New algorithm to enable 400+ TFlop/s sustained performance in simulations of disorder effects in high-T_c superconductors

G. Alvarez*, M. S. Summers*, D. E. Maxwell*, M. Eisenbach*, J. S. Meredith*, J. M. Larkin†, J. Levesque†, T. A. Maier*, P. R. C. Kent*, E. F. D'Azevedo*, and T. C. Schulthess*
*Oak Ridge National Laboraotry, Oak Ridge TN 37831-6164
†Cray Incorporated, Oak Ridge TN 37831-6008

***Abstract*—Staggering computational and algorithmic advances in recent years now make possible systematic Quantum Monte Carlo (QMC) simulations of high temperature (high-Tc) superconductivity in a microscopic model, the two dimensional (2D) Hubbard model, with parameters relevant to the cuprate materials. Here we report the algorithmic and computational advances that enable us to study the effect of disorder and nano-scale inhomogeneities on the pair-formation and the superconducting transition temperature necessary to understand real materials. The simulation code is written with a generic and extensible approach and is tuned to perform well at scale. Significant algorithmic improvements have been made to make effective use of current supercomputing architectures. By implementing delayed Monte Carlo updates and a mixed single-/double precision mode, we are able to dramatically increase the efficiency of the code. On the Cray XT4 systems of the Oak Ridge National Laboratory (ORNL), for example, we currently run production jobs on 31 thousand processors and thereby routinely achieve a sustained performance that exceeds 200 TFlop/s. On a system with 49 thousand processors we achieved a sustained performance of 409 TFlop/s. We present a study of how random disorder in the effective Coulomb interaction strength affects the superconducting transition temperature in the Hubbard model.**

CONTENTS

I Introduction 1

II Overview of the DCA/QMC method 3
- II-A Green functions of many-body theory and the Hubbard model with disorder . 3
- II-B The Dynamical Cluster Approximation 3
- II-C A new Hirsch-Fye QMC cluster solver algorithm with delayed updates 4

III Simulation results: disorder effect on the superconducting transition temperature T_c 5

IV Implementation and performance at scale 6
- IV-A Accelerating time to solution: mixed precision and optimal delayed updates 6
- IV-B Parallel efficiency 7
- IV-C Sustained performance 8

V Summary and outlook 8

VI Acknowledgment 9

I. INTRODUCTION

Superconductors are materials that conduct electricity with zero resistance below a certain critical temperature T_c, and, at the same time, show perfect diamagnetic behavior, i.e., perfectly repel magnetic fields. Given the intricacies of the current electric grid where electricity is transmitted at tens or hundred of kilovolts to avoid unacceptable energy losses, as well as the desperate need for new low-power paradigms in microelectronics, it is difficult to overstate the potential impact of a major breakthrough in superconductivity research. The critical temperature in conventional superconductors is a few tens of degree Kelvin (or less) and electronic applications of these superconductors require cooling with liquid helium. Expectations rose very high some two decades ago, with the discovery of superconductivity at transition temperatures as high as 150K in the cuprates – only liquid nitrogen was required to cool the materials below T_c. But despite intense experimental and theoretical studies, cuprates have not found widespread use in technology as was expected at the time of their discovery. The highest known transition temperatures are still far below room temperature, and other material limitations such as poor ductility or naturally occurring grain boundaries that limit the critical currents have impaired the usefulness of the cuprates. Unlike other classes of materials that are well understood theoretically and where a systematic knowledge driven materials design and development has led to technology revolutions (e.g., semiconductors in microelectronics or magnetic heterostructures in spintronics), there is presently no generally accepted theory of the origin of the pairing interaction in the superconducting state or the anomalous normal state properties of the cuprates.

The essential features of the cuprates were recognized soon after their discovery (3) in 1986 and summarized in a paper by Anderson in 1987 (1): (1) the structural units are the two-dimensional CuO_2 planes (Fig. 1a), and (2) the superconductivity is created by doping these planes that would otherwise be a Mott insulator with charge carriers (electrons or holes). Zhang and Rice pointed out (20) that due to the strong on-site coulomb repulsion, these carriers form a localized singlet state consisting of the Cu $d_{x^2-y^2}$ orbital hybridized with the respective p_x and p_y orbitals of the neighboring O atoms (Fig. 1b). These considerations lead to a remarkably simple model description, the two-dimensional (2D) Hubbard Model, in which the carriers can hop between their nearest

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SC2008 November 2008, Austin, Texas, USA 978-1-4244-2835-9/08 $25.00 ©2008 IEEE

neighbor sites on a square lattice with transition amplitude t and interact through an onsite Coulomb repulsion U (Fig. 1c). Due to the strength of the Coulomb repulsion (U $\sim 5-10$ eV $\gtrsim 8t$), the sites will be mostly singly occupied, hence leading to the formation of atomic moments due to the spin 1/2 character of the carriers. Furthermore, because of the Pauli exclusion principle for Fermions, which precludes double occupation of a site with two carriers of the same spin orientation, the magnetic moments on neighboring sites will tend to order antiferromagnetically in order to minimize the kinetic energy. At the doping level where superconductivity is seen experimentally, this antiferromagnetic arrangement is frustrated and no long-range order exists. However, neutron scattering experiments clearly indicate that antiferromagnetic fluctuations exist over a length scale of a few nanometers. It has been suggested that these antiferromagnetic fluctuations play a key role in the formations of the Cooper pairs for the superconducting state (4; 17; 16). Superconductivity itself is a macroscopic effect in which order 10^{23} paired carriers form a coherent quantum state that can be observed in wires that reach hundreds of meters in length.

The inherent multi-scale nature of the problem – atomic scale coulomb repulsion and moment formation, nano-scale antiferromagnetic fluctuations (and probably other nano-scale properties of the materials discussed below), and macroscopic quantum coherence – has made the solution of the 2D Hubbard Model for the cuprates such a difficult endeavor. Over two decades, many thousands of papers have been published on the subject, but only recently, with the advent of multi-teraflop/s computers and new computational algorithms that can systematically tackle the multi scale problem, we see numerical studies that can answer fundamental questions about the pairing mechanism and high temperature superconductivity.

