LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-PROC-641527

The Origin of Mass

P. Boyle, M. Buchoff, N. Christ, T. Izubuchi, C. Jung, T. Luu, R. Mawhinney, C. Schroeder, R. Soltz, P. Vranas, J. Wasem

July 29, 2013

Supercomputing 2013
Denver, CO, United States
November 17, 2013 through November 22, 2013

Disclaimer

The Origin of Mass

Peter Boyle
U. of Edinburgh, UK
paboyle@ph.ed.ac.uk

Michael I. Buchoff
LLNL, CA, USA
buchoff1@llnl.gov

Norman Christ
Columbia U., NY, USA
nhc@phys.columbia.edu

Taku Izubuchi
BNL, NY, USA
izubuchi@bnl.gov

Chulwoo Jung
BNL, NY, USA
chulwoo@physics.columbia.edu

Thomas C. Luu
LLNL, CA, USA
luu5@llnl.gov

Robert Mawhinney
Columbia U., NY, USA
rdm@physics.columbia.edu

Chris Schroeder
LLNL, CA, USA
schroeder10@llnl.gov

Ron Soltz
LLNL, CA, USA
soltz@llnl.gov

Pavlos Vranas
LLNL, CA, USA
vranas2@llnl.gov

Joseph Wasem
LLNL, CA, USA
wasem2@llnl.gov

ABSTRACT

The origin of mass is one of the deepest mysteries in science. Neutrons and protons, which account for almost all visible mass in the Universe, emerged from a primordial plasma through a cataclysmic phase transition microseconds after the Big Bang. However, most mass in the Universe is invisible. The existence of dark matter, which interacts with our world so weakly that it is essentially undetectable, has been established from its galactic-scale gravitational effects. Here we describe results from the first truly physical calculations of the cosmic phase transition and a groundbreaking first-principles investigation into composite dark matter, studies impossible with previous state-of-the-art methods and resources. By developing a powerful new algorithm, "DSDR," and implementing it effectively for contemporary supercomputers, we attain excellent strong scaling, perfect weak scaling to the two million cores of the LLNL BlueGene/Q, sustained speed of 7.2 petaflops, and time-to-solution speedup over the previous state of the art of 200.

Categories and Subject Descriptors

I.6.8 [**Simulation and Modeling**]: Types of Simulation—*Monte Carlo, Parallel*; I.6.3 [**Simulation and Modeling**]: Applications; J.2 [**Physical Sciences and Engineering**]: Physics; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems—*Computations in finite fields, Computations on matrices*; D.1.1 [**Programming Techniques**]: Applicative (Functional) Programming; D.1.3 [**Programming Techniques**]: Concurrent Programming—*Distributed programming, Parallel programming*

General Terms

Gordon Bell Prize categories: Scalability and Time to solution.

Keywords

SC13 proceedings.

SC '13, November 17 - 21 2013, Denver, CO, USA

http://dx.doi.org/10.1145/2503210.2504561

1. OVERVIEW

Mass is the cornerstone of our existence, so familiar that we take it for granted, but the existence of mass is one of deepest mysteries of science. Almost all the visible mass in the Universe today was created about ten microseconds after the Big Bang. This is the mass that resides primarily in the neutrons and protons that make up the nuclei of all the atoms in the Universe. Neutrons and protons are composite particles, containing more elementary constituents, quarks and gluons, within. Gluons are the carrier-particles of the strong nuclear interaction, which holds quarks together in bound states. In fact, the interaction is so strong that under ordinary conditions, quarks cannot escape, a property known as confinement. However, in the very early and hot Universe, quarks and gluons coexisted in a deconfined state. Only once the Universe expanded and cooled to below 2 trillion degrees Celsius, and underwent a phase transition, did the quark-gluon plasma transform into the stable packets of quarks and gluons that are the protons and neutrons we know today. It is from this transition that almost all the visible mass in our Universe emerged, and understanding it is of paramount importance. The physics of this phenomenon is staggering if you consider that 99% of the proton and neutron masses is energy stored in the field of the massless gluons, with only 1% contributed by the massive (but light) constituent quarks.

The quark-gluon plasma was first produced in the laboratory at the Relativistic Heavy Ion Collider in 2004 [1], followed by the Large Hadron Collider in 2010 [2]. Also in 2004, the architects of the theory that captures the physics of quarks and gluons, Quantum Chromodynamics (QCD), which decades of experiment have verified to extraordinary precision, were awarded the Nobel Prize [3, 4, 5, 6, 7]. However, the transition has yet to be studied in a way that includes all the relevant physics because to span the vast range of scales starting from the quarks and gluons, including fluctuations in the plasma, and ending with macroscopic, interacting nuclear matter is extremely challenging. Many questions remain to be answered with controlled systematics. Until this is done, our understanding of the transition that gave rise to the Universe as we know it will be incomplete.

Bridging the immense range of scales relevant to the QCD transition can only be done by numerical simulation. The most powerful approach is Lattice Gauge Theory (LGT), invented in 1974 by Wilson [8, 9]. One discretizes space-time onto a grid called the lattice, associates quarks with the sites of this lattice and gluon fields with its links. One then performs an approximate Feynman path integral over all possible states of the system using molecular dynamics-driven Markov chain Monte Carlo. While simulating gluons turns out to be straightforward and efficient, quarks are complicated and require solving millions of times a sparse, $O(10^{10})$-dimension matrix-vector equation, involving what is known as the fermion matrix, which evolves with the system. These solves account for over 99% of the operations. Furthermore, vital properties of the quarks are grossly distorted by the lattice discretization and are recovered only as the lattice becomes asymptotically fine (the continuum limit) [10]. Several improved discretization methods have been developed to lessen these distortions, two of the most popular being "staggered fermions" and "domain wall fermions" (DWF); but as we show here, these can not faithfully reproduce the physics of the transition unless prohibitively large and currently non-existent computing resources are used.

Quarks and gluons combine to form a plethora of other particles in addition to protons and neutrons. These are unstable and exist for only tiny fractions of a second, but still play a crucial role in the phase transition. The lightest particles, known as pions, play a particularly important role since their small masses allow them to communicate information over long distances throughout the plasma. There are three pions with approximately equal masses each about one-seventh the mass of the proton ($M_{\text{pion}} \approx 140\ \text{MeV}/c^2$, $M_{\text{proton}} \approx 940\,\text{MeV}/c^2$, $1\,\text{MeV}/c^2 \approx 2{\times}10^{-30}$ kg). Ordinary discretizations break the symmetries responsible for these small masses to an unacceptable degree unless extremely small lattice spacings are employed, which is prohibitively expensive. Even with the somewhat improved staggered fermion approach, simulations feasible today fall short by at least an order of magnitude, in terms of computational cost, of the lattice spacings required to control these effects. Alternatively, one can use the more advanced DWF approach, which achieves dramatic symmetry restoration by restructuring the fields using an auxiliary fifth dimension. Still, physical pion masses require very large fifth dimension size and very small quark masses, and since the condition number of the fermion matrix scales with the quark masses, the cost of the many inversions becomes exorbitant. As a result, simulating the transition with the correct matter content has remained an outstanding challenge in nuclear and particle physics for the past 40 years; but no more.

