Feature Article: Molecular Dynamics Simulation

Scalable Reactive Molecular Dynamics Simulations for Computational Synthesis

Ying Li
Argonne National Laboratory

Ken-Ichi Nomura
University of Southern California

Joseph A. Insley
Argonne National Laboratory

Vitali Morozov
Argonne National Laboratory

Kalyan Kumaran
Argonne National Laboratory

Nichols A. Romero
Argonne National Laboratory

William A. Goddard III
California Institute of Technology

Rajiv K. Kalia
University of Southern California

Aiichiro Nakano
University of Southern California

Priya Vashishta
University of Southern California

***Abstract*—Reactive molecular dynamics (MD) simulation is a powerful research tool for describing chemical reactions. We eliminate the speed-limiting charge iteration in MD with a novel extended-Lagrangian scheme. The extended-Lagrangian reactive MD (XRMD) code drastically improves energy conservation while substantially reducing time-to-solution. Furthermore, we introduce a new polarizable charge equilibration (PQEq) model to accurately predict atomic charges and polarization. The XRMD code based on hybrid message passing+multithreading achieves a weak-scaling parallel efficiency of 0.977 on 786 432 IBM Blue Gene/Q cores for a 67.6 billion-atom system. The performance is portable to the second-generation Intel Xeon Phi, Knights Landing. Blue Gene/Q simulations for the computational synthesis of materials via novel exfoliation**

Digital Object Identifier 10.1109/MCSE.2018.110150043
Date of publication 11 January 2018; date of current version
15 August 2019.

1521-9615 © 2018 IEEE

mechanisms for synthesizing atomically thin transition metal dichalcogenide layers will dominate nanomaterials science in this century.

Molecular dynamics (MD) simulations play pivotal roles in physics, chemistry, biology, and materials science by describing material properties and processes in atomistic details. MD simulations follow time evolution of the positions, $\mathbf{r}^N = \{\mathbf{r}_i \mid i = 1, \ldots, N\}$, of N atoms by numerically integrating Newton's equations of motion, where the atomic force law is mathematically encoded in the interatomic potential energy $E(\mathbf{r}^N)$.[1] Reliable interatomic potentials are a key to accurately describing thermomechanical properties of materials. To describe material processes involving chemical reactions, in particular, quantum molecular dynamics (QMD) simulations[2] determine the force law by minimizing the potential energy

$$E(\mathbf{r}^N) = \min_{\psi^{N\mathrm{el}}} E(\mathbf{r}^N, \psi^{N\mathrm{el}}) \tag{1}$$

as a function of electronic wave functions, $\psi^{N\mathrm{el}}(\mathbf{r}) = \{\psi_n(\mathbf{r}) \mid n = 1, \ldots, N_{\mathrm{el}}\}$ (N_{el} is the number of wave functions of the order of N), often in the framework of density functional theory (DFT).[3] Despite remarkable progresses in $O(N)$ DFT algorithms,[4] the largest QMD simulations to date have been limited to $N \sim 10^4$ for the duration of $\tau = 10^{-11}$ s.[5] Unfortunately, the spatiotemporal scales covered by QMD simulations are not sufficient for studying the synthesis of advanced materials, such as functional layered materials (LM), which have outstanding electronic, optical, magnetic, and chemical properties. To rationally guide the synthesis of stacked LMs by chemical vapor deposition, exfoliation and intercalation, "computational synthesis" should extend beyond $N \sim 10^9$ and $\tau = 10^{-9}$ s.

To extend the limited spatiotemporal scales covered by QMD simulations, reactive molecular dynamics (RMD) simulation methods[6] have been developed based on the first principles-informed reactive force field (ReaxFF) approach.[7,8] The ReaxFF approach significantly reduces the computational cost of simulating chemical reactions, while reproducing the energy surfaces and barriers as well as charge distributions of DFT calculations. RMD simulations describe formation and breakage of chemical bonds using reactive bond orders (BOs).[7,9] In the ReaxFF approach, the potential energy thus comprises valence interactions described with BO as well as noncovalent interactions as follows:[10]

$$\begin{aligned} &E(\{\mathbf{r}_{ij}\}, \{\mathbf{r}_{ijk}\}, \{\mathbf{r}_{ijkl}\}, \{q_i\}, \{\mathrm{BO}_{ij}\}) \\ &\quad = E_{\mathrm{bond}} + E_{lp} + E_{\mathrm{over}} + E_{\mathrm{under}} \\ &\qquad + E_{\mathrm{val}} + E_{\mathrm{pen}} + E_{\mathrm{coa}} + E_{\mathrm{tors}} \\ &\qquad + E_{\mathrm{conj}} + E_{h\mathrm{bond}} + E_{vdW} + E_{\mathrm{Coulomb}} \end{aligned} \tag{2}$$

in which the valence interactions include the bonding energy E_{bond}, lone-pair energy E_{lp}, overcoordination energy E_{over}, undercoordination energy E_{under}, valence-angle energy E_{val}, penalty energy E_{pen}, 3-body conjugation energy E_{coa}, torsion-angle energy E_{tors}, 4-body conjugation energy E_{conj}, and hydrogen bonding energy E_{hbond}. In (2), the total energy is a function of relative positions of atomic pairs $\mathbf{r}_{ij}$, triplets $\mathbf{r}_{ijk}$, and quadruplets $\mathbf{r}_{ijkl}$, as well as atomic charges q_i and bond orders BO_{ij} between atomic pairs. Here, it should be noted that BO_{ij} themselves are functions of $\{\mathbf{r}_i\}$. Accordingly, the computation of interatomic forces

$$F_i = -\frac{\partial}{\partial \mathbf{r}_i} E \tag{3}$$

involves atomic n-tuples for $n \leq 4$ explicitly and $n \leq 6$ due to chain-rule differentiation. The noncovalent interactions comprise van der Waals (vdW) energy E_{vdW} and Coulombic energy E_{Coulomb}, which are screened by a taper function.[7]

The most intensive computation in RMD simulations arises from a charge-equilibration (QEq) scheme[11,12] to describe charge transfer between atoms. QEq treats atomic charges as dynamic variables $q^N = \{q_i \mid i = 1, \ldots, N\}$. The charges and the resulting force law are determined by minimizing the potential energy as

$$E(\mathbf{r}^N) = \min_{q^N} E(\mathbf{r}^N, q^N) \tag{4}$$

with respect to q^N at every MD time step. This variable N-charge problem is commonly solved iteratively, e.g., with the conjugate gradient (CG) method,[13] where a large number of CG iterations are required to reach sufficient convergence of charges q^N to guarantee the conservation of the total energy as a function of time. Insufficiently converged charges act as an artificial heat sink of energy, and the resulting broken time reversibility causes the total energy to drift over time. A similar tradeoff between the computational speed and energy conservation is encountered in QMD simulations, where insufficient convergence of the iterative refinement of wave functions $\psi^{N\mathrm{el}}$ in (1) causes serious energy drift. Cawkwell and Niklasson proposed an extended-Lagrangian scheme[14,15] that achieves excellent long-time energy conservation with drastically reduced number of iterations. In fact, an extended-Lagrangian scheme with no iteration (i.e., requiring only one evaluation of energy gradient per MD time step) has recently been demonstrated.[16]