The first simulations to show evidence that the 2D Hubbard model with parameters t and U that are representative of the cuprates do describe high-temperature superconductivity (15), were done on the Cray X1E system of the NCCS in 2005. The simulations employed a quantum cluster method known as the Dynamical Cluster Approximation (DCA) (7; 10) in conjunction with the Hirsch-Fye QMC (HF-QMC) algorithm (8). The approach to the multi-scale problem in these simulations is to solve the quantum many-body problem at the atomic and nano-scale exactly with QMC, and to account for the macroscopic length scales within a mean-field approximation by self-consistently and coherently embedding the cluster solution into an effective medium. In the limit of infinitely large clusters, the DCA/QMC recovers the exact solution. On the NCCS Cray X1E with its 18 TFlop/s peak performance where the codes ran at about 50 percent efficiency, the computer power sufficed to simulate clusters of up to 36 atoms, reaching length scales in the QMC part of the simulation that are large enough to converge the computed values of T_c (15). Subsequent simulations with the same method performed a systematic analysis of the pairing mechanism in the Hubbard model (19; 11; 12; 13; 14) and confirmed in an unbiased way the earlier arguments that the pairing interaction is due to spin fluctuations.

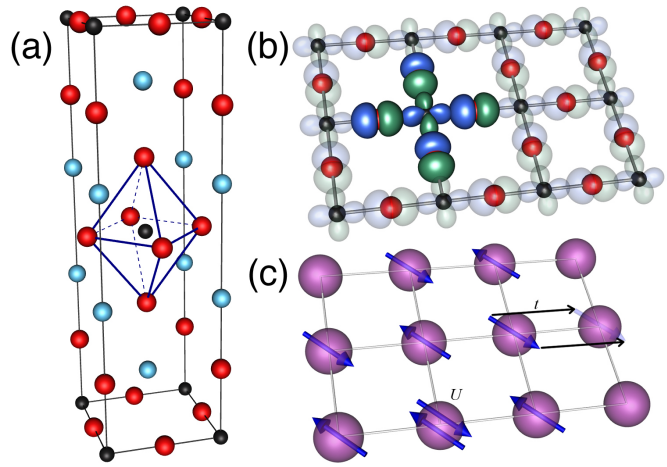

Fig. 1. (a) The crystal structure of La_2CuO_4, a typical cuprate, where black, red, and blue spheres represent Cu, O, and La, respectively. (b) The CuO_2 plane with outlines of the Cu $d_{x^2-y^2}$ and O p_x and p_y orbitals. Also shown in full color is the Zhang-Rice singlet state that forms from hybridization of the Cu orbitals with the neighboring O orbitals (see text). (c) Pictorial representation of the single band 2D Hubbard model with on- site Coulomb repulsion U and inter-site hopping t.

In the present paper we discuss the DCA++ code that implements the DCA/QMC and other quantum cluster methods in the framework of generic programming techniques and allows for effective extensions of the models. In particular, we focus on an extension of the model that will allow us to study models with random disorder or other types of nano-scale variations of the effective coulomb interaction. These extensions are motivated by recent experiments that have found spatial variations of T_c and even identified local regions in which the signatures of superconductivity (superconducting gap in the excitation spectrum) persisted to temperatures well above the macroscopically measured T_c (5; 18). With new and much larger supercomputers scheduled to come online in the coming year, a systematic study of the disorder effects on large clusters will be possible. In section 2 of this paper we present an overview of the DCA/QMC algorithm that should give the reader enough background to understand the generic implementation in the DCA++ code. We also discuss an algorithmic improvement that dramatically enhances the performance on scalar processors and allows us to perform simulations of large clusters on current Intel and AMD processors. In section 3 we discuss a series of simulation results generated with the DCA++ code and that unequivocally show a rather strong disorder dependence of the superconducting transition temperature. In section 4 we discuss implementation and performance characteristics of the production runs. On the Cray XT4 systems at ORNL with its 420TFlop/s peak performance, we have found the code to perform at a sustained double precision performance in excess of 200 Teraflop/s. When the QMC parts of the calculations are running in single

precision accuracy, we find that *without significant reduction in accuracy of the results*, the speed of the calculations can be increased by up to a factor 2 and the mixed precision code runs at 409 Teraflop/s equivalent sustained performance.

II. Overview of the DCA/QMC method

A comprehensive review of quantum cluster methods and the DCA can be found in a recent review article (10), and a detailed description of the DCA/QMC algorithm has been given in reference (9). We limit our discussion of the DCA/QMC method here to a brief overview.

A. Green functions of many-body theory and the Hubbard model with disorder

We are interested here in the 2D Hubbard model with atomic scale disorder in the Coulomb interaction parameter U. The Hamiltonian operator of a particular distribution ν of onsite interaction $\{U_i^{(\nu)}\}$ is conveniently written in second quantized notation

$$H^{(\nu)} = -\sum_{ij\sigma} t_{ij} c_{i\sigma}^\dagger c_{j\sigma} + \sum_i U_i^{(\nu)} n_{i\uparrow} n_{i\downarrow}, \qquad (1)$$

where the operator $c_{i\sigma}^\dagger$ ($c_{j\sigma}$) creates (annihilates) a particle with spin σ on site i, $t_{\rm ij}$ is the hopping amplitude between sites, and $n_{i\sigma} = c_{i\sigma}^\dagger c_{i\sigma}$ is the density operator. The first part of this Hamiltonian is usually refered to as the non-interacting part, $H_0 = -\sum_{{\rm ij}\,\sigma} t_{\rm ij} c_{i\sigma}^\dagger c_{j\sigma}$. The Hamiltonian operates on wave functions in Fock space, and in some problems of quantum chemistry and nuclear physics, one can attempt to solve directly for its eigenfunctions (2). This is not practical in condensed matter physics, where one usually considers systems with exorbitant numbers of electrons (quantum many particle problem), and one typically resorts to methods used in quantum field theory. One usually studies the single particle Green function of many-body theory, which may be written as

$$G_\sigma(x_i, \tau; x_j, \tau') = -\left\langle \mathcal{T} c_{i\sigma}(\tau) c_{j\sigma}^\dagger(\tau') \right\rangle, \qquad (2)$$

where the brackets mean a thermodynamic average over all states in Fock space, x_i is the position of site i, τ is the time variable, and $\mathcal{T}$ is the time ordering operator. Note that in the Hamiltonian, the hopping and interaction processes are assumed to be instantaneous. The Green's function is sometimes called a propagator, since it describes how a particle that is introduced into the system at time τ' on site j propagates in the presence of all other particles to site i where it is removed at time τ. The particular meaning of the time variable depends on the context in which Green's functions are used. We are using a statistical physics context where τ is an imaginary time variable that represents the inverse temperature of the system and the Green function only depends on one time variable due to symmetry. The Green's function of a non-interacting system is identical to the resolvent of the operator H_0, which, after taking the lattice Fourier transform, can be written as

$$G_0(k, z) = (z - \epsilon_0(k))^{-1}, \qquad (3)$$

where $\epsilon_0(k)$ is the spectrum of the lattice Fourier transformed non-interacting Hamiltonian $H_0(k)$.

