Institute for Advanced Simulation

Large Spatiotemporal-Scale Material Simulations on Petaflops Computers

Ken-ichi Nomura, Weiqiang Wang, Rajiv K. Kalia,
Aiichiro Nakano, Priya Vashishta, and Fuyuki Shimojo

published in

Multiscale Simulation Methods in Molecular Sciences,
J. Grotendorst, N. Attig, S. Blügel, D. Marx (Eds.),
Institute for Advanced Simulation, Forschungszentrum Jülich,
NIC Series, Vol. **42**, ISBN 978-3-9810843-8-2, pp. 321-336, 2009.

Large Spatiotemporal-Scale Material Simulations on Petaflops Computers

Ken-ichi Nomura[1], **Weiqiang Wang**[1], **Rajiv K. Kalia**[1], **Aiichiro Nakano**[1],
Priya Vashishta[1], **and Fuyuki Shimojo**[2]

[1] Collaboratory for Advanced Computing and Simulations
Department of Computer Science
Department of Physics and Astronomy
Department of Chemical Engineering and Material Science
University of Southern California, Los Angeles, CA 90089-0242, USA
E-mail: (knomura, wangweiq, rkalia, anakano, priyav)@usc.edu
[2] Department of Physics, Kumamoto University, Kumamoto 860-8555, Japan
E-mail: shimojo@kumamoto-u.ac.jp

We have developed a parallel computing framework for large spatiotemporal-scale atomistic simulations of materials, which is expected to scale on emerging multipetaflops architectures. The framework consists of: (1) an embedded divide-and-conquer (EDC) framework to design linear-scaling algorithms for high complexity problems; (2) a space-time-ensemble parallel (STEP) approach to predict long-time dynamics while introducing multiple parallelization axes; and (3) a tunable hierarchical cellular decomposition (HCD) parallelization framework to map these $O(N)$ algorithms onto a multicore cluster. The EDC-STEP-HCD framework has achieved: (1) inter-node parallel efficiency well over 0.95 for 218 billion-atom molecular-dynamics (MD) and 1.68 trillion electronic-degrees-of-freedom density functional theory-based quantum-mechanical simulations on 212,992 IBM BlueGene/L processors; (2) high intra-node, multithreading and single-instruction multiple-data parallel efficiency; and (3) nearly perfect time/ensemble parallel efficiency. The spatiotemporal scale covered by MD simulation on a sustained petaflops computer per day (i.e. petaflops•day of computing) is estimated as NT = 2.14 (e.g. N = 2.14 million atoms for T = 1 microseconds). Results of multimillion-atom reactive MD simulations on nano-mechano-chemistry reveal various atomistic mechanisms for enhanced reaction in nanoenergetic materials: (1) a concerted metal-oxygen flip mechanism at the metal/oxide interface in thermites; (2) a crossover of oxidation mechanisms in passivated aluminum nanoparticles from thermal diffusion to ballistic transport at elevated temperatures; and (3) nanojet-catalyzed reactions in a defected energetic crystal.

1 Introduction

Fundamental understanding of complex system-level dynamics of many-atom systems is hindered by the lack of validated simulation methods to describe large spatiotemporal-scale atomistic processes. The ever-increasing capability of high-end computing platforms is enabling unprecedented scales of first-principles based simulations to predict system-level behavior of complex systems.[1] An example is large-scale molecular-dynamics (MD) simulation involving multibillion atoms, in which interatomic forces are computed quantum mechanically to accurately describe chemical reactions.[2] Such simulations can couple chemical reactions at the atomistic scale and mechanical processes at the mesoscopic scale to solve broad mechano-chemistry problems such as nanoenergetic reactions, in which reactive nanojets catalyze chemical reactions that do not occur otherwise.[3] An even harder problem is to predict long-time dynamics, because the sequential bottleneck of time precludes efficient parallelization.[4,5]

The hardware environment is becoming challenging as well. Emerging sustained petaflops computers involve multicore processors,[6] while the computer industry is facing a historical shift, in which Moore's law due to ever increasing clock speeds has been subsumed by increasing numbers of cores in microchips.[7] The multicore revolution will mark the end of the free-ride era (i.e., legacy software will run faster on newer chips), resulting in a dichotomy—subsiding speedup of conventional software and exponential speedup of scalable parallel applications.

To address these challenges, we have developed key technologies for parallel computing with portable scalability. These include an embedded divide-and-conquer (EDC) algorithmic framework to design linear-scaling algorithms for broad scientific and engineering applications (e.g. equation solvers, constrained optimization, search, visualization, and graphs involving massive data) based on spatial locality principles.[8] This, combined with a space-time-ensemble parallel (STEP) approach[9] to predict long-time dynamics based on temporal locality[10] and a tunable hierarchical cellular decomposition (HCD) parallelization framework, maximally exposes concurrency and data locality, thereby achieving reusable "design once, scale on new architectures" (or metascalable) applications.[11, 12] It is expected that such metascalable algorithms will continue to scale on future multicore architectures. The "seven dwarfs" (a dwarf is an algorithmic method that captures a pattern of computation and communication) have been used widely to develop scalable parallel programming models and architectures.[6] We expect that the EDC-STEP-HCD framework will serve as a "metascalable dwarf" to represent broad large-scale scientific and engineering applications.[12]

We apply the EDC-STEP-HCD framework to a hierarchy of atomistic simulation methods. In MD simulation, the system is represented by a set of N point atoms whose trajectories are followed to study material properties.[4, 13, 14] Quantum mechanical (QM) simulation further treats electronic wave functions explicitly to describe chemical reactions.[15–17] To seamlessly couple MD and QM simulations, we have found it beneficial to introduce an intermediate layer, a first principles-based reactive force field (ReaxFF) approach,[18, 19] in which interatomic interaction adapts dynamically to the local environment to describe chemical reactions. The ReaxFF is trained by performing thousands of small QM calculations.

The metascalable simulation framework is enabling the study of a number of exciting problems, in particular, how atomistic processes determine material properties. Examples include the mechanical properties of nanocomposite materials and nanoindentation on them,[20] oxidation of nanoenergetic materials,[21] hypervelocity impact damage,[22] and fracture.[23, 24] We also study both colloidal[25] and epitaxial[26] quantum dots, and their interface with biological systems. It is noteworthy that experimentalists can now observe these phenomena at the same resolution as our simulations. For example, experimentalists perform nano-shock experiments using focused laser beams[27] and nano-fracture measurements using atomic force microscopy.[28] This lecture note focuses on one application related to nano-mechano-chemistry, i.e., enhanced reaction mechanisms in nanostructured energetic materials.

