Transient Response of Strongly Correlated Materials to Large Electric Fields: Utilizing the Large Memory Capacity of ARSC's Midnight Machine in a Capability Applications Project

J.K. Freericks
Department of Physics, The Georgetown University, Washington, DC
freericks@physics.georgetown.edu

Abstract

Strongly correlated materials can be tuned to pass through a metal-insulator transition as a function of doping, pressure, or temperature. This high tunability makes them strong candidates for use in so-called "smart materials" that exhibit tunability of their properties. In this Capabilities Application Project (CAP) Phase II project we used the Arctic Region Supercomputing Center (ARSC) Opteron computer (Midnight) to exactly solve for the nonlinear response of these materials to a strong electric field. Our computational algorithm breaks into two portions, one serial and one parallel; the serial part does not scale while the parallel part scales perfectly with the number of compute nodes. The code is transportable having been run on a wide variety of machines with high efficiency and requiring only FORTRAN, LAPACK and BLAS. The main bottlenecks that determine how large a system can be simulated stem from the memory per node on the machine, and the total computational time available. We found that the performance on Midnight was excellent for large jobs (where the executable size was larger than 4GB) due to the fact that the job could be run with shared memory on one board and fewer processors. Doing so required simply using the proper calling protocol within Message Passing Interface (MPI) to exclusively share memory within the same motherboard. Midnight also showed improved performance over other machines in the 2GB–4GB range for the executable because each processor can use up to 4Gb memory without any sharing, and many other machines require sharing of memory when larger than 2GB.

1. Introduction

In this work, we examine how a strongly correlated material responds to the turning on of a large electric field. Ohm's law[1] describes the steady-state response of materials in the linear regime, where the current is proportional to the field (the coefficient being the conductivity); here we are interested in what happens as the field is made larger—so large that the linear-response approach breaks down—and we are interested in understanding the short-time transient behavior before the system reaches its steady-state response. The underlying theory and formalism for treating this nonequilibrium quantum statistical mechanics problem was developed over 40 years ago[2,3]. Modern computers now allow these systems to be solved exactly. In this work, we report on a CAP that utilized the ARSC Sun Opteron machine (Midnight) to solve these equations for a few very large cases. In total, about 600,000 CPU hours were used on the project over about a six week period.

The nonlinear response of a material is important because it is often used in devices to achieve specific goals, such as amplification of a signal, fast switching, sensing of external fields or chemical species, etc. As we move toward developing tunable electronics, we need to understand how the device properties can be tuned, and what effect that has on the nonlinear behavior. Strongly correlated materials may play an important role in such "smart electronics" devices, because they often can show high tunability of their properties, especially when they are close to a Mott metal-insulator transition. In addition, large fields are common across a device as electronics move into the nanoscale. When a feature size is on the order of 100 nm, a potential difference of 1 V produces an electric field of $E \sim 10^7$V/cm over the feature area. Such a large field will cause nonlinear effects. Finally, the military is interested in the robustness of electronics devices to large pulsed fields that can arise from natural sources like lightning, or from man-made sources like those employed in electronic warfare. These high energy-density short-time pulsed fields may be difficult to filter out of a device and can cause the device to "burn out". Such effects can also be treated within our transient response formalism, but we concentrate on a constant

HPCMP USERS GROUP CONFERENCE 2007 (HPCMP-UGC 2007)
0-7695-3088-5/07 $25.00 © 2007 IEEE

electric field turned on at a specific time in this initial phase of our work. Strongly correlated electron materials require a full quantum-mechanical treatment of the mutual electron-electron repulsion in determining how the material responds to external perturbations; in conventional materials, we can simply assume that an electron moves in the average background field generated by the other electrons. This average background approach is used in the so-called band-theory which has been highly successful in describing many real-materials properties. But as the electron-electron interactions are made stronger relative to the kinetic energy of the electrons, the electron correlations need to be treated properly; this is necessary in only a small subset of materials, but those materials often have interesting properties for use in devices. Hence electrons will not be treated in an average way, but instead we need to take into account where the electrons are and how they move as every other electron moves. This adds significant complexity to the quantum-mechanical treatment of the problem. It also allows new and exciting physical results, such as the existence of a metal-insulator transition as the interaction between the electrons grows; this transition is called the Mott-Hubbard transition and the insulating phase is a Mott insulator (to distinguish it from the ordinary band insulator). It occurs when there is on average one electron per lattice site and the electron-electron interaction is so strong that two electrons cannot sit on the same lattice site at the same time—then they crystallize into a frozen state where electron motion is not allowed and the material becomes an insulator.

The simplest model which takes into account strong electron-electron correlations is the Falicov-Kimball (FK) model[4]. This model has two kinds of electrons: itinerant electrons and localized electrons. They interact by a Coulomb repulsion only when they both occupy the same unit cell of the lattice. If the number of itinerant electrons plus the number of localized electrons is equal to the number of lattice sites, then the system will undergo a metal-insulator transition as the Coulomb repulsion is increased because the electrons become frozen at their lattice sites, unable to move because their motion involves a double occupancy.

