LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-BOOK-608553

OpenAtom: Ab initio Molecular Dynamics for Petascale Platforms

G. J. Martyna, E. J. Bohm, R. Venkataraman, A. Arya, L. V. Kale, A. Bhatele

December 18, 2012

Parallel Science and Engineering Applications: The Charm++ Approach

Disclaimer

Chapter 5

OpenAtom: Ab-initio Molecular Dynamics for Petascale Platforms

Glenn J. Martyna

Physical Sciences Division, IBM Research

Eric J. Bohm, Ramprasad Venkataraman, Laxmikant V. Kale

Department of Computer Science, University of Illinois at Urbana-Champaign

Abhinav Bhatele

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory

5.1 Introduction

OpenAtom is parallel simulation software for studying atomic and molecular systems based on quantum chemical principles. In contrast to classical

computational molecular dynamics which is based on newtonian mechanics, OpenAtom uses the Car-Parrinello Ab Initio Molecular Dynamics (CPAIMD) approach. This allows it to study complex atomic and electronic physics in semiconductor, metallic, biological and other molecular systems. The application has been designed to expose maximal parallelism via small grains of data and computation. The resulting implementation atop Charm++ is highly scalable, and has exhibited portable performance across three generations of the IBM Blue Gene class of supercomputers, apart from other supercomputing platforms.

Instead of using an empiricial force function, the CPAIMD algorithm computes the forces acting on each atom as the summation of multiple terms derived from plane-wave density functional theory. Unlike traditional bulk-synchronous parallelization that simply decomposes the data, OpenAtom exploits the underlying mathematics via a seamless mix of both data and functional decompositions. This results in greater expressed parallelism, and several overlapping phases of computation combined with a longer critical path of dependent computations.

Such a design is enabled, and greatly facilitated, by the Charm++ tenet of *parallel program design and decomposition using units that are natural to the application domain*. Instead of dividing data into as many pieces as processors, OpenAtom simply decomposes the data *and* the computation across a number of chare objects. The type or number of these pieces are not limited by the number of processors. Rather, they depend on the CPAIMD algorithm and the desired grain size. For example, an electronic state is a unit of data that is natural to the CPAIMD algorithm and is one of the types of objects in the application.

We attempt, in this chapter, to further expand on such an approach to designing successful, scalable parallel programs (section 5.3). We preface the description of our parallel design with a discussion of the underlying physics (section 5.2). This includes a description of the computational algorithm, as well as the time and space complexities of each portion of the computation. The success of such a design approach is substantiated with performance results in section 5.5. Like several other successful Charm++ applications, OpenAtom has also inspired abstractions, libraries and other features that have made it back into the Charm++ parallel programming ecosystem. Section 5.4 briefly describes some of these features. We finally conclude with a few scientific studies that have used OpenAtom and the work planned for the future.

5.2 Car-Parrinello Molecular Dynamics

Car-Parrinello *ab initio* Molecular Dynamics (CPAIMD) [37, 209] is a key computational technique employed in the study of structure and dynam-

ics of atomistic systems of great interest throughout science and technology (S&T).The number of citations to the original research paper has grown exponentially and the method's use has spread from the physical sciences of chemistry, biology, geology and physics into the materials science and engineering disciplines. CPAIMD has indeed become an essential and ubiquitous tool for the investigation of the properties of matter of all types.

The power of the CPAIMD method lies in the novel combination of increasingly accurate electronic structure (ES) methods with increasingly efficient molecular dynamics (MD) techniques in such a way that they can be simulated/solved on the largest parallel High Performance Computing (HPC) platforms in existence today. Combining ES and MD allows the study of highly complex atomistic systems that involve changes in chemical bonding patterns or simply non-standard bonding under both equilibrium and non-equilibrium conditions. Unlike stand-alone *ab initio* methods where the atoms are typically fixed or may move along only an energy minimized pathway, CPAIMD allows the atoms to evolve naturally under the influence of Newton's equation of motion, molecular dynamics, with forces derived from ES theory. In this way, the effects of pressure, temperature and field gradients on systems with complex electronic structure, for example, can be discerned and the properties of liquids and amorphous materials that do not have a single identifiable representative structure can be illuminated. CPAIMD can be coupled to advanced sampling MD techniques to increase the time scales that can be accessed, and with path integral methods to determine nuclear quantum effects such as tunneling to increase the range of validity of the technique.

The CPAIMD method has been successfully applied in geophysics to describe the behavior of the cores of gas giant planets [38], in chemistry to understand the fundamental principles of aqueous acids and bases [169], in physics to study the properties of metal-insulator transitions [223], in engineering to study the behavior of devices and in materials science to study novel materials such as complex oxides [225]. This and other seminal work has had important impact across S&T leading to new scientific insight and engineering applications.