Here, for the first time ever, we have successfully simulated the QCD phase transition with exactly three pions all with the correct mass. We have reached this monumental landmark by inventing and implementing a new algorithm known as the Dislocation Suppressing Determinant Ratio Domain Wall Fermion method (DSDR DWF), which has reduced the time to solution for our problem by more than a factor 10. In addition, we optimized our code to achieve an efficiency of 30%, which is on par with the former state of the art, so that we benefit greatly from advances in hardware, a factor of roughly 20 for the Blue Gene/Q over the Blue Gene/P. We attain excellent strong scaling and perfect weak scaling for up to two million cores on the LLNL Sequoia Blue Gene/Q. Moreover, much of our optimization applies equally well to other modern supercomputers with large numbers of on-node cores. Altogether, we have reduced the time to solution for our calculations by a factor of over 200, enabling this unprecedented numerical simulation.

But the QCD transition only accounts for the *visible* mass, which has been found to comprise only 17% of the total mass of the Universe. The other 83% is hidden from us. Light does not bounce off of it and visible matter passes through it with only the feeblest of interactions. It has been termed dark matter and its existence is another great puzzle of modern science. Though we have not been able to "see" dark matter yet, the effects of its gravitational pull on visible matter on galactic scales are extensive and amazing [11, 12, 13, 14, 15]. There are few clues to how much of it should exist, but the fact that its cosmological density is only a few times that of visible matter suggests a relation between the origins of the two. A tantalizing possibility is that, like neutrons, dark matter particles are composite, consisting of new elementary particles tightly bound by a new strong interaction of Nature. Like QCD, this Strong Force Dynamics (SFD) [16, 17] can be studied in detail only by numerical simulation, i.e. LGT. We present below the very first results in this new field of research [18].

Our work is a synergy of algorithms, implementation, and effective utilization of the latest generation of supercomputers to provide answers to one of the most important scientific problems of our time, the origin of mass. Our results are of importance to all practitioners of particle physics, possibly to condensed matter physicists, to supercomputing application developers, and to architects of the next generation of supercomputers. In the sections that follow, we describe all these components, and give the performance characteristics and the physics results of our simulations, which were all performed on the Sequoia+Vulcan 25 Petaflops Blue Gene/Q supercomputer at the Lawrence Livermore National Laboratory.

2. PREVIOUS STATE OF THE ART AND OUR NEW RESULTS

As discussed in section 1, our work represents a very significant improvement over the previous state of the art in the effort to understand the origins of mass both visible and dark. The previous state of the art in lattice numerical simulations of QCD comes from work using staggered fermions [19] and domain wall fermions without DSDR [20].

The main handicap of these methods is the inability to simulate with three pions with physical masses; instead, calculations are carried out with several times too many pions or significantly larger pion masses. Since the physics involves the transition from one phase (quark-gluon plasma) to another (protons and neutrons), collective phenomena of the degrees of freedom over large distances are dominant, and proper treatment of the lightest particles, the pions, is essential. Small deviations from the physical pion mass could distort the physics in drastic ways and are a well known source of systematic uncertainty. Both the staggered and (non-DSDR) DWF methods suffer from this problem.

For staggered fermions, only one pion assumes its physical mass, the other two are heavier by about 50%. Furthermore, this method unavoidably introduces twelve additional, unphysical pions. These are lattice artifacts and disappear in the continuum limit (finer lattice spacing), but still require the use of the questionable "rooting" method to reduce the effective number of pions. To put this into perspective, the computational cost of reducing the lattice spacing by a factor of 2 in all four dimensions would require $2^{11} \approx 2,000$ times more computing.

In contrast, there are exactly three pions with DWF. Even at finite lattice spacing, domain wall fermions preserve one of the most important symmetries of the theory (chiral symmetry) and introduce no unphysical pions. However, this comes at a price, since the computational cost increases linearly with the 5-D volume, i.e., linearly in the extent of the fifth dimension, L_s. Moreover, since DWF do possess a discretization effect which increases the pion masses for finite L_s, and former state-of-the-art calculations with DWF were limited by cost to $L_s \leq 32$, a lower bound on the attainable pion masses approximately 40% larger than their physical values was imposed. It has been estimated [21] that reaching the true pion masses with DWF would require $L_s \approx 320$, requiring an order of magnitude more computation than was previously feasible.

As described in the following sections 4 and 6, we were able to overcome the limitations of standard DWF using a new algorithm and an implementation that gives us outstanding performance using contemporary supercomputers, in particular, the Blue Gene/Q. The results are shown in Figure 1. For the new results (blue diamonds) all three pion masses are set to the physical value of 140 MeV/c^2. The best previous calculation [20] (red circles) was done with unphysically heavy, 200 MeV/c^2 pions. The difference between the two is striking. It is clear that the former state-of-the-art calculation underestimated the quark field fluctuations, given by the height of the peak, by a factor of two and overestimated the transition temperature, given by the location of the peak, by several percent. Both effects were anticipated for DWF, but their sizes were unknown. This is a unique result and a very significant improvement over the previous state of the art.

Our research in composite dark matter numerical simulations [18] is the first of its kind, with no previous standard for comparison. The results for our groundbreaking computations of cross sections (roughly, probabilities that the XENON100 experiment would detect dark matter) for two models are shown in Figure 2 along with the bound measured by the XENON100 experiment [22]. Each solid line represents the cross section for the electromagnetic interaction between the dark matter and the detector, for a particular model, for a range of values of the composite dark matter particle mass (M_χ), and for the two primary components of the interaction (the magnetic moment and charge radius). Each dashed line is the same but with the magnetic moment set to zero. The bottom line is that much of parameter space for these models (and closely related ones) is ruled out by experiment. To be viable, either M_χ must be greater than 10 TeV/c^2 (or 10^7 MeV/c^2) or the magnetic moment must be zero and M_χ must be larger than 1 TeV/c^2 (i.e., everything above the dotted line is ruled out). For comparison, the mass of a neutron again is 1 MeV/c^2, $1 - 10,000$ times smaller. The two models studied here are like QCD in that there are three "quark" colors, but different in that one has

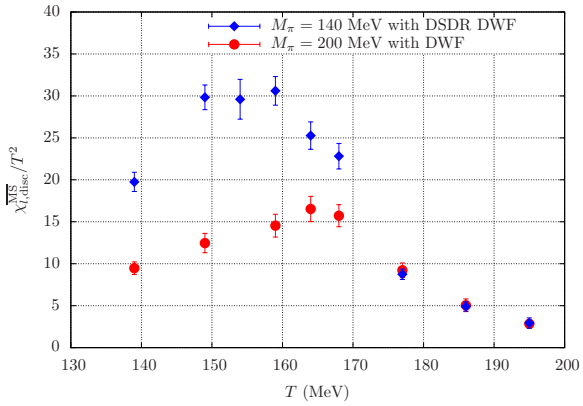

Figure 1: The chiral susceptibility as a function of temperature. The peak signifies the transition temperature. The blue diamonds represent our result, with three pions each with a physical mass of 140 MeV. The red circles are results for the best previous calculation to date, with unphysical, 200 MeV pions [20].