The lattice Fourier transformed Green function of the interacting system, averaged over all disorder configurations, can formally be written as

$$G(k, z) = [z - \epsilon_0(k) - \Sigma(k, z)]^{-1}, \qquad (4)$$

where the self-energy $\Sigma(k, z)$ encapsulates all the quantum correlations of the many-body system (note that we have dropped the spin variable since $G_\uparrow = G_\downarrow$) and has to be computed in some approximate form. In the present case, the vectors k are two-dimensional and since $G(k, z)$ has the periodicity of the lattice, k can be limited to the first Brillouin zone or the unit cell of the reciprocal lattice. $G(k, z)$ is always an analytic function of the complex variable z in the upper and lower half plane and it is important that any approximation scheme to compute the self-energy preserves this property of the Green function.

B. The Dynamical Cluster Approximation

After this very brief introduction to the notation and essential quantities of quantum many-body theory, we are in a position to discuss how the self-energy is computed in the DCA. The basic idea is that quantum correlations only have to be treated explicitly within a cluster of N_c atoms, and that all correlations outside this cluster can be treated in a mean-field way, that is, the system outside the cluster is represented by an effective medium. In this approximation, the self-energy is fully described on the set of *coarse grained* k-points K, which correspond to the lattice Fourier transform of the cluster. There are N_c such k-points in the Brillouin zone and we introduce $M(k)$, a function that maps every k-point of the Brillouin zone onto the nearest coarse grained point K. Within the DCA, and assuming that the self-energy $\Sigma(K, z)$ can be somehow computed, the Green function can be written as

$$G(k, z) = [z - \epsilon_0(k) - \Sigma(M(k), z)]^{-1}. \qquad (5)$$

In practice, this function is represented on a discrete but dense uniform mesh of N k-points in the Brillouin zone. In the first iteration of the DCA/QMC algorithm, one usually starts out by setting Σ to zero or by using a self-energy that has been computed with a computationally inexpensive approximation (such as second order perturbation theory). One now proceeds by computing the Coarse Grained Green function

$$\bar{G}(K, z) = \frac{N_c}{N} \sum_{k=K+\tilde{k}} \left[z - \epsilon_0(K + \tilde{k}) - \Sigma(K, z) \right]^{-1}, \qquad (6)$$

and with it computes the cluster excluded Green function of the effective medium

$$G_0'(K, z) = \left[\bar{G}^{-1}(K, z) + \Sigma(K, z) \right]^{-1}. \qquad (7)$$

Note that this cluster excluded Green function is only equal to the non-interacting Green function $G_0(k, z)$ of the previous subsection in the limit of an infinite cluster size, i.e., when $N_c = N \to \infty$ and the self-energy is exact. However, if we

lattice Fourier transform $G_0'(K,z)$ into real space, the resulting real space cluster excluded Green function $G_0'(X_i - X_j, z)$ plays the role of the non-interacting real space Green function of the embedded cluster $[z - \tilde{H}_0(X_i, X_j; z)]^{-1}$. (Note that when we solve the quantum many-body problem on the cluster the $\{X_i\}$ denote the sites of the cluster and are therefore a subset of $\{x_i\}$). Here, $\tilde{H}_0(X_i, X_j; z)$ describes the dynamics of the electrons on the non-interacting cluster and their time-dependent excursions into the effective medium. The next step in the DCA algorithm is to use an adequate quantum cluster solver to determine the cluster Green function $G_c^{(\nu)}(X_i, X_j, z)$ for a particular configuration $\{U_i^{(\nu)}\}$ of on-site Coulomb repulsions. The configurationally averaged cluster Green function is given by

$$G_c(X_i - X_j, z) = \frac{1}{N_d} \sum_{\nu=1}^{N_d} G_c^{(\nu)}(X_i, X_j, z). \quad (8)$$

Note that because the cluster excluded Green's function is the lattice Fourier transform and since we average over all configurations (for example if we have n possible values of U per site we would have $N_d = n^{N_c}$ configurations $\{U_i^{(\nu)}\}$), the cluster excluded Green function will have periodic boundary conditions and can be lattice Fourier transformed into $G_c(K,z)$. The new estimate to the self energy is given by

$$\Sigma(K,z) = G_0^{-1}(K,z) - G_c^{-1}(K,z), \quad (9)$$

and can be used to compute a new cluster excluded Green function with equation (7). The DCA method is iterated until the self energy converges. With a converged self-energy, one can compute the Green function $G(k,z)$ of the system as well as many other quantities needed to analyze the solutions of the Hubbard model.

C. A new Hirsch-Fye QMC cluster solver algorithm with delayed updates

The quantum many-body problem that has to be solved on the cluster is given by the cluster excluded Green function. It describes the non-interacting part of the cluster and the coupling to the host lattice, as well as the interacting part of the Hubbard Hamiltonian

$$H_I^{(\nu)} = \sum_i U_i^{(\nu)} n_{i\uparrow} n_{i\downarrow} \quad (10)$$

for a particular set of local interaction parameters $\{U_i^{(\nu)}\}$. One can formulate the solution to the cluster problem in terms of Green functions, but a formulation of the problem in terms of a Hamiltonian is complicated by the fact that the non-interacting equivalent is only known in terms of $G_0'(X_i - X_j, z)$. For the present purpose, we have implemented a modified version of the HF-QMC algorithm (8), but would like to stress that the DCA will work with any many-body technique that can be cast into an action based approach. The modifications to the conventional HF-QMC algorithm are necessary to achieve high performance on most supercomputers today. A straightforward implementation is memory bandwidth limited with few FLOPs per memory access.

