ICCM 2007 · Vol. III · 253–268

Robust and Efficient Numerical Linear Algebra Solvers and Applications in Quantum Mechanical Simulations

Zhaojun Bai* Wenbin Chen† Richard Scalettar‡
Ichitaro Yamazaki§

Abstract

Optimization of large scale linear algebra computations is a long-standing problem in numerical analysis and scientific computing communities. In this pape, we describe our recent synergistic effort on the development of robust, accurate and efficient linear algebra techniques and applications to quantum mechanical simulation. We demonstrate the feasibility, through the use of newly developed linear algebra solvers, of 1000-electron quantum monte carlo simulations on a modern desktop machine. Such simulations would allow us to address important questions concerning the magnetic and transport properties of materials with strong electron-electron interactions.

The results of robust and efficient linear algebra solvers have more general impact on forefront scientific computing beyond the application discussed here. As one example, the methodology described has close connections to problems in lattice gauge theory, dynamical mean field theory, and localization.

2000 Mathematics Subject Classification: 15A06, 65F05, 65F10, 65F40.
Keywords and Phrases: Linear systems, Iterative methods, Hubbard model, Monte carlo simulation.

1 Introduction

Forefront problems of scientific and engineering computing often require solutions involving large scale matrix computations. Great progress has been made both in general procedures and also in more focused situations which exploit specific

*Department of Computer Science and Department of Mathematics, University of California, Davis, CA 95616, USA. E-mail: bai@cs.ucdavis.edu.

†School of Mathematical Science, Fudan University, Shanghai 200433, China. E-mail: wbchen@fudan.edu.cn.

‡Department of Physics, University of California, Davis, CA 95616, USA. E-mail: scalettar@physics.ucdavis.edu.

§Department of Computer Science, University of California, Davis, CA 95616, USA. E-mail: yamazaki@cs.ucdavis.edu.

matrix structure and sparsity patterns. A generation of excellent algorithms has been developed and presented in textbooks, see for example [22, 14, 46], and research monographs, such as [34, 37, 38, 10, 26, 23, 48]. Vast amounts of public and proprietary software libraries, packages and templates have been made widely accessible, such as LAPACK and ScaLAPACK among many others. The impact of progress in large scale matrix computations to the advancement of computational science and engineering cannot be overstated.

However, challenges of the advancement of multi-scale mathematical modeling and simulation to numerical linear algebra theory and algorithms have not been widely addressed. Most solvers are not designed in ways that are robust and efficient for underlying multi-length scale simulations. This is an emerging research area. A broad range of scientific and engineering modeling and simulation problems involve multiple scales for which traditional monoscale approaches have proven to be inadequate, even with the largest supercomputers.

In this paper, we describe a range of synergistic activities on development of robust and efficient linear algebra solvers which are specially designed for multi-length scale numerical linear algebra problems arising from quantum mechanical simulations of materials. This includes new algorithm design and analysis, hybrid use of existing algorithms and development of high-performance software. We focus on the following computational kernels of quantum simulations to be described later: (a) Study of the dynamics of the eigenvalue distributions, condition numbers and other critical properties of multi-length scale matrices. (b) Development of robust and efficient self-adapting linear algebra solvers. (c) Development of high performance software solving real multi-length scale phenomena.

2 Computational material science and Hubbard model

Our effort is triggered by a specific application which forms one of the core problems in materials science: How do the interactions between electrons in a solid give rise to properties like magnetism, superconductivity, and metal-insulator transitions? Our ability to solve this central question in quantum statistical mechanics is presently limited to systems of a few hundred electrons. While simulations at this scale have taught us a considerable amount about certain classes of materials, they have very significant limitations, especially for recently discovered materials which have mesoscopic magnetic and charge order.

High temperature superconductors [13] are one example. These materials have 'parent' phases which are antiferromagnetic insulators. It was suggested very early on that superconductivity arises from the exchange of vibrations of the antiferromagnetically aligned spins ("spin waves") instead of the usual vibrations of the nuclear positions ("phonons") in conventional materials. The original picture was that the magnetism and superconductivity more or less coexist uniformly throughout the material. It is now believed that instead, "striped" and "checkerboard" patterns arise in which antiferromagnetic and superconducting regions are *separate*. The work [24] shows a map of the conductance obtained when tunneling

into a cuprate material. Larger scale checkerboard structures are evident. Exactly how these arise is still a puzzle, especially since the models which successfully describe much of their behavior contain only local interactions. Manganites are a second example [16, 12]. These materials are key ingredients in magnetic storage disks owing to their giant magneto-resistance – the large change in resistance they exhibit under the application of a small magnetic field. In the manganites, coexisting mesoscopic clusters of ferromagnetic and charge ordered states arise at scales which are much larger than the simple atomic separations. The delicate balance between the two causes the great sensitivity of the resistance to an external magnetic field, and leads to their technological usefulness.