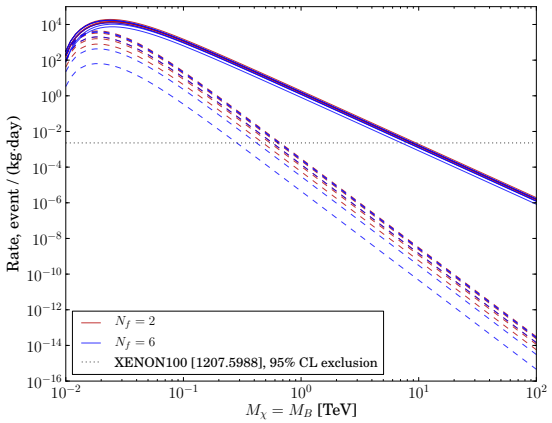

Figure 2: Event rates for composite dark matter models calculated in our work [18] compared to model-independent results from the XENON100 experiment [22]: theoretical rates including all terms (solid lines) and including only the charge radius contribution (dashed lines) for comparison; the upper bound on the event rate is the dotted line. See the text for discussion.

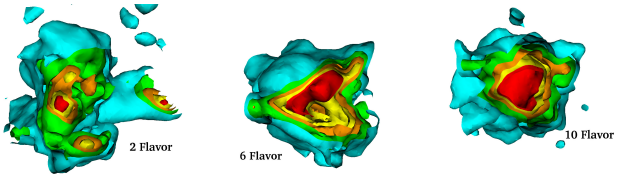

Figure 3: Cutaway of a 3-D contour plot of the magnitude of the neutral composite dark matter candidate electric charge density with 3 colors and 2, 6, and 10 species (flavors). Low-density regions are colored blue while regions of increasing density are colored green, orange, yellow, and finally red.

two "quark" flavors N_f (or species, e.g., up and down) and the other has six. While the cross section is evidently largely insensitive to the number of flavors, we have uncovered significant flavor dependence in the dark matter particle's electric charge density, which encodes how the electric charges of the constituents are arranged inside of the composite object. Results for the two- and six-flavor models prompted us to simulate the ten-flavor model as well, all illustrated in Figure 3. These are landmark results in numerical simulations of composite dark matter and are currently the only ones of their kind.

3. PERFORMANCE MEASUREMENT METHODS AND THE LATTICE

We measured performance of our lattice simulations in a straightforward way using cycle-accurate time-base measurement in conjunction with manual flop counting since we know exactly how many flops are done by our kernel as described in equation 1 below. Furthermore, we checked against instruction records in the IBM virtual simulator and FPGA emulator logs. We also used the Blue Gene Performance Monitoring (BGPM) API. Our performance results and discussion are given in section 6. Our physics application and the computations involved are described in some detail below.

Most quantitative studies of QCD and QCD-like models require numerical methods, the most powerful of which is Lattice Gauge Theory, introduced above. Its inception by Wilson in 1974 [8] in the mid 1970's has been followed by a steady stream of algorithmic advances [23, 24, 25], leading to the methods and algorithms that are practiced at present [26].

The codes associated with LGT are large (hundreds of thousands of lines). Some of the authors of this paper are lead developers of the Columbia Physics System (CPS) [27] used in this work. CPS is written in C++ and equipped with MPI for inter-node communications (from its beginning) and OpenMP for shared memory (as of this work). It is highly portable, can run on any system that supports C++, MPI and OpenMP, and has been used for production on many platforms ranging from Blue Gene systems to Linux clusters. The core algorithm is Markov chain Monte Carlo (MCMC) with Molecular Dynamics (MD) evolution of the matter and interaction fields based on an energy functional derived from the theory under investigation. A typical simulation begins with some initial field configuration (i.e., set of initial values for every degree of freedom) and evolves these fields stochastically to one new configuration after another, guided by the energy functional. After a set of configurations is generated, it is used to compute path integrals which yield values of physical observables that can be compared to experimental results. For example, one can compute the correlation function of the composite neutron field and extract from it the neutron mass.

About 99% of the code carries out less than 1% of the operations and so can be written at a high level of abstraction without tuning for performance. The remaining 99% of the operations are performed in a relatively short routine, referred to as the kernel, which solves the fermion matrix-vector equation required to compute the quark contributions to the energy functional and forces. The matrix, also known as the "Dslash" operator (or just the "Dslash"), connects different fermionic states of the system and is therefore enormous. Its dimension is $O(10^{10})$, a product of the 5-D lattice volume ($O(100^4 \times 10)$ for modern calculations), the space-time dimension (4), and the number of quark colors (3 for this work).

To solve this system, we use the conjugate gradient method, which ordinarily converges in $O(1,000)$ to $O(10,000)$ iterations, each consisting mainly of one Dslash operation. This, along with fact that we must solve the system thousands of times to generate each new configuration, is why the Dslash dominates the computation. Thus, highly optimizing it is essential. It is remarkable that the dimension of the Blue Gene/Q torus matches that of our 5-D DWF system exactly, making short work of network mapping. Our lattices are typically many times larger than the Blue Gene/Q torus grid, and we map a small sub-lattice to each node (for example, $8^4 \times 16$). In the Dslash, only nearest-neighbor information exchange is necessary, so for sufficiently large sub-lattice volumes, all communication costs can be hidden by overlapping with computation. Note that while we have tuned the Dslash for the Blue Gene/Q, it is straightforward to achieve similar efficiency on basically any modern massively parallel supercomputer.

The Dslash is given by the following equation (for zero mass):

$$D(x,x';s,s') = \delta^x_{x'}\left[\delta^s_{s'}(m_0-5) + \delta^{s+1}_{s'}P^-_5 + \delta^{s-1}_{s'}P^+_5\right] + \delta^s_{s'}\sum_{\mu=1}^4\left[\delta^{x+\mu}_{x'}U_\mu(x)P^+_\mu + \delta^{x-\mu}_{x'}U^\dagger_\mu(x-\mu)P^-_\mu\right] \quad (1)$$

where m_0 is the domain wall height, δ^i_j is the Kronecker delta (1 if $i=j$, 0 if not), and $P^\pm_i \equiv (1\pm\gamma_i)/2$ are matrices which govern how the spins of different particles interact. Indices running over the lattice space-time (x and x') and the fifth dimension (s and s') are shown; spin and color indices have been suppressed for presentation purposes but are carried by the complex 4×4 spin matrices (P) and the complex 3×3 color matrices (U). The index μ indicates different directions in space-time. In practice, each space-time index is split into one index running over on-node sub-lattice points V_s and another running over the sub-lattices V_n (the number of the nodes in each dimension), such that the total volume $V = V_s \times V_n$.

