2016 IEEE International Parallel and Distributed Processing Symposium

Massively parallel first-principles simulation of electron dynamics in materials

Erik W. Draeger*, Xavier Andrade*, John A. Gunnels†, Abhinav Bhatele*, André Schleife‡, Alfredo A. Correa*

*Lawrence Livermore National Laboratory, Livermore, California 94551 USA
†IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598 USA
‡University of Illinois at Urbana-Champaign, Urbana, Illinois 61801 USA
E-mail: {draeger1, xavier, bhatele, correatedesco1}@llnl.gov, gunnels@us.ibm.com, schleife@illinois.edu

***Abstract*—We present a highly scalable, parallel implementation of first-principles electron dynamics coupled with molecular dynamics (MD). By using optimized kernels, network topology aware communication, and by fully distributing all terms in the time-dependent Kohn-Sham equation, we demonstrate unprecedented time to solution for disordered aluminum systems of 2,000 atoms (22,000 electrons) and 5,400 atoms (59,400 electrons), with wall clock time as low as 7.5 seconds per MD time step. Despite a significant amount of non-local communication required in every iteration, we achieved excellent strong scaling and sustained performance on the Sequoia Blue Gene/Q supercomputer at LLNL. We obtained up to 59% of the theoretical sustained peak performance on 16,384 nodes and performance of 8.75 Petaflop/s (43% of theoretical peak) on the full 98,304 node machine (1,572,864 cores). Scalable explicit electron dynamics allows for the study of phenomena beyond the reach of standard first-principles MD, in particular, materials subject to strong or rapid perturbations, such as pulsed electromagnetic radiation, particle irradiation, or strong electric currents.**

***Keywords*-first-principles, electron dynamics, molecular dynamics, communication optimization**

I. Overview

In order to gain fundamental understanding of the physics and chemistry of materials and to achieve targeted, bottom-up materials design, we need to be able to accurately model systems at the atomic scale from first principles. Using high-performance computing resources, it is now possible to simulate quantum systems of unprecedented size and complexity. This has enabled the study of previously inaccessible phenomena in important areas such as renewable energy generation and storage, drug discovery, and catalysis, to name just a few [1]. Despite many remarkable achievements, one key limitation of nearly all materials simulations to date has been the assumption of decoupling between the electronic and nuclear degrees of freedom (see Figure 1). In *ab initio* molecular dynamics (AIMD) for instance, nuclei and electrons are treated separately by moving the nuclei classically under the forces computed from the corresponding electronic ground state for that configuration of atoms [2]. While these approximations work well for many systems, they are unable to accurately model systems where electronic and nuclear time scales are not well separated, such as the response of matter to photons or radiation. To capture these phenomena, direct simulation of electron dynamics is needed.

First-principles simulations of electron dynamics have long been an area of research interest. In 1996 Yabana and Bertsch studied the time-resolved response to a short electromagnetic pulse, elucidating the interplay of real-time electron dynamics and optical absorption for molecules and atomic clusters [3]. Electron dynamics simulations have since been applied to study a number of important scientific problems in atomic systems, including response coefficients such as optical activity [4], [5], hyperpolarizabilities [6], and Raman tensors [7]. The method was extended to calculate the dielectric response of crystalline systems [8] and can be used to describe the response of electrons to strong perturbations, such as the interaction of matter with laser fields [9]. This has given insight into high-harmonic generation [10], electron photoemission [11], optimal control of quantum systems [12], optical breakdown [13], multiphoton-absorption [14], and molecular conductance [15], [16], [17].

An efficient, scalable, parallel implementation to simulate electron dynamics will allow these types of calculations to be extended to larger, more complex systems such as interfaces and surfaces. More significantly, however, it enables molecular dynamics (MD) simulations wherein *electrons and nuclei are evolved in time simultaneously*, to capture phenomena such as photoisomerization [18] and dissociation due to strong fields [10], molecular collisions [19], and electronic stopping [20]. Such simulations must necessarily use significantly smaller time steps than AIMD to capture fast electronic response, requiring a time to solution per MD iteration of seconds rather than minutes. In this paper, we present the first such implementation of coupled electron-nuclei dynamics capable of simulating thousands of atoms and tens of thousands of electrons with sub-minute iteration times. We also demonstrate excellent strong scaling efficiency up to 1.5 million compute cores and sustained performance of 8.75 Petaflop/s on the Sequoia Blue Gene/Q machine.

II. Electron dynamics: Current state of the art

While quantum mechanics provides a very precise description of electrons via the Schrödinger's equation, the

1530-2075/16 $31.00 © 2016 IEEE
DOI 10.1109/IPDPS.2016.46

Method	Nuclei	Electrons
Classical MD	Newtonian dynamics with parametrized forces that replicate the effect of the electrons	The electrons are not simulated
Ab Initio Adiabatic MD	Newtonian dynamics with forces from the electrons	Electrons do not have dynamics, they are forced to stay in the ground state
Ab Initio Non-adiabatic MD	**Newtonian dynamics with forces from the electrons**	**Full electron dynamics**

Figure 1. Ladder of approximations for molecular dynamics (MD). As we move down the ladder, computations include more physical effects and become more expensive. Thus, the size of the system that can be studied becomes smaller and the time scales shorter.

exponential cost of solving it exactly makes computational simulations prohibitive for more than a few electrons. For practical calculations, approximations are required. For large scale systems, the method of choice is density functional theory (DFT) [21], [22] with an approximated semi-local exchange and correlation functional. While DFT describes electrons in the ground state, time-dependent DFT (TDDFT) [23], [24] is required to simulate excited electrons. The linear response formulation of TDDFT has been widely used, e.g. to obtain excitation energies and other frequency-dependent properties [25]. For electron dynamics, the real-time formulation of TDDFT can account for response to strong perturbations, as is required by the application described in this paper . It is possible to go beyond TDDFT to achieve a more accurate description of the electronic degrees of freedom, for example by using the time-dependent Bethe-Salpeter equation [26] or the time-dependent coupled cluster method [27]. Unfortunately these approaches are considerably more computationally expensive, and at this moment are not suited for large-scale simulations.

Although the density functional approach reduces the exponential complexity of the Schrödinger equation to a more tractable $\mathcal{O}(N^3)$, where N is the total number of electrons, it remains very computationally demanding, limiting most researchers to the study of hundreds of atoms. Moreover, it does not remove the inherent non-locality of most quantum systems. In a parallel simulation on a supercomputer, this property translates into long-range communication across the full machine, requiring careful data management and keen awareness of the communication topology in order to achieve scalability.

At the same time, the ultrafast time scales of electron dynamics require very short time steps in the simulations and oftentimes large number of simulated particles (atoms and electrons). As a result, researchers have started to focus on high efficiency and large-scale parallelization. The development of an efficient, real-time, electron dynamics simulation was pioneered by the developers of the Octopus code [28], [29], which is based on a real-space grid implementation of DFT/TDDFT and combines parallelization in states with parallelization over real-space domains. Octopus has been shown to scale well up to 32,768 nodes on a Blue Gene/P system for a simulation of a molecule composed of 5,879 atoms (15,825 valence electrons), achieving approximately 10 % of peak performance [30]. Scaling to more processors was limited by the cost of solving the Poisson equation [31], leading to the development of more scalable Poisson solvers [32]. More recent versions of the software feature reduced memory requirements and optimized parallel data distributions [33] that have allowed to predict the light absorption of large photosynthetic complexes with thousands of atoms [34].

In 2014, Noda *et al.* presented a parallel, real-time, real-space implementation of TDDFT running on the K computer at the RIKEN Advanced Institute for Computational Science [35]. To test their implementation, they simulated a molecular crystal of C_{60} fullerenes composed of 1,920 atoms (7,680 valence electrons). They achieved good scaling up to 1,920 nodes, but runs at greater computational scales exhibited a significant increase in the communication cost. In their largest run on 7,920 nodes (61,440 cores), communication accounted for 70% of the execution time. The overall throughput for that run was 7% of the theoretical peak performance.

