) so that the correct physics is
reproduced.

However, there is another way to address the doubling problem by introducing a fifth dimension
that has special boundaries for the quark fields. These boundaries, called domain walls, have the
property that localize the quark fields in their vicinity. As the size of the fifth dimension is increased
chiral symmetry is recovered even at finite lattice spacing. This way the continuum limit of vanishing
lattice spacing has been separated by the chiral limit. Not only does this have many elegant properties,
but also the numerical cost of increasing the lattice sites of the fifth dimension is much less than the
cost of reducing the lattice spacing in all four dimensions (and therefore increasing the lattice sites of
the whole four-dimensional volume). This method is called the Domain Wall Fermion (DWF) method.
It is now one of the mainstream numerical methods. For an introduction see [3], [11].

As illustrated in Figure 1, the quark fields are defined on the lattice sites while the gauge fields
(which are the mediators of the strong nuclear force) are defined on the lattice links. The size of the
fields is also indicated. For an introduction to Lattice Gauge Theory see, for example, [5].

4. The QCD numerical simulation.
The theory is governed by a functional of the fields called the QCD action “S”. The theory can be
simulated numerically using statistical methods that involve the exponentiated action as a “Boltzman”
weight. In particular, molecular dynamics methods are used that “evolve” a field “configuration”
through the configuration space in a statistically significant “trajectory”. Quantities that involve the
fields are calculated for sets of statistically independent field configurations. The result of the
numerical simulation is the average value (and error) of these quantities. This way a wealth of
quantities relevant to experimentally measured properties, such as the mass of the proton or the critical
temperature of the thermal transition, can be calculated.

The calculation of the QCD action “S” involves the inversion of a sparse matrix with dimension
NxN where N is proportional to the number of lattice points. Typically the matrix “connects” fields on
the same lattice point or at nearest neighbor lattice points. The matrix is called the lattice Dirac
operator or simply D-slash. The numerical inversion of this matrix is by far the most computationally
consuming part of the simulation. In many cases about 90% of the simulation time is consumed by
this inversion. Because the matrix is large but sparse it is not stored in memory. Instead it is defined
operationally by its action on column vectors and is inverted by algorithms such as Conjugate
Gradient. Therefore the core operation is the action of this operator on a column vector. The code that
performs this operation is called the QCD kernel. While the full code is several thousand lines long,
the QCD kernel is only a few hundred lines long.

This disproportion of code and computing consumption immediately suggests that if one can
optimize the QCD kernel very well the full QCD numerical application would also perform very well.
Also, because of this the efficiency of the QCD code on supercomputers is measured as the efficiency
of the QCD kernel. In this spirit it is important to optimize the QCD kernel for the hardware at hand
in order to exploit every possible advantage the hardware has to offer. In the past QCD kernels have
been tailored after the massively parallel supercomputers at hand. This is the case for the BG/L QCD
kernel that will be discussed below. Also, many of these considerations have been part of the
development of several supercomputers. For example, the QCDSP (quantum chromodynamics on digital signal processors) and QCDOC (QCD-on-a-chip) supercomputers [6] were developed specifically for the study of QCD and have influenced the design of the IBM BG/L supercomputer.

5. The machines wait for no one; The IBM BlueGene/L

But before one can highly optimize the QCD kernel for BG/L one has to know the hardware features of the machine. Here is a brief description of the relevant hardware features of BG/L. For an extended discussion on BG/L see [7].

![BG/L node diagram](image)

**Figure 2:** The BG/L node.

The BlueGene/L is a massively parallel supercomputer that consists of a large number of nodes. The nodes are connected to form a three-dimensional grid with the topology of a torus. High-speed links interconnect each node with its nearest neighbors. The nodes and links form a virtual cut-through network. Any node can send data to any other node without CPU intervention. Dedicated hardware routers that exist in every node handle the routing.

A single node is a single ASIC microchip. Two nodes are assembled on a small card that contains external DRAM memory. Sixteen such boards are connected to a larger board using dim sockets. These larger boards in turn are connected to a large vertical midplane board using large connectors. A total of 1024 nodes are housed in a "refrigerator-sized" cabinet. Cabinets are connected with cables to form a larger system. Systems with different numbers of cabinets are possible. The largest such system (at LLNL) has 64 cabinets. This system contains a total of $32 \times 32 \times 64 = 65,536$ nodes.

Figure 2 depicts the main components of a node. Each node has two CPU cores (IBM 440 operating at 700 MHz). Each CPU core has two fused-multiply/add (MADD) floating point units. Therefore, each core is capable of executing 4 floating point operations per cycle and the two cores of the node can execute 8 floating point operations per cycle or 5.6 GFlops. Each CPU core has a private 32 KBytes L1 cache. These connect to a single L2 cache. The L2 serves as a multi-stream sequential pre-fetcher and has very little storage capacity. The L2 connects to a 4 MByte on chip L3 cache. The
L3 connects to external DRAM via a double data rate (DDR) controller. Typically the external memory is 0.5 GByte per node.