The large V_n computation is done in a typical SIMD way by spreading the values of the fields across the nodes of the machine, performing the nearest-neighbor communica-

tion as needed. The other sums contain fewer terms and are not distributed across the torus grid. Instead, their data dependencies are exploited to efficiently spread the operations over the 16 on-node cores. A brief description of the operations involved is:

1. Load the gauge interaction (i.e., gluon) fields U (typically 72 doubles per site).

2. Load the matter (i.e., quark) fields (24 to 96 doubles per site).

3. Initiate the communication of the matter fields needed for the next iteration.

4. Compute the terms of the above equation.

As can be seen from this ordering, the memory access per operation as well as the communication per operation is about one double per flop and severely exposes the application to hardware and software latencies. In a sense, our application "lives" in the strong scaling regime. This makes coding it efficiently very challenging.

4. INNOVATIONS

Our innovative new algorithm reduces the total time to solution by a factor of about 10. In addition, we have implemented the algorithm, and optimized pre-existing code, in a manner that greatly benefits from the new supercomputer architectures by maintaining $\approx$ 30% of peak (efficiency on par with former state-of-the-art implementations of previous-generation methods) while it exhibits perfect weak scaling to 2 million cores and excellent strong scaling. In addition, we note that our time to solution, approximately 180 Blue Gene/Q rack-days, would scale perfectly with resource size and availability (i.e., we could solve the same problems using 120 racks for 1.5 days or 1 rack for 180 days). We derive the full benefit of hardware advances, a factor of about 20 compared to previous state-of-the-art machines (of approximately one petaflops size), such as Blue Gene/P. Combining the new algorithm, effective implementation, and hardware evolution, we have reduced the overall time to solution for this work by factor of approximately 200 compared to the previous state of the art. We describe our algorithmic innovations and kernel implementation advances in the the following sections.

4.1 The DSDR algorithm

The Dislocation Suppressing Determinant Ratio method (DSDR) [28, 21, 29] was invented to solve the notorious problem described in section 1, that simulating at the physical value of the pion mass would require enormous amounts of computational resources (approximately 100 Blue Gene/P rack-years). We have employed DSDR here in order to perform for the first time ever thermodynamics calculations with physical pions. The DWF method [30] was first employed in numerical simulations in [31]. The DWF method defines an auxiliary fifth dimension of size L_s. As L_s is increased, the unphysical violation of chiral symmetry is suppressed, and the quark and pion masses can be made smaller with a computational cost linear to L_s. The problem is that very large L_s values (a few hundred) are needed to reach the physical pion mass.

The reason for this is traced to the eigenvalue spectrum of the underlying transfer matrix along this new fifth dimension. Roughly, the determinant of the five-dimensional fermion matrix D is the exponential of the four-dimensional transfer matrix D_T. If we denote the lowest eigenvalue of the transfer matrix by λ_T, then the lowest eigenvalue of the 5-D fermion matrix, λ_5, is related to λ_T by $\lambda_5 \sim \exp -L_s * \lambda_T$. It is this 5-D eigenvalue that sets the value of the pion mass by reducing the residual quark mass m_{res}. Provided that $\lambda_T > 0$, one can make λ_5, and therefore m_{res}, small by increasing L_s. There is an interesting interplay in that the lowest eigenvalue of the 4-D matrix must be large in order for the lowest eigenvalue of the 5-D matrix to be small.

The value of λ_T depends on the dynamics of the theory and it is very nearly zero for the lattice spacings that are feasible today. For a given lattice spacing, one has no control over λ_T and can only increase L_s in order to compensate which is an expensive proposition; however, λ_T does increase as the lattice becomes finer (another prohibitively expensive proposition as described in section 1). This suggests that the small value of λ_T is not physical but rather a lattice artifact and that one might be able to increase λ_T, and thereby reduce λ_5, through improved discretization.

This artifact is related to lattice dislocations; gluonic fields on the lattice are able to transform "between the cracks" (i.e. due to the gaps between sites) in ways that are not possible in the continuum. This unphysical behavior is reflected in the 4-D transfer matrix, D_T. If one multiplies the determinant of the physical 5-D matrix with the determinant of the unphysical 4-D matrix, $\det(D_T)$, it is possible to steer the molecular dynamics away from field configurations with small $\det(D_T)$, which are precisely those with small λ_T, without affecting physical properties of the calculation. This indeed has the effect of reducing m_{res} for a given L_s and allows one to achieve a small pion mass for moderate values of L_s.

In other words (more familiar in condensed matter physics), domain wall fermions are surface states, living in five dimensions but localized to a four-dimensional boundary, which corresponds to real space-time. This localization is essential to describe the 4-D physics, but a certain discretization artifact enables DWF to delocalize, spreading out into the auxiliary fifth dimension and giving rise to the symmetry breaking which prohibits the simulation of light composite states. DSDR eliminates this artifact and strongly suppresses delocalization, by imposing a penalty precisely on the artifact in the energy functional which drives the molecular dynamics evolution. The phenomenology of surface states is well known in condensed matter physics; for the connection, see [32] and references therein. In that case, the states typically live in 3-D and are localized on a 2-D surface. Our DSDR algorithm may be of direct interest to any practitioner of condensed matter physics studying surface state localization problems.

The reduction of m_{res} is demonstrated in Figure 4 from [21]. For $L_s = 32$, $m_{res} \approx 2 \times 10^{-4}$ for DWF with DSDR (blue diamonds). To achieve a similar value for m_{res} using DWF without DSDR (red circles) requires $L_s \approx 330$ (extrapolating the data shown using the simple form $m_{res} = c/L_s$, which fits the data well for $L_s > 16$, giving $\chi^2/dof \approx 0.5$ and $c \approx 0.064$), which is more than a factor of 10 larger and would come at a computational cost more than 10 times greater. This discussion is conservative since, as can be seen