We note that these last two approaches only consider integration in time of the electrons while keeping the nuclei frozen. While the integration of the nuclear equations of motions takes a negligible amount of time, the calculation of the forces and the recalculation of the structure factors and other objects related to the electron-nuclei interaction, that depend on the atomic positions, incur a significant additional cost.

III. Optimizing Qb@ll for performance and scalability

In this work we use our novel, highly-scalable implementation to address the long-standing problem of electron dc-conductivity at extreme current densities [36]. In this regime, it is necessary to directly simulate the quantum dynamics of scattering of each individual electron that participates in the electric current [37]. This is the first time that this challenging goal has been accomplished in an atomistic calculation for a bulk material. Section IV describes in more detail the specifics and novelty of the scientific application. Since the simulations require large supercells, with 1,000 to 5,000 atoms to represent thermal disorder, this problem presents significant new challenges that were addressed by a combination of new techniques and implementation improvements that will be described in the following.

To treat such large systems, we implemented TDDFT in Qb@ll ("cue ball") [38], a redesigned version of the DFT code, Qbox [39]. Qbox was originally developed at

Lawrence Livermore National Laboratory to achieve excellent scalability and peak performance on machines such as Blue Gene/L [40], [41], [42]. Written in C++, the code uses MPI to carry out communication between compute nodes and a mix of OpenMP and threaded kernels to efficiently utilize all floating-point units on each node. Parallel linear algebra is handled by the ScaLAPACK library [43], and one-dimensional Fourier transforms are computed with FFTW [44] or ESSL [45]. Norm-conserving non-local pseudopotentials [46] were used in separable Kleinman-Bylander form [47]. The viability of an explicit time-integrator within Qb@ll has already been demonstrated [48] and performance results with a stopping power calculation involving 1,600 gold atoms (27,200 electrons) were shown [49].

A. Electron dynamics algorithm

In real-time TDDFT, the electrons are represented by a set of single-particle orbitals $\{\varphi_i(\vec{r},t)\}$. The real-time dynamics of any given state φ_i is described by the time-dependent Kohn-Sham equation,

$$\mathrm{i}\hbar\frac{\partial}{\partial t}\varphi_i(\vec{r},t) = \underbrace{\left\{-\frac{\hbar^2}{2m}\nabla^2 + V_{\text{ext}}(\vec{r},t) + V_{\text{HXC}}[n(\vec{r},t)]\right\}}_{\hat{H}[n(\vec{r},t)]}\varphi_i(\vec{r},t) \tag{1}$$

where t is time and V_{ext} is the external electrostatic Coulomb potential created by the ions. $V_{\text{HXC}}[n(\vec{r},t)]$ is the mean-field contribution to the potential created by the other electrons and includes their electrostatic Hartree interaction and a term derived from quantum-mechanical exchange and correlation. The operator on the right-hand side is collectively known as the instantaneous Kohn-Sham Hamiltonian, $\hat{H}[n]$, and depends on the electronic density $n(\vec{r},t) = \sum_i |\varphi_i(\vec{r},t)|^2$. The real-time propagation of Eq. (1) therefore involves at least two non-trivial steps: the calculation of the Kohn-Sham Hamiltonian from the density, and the application of this operator to the wave function degrees of freedom.

The Kohn-Sham formalism requires the set of orbitals $\{\varphi_i(\vec{r},t)\}$ to be orthogonal to each other. In AIMD methods such as Born-Oppenheimer or Car-Parrinello [50], orthogonality of the orbitals is not preserved and has to be imposed at each iteration through an expensive $\mathcal{O}(N^3)$ procedure that mixes different orbitals [51], [52]. In the context of parallel computing, this requires large-scale linear algebra operations that involve all nodes [40]. In electron dynamics, by contrast, Eq. (1) preserves the orthogonality of the φ_i set during the propagation [53]. By choosing an adequate propagator, this property is retained automatically by the numerical implementation. The absence of an orthogonalization step makes the overall computational cost lower, but more importantly, less inter-processor communication is required, as most operations involve communication along the columns or rows of the process grid.

To do the time integration of the electrons, we implemented both, a fourth-order Runge-Kutta propagator [48] as well as an enforced time reversal symmetry (ETRS) algorithm [10]. The latter propagator is designed specifically for Schrödinger type equations. The propagation of an orbital from time t to time $t+\Delta t$ is given by

$$\begin{aligned}\varphi_i(\vec{r},t+\Delta t) = &\exp\left(-i\frac{\Delta t}{2\hbar}\,\hat{H}\left[n(\vec{r},t+\Delta t)\right]\right)\\ &\times\exp\left(-i\frac{\Delta t}{2\hbar}\,\hat{H}\left[n(\vec{r},t)\right]\right)\varphi_i(\vec{r},t)\ .\end{aligned} \tag{2}$$

As Eq. (1) is non-linear, this propagator becomes implicit; while the second exponential can be readily applied, the first one requires the Hamiltonian operator at time $t+\Delta t$ which depends, through the density, on the orbitals that we want to calculate. We first obtain an approximation for $n(\vec{r},t+\Delta t)$ by propagating the full step with the exponential of $\hat{H}[n(\vec{r},t)]$. This can be considered as a two-step self-consistent iteration. It is possible to introduce a more sophisticated algorithm with a variable number of iterations, that ensures convergence at every step. But we have found this is not required in practice, as the time step is sufficiently small in comparison with the strength of the perturbation for most simulations.

For the calculation of the exponential of the Hamiltonian in Eq. (2), we found that a simple Taylor expansion truncated to fourth order provides a good trade-off between accuracy and computational cost, even when compared to more sophisticated approximations based on Chebyshev polynomials or the Lanczos procedure [10].

Both Runge-Kutta and ETRS propagators are stable over thousands of time steps. While the fourth-order Runge-Kutta has a shorter wall clock time per MD time step, the ETRS propagator is stable over time steps that are twice as large, making the time to solution of the two methods roughly equivalent. As ETRS is time-reversible by construction, it has better conservation of the total energy, electronic charge, and orbital orthogonality.

B. Data layout and communication

Like Qbox, Qb@ll uses a two-dimensional logical process grid to distribute data and limit communication scope (see Figure 2). Electronic orbitals are distributed across process columns, with their plane-wave basis coefficients distributed across the process rows. This structure was instrumental in the excellent strong scalability that resulted in the 2006 Gordon Bell Award [54] for peak performance.

Moving from adiabatic MD to non-adiabatic MD requires a significant increase in the overall scalability of the code; removing the need to reorthogonalize the orbitals at every iteration (adiabatic AIMD additionally requires to perform a subspace diagonalization at each step to recompute occupancy in the case of metals) eliminates a substantial amount

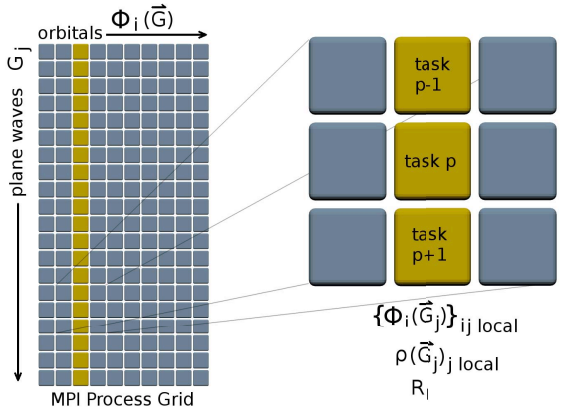

Figure 2. The Qb@ll MPI process grid. The task ordering is column-major, with each column owning a fraction of the electronic orbitals $\phi(\vec{G})$ and a copy of the electronic density in reciprocal space $\rho(\vec{G})$. All tasks also store a copy of the current ionic positions R_I.

of communication, but exposes numerous terms with limited scalability.