The lecture note is organized as follows. Section 2 describes our metascalable computing framework for large spatiotemporal-scale simulations of chemical reactions based on spatiotemporal data-locality principles. Results of nano-mechano-chemistry simulations are given in section 3, and section 4 contains conclusions.

2 A Metascalable Dwarf

2.1 Embedded Divide-and-Conquer (EDC) Algorithmic Framework

In the embedded divide-and-conquer (EDC) algorithms, the physical system is divided into spatially localized computational cells.[2] These cells are embedded in a global field that is computed efficiently with tree-based algorithms (Fig. 1).

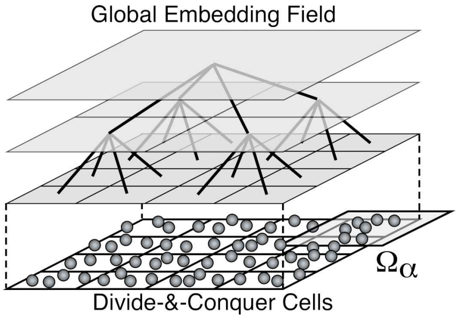

Figure 1. Schematic of embedded divide-and-conquer (EDC) algorithms. The physical space is subdivided into spatially localized cells, with local atoms constituting subproblems, which are embedded in a global field solved with tree-based algorithms.

Within the EDC framework, we have designed a number of $O(N)$ algorithms (N is the number of atoms). For example, we have designed a space-time multiresolution MD (MRMD) algorithm to reduce the $O(N^2)$ complexity of the N-body problem to $O(N)$.[13] MD simulation follows the trajectories of N point atoms by numerically integrating coupled ordinary differential equations. The hardest computation in MD simulation is the evaluation of the long-range electrostatic potential at N atomic positions. Since each evaluation involves contributions from $N-1$ sources, direct summation requires $O(N^2)$ operations. The MRMD algorithm uses the octree-based fast multipole method (FMM)[29,30] to reduce the computational complexity to $O(N)$ based on spatial locality. We also use multiresolution in time, where temporal locality is utilized by computing forces from further atoms with less frequency.[31]

We have also designed a fast ReaxFF (F-ReaxFF) algorithm to solve the $O(N^3)$ variable N-charge problem in chemically reactive MD in $O(N)$ time.[19] To describe chemical bond breakage and formation, the ReaxFF potential energy is a function of the positions of atomic pairs, triplets and quadruplets as well as the chemical bond orders of all constituent atomic pairs.[18] To describe charge transfer, ReaxFF uses a charge-equilibration scheme, in which atomic charges are determined at every MD step to minimize the electrostatic energy with the charge-neutrality constraint. This variable N-charge problem amounts to solving a dense linear system of equations, which requires $O(N^3)$ operations. The F-ReaxFF algorithm uses the FMM to perform the matrix-vector multiplications with $O(N)$ operations. It further utilizes the temporal locality of the solutions to reduce the amortized computational cost averaged over simulation steps to $O(N)$. To further speed up the solution, we use a multilevel preconditioned conjugate gradient (MPCG) method.[21,32] This method splits the

Coulomb interaction matrix into far-field and near-field matrices and uses the sparse near-field matrix as a preconditioner. The extensive use of the sparse preconditioner enhances the data locality, thereby increasing the parallel efficiency.

To approximately solve the exponentially complex quantum N-body problem in $O(N)$ time,[33,34] we use an EDC density functional theory (EDC-DFT) algorithm.[17,35] The DFT reduces the exponential complexity to $O(N^3)$, by solving N_{el} one-electron problems self-consistently instead of one N_{el}-electron problem (the number of electrons, N_{el}, is on the order of N). The DFT problem can be formulated as a minimization of an energy functional with respect to electronic wave functions. In the EDC-DFT algorithm, the physical space is a union of overlapping domains, $\Omega = \sum_\alpha \Omega_\alpha$ (Fig. 1), and physical properties are computed as linear combinations of domain properties that in turn are computed from local electronic wave functions. For example, the electronic density $\rho(\mathbf{r})$ is calculated as $\rho(\mathbf{r}) = \sum_\alpha p^\alpha(\mathbf{r}) \sum_n f(\epsilon_n^\alpha)|\psi_n^\alpha(\mathbf{r})|^2$, where the support function $p^\alpha(\mathbf{r})$ vanishes outside domain $\Omega_\alpha$and satisfies the sum rule, $\sum_\alpha p^\alpha(\mathbf{r}) = 1$, and $f(\epsilon_n^\alpha)$ is the Fermi distribution function corresponding to the energy ϵ_n^α of the n-th electronic wave function (or Kohn-Sham orbital) $\psi_n^\alpha(\mathbf{r})$ in Ω_α . For DFT calculation within each domain, we use a real-space approach based on high-order finite differencing,[36] where iterative solutions are accelerated using the multigrid preconditioning.[37] The multigrid is augmented with high-resolution grids that are adaptively generated near the atoms to accurately operate atomic pseudopotentials.[17] The numerical core of EDC-DFT thus represents a high-order stencil computation.[38,39]