2. Formalism and Numerical Algorithm

We consider the FK model in the presence of an external electric field that is spatially uniform, but time-dependent, and can have an arbitrarily large amplitude. Such a pure time-dependent electric field does not rigorously satisfy Maxwell's equations, since a time varying electric field cannot be spatially uniform and a time varying electric field generates a coupled time-varying magnetic field. But both of those effects are small if the time dependence is smooth or slow, and the corrections can be added later via the so-called gradient expansion, so we ignore them here.

The FK model has two kinds of electrons: itinerant electrons with creation and annihilation operators $c_i^\dagger$ and c_i for conduction electrons at site i and localized electrons with the corresponding operators $f_i^\dagger$ and f_i. The FK Hamiltonian is

$$\mathcal{H}(0) = -\sum_{ij} t_{ij} c_i^\dagger c_j + U \sum_i c_i^\dagger c_i f_i^\dagger f_i - \mu \sum_i c_i^\dagger c_i \, , \quad (1)$$

where t_{ij} is the nearest-neighbor hopping matrix, U is the on-site repulsion between c and f electrons, and μ is the chemical potential of the conduction electrons. The f-electrons are localized and do not move (hence they do not interact with the electric field). We choose them to occupy half of the lattice sites. We also choose the conduction electrons to occupy half of the lattice sites ($\mu = U/2$). This is called the case of half-filling, and it can lead to a metal-insulator transition if U is large enough, because the conduction electrons will then avoid the lattice sites occupied by the f-electrons, and since there will be no empty sites left over that allow them to move, the entire system will be "frozen" and cannot conduct electricity.

The uniform electric field $\mathbf{E}(t)$ is described by a vector potential $\mathbf{A}(t)$in the Hamiltonian gauge where the scalar potential vanishes:

$$\mathbf{E}(t) = -\frac{1}{c}\frac{\partial \mathbf{A}(t)}{\partial t} \, . \quad (2)$$

We assume that the vector potential $\mathbf{A}(t)$ is smooth enough, that we can neglect the magnetic field normally associated with the time-varying electric field.

The electric field is introduced into the Hamiltonian in Eq. (1) by the so-called Peierls' substitution[5,6]

$$t_{ij} \rightarrow t_{ij} \exp\left[-\frac{ie}{\hbar c}\mathbf{A}(t)\cdot(\mathbf{R}_j - \mathbf{R}_i)\right]; \quad (3)$$

interband transitions are neglected because there is only one band that couples to the electric field (the hopping matrix t_{ij} is nonzero only for nearest-neighbor sites i and j). The field-dependent Hamiltonian $\mathcal{H}(A)$ is then the field-free Hamiltonian, with the hopping term replaced by the Peierl's substituted hopping term.

We examine the case of a d-dimensional hypercubic lattice in the limit of large spatial dimensions $d \rightarrow \infty$ because the full quantum-mechanical many-body problem simplifies for this case. The electron self-energy becomes local (i.e., has no momentum dependence), which greatly simplifies both the formalism and the numerical calculations. This approach is called dynamical mean-field theory (DMFT)[7] and provides one of the known ways to exactly solve the many-body problem. The simplest electric field to consider (for computation) is one that lies along the unit cell diagonal[8]:

$$\mathbf{A}(t) = A(t)(1,1,\ldots,1)\,. \tag{4}$$

After the Peierls' substitution, the "band-structure" in the electric field becomes

$$\epsilon_{\mathbf{k}} = -2t\sum_{l}\cos\left[a\left(\mathbf{k}_l - \frac{e\mathbf{A}_l(t)}{\hbar c}\right)\right], \tag{5}$$

with a the lattice spacing which we will take to be one. With our choice for the electric field along the diagonal, the Peierl's substituted band structure becomes

$$\epsilon_{\mathbf{k}} = \cos\left(\frac{eA(t)}{\hbar c}\right)\varepsilon_{\mathbf{k}} + \sin\left(\frac{eA(t)}{\hbar c}\right)\bar{\varepsilon}_{\mathbf{k}}\,, \tag{6}$$

with generalized energy functions

$$\varepsilon_{\mathbf{k}} = -\frac{t^*}{\sqrt{d}}\sum_{l}\cos k_l \tag{7}$$

and

$$\bar{\varepsilon}_{\mathbf{k}} = -\frac{t^*}{\sqrt{d}}\sum_{l}\sin k_l\ . \tag{8}$$

and t^* being the renormalized hopping parameter: $t = t^*/2\sqrt{d}$ in the limit $d \to \infty$[7]; t^* will be used as our energy unit.