Condensed matter physicists have developed a number of models which seem to capture both the different types of order in these solids, and their competition. The Hubbard Hamiltonian [15] is a popular one. Specifically, a two-dimensional Hubbard model is defined by the Hamiltonian

$$\mathcal{H} = \mathcal{H}_K + \mathcal{H}_\mu + \mathcal{H}_V, \tag{2.1}$$

where $\mathcal{H}_K$, $\mathcal{H}_\mu$ and $\mathcal{H}_V$ stand for kinetic, chemical and potential energy, respectively, and are defined as

$$\begin{aligned}
\mathcal{H}_K &= -t \sum_{\langle i,j\rangle,\sigma} (c^\dagger_{i\sigma} c_{j\sigma} + c^\dagger_{j\sigma} c_{i\sigma}), \\
\mathcal{H}_\mu &= -\mu \sum_i (n_{i\uparrow} + n_{i\downarrow}) \\
\mathcal{H}_V &= U \sum_i (n_{i\uparrow} - \frac{1}{2})(n_{i\downarrow} - \frac{1}{2})
\end{aligned}$$

and

- i and j label the spatial sites of the lattice. $\langle i, j\rangle$ represents a pair of nearest-neighbor sites in the lattice and incidates that the electrons only hopping to nearest neighboring sites.
- $c^\dagger_{i\sigma}$ and $c_{i\sigma}$ are the fermion creation and annihilation operators for electrons located on the ith lattice site with z component of spin-up ($\sigma = \uparrow$) or spin-down ($\sigma = \downarrow$), respectively.
- The operators $n_{i\sigma} = c^\dagger_{i\sigma} c_{i\sigma}$ are the number operators which count the number of electrons of spin σ on site i.
- t is the hopping parameter of kinetic energy, and describes the motion of electrons between the atoms.
- U is an interaction parameter and measures the cost of two electrons occupying the same atomic site.
- μ is the chemical potential parameter which controls the electron numbers (or density).

The physics of the Hubbard Hamiltonian is determined by the competition between these two scales, t and U, and by the temperature T, and by the density of electrons. When U dominates, magnetic order tends to be favored at low temperatures. Similarly, at appropriate electron densities, large values of U cause insulating states to form.

The Hubbard Hamiltonian has successfully described many of the qualitative features of materials like the cuprate superconductors- their antiferromagnetism [28], unconventional superconductivity [40, 39], mesoscopic charge ordering [17, 18, 45]. and the charge and spin patterns in a limit of the Hubbard model [51]. Despite this success, the restriction of such numerical studies to a few hundred sites/electrons has prevented many fundamental issues from being addressed, not only for the cuprate superconductors, but also for many strongly correlated systems, including manganites and cobaltates. The inhomogeneous phases exhibited by these materials pose an enormous challenge to simulations since the lattice sizes required to capture the physics is considerably larger than for homogeneous states. Therefore modeling these important materials constitutes a valuable application for our multiscale matrix algorithms. In addition, it should be emphasized that a host of other applications awaits the development of multiscale linear algebra solvers. Problems very closely related to the Hubbard Hamiltonian described here include lattice gauge theory [20, 36, 5], which shares many of the matrix structures and algorithms with condensed matter QMC simulations, dynamical mean field theory [50, 21], and Hatano-Nelson problems in localization [25].

3 Multi-length scale numerical linear algebra

Considerably less attention has been devoted to multi-length scale numerical linear algebra problems, where the underlying matrices for example, have a block structure defining one scale (the size of the blocks) but then also internal scales within the blocks themselves. The existence of multiple 'length (block)' scales is often accompanied by the presence of multiple scales in the size of the matrix elements. In the case of the applications we shall consider, these size scales are associated with the several energy scales in our materials. As we shall describe in the subsequent sections, we have investigated several important aspects of multi-length scale numerical linear algebra problems. Before turning to this work, let us describe the specific multi-length scale matrices arising in the quantum monte carlo simulations of the Hubbard model which will enable us to describe the fundamental issues within a concrete context.

The matrix computation kernels of the determinant and hybrid QMC simulations of the Hubbard model [9, 27, 41, 4] form an important practical illustration of multi-length scale numerical linear algebra problems. The matrices start with the simple block cyclic structure

$$M = \begin{bmatrix} I & & & & B_1 \\ -B_2 & I & & & \\ & -B_3 & I & & \\ & & \ddots & \ddots & \\ & & & -B_L & I \end{bmatrix},$$

where the nontrivial blocks B_ℓ are constructed as the product of the exponentials of two matrices arising from the kinetic and potential energies:

$$B_\ell = e^{t\Delta\tau K} e^{\sigma\nu V_\ell(h)},$$

where K is a lattice structure matrix, namely, $K_{ij} = 1$ if i and j are nearest neighbors, otherwise, $K_{ij} = 0$. $V_\ell(h) = \mathrm{diag}(h_{\ell i})$, $h_{\ell i}$ are Hubbard-Stratonovich (random) variables which must be sampled stochastically using monte carlo or molecular dynamics. The integer $L = \beta/\Delta\tau$ is given by the ratio of the inverse of the temperature being simulated to a discretization stepsize $\Delta\tau$. $\sigma = \pm$ are spins, and $\nu = \cosh^{-1}(2U\Delta\tau)$. More complex structures of M are possible, and will arise in applications. For instance, the matrices V_ℓ describing the potential energy become non-diagonal if interactions between electrons on different sites or orbitals are considered. The key features of these matrices can be summarized as the following:

- M incorporates multiple structural scales: The inverse temperature β determines the number of blocks $L = \beta/\Delta\tau$. The dimension of the individual blocks is set by N the number of spatial sites. In a 2D simulations $N = N_x \times N_y \propto 10^2$. Thus the total dimension of the M currently being studied is 10^4. Our goal is to extend this by an order of magnitude to 10^5.
- M incorporates multiple energy scales: The parameter t which enters K determines the kinetic energy of the electrons, and the interaction energy scale U enters $V_\ell(h)$.
- M is a function of a collection of NL variables, the Hubbard-Stratonovich field $h_{\ell i}$. The role of the simulation is to determine the configurations of these variables which make large contributions to operator expectation values, and then to sum over those configurations. Therefore, the associated matrix computation problems need to be solved $O(10^4)$ times in a full simulation.