In this paper, we report successful elimination of the speed-limiting charge iteration in MD simulations by adapting the extended-Lagrangian scheme. The resulting extended-Lagrangian reactive MD (XRMD) simulation code has drastically improved energy conservation while substantially reducing the time-to-solution. The parallel XRMD code has achieved unprecedented scalability and orders-of-magnitude improvement of the time-to-solution over the previous state-of-the-art based on the hybrid message passing + multithreading implementation and various code transformations. Here, it should be noted that the extended-Lagrangian method is used in tandem with highly efficient preconditioned conjugate-gradient (PCG) methods[17,18] to achieve a high level of convergence.

To improve the fidelity of MD simulations, we have recently proposed a new general methodology, called polarizable charge equilibration (PQEq) model,[19] which is capable of predicting rapidly and dynamically accurate atomic charges and polarization under electric fields. In the PQEq, the atomic-sized Gaussian-shaped electron density (shell) is used for descripting the polarization, which can be polarized away from the core in response to internal and external electric fields. The resulting new ReaxFF model is denoted as polarizable reactive force-field (ReaxPQ). In ReaxPQ, all electrostatic interactions are described with the finite-sized charge distributions of PQEq, rather than as point charges as in the original ReaxFF. We have developed efficient schemes to update the atomic charges on the cores and assume massless shell so that it relaxes instantaneously to its zero-force position, with no inertial delay.

ALGORITHMIC INNOVATIONS

Extended-Lagrangian QEq Scheme

The extended-Lagrangian QEq scheme introduces auxiliary variables $\theta^N = \{\theta_i \mid i = 1, \ldots ,N\}$. System dynamics for XRMD simulations is derived from the extended Lagrangian as

$$L_{\mathrm{XRMD}} = L_{\mathrm{RMD}} + \frac{\mu}{2}\sum_i \dot{\theta}_i^2 - \frac{\mu\omega^2}{2}\sum_i (\theta_i - q_i)^2 \quad (5)$$

where L_{RMD} is the RMD Lagrangian as follows:

$$L_{\mathrm{RMD}} = \sum_i \frac{m_i}{2}\dot{\mathbf{r}}_i^2 - E(\mathbf{r}^N, q^N) \quad (6)$$

with m_i being the mass of the ith atom and the dot denoting time derivative. In (5), μ and ω are fictitious mass and frequency parameters for the auxiliary charge degrees of freedom.

The time evolution of the dynamical system described by L_{XRMD} is determined by Euler–Lagrange equations of motion. In the limit of $\mu \to 0$, the equations become

$$m_i\ddot{\mathbf{r}}_i = -\frac{\partial}{\partial \mathbf{r}_i}E(\mathbf{r}^N, q^N) \quad (7)$$

$$\ddot{\theta}_i = \omega^2(q_i - \theta_i). \quad (8)$$

We numerically integrate (7) and (8) using the velocity Verlet algorithm with a unit time step of δt. At each time step, we perform just one step of CG iteration toward minimizing the Coulombic energy as a function of q^N. We use the auxiliary variable θ^N as an initial guess for the CG optimization of q^N. For extended-Lagrangian-based QMD simulations, the recommended value for the dimensionless parameter, $K = \omega^2\delta t^2$, is 2.[15] We have found that the same value works for XRMD as well, and $K = 2.0$ will be use in the numerical tests discussed next.

PQEq Method

As described in the introduction, the interatomic potential energy $E(\mathbf{r}^N, q^N)$ in ReaxFF is composed of a number of bonded and nonbonded terms.[7,8] The nonbonded terms are

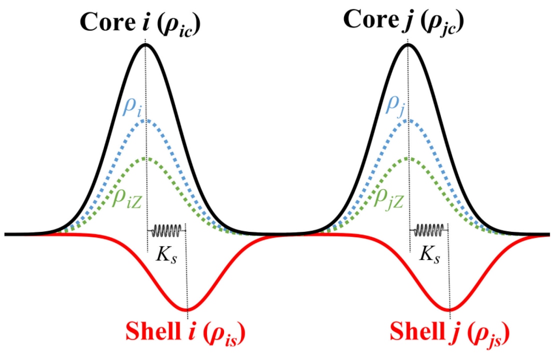

Figure 1. Partition of a two-atom system into core and shell in the PQEq model.

Coulombic and vdW energies. Here, we describe the Coulombic energy using the new PQEq methodology.[19] In PQEq, each atom *i* is partitioned into two charged sites (i.e., core and shell). The core (ρ_{ic}) consists of two parts: ρ_i with a variable total charge q_i and ρ_{iZ} with a fixed total charge Z_i. The shell ρ_{is} is massless and has a fixed total charge of $-Z_i$. The shell and core of an atom are connected by an isotropic harmonic spring with force constant K_s (see Figure 1).

Defining the total charge (core plus shell) on the *i*th atom as q_i, the individual charges on the core and shell are $q_{ic} = q_i + Z_i$ and $q_{is} = -Z_i$, respectively. The Coulombic energy is expressed as

$$E_{\text{Coulomb}}(\{\mathbf{r}_{ic}, \mathbf{r}_{is}, q_i\}) = \sum_i^N \left[\chi_i^0 q_i + \frac{1}{2} J_{ii}^0 q_i^2 + \frac{1}{2} K_s r_{ic,is}^2\right] + \sum_{ik>jl} T(r_{ik,jl}) C_{ik,jl}(r_{ik,jl}) q_{ik} q_{jl} \tag{9}$$

where *i* and *j* are atomic indices, and *k* and *l* represent core (c) or shell (s). In (9), $r_{ik,jl} = |\mathbf{r}_{ik} - \mathbf{r}_{jl}|$ is the distance between the *i*th atom's core or shell with the *j*th atom's core or shell. χ_i^0 is the Mulliken electronegativity and J_{ii}^0 is the idempotential (hardness) or electron capacity of the *i*th atom. The second sum is the pairwise shielded Coulombic interaction energy between all cores and shells. In (9), the electrostatic energy between two Gaussian charges is given by