Important quantities, like the current flowing through the system, involve a summation over momenta of functions of ε and $\bar{\varepsilon}$. Since the momentum space is infinite-dimensional, it is not practical to perform an integration directly over the Brillouin zone. Instead, the summations are performed by calculating a joint density of states for the two energies in Eqs. (7 and 8) $\rho_2(\varepsilon,\bar{\varepsilon}) = \sum_{\mathbf{k}}\delta(\varepsilon-\varepsilon_{\mathbf{k}})\delta(\varepsilon-\bar{\varepsilon}_{\mathbf{k}})$; the joint density of states in the limit of the infinite dimensions[9] satisfies:

$$\rho_2(\varepsilon,\bar{\varepsilon}) = \frac{1}{\pi t^{*2}}\exp\left[-\frac{\varepsilon^2}{t^{*2}} - \frac{\bar{\epsilon}^2}{t^{*2}}\right].$$

So, the summation over the infinite-dimensional Brillouin zone is transformed into a two-dimensional (2D) Gaussian weighted integral. In our calculations, the most important integrals we need to evaluate are matrix-valued integrals, which we evaluate via discrete Gaussian quadrature in each of the two energy dimensions.

In this work, we focus our computational efforts on determining the so-called Green's functions. These objects measure how easy it is to create an electron at a particular time, and destroy it at a later time. In nonequilibrium problems, particularly those that include the transient response, these Green's functions depend on two time variables, because the system is not time-translation invariant due to the field being turned on at a particular time. We consider two Green's functions which determine all of the properties of the system: (i) the retarded Green's function, which measures how the quantum-mechanical states are distributed in energy, and (ii) the lesser Green's function, which measures the statistical occupancy of those quantum states as a function of time. Both Green's functions can be determined from the so-called contour-ordered Green's function, which is defined on the Kadanoff-Baym-Keldysh contour shown in Figure 1. The definition of the contour-ordered Green's function is

$$G^c_{ij}(t,t') = -\frac{i}{\mathcal{Z}}\operatorname{Tr} e^{-\beta\mathcal{H}(0)}\mathcal{T}^c_t c_i(t)c^\dagger_j(t')\,, \tag{9}$$

where $\mathcal{Z}$ = Tr exp $[-\beta\mathcal{H}(0)]$ is the initial equilibrium partition function, the Fermionic operators are expressed in the time-dependent Heisenberg picture (i.e., the temporal evolution is determined by $\mathcal{H}(A)$), the symbol $\mathcal{T}^c_t$ denotes time ordering along the Kadanoff-Baym-Keldysh contour, and β = 1/T is the inverse temperature of the system prior to the field being turned on. In this case, the lesser Green's function is found when t lies on the lower real branch and t' lies on the upper branch of the Kadanoff-Baym-Keldysh contour. The retarded Green's function can also be determined, but is somewhat more complicated to write out. The procedure to numerically calculate the Green's functions is complicated, and has been sketched in References 10–14. In this work, we describe in detail the numerical and computational aspects of our algorithm.

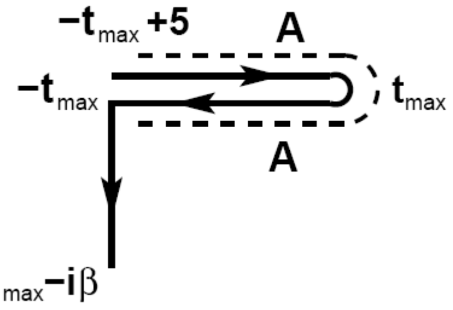

Figure 1. Kadanoff-Baym-Keldysh integration contour for the time variables. The time-domain cutoffs are symmetric at ±t_{max}. The direction for the integration of the line integral is indicated by the arrows. The dashed line schematically shows where we typically turn on the electric field, as represented by the vector potential; it is commonly turned on when the time is equal to −t_{max} + 5. Note that for the lesser functions, we choose the first time argument on the upper real time branch, and the second time argument on the lower real branch. When the contour is discretized, we use a step spacing of Δt along the real axis, and a step size of 0.1 along the imaginary axis. All calculations presented here have β = 10 corresponding to one hundred steps along the imaginary axis.

The dynamical mean-field theory equations are a set of nonlinear equations that must be solved self-consistently. The system of equations is solved by iteration, which turns out to be quite robust. We simply start from some initial guess for the self-energy (usually chosen to be the equilibrium self-energy), calculate the local Green's function by solving the field-dependent Dyson's equation, determine the effective medium by removing the self-energy from the local Green's function, solve an effective time-dependent impurity problem in the effective medium for the local Green's function and extract the local self-energy for the impurity. This procedure is then iterated until it converges. The most intensive piece of the iteration is the calculation of the local Green's function from the local self-energy. This is because it is calculated via an integral over the 2D joint DOS and the integrand is a matrix valued function that requires one matrix multiplication and one matrix inversion to determine it. Using approximately 100 integration points for each dimension requires 10,000 inversions and matrix multiplications for each iteration, and it naturally parallelizes and can be coded to scale essentially linearly with the number of processors.