For example, because the number of local electronic orbitals on a given process column is typically much larger than one, Qbox computes functions of charge density and ionic positions redundantly on all process columns and tasks, respectively, historically with negligible additional cost. At the extreme strong scaling limit, however, such operations become a significant fraction of the total time. To improve scalability in Qb@ll, we distributed density-dependent work such as the exchange-correlation potential calculation over process columns, with each column computing an equal amount and then accumulating the results with a sub-communicator `MPI_Allreduce` over process rows. Terms that depend on both ionic positions and plane waves, such as the structure factor, were similarly distributed, while double loops ($\mathcal{O}(N^2)$) over ionic positions such as the electrostatic repulsion were spread across both process rows and columns.

We also explored reducing time to solution by distributing computation of the intermediate A_{nl} matrix used to store the projectors of the non-local pseudopotential before they are applied to the electronic orbitals,

$$A_{nl} = (-i)^l e^{\vec{G}_j \cdot \vec{R}_I} w_l(\vec{G}_j) v_l(\vec{G}_j), \tag{3}$$

where $\vec{R}_I$ is the position of the nuclei, $\vec{G}_j$ is the local plane wave basis point, w_l is the projector of angular momentum, l and v_l is the non-local potential. Because the total number of projectors is proportional to the total number of atoms, storing the full projection matrix even, for just the local basis points, becomes prohibitive with thousands of atoms. Instead, a blocked loop is used to limit the local size of A_{nl}, with all tasks computing the contribution for their local basis points from all atoms in fixed-size pieces (for this work, the block size was 256 atoms). Historically, this calculation has been a small fraction of the total DFT iteration loop time. However, per Amdahl's Law, this unscalable term will eventually dominate. To mitigate this, we implemented the option to distribute the blocked loop across columns, with each process broadcasting A_{nl} corresponding to its block of atoms across its row sub-communicator in turn. This has the added benefit of allowing persistent storage of A_{nl} across multiple evaluations of the Hamiltonian, updating it only when the atom (nuclei) positions change. (Some electron dynamics calculations are done with fixed nuclei.) We found that trading computation for communication in this way can be favorable to time to solution, provided communication bandwidth is sufficient.

C. Mapping MPI tasks onto the 5D torus

The heavy communication requirements of Qb@ll and the five-dimensional torus topology of Blue Gene/Q make the performance of Qb@ll highly sensitive to the mapping or layout of MPI tasks on the torus. Communication performance of the code is affected both by contention for injection bandwidth and for link bandwidth. By default, the job scheduler assigns MPI tasks to nodes and cores using a scheme called 'ABCDET' in which tasks are placed on the cores within a node first and then along the 'E' dimension, then the 'D' dimension and so on. However, this default mapping tends to scale poorly because of resource contention in the injection FIFOs and network links. Finding the optimal mapping is NP-hard and even developing good heuristics is complicated by the fact that the shape of the 5D partition allocated for a job can change from one job to another, especially at lower node counts.

In order to improve the communication characteristics and overall performance of the code, we investigated using other layouts such as 'CTEABD' 'ATEBCD' and 'BEACTD' All of these spread consecutive MPI tasks to different nodes instead of placing them on different cores of the same node. We also used a Python based task mapping tool called Rubik [55] to improve link bandwidth utilization by placing communication pairs so that they are separated by links in multiple directions of the torus. We found that suboptimal task mapping had a substantial impact on performance, as discussed in Section VI.

D. An optimized threaded xGEMM kernel for Blue Gene/Q

Because the majority of the non-local pseudopotential evaluation relies upon matrix multiplication, an efficient xGEMM kernel is essential. While the performance of matrix multiplication is contingent upon carefully leveraging the hardware features relevant at each level of the memory hierarchy, the performance rewards reaped and the complexity required increases as one approaches the highest (smallest) levels of the memory pyramid. Correspondingly, the

structure of the implementation is most coherently described from the top down (i.e., floating point unit to main memory).

The inner kernel of matrix-multiply is constructed as an outer product formulation with the largest surface-to-volume ratio possible, in order to reduce bandwidth requirements, while sparing enough registers to cover expected latencies. Here, the DGEMM kernel is an 8×8 outer product with the C matrix occupying 16 (4-way) SIMD registers in a 2×8 register grid. Conformal to this, the A matrix component occupies a 2×1 vector register slice (logically 8×1) while the B matrix occupies a 1×8 set of registers (logically also 1×8). The A and B matrices are not treated symmetrically, as the A matrix elements are loaded componentwise (e.g., $\{a1, a2, a3, a4\}$) while the B matrix components are loaded redundantly (splatted) into the register file (e.g., $\{bx, bx, bx, bx\}$). This differential treatment stems from the fact that there are a limited number of cross-instructions in the QPX instruction set and the component interactions required in an outer product style calculation (preferable for reasons of bandwidth consumption and latency scheduling) indicate this treatment. Analogously, the core of the ZGEMM routine utilizes an 8×4 (complex) outer product with the C matrix occupying 16 SIMD registers in a 4×4 register grid.

In both DGEMM and ZGEMM, the elements of C are loaded in a manner conformal to the A matrix. For the operands that experience reuse, A and B, these alignment requirements are met through data reformatting, which is standard practice in this area. The QPX unit employs a rich set of load and permute instructions that allow this reformatting step (which often includes a multi-level transpose) to be done efficiently and, from the perspective of the assembly language required, cleanly.

The next step is to consider the interface between the register file and the multiple cache levels. Consider the ZGEMM kernel. Viewed as an isolated unit of computation the kernel must load 12 complex values over the course of 32 cycles, equating to 6 bytes/cycle, which the L2 cache of Blue Gene/Q can easily furnish. While the computational kernel code is scheduled so as to tolerate a great deal of latency, this is only required when new data streams are being initiated. After a stream has been established, the L1 prefetch cache (L1P) reduces the effective latency from 82 to 24 cycles, below the latency tolerance of the kernel routine even when it is run on a single thread. Thus, our implementation differs from the traditional design insofar as we do not rely upon the high-bandwidth, low-latency characteristics of the L1D cache to realize performance.

The use of multiple threads is well-known to improve latency tolerance. In the work presented here, we utilize the SMT threads to reduce bandwidth demand. Consider four threads, each executing the 8x8 DGEMM inner kernel. While each thread, independently, would require 8 bytes/cycle to proceed at peak speed, if one views these outer products on a 2x2 grid wherein the A and B matrices are each two-way shared, the bandwidth demands are cut in half. Novel enhancements, such as mutual (partner) prefetch [56], using the L1D cache as a coordination area, were used to enforce this behavior by implicitly synchronizing cores as well as reducing, by the elimination of redundancy, the total number of prefetch instructions issued.

The matrix multiplication routines, outlined above, execute at a high percentage of peak on the Blue Gene/Q system. For example, the ZGEMM implementation processes some of the matrices of interest in Qb@ll at over 95% of peak when using 64 threads per node. Our design differs from traditional designs as regards its treatment of the L1D cache, the use of mutual prefetching, and general "over-engineering" to reduce bandwidth requirements and increase latency tolerance to levels that the system is more than capable of delivering, allowing the code to run at near peak rates across a wide spectrum of matrix sizes and thread counts. Specific improvements for the work presented, over previous realizations of matrix-multiplication kernels on Blue Gene/Q [57], include low-level improvements regarding the timing of store instructions and multi-level blocking to effectively deal with the extreme aspect ratios of some of the matrices of interest.

IV. Application: dc-conductivity in disordered aluminum

In this section we describe an approach to compute electrical dc-conductivity by direct simulation of a current in a metallic system, including the reaction of the lattice atoms. The specific physical systems we selected in order to measure performance are two supercells of 2,000 and 5,400 aluminum atoms, with 22,000 to 59,400 electrons respectively, in the presence of a high density electronic current of the order of $10^{11}\ \mathrm{A/cm^2}$. Non-local norm-conserving pseudopotentials were used to represent the electron-ion interaction of the aluminum nuclei, with 11 valence and semi-core electrons per atom using a plane wave basis energy cutoff of 65 Ry. In order to measure properties such as the electronic dc-conductivity of metals and plasmas at high temperature, large supercells are needed to represent thermal disorder and break the spatial coherence of Bloch waves. A crystalline fcc structure with added random disorder is used for the atomic positions. We use an elongated supercell of $5 \times 5 \times 20$ and $5 \times 5 \times 54$ fcc units cells.