The HF-QMC algorithm employs a path integral formalism on a space-(imaginary)time lattice of size $N_t = N_c \times N_l$, where N_l represents the number of time slices (time steps) used in the path integral. The canonical Hubbard-Stratonovich (HS) transformation is used to replace the quadratic term in $n_{i\sigma}$ of the interaction with a linear term and an additional integral over an auxiliary degree of freedom (known as the HS field) at every point in space-time. Importance sampling is used to integrate over the HS field configurations in a Markov process. The acceptance criterion for the proposed new HS field configurations in the HF algorithm consists of a ratio of determinants of the Green function, and can be computed very efficiently. However, when a HS spin flip at position p_k is accepted, the $N_t \times N_t$ Green's function matrix at step k (matrix $\mathbf{G}_k$) has to be updated according to the rank one vector outer product update

$$\begin{aligned} \mathbf{G}_{k+1} &= \mathbf{G}_k + \alpha_k(\mathbf{G}_k(:,p_k) - \mathbf{e}_{p_k})\mathbf{G}_k(p_k,:) && (11)\\ &= \mathbf{G}_k + \mathbf{a}_k\mathbf{b}_k^t && (12)\\ &= \mathbf{G}_0 + \mathbf{a}_0\mathbf{b}_0^t + \mathbf{a}_1\mathbf{b}_1^t + \cdots + \mathbf{a}_k\mathbf{b}_k^t && (13) \end{aligned}$$

where e_{p_k} is the p_k-th column of the identity matrix, $\mathbf{a}_k = \alpha_k(\mathbf{G}_k(:,p_k) - \mathbf{e}_{p_k})$ is computed from a column of $\mathbf{G}_k$, $\mathbf{b}_k$ from a row of $\mathbf{G}_k$, and scalar α_k depends the diagonal entry

$$\alpha_k = \frac{\gamma_k}{1 + (1 - \mathbf{G}_k(p_k,p_k))\gamma_k} \quad (14)$$

with another scalar γ_k that does not depend on the Green function matrix. The transition probability R from state k to state $k+1$ is given by the ratio of determinants of the corresponding inverse Green's functions and for a single spin-flip update reduces to

$$R = \frac{\det(\mathbf{G}_k)}{\det(\mathbf{G}_{k+1})} = 1 + \gamma_k(1 - \mathbf{G}_k(p_k,p_k)). \quad (15)$$

Thus, R depends only on $\mathbf{G}_k(p_k,p_k)$, the diagonal entry of $\mathbf{G}_k$ at position p_k.

The usual computation of R is obtained by first updating the Green's function according to Eq. (11) (BLAS xGER operation) and then computing R using Eq. (15). This requires $\mathcal{O}(N_t^2)$ computations per step for the Green's function matrix update.

By realizing that only the determinant ratio R and hence a diagonal entry of $\mathbf{G}_k$ is needed to accept or reject a Monte Carlo move, it is possible to delay multiple rank-1 updates to perform a more efficient rank-k update to $\mathbf{G}_k$. The diagonal entries of $\mathbf{G}_k$ are updated separately in vector $\mathbf{d}_k$ to reduce the complexity of calculating the ratio R. At the k-step, row p_k and column p_k of $\mathbf{G}_k$ are regenerated as

$$\mathbf{col}_k = \mathbf{col}_0 + \sum_{i=0}^{k-1} \mathbf{a}_i \mathbf{b}_i(p_i) \quad (16)$$

$$\mathbf{row}_k = \mathbf{row}_0 + \sum_{i=0}^{k-1} \mathbf{a}_i(p_i) \mathbf{b}_i\,. \quad (17)$$

The diagonal entries of the Green's function matrix are then obtained from $\mathbf{a}_k = \alpha_k(\mathbf{col}_k - \mathbf{e}_{p_k})$, $\mathbf{b}_k = \mathbf{row}_k$ and

$$\mathbf{d}_{k+1}(p) = \mathbf{G}_{k+1}(p,p) = \mathbf{d}_k(p) + \mathbf{a}_k(p)\mathbf{b}_k(p), \text{ for } p = 1 : N_t. \tag{18}$$

The computational complexity of calculating the transition probability R in this delayed algorithm is thus reduced to $\mathcal{O}(kN_t)$ from $\mathcal{O}(N_t^2)$ if the updating of the Green's function is delayed by k steps. This also means that occasional Green's function matrix updates are required so that the complexity of the delayed algorithm does not exceed the complexity of the ordinary algorithm. The Green's function update is performed as a rank-k update (BLAS xGEMM matrix-matrix multiply operation) according to

$$\mathbf{G}_{k+1} = \mathbf{G}_0 + [\mathbf{a}_0|\mathbf{a}_1|\cdots|\mathbf{a}_k]\,[\mathbf{b}_0|\mathbf{b}_1|\cdots|\mathbf{b}_k]^t\ . \tag{19}$$

As in the original algorithm, this requires $\mathcal{O}(kN_t^2)$ operations. There is a small amount of redundant computation in updating the diagonal vector $\mathbf{d}$ that is not required in the non-delayed algorithm; however, the matrix-matrix multiply operation has many more FLOPs per memory access than the rank-one matrix update of the original algorithm, and therefore performs much better on most architectures (more details will be given in section IV-A and in Figure 5).

As is usual in Monte Carlo simulations, measurements of physical quantities such as the cluster Green function $G_c^{(\nu)}(X_i - X_j, z)$ or two-particle correlation functions, are performed along the Markov chain. Several update sweeps are performed between measurements to ensure that the measurements are fully decorrelated.

Algorithm 1 DCA/QMC Algorithm with QMC cluster solver (lines 5-10), disorder averaging (lines 4, 11-12), and DCA cluster mapping (line 3, 13)

```
1:  Set initial self-energy
2:  repeat
3:     Compute the coarse-grained Green Function
4:     for Every disorder configuration (in parallel) do
5:        Perform warm-up steps
6:        for Every Markov chain (in parallel) do
7:           Update auxiliary fields
8:           Measure Green Function and observables
9:        end for
10:       Accumulate measurements over Markov chains
11:    end for
12:    Accumulate measurements over disorder configurations.
13:    Re-compute the self-energy
14: until  self consistency is reached
```

The main parts in the simulation sequence of the HF-QMC can be summarized as follows: during the "warm-up" or thermalization phase of the calculations, only Monte Carlo moves with updates of the Green function are performed until the HS field configurations are thermalized. Measurements should not be performed during this phase, as the auto-correlation time is very long and one would not measure independent samples. This phase typically lasts for approximately 50 sweeps though the space-time lattice. The "measurement" phase of the calculations consists of Monte Carlo moves and updates as well as measurements that are typically performed every two, four, or eight sweeps depending on the simulated temperature (larger numbers of sweeps are needed for lower temperatures as the auto-correlation time is longer when the temperature is reduced). The HF-QMC with delayed updates is executed for every disorder configuration. Averaging over disorder configurations leads to the cluster Green function of a given DCA iteration, from which the self-energy is recomputed. The DCA loop is iterated until the self-energy is converged (see depiction in Figure 2 and algorithm template 1).