$$C_{ik,jl}(r_{ik,jl}) = \frac{1}{r_{ik,jl}} \operatorname{erf}\left(\sqrt{\frac{\alpha_{ik}\alpha_{jl}}{\alpha_{ik} + \alpha_{jl}}} r_{ik,jl}\right) \tag{10}$$

where the width of the distribution is given by $\alpha_{ik} = \lambda / R_{ik}^2$. Here, R_{ik} is the covalent atomic radius and λ is a parameter that converts the overlap of two Gaussian charges to an effective shielding. In the case of $\mathbf{r}_{ij} = \mathbf{r}_{kl}$, $C_{ik,jl}$ reduces to

$$C_{ik,jl}^0 = \lim_{r \to 0} C_{ik,jl}(\mathbf{r}) = \frac{2}{\sqrt{\pi}} \sqrt{\frac{\alpha_{ik}\alpha_{jl}}{\alpha_{ik} + \alpha_{jl}}} \tag{11}$$

which means that the Gaussian shielding in PQEq results in a finite Coulombic interaction energy even in the limit of zero interatomic distance. Equation (9) can be expanded to give the total electrostatic energy as

$$\begin{aligned} E_{\text{Coulomb}}(\{\mathbf{r}_{ic}, \mathbf{r}_{is}, q_i\}) = {} & \sum_i^N \left[\chi_i^0 q_i + \frac{1}{2} J_{ii}^0 q_i^2 + \frac{1}{2} K_s r_{ic,is}^2\right] \\ & + \sum_{i>j} [T(r_{ic,jc}) C_{ic,jc}(\mathbf{r}_{ic,jc}) q_{ic} q_{jc} \\ & - T(r_{ic,js}) C_{ic,js}(\mathbf{r}_{ic,js}) q_{ic} Z_j \\ & - T(r_{is,jc}) C_{is,jc}(\mathbf{r}_{is,jc}) q_{jc} Z_i \\ & + T(r_{is,js}) C_{is,js}(\mathbf{r}_{is,js}) Z_i Z_j]. \end{aligned} \tag{12}$$

The Coulombic interaction is screened using a taper function *T*(*r*), which has a cutoff length r_{cnb} as

$$T(r_{ik,jl}) = \sum_{\alpha=0}^{7} T_\alpha \left(\frac{r_{ik,jl}}{r_{cnb}}\right)^\alpha. \tag{13}$$

In the original ReaxFF, the electronegativity equalization method (EEM) was used to calculate charges. Later, EEM was replaced by the original QEq model in the LAMMPS and PuReMD codes.[17] With the new ReaxPQ, we now calculate all electrostatic interactions using finite-sized charge distributions.

In ReaxPQ, the atomic charges q_i are variables that change dynamically in time. When atomic positions are updated during MD simulation, the PQEq subroutine updates charge distribution q^N by minimizing E_{Coulomb} subject to the conditions that the chemical potentials $\partial E_{\text{Coulomb}}/\partial q_i$ are equal for all atoms (which provides $N-1$ conditions) and that the total charge is conserved

$$\sum_i (q_{ic} + q_{is}) = \sum_i q_i = Q \tag{14}$$

where *Q* is the total charge of the system. We use Lagrange multipliers to guarantee this constraint as the charges are optimized

$$g_i \equiv -\frac{\partial E_{\text{Coulomb}}}{\partial q_i} = -\mu_{\text{ec}} \tag{15}$$

where μ_{ec} is the electrochemical potential. We solve this problem iteratively using a PCG method, which has shown satisfactory performance, stability, and accuracy for various model systems.[20] As shown in the following, we couple the PCG method with shell relaxation to calculate the PQEq charges while updating the shell position. The shell position for each atom is obtained by balancing the effect of the electrostatic field due to all external atoms with intra-atomic interactions involving only the core and shell

$$F_{\mathrm{intra}} = -\frac{\partial}{\partial r_{is}}\left(\frac{1}{2}K_s r_{ic,is}^2\right) \tag{16}$$

$$F_{\mathrm{external}} = -\frac{\partial}{\partial r_{is}}\left\{\sum_{ik>jl} T(r_{ik,jl})C_{ik,jl}(r_{ik,jl})q_{ik}q_{jl}\right\}. \tag{17}$$

We solve (16) and (17) to find the optimal positions of shells $\mathbf{r}_{is}$ using a single iteration of the Newton–Raphson method. We assume here that the shell is massless so that it relaxes instantaneously to its zero-force position. The PQEq parameters are derived from valence-averaged experimental ionization potential and electron affinity data and standard bond distances for all elements up to Nobelium (atomic number = 102).

PQEq has been validated by comparing to quantum–mechanical (QM) interaction energies as probe dipoles are brought along various directions to various molecules.[19] It was found that the interaction energies from PQEq are in excellent agreement with QM results, much better than other common charge models, generally fixed QM charges based on Mulliken or electrostatic potential models, and those from standard force fields.

Parallelization

We have implemented the extended-Lagrangian QEq scheme previously in a scalable parallel reactive MD simulation code. The new PQEq methodology described previously requires several modifications of the MPI parallelization scheme described in.[21] For large granularity (the number of atoms per spatial domain, $N/D > 10^2$, where D specifies the number of spatial domain), simple spatial decomposition (i.e., each MPI rank is responsible for the computation of the forces on the atoms within its subsystem) suffices, whereas for finer granularity ($N/D \sim 1$), neutral-territory[22] or other hybrid decomposition schemes[23–25] can be incorporated into the framework. Our parallelization framework also includes load-balancing capability. For irregular data structures, the number of atoms assigned to each MPI rank varies significantly, and this load imbalance degrades the parallel efficiency. We minimize the load-imbalance cost as well as the size and number of messages. We employ a topology-preserving spatial decomposition, which allows message passing to be performed in a structured way in only six steps, so that the number of messages (hence the latency) is minimized.

Within each computing node, we introduce an additional layer of shared-memory parallelism using the Open Multi-Processing (OpenMP) application programming interface. OpenMP allows XRMD to take advantage of the simultaneous multithreading support provided by modern processors, such as the PowerPC A2 architecture of IBM Blue Gene/Q and the second-generation Intel Xeon Phi named Knights Landing (KNL), to achieve better utilization of the computing resources within each core. With multithreading, the most computationally expensive BO and force computations within XRMD are greatly accelerated, serving to reduce the overall runtime. A secondary benefit of multithreading is that it allows MPI ranks to be exchanged for local threads, thereby reducing the total number of ranks in an MPI job and similarly reducing the communication and atom-caching overheads at large scales.