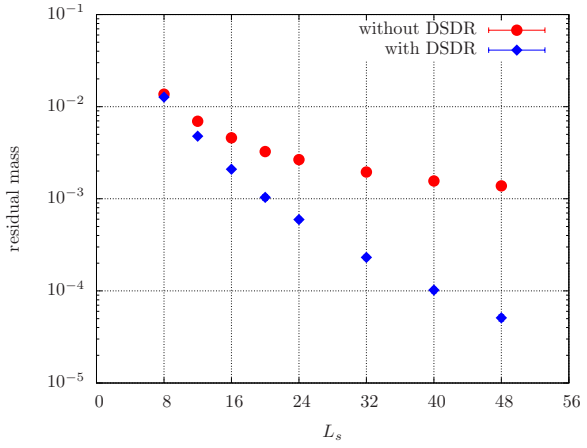

Figure 4: m_{res} vs. L_s for DWF without DSDR (red circles) and DWF with DSDR (blue diamonds) [21]. One can see that DSDR produces significantly smaller m_{res} for the same L_s. The technical details are: quenched backgrounds at $a^{-1} \approx 1.4GeV$ with m_0 = 1.9 and $m_f = 0.02$.

in the figure, for $L_s > 16$, m_{res} continues to decrease exponentially for DWF with DSDR but falls only linearly for standard DWF. In fact, to reach the physical pion mass at a temperature of 140 MeV (the coarsest lattice simulated) using DSDR DWF requires $L_s = 48$ only, while reaching this low mass using standard DWF would require $L_s \approx 1,000$ - over 20 times larger. The physics results in Figure 1 were obtained using $L_s = 48$ at the lowest temperatures and $L_s = 32$ at the rest. **By developing and deploying DSDR, we have reduced our time to solution more than 10-fold**.

For technical reasons, $\det(D_T)$ must be divided by a similar determinant to allow for a small controlled number of dislocations to occur. The reason for this is beyond the scope of this paper and is discussed extensively in [28, 21, 29]. Suffice to say, it does not affect the above discussion in a significant way. Its effects are offset by using the Möbius algorithm [33, 34], which performs well in tandem with DSDR, reducing our time to solution somewhat further.

4.2 Kernel implementation innovations on contemporary supercomputers

As discussed in section 3, we have designed the application kernel with great care by employing several innovations in order to take advantage of the generic features of modern massively parallel supercomputers. These innovations are presented below for the particular example of the Blue Gene/Q architecture as described in section 5; however, the features we leveraged are quite generic in today's supercomputers, and our methods are applicable to other platforms as well. For example, hardware features such as a large number of cores per chip, inter-node communication, memory sharing, and wide FPUs are relatively common. Our kernel can be ported to other platforms with only moderate effort.

QPX: In order to achieve the SIMD parallelism required to feed the 4-multiply-add-per-cycle QPX unit, the data must be laid out carefully. In particular, it is optimal to store data that will participate in one QPX instruction sequentially in an L1 cache line, so that the data can be loaded directly into a quad-double QPX register. The numerically intensive part of our calculations involves complex arithmetic, so a natural layout is one with two complex numbers z_1 and z_2 adjacent in memory with alternating real (r) and imaginary (i) parts [r_1,i_1,r_2,i_2]. Our application can achieve SIMD parallelism through other data layouts as well, but we have found that effects of register pressure make this the best choice. Using P.B.'s BAGEL assembly generator, we have produced assembly code for the Dslash operator to ensure that every flop is 4-way parallel and every opportunity to use the fused multiply-add is exercised, yielding a speedup of the Dslash of nearly 3.2.

L2 cache: The 16 L1 caches have an aggregate peak bandwidth to L2 that is about 10 times greater than the bandwidth from L2 to external memory (410 vs. 42 GB/s). The L2 cache size is 32 MB, which is large enough for our application provided some care is taken to reuse data. As can be seen from equation 1, the term that involves the computationally intensive operations with the gauge field U is diagonal in the 5th dimension (s), which means L_s vectors are multiplied by each U matrix; thus, high reuse of the U matrices is possible, which is key since the Dslash is otherwise bandwidth-limited. Since a U matrix is 72 bytes and a fermion vector is 48 bytes (in single precision), using each U matrix 32 (or 48) times per load, as we do, reduces the bandwidth requirement by over 60%.

Cache interference: While caches serve a very basic and important role, our application would benefit from full control of some of the L1 cache resource. Specifically, accumulating results from equation 1 in L1 creates back-pressure from L2 because of large write-through traffic and degrades performance. It would be beneficial if the user could assign a small part of L1 to behave as scratch memory, but this feature is not available, and so we must accumulate our results in-register to maximize performance. The drawback of this strategy is that it significantly reduces register availability; however, we largely mitigate this drawback through effective use of the L1 prefetcher.

L1 prefetcher: The L1 prefetcher hardware unit is located on the chip between the L1 and the L2 switch interface. Because the memory access pattern of our application is fully deterministic and repeats itself in regular intervals, we are able to program the L1p to hide almost completely the load latencies which would otherwise result from the small bursts of data loading that stem from our register management strategy (described above).

L1 locking: The PowerPC A2 core allows the user to lock L1 lines for reuse. As described above, the U fields can be reused many times because of our data ordering strategy. In order to maximize this reuse, we lock the values into L1 in order to avoid undesired eviction and consequent reloading. The hardware locking mechanism is made possible because of an L1p innovation that hides the "locking" from L2, eliminating L2 overlocking and L1-L2 locking message exchanges.

Threads: Each core is capable of hosting four hardware threads for a possible total of 64 hardware threads among the 16 on-chip cores. Our code employs thread-level parallelism with OpenMP thread creation and Blue Gene/Q SPI barrier synchronization, running optimally with 1 task and 64 threads per node. The many threads of Blue Gene/Q allow the processor pipelines to operate at high utilization. For example, it is possible for one thread to execute a floating-point operation while another thread performs a load oper-

ation during the same cycle.

SPI communications: The bulk of our code has been written with MPI communications as implemented in the QMP SciDAC interface library. For the more critical communications, a layer was written at the SPI level that allows for communicating relatively short data streams with low latency. This SPI makes use of the injection/reception torus network-to-memory DMA engines to overlap calculations with communications. Because the number of compute and communication cycles is very similar, this overlap provides a speedup of nearly 100%. We also use SPI global barriers and reductions, which we determined to be necessary, in spite of the relatively large number of cycles between collective communications in our application, because we found standard MPI collectives to be very costly initially on the Blue Gene/Q.

Loop ordering: Different loop orderings are possible in our application allowing for different strategies. We found the following to be the most effective loop ordering to calculate equation 1 (Q denotes the input vector to which the Dslash matrix is applied, and Q' the result):

for x in V **do**
  Lock the gluon matrices $U(x, \mu = 1, ..., 4)$ into L1
  for $s = 1$ to L_s **do**
    for $\mu = 1$ to 4 **do**
      Load nearest-neighbor fermion vectors Q to registers
      Multiply Q by the (constant) spin matrices $P_\mu^\pm$
      Load locked U from L1 into registers
      Perform the $Q' = U \times Q$ matrix vector mult.
      Sum the results in register
    end for
  end for
end for

5. THE LLNL BLUE GENE/Q SEQUOIA AND VULCAN

We performed our simulations and performance measurements on LLNL's Sequoia and Vulcan Blue Gene/Q supercomputer system. Sequoia is a 96-rack system and Vulcan is a 24-rack system. During the early science period, the racks of the two machines were wired as a single machine consisting of 120 racks making the combined system the fastest supercomputer in the world. The peak speed of one rack is 209.7 teraflops for a total system peak speed of 25.2 petaflops. Each rack consists of 1024 compute nodes and therefore the full system has 122,880 nodes or 1,966,080 cores since there are 16 cores per node. All nodes are interconnected with a low-latency, high-bandwidth custom network that connects neighboring nodes in a 5-dimensional grid.