Initial conditions require a strong current be imposed on the electrons (relative to the ions frame of reference) in order to observe the relation between the electric field, electronic current and the decay in time. While such a current can be induced via an electric field or a phase factor for each electronic orbital, we use a simpler approach: By assigning all atoms an initial velocity we simulate a condition that is equivalent to an initial non-equilibrium electronic current that is allowed to evolve in time. (A Galilean system of

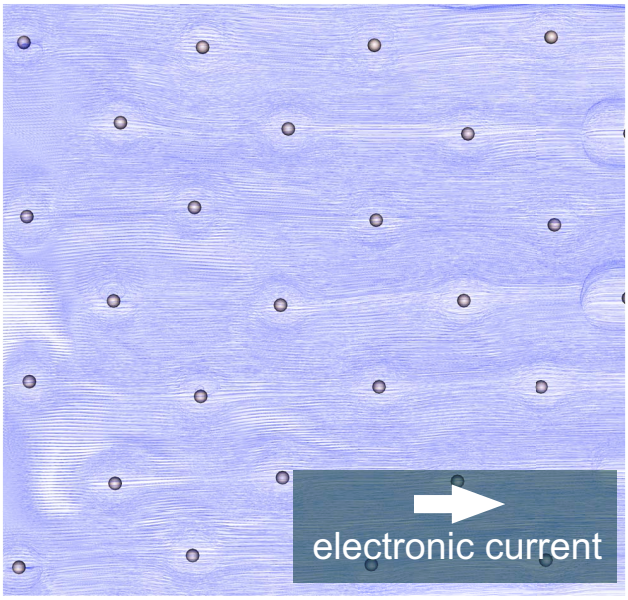

Figure 3. Streamlines of the electronic current density in the aluminum supercell 100 femtoseconds after establishing the initial current. Current was induced to flow from left to right at $t = 0$ and decays with time.

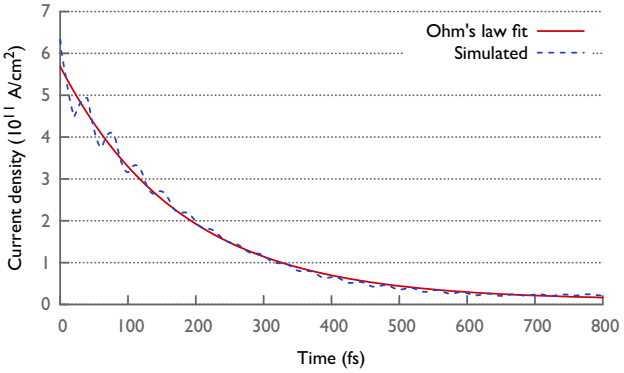

Figure 4. Calculated electronic current density as a function of time in aluminum supercell with random disorder. Red line shows a fit of the current to the exponential decay predicted by Ohm's Law.

reference transformation allows us to study the problem in any inertial reference system.)

We can now compute electronic transport properties such as the electronic current density (based on the definition $\vec{j} = \sum_i \frac{e\hbar}{2mi}(\varphi_i \vec{\nabla}\varphi_i^* - \varphi_i \vec{\nabla}\varphi_i^*)$), the internal electrostatic field generated, and the back reaction forces on the ions. By following the decay of the induced current density (depicted in Figure 3) we can obtain the bulk conductivity. The results are shown in Figure 4. By fitting the data to an exponential decay model predicted by Ohm's Law, we can estimate a value for the conductivity of 8.2×10^8 S/m (under these extreme currents). Just for comparison, the experimental value for the conductivity of aluminum at a temperature of 293 K is 2.8×10^8 S/m. The observed discrepancy is still being investigated, but we anticipate improved agreement once we average over multiple atomic configurations generated from an adiabatic MD simulation at that temperature.

V. System: Sequoia Blue Gene/Q

Blue Gene/Q is the third generation of Blue Gene architectures from IBM. It follows the same design principle as its predecessors and is built from a large number of low-power and homogeneous nodes. In total, the largest installation currently deployed at Lawrence Livermore National Laboratory comprises 96 racks with 1024 nodes each. Each node consists of a single processor with 17 64-bit PowerPC A2 compute cores running at 1.6 GHz, of which 16 are available to the application and the 17th is used for progress threads and system management functionality. Each core provides four hardware threads using simultaneous multithreading (SMT), at least two must be used to achieve full issue bandwidth. This results in a total of 1,572,864 compute cores or 6,291,456 hardware threads, providing an unprecedented level of concurrency.

In addition to the main ALUs, each core also contains a Quad-FPU unit (QPX), which offers four double precision pipelines that can be configured for scalar FPU processing, 4-wide FPU SIMD, or 2-wide complex arithmetic SIMD. The Power ISA has been extended to accommodate additional instructions for the Quad-FPU and the load/store unit supports multiple alignments, which is an improvement when compared to previous Blue Gene designs.

The memory system consists of a 16 kB L1 data and 16 kB instruction cache integrated into the core, as well as an on-chip 32 MB L2 cache with a cache line size of 128 bytes, which is split into 16 2-MB slices. Groups of eight slices are connected to one of two on-chip memory controllers, which provide access to a total of 16 GB per node, or 1.5 PB in the overall machine. The L2 cache is connected to the compute nodes through a crossbar interconnect operating at 800 MHz. Additionally, the L2 cache is multi-versioned to support novel techniques such as hardware transactions or speculative loop parallelization. However, neither of these options is used in this work since we are relying on deterministic and predictable application patterns, making speculation unnecessary.

All compute nodes are connected through a 5 dimensional torus network. As the smallest unit of allocation, each compute card, consisting of 32 nodes, is structured as 2x2x2x2x2 hypercube, which can be electrically isolated. Multiple compute cards are then assembled to create a 4x4x4x4x2 half rack midplane. The complete machine, as built at LLNL, is organized as 16x12x16x16x2 5D torus. Each torus link is bidirectional and provides a peak bandwidth of 2 GB/s in each direction. In addition, each compute

card is connected to an I/O node with the same architecture. I/O nodes, physically located on top of the compute racks, provide Infiniband adapters for connection to the file system and external access.

Blue Gene/Q has a large set of hardware performance counters that enabled us to measure the sustained performance of our application directly. Data was collected and reported using the HPM library provided by IBM. Function calls were inserted around the main iteration loop to exclude initialization costs. All communication times and sustained performance results reported in this paper include only the main iteration loop, over a minimum of three iterations.

VI. Performance Results

We evaluate the scalability and time to solution of Qb@ll on Sequoia for both 2,000-atom (22,000 valence electrons) and 5,400-atom (59,400 valence electrons) aluminum. All runs used 4 MPI tasks per node and 16 threads per task and an MPI task mapping so that the number of process rows was a multiple of leading torus dimensions. For each system, the wall time per iteration of MD with explicit electron dynamics was measured using both the fourth-order Runge-Kutta and ETRS exponential propagators described in Section III-A as well as both the fully-distributed and column-distributed implementations of the A_{nl} matrix described in Section III-B. Figure 5 shows the strong scaling behavior of all systems. At full scale, the 2,000-atom system had iteration times of 7.5 sec for the Runge-Kutta propagator and 16.1 sec for ETRS, the fastest time ever reported for a large metallic first-principles MD calculation. The 5,400-atom system had iteration times of 53.2 sec and 115.8 sec for fourth-order Runge-Kutta (FORK) and ETRS, respectively. At smaller node counts, the time per iteration for the ETRS propagator was consistently about twice that of Runge-Kutta, making them roughly equivalent in time to solution once the size of the time step is taken into account. In nearly all cases, the distributed A_{nl} approach was faster, with the additional communication being more than mitigated by the reduction in computation. The one exception was the full scale 2,000-atom system with Runge-Kutta integration, as local data sizes became small enough that the extra computation was faster than the communication costs of distribution. In nearly all cases, both methods for calculating A_{nl} had very similar times to solution. The parallel efficiency of the 2,000-atom runs was 34–38% over a 96-fold increase in node count, while the parallel efficiency of the 5,400-atom runs was 70 and 76% over 24-fold and 12-fold increases in node count, respectively.