Numerical linear algebra which enter quantum simulations are the following:

1. Solution of $Mx = b$ for x.
2. Solution of $M^T Mx = b$ is needed in a molecular dynamics step.
3. Computation of specific elements of the inverse $(M^{-1})_{ij}$. These determine all the physical observables: energy, density, magnetic moments, etc.
4. Computation of trace(M^{-1}).
5. Computation of $\frac{\det(\widehat{M})}{\det(M)}$, where $\widehat{M}$ is a low-rank update of M is needed in the accept/reject decision of a monte carlo move involving a change to a small number of $h_{\ell i}$.

The computational challenge is to increase the spatial dimension $N = N_x \times N_y$ from $O(10^2)$ to $O(10^3)$, that is, to do a 1000 electron QMC simulation. Such an increase would have a tremendous impact on our understanding of strongly interacting materials because it would allow for the first time the simulation of systems incorporating a reasonable number of the mesoscopic structures, such as the checkerboards and stripes shown in [24].

4 Multi-length scale matrix analysis

Before achieving our goals of developing robust and efficient algorithmic techniques and high performance software, we first study the dynamics and transitional behavior of mathematical and numerical properties of the underlying matrices, such

as eigenvalue distribution and condition number as functions of the multi-lenght scales. As described in section 3, the Hubbard matrix M is characterized by structural scales $N = N_x \times N_y$ and L, and energy scales U and t and others. Figure 1 shows the eigenvalue distributions when N, L, U and t vary independently in various simple limits. The eigenspectrum behavior is much more dramatic and complex when N, L, U and t vary simultaneously in ranges of interest in simulations. In such cases the eigenvalue distributions are typically much more dense and shifted away from the center $\lambda = (1, 0)$. This is the origin of difficult arising in the numerical computations. For the non-interacting limit ($U = 0$) of the 2-D Hubbard model, we have extended the previous known result [49] for an analytical expression of eigenvalues of M:

$$\lambda(M) = 1 - e^{t\Delta\tau\epsilon_{x,y}} e^{\mathrm{i}\frac{(2\ell+1)\pi}{L}}, \quad 0 \leqslant \ell \leqslant L-1,$$

where $\epsilon_{x,y} = 2(\cos\theta_x + \cos\theta_y)$ and $\theta_x = \frac{2k_x\pi}{N_x}$, $\theta_y = \frac{2k_y\pi}{N_y}$ for $k_x = 1, 2, \cdots N_x$ and $k_y = 1, 2, \cdots N_y$.

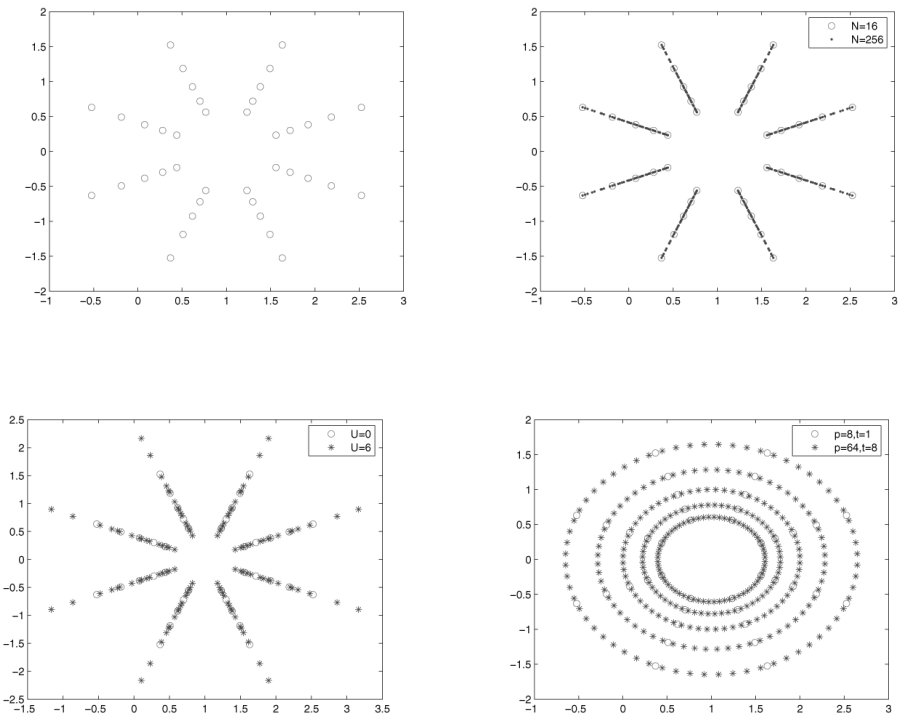

Figure 1 Eigenvalue distributions with respect to the changes of length scale parameters N and L, and energy scale parameters U and t.