2.2 Space-Time-Ensemble Parallelism (STEP) for Predicting Long-Time Dynamics

A challenging problem is to predict long-time dynamics because of the sequential bottleneck of time.[4,5] Due to temporal locality, however, the system stays near local minimum-energy configurations most of the time, except for rare transitions between them. In such cases, the transition state theory (TST) allows the reformulation of the sequential long-time dynamics as computationally more efficient parallel search for low activation-barrier transition events.[10,40] We also introduce a discrete abstraction based on graph data structures, so that combinatorial techniques can be used for the search.[40] We have developed a directionally heated nudged elastic band (DH-NEB) method,[9] in which a NEB consisting of a sequence of S states,[41] $\mathbf{R}_s \in \Re^{3N}(s = 0, ..., S-1, \Re$ is the set of real numbers, and N is the number of atoms), at different temperatures searches for transition events (Fig. 2(a)):

$$\mathbf{M}\frac{d^2}{dt^2}\mathbf{R}_s = \mathbf{F}_s - \mathbf{M}\gamma_s\frac{d}{dt}\mathbf{R}_s \tag{1}$$

where $\mathbf{M} \in \Re^{\mathbf{3N\times3N}}$ is the diagonal mass matrix and γ_s is a friction coefficient. Here, the forces are defined as

$$\mathbf{F}_s = \begin{cases} -\frac{\partial V}{\partial \mathbf{R}_s}|_\perp + \mathbf{F}_s^{spr}|_\parallel \\ -\frac{\partial V}{\partial \mathbf{R}_s} \end{cases} \tag{2}$$

where $V(\mathbf{R})$ is the interatomic potential energy, $\mathbf{F}_s^{spr}$ are spring forces that keep the states equidistance, and $\perp$ and $\parallel$ denote respectively the projections of a $3N$-element vector perpendicular and parallel to the tangential vector connecting the consecutive states.

We use an ensemble consisting of B bands to perform long-time simulation—molecular kinetics (MK) simulation–in the framework of kinetic Monte Carlo simulation.[9] Here, our space-time-ensemble parallel (STEP) approach combines a hierarchy of concurrency, i.e., the number of processors is

$$P = BSD : \tag{3}$$

(1) spatial decomposition within each state (D is the number of spatial subsystems, see section 2.3); (2) temporal parallelism across S states within each band; and (3) ensemble parallelism over B bands (Fig. 2(b)).

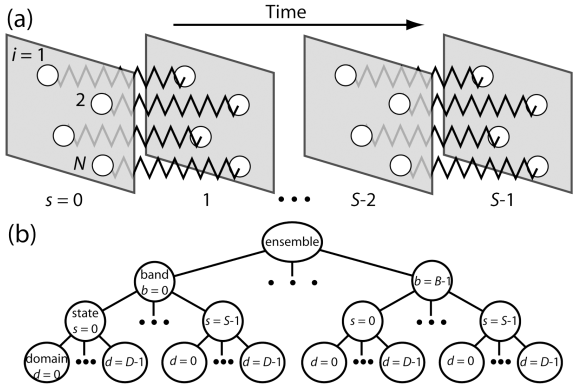

Figure 2. Schematic of the space-time-ensemble parallel (STEP) approach. (a) A nudged elastic band consists of a sequence of S states (gray parallelograms), $\mathbf{R}_s$ (s = 0,...,$S-1$), where each state consists of N atoms (white spheres), i = 1,...,N. Corresponding atoms in consecutive states interact via harmonic forces represented by wavy lines. (b) Tree-structured processor organization in the STEP approach. An ensemble consists of B bands, each consisting of S states; each state in turn contains D spatial domains.

2.3 Tunable Hierarchical Cellular Decomposition (HCD) for Algorithm-Hardware Mapping

To map the $O(N)$ EDC-STEP algorithms onto parallel computers, we have developed a tunable hierarchical cellular decomposition (HCD) framework.

Massively distributed scalability via message passing—Superscalability: Our parallelization in space is based on spatial decomposition, in which each spatial subsystem is assigned to a compute node in a parallel computer. For large granularity (the number of atoms per spatial subsystem, $N/D > 10^2$), simple spatial decomposition (i.e., each node 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[5,42] or other hybrid decomposition schemes[4,43–45] 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 processor varies significantly, and this load imbalance degrades the parallel efficiency. Load balancing can be stated as an optimization problem.[46–48] We minimize the load-imbalance cost as well the size and the number of messages. Our topology-preserving spatial decomposition allows message passing to be performed in a structured way in only 6 steps, so that the number of messages is minimized. To minimize the load imbalance cost and the size of messages, we have developed a computational-space decomposition scheme.[49] The main idea is that the computational space shrinks in a region with high workload density, so that the workload is uniformly distributed. The sum of load-imbalance and communication costs is minimized as a functional of the computational space using simulated annealing. We have found that wavelets allow compact representation of curved partition boundaries and thus speed up the optimization procedure.[50]

Multicore scalability via multithreading—Nanoscalablity: In addition to the massive inter-node scalability, ”there is plenty of room at the bottom,” as Richard Feynman noted. At the finest level, EDC algorithms consist of a large number of computational cells (Fig. 1), such as linked-list cells in MD[13] and domains in EDC-DFT,[17] which are readily amenable to parallelization. On a multicore compute node, a block of cells is assigned to each thread for intra-node parallelization. Our EDC algorithms are thus implemented as hybrid message passing + multithreading programs. Here, we use the POSIX thread standard, which is supported across broad architectures and operating systems. In addition, our framework[2] includes the optimization of data and computation layouts,[51, 52] in which the computational cells are traversed along various spacefilling curves[53] (e.g. Hilbert or Morton curve). To achieve high efficiency, special care must be taken also to make the multithreading free of critical sections. For example, we have designed a critical section-free algorithm to make all interatomic force computations in MRMD independent by reorganization of summation of atomic pair and triplet summations.[12] Our multithreading is based on a master/worker model, in which a master thread coordinates worker threads that actually perform force computations. We use POSIX semaphores to signal between the master and worker threads to avoid the overhead of thread creation and joining in each MD step. There are two check points at each MD time step, where all worker threads wait a signal from the master thread: (1) before the two-body force calculation loop, which also constructs the neighbor-lists, after atomic coordinates are updated; and (2) before three-body force calculation, after having all atoms complete neighbor-list construction. We have also combined multithreading with single-instruction multiple-data (SIMD) parallelism based on various code transformations.[39] Our SIMD transformations include translocated statement fusion, vector composition via shuffle, and vectorized data layout reordering (e.g. matrix transpose), which are combined with traditional optimization techniques such as loop unrolling.