The timing breakdown of the main iteration loop is shown in Figure 6. Three primary code regions are highlighted: the calculation of the electronic charge density ("charge"), the calculation of the Hamiltonian from all terms except the non-local potential ("hpsi"), and the calculation of energy, forces and the contribution to the Hamiltonian from

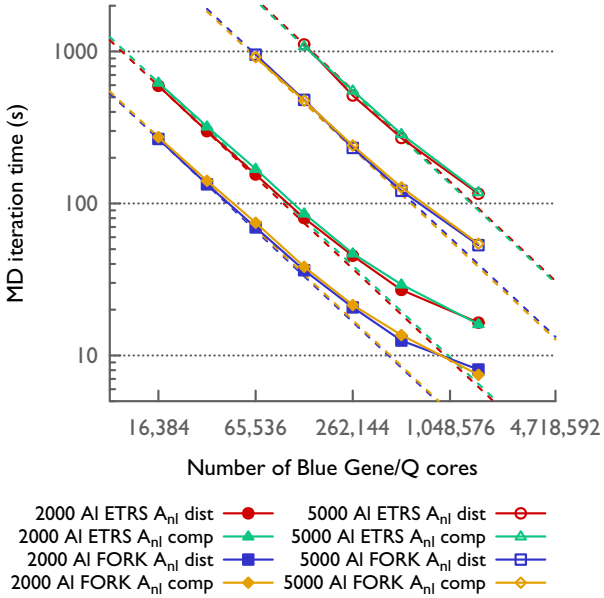

Figure 5. Strong scaling of molecular dynamics iteration time for disordered aluminum on Sequoia for both 2,000-atom (22,000 electron) and 5,400-atom (59,400 electron) systems using either fourth-order Runge-Kutta (FORK) or Enforced Time-Reversal Symmetry (ETRS). Dashed lines indicate perfect scaling.

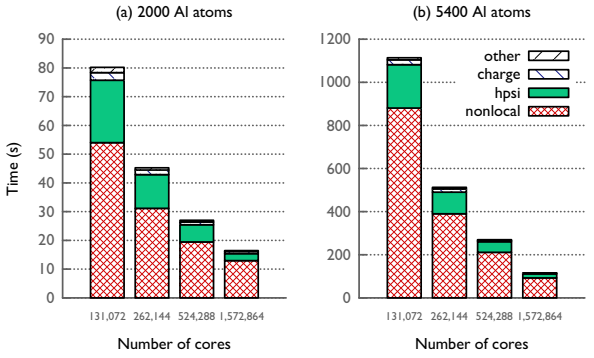

Figure 6. Timing breakdown of dominant terms in iteration loop time for (a) 2000 Al atoms and (b) 5400 Al atoms, using the ETRS propagator and distributed A_{nl}. The "nonlocal" label refers to the time evaluating all terms related to the non-local pseudopotential (energy, Hamiltonian and forces), 'other' refers to the contribution of all other terms in the potential to the Hamiltonian, and 'charge' refers to the calculation of the electronic charge density from the orbitals.

the non-local potential ("nonlocal"). The first two terms, charge and hpsi, are dominated by parallel 3D Fast Fourier Transforms. Qb@ll uses a custom 3D Fourier Transform designed to match the code's data layout, with grid points distributed across process columns in xy-planes. Transposes are handled with `MPI_Alltoallv` calls on column sub-communicators, and the FFTW or ESSL library is used

to compute the local 1D transforms. For the case of 4 MPI tasks and 16 threads, we found that an OpenMP loop over 1D FFTW calls gave the fastest time to solution. The charge density includes an `MPI_Allreduce` over row sub-communicators to accumulate contributions from the distributed orbitals. The non-local term is dominated by calls to ZGEMM and DGEMM, with calls to `MPI_Allreduce` over row sub-communicators to accumulate orbital contributions. When the distributed A_{nl} algorithm is used, "non-local" also includes `MPI_Bcast` calls to distribute the blocked data.

The communication profile for the largest runs is shown in Figure 7. Because communication was almost entirely limited to row and column sub-communicators, communication times scaled well with the number of cores provided the tasks were mapped onto the torus optimally. Figure 8 shows the effect of communication bottlenecks caused by suboptimal task mapping, particularly when A_{nl} was fully distributed. Because the process grid is column-major, row communication suffers the most from oblivious or poorly-aligned task mapping. As such, the default task-first mapping ('ABCDET') is often the worst choice, as not only do process columns not align with the torus dimensions but neighboring rows will be attempting to do sub-communicator collectives simultaneously on the same network links. In the case of the 8192 node runs shown in Figure 8, the `MPI_Bcast` time increases by over a factor of six, dropping the effective bandwidth from from 1.15 GB/s to 46.9 MB/s.

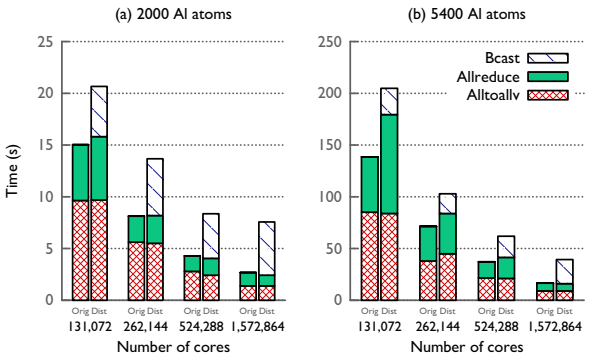

Figure 7. Comparison of the time spent in different MPI routines for the original versus distributed computation of the intermediate A_{nl} matrix, using the ETRS propagator.

The sustained floating-point performance is shown in Figure 9. Despite a significant fraction of time spent in communication and FFTs at full scale, the 2,000-atom system still achieves 3.6-4.0 Petaflop/s. The 5,400-atom system had a measured sustained performance of 8.18 Petaflop/s with the FORK integrator and 8.75 Petaflop/s with the ETRS integrator, a record for an application with non-local communication. This corresponds to 43.5% of peak at full

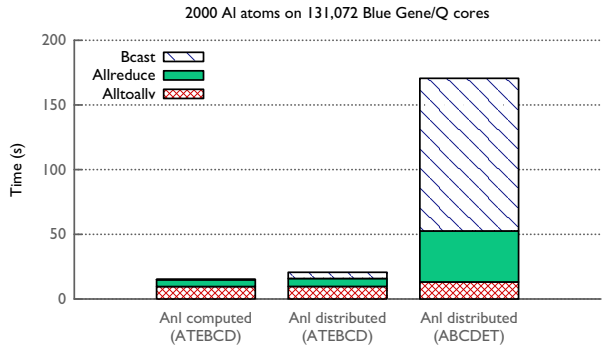

Figure 8. Communication costs associated with task mapping for the 2000-atom system on 8,192 Blue Gene/Q nodes. With torus dimensions of 4x4x16x16x2 and a 1024x32 MPI process grid, the 'ATEBCD' mapping ensures that process columns were aligned along torus dimensions. The default 'ABCDET' mapping, on the other hand, not only had mismatched alignment, but the task-fastest ordering resulted in significant resource contention during row sub-communicator collectives.