PERFORMANCE PLATFORM

Experimental Platform

Initial numerical tests were performed on an IBM Blue Gene/Q supercomputer[26] at the Argonne Leadership Computing Facility (ALCF) named Mira. This supercomputer consists of 48 racks, each with 1024 nodes. Each Blue Gene/Q node has a 16-core processor operating at 1.6 GHz, 32 MB L2 cache with hardware transactional memory and speculative operation functionalities, and 16 GB DDR3 memory. The processor employs the PowerPC A2 architecture that supports quad floating-point processing units, 16 KB L1 instruction cache, 16 KB L1 data cache, and 4-way multithreading per core enabling 64 concurrent threads on one node. Although it is highly energy efficient

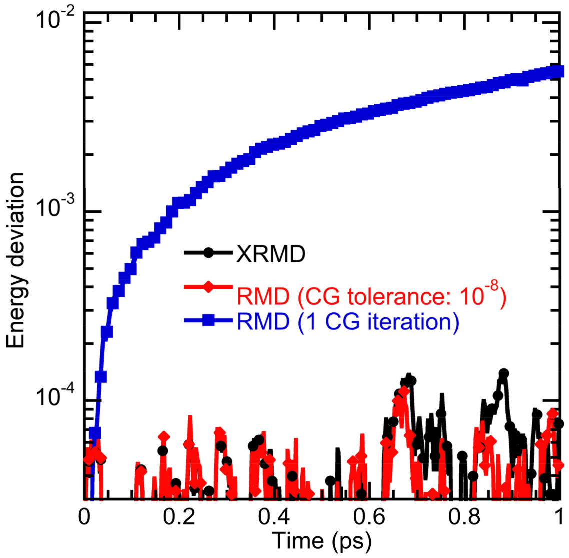

Figure 2. Absolute value of the deviation of the total energy from its initial value (normalized by the initial total energy) as a function of time for the new XRMD code (black curve), compared with two results using the original RMD code: converged CG iterations with a tolerance of 10^{-8} (red) and one CG iteration per MD step (blue).

(55 W per node) due to the relatively low clock speed, the Blue Gene/Q chip delivers a peak performance of 204.8 GFLOP/s. At every clock cycle, each PowerPC A2 core can issue one instruction to the auxiliary execution unit (XU) for floating-point operations and another to the main XU for load/store, branch and integer operations, etc. It thus requires at least two instruction streams per core to maximize the instruction throughput. Assigning four threads per core could further improve the performance by hiding latency, assuming that the memory bandwidth is not saturated.

Additionally, we perform benchmarks of the XRMD code on ALCF's Intel Xeon Phi KNL based supercomputer named Theta. KNL is Intel's second-generation processor in the Xeon Phi series.[27] On Theta, each computing node contains 64 cores and each core has 4 hardware threads. Each core has 32 KB L1 instruction cache, 32 KB L1 data cache, and two independent floating-point units (FPUs) capable of executing 512-b-wide SIMD instructions. The peak instruction throughput of the KNL microarchitecture is two instructions per clock cycle, and they can be issued back-to-back from the same thread. Two cores and a shared 1 MB L2 cache form a tile, and the tiles are connected with a two-dimensional (2-D) mesh network-on-chip with 16 GB of high-bandwidth in-package multichannel DRAM memory (MCDRAM) and 192 GB of regular DRAM.

SCALABILITY AND TIME-TO-SOLUTION

Algorithmic Improvements

We first validate the extended-Lagrangian QEq scheme using oxidation of a silicon carbide nanoparticle (nSiC)[28] as an example, where the initial oxidation is found to produce a molten silica shell that acts as an autocatalytic "nanoreactor"[29,30] by actively transporting oxygen while protecting the nanocarbon product from harsh oxidizing environment. Figure 2 shows the deviation of the total energy from its initial value as a function of time during simulation, which specifies how a few drifts could be yielded for production science. We compare three cases: 1) the new XRMD code; 2) the original RMD code, where CG iterations at each MD step are continued until the change in the Coulombic energy E_{Coulomb} per iteration falls below 10^{-8} of $|E_{\mathrm{Coulomb}}|$; and 3) the RMD code performing only one CG iteration per MD step. Total energy conservation of XRMD is comparable to well-converged RMD, while RMD with single CG step exhibits significant energy drift. Most importantly, the XRMD total energy only fluctuates around an average value but does not exhibit a systematic drift, signifying the long-time stability of the simulation. As demonstrated above, XRMD provides sufficient accuracy that is comparable to a well-converged RMD but at much less computational cost, which could reduce the number of CG iterations on average by a factor of $10{\sim}10^2$.

Weak and Strong Scaling on Blue Gene/Q

We perform an isogranular-scaling test of the XRMD code with an MPI-only version on the Blue Gene/Q, in which the number of atoms per core N/P is kept constant. To fully utilize the available cores on the Blue Gene/Q, the pure MPI implementation takes one MPI rank per core, 16 ranks per node. We measure the wall-clock time per XRMD simulation step with scaled workloads—86,016P-

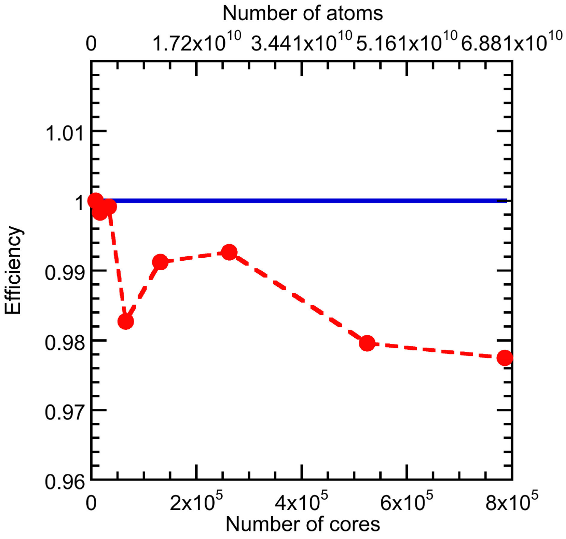

Figure 3. Weak-scaling parallel efficiency of the parallel XRMD code, with scaled workloads—86 016*P*-atom RDX system on *P* cores (*P* = 8192,. . .,786 432) of Blue Gene/Q. The measured efficiency values (solid red circles) are compared with the ideal efficiency of 1 (blue line).