Long-time scalability via space-time-ensemble parallelism (STEP)—Eon-scalability: With the spatial decomposition, the computational cost scales as N/D, while communication scales in proportion to $(N/D)^{2/3}$.[13] For long-range interatomic potentials used in MD simulations, tree-based algorithms such as the fast multipole method (FMM)[29, 30] incur an $O(\log D)$ overhead, which is negligible for coarse grained $(N/D \gg D)$ applications.[30] The communication cost of the temporal decomposition is $O(N/D)$ for copying nearest-neighbor images along the temporal axis, but the prefactor is negligible compared with the computation. Ensemble decomposition duplicates the band calculation, each involving SD processors, B times on $P = BSD$ processors. It involves $O((N/D)\log(BS))$ overhead to multicast the new initial state among the processors assigned the same spatial domain, i.e., those with the same p mod D.[9] Here, $p = bSD + sD + d$ is the sequential processor ID, where processor p is assigned the d-th spatial subsystem of the s-th state in the b-th band. The multicast cost at the beginning of each molecular-kinetics (MK) simulation step is greatly amortized over $10^3 - 10^4$ MD steps performed for the DH-NEB method per MK iteration.[9]

Intelligent tuning: The hierarchy of computational cells provides an efficient mechanism for performance optimization as well we make both the layout and size of the cells as tunable parameters that are optimized on each computing platform.[2] Our EDC-STEP algorithms are implemented as hybrid message-passing + multithreading programs in the tunable HCD framework, in which the numbers of message passing interface (MPI) processes and POSIX threads are also tunable parameters. The HCD framework thus maximally exposes data locality and concurrency. We are currently collaborating with compiler and artificial intelligence (AI) research groups to use: (1) knowledge-representation techniques for expressing the exposed concurrency; and (2) machine-learning techniques for optimally mapping the expressed concurrency to hardware.[54]

2.4 Scalability Tests

The scalability of our EDC-STEP-HCD applications has been tested on various high-end computing platforms including 212,992 IBM BlueGene/L processors at the Lawrence Livermore National Laboratory and 131,072 IBM BlueGene/P processors at the Argonne National Laboratory.

Inter-node (message-passing) spatial scalability: Figure 3 shows the execution and communication times of the MRMD, F-ReaxFF and EDC-DFT algorithms as a function of the number of processors P on the IBM BlueGene/L and P. Figure 3(a) shows the execution time of the MRMD algorithm for silica material as a function of P. We scale the problem size linearly with the number of processors, so that the number of atoms N = 2,044,416P. In the MRMD algorithm, the interatomic potential energy is split into the long- and short-range contributions, and the long-range contribution is computed every 10 MD time steps. The execution time increases only slightly as a function of P on both BlueGene/L and P, and this signifies an excellent parallel efficiency. We define the speed of an MD program as a product of the total number of atoms and time steps executed per second. The isogranular speedup is the ratio between the speed of P processors and that of one processor. The weak-scaling parallel efficiency is the speedup divided by P, and it is 0.975 on 131,072 BlueGene/P processors. The measured weak-scaling parallel efficiency on 212,992 BlueGene/L processors is 0.985 based on the speedup over 4,096 processors. Figure 3(a) also shows that the algorithm involves very small communication time. Figure 3(b) shows the execution time of the F-ReaxFF MD algorithm for RDX material as a function of P, where the number of atoms is N = 16,128P. The computation time includes 3 conjugate gradient (CG) iterations to solve the electronegativity equalization problem for determining atomic charges at each MD time step. On 212,992 BlueGene/L processors, the isogranular parallel efficiency of the F-ReaxFF algorithm is 0.996. Figure 3(c) shows the performance of the EDC-DFT based MD algorithm for 180P atom alumina systems. The execution time includes 3 self-consistent (SC) iterations to determine the electronic wave functions and the Kohn-Sham potential, with 3 CG iterations per SC cycle to refine each wave function iteratively. On 212,992 BlueGene/L processors, the isogranular parallel efficiency of the

EDC-DFT algorithm is 0.998 (based on the speedup over 4,096 processors). Our largest benchmark tests include 217,722,126,336-atom MRMD, 1,717,567,488-atom F-ReaxFF, and 19,169,280-atom (1,683,216,138,240 electronic degrees-of-freedom) EDC-DFT calculations on 212,992 BlueGene/L processors.

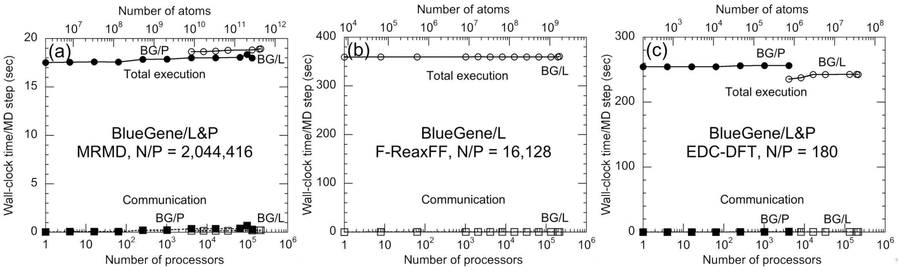

Figure 3. Total execution (circles) and communication (squares) times per MD time step as a function of the number of processors P of BlueGene/L (open symbols) and BlueGene/P (solid symbols) for three MD simulation algorithms: (a) MRMD for 2,044,416P atom silica systems; (b) F-ReaxFF MD for 16,128P atom RDX systems; and (c) EDC-DFT MD for 180P atom alumina systems.

Intra-node (multithreading) spatial scalability: We have tested the multithreading scalability of MRMD on a dual Intel Xeon quadcore platform. Figure 4 shows the speedup of the multithreaded code over the single-thread counterpart as a function of the number of worker threads. In addition to the speedup of the total program, Fig. 4 also shows the speedups of the code segments for two-body and three-body force calculations separately. We see that the code scales quite well up to 8 threads on the 8-core platform. We define the multithreading efficiency as the speedup divided by the number of threads. The efficiency of two-body force calculation is 0.927, while that for three-body force calculation is 0.436, for 8 threads. The low efficiency of the three-body force calculation may be due to the redundant computations introduced to eliminate critical sections. Nevertheless, the efficiency of the total program is rather high (0.811), since the fraction of the three-body calculation is about one third of the two-body force calculation. This result shows that the semaphore-based signaling between master and worker threads is highly effective. In a test calculation for a 12,228-atom silica system, the running time is 13.6 milliseconds per MD time step.