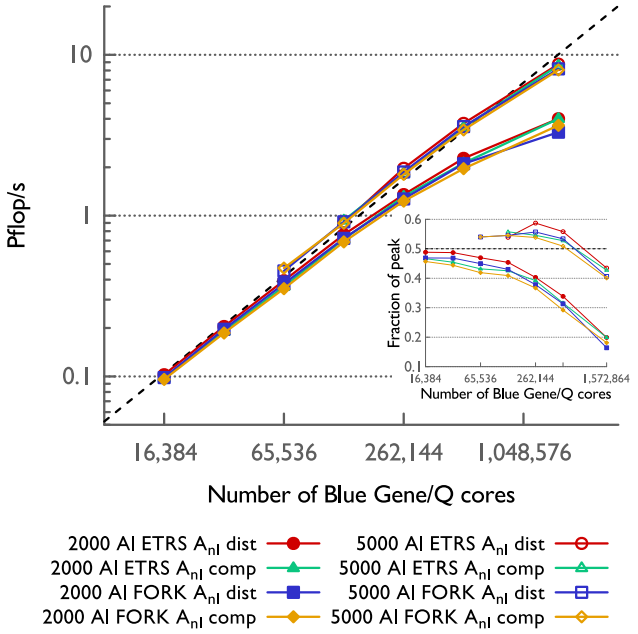

Figure 9. Sustained performance of Qb@ll simulations of disordered aluminum on Sequoia, for both 2,000-atom (22,000 electron) and 5,400-atom (59,400 electron) systems using either fourth-order Runge-Kutta (FORK) or Enforced Time-Reversal Symmetry (ETRS). The lower right inset shows the corresponding fraction of theoretical peak performance for Sequoia (20.1 Petaflop/s), with the brown dashed line in both plots showing 50% of peak to guide the eye.

scale, although fractions of peak as high as 58.7% were observed at 16,384 nodes with the ETRS propagator.

VII. Conclusion and Future Outlook

We have demonstrated a highly scalable implementation of non-adiabatic first-principles MD with explicit electron

dynamics. This capability will enable first-principles simulation of transport properties, such as electrical and heat conductivity, electron diffusion under high currents, electronic stopping power, particularly in the extreme non-linear domain where perturbative theories cannot be used. Other interesting avenues of research include the mechanism of light absorption and energy transfer within photosynthetic complexes, which requires simulations of thousands of atoms and many thousands of time steps [58]. Another interesting problem is that of electromigration [59]. Electromigration is the displacement of the atoms in a conductor due to the momentum transfer from a current of electrons. It is an important effect in microelectronics due to the high current densities and small size of the conductors. Using explicit electron dynamics, it is possible to simulate the current and the resulting nuclei-electron interactions, although long simulation times are required to capture the movement of nuclei. Studying the nanoscopic features of electromigration and the factors that control it would be extremely useful for numerous fields and industries. Understanding the interaction between electronic currents and nuclei also opens the door to molecular-sized motors [60]. In addition, recent signal analysis techniques have been developed to maximize the information from time resolved properties [61] that may further accelerate the time to solution of electron dynamics simulations.

More generally, the capability to perform large scale simulations of electron dynamics opens the possibility of simulating materials that are subject to strong or rapid perturbations, such as pulsed electromagnetic radiation (e.g. high intensity lasers, photovoltaic processes), particle irradiation (e.g. protons or nuclear decay products) or strong currents (e.g. in hot plasmas). This has numerous important applications in energy research.

As supercomputers continue to grow more powerful, it is expected that the trend toward heterogeneity and complexity will continue as well. Qb@ll's heavy reliance on general kernels such as FFTs and matrix multiplication provides a clear path to take full advantage of new hardware architectures, including the next generation of GPU architectures.

ACKNOWLEDGMENTS

The authors wish to thank the Livermore Computing team for their tireless dedication and support, the NNSA Advanced Scientific Computing program for their support of this work and William Krauss at LLNL for visualization support. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-CONF-669964). Computing support for this work came from the Lawrence Livermore National Laboratory Institutional Computing Grand Challenge program.

REFERENCES

[1] P. Dederichs, "Ψ_k Scientific Highlight Of The Month," 2014. [Online]. Available: http://www.psi-k.org/newsletters/News_122/Highlight_122.pdf

[2] D. Marx and J. Hutter, *Ab Initio Molecular Dynamics*. Cambridge University Press (CUP), 2009. [Online]. Available: http://dx.doi.org/10.1017/cbo9780511609633

[3] K. Yabana and G. F. Bertsch, "Time-dependent local-density approximation in real time," *Phys. Rev. B*, vol. 54, no. 7, pp. 4484–4487, Aug 1996. [Online]. Available: http://dx.doi.org/10.1103/physrevb.54.4484

[4] ——, "Application of the time-dependent local density approximation to optical activity," *Phys. Rev. A*, vol. 60, no. 2, pp. 1271–1279, Aug 1999. [Online]. Available: http://dx.doi.org/10.1103/physreva.60.1271

[5] D. Varsano, L. A. Espinosa-Leal, X. Andrade, M. A. L. Marques, R. di Felice, and A. Rubio, "Towards a gauge invariant method for molecular chiroptical properties in TDDFT," *Phys. Chem. Chem. Phys.*, vol. 11, no. 22, p. 4481, 2009. [Online]. Available: http://dx.doi.org/10.1039/b903200b

[6] Y. Takimoto, F. D. Vila, and J. J. Rehr, "Real-time time-dependent density functional theory approach for frequency-dependent nonlinear optical response in photonic molecules," *J. Chem. Phys.*, vol. 127, no. 15, p. 154114, 2007. [Online]. Available: http://dx.doi.org/10.1063/1.2790014

[7] M. Thomas, F. Latorre, and P. Marquetand, "Resonance Raman spectra of ortho-nitrophenol calculated by real-time time-dependent density functional theory," *J. Chem. Phys.*, vol. 138, no. 4, p. 044101, 2013. [Online]. Available: http://dx.doi.org/10.1063/1.4776218

[8] G. F. Bertsch, J.-I. Iwata, A. Rubio, and K. Yabana, "Real-space real-time method for the dielectric function," *Phys. Rev. B*, vol. 62, no. 12, pp. 7998–8002, Sep 2000. [Online]. Available: http://dx.doi.org/10.1103/physrevb.62.7998

[9] K. Yabana, T. Otobe, and J.-I. Iwata, "First-Principles Density-Functional Approach for Many-Electron Dynamics Under Intense Laser Fields," in *Progress in Ultrafast Intense Laser Science Volume I*. Springer Science + Business Media, 2006, pp. 77–94. [Online]. Available: http://dx.doi.org/10.1007/3-540-34422-5_5

[10] A. Castro, M. A. L. Marques, J. A. Alonso, G. F. Bertsch, and A. Rubio, "Excited states dynamics in time-dependent density functional theory," *Eur. Phys. J. D*, vol. 28, no. 2, pp. 211–218, Feb 2004. [Online]. Available: http://dx.doi.org/10.1140/epjd/e2003-00306-3

[11] U. D. Giovannini, D. Varsano, M. A. L. Marques, H. Appel, E. K. U. Gross, and A. Rubio, "Ab initio angle- and energy-resolved photoelectron spectroscopy with time-dependent density-functional theory," *Phys. Rev. A*, vol. 85, no. 6, Jun 2012. [Online]. Available: http://dx.doi.org/10.1103/physreva.85.062515

[12] A. Castro, J. Werschnik, and E. K. U. Gross, "Controlling the Dynamics of Many-Electron Systems from First Principles: A Combination of Optimal Control and Time-Dependent Density-Functional Theory," *Phys. Rev. Lett.*, vol. 109, no. 15, Oct 2012. [Online]. Available: http://dx.doi.org/10.1103/physrevlett.109.153603

[13] T. Otobe, M. Yamagiwa, J.-I. Iwata, K. Yabana, T. Nakatsukasa, and G. F. Bertsch, "First-principles electron dynamics simulation for optical breakdown of dielectrics under an intense laser field," *Phys. Rev. B*, vol. 77, no. 16, Apr 2008. [Online]. Available: http://dx.doi.org/10.1103/physrevb.77.165104

[14] T. Otobe, "First-principles calculations for multiphoton absorption in $alpha$-quartz under intense short laser irradiation," *Journal of Physics: Condensed Matter*, vol. 22, no. 38, p. 384204, Sep 2010. [Online]. Available: http://dx.doi.org/10.1088/0953-8984/22/38/384204