Let us now consider how this evolution of the eigenspectrum is reflected in the condition number of the matrix M with respect to inversion, namely, $\kappa(M) =$

$\|M\|\,\|M^{-1}\|$. First, if $U = 0$, the matrix M is independent of the Hubbard-Stratonovich field configuration h. In this case, we can derive a theoretical bound to show that the condition number of M only increases slowly with respect to the increase of the multiscale parameter L:

$$\kappa(M) \leqslant \frac{1 + e^{4t\Delta\tau}}{\sin\frac{\pi}{L}} \approx \mathcal{O}(L).$$

However, if $U \neq 0$, there is no known rigorous theoretical bound. Figure 2 shows the average condition numbers for 100 Hubbard-Stratonovich field configurations as a function of L for $U = 2, 4, 6$. The figure illustrates two key points concerning the transition from well-conditioned to ill-conditioned behavior: When $U \neq 0$ the condition number increases much more rapidly than the linear rise which we know analytically at $U = 0$; Not only does the condition number increase with U, but also so do its fluctuations over the 100 chosen field configurations. The first observation tells us the parameter L is critical to the difficulty of our solvers. The second suggests that widely varying condition number might be encountered in the course of a simulation, and therefore that codes might need to have the ability to adopt different solution strategies on the fly.

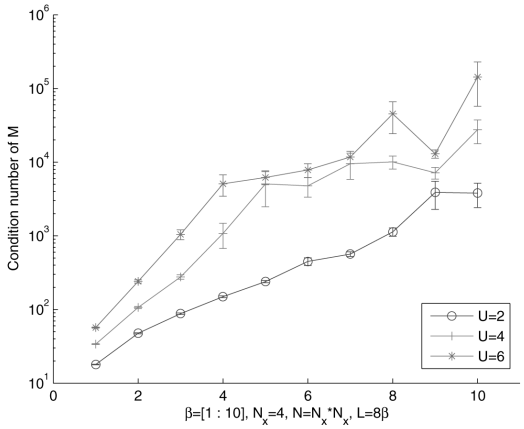

Figure 2 Average condition numbers of sample matrices and theoretical and empirical bounds

There are a number of directions to extend both theoretical analysis and empirical observation of the influence of parameters N, L, U and t on distributions of eigenvalues and condition numbers of Hubbard matrices M. This study will reveal the conditioning of the problems to be solved and the impact to the stability and convergence behaviors of direct and iterative methods we will develop. The methodology and paradigms developed here will also help in the study of multi-length scale matrices arising in other applications. The most closely related problem is that of lattice gauge theory, an area which has traditionally been one of the central fields of computational science (with many of the largest blocks of time at national supercomputer centers devoted to its solution). There is in fact

an incredible similarity in the formalisms and computational kernels of the two fields.

5 Self-adaptive direct solvers

As we have discussed in the previous sections, one of our major goals in this project is to develop algorithmic techniques and paradigms that can robustly and efficiently solve the linear algebra problems with underlying multi-length scale matrices in a self-adapting fashion to achieve a required simulation accuracy. We have exploited the development of such an algorithmic technique for the linear system of equations associated with the Hubbard matrix M as described in section 3 The matrix M exhibits the form of a so-called block p-cyclic consistently ordered matrix [49]. p-cyclic matrices arise in a number of important contexts in applied mathematics, including numerical solution of boundary value problems for ordinary differential equations [47], finite-difference equations for the steady-state solution of a parabolic equation with periodic boundary conditions [43], and computing the stationary solution of Markov chains with periodic graph structure [42].

It is known that the block Gaussian elimination with and without pivoting for solving p-cyclic linear systems can be numerically unstable, similar to the case of multiple shooting method for solving two-point boundary value problems [53, 19] and Markov chain modeling [35]. Block cyclic reduction [11] is a powerful idea to solve such p-cyclic system. However, a full block cyclic reduction is applicable only for small energy scales, namely, $U \leqslant 1$, due to the emerging of ill-conditioning of the reduced system. A stable p-cyclic linear system solver is based on the structural orthogonal factorization [52, 19]:

$$M = QR, \tag{5.1}$$

where Q is an orthonormal matrix, and R is a block upper triangular matrix of the form

$$R = \begin{bmatrix} R_{11} & R_{12} & & & R_{1p} \\ & \ddots & \ddots & & \vdots \\ & & & R_{p-1,p-1} & R_{p-1,p} \\ & & & & R_{pp} \end{bmatrix}.$$

Unfortunately, the orthogonal factorization forces the significant increase of fill-in in R, and requires $O(N^2L)$ memory in general, which could be prohibitively expensive when the length scales N and L increase. Furthermore, the computational cost of floating point operations is $O(N^3L)$.

To take advantage of significant reduction of memory requirement and floating point computations in the block cyclic reduction and numerical stability of the orthogonal factorization method, and to carefully examine the accuracy needs in our quantum monte carlo simulation, we have studied a hybrid method. The method has three stages:

1. Perform a factor k block cyclic reduction:

$$Mx = b \quad \Longrightarrow \quad M^{(k)}x^{(k)} = b^{(k)}.$$

Namely, the initial block L–cyclic system is cyclically reduced to a block $L_k = \frac{L}{k}$–cyclic system with the coefficient matrix

$$M^{(k)} = \begin{bmatrix} I & & & & B_1^{(k)} \\ -B_2^{(k)} & I & & & \\ & -B_3^{(k)} & I & & \\ & & \ddots & \ddots & \\ & & & -B_{L_k}^{(k)} & I \end{bmatrix}.$$

and

$$\begin{aligned} B_1^{(k)} &= B_k B_{k-1} \cdots B_2 B_1 \\ B_2^{(k)} &= B_{2k} B_{2k-1} \cdots B_{k+2} B_{k+1} \\ &\vdots \\ B_{L_k}^{(k)} &= B_L B_{L-1} \cdots B_{(L_k-1)k+1}. \end{aligned}$$

The conditioning number of $M^{(k)}$ increases with the increase of the reduction factor k.