The first numerical issue that we need to resolve is that we need to find a way to discretize the continuous matrix operators so they can be represented as discrete matrices and we can use linear algebra packages to perform operations on them. We do this by choosing a step size Δt on the top two branches of the contour, and a step size $\Delta\tau = 0.1$ on the imaginary axis. The imaginary branch always has a fixed step size of 0.1, while we vary the step size on the upper and lower branches from 0.1 to 0.0143. We typically take the time cutoff at $t_{\max} = 20$. The matrices are general complex matrices, which range in size from 900 × 900 up to 5,700 × 5,700. The computational time grows like the cube of the linear dimension of these matrices while the size of the executable grows like the square.

Next we need to efficiently evaluate the two-dimensional Gaussian-weighted matrix-valued integral. This is handled by using Gaussian integration along each of the dimensions. We take the average of the integral with 54 points and 55 points in each dimension. This results in $54 \times 54 + 55 \times 55 = 5{,}941$ quadrature points for each iteration (see Reference 10 for a discussion of the accuracy and limitations of the Gaussian integration scheme).

Finally, we use LAPACK and BLAS routines for the required matrix inversions and matrix multiply operations. The matrix inversions are the most time-consuming parts of the algorithm. We need 5,941 inversions to determine the local Green's function for each iteration, which is followed by six more inversions to solve the impurity problem, which completes the DMFT iteration; typically between 10 and 100 iterations are needed to achieve a self-consistent solution.

As can be quickly seen, this algorithm has a clear parallelizable part (the matrix inversions for each quadrature point), and a clear serial part (solving the impurity problem), since the inversions for the impurity-problem solver must all take place sequentially. Hence we expect the algorithm to be the combination of a fixed time plus a time that scales inversely with the number of processors. The algorithm is also easily parallelized in the master-slave format. When examining strong scaling, the serial part will not scale, but the parallel part will, if the number of slave nodes is chosen so that the slave nodes are all active most of the time.

Once the DMFT algorithm has been solved, then we can determine the properties of the system as a function of time. One interesting quantity is the current density that is driven by the external electric field:

$$\mathbf{j}_l\left(T\right) = -i\frac{et^*}{\sqrt{d}}\sum_{\mathbf{k}}\sin\left(\mathbf{k}_l - \frac{e\mathbf{A}_l\left(T\right)}{\hbar c}\right)G^<\left(\varepsilon_{\mathbf{k}},\bar{\varepsilon}_{\mathbf{k}},t,t\right), \quad (10)$$

expressed in terms of the momentum-dependent lesser Green's function, with each vector component identical when the electric field lies along the diagonal.

The current response of a noninteracting metal turns out to be oscillating even from a dc electric field. This occurs because the electron accelerates in the first Brillouin zone until it reaches the zone boundary, where it Bragg reflects and is moved to the opposite side of the Brillouin zone where it is accelerated again. This periodic motion creates an oscillating ac current with the period of the oscillation determined by the strength of the electric field. This phenomenon is called a Bloch oscillation[15–17], and it should be seen in any material that is free enough of defects and other sources of scattering. No conventional metal has ever been grown that has small enough scattering to exhibit Bloch oscillations. Instead, the scattering occurs so rapidly, that the steady-state current is a constant, which increases linearly with the electric field until nonlinear effects take over. Bloch oscillations have been seen in semiconducting heterostructures[18].

Bloch oscillations are also seen in DMFT[13], with a time-independent electric field (E constant and $A(t) = -Ect$ for $t < 0$)[8]

$$j\left(T\right) \sim \sin\left(\frac{eA\left(t\right)}{\hbar c}\right)\int d\varepsilon\frac{df\left(\varepsilon-\mu\right)}{d\varepsilon}\rho\left(\varepsilon\right), \quad (11)$$

producing an oscillating current density [$\rho(\varepsilon)$is the noninteracting density of states, which is equal to the integral of ρ_2 over $\bar{\varepsilon}$ and $f(\varepsilon) = 1/1 + \exp(\beta\varepsilon)$ is the Fermi-Dirac distribution]. The frequency of the oscillation is $\omega_{\text{Bloch}} = eE/\hbar$ and is called the Bloch oscillation frequency. These oscillations survive in the presence of scattering (but are damped) when the field is large enough that the relaxation time due to scattering is significantly larger than the Bloch oscillation period. They even survive in the Mott insulating phase, but they become irregular oscillations that are more sharply damped. The frequency of oscillation is undoubtedly too high for the Bloch oscillations to be directly observed $\omega_{\mathrm{Bloch}} > 10^{12}$ Hz for $E = 1$).

3. Scaling to Large Numbers of Processors for Large Jobs

This code was originally tested in a CAP Phase II project in 2006 on the Cray XT3 (Sapphire) and since then has been run on National Aeronautics and Space Administration's (NASA's) Columbia supercomputer as part of a National Leadership Computing System initiative. Originally, the code did not scale well past about 900 processors due to a many-to-one communications bottleneck, but that bottleneck was removed by using what we call a recursive binary gather operation[12], where each slave node accumulates results of the quadrature locally, then half send the results to the other half for accumulation and that is repeated until all the accumulated results lie on one slave node, which then transfers them to the master. The code has achieved a sustained performance in excess of 60% of the peak operating speed on the 2,048 subcluster of Columbia at about 8 Teraflops[19]. This is all done with conventional FORTRAN compilers, and LAPACK and BLAS libraries; no assembly language coding is needed.