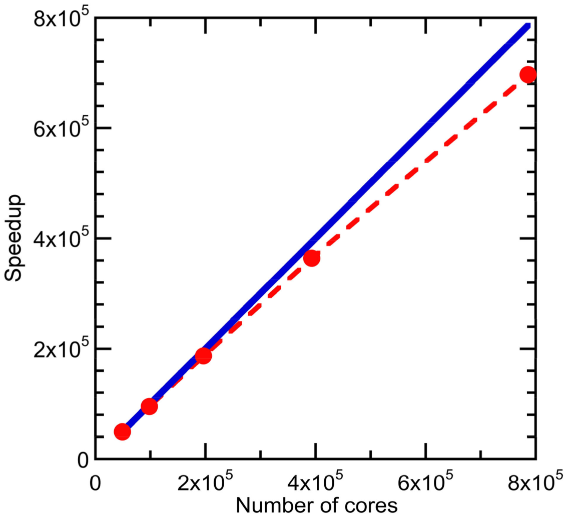

Figure 4. Strong-scaling speedup of the parallel XRMD code with a fixed problem size—4 227 858 432-atom RDX system on *P* cores (*P* = 49 152,. . .,786 432) of Blue Gene/Q. The measured speedup values (solid red circles) are compared with the ideal speedup (blue).

atom cyclotrimethylenetrinitramine ($C_3H_6N_6O_6$ or RDX) molecular crystal on *P* cores of Blue Gene/Q. By increasing the number of atoms linearly with the number of cores, the wall-clock time remains almost constant, indicating excellent weak scalability. To quantify the parallel efficiency, we first define the speed of the XRMD code as a product of the total number of atoms and the number of MD steps executed per second. The isogranular (or weak-scaling) speedup is given by the ratio between the speed of *P* cores and that of 8192 cores as a reference system. The weak-scaling parallel efficiency is the isogranular speedup divided by *P*/8,192. With the granularity of 86 016 atoms per core, the parallel efficiency is 0.977 on $P =$ 786 432 for a 67 645 734 912-atom system. Figure 3 shows a very high scalability for the XRMD code.

We next perform a strong-scaling test by simulating the RDX crystal containing a total of 4 227 858 432 atoms. In this test, the number of cores ranges from $P =$ 49 152 to 786 432, while keeping the total problem size constant. We measure the wall-clock time per MD simulation step as a function of *P*. The time-to-solution is reduced by a factor of 14.17 on 786 432 cores compared with the 49 152-core run (i.e., using 16-times larger number of cores). This signifies a strong-scaling speedup of 14.17, with the corresponding strong-scaling parallel efficiency of 0.886. Figure 4 shows the measured strong-scaling speedup as a function of the number of cores (solid red circles), while the blue line denotes the ideal speedup. It is more difficult to achieve high strong-scaling parallel efficiency compared with weak-scaling parallel efficiency, as a comparison of Figures 3 and 4 suggests. This is due to decreasing granularity, and accordingly increasing communication/computation ratio for larger number of processors, in the former. With a system size 16 times smaller than the weak-scaling test, the observed strong-scaling parallel efficiency is considered excellent.

Performance on KNL

We first port the pure MPI implemented XRMD code onto a KNL-based supercomputer, Theta, to verify the portability and to compare the performance. Then, we also measure the performance of the XRMD code on Theta, using the newly implemented hybrid MPI + OpenMP approach for ReaxFF BO and force calculations. Here, we use a fine granularity to test an alternative high-throughput scenario that acquires an extremely fast turnaround.

Table 1. Wall-clock time of 1000 MD steps for 96 768-atom RDX system using four computing nodes of Blue Gene/Q versus one computing node of KNL, both from the pure MPI implementation, with varying configurations of hardware threads. The shortest wall-clock time on each platform is marked by boldface.

Platform	Number of nodes	Cores/node	MPI ranks/ core	Wall-clock time (s)	Speed-up
Blue Gene/Q	4	16	1	2173.6	1
	4	16	2	**1638.7**	1.33
	4	16	4	1849.8	1.18
KNL	1	64	1	2319.5	1
	1	64	2	1325.3	1.75
	1	64	4	**791.3**	2.93

In the first experiment, we study the capabilities of a core to execute multiple processes simultaneously. Even though both Blue Gene/Q's Power A2 and Theta's KNL processors can execute up to two instructions per clock cycle and have nearly identical frequencies, reaching peak instruction throughput requires different execution conditions. In all experiments, we simulate RDX crystal with 96 768 atoms. We use four computing nodes of Blue Gene/Q versus 1 node of KNL in order to keep the total number of cores to be 64 on both systems. We place 1, 2, and 4 MPI ranks on each core for the pure MPI implementation so that the level of concurrency given to each core is also the same on Power A2 and on KNL.

The results of this test are shown in Table 1. As expected, placing 1 MPI rank on a core of either system does not provide a good performance. On Blue Gene/Q, we must have at least two processes running on a core to have peak instruction throughput, but placing four processes eventually put more pressure on shared resources. On the other hand, the KNL architecture allows having the peak instruction throughput even with one process on a core; however, long data-dependency chains prevent instructions from progressing, which degrade the performance. Placing more processes on a KNL core increases the instruction-level parallelism and gives more opportunities to fill the execution pipelines. Thus, it is not surprising that core utilization is maximal when two processes are running on a Power A2 core, and when four processes are running on a KNL core (denoted by the wall-clock times in boldface in Table 1). As expected, a 2-FPU KNL core is about twice faster than a 1-FPU Power A2 core.

We next study the effect of the number of OpenMP threads on the time-to-solution on KNL. Figure 5 shows the wall-clock time of 1000 MD simulation steps as a function of the number of computing nodes m on KNL, with varying numbers of OpenMP threads per core t. We see nearly constant wall-clock times from $m = 2$ to 256,

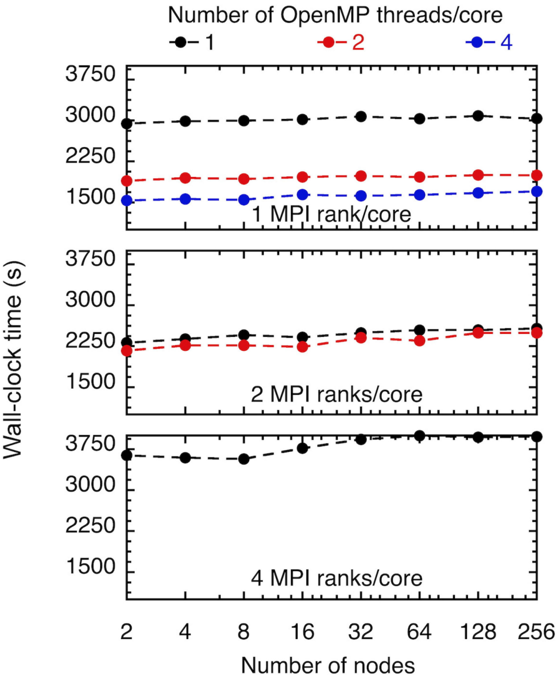

Figure 5. Wall-clock time of the hybrid MPI + OpenMP implementation of the XRMD code for 1000 time steps with scaled workloads—96 768m-atom RDX system on m computing nodes (m = 2, 4, 8, . . . , 256) of KNL. Different curves correspond to different numbers of threads per core, t = 1 (black), 2 (red), 4 (blue).