[15] S. Kurth, G. Stefanucci, C.-O. Almbladh, A. Rubio, and E. K. U. Gross, "Time-dependent quantum transport: A practical scheme using density functional theory," *Phys. Rev. B*, vol. 72, no. 3, Jul 2005. [Online]. Available: http://dx.doi.org/10.1103/physrevb.72.035308

[16] C.-L. Cheng, J. S. Evans, and T. V. Voorhis, "Simulating molecular conductance using real-time density functional theory," *Phys. Rev. B*, vol. 74, no. 15, Oct 2006. [Online]. Available: http://dx.doi.org/10.1103/physrevb.74.155112

[17] X. Qian, J. Li, X. Lin, and S. Yip, "Time-dependent density functional theory with ultrasoft pseudopotentials: Real-time electron propagation across a molecular junction," *Phys. Rev. B*, vol. 73, p. 035408, Jan 2006. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevB.73.035408

[18] I. Tavernelli, U. F. Röhrig, and U. Rothlisberger, "Molecular dynamics in electronically excited states using time-dependent density functional theory," *Molecular Physics*, vol. 103, no. 6-8, pp. 963–981, Mar 2005. [Online]. Available: http://dx.doi.org/10.1080/00268970512331339378

[19] G. Avendaño-Franco, B. Piraux, M. Grüning, and X. Gonze, "Time-dependent density functional theory study of charge transfer in collisions," in *Highlights in Theoretical Chemistry*. Springer Science + Business Media, Oct 2012, pp. 241–250. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-41315-5_20

[20] A. Schleife, Y. Kanai, and A. A. Correa, "Accurate atomistic first-principles calculations of electronic stopping," *Phys. Rev. B*, vol. 91, no. 1, Jan 2015. [Online]. Available: http://dx.doi.org/10.1103/physrevb.91.014306

[21] P. Hohenberg and W. Kohn, "Inhomogeneous Electron Gas," *Phys. Rev.*, vol. 136, no. 3B, pp. B864–B871, Nov 1964. [Online]. Available: http://dx.doi.org/10.1103/physrev.136.b864

[22] W. Kohn and L. J. Sham, "Self-Consistent Equations Including Exchange and Correlation Effects," *Phys. Rev.*, vol. 140, no. 4A, pp. A1133–A1138, Nov 1965. [Online]. Available: http://dx.doi.org/10.1103/physrev.140.a1133

[23] E. Runge and E. K. U. Gross, "Density-Functional Theory for Time-Dependent Systems," *Phys. Rev. Lett.*, vol. 52, no. 12, pp. 997–1000, Mar 1984. [Online]. Available: http://dx.doi.org/10.1103/physrevlett.52.997

[24] M. Marques and E. Gross, "Time-Dependent Density Functional Theory," *Annu. Rev. Phys. Chem.*, vol. 55, no. 1, pp. 427–455, Jun 2004. [Online]. Available: http://dx.doi.org/10.1146/annurev.physchem.55.091602.094449

[25] E. K. U. Gross and N. T. Maitra, "Introduction to TDDFT," in *Fundamentals of Time-Dependent Density Functional Theory*. Springer Science + Business Media, 2012, pp. 53–99. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-23518-4_4

[26] C. Attaccalite, M. Grüning, and A. Marini, "Real-time approach to the optical properties of solids and nanostructures: Time-dependent Bethe-Salpeter equation," *Phys. Rev. B*, vol. 84, no. 24, Dec 2011. [Online]. Available: http://dx.doi.org/10.1103/physrevb.84.245110

[27] S. Kvaal, "Ab initio quantum dynamics using coupled-cluster," *J. Chem. Phys.*, vol. 136, no. 19, p. 194109, 2012. [Online]. Available: http://dx.doi.org/10.1063/1.4718427

[28] A. Castro, H. Appel, M. Oliveira, C. A. Rozzi, X. Andrade, F. Lorenzen, M. A. L. Marques, E. K. U. Gross, and A. Rubio, "octopus: a tool for the application of time-dependent density functional theory," *physica status solidi (b)*, vol. 243, no. 11, pp. 2465–2488, Sep 2006. [Online]. Available: http://dx.doi.org/10.1002/pssb.200642067

[29] X. Andrade, D. Strubbe, U. D. Giovannini, A. H. Larsen, M. J. T. Oliveira, J. Alberdi-Rodriguez, A. Varas, I. Theophilou, N. Helbig, M. J. Verstraete, L. Stella, F. Nogueira, A. Aspuru-Guzik, A. Castro, M. A. L. Marques, and A. Rubio, "Real-space grids and the Octopus code as tools for the development of new simulation approaches for electronic systems," *Phys. Chem. Chem. Phys.*, 2015. [Online]. Available: http://dx.doi.org/10.1039/c5cp00351b

[30] J. Alberdi-Rodriguez, X. Andrade, A. Arruabarrena, J. Muguerza, and A. Rubio, "Improving Octopus Towards the new Generation of HPC Systems," in *Jülich Blue Gene/P Extreme Scaling Workshop*. FORSCHUNGSZENTRUM JÜLICH GmbH, 2011, pp. 3–6.

[31] X. Andrade, J. Alberdi-Rodriguez, D. A. Strubbe, M. J. T. Oliveira, F. Nogueira, A. Castro, J. Muguerza, A. Arruabarrena, S. G. Louie, A. Aspuru-Guzik, A. Rubio, and M. A. L. Marques, "Time-dependent density-functional theory in massively parallel computer architectures: the octopus project," *Journal of Physics: Condensed Matter*, vol. 24, no. 23, p. 233202, May 2012. [Online]. Available: http://dx.doi.org/10.1088/0953-8984/24/23/233202

[32] P. García-Risueño, J. Alberdi-Rodriguez, M. J. T. Oliveira, X. Andrade, M. Pippig, J. Muguerza, A. Arruabarrena, and A. Rubio, "A survey of the parallel performance and accuracy of Poisson solvers for electronic structure calculations," *J. Comput. Chem.*, vol. 35, no. 6, pp. 427–444, Nov 2013. [Online]. Available: http://dx.doi.org/10.1002/jcc.23487

[33] J. Alberdi-Rodriguez, M. J. T. Oliveira, P. García-Risueño, F. Nogueira, J. Muguerza, A. Arruabarrena, and A. Rubio, "Recent Memory and Performance Improvements in Octopus Code," in *Computational Science and Its Applications – ICCSA 2014*. Springer Science + Business Media, 2014, pp. 607–622. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-09147-1_44

[34] J. Jornet-Somoza, J. Alberdi-Rodriguez, B. F. Milne, X. Andrade, M. A. L. Marques, F. Nogueira, M. J. T. Oliveira, J. J. P. Stewart, and A. Rubio, "Insights into colour-tuning of chlorophyll optical response in green plants," *Phys. Chem. Chem. Phys.*, pp. –, 2015. [Online]. Available: http://dx.doi.org/10.1039/C5CP03392F

[35] M. Noda, K. Ishimura, K. Nobusada, K. Yabana, and T. Boku, "Massively-parallel electron dynamics calculations in real-time and real-space: Toward applications to nanostructures of more than ten-nanometers in size," *Journal of Computational Physics*, vol. 265, pp. 145–155, May 2014. [Online]. Available: http://dx.doi.org/10.1016/j.jcp.2014.02.006

[36] M. P. Desjarlais, J. D. Kress, and L. A. Collins, "Electrical conductivity for warm dense aluminum plasmas and liquids," *Phys. Rev. E*, vol. 66, no. 2, Aug 2002. [Online]. Available: http://dx.doi.org/10.1103/PhysRevE.66.025401

[37] H. Reinholz, G. Röpke, S. Rosmej, and R. Redmer, "Conductivity of warm dense matter including electron-electron collisions," *Phys. Rev. E*, vol. 91, no. 4, Apr 2015. [Online]. Available: http://dx.doi.org/10.1103/PhysRevE.91.043105

[38] E. W. Draeger and F. Gygi, "Qbox code, Qb@ll version," Lawrence Livermore National Laboratory.