At present CPAIMD is limited to systems dominated by their ground state ES properties; it assumes the Born-Oppenheimer approximation wherein the nuclei evolve on a potential energy surface formed by the electronic ground state energy and nuclear-nuclear Coulombic interactions. The accuracy of ES methods intrinsic to the CPAIMD technique, which are necessarily approximate, are not currently sufficient to treat some critical systems such as diradicals, and systems dominated by dispersion interactions such as biological membranes with tractable computational efficiency [44]. The CPAIMD method is often applied using a plane wave basis set to describe the electronic states within the Gradient Corrected Local Density Approximation (GG-LDA or GGA) [21, 155, 201] to Density Functional Theory (DFT) [109, 141]. Research is underway to improve all aspects of the CPAIMD technique so as to increase accuracy, computational efficiency and applicability.

One of the important factors that has lead to the wide adoption of the CPAIMD method is the availability of highly (parallel) scalable, user-friendly HPC software. Some of the major plane wave based DFT packages include CPMD, Quantum Espresso, AbInit, QBOX and OpenAtom. CPMD, QBOX and OpenAtom have superior parallel scaling; AbInit, CPMD and Quantum Espresso have large user bases while OpenAtom is a Charm++ based experimental package designed and used primarily for CS based parallel HPC software and scientific physics-based methodological development in addition to materials research. CPMD, Quantum Espresso and OpenAtom have fairly open user licenses. All code bases have produced important application studies highly relevant to S&T.

5.2.1 Density Functional Theory, KS Density Functional Theory and the Local Density Approximation

Density Functional Theory states that the ground state energy of an electronic system can be expressed, exactly, as the minimum of a functional of the electron density [109, 180],

$$E[n(\mathbf{r})] = \int d\mathbf{r} n(\mathbf{r}) v^{ext}(\mathbf{r};\mathbf{R}) + F[n(\mathbf{r})] - \mu \left[\int d\mathbf{r} n(r) - n_e \right] \quad (5.1)$$

$$\frac{\delta E[n(r)]}{\delta n(\mathbf{r})} = = 0 \quad (5.2)$$

Here, the e-nuclear interaction potential is $v_{ext}(\mathbf{r}; R)$, and the unknown functional, $F[n(\mathbf{r})]$, can be expressed as the sum of physically intuitive terms:

$$F[n(\mathbf{r})] = E_H[n(\mathbf{r})] + T[n(\mathbf{r})] + E_{xc}[n(\mathbf{r})]$$

$$E_H[n(\mathbf{r})] = \int d\mathbf{r} \int d\mathbf{r}' \frac{n(\mathbf{r}) n(\mathbf{r}')}{\mid \mathbf{r} - \mathbf{r}' \mid}$$

The Hartree energy (E_H) is the interaction of classical charge distributions, the electronic quantum kinetic energy is T, and the "xchange-correlation" functional is E_{xc} which accounts for Fermi-statistics ("exchange") and other many body quantum effects ("correlation"). The Lagrangian multiplier μ insures the density represents the correct number of electrons n_e and is physically the chemical potential of the system. In general, the exchange-correlation functional may be separately divided into an exchange part (which is known exactly in certain limits) and a correlation part. Again, when minimized, $E[n(\mathbf{r})] = E_0$.

In order to allow the development of good approximate functionals, Kohn and Sham decomposed the electron density into a sum over a set of orthonormal electronic states [141],

$$n(\mathbf{r}) = \sum_s \Psi_s^2(\mathbf{r}) \quad (5.3)$$

to yield

$$\begin{aligned} E[n(\mathbf{r})] &= \int d\mathbf{r} n(\mathbf{r}) v^{ext}(\mathbf{r};\mathbf{R}) + F[n(\mathbf{r})] - \sum_{ss'} \lambda_{ss'} \left[\int d\mathbf{r} \Psi_s^*(\mathbf{r}) \Psi_{s'}(\mathbf{r}) - 2\delta_{ss'} \right] \\ F[n(\mathbf{r})] &= E_H[n(\mathbf{r})] + T_S[n(\mathbf{r})] + E_{xc,KS}[n(\mathbf{r})] \\ T_S[n(\mathbf{r})] &= -\frac{\hbar^2}{2m} \sum_s \int d\mathbf{r} \Psi_s(\mathbf{r}) \nabla^2 \Psi_s(\mathbf{r}) \end{aligned} \tag{5.4}$$

where each electronic state is doubly occupied, consistent with the Pauli-exclusion principle (1-spin down and 1-spin up electron occupy each state). We restrict ourselves to the spin-paired electron case here. A set of Lagrange multipliers $\lambda_{ss'}$ assures the normalization of the states. The exchange correlation functional is now relative to the non-interaction system and hence noted E_{xc}^{KS}; this distinction shall be dropped below. A widely used approximate functional is termed the Gradient Corrected Local Density Approximation (GG-LDA or GGA) to DFT and is written as [21, 155, 201]

$$E_{xc}[n(\mathbf{r})] = \int d\mathbf{r} \epsilon_{xc}(n(\mathbf{r}), \nabla n(\mathbf{r})) n(\mathbf{r}) \tag{5.5}$$

We restrict our discussion in this chapter to the GG-LDA approximation to KS-DFT and hereafter, simply refer to the technique as "DFT" to preserve simplicity.