2. Solve the reduced cyclic system by the structural block orthogonal factorization

$$Q_{\frac{L}{k}-1}^T \cdots Q_1^T M^{(k)} = R^{(k)}.$$

In this way, the memory and computational costs are effectively reduced by a factor of k comparing to the original system.

3. Forward and back substitute to find the remaining block components x_i of the solution x:

$$x_i \quad \longleftarrow \quad x^{(k)} \quad \longrightarrow \quad x_j.$$

We use both forward and back substitutions to minimize the propagation of errors induced at the steps 1 and 2.

We see that by Step 1, the order of $M^{(k)}$ is reduced by a factor of k, therefore, it is desirable that the larger k, the better. The computational cost is reduced from $O(N^3 L)$ to $O(N^3 \frac{L}{k})$, an effective factor k speedup. However, the condition number of $M^{(k)}$ increases when k increases, which means the accuracy of the computed solution decreases. Therefore, the critical question turns to how to find a reduction factor k, such that the computed solution has the required accuracy for the application. Such a reduction factor k should be determined in a *self-adapting* fashion with respect to the changes of underlying problem length and energy scales.

By a rigorous error analysis, we show that the relative error in the computed block components $\widehat{x}_\ell$ is governed by $\kappa(M^{(k)})\epsilon$ and the propagation error in the substitution. Specifically,

$$\frac{\|x_\ell - \widehat{x}_\ell\|}{\|x_\ell\|} \leqslant \|B_{\frac{k}{2}}\| \cdots \|B_2\| \, \|B_1\| \, \kappa(M^{(k)})\epsilon,$$

where ϵ is the machine precision. Subsequently, the focus turns into how to estimate the condition numbers of reduced matrices $\kappa(M^{(k)})$. By exploiting the reduction process and the structure of the matrices involved, we have shown that

$$\kappa(M^{(k)}) \leqslant ce^{k(4t\Delta\tau+\nu)}\kappa(M),$$

where $t, \Delta\tau$ and $\nu(U)$ are energy scale parameters. c is a constant independent of these parameters. The left plot of Figure 3 shows the actual condition numbers of sample matrices $\kappa(M^{(k)})$ associated with the increase of the reduction factor k, and the theoretical bound. The plot suggests that the growth of the condition numbers of the reduced cyclic matrices $\kappa(M^{(k)})$ is more close to $e^{\frac{k}{2}(4t\Delta\tau+\nu)}\kappa(M)$. It is a subject of future study.

Combining these analyses, for a desired accuracy "tol" of the solution vector x with machine precision ϵ, then a reduction factor k can be *self-adaptively* determined by

$$k = \left\lfloor \frac{\frac{2}{3}\ln(\mathrm{tol}/\epsilon)}{4t\Delta\tau + \nu} \right\rfloor.$$

The right plot of Figure 3 shows the reduced blocksizes $L^{(k)}$ in a self-adaptive fashion with respect to the variations of energy parameters U and $\beta(t)$. All computed solutions satisfy the specified relative errors $\mathrm{tol} = 10^{-8}$ with double precision arithmetic $\epsilon = 10^{-16}$. We observed that the reduction factor k is immediately translated into a factor k of speedup in computational time.

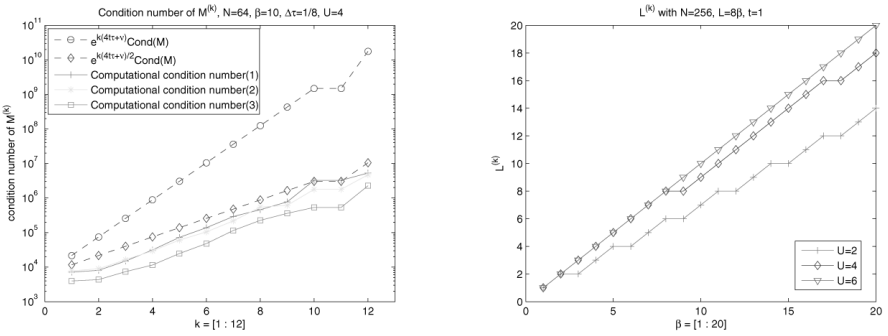

Figure 3 Left: condition numbers of the reduced $L^{(k)}$–cyclic system and theoretical bounds. Right: Self-adaptively determined reduction factor k with respective to different scales of U and β.

The self-adapting block cyclic reduction solver is robust, and effectively reduces both memory and computation by a factor of k. We expect it will be a

workhorse in the whole QMC simulation package we are currently working on. We note that block cyclic reduction is natural for parallelism. On the orthogonal factorization, we will exploit the parallel structural orthogonal factorization techniques, such as developed in [52, 2, 19].

6 Preconditioning techniques

Iterative methods with the proper preconditioning techniques often reduce the computational complexity of solving large scale linear algebra problems by an order of magnitude or more. For our particular targeted application domain of the multiscale quantum statistical mechanical simulations, it might be the only path toward optimal simulations, in which the computational complexity increases *linearly* with the number N of electrons or the lattice size.