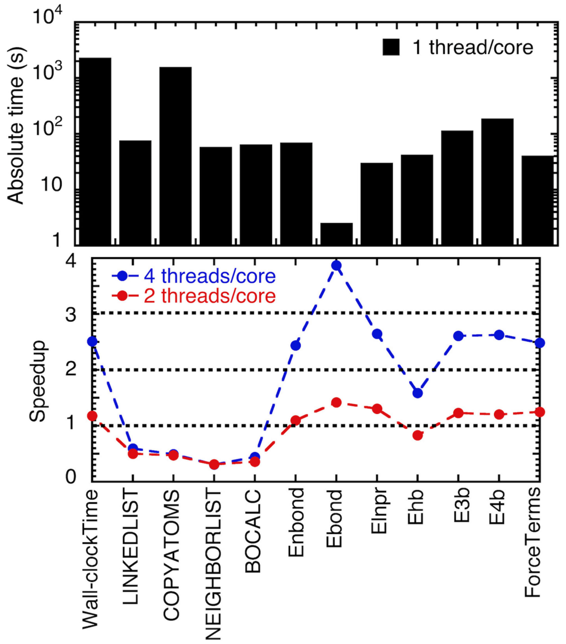

Figure 6. Absolute time of the most time-consuming subroutines in XRMD code for 1000 MD steps on 1 node of KNL for $t = 1$ thread per core for 96 768-atom RDX system in logarithmic scale. Also shown are the speedup of those subroutines and the wall-clock time in the XRMD code by increasing the number of OpenMP threads on KNL. The red and blue circles denote the speedups for the numbers of threads per core, $t = 2$ and 4, respectively.

indicating a high weak-scaling (isogranular) speedup for all values of *t*. We also observe decreasing wall-clock times for increasing numbers of threads *t*. However, the relative speedup of using two OpenMP threads per core is less than 2 compared to one OpenMP thread per core for KNL.

To understand why the speedup is sublinear to the number of OpenMP threads per core for KNL, we analyze the relative speedups of the most time-consuming subroutines in the XRMD code by increasing the number of OpenMP threads per core, as shown in Figure 6. It is clearly seen that ReaxFF force calculations (e.g., Enbond, Ebond, Elnpr, Ehb, E3b, E4b, ForceTerms) scale well with respect to the increasing number of OpenMP threads. In contrast to these highly numerical subroutines, less-numerically intensive list-processing subroutines such as LINKEDLIST and NEIGHBORLIST are less scalable and would require further optimization.

COMPUTATIONAL SYNTHESIS APPLICATIONS

The accurate description of atomic charges and polarization by the new ReaxPQ method, combined with the extended-Lagrangian method to eliminate the speed-limiting charge iterations and their scalable parallel implementations, have opened a new avenue for computational synthesis of novel materials. This section demonstrates the capability of the XRMD code for computational synthesis of atomically thin LMs. The attractiveness of LMs lies not only in their outstanding electronic, optical, magnetic, and chemical properties, but also in the possibility of tuning these properties in desired ways by building vdW heterostructures composed of unlimited combinations of atomically thin layers.

XRMD simulations can guide the synthesis of stacked LMs by exfoliation. Liquid exfoliation is an ideal route for synthesizing 2-D materials such as graphene, BN, and transition metal dichalcogenides (TMDC). Although several experimental studies have reported the selection and optimization of exfoliated LMs, the exfoliation mechanism is not completely understood and hence it is difficult to engineer the size, number of layers, and quality of the exfoliated material. Our hypothesis is that sonication generates cavitation bubbles in a liquid and collapsing bubbles produce high-energy water jets that break up bulk materials into nanosheets. While previous XRMD simulations showed bubble collapse-induced damage of solid surfaces, neither experimental nor theoretical work has demonstrated the hypothesized bubble collapse-induced exfoliation.

To study possible exfoliation of TMDC due to the collapse of cavitation bubbles in solution, we perform XRMD simulations of an archetypal TMDC, molybdenum disulfide (MoS_2), in water–isopropyl alcohol mixture, which is a solution commonly used for exfoliation (see Figure 7). A MoS_2 crystal, which is a stack of atomically thin MoS_2 layers, is placed in a 53:47 mixture of water and isopropyl alcohol (in weight), and a cavitation bubble is created by removing the liquid molecules inside a sphere of radius 47 Å, as shown in Figure 7(a). A planar shock wave is applied to cause rapid collapse of the bubble.

In one simulation, where the MoS_2 sheets are parallel to the shock wave, the shock velocity v_s is

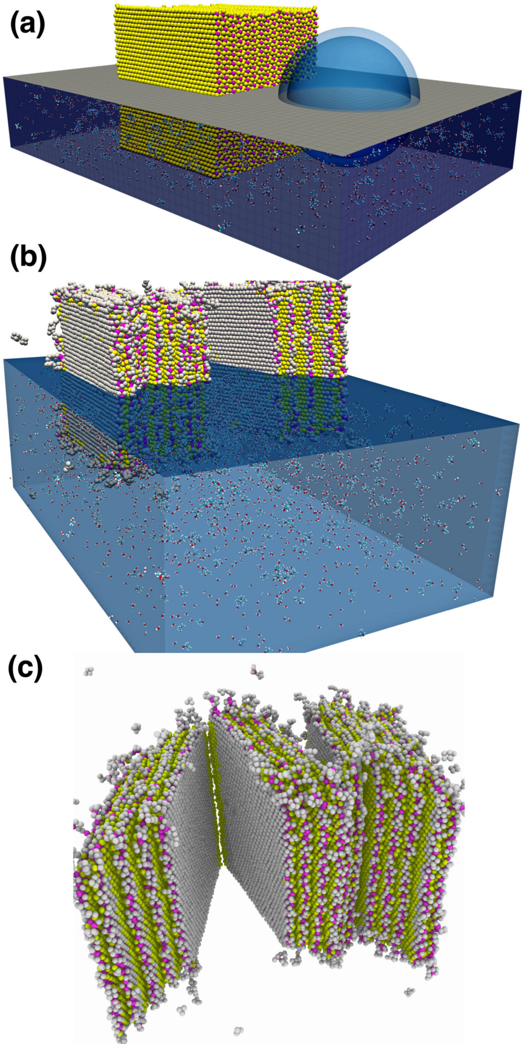

Figure 7. Exfoliation of MoS_2 assisted by cavitation bubble collapse. The atoms are colored as Mo: magenta, S: yellow, H: white, C: cyan, and O: red. For the clarity of presentation, only 4% of the water and alcohol molecules are shown in the bottom half of the simulation box. (a) Initial simulation setup, where the spherical bubble is colored blue. (b) Bulk MoS_2 breaks up into nanosheets after 2 ps, where the atoms exposed on the surface of MoS_2 are colored grey. (c) Close up of the partially exfoliated MoS_2.