In the discussion below, the nuclear-nuclear interaction,

$$\phi_{NN}(\mathbf{R}) = \frac{1}{2} \sum_{ij} \frac{Z_i Z_j}{\mid \mathbf{R}_i - \mathbf{R}_j \mid} \tag{5.6}$$

is assumed to be included in all the energy expressions. If the system is periodic, a sum over all periodic images is introduced and sum evaluated using Ewald method [50].

Lastly, for simplicity, we have written the electron-nuclear interaction as a local function $v^{ext}(\mathbf{r};\mathbf{R})$ only. In practice, it is a non-local term beyond the scope of the current discussion, but is discussed later when parallelization is described.

5.2.2 DFT Computations within Basis Sets

In the evaluation of DFT, it is useful to express the KS electronic states in terms of a set of known, closed form mathematical functions called a basis set

$$\Psi_s(\mathbf{r}) = \sum_k c_{sk} \chi(\mathbf{r}) \tag{5.7}$$

with c_{sk} as the expansion coefficients. In this section, we will concentrate on the application of the plane wave basis set.

$$\Psi_s(\mathbf{r}) = V^{-1/2} \sum_{\mathbf{g}} \tilde{\Psi}_s(\mathbf{g}) \exp\left(i\mathbf{g} \cdot \mathbf{r}\right) \tag{5.8}$$

with $\tilde{\Psi}_s(\mathbf{g})$ as the plane wave basis set coefficients.

The advantages of the plane wave basis set include: i) it is a complete, orthonormal set ensuring smooth convergence, ii) the plane wave basis functions do not depend on atom center position which obviates basis set superposition error of Gaussian approaches, and iii) in numerical simulations, fast Fourier transforms (FFTs) can be used to evaluate many of the terms, greatly increasing computational efficiency. Its main disadvantage is that it scales like N^3 and it is not easy to develop effective $O(N)$ scaling approaches with plane waves.

5.2.3 Molecular Dynamics

In the molecular dynamics method (MD), Newton's equations of Motion in Hamiltonian form for a set of N atoms (or nuclei) [75, 6]

$$\begin{aligned} H(\mathbf{P}, \mathbf{R}) &= \sum_i \frac{P_i}{2m_i} + \phi(\mathbf{R}) \\ \dot{\mathbf{R}}_i &= \frac{\mathbf{P}_i}{m_i} \\ \dot{\mathbf{P}}_i &= F_i = -\nabla_i \phi(\mathbf{R}) \end{aligned} \tag{5.9}$$

are solved numerically on a computer. As the equations of motion themselves can be solved in linear scale (with the number of atoms, N), the scaling of the MD method is determined by the scaling of the force evaluation. In the field, the term molecular dynamics is reserved for cases where the atomic/nuclear forces are derived from a closed form empirical potential function, $\phi(\mathbf{R})$ which is usually assumed to model well the Born-Oppenheimer electronic surface. MD potential functions are often complex but can usually be evaluated with $\mathcal{O}(N)$ computational complexity.

5.2.4 Ab-initio Molecular Dynamics and CPAIMD

One simple way to envision *ab initio* molecular dynamics within the DFT ES structure picture is to simply replace the empirical potential function with the minimized DFT functional. This approach is referred to as "Born-Oppenheimer" Molecular Dynamics (BOMD)) [45]. That is, one freezes the atoms, minimizes the desired density functional to an appropriate tolerance, evolves the atoms one time step forward with the nuclear forces determined from the(numerically/nearly) minimized functional $E[n^{min}(\mathbf{r}); \mathbf{R}]$

and repeats. The BOMD approach is widely employed but the minimization tolerance must be taken small or Hamilton's equations can become unstable. More sophisticated versions of this procedure that fold the minimization procedure into the numerical integration so as to preserve symmetry properties of Hamilton's equations [196] are beyond the scope of this chapter.

The approach we shall adopt here is the extended Lagrangian method pioneered by Car and Parrinello [37]. The coefficients of the basis set expansion coefficients of the KS orbitals are introduced as dynamical variables along with the nuclear degrees of freedom

$$\mathcal{L}_{CP} = \frac{\mu_{faux}}{2}\sum_{s\mathbf{g}}\dot{\tilde{\Psi}}_s^2(\mathbf{g}) + \frac{1}{2}\sum_i m_i\dot{\mathbf{R}}^2 - E[n(\mathbf{r});\mathbf{R}] - \phi_{NN}(\mathbf{R})$$

subject to the constraints

$$\sum_{\mathbf{g}}\tilde{\Psi}'_s(\mathbf{g})\tilde{\Psi}_s(\mathbf{g}) = 2\delta_{ss'} = O_{ss'} \tag{5.10}$$

A set of the Lagrange multipliers that preserve the (holonomic) orthonormality constraint of the KS states are introduced. Using the extended "Car-Parrinello" Lagrangian, equations of motion for the simultaneous evolution of the nuclei and the basis set coefficients can be derived.