The development of preconditioning techniques has been an active research topic for decades, for example [3, 48], and reference therein. For preconditioning techniques, we consider the QMC computational kernel of solving symmetric positive definite systems of the form

$$Ax = b,$$

where $A = M^T M$, M is the multi-length scale Hubbard matrix as described in section 3. We use the preconditioned conjugate gradient (PCG) method to solve symmetrical preconditioned linear system

$$R^{-T} A R^{-1} \cdot Rx = R^{-T} b,$$

where R is a preconditioner. Normally, R is constructed such that (a) $R^T R$ is a good approximation of A in some sense, (b) the cost of constructing R is affordable, and (c) the application of R is not expensive, namely the system $Rz = c$ is much easier to solve than the original system. However, in the multi-length scale setting, we have to consider the dynamics of the underlying matrix M to be preconditioned. Therefore, the preconditioner R should also try to accommodate.

Earlier work on preconditioning techniques to solve the linear system turned out to be unsuccessful, high cost (memory and flops) for the construction of R, and/or large number of iterations and high CPU cost, particularly when $N, U, \beta(L)$ increase. The key question often asked in the quantum simulation community is *"is there a linear-scaling iterative solver?"*. More specifically, when U and β increase, whether there is a solver with the number of iterations grow slowly, not "exponentially" as seen from the previous work?

Incomplete Cholesky (IC) factor R is a popular preconditioner:

$$A = R^T R + E,$$

where R is an upper triangular matrix and E is the error matrix. We have observed that for our multi-length scale system, if by only imposing a certain sparsity pattern of R (based on the block structure of M), or by dropping small elements, then typically, IC leads to high cost to apply R due to large number of fill-ins,

and poor quality, such as large numbers of PCG iterations, Furthermore, the IC preconditioning is not robust. It suffers *pivot break-down* due to loss of positive definiteness of $A - E$ when the energy scale U is large. Mathematically, provable existence for such an incomplete factorization is only for special classes of matrices [31, 30].

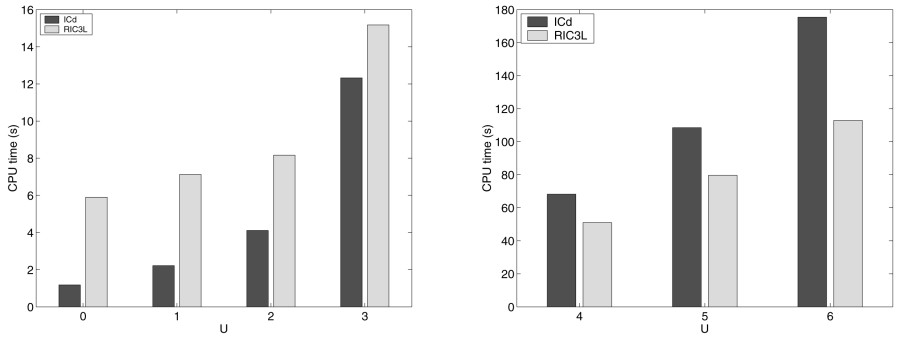

Figure 4 Comparison of the speed of IC_d and RIC3 preconditioners. Left: for small U. Right: for large U. where $N = 32 \times 32$ and $L = 80$.

We have pursued the work of robust incomplete Cholesky (RIC) preconditioners, previously developed in [1, 44, 29, 7, 8]. The RIC preconditioners avoid the pivot-break down by imposing the positive definiteness of $A - E$:

$$\begin{cases} A - E = R^T R \\ \text{subject to } A - E > 0. \end{cases}$$

A simple statical approach is to use a global shifting of the diagonal entries of A [30], i.e, compute $A + \alpha D_A = RR^T + E$, where α is a shift and $D_A = \mathrm{diag}(A)$. We refer to it as the IC_d preconditioner.

Alternatively, one can dynamically impose the robust by further constraints in the structure of the error matrix E. For example, it is proposed to impose $E = R^T F + F^T R + S$ to improve the quality of the preconditioner R, in addition to the requirement of positive definite $A-E$ [44, 29]. We refer such a preconditioner as RIC3.

IC_d is cheaper to construct, but of poor quality for the system with large energy scale U. On the other hand, RIC3 is more expensive to construct, but is a better preconditioner for large U. See Figure 4 for the CPU performance with respect to the different energy scale. In these performance data, we have used a modified version of commonly used compressed sparse rom (CSR) format for the sparse matrix data structure to accommodate the data access pattern in the RIC factorizations.

We have demonstrated that with the preconditioner RIC3, the number of PCG iterations and CPU timing scale linearly with N for small to moderate energy scale $U \leqslant 3$, see Figure 5 Such a linear scaling algorithm is an exciting progress. U-dependence in terms of the number of PCG iterations and CPU is unexpected (not

exponential growth with the increase of U as seen in the previous work). However, when the interaction energies are high, i.e., $U \geqslant 4$, the linear system becomes ill-conditioned and the linear scaling is lost. Extending our success to $U \geqslant 4$ remains a challenge. We have observed that for large energy scale, the PCG stagnates after some initial rapid decline. This stagnation of PCG convergence (plateau) is often due to the slow convergence of components of residual vectors associated with small eigenvalues. Several techniques have been proposed to deflate these components from the residual vector so that the plateau of convergence can be avoided [32, 33].