Time/ensemble scalability Scalability of the STEP-MRMD algorithm (note that the STEP approach can be combined with any of the MRMD, F-ReaxFF and EDC-DFT algorithms to compute interatomic forces) is tested on a cluster of dual-core, dual-processor AMD Opteron (at clock frequency 2 GHz) nodes with Myrinet interconnect. We define the speed of a program as a product of the total number of atoms and MK simulation steps executed per second. The speedup is the ratio between the speed of P processors and that of one processor. The parallel efficiency is the speedup divided by P. We first test the scalability of temporal decomposition, where we fix the number of bands B = 1 and the number of domains per state D = 1. We vary the number of states per band S = 4 to 1024. Here, the simulated system is amorphous SiO_2 consisting of N = 192 atoms, and we perform 600 MD steps per MK simulation step. The test uses all four cores per node. Figure 5(a) shows

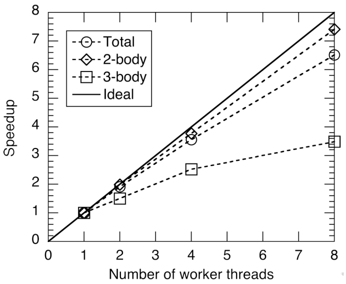

Figure 4. Speedup of the multithreaded MRMD algorithm over a single-threaded counterpart for the total program (circles), the two-body force calculation (diamonds), and three-body force calculation (squares). The solid line shows the ideal speedup.

the speedup of the STEP-MRMD program (we normalize the speedup on 4 processors as 4). The measured speedup on 1,024 processors is 980.2, and thus the parallel efficiency is 0.957. Next, we test the scalability of ensemble decomposition, where we fix the number of states per band S = 4 and the number of spatial domains per state D = 1. The number of bands per ensemble is varied from B = 1 to 256. The simulated system is amorphous $\mathrm{SiO_2}$ consisting of N = 192 atoms. Although multiple events are generated independently by different processor groups, the parallel algorithm involves sequential bottlenecks such as the selection of an event that occurs, and accordingly the parallel efficiency does degrade for a larger number of processors. Figure 5(b) shows the speedup of the STEP-MRMD program on the Opteron cluster as a function of the number of processors (normalized to be 4 on 4 processors). On 1,024 processors, the measured speedup is 989.2, and thus the parallel efficiency of ensemble decomposition is 0.966, which is slightly higher than that of temporal decomposition on the same number of processors.

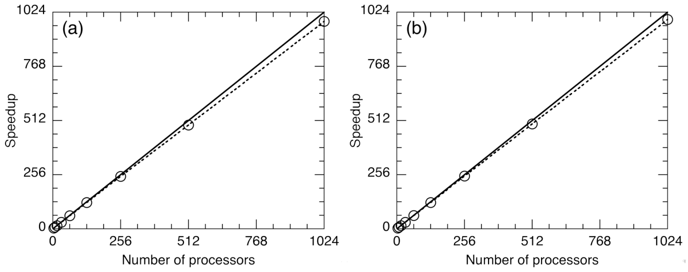

Figure 5. (a) Speedup of temporal decomposition in the STEP-MRMD algorithm (normalized so that the speedup is 4 for P = 4) as a function of the number of processors P (P = 4-1024) for a 192-atom amorphous SiO2 system on dual-core, dual-processor AMD Opteron nodes, where we fix B = D = 1. The circles are measured speedups, whereas the solid line denotes the perfect speedup. (b) Speedup of ensemble decomposition in the STEP-MRMD algorithm as a function of the number of processors P (= 4,...,1024) for silica material (N = 192 atoms). Here, we fix the number of states per band S = 4 and the number of spatial domains per state D = 1, while the number of bands is varied from B = 1 to 256.

3 Nano-Mechano-Chemistry Simulations

Recent advances in the integration of nanowires and nanoparticles of energetic materials into semiconducting electronic structures have opened up the possibility of ”nanoenergetics-on-a-chip (NOC)” technology, which has a wide range of potential applications such as micropropulsion in space and nano-airbags to drive nanofluidics.[55] Most widely used energetic materials for device integration are thermites, which are composites of metals and oxides. These materials have enormous energy release associated with the highly exothermic reduction/oxidation (redox) reactions to form more stable oxides. For example, arrays of Fe_2O_3 and CuO nanowires embedded in an Al matrix have been deposited on solid surfaces.[56] Another example of thermite nanostructures is self-assembly of an ordered array of Al and Fe_2O_3 nanoparticles.[57]

The integration of nanoenergetic materials into electronic circuits requires fundamental understanding and precise control of reaction rates and initiation time. The reactivity of nanoenergetic materials is known to differ drastically from their micron-scale counterparts. For example, experimental studies on the combustion of nanothermites, such as Al/Fe_2O_3, have shown that flame propagation speeds approach km/s when the size of Al nanoparticles is reduced to below 100 nm, in contrast to cm/s for traditional thermites.[58] Another example is the two-stage reaction of Al/CuO-nanowire thermite, in which the first reaction takes place at 500 °C followed by the second reaction at 660 °C (i.e., Al melting temperature).[56]

Such peculiar reactive behaviors of nanothermites cannot be explained by conventional mechanisms based on mass diffusion of reactants, and thus various alternative mechanisms have been proposed. An example is a mechano-chemical mechanism that explains the fast flame propagation based on dispersion of the molten metal core of each nanoparticle and spallation of the oxide shell covering the metal core.[59] Another mechanism is accelerated mass transport of both oxygen and metal atoms due to the large pressure gradient between the metal core and the oxide shell of each metal nanoparticle.[21, 60] In addition, defect-mediated giant diffusivity is important for fast reactions at the nanometer scale.[24, 61, 62]

The above mechanisms are by no means exhaustive, and some unexpected ones could operate in NOCs. It is therefore desirable to study the reaction of nanoenergetic materials by first-principles simulations. However, this poses an enormous theoretical challenge, where quantum-mechanical accuracy to describe chemical reactions must be combined with large spatial scales to capture nanostructural effects. Recent developments in scalable reactive MD simulations as described in the previous section have set the stage for such large first-principles MD simulations.