[39] F. Gygi, "Qbox open source code project," http://eslab.ucdavis.edu/, University of California, Davis.

[40] F. Gygi, E. W. Draeger, B. R. D. Supinski, R. K. Yates, F. Franchetti, S. Kral, J. Lorenz, C. W. Ueberhuber, J. A. Gunnels, and J. C. Sexton, "Large-Scale First-Principles Molecular Dynamics Simulations on the Blue Gene/L Platform using the Qbox Code," *In Proceedings of Supercomputing 2005*, no. 4, p. 24, 2005, conference on High Performance Networking and Computing Gordon Bell Prize finalist.

[41] F. Gygi, E. W. Draeger, M. Schulz, B. R. de Supinski, J. A. Gunnels, V. Austel, J. C. Sexton, F. Franchetti, S. Kral, C. W. Ueberhuber, and J. Lorenz, "Large-Scale Electronic Structure Calculations of High-Z Metals on the Blue Gene/L Platform," *In Proceedings of Supercomputing 2006*, 2006, International Conference on High Performance Computing, Network, Sto rage, and Analysis. 2006 Gordon Bell Prize winner (Peak Performance).

[42] B. R. de Supinski, M. Schultz, and E. W. Draeger, "Flexible Tools Supporting a Scalable First-Principles MD Code," in *Scientific Computer Performance*, D. H. Bailey, R. F. Lucas, and S. Williams, Eds. Taylor and Francis, 2010, pp. 297–314.

[43] L. Blackford, J. Choi, A. Cleary, E. Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G.Henry, A.Petitet, K.Stanley, D.Walker, , and R.C.Whaley, "ScaLAPACK User's Guide," SIAM, Philadelphia, 1997.

[44] M. Frigo and S. G. Johnson, "FFTW: an adaptive software architecture for the FFT," in *Proceedings of ICASSP*, vol. 3, 1998, pp. 1381–1384.

[45] S. Filippone, "Parallel libraries on distributed memory architectures: The IBM Parallel ESSL," in *Applied Parallel Computing Industrial Computation and Optimization*. Springer Science + Business Media, 1996, pp. 247–255. [Online]. Available: http://dx.doi.org/10.1007/3-540-62095-8_26

[46] N. Troullier and J. L. Martins, "Efficient pseudopotentials for plane-wave calculations," *Phys. Rev. B*, vol. 43, no. 3, pp. 1993–2006, Jan 1991. [Online]. Available: http://dx.doi.org/10.1103/physrevb.43.1993

[47] I. Morrison, D. M. Bylander, and L. Kleinman, "Nonlocal Hermitian norm-conserving Vanderbilt pseudopotential," *Phys. Rev. B*, vol. 47, no. 11, pp. 6728–6731, Mar 1993. [Online]. Available: http://dx.doi.org/10.1103/physrevb.47.6728

[48] A. Schleife, E. W. Draeger, Y. Kanai, and A. A. Correa, "Plane-wave pseudopotential implementation of explicit integrators for time-dependent Kohn-Sham equations in large-scale simulations," *J. Chem. Phys.*, vol. 137, no. 22, p. 22A546, 2012. [Online]. Available: http://dx.doi.org/10.1063/1.4758792

[49] A. Schleife, E. W. Draeger, V. M. Anisimov, A. A. Correa, and Y. Kanai, "Quantum Dynamics Simulation of Electrons in Materials on High-Performance Computers," *Computing in Science & Engineering*, vol. 16, no. 5, pp. 54–60, Sep 2014. [Online]. Available: http://dx.doi.org/10.1109/mcse.2014.55

[50] R. Car and M. Parrinello, "Unified Approach for Molecular Dynamics and Density-Functional Theory," *Phys. Rev. Lett.*, vol. 55, no. 22, pp. 2471–2474, Nov 1985. [Online]. Available: http://dx.doi.org/10.1103/physrevlett.55.2471

[51] G. Kresse and J. Furthmüller, "Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set," *Phys. Rev. B*, vol. 54, no. 16, pp. 11 169–11 186, Oct 1996. [Online]. Available: http://dx.doi.org/10.1103/physrevb.54.11169

[52] M. E. Tuckerman and M. Parrinello, "Integrating the Car-Parrinello equations. I. Basic integration techniques," *J. Chem. Phys.*, vol. 101, no. 2, p. 1302, 1994. [Online]. Available: http://dx.doi.org/10.1063/1.467823

[53] X. Andrade, A. Castro, D. Zueco, J. L. Alonso, P. Echenique, F. Falceto, and Á. Rubio, "Modified Ehrenfest Formalism for Efficient Large-Scale ab initio Molecular Dynamics," *J. Chem. Theory Comput.*, vol. 5, no. 4, pp. 728–742, Apr 2009. [Online]. Available: http://dx.doi.org/10.1021/ct800518j

[54] F. Gygi, C. W. Ueberhuber, J. Lorenz, E. W. Draeger, M. Schulz, B. R. de Supinski, J. A. Gunnels, V. Austel, J. C. Sexton, F. Franchetti, and S. Kral, "Gordon Bell finalists I—Large-scale electronic structure calculations of high-Z metals on the Blue Gene/L platform," in *Proceedings of the 2006 ACM/IEEE conference on Supercomputing - SC '06*. Institute of Electrical & Electronics Engineers (IEEE), 2006. [Online]. Available: http://dx.doi.org/10.1145/1188455.1188502

[55] A. Bhatele, T. Gamblin, S. H. Langer, P.-T. Bremer, E. W. Draeger, B. Hamann, K. E. Isaacs, A. G. Landge, J. A. Levine, V. Pascucci, M. Schulz, and C. H. Still, "Mapping applications with collectives over sub-communicators on torus networks," in *2012 International Conference for High Performance Computing Networking, Storage and Analysis*. Institute of Electrical & Electronics Engineers (IEEE), Nov 2012. [Online]. Available: http://dx.doi.org/10.1109/sc.2012.75

[56] J. A. Gunnels, "Making Good Better: Addressing the Multiple Objectives of High-Performance Parallel Software with a Mixed Global-Local Worldview," http://icerm.brown.edu/tw12-1-exascale.

[57] I.-H. Chung, T. N. Sainath, B. Ramabhadran, M. Picheny, J. Gunnels, V. Austel, U. Chauhari, and B. Kingsbury, "Parallel Deep Neural Network Training for Big Data on Blue Gene/Q," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 745–753. [Online]. Available: http://dx.doi.org/10.1109/SC.2014.66

[58] C. Kreisbeck, T. Kramer, and A. Aspuru-Guzik, "Scalable High-Performance Algorithm for the Simulation of Exciton Dynamics. Application to the Light-Harvesting Complex II in the Presence of Resonant Vibrational Modes," *J. Chem. Theory Comput.*, vol. 10, no. 9, pp. 4045–4054, Sep 2014. [Online]. Available: http://dx.doi.org/10.1021/ct500629s

[59] J. Lienig, "Interconnect and current density stress," in *Proceedings of the 2005 international workshop on System level interconnect prediction - SLIP '05*. Association for Computing Machinery (ACM), 2005. [Online]. Available: http://dx.doi.org/10.1145/1053355.1053374

[60] D. Dundas, E. J. McEniry, and T. N. Todorov, "Current-driven atomic waterwheels," *Nature Nanotech*, vol. 4, no. 2, pp. 99–102, Feb 2009. [Online]. Available: http://dx.doi.org/10.1038/nnano.2008.411

[61] X. Andrade, J. N. Sanders, and A. Aspuru-Guzik, "Application of compressed sensing to the simulation of atomic systems," *Proceedings of the National Academy of Sciences*, vol. 109, no. 35, pp. 13 928–13 933, Aug 2012. [Online]. Available: http://dx.doi.org/10.1073/pnas.1209890109