3.7 km/s. After 2 ps, the bubble collapses to create a nanojet, which hits the surface of bulk MoS_2 and peel the MoS_2 sheets apart, as shown in Figure 7(b). Figure 7(c) shows the partially exfoliated MoS_2, where the solution atoms are not shown for the clarity of presentation. We have found that the tensile wave created by the reflection of the shock wave from the backend of MoS_2 facilitates the exfoliation and breaks the MoS_2 into nanosheets. This simulation result supports our hypothesis of novel bubble-collapse-induced exfoliation.

CONCLUSION

To perform accurate and large reactive MD simulations, we have proposed a new ReaxPQ method and eliminated its computational bottleneck—speed-limiting charge iterations—by innovative adaptation of an extended-Lagrangian scheme. The resulting XRMD code has drastically improved the energy conservation compared with the previous RMD codes, while substantially reducing the time-to-solution. With various code transformations and hybrid MPI+OpenMP parallelization, the XRMD code has demonstrated high parallel efficiency, 0.977, on 786 432 Blue Gene/Q IBM processors as well as 2–3.0× speedups on the KNL-based Intel machine. Such scalable XRMD simulations could revolutionize the computational synthesis of advanced materials, as the example in this paper provides a glimpse.

ACKNOWLEDGMENTS

This work was supported as part of the Computational Materials Sciences Program funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, under Award DE-SC00014607. The authors would like to thank P. E. Small, P. Rajak, S. Tiwari, G. Zhou, and S. Naserifar for their help. This research used resources of the Argonne Leadership Computing Facility (ALCF), a U.S. Department of Energy Office of Science User Facility operated under contract DE-AC02-06CH-11357. The work of Y. Li was supported by the Margaret Butler Postdoctoral Fellowship at ALCF.

REFERENCES

1. A. Rahman, “Correlations in The motion of atoms in liquid argon,” *Phys. Rev.*, vol. 136, no. 2A, pp. A405–A411, Oct. 19, 1964.
2. R. Car and M. Parrinello, “Unified approach for molecular-dynamics and density-functional theory,” (in English), *Phys. Rev. Lett.*, vol. 55, no. 22, pp. 2471–2474, Nov. 25, 1985.
3. P. Hohenberg and W. Kohn, “Inhomogeneous electron gas,” *Phys. Rev.*, vol. 136, no. 3B, pp. B864–B871, Nov. 9, 1964.

4. D.R. Bowler and T. Miyazaki, “O(N) methods in electronic structure calculations,” (in English), *Rep. Progress Phys.*, vol. 75, no. 3, Mar. 2012, Art. no. 036503.
5. K. Shimamura, F. Shimojo, R. K. Kalia, A. Nakano, K. Nomura, and P. Vashishta, “Hydrogen-on-demand using metallic alloy nanoparticles in water,” *Nano Lett.*, vol. 14, no. 7, pp. 4090–4096, Jul. 9, 2014.
6. S. B. Sinnott and D. W. Brenner, “Three decades of many-body potentials in materials research,” (in English), *MRS Bull.*, vol. 37, no. 5, pp. 469–473, May 2012.
7. A. C. T. van Duin, S. Dasgupta, F. Lorant, and W. A. Goddard, “ReaxFF: A reactive force field for hydrocarbons,” *J. Phys. Chem. A*, vol. 105, no. 41, pp. 9396–9409, Oct. 18, 2001.
8. A. Nakano *et al.*, “De novo ultrascale atomistic simulations on high-end parallel supercomputers,” *Int. J. High Performance Comput. Appl.*, vol. 22, no. 1, pp. 113–128, Feb. 2008.
9. D. W. Brenner, “Empirical potential for hydrocarbons for use in simulating the chemical vapor-deposition of diamond films,” (in English), *Phys. Rev. B*, vol. 42, no. 15, pp. 9458–9471, Nov. 15, 1990.
10. Y. Li, R. K. Kalia, A. Nakano, K. I. Nomura, and P. Vashishta, “Multistage reaction pathways in detonating high explosives,” (in English), *Appl. Phys. Lett.*, vol. 105, no. 20, Nov. 17, 2014, Art. no. 204103.
11. A. K. Rappe and W. A. Goddard, “Charge equilibration for molecular-dynamics simulations,” (in English), *J. Phys. Chem.*, vol. 95, no. 8, pp. 3358–3363, Apr. 18, 1991.
12. F. H. Streitz and J. W. Mintmire, “Electrostatic potentials for metal-oxide surfaces and interfaces,” (in English), *Phys. Rev. B*, vol. 50, no. 16, pp. 11996–12003, Oct. 15, 1994.
13. T. J. Campbell, R. K. Kalia, A. Nakano, P. Vashishta, S. Ogata, and S. Rodgers, “Dynamics of oxidation of aluminum nanoclusters using variable charge molecular-dynamics simulations on parallel computers,” *Phys. Rev. Lett.*, vol. 82, no. 24, pp. 4866–4869, Jun. 14, 1999.
14. M. J. Cawkwell and A. M. N. Niklasson, “Energy conserving, linear scaling born-oppenheimer molecular dynamics,” (in English), *J. Chem. Phys.*, vol. 137, no. 13, Oct. 7, 2012, Art. no. 134105.
15. A. M. N. Niklasson and M. J. Cawkwell, “Generalized extended Lagrangian Born-Oppenheimer molecular dynamics,” (in English), *J. Chem. Phys.*, vol. 141, no. 16, Oct. 28 2014, Art. no. 164123.
16. P. Souvatzis and A. M. N. Niklasson, “First principles molecular dynamics without self-consistent field optimization,” (in English), *J. Chem. Phys.*, vol. 140, no. 4, Jan. 28, 2014, Art. no. 044117.
17. T. P. Senftle *et al.*, “The ReaxFF reactive force-field: Development, applications and future directions,” *NPJ Comput. Mater.*, vol. 2, Mar. 4, 2016, Art. no. 15011.
18. S. B. Kylasa, H. M. Aktulga, and A. Y. Grama, “PuReMD-GPU: A reactive molecular dynamics simulation package for GPUs,” (in English), *J. Comput. Phys.*, vol. 272, pp. 343–359, Sep. 1, 2014.
19. S. Naserifar, D. J. Brooks, W. A. Goddard, and V. Cvicek, “Polarizable charge equilibration model for predicting accurate electrostatic interactions in molecules and solids,” (in English), *J. Chem. Phys.*, vol. 146, no. 12, Mar. 28, 2017, Art. no. 124117.
20. H. M. Aktulga, S. A. Pandit, A. C. van Duin, and A. Y. Grama, “Reactive molecular dynamics: Numerical methods and algorithmic techniques,” *SIAM J. Sci. Comput.*, vol. 34, no. 1, pp. C1–C23, 2012.
21. K. Nomura, R. K. Kalia, A. Nakano, and P. Vashishta, “A scalable parallel algorithm for large-scale reactive force-field molecular dynamics simulations,” *Comput. Phys. Commun.*, vol. 178, no. 2, pp. 73–87, Jan. 15, 2008.
22. D. E. Shaw, “A fast, scalable method for the parallel evaluation of distance-limited pairwise particle interactions,” (in English), *J. Comput. Chem.*, vol. 26, no. 13, pp. 1318–1328, Oct. 2005.
23. S. J. Plimpton, “Fast parallel algorithms for short-range molecular dynamics,” *J. Comput. Phys.*, vol. 117, pp. 1–19, 1995.
24. J. C. Phillips, G. Zheng, S. Kumar, and L. V. Kale, “NAMD: Biomolecular simulation on thousands of processors,” presented at the Proc. Supercomputing, Baltimore, MD, USA, 2002.
25. B. G. Fitch *et al.*, “Blue matter: Strong scaling of molecular dynamics on BlueGene/L,” in *Proc. Int. Conf. Comput. Sci.*, vol. 3992, 2006, pp. 846–854.
26. R. A. Haring *et al.*, “The IBM Blue Gene/Q compute chip,” (in English), *IEEE Micro*, vol. 32, no. 2, pp. 48–60, Mar./Apr. 2012.
27. A. Sodani *et al.*, “Knights Landing: Second-generation Intel Xeon Phi product,” (in English), *IEEE Micro*, vol. 36, no. 2, pp. 34–46, Mar./Apr. 2016.
28. K. Nomura et al., “Nanocarbon synthesis by high-temperature oxidation of nanoparticles,” (in English), *Sci. Rep.*, vol. 6, Apr. 20, 2016, Art. no. 24109.
29. Y. Li, R. K. Kalia, A. Nakano, and P. Vashishta, “Size effect on the oxidation of aluminum nanoparticle: Multimillion-atom reactive molecular dynamics