$$\begin{aligned}\frac{d}{dt}\frac{\partial\mathcal{L}_{CP}}{\partial\dot{\tilde{\Psi}}_s} - \frac{\partial\mathcal{L}_{CP}}{\partial\tilde{\Psi}_s} &= 0 \\ \frac{d}{dt}\frac{\partial\mathcal{L}_{CP}}{\partial\dot{\mathbf{R}}_i} - \frac{\partial\mathcal{L}_{CP}}{\partial\mathbf{R}} &= 0\end{aligned} \tag{5.11}$$

If the basis set coefficients are assigned a "faux mass" parameter μ_{faux} that is sufficiently small, the initial faux kinetic energy in the basis set "coefficient velocities" is taken sufficiently small and the density functional is initially minimized, then an adiatatic separation can be invoked such that the basis set coefficients will evolve dynamically so as to keep the functional nearly minimized as the nuclei slowly evolve. Well understood MD techniques called Shake and Rattle [75] can be used to enforce the orthogonality constraints on the basis set coefficients.

5.2.5 Path Integrals

While MD and/or CPAIMD yield the motion of classical nuclei on the Born-Oppenheimer surface, this is sometimes insufficient to generate an adequate picture of the physics of a given system of interest. We shall consider two improvements in this chapter - path integral methods to add nuclear quantum effects on the ground BO electronic surface at the level of Boltzmann statistics and Parallel Tempering (Replica Exchange) to increase statistical sampling in systems with large energy barriers separating stable thermodynamic states (e.g. rough energy landscapes).

Feynman's path integral picture of quantum statistical mechanics [68] in the Boltzmann limit is particularly well suited for combination with CPAIMD. In Feynman's method, the single atom of classical mechanics is replaced by a classical ring polymer of length P beads held together with harmonic nearest-neighbor links. The classical limit arises when $P = 1$ and as P approaches infinity the results converge to the true quantum limit; the basic path integral method converges as P^{-2} and for most systems of interest $P \leq 64$ will suffice. Under Boltzmann statistics, each bead in the chain is assigned a number and only beads of different atoms with the same index interact with the 1 Pth of the potential $\phi(\mathbf{R}_i)/P$ with i the bead index. This picture is referred to as the classical isomorphism. Using advanced MD methods, it is possible to perform accurate PIMD simulations [244].

The CPAIMD method is easily grafted upon the PIMD technique to create CPAIPIMD. Simply put, the path integral method requires P electronic computations to generate $E[n_i(\mathbf{r}); \mathbf{R}_i]$ from which nuclear forces can be derived (e.g. the functional replaces the empirical potential of PIMD in a similar way as the same replacement takes MD to CPAIMD) and the CP Lagrangian can easily be extended to accomplish this change. The CPAIPIMD method can be parallelized effectively as the electron structure computations to do not interact directly; quantum effect arise indirectly from the harmonic forces confining the ring polymer of each atom into a small blob ("wave-packet"). The more "quantum" the particle, the wider the spread of the beads which for example allows the isomorphism to treat quantum tunneling. Parallelization is discussed in more detail in later sections.

5.2.6 Parallel Tempering

In systems with rough energy landscapes, barrier crossing events become sufficient rare that the results of a simulation study may not reflect the underlying physics. A system may simple become kinetically trapped in a local (free) energy minima and hence not "visit" the important regions of phase space. The same physical system at elevated temperature may, however, "traverse" phase space quite readily. In parallel tempering MD simulations (PTMD), M identical independent physical systems are run simultaneously at a set of temperatures T_i. Every a fixed number of MD steps, nearest neighbors in temperature space attempt to swap temperature with probability

$$P = \mathrm{Min}\left[1, \exp\left(\delta\beta_{ij}\delta H_{ij}\right)\right] \tag{5.12}$$

This can be shown to lead to M properly sampled systems at the M specified temperatures [60]. There are some formal difficulty using constant temperature MD methods to drive the dynamics of the M systems but these are considered minor and PT-MD is a well established method. It is most simple to use BOMD to implement PT within *ab initio* techniques and this is the course we are currently pursuing. For parallel computations, PT-BOMD is

quite attractive as the M BOMD simulations rarely communicate, and when they do, they need only exchange energies and temperatures.

5.3 Parallel Application Design

OpenAtom was envisioned from its inception as a fine grained implementation of Car-Parrinello *Ab-initio* MD using Charm++ as its parallelization substrate. Prior work, by our collaborators Glenn Martyna and Mark Tuckerman, in developing the PINYMD physics engine had already overcome the challenges of method implementation and validation. Hence, we elected to integrate the sequential simulation components from PINYMD into the design of OpenAtom. This resulted in a two level design with parallel control structures, Parallel Driver, implemented in Charm++ making calls to the integrated PINYMD routines implemented in C++ and Fortran.

5.3.1 Modular Design and Benefits

The overall CPAIMD algorithm is composed of several data manipulation and computation steps. One can envision the electronic states and the eventual atomic forces as the fundamental data that are computed and evolved through the simulation. Typical numerical algorithms express computations as a sequence of steps that operate on input data. Parallelization occurs by simply dividing large volumes of this data into smaller pieces. This naturally yields a procedural, bulk-synchronous expression of the algorithm suitable for coarse-grained weak scaling.