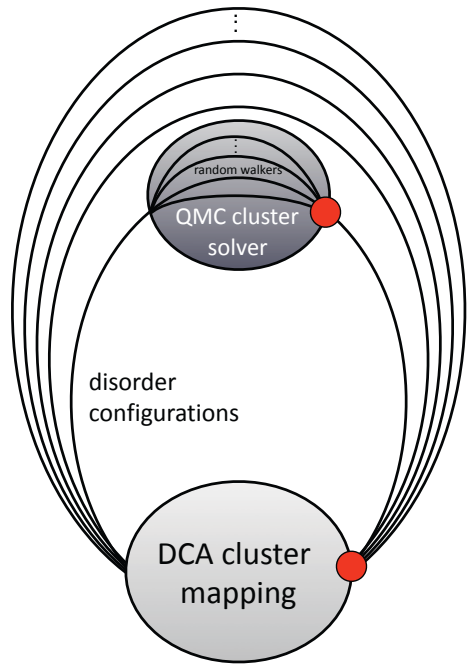

Fig. 2. The DCA self-consistency loop that has to be iterated consists of two essential parts. The Cluster mapping in which the new self-energy (using Eq. (9) for all but the first iteration), the coarse grained Green function in Eq. (6) and the cluster excluded Green function in Eq. (7) are computed. In the present case, the quantum cluster solver implements the HF-QMC algorithm with delayed updates (section II-C). Between these two steps the Green's functions are lattice Fourier transformed. The top level parallelization is over disorder configurations. For every disorder configuration we run one cluster solver that itself is parallelized over Markov chains. The two red circles indicate points where communications occurs. At the end of every cluster solver, the measured Green's function and charge susceptibility have to be accumulated for every disorder configuration. At the beginning of every cluster mapping step, the Green's function and charge susceptibility are averaged over all configurations.

III. Simulation results: disorder effect on the superconducting transition temperature T_c

In the present simulations we are interested in a transition to a superconducting state with d-wave symmetry. The order

parameter that signals a transition to this state is given by

$$\Delta_d^\dagger = \sum_{\mathbf{k}} g(\mathbf{k}) c^\dagger_{\mathbf{k},\uparrow} c^\dagger_{-\mathbf{k},\downarrow} \,, \tag{20}$$

where $g(\mathbf{k}) = \cos \mathbf{k}_x - \cos \mathbf{k}_y$ is a d-wave form-factor. In linear response theory, the pair-field susceptibility formed from this order parameter

$$P_d = \int_0^\beta d\tau \langle T_\tau \Delta_d(\tau) \Delta_d^\dagger(0) \rangle \tag{21}$$

diverges at the transition temperature T_c to the d-wave superconducting state. Using the Bethe-Salpeter equation, the pair-field susceptibility may also be written in terms of a particle-particle vertex function Γ^{pp}

$$P_d = P_d^0 + P_d^0 \Gamma^{pp} P_d \,. \tag{22}$$

Here, P_d^0 and Γ^{pp} are matrices of size $NL \times NL$, where N is the number of lattice sites and L the number of Matsubara-frequencies, P_d^0 is the pair-field susceptibility of the non-interacting part of the model and the right hand side contains an implicit sum over the matrix elements. In the DCA, the vertex function Γ^{pp} is approximated by its corresponding cluster quantity which is a smaller matrix of size $N_c L \times N_c L$. This matrix is calculated from the cluster pair-field susceptibility which is measured in the HF-QMC process in the last DCA iteration. Since Eq. (22) can be written as

$$P_d = \frac{P_d^0}{1 - \Gamma^{pp} P_d^0} \,, \tag{23}$$

one can conveniently determine instabilities by calculating the eigenvalues and eigenvectors of the pairing matrix $\Gamma^{pp} P_d^0$, i.e., solving

$$-\frac{T}{N_c} \sum_{K'} \Gamma^{pp}(K, K') P_d^0(K') \phi_\alpha(K') = \lambda_\alpha \phi_\alpha(K) \,. \tag{24}$$

The susceptibility diverges when the leading eigenvalue λ_α crosses one which determines T_c. The symmetry of the ordered state is then given by the K dependence of the corresponding eigenvector $\phi_\alpha(K)$.

At low temperatures the leading eigenvector has $d_{x^2-y^2}$-symmetry. The leading eigenvalue is shown as a function of temperature in Fig. 3

In the system without disorder ($dU = 0$), the leading eigenvalue crosses one at the transition temperature $T_c^{\text{clean}} = 0.08t$ (11). In the disordered system, T_c is reduced, although the reduction is only significant for $dU = 0.5$. One sees that as the disorder strength is increased, T_c decreases. It therefore must be concluded that T_c is suppressed by disorder in the interaction strength. It will be interesting to perform an analysis of the pairing interaction similar to what was done for the homogeneous system (19; 11; 12) to determine the cause of the reduction of T_c with increasing disorder strength.

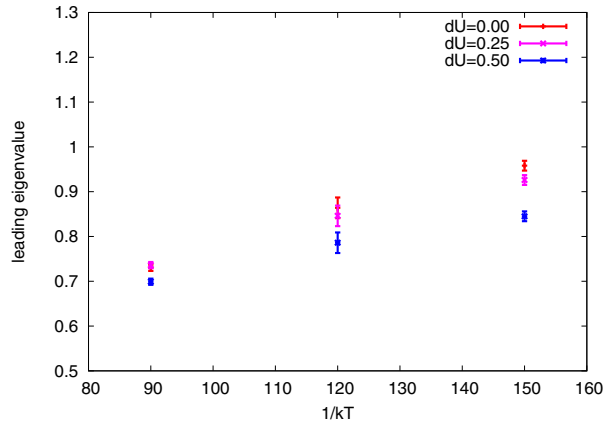

Fig. 3. The leading eigenvalue λ_d of the Bethe-Salpeter equation, Eq. (24), calculated on a 16-site cluster for $U_i = 4t(1 + \xi_i dU)$, where ξ_i is a random number with value ± 1 for three different values of disorder.

IV. Implementation and performance at scale

The DCA++ code implements the DCA/QMC algorithm described in section II with a generic and therefore extensible approach that builds on the programming model of the C++ Standard Template Library (STL). Since special attention has been paid to portability from workstations to clusters and high-end supercomputers, we are limiting the generic programming paradigms used to those of the STL (for example, we avoid using expression templates), and we are limiting the dependency on external libraries to well established packages that are needed for performance (BLAS/LAPACK), parallel programming (MPI and OpenMP), or numerical reliability (SPRNG). We limit the discussion here to the parts of the code that are critical for scaling and performance on multi-core architectures.