We have performed embedded divide-and-conquer (EDC) density functional theory (DFT) based MD simulations to study the thermite reaction at an Al/Fe_2O_3 interface (Fig. 6).[63] The results reveal a concerted metal-oxygen flip mechanism that significantly enhances the rate of redox reactions. This mechanism leads to two-stage reactions—rapid initial reaction due to collective metal-oxygen flips followed by slower reaction based on uncorrelated diffusive motions, which may explain recent experimental observation in thermite nanowire arrays.[56]

Here, we simulate a stack of Al and Fe_2O_3 layers involving 1,152 (144 Fe_2O_3 + 432 Al) atoms with periodic boundary conditions. The hematite (Fe_2O_3) crystal, cut along (0001) planes to expose Fe planes, is placed in the supercell with the (0001) direction parallel to the z direction (Fig. 6(a)). The Fe planes of the hematite are attached to (111)

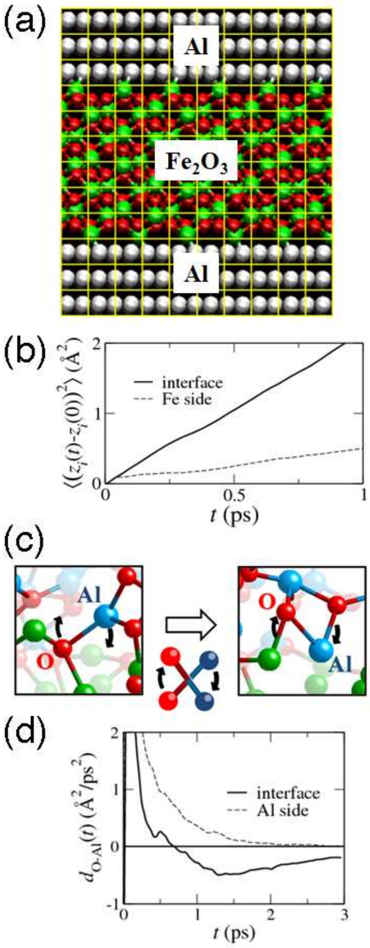

Figure 6. (a) Atomic configuration of Al/Fe$_2$O$_3$ interface. The green, red and grey spheres show the positions of Fe, O and Al atoms, respectively. Yellow meshes show the non-overlapping cores used by the EDC-DFT method. (b) Enhanced diffusion at the metal-oxide interface. Mean square displacements of O atoms along the z direction are plotted as a function of time. The solid and dashed curves are for O atoms in the interfacial and Fe-side regions, respectively. (c) Concerted metal-oxygen flip at the Al/Fe$_2$O$_3$ interface. (d) Negative correlation associated with concerted Al and O motions at the interface. Correlation functions between displacements of O and Al atoms along the z direction are shown as a function of time. The solid and dashed curves are obtained in the interfacial and Al-side regions.

planes of the face-centered cubic Al crystal at the two interfaces. Simulation results show enhanced mass diffusivity at the metal/oxide interface (Fig. 6(b)). To understand the mechanism of the enhanced diffusivity at the interface, we have examined the time evolution of the atomic configuration in the interfacial region and found a concerted metal-oxygen flip mechanism (Fig. 6(c)). That is, O atoms switch their positions with neighboring Al atoms while diffusing in the z direction. Careful bond-overlap population analysis shows that the switching motion between O and Al atoms at the interface is triggered by the change of chemical bonding associated with these atoms. To quantify the collective switching motion between O and Al atoms, we calculate the correlation function between the displacements of atoms along the z direction. The results in Fig. 6(d) (solid curve) reveal negative correlation for $t > 0.5$ ps, which reflects the collective switching motion between O and Al atoms at the interface as shown in Fig. 6(c). Such negative correlation does not exist on the Al side (the dashed curve in Fig. 6(d)), indicating independent diffusive motions of Al and O atoms.

Reactivity of nanoenergetic materials is often enhanced drastically from their micron-scale counterparts, which cannot be explained by conventional mechanisms based on mass diffusion of reactants. We have studied atomistic mechanisms of oxidation of an aluminum nanoparticle under extreme environment using multimillion atom reactive (ReaxFF) MD simulations, where the aluminum nanoparticle is coated with crystalline alumina shell and is ignited by heating the aluminum core to high temperatures, as is done in recent laser flash-heating experiments (Fig. 7).[27] The metallic aluminum and ceramic alumina are modeled by embedded atom model and many-body ionic-covalent potential form, respectively, which are interpolated with a bond-order based scheme validated quantum mechanically.

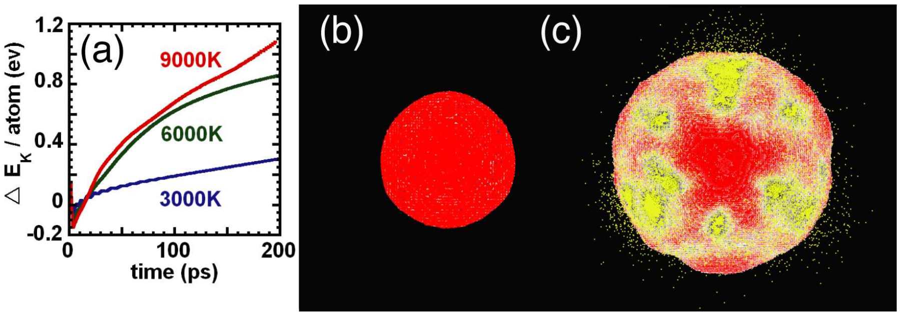

Figure 7. (a) Time variation of kinetic energy per aluminum atom during explosion with different initial temperatures T=3,000K (blue), 6,000K (green), and 9,000K (red), respectively. (b) Snapshot of nanoparticle at 100 ps (T=3,000K). (c) Snapshot of nanoparticle at 100 ps (T=9,000K). Core Al atoms (yellow) jet out through holes on the nanoparticle shell (red).

Simulation results reveal a transition of the reaction mechanism from thermodynamic to mechano-chemical regime, resulting in faster oxidation reaction of the aluminum nanoparticle, at elevated temperatures (Fig. 7(a)). The breakdown of the shell and the change of shell's morphology and composition during oxidation are found to play an important role for the transition. Specifically, we have identified three major changes of the shell, which are related to three mechanisms of atom migration: Diffusion (Fig. 7(b)), ballistic transport followed by diffusion, and ballistic transport followed by coalescing of atoms into few-atom clusters (Fig. 7(c)).