However, OpenAtom achieves its fine-grained parallelization by identifying the major steps of the algorithm and expressing each piece of the computation separately. Chare classes encapsulate the logic needed for each such piece of the computation along with the state needed for that piece across multiple iterations. The data of interest (electronic states, forces etc) then simply flow back and forth across these pieces as they evolve over the simulation. The different pieces of the algorithm are wired together by directing the output of a piece (class) as a message that triggers the computation in the next piece. This takes the form of remote method invocations. Traditional data decomposition is also trivially expressed by partitioning the input, output and computations in each piece of the algorithm (chare class) across many instances (objects) of that piece. There are several benefits that stem from such a design exercise.

- Parallelism arises from both a functional decomposition of the computation and the traditional decomposition of the data being operated upon.
- Computations are driven by the data sender. Data receivers do not have

to post receives, or take any preparatory actions to receive this data. In fact, objects that receive data (or messages) do not have to be aware of type, location or even the existence of a sender. This semantic promotes looser coupling of interacting pieces in the software. Class interfaces separate components that produce or consume each other's data. Such approaches are not new, but well-trodden paths in other domains. The challenge in HPC has been to convince the larger community that high performance can tolerate such loose coupling and other software engineering ideals. Frameworks like CHARM++ help scientific applications realize performance without sacrificing modular and maintainable software.

- Loose coupling also permits easy selection of different numerical methods / functionality by simply instantiating the objects of the appropriate chare classes. As long as they provide the same interfaces and data guarantees, the remaining application is unmodified.

- The messaging model in CHARM++ permits modifications to the communication structure of an individual parallel component without concerns for introducing subtle parallel bugs like deadlocks or races. CHARM++ does not preclude such issues completely, but only mitigates the need to understand the global communication state (all sends, receives etc) before introducing other parallel communication.

- Unit testing for numerical software is somewhat challenging. Correctness or failure may be determined far to the right of the decimal point! Usually, in numerical algorithms only the initial inputs and the final output can be easily accessed or validated. However, the loose coupling described above makes it easier to test individual software components. Mock environments and test harnesses are easily setup to interact with isolated components of a large, parallel application. OPENATOM has used this capability on several occasions to detect regressions and bugs in individual parallel components in a setting that is independent of the remaining application.

- Software components, in our experience, have differing rates of change. Some pieces are very stable and only need minor, occasional tweaks. Other parts experience constant modifications, enhancement or tuning. We have found this true of OPENATOM too. We found it helpful to introduce parallel interfaces at these layers of shear between differing rates of evolution. By isolating rapidly changing parallel components behind chare interfaces, we were able to insulate domain logic and other parallel modules from refactoring.

5.3.2 Parallel Driver

OpenAtom's parallel driver is composed of classes that match the logical expression of the terms of the Car-Parrinello method, supplemented by classes for optional numerical methods and features. Many of these classes that represent key steps of the computation or key representations of the data are promoted to chares classes with entry methods that clearly represent phases or stages of the computation. Instances of these classes are usually collected into chare arrays across one or more dimensions.

5.3.2.1 Decomposition

Multi dimensional chare arrays are the primary expression of decomposition. The plane-wave pseudopotential is expressed in terms of electronic states; each state is a 3D collection (usually a rectilinear box) of points. To facilitate a decomposition finer than the number of electronic states, a slice along one dimension is performed to form planes, that produces a natural decomposition along the dimensions of states and planes.

Let us preface our discussion with the following terms:

S : the number of electron states

N_c : the number of chunks of gspace plane-wave

N_d : the number of chunks of gspace density

P : the number of planes in the x-dimension of real space plane-wave

Sub : sub planes of decomposition for electron density

A : number of atoms

n_{atype} : number of atom types

The primary chare classes involved in expressing the CPAIMD algorithm in OpenAtom are listed below:

GSpace Driver (2D chare array $[N_c \times S]$) Handles flow of control within an instance, always same dimensional cardinality and mapping as gspace.

Electronic State Planewave GSpace (2D chare array $[N_c \times S]$) Handles the electronic structure in Fourier space (referred to as GSpace hereafter). Due to spherical cutoff sparsity, GSpace is rearranged into approximately equal size chunks. The number of chunks N_c is a free parameter selected at runtime.

Electronic State Planewave Real Space (2D chare array $[P \times S]$) Handles electronic structure in real space. The points of planewave pseudopotential are cut along the x-dimension for finer parallelization.

Electron Density Real Space (2D chare array $[P \times Sub]$) Handles electron

density in real space. Each plane may be further subdivided into subplanes at runtime for additional parallelism.

Electron Density GSpace (1D chare array $[N_d]$) Handles electron density in Fourier space. Due to spherical cutoff sparsity, GSpace is rearranged into approximately equal size chunks. The number of chunks N_d is a free parameter selected at runtime.

Electron Density Real Space Hart (3D chare array $[P \times Sub \times n_{atype}]$) Handles electron density hartree computation in real space.

Electron Density GSpace Hart (2D chare array $[N_d \times n_{atype}]$) Handles electron density hartree computation in Fourier Space.