In spite of these successes, there were challenges in running the code for larger matrices (the matrices used are general complex double precision matrices). Larger matrices are needed if one wants to go out further in time, or if one needs a smaller discretization time step Δt along the real axis. The latter is needed for the Mott-insulating cases, where small step sizes are critical to maintaining the overall precision of the calculation. On the original runs on Engineer Research and Development Center's (ERDC's) XT3 machine, the memory size per node was limited at 1GB. Since we need to store seven general complex matrices during the calculations, the executable became too large when the matrix size was larger than about 2,100 × 2,100. On NASA's Columbia machine, the memory per node is 2GB, and the machine is a shared memory machine, so larger size jobs can be run at the expense of using fewer processors. There, we were able to run 3,300 × 3,300 jobs without any sharing of memory, but larger jobs, which required shared memory began to slow down due to the need for the memory to be shared across different motherboards. ARSC's Midnight machine allows us to push the limit much further. Midnight is a SUN Opteron machine with two kinds of compute nodes: the Taurus nodes (X2200) are quad core processors with 16GB shared memory per node and the Galaxy nodes (X4600) are sixteen-fold core processors with 64GB shared memory per node. The throughput per core for this code is somewhat higher for the Taurus nodes than the Galaxy nodes. This architecture has allowed us to easily run 4,100 × 4,100 and 4,900 × 4,900 cases which both fit within the 4GB limit. We even were able to run two cases with 5,700 × 5,700 matrices which has not been attempted with this code on any other machine. In fact, we started to run into runtime limitations more than memory limitations on Midnight, i.e., the processor time was becoming too long for the jobs to be completed in the real time allowed for the CAP.

When we performed a scaling analysis of this code on the ERDC XT3 and the NASA Columbia machines, we saw excellent strong scaling results, where the compute time for the parallel part of the code decreased linearly with the number of processors, while the serial part remained the same. On the ARSC Opteron, however, we saw more run to run and processor number to processor number variations in the code when we performed a scaling analysis and we found the raw run time for smaller jobs was not as fast as on other machines (see Figure 2). This is because these codes allocate a fixed amount of memory per processor (rather than a fixed amount for a problem size). Therefore as the problem size increases, the per-processor memory requirements also increase, no matter how many processors are used, so machines with large memory "near" the processor, such as Midnight, will perform best on large problems. On Midnight, we were able to use the medium memory model and options to the PBS batch script to map three MPI tasks (instead of four) to a four-core x2200 node enabling us to run with more than 4GB of memory per core at the expense of leaving one core per node idle. (Only the 5,700 × 5,700 case, requiring 4.5GB per core, needed this technique.)

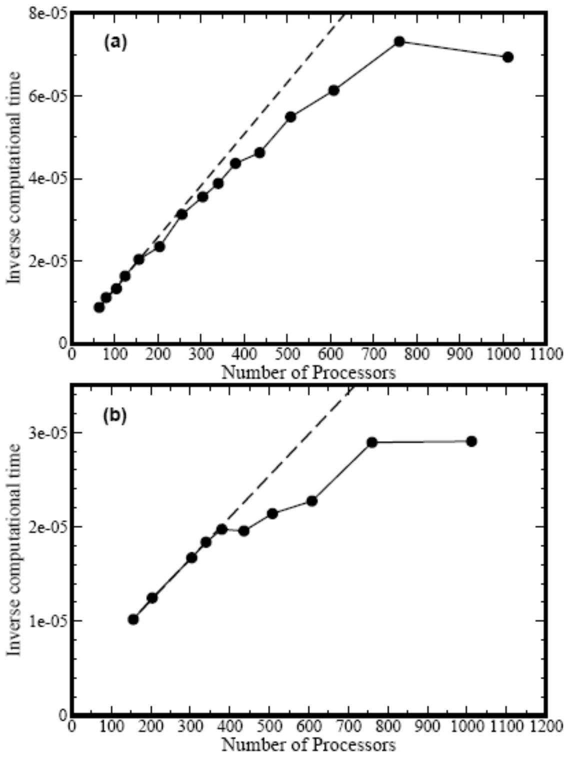

Figure 2. Scaling plot of the inverse of the total computational time as a function of the number of Taurus (four-way) processors used on Midnight. The problem size is 2,500 × 2,500 in panel (a) and 4,100 × 4,100 in panel (b). If we had linear scale up of the algorithm, the inverse computational time would scale linearly, as indicated by the dotted line. But because there is a serial part of the code, the scale-up is not as fast. Nevertheless, we achieve greater than 70% of linear scale-up with 760 processors. Note however, that there is significant scatter in the data. We were not able to trace the source of this run-to-run noise with large numbers of processors, or the poor performance in extending to 1,000 processors.