There are several networks on the machine. The main network is the “torus” network mentioned above. The sophisticated hardware that handles the routing on each node is part of the chip. Each CPU core can send/receive packets of data by writing/reading from hardware mapped addresses that correspond to dedicated SRAMs inside the torus hardware. The torus hardware then responsible for routing the packets to their destination. As a packet travels from node to node, the node torus hardware directs it to the correct exit links until it reaches its destination. The torus hardware is identical on all nodes making the design highly modular. A separate global tree network is also part of the chip and is capable of a combine-broadcast operation in 5 microseconds. Boot, diagnostic and global interrupt networks are also present.

6. The map and coding of QCD on BlueGene/L.

The BG/L architecture suggests a natural map of QCD on the hardware. Because the lattice Dirac operator connects nearest neighbor lattice points one can map the lattice directly onto the three-dimensional torus of nodes with the fourth lattice dimension mapped internally on each node along the two CPU cores. The two CPU cores can then be thought of as independent “virtual nodes”. Communication between the two on chip CPU cores is done via common memory write and read operations.

Depending on the machine size the lattice is divided into smaller equal size sub-lattices so that one sublattice is assigned to each virtual node. Then each virtual node does calculations on the surface of the sub-lattice. In this way the computation and communications load is evenly distributed not only across nodes but also across virtual nodes, i.e. CPU-cores. As a result, the code can fully exploit both CPU cores and their double multiply-add floating point units. Also, the two global sums that are required per Conjugate Gradient iteration are executed using the global tree network. All virtual nodes participate in the global sum by contributing the local sum of their sub-lattice.

Because there is no network hardware DMA (Direct Memory Access) machine the CPU must read/write the data to be communicated from/to memory and then write/read the torus network send/receive SRAMs. These load/store operations cannot be overlapped with computation or other memory access since they occupy the CPU. As a result the communications cost an amount of cycles that affects the sustained performance of the code. It is important to minimize this cost in order to achieve good performance.

In a typical QCD kernel code the amount of data that needs to be brought from/to memory or be communicated across nodes is “fragmented”. Data sizes of about 1 KByte needs to be communicated between operations that act on data of similar size. This makes the code very sensitive to memory and network latencies. Because of this and the fact that communications and computations cannot be overlapped one has to code the QCD kernel very carefully and directly onto the hardware.

Because QCD uses complex valued fields all operations are complex and therefore are naturally suited for execution in the double multiply-add floating point unit. All floating point instructions are multiply-add double floating point instructions. Furthermore, they are carefully arranged in order to avoid CPU pipeline conflicts. Memory storage of the fields is chosen so that minimal pointer arithmetic is needed. Load/store instructions are carefully arranged to take advantage of the cache hierarchy and the CPU’s ability to issue up to three outstanding loads. The CPU can execute floating point instructions at the same time as load/store instructions. The code is written so that computations almost fully overlap with memory load/store instructions. Local performance is bounded by memory
access to L3. For the communications, a very thin and effective nearest-neighbor communication layer that interacts directly with the torus network hardware to do the data transfers was written. Given the sensitivity to latencies MPI is not an option. The two global sums per CG iteration are done via fast torus or tree routines and affect performance in a minimal way.

Table 1: Sustained performance for various local sub-lattice sizes. The values represent percentage of peak performance. (CG: conjugate gradient.).

| Local sub-lattice size | $2^4$ | $4 \times 2^4$ | $4^4$ | $8 \times 4^3$ | $8^4 \times 4^3$ | $16 \times 4^4$ |
|------------------------|-------|----------------|-------|----------------|-----------------|----------------|
| D-slash without comms  | 31.5  | 28.2           | 25.9  | 27.1           | 27.1            | 27.8           |
| D-slash with comms     | 12.6  | 15.4           | 15.6  | 19.5           | 19.7            | 20.3           |
| CG inverter            | 13.1  | 15.3           | 15.4  | 18.7           | 18.8            | 19.0           |

Figure 3: The QCD Dirac operator (D-slash) and Conjugate Gradient inverter speedup on the BG/L supercomputer as the number of CPU cores is increased up to the full LLNL machine size of 131,072 CPU cores. The highest sustained speed in this graph is 70.5 TFlops. The total lattice has size $128 \times 128 \times 256 \times 32$, while the CPU cores form a grid with size $32 \times 32 \times 64 \times 2$. Therefore, the local lattice on a CPU is of size $4 \times 4 \times 4 \times 16$.