The node (shown in Figure 5) consists of a card that contains the IBM Blue Gene/Q ASIC, external DRAM memory, and all necessary on-node connections. The ASIC is a sophisticated IBM chip with a wealth of features. There are 16 user accessible CPU cores, each with 4 hardware threads, and its own L1 cache memory. They are connected through a high performance crossbar to a large shared L2 memory that is followed in the hierarchy by external DRAM memory. Memory is coherent at the L1 level. A sophisticated L1 prefetcher connects each L1 with the L2. The prefetcher is capable not only of streamline prefetching but is also programmable, and can prefetch according to learned memory access patterns.

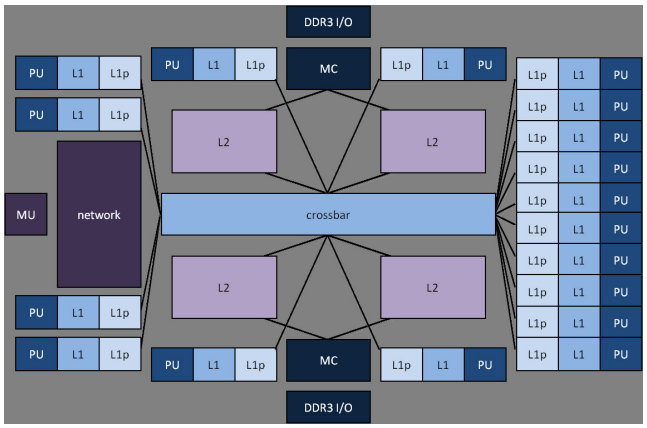

Figure 5: Diagrammatic representation of a single Blue Gene/Q node [35].

Each core can execute 4 multiply-add instructions per cycle using custom hardware called QPX. This SIMD unit is implemented at the highest chip frequency and it includes 4 sets of 32 double precision registers. A QPX hardware instruction set is implemented in order to load/store data directly from an L1 line into the 4 sets of registers and perform numerical operations. This unit defines the peak performance of Blue Gene/Q as 8 floating point operations per cycle per core at the highest frequency domain.

The interconnect is a five-dimensional nearest-neighbor torus. The fifth dimension has only 2 nodes. Each node has specialized router hardware that implements cut-through routing that allows any node to communicate with any other without CPU intervention. Furthermore, the on-node network hardware has a Direct Memory Access engine that directly loads data from the memory to the network and offloads network data directly into memory, also without CPU intervention. This is crucial for our application since it allows for overlap of communication and computation.

6. PERFORMANCE RESULTS

The performance results presented in this section were obtained on the LLNL Sequoia+Vulcan Blue Gene/Q system as described in section 5. All results shown are for the conjugate gradient solver of our kernel as described in sections 3 and 4.2. As explained there, this accounts for about 99% of all operations and, therefore, the performance plots presented here basically express the full application performance. As can be seen from Figure 6, the weak scaling of our application is nearly perfect to 120 racks (1,966,080 cores) sustaining 3.7 Gflops/core indicated by the solid black line. **At the largest machine size, our application achieves a sustained speed of 7.2 petaflops**.

We present the strong scaling behavior in Figure 7. Machine sizes ranging from four racks (65K cores) to 64 racks (1M cores) were used for a fixed problem size of $64^2 \times 128^2$ lattice sites. Again, the solid black line represents performance of 3.7 Gflops/core. As can be seen, the strong scaling is nearly linear up to 16 racks (256K cores); the increase is sub-linear for larger machine sizes, but the deviations from linear are small. At the largest machine size, the performance is only 30% below linear. This reflects typical effects

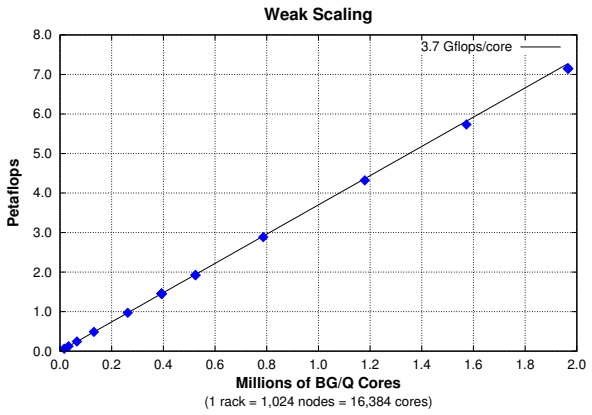

Figure 6: Weak scaling with 8^4 lattice sites per node on up to 120 racks (1,966,080 cores) of LLNL Sequoia+Vulcan Blue Gene/Q. The speedup increases linearly with a rate of 3.7 Gflops/core (solid black line).

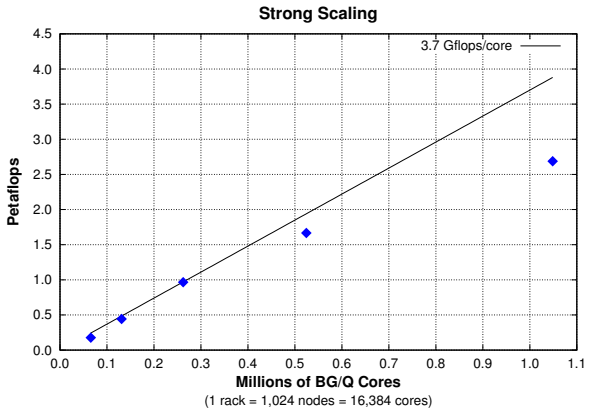

Figure 7: Strong scaling for a fixed size problem ($64^2 \times 128^2$ lattice) in different partition sizes ranging from 4 racks (65K cores) to 64 racks (1M cores). The solid black line represent linear increase with rate 3.7 Gflops/core.

of communication latencies and bandwidths. As the number of nodes increases, the local problem shrinks, the surface-to-volume ratio grows, and more data must be communicated per flop, exposing the performance to network limitations inherent in the hardware.