But when we went to the larger jobs, we began to find a significant increase in speed on Midnight due to the fact that for all jobs we could keep the executable within memory on a single motherboard by using the proper PBS options for mapping MPI tasks to processors to exclusively place the given number of processors on the particular nodes, and using the remaining memory as shared memory for the job (only the 5,700 × 5,700 case required using shared memory, as the executable size was about 4.5GB). An example of this speed up is shown in Table 1 which compares the speeds on Midnight and Columbia for large memory jobs. Note how the Altix performs better than both the XT3 and the Opteron for small jobs, primarily due to the larger total cache size (and the slightly larger Gflop rate of its processors), but because the XT3 is not a shared memory machine, once the executable becomes too large it cannot be run there. The Altix sees a significant slowdown once the machine needs to share memory outside of the single motherboard, but because the Opteron has so much more memory on a node, its computational time continues to grow almost as the cube of the matrix dimension, as expected. This illustrates the significant benefits of Midnight for large memory jobs.

Table 1. Total computational time for one iteration for different size problems on different supercomputers (in hours)

Job size	ARSC Opteron	NASA Altix	ERDC XT3
900 × 900	12.5	6	10
4,100 × 4,100	1,200	1,925	X
4,900 × 4,900	2,250	10,500	X
5,700 × 5,700	4,200	X	X

4. Results

The quantum-mechanical problem that we are solving requires us to determine results based on continuous matrix operators. Since such operators cannot be described on a computer, we discretize the operators into discrete matrices and perform calculations directly on the discrete matrices. In the end, however, we need to be able to take the limit where the discretization size goes to zero, otherwise our results will have quantitative errors in them. Hence, we need to scale our results to $\Delta t \rightarrow 0$. This is done by performing calculations at various different discretization step sizes and then fitting the results of a set of different step sizes via a Lagrangian interpolation formula and then extrapolating the interpolation formula to $\Delta t \rightarrow 0$. In order for this procedure to work robustly, we need to have good quality data that has a small enough step size that it is in the scaling regime. As the strength U of the interactions between particles increases, the needed step size to get into the scaling regime is reduced. As the strength of the field E is reduced, the needed step size is also reduced. Since calculations for a given time domain require more steps, and hence larger matrices, as the step size is reduced, the calculations can rapidly go beyond what is available in current computer resources. Indeed, this is typically the limiting factor we have in our calculations. Usually the limitation arises from the executable memory being too large, but now we are starting to see limitations occur due to the runtime being too long.

We have two ways to independently check the quality of the calculations and the scaling hypothesis. The first is by examining the current as a function of time. We know that the current vanishes for times before the time when the electric field is turned on, but the calculation for a given discretization step size always shows a nonzero current. We know the scaling hypothesis is working well when the scaled current is zero for those early times. The second involves examining the zeroth and second moment of the spectral function. These moments satisfy known exact sum rules. We can calculate them from our Green's functions by examining the equal time Green's function values for the zeroth moment and the equal time second derivative for the second moment. By comparing the exact results to the scaled results we also can gauge the quality of the scaled answers. The runs that we performed on Midnight were important to the cases of transport in Mott insulators, because those cases require small step sizes for good computational accuracy, and the size of the matrices needed can grow to be quite large.

An example of what we were able to achieve with the CAP work is shown in Figure 3. In this plot, we show the equal time retarded Green's function, which should equal 1 for all times. The raw data for different step sizes are plotted with different symbols and colors. Note how poor the large step size results are, indicative of the difficulty in obtaining good results for the Mott insulator (here we have E = 1 and U = 3). In fact, with data down to Δt = 0.02, any extrapolation produced results with unacceptably large errors. It is only after the last two data sets were included that one could achieve reasonably good scaled results.

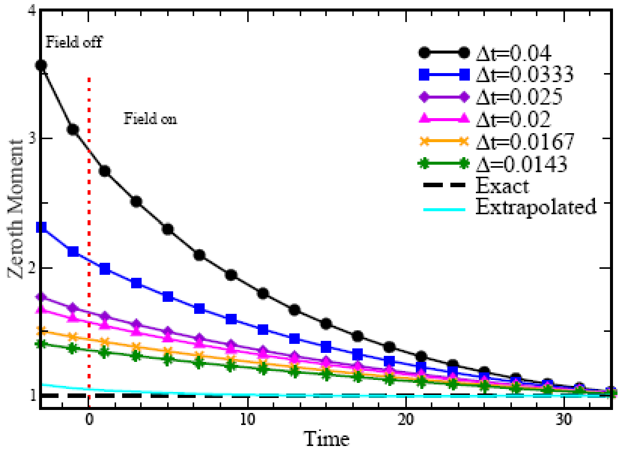

Figure 3. Zeroth moment as calculated for different step sizes and the extrapolated results for the Mott insulator with *E* =1 and *U* =3 (the field is turned on at time *t* =0). Previous to the CAP, we did not have the last two data sets. The results for larger step sizes did not extrapolate well, and we were left with errors on the order of 20–50%. After running the smaller step size cases, we were able to get into the scaling regime (the extrapolated results scaled quadratically with the last three data sets). In this case the average error is less than 5%.