simulations," (in English), *J. Appl. Phys.*, vol. 114, no. 13, Oct. 7, 2013, Art. no. 134312.

30. Y. Li, R. K. Kalia, A. Nakano, and P. Vashishta, "Oxidation dynamics of aluminum nanorods," (in English), *Appl. Phys. Lett.*, vol. 106, no. 8, Feb. 23, 2015, Art. no. 083101.

Ying Li is a postdoctoral scholar at the Argonne Leadership Computing Facility (ALCF). Her research interests include using molecular dynamics, quantum mechanics, and combining high-performance computing (HPC) and machine learning techniques to study materials systems. She received the M.S. degree in computer science and the Ph.D. degree in chemical engineering and materials science from the University of Southern California, Los Angeles, CA, USA. She is a member of the Materials Research Society. Contact her at yingli@anl.gov.

Ken-Ichi Nomura is a senior lecturer of chemical engineering and materials science at the University of Southern California (USC). His research interests include reactive atomistic modeling and high-performance parallel algorithms for materials simulation and visualization. He received the M.S. degree in computer science and the Ph.D. degree in physics from USC, Los Angeles, CA, USA. He is a member of the American Physical Society. Contact him at knomura@usc.edu.

Joseph A. Insley is the visualization and data analysis team lead at ALCF, where he researches parallel scalable methods for data analysis and visualization on current and next-generation systems, and a professor in the School of Art and Design, Northern Illinois University. He received the M.S. degree in computer science and an M.F.A. degree from the University of Illinois at Chicago, Chicago, IL, USA. He is a Senior Member of the IEEE Computer Society and the ACM. Contact him at insley@anl.gov.

Vitali Morozov is the lead for application performance and performance modeling at ALCF. At ALCF, he has been working on performance projections and studying the hardware trends in HPC, as well as porting and tuning applications. He received the B.S. degree in mathematics from Novosibirsk State University, Novosibirsk, Russia, and the Ph.D. degree in computer science from Ershov's Institute for Informatics Systems, Novosibirsk, Russia. Contact him at morozov@anl.gov.

Kalyan Kumaran is a senior computer scientist and the deputy director of science-advanced technologies and data at ALCF. His research interests include performance modeling and performance engineering. He is the long-time chair of the Standard Performance Evaluation Corporation High Performance Group and is a key developer of several widely used standard benchmarks. Contact him at kumaran@alcf.anl.gov.

Nichols A. Romero is a principal project specialist in computational science and team lead at ALCF. His primary research interest is massively parallel electronic structure calculations in condensed matter physics. He received the Ph.D. degree in physics from the University of Illinois at Urbana–Champaign, Champaign, IL, USA. He is a member of the APS. Contact him at naromero@anl.gov.

William A. Goddard III is the Charles and Mary Ferkel professor of chemistry, materials science, and applied physics at the California Institute of Technology. His research objective has been to describe the properties of chemical, biological, and materials systems directly from first principles (without the necessity of empirical data). To accomplish this, his research group has been developing new theory, new methods, and new software. Contact him at wag@wag.caltech.edu.

Rajiv K. Kalia is a professor of physics and astronomy, chemical engineering and materials science, and computer science at the University of Southern California. His research focuses on multiscale simulations involving atomistic, mesoscale, and continuum approaches on parallel supercomputers. He received the Ph.D. degree in physics from Northwestern University, Evanston, IL, USA. He is a Fellow of the APS. Contact him at rkalia@usc.edu.

Aiichiro Nakano is a professor of computer science, physics and astronomy, chemical engineering and materials science, and biological sciences at the University of Southern California. His research focuses on scalable parallel algorithms for materials science. He received the Ph.D. degree in physics from the University of Tokyo, Tokyo, Japan. He is a Fellow of the APS. Contact him at anakano@usc.edu.

Priya Vashishta is a professor of chemical engineering and materials science, physics and astronomy, and computer science at the University of Southern California. His research interests include HPC and visualization for large multiscale simulations of materials and processes. He received the Ph.D. degree in physics from the Indian Institute of Technology Kanpur, Kanpur, India. He is a Fellow of the APS. Contact him at priyav@usc.edu.