Mechanical stimuli in energetic materials initiate chemical reactions at shock fronts prior to detonation. Shock sensitivity measurements provide widely varying results, and quantum mechanical calculations are unable to handle systems large enough to describe shock structure. Recent developments in ReaxFF-MD combined with advances in parallel computing have paved the way to accurately simulate reaction pathways along with the structure of shock fronts. Our multimillion-atom ReaxFF-MD simulations of 1,3,5-trinitro-1,3,5-triazine (RDX) (Figs. 8(a) and (b)) reveal that detonation is preceded by a transition from a diffuse shock front with well ordered molecular dipoles behind it to a disordered dipole distribution behind a sharp front.[3]

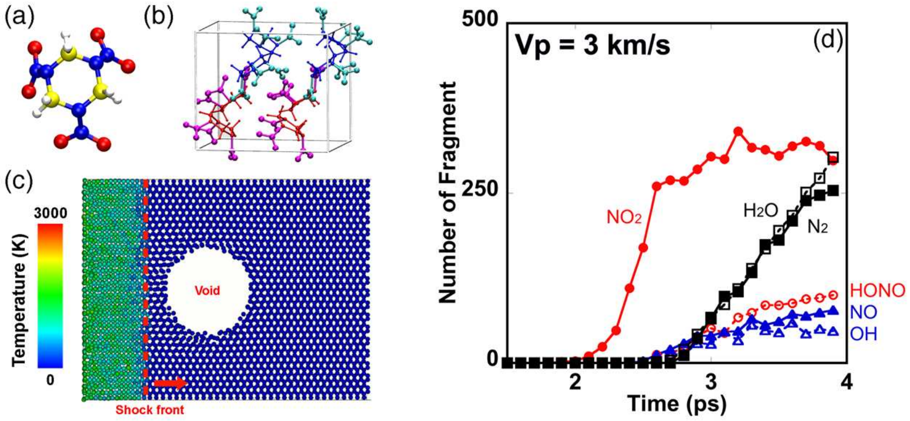

Figure 8. (a) An RDX molecule with carbon (yellow), hydrogen (white), oxygen (red), and nitrogen (blue) atoms. (b) The unit cell of an RDX crystal contains 8 RDX molecules, which are colored blue and red depending on whether the NO_2 groups faces away from (group1) or faces towards (group2) the shock plane. (c) Distribution of molecular vibrational temperature around the void at a particle velocity of 3 km/s. A red dotted-line represents the position of shock front. (d) Number of molecular fragments near the void surface. As the void collapses, two distinct reaction regimes are observed. From the arrival of the shock wave until the void closure ($\sim$ 2.6 ps), a rapid production of NO_2 is observed. Shortly after that, when molecules strike the downstream wall (2.6 – 3.9 ps), various chemical products such as N_2, H_2O and HONO are produced.

Nanofluidics of chemically reactive species has enormous technological potential and computational challenge arising from coupling quantum-mechanical accuracy with large-scale fluid phenomena. We have performed multimillion-atom ReaxFF-MD simulation of shock initiation of an RDX crystal with a nanometer-scale void (Fig. 8(c)).[64] The simulation reveals the formation of a nanojet that focuses into a narrow beam at the void. This, combined with the excitation of vibrational modes through enhanced intermolecular collisions by the free volume of the void, catalyzes chemical reactions that do not occur otherwise (Fig. 8(d)). We also observe a pinning-depinning transition of the shock wave front at the void at increased particle velocity and the resulting localization-delocalization transition of the vibrational energy. More recently, we have simulated nanoindentation of the (100) crystal surface of RDX by a diamond indenter.[65] Nanoindentation causes significant heating of the RDX substrate in the proximity of the indenter, resulting in the release of molecular fragments and subsequent ”walking” motion of these molecules on the indenter surfaces.

4 Conclusions

In summary, we have developed high-end reactive atomistic simulation programs to encompass large spatiotemporal scales with common algorithmic and computational frameworks based on spatiotemporal data-locality principles. In fact, our ”metascalable dwarf” extends far beyond atomistic simulations: Diverse applications, which encompass all of the original seven dwarfs, can be reduced by common techniques of embedding and divide-and-conquer to a highly scalable form. According to the scalability tests presented in this lecture note, they are likely to scale on future architectures beyond petaflops. The simulation algorithms are already enabling million-to-billion atom simulations of mechano-chemical processes, which have applications in broad areas such as energy and environment.

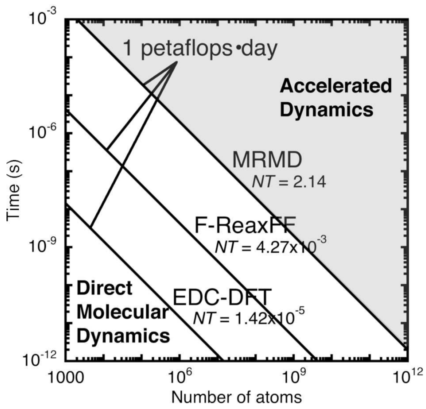

Figure 9. Spatiotemporal scales NT accessible by direct molecular-dynamics (white background) and approximate accelerated-dynamics (gray) simulations with a petaflops•day of computing. The lines are the NT achieved per petaflopsday of computing for MD (MRMD), chemically reactive MD (F-ReaxFF), and quantum-mechanical MD (EDC-DFT) simulations, respectively.

A critical issue, however, is the time scale studied by MD simulations. We define the spatiotemporal scale, NT, of an MD simulation as the product of the number of atoms N and the simulated time span T. On petaflops computers, direct MD simulations can be performed for NT = 1–10 atomseconds (i.e. multibillion-atom simulation for several nanoseconds or multimillion-atom simulation for several microseconds). More specifically, a day of computing on a sustained petaflops computer (i.e. one petaflops•day of computing) achieves NT = 2.14 (e.g. 1 million atoms for 2.14 microseconds) (Fig. 9), according to the benchmark test in section 2 (i.e., extrapolated from the measured MRMD performance on the BlueGene/L, which is rated as 0.478 petaflops according to the Linpack benchmark).[12] Accelerated-dynamics simulations[10] such as STEP molecular-kinetics simulations[9] will push the spatiotemporal envelope beyond NT = 10, but they need to be fully validated against direct MD simulations at NT = 1–10. Such large spatiotemporal-scale atomistic simulations are expected to advance scientific knowledge. This work was supported by NSF-ITR/PetaApps/EMT, DOE-SciDAC/BES, ARO-MURI, DTRA, and Chevron-CiSoft.