As discussed, the highest sustained speed obtained by our application is 7.2 petaflops, or 58 Gflops/node. One Blue Gene/Q node has a peak speed of 210 Gflops/node, attainable only by using 4 multiply-add instructions every clock cycle in the fastest clock domain. One multiply-add corresponds to 2 floating point operations; however, the corresponding multiply-add hardware is fused and peak can be achieved only if the application has a perfect pairing of multiply and add operations. In our application, there are a substantial number of adds which cannot be paired with multiplies coming from multiplication by the γ matrices in equation 1. These 4×4 complex matrices have entries that

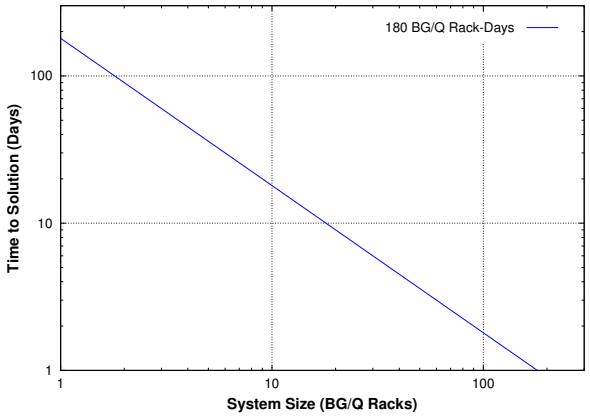

Figure 8: Graphical depiction of the time to solution for the research conducted vs. machine size. The computational cost of this work was 180 Blue Gene/Q rack-days, and does not depend on system size in the range shown.

are known constants throughout the simulation and only amount to multiplication by ± 1. There is no need to use the hardware to perform these multiplications and so we do not; as a result, the maximum performance possible for our application is reduced to 174 Gflops/node, 83% of the peak performance for the hardware. This means that our sustained performance of 58 Gflops/node, which corresponds to 28% of the hardware peak, is 33% of our application's theoretical peak performance. We note that if we would use a simple hardware counter (measuring only the number of times the floating point unit is accessed), rather than determining the flop count by hand, this multiply-add imbalance would be hidden, and our performance would appear to be more than 15% higher than it is (33% of the hardware peak).

As described above, through algorithmic and machine advances, we reduced the time to solution for the calculations that we carried out by a factor of over 200 compared to the previous state-of-the-art. Given (1) the strong scaling results of Figure 7, (2) that the total computation it took to produce the physics result of Figure 1 (blue diamonds) was measured to be 180 rack-days using a wall-clock and (3) that our physics problem involves 18 data-independent runs, we are capable of utilizing a large range of system sizes with basically uniform efficiency. Regarding (3), each point in Figure 1 was produced by combining two data-independent runs, one with zero-temperature initial values for the fields and the other with infinite-temperature initial field values. This is standard practice to ensure that the results are independent of the initial configuration. The nine temperature points also share no data dependencies. This "parallelism" is necessary to determine the peak region in figure 1 and is therefore inherent to our application. Because the strong scaling is basically linear up to 16 racks and because we can perform 18 data-independent runs simultaneously, we can utilize up to 288 racks (well beyond the size of the entire LLNL Sequoia+Vulcan system) without increasing the computational cost. With the full 120-rack LLNL system, our wallclock time to solution would be just 36 hours. This is graphically represented in Figure 8.

7. IMPLICATIONS FOR FUTURE SYSTEMS AND APPLICATIONS

Our results are of importance to all physics practitioners in our field, possibly to condensed matter physicists studying surface states, and also to developers of supercomputing applications and architects of the next generation of supercomputers.

Our research has several implications for future systems as was discussed in some detail in sections 3 and 4.2. These are:

For our application, memory access per operation and communication per operation are about 1 double per flop and severely exposes the application to hardware and software latencies. This is why our application is an exceptional guide for supercomputer architecture design. If a supercomputer cannot sustain a reasonable fraction of peak performance for our simulation, then it is handicapped by large hardware latencies that are likely to impact a host of other applications. Our research suggests then that the next generation of supercomputers, which will likely have an even larger number of compute engines per chip, must focus not only on bandwidth but also on latencies for both on-chip shared memory and inter-node communication.

We have also found that allowing the user to set part of the L1 cache as scratch memory would be very beneficial since it will relieve register pressures and allow for less data movement. For some critical portions of a code, only the application developers know the optimal data layout and access patterns, and so allowing this type of L1 control via high level pragmas can prove to be very advantageous in the more complex hardware world we are heading to.

In addition, we have found that global operations, even if they are infrequent, can substantially handicap performance if they carry large latencies. Optimizing the hardware and software for low-latency global operations in future architectures will be of advantage.

We have also reestablished that the full overlap of communications with computations was crucial for our application. It is basically essential for future architectures to maintain this capability in good standing, especially as the computation rate per node grows.

Our research also demonstrates that the DSDR algorithm is very powerful and of immediate interest to all practitioners of lattice simulations. Our algorithm also may have important implications for condensed matter research that involves surface states, as discussed in section 4.1. The progress reported here now opens the door to a wealth of studies of strongly coupled systems. Some are described below:

One immediate next step is the exploration of the nature (order) of the phase transition as one varies the pion mass below its physical value. It is of great scientific interest to determine the mass value where the transition becomes first order, since this could shed light on the question why our cosmic evolution is what we observe it to be today. Our group has already begun calculations aimed at solving this puzzle.

Research in strongly coupled dark matter using our methods has just begun. Investigating other dark matter candidate models is now possible. Again, our group is already pursuing this with calculations with SU(4) gauge theory (a QCD-like theory with four colors instead of three).

As mentioned above, the question of the origin of quark and electron masses, which account for only 1% of the visible mass in our Universe but are still a crucial ingredient, is another open question. Significant progress in unraveling this mystery has been made recently with the discovery of the Higgs boson at the Large Hadron Collider. According to the Higgs mechanism it is interaction with this particle that gives rise to the masses of the elementary particles such as quarks and electrons. Beyond the Higgs mechanism, exists a great desire to understand the Higgs boson itself, and there is a long-standing conjecture that the Higgs particle is actually composite, with constituents bound very tightly by a new, even stronger interaction of Nature. Such a strongly coupled model cannot be studied in detail without lattice methods, and so research in this area would also greatly benefit from the advances we report here. Indeed, our group has started working to address this mystery as well.

8. ACKNOWLEDGMENTS

The work of MB, TL, CS, RS, PV, and JW was performed under the auspices of the U.S. Department of Energy by LLNL under Contract No. DE-AC52-07NA27344. This research was partially supported by the LLNL LDRD "Unlocking the Universe with High Performance Computing" 10-ERD-033 and by the LLNL LDRD "Illuminating the Dark Universe with Petaflops Supercomputing" 13-ERD-023. Some of the tests were performed using the STFC funded DiRAC facility at Edinburgh. We warmly thank the staff of the Computation Directorate at LLNL for providing early access and assistance to the Sequoia and Vulcan supercomputers. We also wish to warmly thank the Blue Gene IBM team for useful conversations and support.

9. REFERENCES

[1] K. Adcox *et al.*, "Formation of dense partonic matter in relativistic nucleus nucleus collisions at RHIC: Experimental evaluation by the PHENIX collaboration," *Nucl. Phys.*, vol. A757, pp. 184–283, 2005.