7. Performance.
From the discussion in the previous section one can see that the performance of the code for different local sizes would depend on the typical tradeoff of memory and communication access. For a small local size the data mostly fits into the fast access L1 memory reducing the memory access cost. However, for small local size the surface to volume ratio is large and more communications per site are required increasing the communications overhead. On the other hand a large local size will have the data in the slower accessible L3 increasing the cost of memory access. But because the surface to volume ratio would be smaller the communications time would be less.
In order to measure the effect of different local sizes the performance as a % of peak was measured on a single node system with the hardware torus links connected back into the node, i.e. the output link along the plus x direction was connected to the input link along the minus x direction, etc. The hardware on each node contains a “switch” that produces this configuration for test purposes. The communication between the two CPUs was done via local memory copy. Therefore, these measurements involve all hardware and are equivalent to strong scaling. The results are in Table 1.

However, QCD is typically used as a weak scaling application. The nearest-neighbor nature of the communications as well as the existence of a fast global network in BG/L give linear speedup as the number of CPU cores is increased. It was possible to increase the number of cores up to the maximum number present in the fastest supercomputer (as of the date of this writing), the BG/L 64-rack system at LLNL (Lawrence Livermore National Laboratory). Figure 3 shows a maximum of 70.5 TFlops sustained on 131,072 CPUs. The local lattice size is 4\times4\times4\times16, resulting in a maximum global size of 128\times128\times256\times32 since the grid of CPU cores of the full machine is 32\times32\times64\times2. The sustained percent of peak in this figure is 19.3\% for the D-slash kernel and 18.7\% for the full conjugate gradient inverter, which includes the global sum reductions. This result [8] led to the 2006 Special Achievement Gordon Bell Award.

8. HotQCD.
As discussed in the previous sections, many of the ingredients for gaining some fundamental theoretical understanding of the transition to the quark gluon plasma are in place. The theory of QCD, a method for calculating quantities relevant to the transition (Lattice Gauge Theory), and the code and supercomputer resources (BG/L) are now available. To this end, a collaboration was recently formed.

![Figure 4: The chiral condensate (order parameter) versus temperature.](image)

The HotQCD collaboration [9] was initiated by LLNL/LANL to calculate the QCD equation of state with error of 10\% or less on LLNL’s BlueGene/L using lattice gauge theory. The LLNL Computing Division and NNSA have assigned approximately 10\% of the 365 Tflops machine to this calculation. Part of the collaboration’s unique expertise, tools and know-how comes from SciDAC supported research. The results of this calculation will be used as input to hydrodynamics codes that make direct contact with experimental results.
First results from this collaboration have already appeared [10]. In Figure 4 the chiral condensate that serves as an order parameter for the QCD high temperature phase transition is plotted versus temperature. The critical temperature is at the inflection point where the order parameter first becomes small as the temperature is increased.

9. PetaFlops and the QCD levels of parallelism.
The contents of this section are speculative and only reflect the author’s views, which are not necessarily original.

The near future in microelectronics appears to involve multi CPU cores per chip. The BG/L chip is already such an example with two CPU cores. Assuming that this will continue one can reasonably expect that massively parallel supercomputers in the Petaflops range will involve node chips with many CPU cores. Also, one can reasonably expect that the way the cores on a chip would exchange data (maybe via shared memory, on chip networks or other ways) will be different than the inter-chip communication network.

If this is the case, a new level of hardware parallelism arises that needs to be exploited efficiently. In order to do so one must examine the levels of parallelism inherent in the application. Lattice QCD, as described in the previous sections has a main level of parallelism. The data exchange (with the exception of infrequent global sum operations) between sub-lattices is nearest neighbor and as a result the problem can be “spread out” over a physical grid of nodes. This is a powerful property that results among other things to the linear speedup shown in Figure 3.

In Figure 3 the global lattice size is 128x128x256x32. Because of the inherent physical scales of QCD this lattice size is thought of as the point where the lattice spacing and finite volume effects would be very small and well under control. In Figure 3 this lattice is spread over 131,072 CPU cores and each CPU core “contains” a 4x4x4x16 local sub-lattice. Since the global lattice size is already very good an increase in CPU cores will likely not go into increasing the global size but rather into decreasing the local sub-lattice size in order to increase statistics and decrease the time to solution. This “push” towards the strong scaling regime will be challenging since the memory and network latency sensitivity of QCD will be amplified. This is already apparent from Table 1 for BG/L. However, one can also see from that table that a 2x2x2x2 local lattice per CPU-core does not have dramatically less performance than a 4x4x4x16 lattice although it is 64 times less in size. This implies that a machine with 64 more cores per chip than BG/L with similar performance characteristics would be able to run QCD at the 10% performance level and have a peak speed in excess of 23 PetaFlops. Therefore, one could argue that QCD is already posed to run on a machine with peak in the tens of PetaFlops.