We plot similar results for the curvature (or second moment) of the retarded Green's function in Figure 4. Here one can see a much more dramatic effect due to the higher quality data. When we include the smallest Δt data in the extrapolation, we achieve an error of less than 8% after the field is turned on, while the error would have been about 100% if we were stuck with just the Δt = 0.02data (where results could not be scaled).

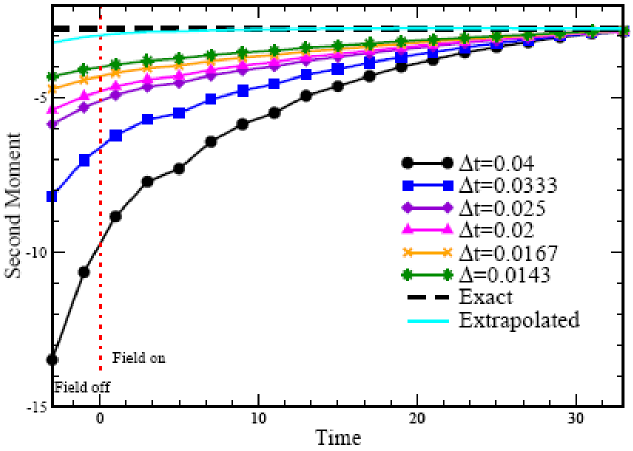

Figure 4. Plot of minus the second moment of the local retarded Green's function at different times (*E* =1 and *U* =3). Note that the electric field is turned on at the origin in this plot. The exact moment is equal to −2.75, and the curves with different symbols correspond to different values of Δt. The solid line is the extrapolated result using a quadratic Lagrange interpolation formula for the extrapolation from the last three data sets. One can see that as the step size is made smaller, the moment comes closer to its true value, and the extrapolated result is quite accurate (better than 8% on average).

These results show once again how some of the more extreme cases that we considered needed precisely the resources that are available with the ARSC Opteron cluster, which allows large memory jobs to be run efficiently.

Now that we have benchmarked our results against exact sum rules, we can show some similar results for the current. Figure 5 depicts the results for the current of a Mott insulator with E = 1 and U = 3. The data is for the same time steps as in the moment figures, and the cyan colored curve is the extrapolated current. Note how it essentially vanishes while the field is off as it must, while it develops irregular oscillations for larger U.

IEEE
COMPUTER
SOCIETY

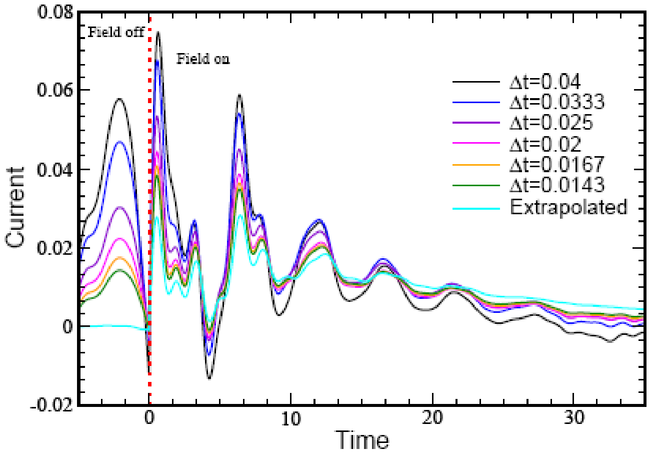

Figure 5. Current for different time steps with *E* = 1 and *U* = 3. Note how the raw data has significant current before the field is turned on, but after extrapolating, because the data does scale, we are able to achieve correct physical behavior, where the current vanishes while the field is off. The current displays quite irregular oscillations that last for a relatively long period of time.

In total we ran eleven cases of size 4,100 × 4,100, three cases of size 4,900 × 4,900, and two cases of size 5,700 × 5,700. While a few runs were completed on the Galaxy (sixteen-way) nodes, most work was completed on the Taurus (four-way) nodes. For the biggest jobs, we ran on 760 processors using additional memory from 252 processors, which essentially utilized all of the Taurus nodes of Midnight. Each of these jobs provided important results that either allowed us to go much further out in time than we had been able to do before or which allowed us to go to much smaller step size which moved us into the scaling regime and allowed our results to be scaled.

5. Conclusions

In this work, we presented a summary of a CAP phase II project on the ARSC Opteron cluster named Midnight. We ran a transportable and efficient code that solves for the nonequilibrium response of a strongly correlated material to a large electric field. This code performed extremely well on Midnight for the very large jobs that we needed to make runs for to be able to complete the study of a number of important cases. The large memory on each motherboard and the ability to use MPI to exclusively select processors on specific motherboards allowed us to be able to efficiently run the code, when it ran into severe memory limitations on other supercomputers. In this sense, the architecture of Midnight was well utilized for being able to complete these large and demanding computations.