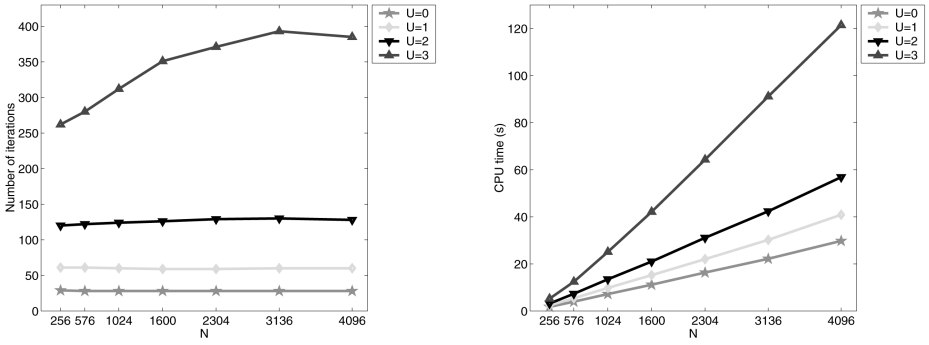

Figure 5 Left: the number of PCG iteration. Right: CPU timing. Both grow linearly with respect to N, where $L = 80$.

It is important to extend both theoretical analysis and empirical observation of the performance of different preconditioners under the influence of scaling parameters N, L, U, β and t. We are considering a technique to use the knowledge gained from the previous solutions in the course of simulation. If the preconditioner may be reused over several solutions, it eliminates the cost to reconstruct the preconditioner. Such study should lead to a precise quantification of the selection of the optimal preconditioners with respect to the changes of multi-length scales. It should be similar to the self-adapting reduction factor k as described section 5 Automatic optimization of sparse matrix operations have been intensely studied in the past years. We have examined Optimized Sparse Kernel Interface (OSKI) [6], which provides automatically tuned computational kernels for sparse matrices on a particular platform. However, because of extreme sparse nature of our matrices, this technique does not result in performance improvement yet.

7 Concluding remarks

Large-scale numerical linear algebra problems arise throughout the computational simulation process of scientific discovery and engineering design. In this paper, we have presented our synergistic effort in the past few years on the development of efficient and accurate numerical linear algebra solvers appropriate for multi-length scale simulations. We focus on: (a) Reliable computing, which includes

theoretical study of existing and new numerical methods, and post-processing verification tools and analysis. (b) Structure exploitation, which is particularly important for the ever larger and more complex problems arising from realistic modeling of important physical systems. (c) Software and toolbox development, particularly user interface design, which requires intensive communication with intended users. (d) Use of high-performance computing, which exploits today's and emerging computer architectures and programming environments.

Acknowledgments. This work was partially supported by the National Science Foundation under Grant 0313390 and DOE SciDAC project on next generation multi-scale quantum simulation software for strongly correlated materials (DE-FC02 06ER25793). Wenbin Chen was also supported in part by the China Basic Research Program under the grant 2005CB321701.

References

[1] M. A. Ajiz and A. Jennings. A robust incomplete Choleski-conjugate gradient algorithm. *Int. J. Numer. Meth. Eng.*, 20:949–966, 1984.
[2] P. Amodio and M. Paprzycki. A cyclic reduction approach to the numerical solution of boundary value ODES. *SIAM J. Sci. Comput.*, 18:56–68, 1997.
[3] O. Axelsson. *Iterative Solution Method*. Cambridge University Press, Cambridge, 1994.
[4] Z. Bai, W. Chen, R. T. Scalettar and I. Yamazaki. Lecture Notes on Advances of Numerical Methods for Quantum Monte Carlo Simulations of the Hubbard Model. *to appear*, 2007
[5] G.G. Batrouni, G.R. Katz, A.S. Kronfeld, G.P. Lepage, B. Svetitsky and K.G. Wilson. Langevin simulations of lattice field theories. *Phys. Rev. D*, 32:2736 (1985).
[6] Berkeley Benchmarking and Optimization Group. OSKI: Optimized sparse kernel interface. http://bebop.cs.berkeley.edu/oski/.
[7] M. Benzi and M. Tuma. A robust incomplete factorization preconditioner for positive definite matrices. *Numer. Lin. Alg. Appl.*, 10:385–400, 2003.
[8] M. Benzi and M. Tuma. A robust preconditioner with low memory requirements for large sparse least squares problems. *SIAM J. Sci. Comput.*, 25:499–512, 2005.
[9] R. Blankenbecler, D. J. Scalapino and R. L. Sugar, Monte Carlo calculations of coupled Boson-fermion systems I, Phys. Rev. D, 24(1981), 2278-2286.
[10] Å. Björck. *Numerical Methods for Least Squares Methods*. SIAM, Philadelphia, 1996.
[11] B.L. Buzbee, G.H. Golub and C.W. Nielson. On direct methods for solving poisson's equations. *SIAM J.Numer. Anal.*, 7:627–656, 1970.
[12] E. Dagotto. Nanoscal phase separation and colossal magnetoresistance, Springer, Berlin, 2002.
[13] E. Dagotto. Correlated electrons in high-temperature superconductors, *Rev. Mod. Phys.* 66, 763 (1994).
[14] J. Demmel. *Applied Numerical Linear Algebra*. SIAM, Philadelphia, 1997.