The main loop of the DCA++ code implements the DCA cycle depicted in Fig. 2, where most of the computing time in production calculations is spent in and around the cluster solver. However, much of the software complexity of the DCA++ implementation is in the coarse-graining procedure in the DCA cluster mapping part of the loop, since the code supports all 2D space-groups for models with arbitrary lattices.

A. Accelerating time to solution: mixed precision and optimal delayed updates

The cluster mapping part of the DCA loop has to be performed with double precision. In the QMC updates and measurements, however, it is possible to perform only single precision calculations. This is demonstrated in Fig. 4, where we show results for the critical temperature T_c, at which the leading eigenvalue λ_d of the Bethe-Salpeter equation, Eq. (24), assumes a value of 1 - according to the discussion of section III this is the temperature at which the transition to the superconducting state occurs. Shown are the results of a mixed precision run (coarse graining in double precision and QMC in single precision) as well as a set of runs in which all parts of the calculation has been done in double precision. The

length of the respective QMC runs were similar in terms of the total number of updates and measurements, and there seems to be no loss in precision of the final result of the calculations. However, the speed-up of the mixed precision runs over the double precision runs can be significant. We find the speed-up to be typically a factor 1.5 or more, with runs for larger clusters the acceleration can be as high as a factor of 2.

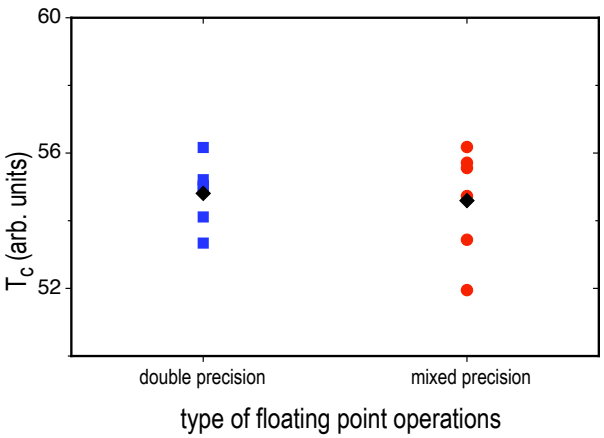

Fig. 4. Comparison of the critical temperatures calculated on a 4-site cluster for $U = 8t$ for the homogeneous system, for multiple runs done in double (red circles) and mixed precision (blue squares). The black diamonds indicate averages of the respective data sets.

In Figure 5 we show how the time to solution changes as a function of number of delays in the update of the Green function using the modified HF-QMC algorithm we discussed in section II-C. Compared to the original HF-QMC, which would correspond to a delay of 1, the acceleration is very substantial. At delays of about 56 and 64 for mixed and double precision runs, respectively, the additional overhead caused by the delayed updates starts to outweigh the benefits of increasing the squareness of the matrices in the s/dgemm calls of the update. For the remaining results of this section and the next, we use a delay of 56.

The ratio of double to single precision floating point operations in the mixed precision runs is about 1:1000 and the single precision part of the calculations is dominated by matrix multiplies in the QMC updates, where the matrix shape is $k \times N_t$ and the inner loop is over the shorter dimension k. We have timed the BLAS SGEMM and DGEMM calls for the type of runs presented in Figure 5, and find that SGEMM is a factor 2.02 faster than DGEMM. The best performance was achieved with K. Goto's implementation of BLAS (6) in which DGEMM runs at 6.2GFlop/s per core, which corresponds to about 74 percent of the theoretical peak performance. These numbers explain why the overall time to solution improvement of the mixed precision runs can be almost a factor 2 compared to the double precision runs.

B. Parallel efficiency

With the QMC cluster solver used in the present simulations, the great majority of time is spent updating the Green

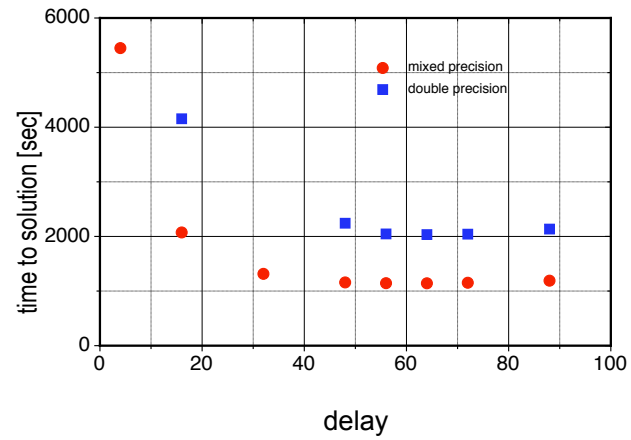

Fig. 5. Time to solution for different numbers of delays in the modified HF-QMC algorithm (see section II-C). The results are for a $N_c = 16$ cluster at inverse temperature $\beta = 150$ using 150 time slices ($N_t = 2400$). The calculations consisted of 4 parallel Markov chains running each on one core on a Cray XT4 with quad-core AMD Budapest processors running at 2.1 GHz.

function and performing measurements. Until self-consistency is reached in the self-energy, that is, typically in all but the last iteration of the DCA loop, only the averaged Green function and quantities such as the cluster charge susceptibility that are inexpensive to compute have to be measured. In this phase of the calculation, the performance is therefore dominated by DGEMM calls with matrices mentioned above. During the last iteration, when the self-energy is converged and one can measure meaningful physical quantities such as spin-spin correlation functions, the measurements can be quite involved and use about the same fraction of the run-time as the update sweeps between measurements. In order to compute the superconducting transition temperatures as in the previous section, we have to measure the pair-field susceptibility. Measuring this quantity, which is performed in reciprocal space, requires two lattice Fourier transforms that have been implemented in terms of a Kronecker product that requires $8 \times N_t$ complex matrix multiplies (using ZGEMM) per measurement - the typical matrix dimensions are (N_c by N_c), (N_c by N_l), and (N_l by N_l).