Atoms (1D chare array $[A]$) Handles atomic positions, velocities, and corresponding data for computation of forces and positions.

AtomsCache (chare group) Provides globally available cache of positions and forces.

Non-local Particle GSpace (2D chare array $[N_c \times S]$) Handles non-local particle force computation Fourier space. Always same dimensional cardinality and mapping as GSpace.

Non-local Particle Real Space (2D chare array $[N_{nlees} \times S]$) Handles non-local particle force computation real space. N_{nlees} is determined by the x-dimension of the EES grid.

Orthonormalization (2D chare array $[S_{nog} \times S_{nog}]$) Handles orthonormalization based on iterative inverse sqrt method. S_{og} is set at runtime to be a factor of S_{pg}, $O_{nog} = S/O_{og}$.

Electron Pair Calculator (4D chare array $[P \times S_{npg} \times S_{npg} \times C]$) Computes the electron state pair matrix multiplication and correction for electronic structure plane wave forces and coefficients. S_{pg} is set at runtime to be a fraction of S. $S_{npg} = S/S_{pg}$.

Euler Exponential Spline Cache (chare group) Provides globally available cache for EES.

Structure Factor Cache (chare group) Provides globally available cache for structure factor.

5.3.2.2 Control Flow

Each major category of objects (see 5.1): GSpace, RealSpace, Density, etc. has a distinct flow of control. That flow is expressed in the RTH Thread suspend/resume syntax extension of the CHARM++ RTS. The flow is implicitly expressed by progress in program order through an event loop, wherein dependencies are explicitly expressed by application condition variable tests

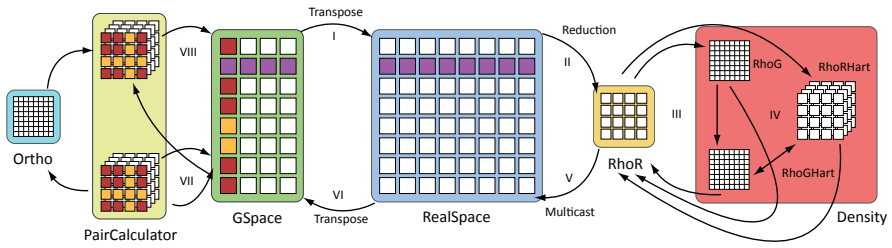

FIGURE 5.1: Control flow between different phases in OpenAtom

that guard local method calls on the chare, along with remote method invocations to communicate results to other objects. Object methods implement the computation for each phase of the computation and resume to the event loop, or the Charm++ RTS scheduler, as necessary. Entry point methods on the chare will set the relevant condition variables for the arriving data and resume into the event loop.

The key advantage of this scheme is that it provides a clear encapsulation of the state of each object, as the intersection of program order and tests of condition variables enforce algorithmic constraints for each object. This synergizes with Charm++'s support for adaptive overlap by allowing each object to safely progress through its state independently, as its constraints are met. It also allows for further application tuning, whereby computation can take place as early or late as the critical path of the application dictates. Furthermore, the communication of results from those computations can be throttled, or expedited, as appropriate.

For example, the FFT operation from *GSpace* to *RealSpace* results in a many to many personalized communication pattern, where each *GSpace* object issues a force message to each *RealSpace* object which shares its electronic state, that contains the portion of the result of the local FFT transform corresponding to the destination's plane index. *GSpace* objects then suspend until the Ψ x VKS data is returned from *RealSpace*. Meanwhile, *RealSpace*'s event loop collects FFT inputs until all have arrived, then completes the transform and initiates a reduction to sum across the states to produce one result for each plane of *Electron Density* objects.

Simultaneously with the previous paragraph's activities, the *Non-local* computations are overlapped with the *Electron Density*, with the latter taking priority. Due to the fact that number of data elements in the electron state grid is typically much larger (by at least an order of magnitude) than the electron density grid, the former has significantly better strong scaling characteristics than the latter. The automatic prioritized overlap allows computation units to do either, or both, at various strong scaling decompositions. This allows the application to efficiently scale up to a larger number of computational units before the Amdahl's Bottleneck from the *Electron Density* dominates performance.

As shown in figure 5.1 (electron state and density phases are shown) the control flow is data dependency directed to evolve the electron states, reduce them for the electron density, initiate the non-local force computation, and integrate nuclear forces based in each step. Decomposing the problem into distinct chares for each phase of the computation allows the implementation and placement decision for each element to be taken independently, or to build upon choices made for related phases, as necessary. See Figure 5.2 for how these decomposition options are applied in the context of network topology aware mapping.

5.3.2.3 Multiple Interacting Instances

The above comprise the primary components required to simulate one instance of a molecular system. In addition, several simulation capabilities require multiple interacting copies (instances) of a molecular system within the same simulation. These are handled by instantiating multiple copies of the above chare collections. Each instance hosts a set of the above interacting components wired to receive and send computation output to each other. All such components within an instance share a common identifier called the UberIndex.