[15] A. Montorsi (ed). The Hubbard Model. World Scientific, 1992.
[16] Y. Tokura (ed). *Collosal magnetoresistance oxides*, gordon and breach. New York, 2000
[17] V.J. Emery, S.A. Kivelson and H.Q. Lin. Phase separation in the $t-J$ model. *Phys. Rev. Lett.* 64, 475 (1990).
[18] S. A. Kivelson et al. How to detect fluctuating stripes in the high-temperature superconductors. *Rev. Mod. Phys.* 75, 1201-1241 (2003).
[19] G. Fairweather and I. Gladwell. Algorithms for almost block diagonal linear systems. *SIAM Review*, 46(1):49–58, 2004.
[20] F. Fucito, E. Marinari, G. Parisi and C. Rebbi. Nucl. *Phys. b* 180, 369 (1981).
[21] A. Georges, G. Kotliar, W. Krauth and M. Rozenberg. *Rev. Mod. Phys.* 68, 13 (1996).
[22] G. Golub and C. Van Loan. *Matrix Computations* (2nd Edition). Johns Hopkins University Press, Baltimore, 1989.
[23] A. Greenbaum. *Iterative Methods for Linear Systems*. SIAM, Philadelphia, 1997.
[24] T. Hanaguri et al. A "checkerboard" electronic crystal state in lightly hole-doped $Ca_{2-x}NaxCuO_2Cl_2$. *Nature*, 430, 1001 (2004).
[25] N. Hatano and D. R. Nelson. How to detect fluctuating stripes in the high-temperature superconductors. *Phys. Rev. Lett.* 77:570-573 (1996).
[26] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, Philadelphia, PA, 1996.
[27] J. E. Hirsch. Two-dimensional Hubbard model: numerical simulation study. *Phys. Rev. B*, 31(1985), 4403-4419.
[28] J. E. Hirsch and S. Tang. Antiferromagnetism in the two-dimensional Hubbard model, *Phys. Rev. Lett.* 62:591 - 594, (1989).
[29] I. E. Kaporin. High quality preconditioning of a general symmetric positive definite matrix based on its $u^tu + u^tr + r^tu$-decomposition. *Numer. Linear Algebra Appl.*, 5:483–509, 1998.
[30] T. A. Manteuffel. An incomplete factorization technique for positive definite linear systems. *Mathematics of Computation*, 34(150):473–497, 1980.
[31] J. Meijerinkand and H. A. van der Vorst. An iterative solution method for linear systems of which the coefficient matrix is a symmetric m-matrix. *Math. Comput.*, 31(137):134–155, 1977.
[32] R. A. Nicolaides. Deflation of conjugate gradients with application to boundary value problems. *SIAM J. Numer. Anal.*, 24:355–365, 1987.
[33] A. Padiy, O. Axelsson and B. Polman. Generalized augmented matrix preconditioning approach and its application to iterative solution of ill-conditioned algebraic systems. *SIAM J. Matrix Anal. Appl.*, 22:793–818, 2000.
[34] B. N. Parlett. *The Symmetric Eigenvalue Problem*. Prentice-Hall, Englewood Cliffs, N.J., 1980.
[35] B. Philippe, Y. Saad, and W. J. Stewart. Numerical methods in markov chain modeling. *Operations Research*, 40(6):1156–1179, 1992.
[36] J. Polonyi and H.W. Wyld. *Phys. Rev. Lett.* 51, 2257 (1983).
[37] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. Halsted Press, Div. of John Wiley & Sons, Inc., New York, 1992.

[38] Y. Saad. *Iterative Methods for Linear Systems*. PWS Publishing, Boston, 1996.
[39] D. J. Scalapino. *Phys. reports* **250**, 329 (1995).
[40] D. J. Scalapino. Proceedings of the International School of physics, edited by R. R. Rroglia and R. R. schrieffer, North-Holland, New York, 1994
[41] R. T. Scalettar, D. J. Scalapino, R. L. Sugar and D. Toussaint. Hybrid molecular-dynamics algorithm for the numerical simulation of many-electron systems. *Phys. Rev. B*, 36(1987), 8632-8640.
[42] G. W. Stewart. Perturbation theory for rectangular matrix pencils. *Lin. Alg. Appl.*, 208/209:297–302, 1994.
[43] G. J. Tee. An application of p-cyclic matrices for solving periodic parabolic problems. *Numer. Math.*, 6:142–159, 1964.
[44] M. Tismenetsky. A new preconditioning technique for solving large sparse linear systems. *Lin. Alg. Appl.*, 154–156:331–353, 1991.
[45] J.M. Tranquada, H. Woo, and T.G. Perring et al. *Nature* 429, 534, (2004).
[46] L. N. Trefethen and D. III Bau. *Numerical Linear Algebra*. SIAM, Philadelphia, 1997.
[47] U. M. Ascher, R. M. M. Mattheij and R. D. Russell. *Numerical solution of boundary value problems for ordinary differential equations*. Prentice-Hall, Englewood Cliffs, 1988.
[48] H. van der Vorst. *Iterative Krylov Methods for Large Linear Systems*. Cambridge University, Cambridge, 2003.
[49] R. S. Varga. *Matrix iterative analysis*, 2nd Edition. Springer, Berlin/ Heidelberg, 2000.
[50] D. Vollhardt. *Correlated electron systems,* edited by V. J. Emery, World Scientific, Singapore, 57 (1993).
[51] S. R. White and D. J. Scalapino. Density matrix renormalization group study of the striped phase in the 2D $t - J$ model. *Phys. Rev. Lett.* 80, 1272 - 1275 (1998).
[52] S. J. Wright. Stable parallel algorithms for two-point boundary value problems. *SIAM J. Sci. Statist. Comput.*, 13(1):742–764, 1992.
[53] S. J. Wright. A collection of problems for which Gaussian elimination with partial pivoting is unstable. *SIAM J. Sci. Statist. comput.*, 14(1):231–238, 1993.