There are three natural levels of parallelism that have been implemented (see Figure 2 and algorithm template 1 for top two levels). At the top level we have the parallelization over disorder configurations, which is straightforwardly implemented with MPI. Note that for a typical cluster size of $N_c = 16$ sites and binary disorder, we have to average by sampling over 2^{16} configurations. The second level is within the HF-QMC cluster solver, where the measurements are spread over parallel Markov chains. At the end of every HF-QMC phase of the runs, we average the measured cluster Green function as well as other quantities over the parallel Markov chains, and subsequently we average over all disorder configurations before entering the coarse-graining phase of the DCA loop. The last HS field configuration is then used as a starting point in the cluster solver for the next DCA

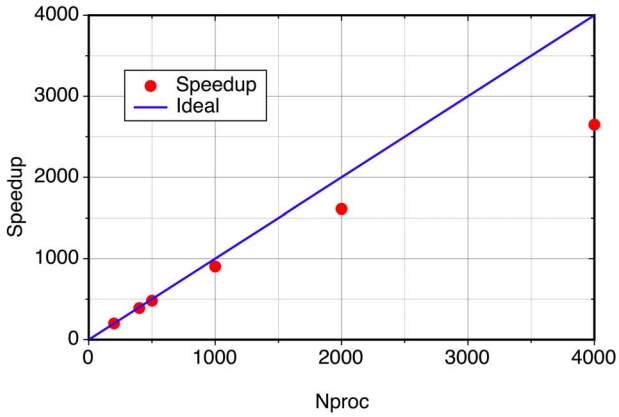

Fig. 6. Strong scaling results for DCA++ on Cray XT4 with dual core processors for a typical production run with one disorder configuration at moderate temperatures with 40,000 measurements and two sweeps between measurements, and where $N_c = 16$, $N_l = 80$.

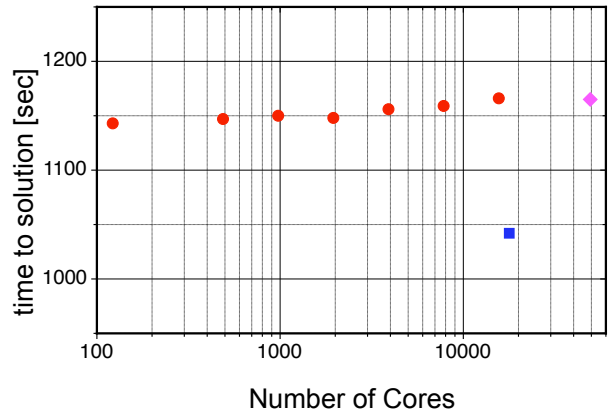

Fig. 7. Weak scaling results for the DCA++ on the Cray XT4 with quad core processors for typical productions runs at lower temperatures with $N_c = 16$, $N_l = 150$, 60 measurements per core, 4 sweeps between measurements, and 12 sweeps during the warmup phase in the QMC part of the calculation. The machine consists of 31,328 and 18,048 cores running at 2.1 GHz and 2.3 GHz, respectively. Red circles ran on the 2.1 GHz nodes, the blue square represents a job running on the 2.3 GHz nodes, and the magenta diamond represents a run on the combined nodes.

iteration, in order to shorten the auto-correlation time and therefore reduce the number of sweeps during the warm-up phase of the HF-QMC part of subsequent iterations. The thermalization phase is the part of the simulation that does not scale, since it has to be executed by every Markov chain. Following Amdahl's law, it is this part of the simulation that dominates the calculation when the second parallelization level is scaled to very large numbers of random walkers per disorder configurations. However, as is illustrated in Fig. 6, we can have as many as 1000 Markov chains without seeing significant deviations from optimal scaling. The third parallelization level in the DCA++ code is in the kernel of the Green function updates and measurements, and uses a shared-memory programming model with OpenMP. This level can be useful in studies of models without disorder that have to be scaled to thousands of processors. This situation, however, does not apply to our studies of cuprates, where the availability of systems with several hundred TFlop/s will allow us to average over disorder configurations and hence allow us to study the effect of disorder and nano-scale inhomogeneities on the superconducting transition temperature.

In Fig. 7, we show the weak scaling behavior of the code when the number of disorder configurations is increased commensurately with the number of processors. We are using 122 Markov chains in parallel for every disorder configurations, in order to keep the QMC simulations at optimal efficiency. The largest run corresponds to 404 disorder configurations and used 49288 cores. It is seen that the DCA++ code scales rather well with the number of cores and almost perfectly with the clock frequency of the processors.

C. Sustained performance

The production runs for the results we reported in figure 3 and the scaling benchmarks were run with the same basic setups. The benchmark runs were scaled down to run in less than 30 minutes, whereas the results in figure 3 represent runs that took between 4 and 16 hours on 31,000 cores. In order to estimate the executed floating point operations of the benchmark runs, we have instrumented the DCA++ code with PAPI calls. The average number of floating point operations per second reported in table I is computed from the total number of retired floating point operations as reported by PAPI_FP_OPS events divided by the measured execution time of the runs. The run time has been measured with PAPI calls to PAPI_get_real_usec as well as calls to the C time routine.

The sustained performance of these benchmark runs are summarized in Table I. A few points are worth noting besides the highest attained peak performance of 408.9 TFLOP/s in the mixed precision runs on 49,288 cores: (1) there is a noticable drop in performance going from a serial run (1 core) to a 4 core run, because the four core share memory bandwidth; (2) There is about a five percent drop in going from four cores on one processor to more cores on multiple processors because communication between MPI tasks runs via the Hypertransport link in the former and via the Seastar 2 interconnect in the latter case; (3) The performance per core on the runs with mixed frequency run is similar to the performance of the slower nodes as can be expected from the communication model we use.

V. Summary and Outlook

Despite nearly two decades of intense research, application of high transition temperature superconductor still fall short of their potential to contributed to a revolution in grid technology as well as microelectronics. However, recent advances in simulation capabilities, both from a method and computer performance point of view, open the door to solving long-standing theoretical questions about the cuprates. Here we have presented the DCA++ code and simulations of the disorder dependence of the superconducting transition temperature in the 2D Hubbard model. Simulation results on 16

TABLE I
SUSTAINED PERFORMANCE OF THE DCA++ CODE DURING SELF-CONSISTEN ITERATIONS OF THE DCA LOOP. THE FIRST TWO ROWS REPRESENT DOUBLE PRECISION RUNS AND THE REMAINING ROWS ARE FOR MIXED PRECISION RUNS. ALL RUNS WITH 122 OR MORE CORES USED 122 RANDOM WALKER IN THE QUANTUM CLUSTER SOLVER, FOR ALL OTHERS THE NUMBER OF WALKERS WAS THE SAME AS THE NUMBER OF CORES.