The UberIndex is a higher level of organization imposed upon the instances of the chare arrays to implement features requiring the interaction of multiple variants of a system, such as Path Integrals, K-Points, Tempering, and Spin Orbitals. Each UberIndex contains an instance of each of the chare array classes and forms a complete description of a target molecular system. Coordination across UberIndices occurs in accordance with the synchronization required by each feature and users may use between zero and all Uber features, such as both Path Integrals and K-Points, in the same simulation.

Let these terms quantify UberInstance selection on decomposition :

I : number of instances

T : number of temperatures

B : number of path integral beads

Kp : number of k-points

S : number of spin directions (when enabled this is 2 for up and down).

Simulations based on multiple instances require the following additional chare classes:

Instance Controller (1D chare array $[I]$) Handles the instance initialization and coordination of cross instance communication. Handles coordination specific to cross B, cross Kp, cross S, and across all I.

Path Integral Bead Atom Integrator (1D chare array $[A]$) Handles integration of atom positions across path integral beads.

Temperature Controller (1D chare array [T]) Handles exchanging temperatures across Tempers.

5.3.3 Topology Aware Mapping

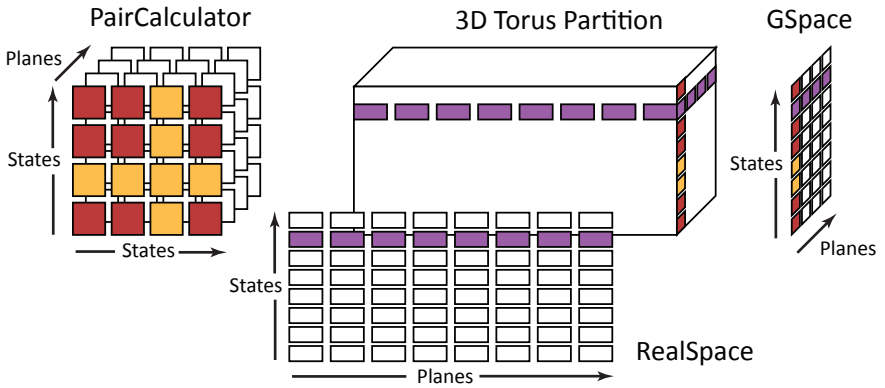

FIGURE 5.2: Mapping of GSpace, RealSpace and PairCalculator chare arrays to the 3D torus of a Blue Gene machine

The CHARM++ runtime maps various chare arrays in OPENATOM to the physical nodes and cores automatically. This default mapping is load balanced but possibly not optimized with respect to the specific communication patterns in OPENATOM. The runtime gives the freedom to the application developer to decide the placement of the chare arrays. Since OPENATOM is communication-bound, a load balanced mapping aimed at minimizing the inter-node communication was developed. Even with this optimized mapping, OPENATOM suffered from performance problems. Performance analysis on a large number of processors hinted at network contention problems due to heavy communication. To mitigate network contention, we started exploring interconnect topology aware mappings of the communication-bound phases in OPENATOM. Topology aware mapping aims at reducing the number of hops/links traversed by messages on the network to minimize link sharing and hence contention. This is achieved by placing objects that communicate frequently close together on the physical network.

Figure 5.1 presents the important phases and chare arrays in OPENATOM and the communication between them. The two-dimensional (2D) GSpace array communicates with the 2D RealSpace array plane-wise through transpose operations. The same GSpace array communicates with the 3D PairCalculator array state-wise through reductions. Optimizing one communication requires putting all planes of each state in GSpace together whereas the other communication benefits from placing all states of each plane in GSpace together. A hybrid approach that balances and attempts to favor both communications

has been developed. There are other communications between RealSpace and the density chares and ortho and the PairCalculator chares which also need to be considered.

Heuristics that optimize both the GSpace ⟷ PairCalculator and GSpace ⟷ RealSpace communication were considered and mappings of these chare arrays to three-dimensional torus networks were developed [24]. Figure 5.2 shows the mapping of these three chare arrays to a 3D torus partition. The GSpace array is mapped first to prisms (sub-tori) in the allocated 3D job partition. The RealSpace and PairCalculator chares are then mapped proximally to the GSpace array. This leads to significant performance improvements as shown in Figure 5.3.

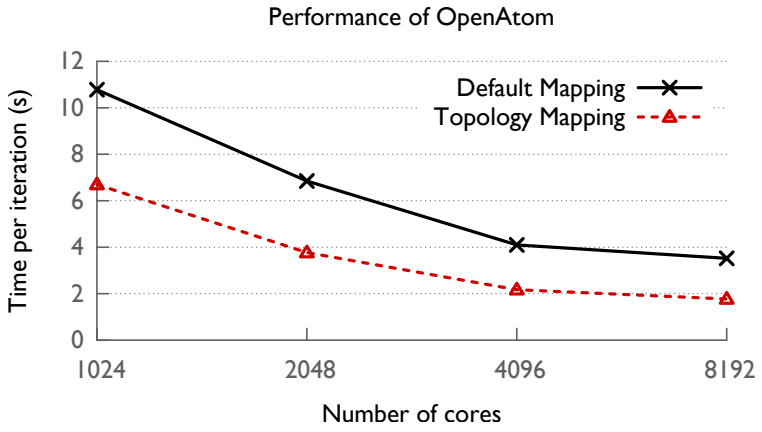

FIGURE 5.3: Performance improvement of nearly two times on 8192 cores of Blue Gene/P by using a topology aware mapping of OpenAtom (`WATER_256M_70Ry`)