The above scenario can be taken even further by assigning a single lattice site to each CPU-core. This approach would be even more challenging for the same reasons but would provide an additional factor of 2x2x2x2=16. Such an approach may be more effective for some multi-chip architectures than others. Again, by simply scaling the peak speed of BG/L this would correspond to a machine with 4x4x4x16 = 1,024 times the BG/L speed.

For PetaFlops scale QCD computing one would expect that some questions that require firm control of chiral symmetry would be addressed with the DWF method. As discussed in section 3 this method introduces an extra fifth dimension to the problem. Typically one expects that 8 to 32 sites of this dimension would be used. The level of parallelism of this dimension is the same as the one in the four dimensional case where data exchange between sub-lattices is nearest neighbor as well. Then, this dimension could be “spread-out” along the on chip CPU-cores perhaps in addition to “spreading-out” the time direction as in BG/L. Again, a simple scaling of the BG/L peak speed including all the
above-mentioned parallelism would correspond to a machine with peak speed in the several ExaFlops range. Obviously in this case the memory and network bandwidth and latency challenges would be high both for the hardware and for lattice QCD.

At this point it may appear that QCD has “run out” of parallelism. This is not so. The lattice Dirac operator has internal indices in addition to the four-dimensional (five-dimensional for DWF) space-time indices. The color charge index takes three values, the spin index takes four values (for certain fermion types such as DWF), while the arithmetic is complex involving real and imaginary parts. This level of parallelism is unlikely to be realized across multiple CPU-cores. The operations per index before data exchange are necessarily very few. In order to exploit this level of parallelism one may have to employ extended floating point units of appropriate architecture. This is the case for BG/L where the complex arithmetic is handled in the double multiply-add floating point unit.

10. Conclusions
In the past the path of lattice gauge theory has always been at the forefront of supercomputing. The knowledge, expertise and culture that have been developed in the lattice community are a strong driving force for the future of physics and supercomputing. And it is more so today with Petascale computing in the horizon and with the extraordinary incentive to calculate and maybe even predict the outcome of the new experiments as they unravel a yet undiscovered layer of nature’s inner workings.

Acknowledgments
I would like to thank Lawrence Livermore National Lab PAT and Computing divisions, NNSA and IBM Watson Research Center for supporting this research.

References
[1] Adcox K et al. 2005 Nucl. Phys. A 757, 184.
   Adams J et al. 2005 Nucl. Phys. A 757, 102.
   Arsene I et al. 2005 Nucl. Phys. A 757, 1.
   Back B et al. 2005 Nucl. Phys. A 757, 28.
[2] Wilson K G 1974 Phys. Rev. D10, 2445.
[3] Mawhinney R (2007) SciDAC 2007 proceedings.
   Lapage P (2007) SciDAC 2007 proceedings.
[4] Wilczek F 2000 Nucl. Phys. A 663, 3.
[5] Kogut J B and Stephanov MA 2004 “The Phases of Quantum Chromodynamics: From Confinement to Extreme Environments” Cambridge Monographs on Particle Physics.
   Monvay I and Munster G 1994 “Quantum Fields on a Lattice” Cambridge University Press.
   Creutz M 1983 “Quark, gluons and lattices” Cambridge University Press.
[6] Boyle P A et. al. 2005 IBM Journal of Research and Development Vol 49, 2/3, 351.
[7] The Blue Gene L team 2005 IBM Journal of R&D Vol 49, 2/3.
[8] Vranas P, Bhon J, Blumrich M, Chen D, Gara A, Giampapa M, Heidelberger P, Salapura V, and Sexton J C 2006 2006 Gordon Bell Award for Special Achievement, Proceedings of Supercomputing 2006, Tampa, Florida; Massively parallel QCD 2007 IBM Journal of Research and Development to appear.
[9] HotQCD collaboration: Bhattacharya T LANL, Cheng M Columbia Univ., Christ N Columbia Univ., DeTar C Univ. of Utah, Gottlieb S Indiana Univ., Gupta R LANL, Heller U BNL, Huebner K BNL, Jung C BNL, Karsch F BNL & Bielefeld Univ., Laermann E. Bielefeld Univ., Levkova L Univ. of Utah, Mawhinney R Columbia Univ., Petreczky P BNL, Renfrew D Columbia Univ., Schmidt C. BNL, Soltz R LLNL, Soelder W BNL, Sugar R UCSD, Toussaint D Univ. of Arizona, Vranas P LLNL.
[10] Soltz R and Gupta R 2006 proceedings of Quark Mater 2006.
[11] Vranas P 2001 Nucl. Phys. Proc. Suppl. 94, 177.