From a science standpoint, the ability to solve and understand the nonequilibrium response of a Mott insulator is a long-standing problem that has now been solved. This work is providing the first new results in this area, but much more work is needed to be completed before a full understanding of the system can be achieved.

Acknowledgments

We acknowledge support of the Office of Naval Research under Grant Number N00014-05-1-0078 and from the National Science Foundation under Grant Number DMR-0210717. Supercomputer resources (Sun Opteron) were provided by ARSC under a CAP Phase II grant. In addition, useful discussions and collaborations with A. Joura, V. Turkowski, and V. Zlatić are acknowledged. Technical support with the ARSC Opteron by Ed Kornkven and Don Bahls is greatly appreciated.

References

1. Ohm, G.S., *The Galvonic current investigated mathematically*, J.G.F. Kniestädt, Berlin, 1827.

2. Kadanoff, L.P. and G. Baym, *Quantum Statistical Mechanics*, W.A. Benjamin, Inc., New York, 1962.

3. Keldysh, L.V., “Diagram technique for nonequilibrium processes.” *J. Exptl. Theoret. Phys.*, 47, pp. 1515–1527 1964 [*Sov. Phys. JETP*, 20, pp. 1018–1026, 1965].

4. Falicov, L.M. and J.C. Kimball, “Simple model for semiconductor-metal transitions: SmB_6 and transition-metal oxides.” *Phys. Rev. Lett.*, 22, pp.997–999, 1969.

5. Peierls, R.E., “Theory of diamagnetism of conduction electrons.” *Z. Phys.*, 80, pp. 763–791, 1933.

6. Jauho, A.P. and J.W. Wilkins, “Theory of high-electric-field quantum transport for electron-resonant impurity systems.” *Phys. Rev. B*, 29, pp 1919– 1938, 1984.

7. Metzner, W. and D. Vollhardt, “Correlated lattice fermions in $d = \infty$ Dimensions.” *Phys. Rev. Lett.*, 62, pp. 324–327, 1989.

8. Turkowski, V. and J. K Freericks,. “Nonlinear response of Bloch electrons in infinite dimensions”, *Phys. Rev. B*, 71, 085104-1–11, 2005.

9. Schmidt, P., “Time-dependent dynamical mean-field theory.” *Diplome thesis*, University of Bonn, 1999.

10. Freericks, J.K., V.M. Turkowski, and V. Zlatić, “Real-time formalism for studying the nonlinear response of ‘smart’ materials to an electric field.” In *Proceedings of the HPCMP Users Group Conference 2005*, Nashville, TN, June 28–30, 2005, edited by D.E. Post (IEEE Computer Society, Los Alamitos, CA, 2005), pp. 25–34.

11. Turkowski, V.M. and J.K. Freericks, “Spectral moment sum rules for strongly correlated electrons in time-dependent electric fields.” *Phys. Rev. B*, 73, 075108-1–15, 2006; Erratum, *Phys. Rev. B*, 73, 209902(E)-1–1, 2006.

12. Freericks, J.K., V.M. Turkowski, and V. Zlatić, “Nonlinear response of strongly correlated materials to large electric

fields.", In *Proceedings of the HPCMP Users Group Conference 2006,* Denver, CO, June 26–29, 2006, edited by D.E. Post (IEEE Compuer Society, Los Alamitos, CA, 2006), pp. 218–226.

13. Freericks, J.K., V.M. Turkowski, and V. Zlatić, "Nonequilibrium dynamical mean-field theory." *Phys. Rev. Lett.* , 97, 266408-1–4 2006.

14. Turkowski, V. and J.K. Freericks, "Nonequilibrium dynamical mean-field theory of strongly correlated electrons." To be published in *Strongly Correlated Systems: Coherence and Entanglement*, World Scientific, Singapore, 2007.

15. Bloch, F., "Quantum mechanics of electrons in crystals." *Z. Phys.*, 52, pp. 555–599, 1928.

16. Zener, C., "A theory of the electrical breakdown of solid dielectrics." *Proc. R. Soc. (London) Ser. A*, 145, pp. 523-529, 1934.

17. Aschcroft, N.W. and N. D. Mermin, *Solid state physics*, Holt, Rinehart, and Winston, Philadelphia, 1976.

18. Waschke, C., H.G. Roskos, R. Schwedler, K. Leo, H. Kurz, and K Köhler,. "Coherent submillimeter-wave emission from Bloch oscillations in a semiconductor superlattice." *Phys. Rev. Lett.* , 70, pp. 3319–3322, 1993.

19. Freericks, J.K., Y.-T. Chang, and J. Chang, "8.53 Tflops Simulation of Driven Mott Insulators on 2032 Processors of the NASA Columbia Supercomputer with Efficient Portable Code"., Submitted to the Supercomputing Conference 2007.

0-7695-3088-5/07 $25.00 © 2007 IEEE