[2] K. Aamodt *et al.* [ALICE Collaboration], "Charged-particle multiplicity density at mid-rapidity in central Pb-Pb collisions at $\sqrt{s_{NN}} = 2.76$ TeV," *Phys. Rev. Lett.*, vol. 105, no. 252301, 2010.

[3] "Nobel foundation," Dec 2004. [Online]. Available: http://www.nobelprize.org/nobel_prizes/physics/laureates/2004/index.html

[4] D. J. Gross and F. Wilczek, "ULTRAVIOLET BEHAVIOR OF NON-ABELIAN GAUGE THEORIES," *Phys. Rev. Lett.*, vol. 30, pp. 1343–1346, 1973.

[5] H. D. Politzer, "RELIABLE PERTURBATIVE RESULTS FOR STRONG INTERACTIONS?" *Phys. Rev. Lett.*, vol. 30, pp. 1346–1349, 1973.

[6] S. Bethke *et al.*, "Experimental Investigation of the Energy Dependence of the Strong Coupling Strength," *Phys. Lett.*, vol. B213, p. 235, 1988.

[7] J. Charles *et al.*, "CP violation and the CKM matrix: Assessing the impact of the asymmetric B factories," *Eur. Phys. J.*, vol. C41, pp. 1–131, 2005.

[8] K. G. Wilson, "CONFINEMENT OF QUARKS," *Phys. Rev.*, vol. D10, pp. 2445–2459, 1974.

[9] J. B. Kogut and L. Susskind, "Hamiltonian Formulation of Wilson's Lattice Gauge Theories," *Phys. Rev.*, vol. D11, p. 395, 1975.

[10] H. B. Nielsen and M. Ninomiya, "No Go Theorem for Regularizing Chiral Fermions," *Phys. Lett.*, vol. B105, p. 219, 1981.

[11] F. Zwicky, "Spectral displacement of extra galactic nebulae," *Helv. Phys. Acta*, vol. 6, pp. 110–127, 1933.

[12] V. C. Rubin and J. Ford, W. Kent, "Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions," *Astrophys. J.*, vol. 159, pp. 379–403, 1970.

[13] A. Borriello and P. Salucci, "The Dark Matter Distribution in Disk Galaxies," *Mon. Not. Roy. Astron. Soc.*, vol. 323, p. 285, 2001.

[14] H. Hoekstra, H. Yee, and M. Gladders, "Current status of weak gravitational lensing," *New Astron. Rev.*, vol. 46, pp. 767–781, 2002.

[15] R. B. Metcalf, L. A. Moustakas, A. J. Bunker, and I. R. Parry, "Spectroscopic Gravitational Lensing and Limits on the Dark Matter Substructure in Q2237+0305," *Astrophys. J.*, vol. 607, pp. 43–59, 2004.

[16] S. Weinberg, "Implications of Dynamical Symmetry Breaking: An Addendum," *Phys.Rev.*, vol. D19, pp. 1277–1280, 1979, (For original paper see Phys.Rev.D13:974-996,1976).

[17] L. Susskind, "Dynamics of Spontaneous Symmetry Breaking in the Weinberg- Salam Theory," *Phys. Rev.*, vol. D20, pp. 2619–2625, 1979.

[18] T. Appelquist et. al., the LSD collaboration, "Lattice calculation of composite dark matter form factors", e-Print: arXiv:1301.1693 [hep-ph], submitted to *Phys. Rev. D*, 2013.

[19] S. Borsanyi, G. Endrodi, Z. Fodor, A. Jakovac, S. D. Katz, S. Krieg, C. Ratti, K. K. Szabo, "The QCD equation of state with dynamical quarks", *JHEP*, 1011:077, 2010.

[20] The HotQCD collaboration, "The chiral transition and U(1)A symmetry restoration from lattice QCD using Domain Wall Fermions", *Phys.Rev.*, vol. D86, 094503, 2012. ; The HotQCD collaboration in preparation.

[21] P. Vranas, "Gap Domain Wall Fermions" *Phys.Rev.*, vol. D74, 034512, 2006.

[22] Aprile et al., "Dark Matter Results from 225 Live Days of XENON100 Data", *Phys. Rev. Lett.*, 109, 181301, 2012.

[23] P. H. Ginsparg and K. G. Wilson, "A Remnant of Chiral Symmetry on the Lattice," *Phys. Rev.*, vol. D25, p. 2649, 1982.

[24] S. A. Gottlieb, W. Liu, D. Toussaint, R. L. Renken, and R. L. Sugar, "Hybrid Molecular Dynamics Algorithms for the Numerical Simulation of Quantum Chromodynamics," *Phys. Rev.*, vol. D35, pp. 2531–2542, 1987.

[25] M. Luscher, S. Sint, R. Sommer, P. Weisz, and U. Wolff, "Non-perturbative O(a) improvement of lattice QCD," *Nucl. Phys.*, vol. B491, pp. 323–343, 1997.

[26] C. Gattringer and C. B. Lang, "Quantum chromodynamics on the lattice," *Lect.Notes Phys.*, vol. 788, pp. 1–211, 2010.

[27] "Columbia physics system." [Online]. Available: http://qcdoc.phys.columbia.edu/cps.html

[28] P. Vranas, "Domain wall fermions in vector theories", Proceedings, NATO Advanced Research Workshop, Dubna, Russia, October 5-9, 1999, C99-10-05.3, p.11-26, 1999.

[29] D. Renfrew, T. Blum, N. Christ, R. Mawhinney, P. Vranas, "Controlling Residual Chiral Symmetry Breaking in Domain Wall Fermion Simulations", *PoS LATTICE2008*, p. 0.48, 2008.

[30] D. B. Kaplan, "A Method for simulating chiral fermions on the lattice," *Phys. Lett.*, vol. B288, pp. 342–347, 1992.

[31] P. M. Vranas, "Chiral symmetry restoration in the Schwinger model with domain wall fermions," *Phys. Rev.*, vol. D57, pp. 1415–1432, 1998.

[32] M. Creutz, I. Horvath, "Surface states and chiral symmetry on the lattice", *Phys.Rev.*, vol. D50, pp. 2297-2308, 1994.

[33] R.C. Brower, R. Babich, K. Orginos, C. Rebbi, D. Schaich, P. Vranas, "Moebius Algorithm for Domain Wall and GapDW Fermions", *PoS LATTICE2008*, p. 034, 2008.

[34] R.C. Brower, H. Neff, K. Orginos, "The Moebius Domain Wall Fermion Algorithm", e-Print: arXiv:1206.5214 [hep-lat].

[35] R. A. H. et al., "The IBM Blue Gene/Q Compute Chip," *Micro, IEEE*, vol. PP, p. 1, Dec 2011.