disorder configurations	total cores	frequency of core	GFlop/s per core	TFlop/s total
1	122	2.1 GHz	4.73	0.5771
404	49,288	mixed	4.105	202.3
1	1	2.1 GHz	11.4	0.01140
1	4	2.1 GHz	8.87	0.03549
1	8	2.1 GHz	8.44	67.51
1	122	2.1 GHz	8.46	1.032
4	488	2.1 GHz	8.43	4.112
16	1,952	2.1 GHz	8.42	16.43
64	7,808	2.1 GHz	8.34	65.13
128	15,616	2.1 GHz	8.29	129.5
146	17,812	2.3 GHz	9.28	165.3
404	49,288	mixed	8.30	408.9

site clusters show strong dependence of the superconducting transition temperature on disorder. The DCA++ code provides a modern implementation of quantum cluster methods applied to models of strongly correlated systems. Our implementation of the DCA/HF-QMC method applied to the 2D-Hubbard model is focused on fastest time to solution and runs efficiently at scale on the largest supercomputers currently available - reaching 409 TFlop/s sustained performance on the Cray XT4 systems at ORNL. This efficiency was achieved through a combination of algorithmic adaptation, in which we replaced a rank-one matrix update by a slender regular rank-k matrix multiply by delaying the QMC Green's function updates, and a mixed single-precision/double-precision computation. With availability of supercomputers with many hundreds of TFlop/s as well as the petascale systems that are coming online in the near future, systematic studies of disorder effects as well as the effect of nano-scale inhomogeneities and confinement on the superconducting transition temperatures will be possible. These simulations will provide an invaluable tool to understand spectacular recent experiments (18), which indicate that the superconducting transition temperatures could be much higher than what is currently observe macroscopically.

VI. ACKNOWLEDGMENT

We acknowledge use of resources of the National Center for Computational Sciences and the Center for Nanophase Materials Sciences, which are sponsored by the respective facilities divisions of the offices of Advanced Scientific Computing Research (ASCR) and Basic Energy Sciences (BES) of the U.S. Department of Energy. We thank the National Institute for Computational Sciences of the University of Tennessee, which is funded by the National Science Foundation, for allowing us to use their resource for some of the benchmark runs. The present research was supported by the Laboratory Directed Research and Development fund of ORNL, as well BES and ASCR.

REFERENCES

[1] ANDERSON, P. W. The resonating valence bond state in la_2cuo_4 and superconductivity. *Science 235* (1987), 1196.

[2] BARTLETT, R. J., AND MUSIAL, M. Coupled-cluster theory in quantum chemistry. *Rev. Mod. Phys. 79* (2007), 291–352.

[3] BEDNORZ, J., AND MULLER, K. Possible high-tc superconductivity in the ba-la-cu-o system. *Zeitschrift Fur Physik B-Condensed Matter 64* (1986), 189–193.

[4] BERK, N., AND SCHRIEFFER, J. Effect of ferromagnetic spin correlations on superconductivity. *PRL 17* (1966), 433.

[5] GOMES, K. K., PASUPATHY, A. N., PUSHP, A., ONO, S., ANDO, Y., AND YAZDANI, A. *Nature 447*.

[6] GOTO, K., AND VAN DE GEIJN, R. High performance implementation of the level-3 BLAS. *ACM Transactions on Mathematical Software 35*, 1 (July 2008), 4. Article 4, 14 pages.

[7] HETTLER, M., TAHVILDAR-ZADEH, A., JARRELL, M., PRUSCHKE, T., AND KRISHNAMURTHY, H. Nonlocal dynamical correlations of strongly interacting electron systems. *Phys. Rev. B 58* (1998), R7475.

[8] HIRSCH, J., AND FYE, R. Monte carlo method for magnetic impurities in metals. *Phys. Rev. Lett. 56* (1986), 2521.

[9] JARRELL, M., MAIER, T., HUSCROFT, C., AND MOUKOURI, S. A quantum monte carlo algorithm for non-local corrections to the dynamical mean-field approximation. *Phys. Rev. B 64* (2001), 195130.

[10] MAIER, T., JARRELL, M., PRUSCHKE, T., AND HETTLER, M. H. Quantum cluster theories. *Rev. Mod. Phys. 77* (2005), 1027.

[11] MAIER, T., JARRELL, M., AND SCALAPINO, D. Pairing interaction in the two-dimensional hubbard model studied with a dynamic cluster quantum monte carlo approximation. *Phys. Rev. B 74* (2006), 094513.

[12] MAIER, T., JARRELL, M., AND SCALAPINO, D. Spin susceptibility representation of the pairing interaction for the two-dimensional hubbard model. *Phys. Rev. B 75* (2007), 134519.

[13] MAIER, T., MACRIDIN, A., JARRELL, M., AND SCALAPINO, D. Systematic analysis of a spin susceptibility representation of the pairing interaction in the 2d hubbard model. *Phys. Rev. B 76* (2007), 144516.

[14] MAIER, T., POILBLANC, D., AND SCALAPINO, D. Dynamics of the pairing interaction in the hubbard and t-j models of high-temperature superconductors. *Phys. Rev. Lett. 100* (2008), 237001.

[15] MAIER, T. A., JARRELL, M., SCHULTHESS, T. C., KENT, P. R. C., AND WHITE, J. B. Systematic study of d-wave superconductivity in the 2d repulsive hubbard model. *Phys. Rev. Lett. 95* (December 2005), 237001.

[16] MORIYA, T., AND UEDA, K. *Adv. in Phys. 49* (2000), 555.

[17] SCALAPINO, D. The case for $d_{x^2-y^2}$ pairing in the cuprate superconductors. *Phys. Rep. 250* (1995), 329.

[18] SONIER, J., ILTON, M., PACRADOUNI, V., KAISER, C., SABOK-SAYR, S., ANDO, Y., KOMIYA, S., HARDY, W., BONN, D., LIANG, R., AND ATKINSON, W. Inhomogeneous magnetic-field response in $\mathrm{yba_2cu_3o}_y$ and $\mathrm{la}_{2-x}\mathrm{sr}_x\mathrm{cuo}_4$ persisting far above the bulk superconducting transition tc. arxiv.org/abs/0801.3481, 2008.

[19] T.A.MAIER, JARRELL, M., AND SCALAPINO, D. Structure of the pairing interactions in the two-dimensional hubbard model. *Phys. Rev. Lett. 96* (February 2006), 047005.

[20] ZHANG, F., AND RICE, T. Effective hamiltonian for the superconducting cu oxides. *Phys. Rev. B 37* (1988), R3759.