We thank Richard Clark, Liu Peng, Richard Seymour, and Lin H. Yang for fruitful collaborations.

References

1. S. Emmott and S. Rison, Towards 2020 Science (Microsoft Research, Cambridge, UK, 2006).
2. A. Nakano et al., Int'l J High Performance Comput Appl **22**, 113 (2008).
3. K. Nomura et al., Phys Rev Lett **99**, 148303 (2007).
4. J. C. Phillips et al., in Proc of Supercomputing (SC02) (ACM/IEEE, 2002).
5. D. E. Shaw et al., ACM SIGARCH Computer Architecture News **35**, 1 (2007).
6. K. Asanovic et al., The Landscape of Parallel Computing Research: A View from Berkeley (University of California, Berkeley, 2006).
7. J. Dongarra et al., CTWatch Quarterly **3**, 11 (2007).
8. A. Nakano et al., Comput Mater Sci **38**, 642 (2007).
9. A. Nakano, Comput Phys Commun **178**, 280 (2008).
10. A. F. Voter, F. Montalenti, and T. C. Germann, Annual Rev Mater Res **32**, 321 (2002).
11. F. Shimojo et al., J Phys: Condens Matter **20**, 294204 (2008).
12. K. Nomura et al., in Proc of International Parallel and Distributed Processing Symposium (IPDPS) (IEEE, 2009).
13. A. Nakano et al., in Proc of Supercomputing (SC01) (ACM/IEEE, 2001).
14. J. N. Glosli et al., in Proc of Supercomputing (SC07) (ACM/IEEE, 2007).
15. J. Nieplocha, R. J. Harrison, and R. J. Littlefield, in Proc of Supercomputing (SC94) (ACM/IEEE, 1994).
16. F. Gygi et al., in Proc of Supercomputing (SC05) (ACM/IEEE, 2005).
17. F. Shimojo et al., Phys Rev B **77**, 085103 (2008).
18. A. C. T. van Duin et al., J Phys Chem A **105**, 9396 (2001).
19. K. Nomura et al., Comput Phys Commun **178**, 73 (2008).
20. I. Szlufarska, A. Nakano, and P. Vashishta, Science **309**, 911 (2005).
21. T. J. Campbell, et al., Phys Rev Lett **82**, 4866 (1999).
22. P. S. Branicio et al., Phys Rev Lett **96**, 065502 (2006).
23. Z. Lu et al., Phys Rev Lett **95**, 135501 (2005).
24. Y. Chen et al., Phys Rev Lett **99**, 155506 (2007).
25. S. Kodiyalam et al., Phys Rev Lett **93**, 203401 (2004).
26. E. Lidorikis et al., Phys Rev Lett **87**, 086104 (2001).
27. Y. Q. Yang et al., Appl Phys Lett **85**, 1493 (2004).
28. F. Celarie et al., Phys Rev Lett **90**, 075504 (2003).
29. L. Greengard and V. Rokhlin, J Comput Phys **73**, 325 (1987).
30. S. Ogata et al., Comput Phys Commun **153**, 445 (2003).
31. A. Nakano, R. K. Kalia, and P. Vashishta, Comput Phys Commun **83**, 197 (1994).
32. A. Nakano, Comput Phys Commun **104**, 59 (1997).
33. S. Goedecker, Rev Mod Phys **71**, 1085 (1999).
34. D. R. Bowler et al., J Phys: Condens Matter **20**, 290301 (2008).
35. W. T. Yang, Phys Rev Lett **66**, 1438 (1991).
36. J. R. Chelikowsky et al., Physica Status Solidi B **217**, 173 (2000).
37. J.-L. Fattebert and J. Bernholc, Phys Rev B **62**, 1713 (2000).

38. K. Datta et al., in Proc of Supercomputing (SC08) (ACM/IEEE, 2008).
39. L. Peng et al., in Proc of International Parallel and Distributed Processing Symposium (IPDPS) (IEEE, 2009).
40. A. Nakano, Comput Phys Commun **176**, 292 (2007).
41. G. Henkelman and H. Jonsson, J Chem Phys **113**, 9978 (2000).
42. D. E. Shaw, J Comput Chem **26**, 1318 (2005).
43. S. J. Plimpton, J Comput Phys **117**, 1 (1995).
44. B. G. Fitch et al., Lecture Notes in Computer Science **3992**, 846 (2006).
45. M. Snir, Theor Comput Sys **37**, 295 (2004).
46. R. D. Williams, Concurrency: Practice and Experience **3**, 457 (1991).
47. K. D. Devine et al., Appl Numer Math **52**, 133 (2005).
48. U. V. Catalyurek et al., in Proc of International Parallel and Distributed Processing Symposium (IPDPS) (IEEE, 2007).
49. A. Nakano and T. J. Campbell, Parallel Comput **23**, 1461 (1997).
50. A. Nakano, Concurrency: Practice and Experience **11**, 343 (1999).
51. J. Mellor-Crummey, D. Whalley, and K. Kennedy, Int'l J Parallel Prog **29**, 217 (2001).
52. M. M. Strout and P. D. Hovland, in Proc of the Workshop on Memory System Performance (ACM, 2004).
53. B. Moon et al., IEEE Trans Knowledge Data Eng **13**, 124 (2001).
54. B. Bansal et al., in Proc of the Next Generation Software Workshop, International Parallel and Distributed Processing Symposium (IPDPS) (IEEE, 2007).
55. C. Rossi et al., J Microelectromech Sys **16**, 919 (2007).
56. K. Zhang et al., Appl Phys Lett **91**, 113117 (2007).
57. S. H. Kim and M. R. Zachariah, Adv Mater **16**, 1821 (2004).
58. K. B. Plantier, M. L. Pantoya, and A. E. Gash, Combustion and Flame **140**, 299 (2005).
59. V. I. Levitas et al., Appl Phys Lett **89**, 071909 (2006).
60. A. Rai et al., Combustion Theory and Modelling **10**, 843 (2006).
61. N. N. Thadhani, J Appl Phys **76**, 2129 (1994).
62. M. Legros et al., Science **319**, 1646 (2008).
63. F. Shimojo et al., Phys Rev E **77**, 066103 (2008).
64. K. Nomura et al., Appl Phys Lett **91**, 183109 (2007).
65. Y. Chen et al., Appl Phys Lett **93**, 171908 (2008).