5.4 Charm++ Feature Development

OpenAtom has driven the development of Charm++ in several ways. It is a case study in unified data and functional decomposition, and has also driven many capabilities required to support chare collections that span only a subset of the total number of processors in an execution. Some of the features in Charm++ that were inspired or partially driven by the requirements of OpenAtom are listed below:

Static Balance OpenAtom has no inherent dynamic load imbalance. It

achieves its performance benefit from CHARM++ due to fine-grained decomposition and the automatic overlap of prioritized computation phases. This demonstrates the unadulterated benefits of these techniques, in contrast to other CHARM++ applications which use dynamic load balancing. The aggressive use of prioritized messages in OPENATOM has been a driving use case for the development of a robust and efficient runtime implementation. Additionally, CHARM++ also sports runtime and build time options that can turn off dynamic load balancing and other instrumentation required by dynamically evolving applications.

TopoManager Library The planewise communication phases alternating with statewise communication phases, along with the independent expression of these operations in distinct chare arrays, demonstrates a high sensitivity to topology aware placement. This drove the development of the robust, flexible, cross platform TopoManager library that exposed the underlying network topology of the system across supercomputers with torus networks from different vendors.

CkMulticast Library The communication between Electronic State Planewave GSpace and the Electron Pair Calculator is confined to each plane. When executing at scale, each chare will typically have tens of data exchange partners, each of which must receive a part of its electronic state, and the entire state must be updated, returned, and reassembled. This has driven the development of the CkMulticast library to provide an efficient infrastructure to support the multicast and reduction operations with pipelining, customized control of spanning tree width, prioritization, and fragmentation. Specifically, OpenAtom was a heavy user of operations involving sections of a chare array (a plane of GSpace chares or a prism of Pair Calculator chares). This drove the optimization of multicasts and reduction to chare array sections.

Topology-aware multicasts and reductions OPENATOM's decomposition and design requires many common data movement patterns (multicasts, reductions, scatters, all-to-all). Several of these are expressed as operations involving a regular slice (section) of a chare array. For example data might need to be multicast from a plane of GSpace objects to a prism of Pair Calculator objects. However, the actual communication required does not translate cleanly to the underlying processors. This is because the number of objects is influenced by the problem and grain sizes, and their placement is influenced by topology and load balance considerations. OPENATOM, hence, performs many multicast and reduction operations that typically translate to a clustered but arbitrarily shaped subset of processors within the overall network topology. This is also, and especially, true for other CHARM++ applications that require object migrations to effect load balance.

In the strong scaling regime of the execution spectrum, OPENATOM is fairly sensitive to communication and messaging behavior. The fine grained parallelization results in a greater emphasis on optimized data movement.

Thus OpenAtom created a use case and drove the implementation of network-topology awareness in the Charm++ implementations of the multicast and reduction operations. This is implemented in Charm++ via the construction of topology aware spanning trees. A detailed description of this implementation is beyond the scope of this text.

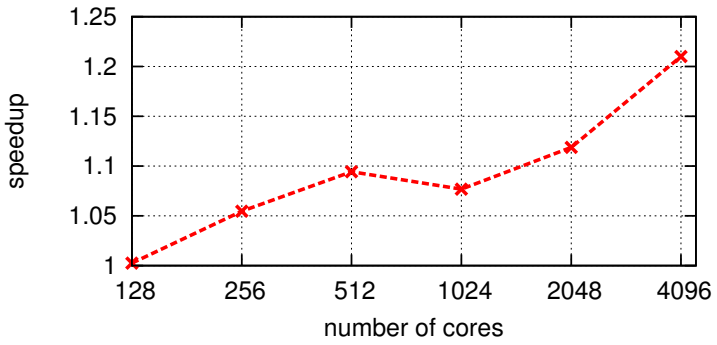

FIGURE 5.4: Speedup of OpenAtom due to topology-aware multicast and reduction operations in Charm++ relative to base cases without such topology aware collective communication. Data was obtained on Surveyor, an IBM Blue Gene/P, for the `WATER_32M_70Ry` dataset

Figure 5.4 illustrates the speedup obtained by OpenAtom when using a version of the Charm++ runtime system that could dynamically construct topology aware spanning trees over subsets of processors. The speedups are relative to the performance of OpenAtom at each of those processor counts without the use of topology aware spanning trees. We note that the application performance improves considerably. All Charm++ applications that perform multicasts and reductions over chare array sections now benefit from topology-aware multicasts and reductions whenever topology information is available.

Arrays Spanning a Subset of the Processor Allocation The most computationally heavy phases in the simulation scale to 20x the number of electronic states. However, several phases on the critical path, such as orthonormalization and electron density calculations, have portions which cannot profitably be decomposed as finely as the rest of the algorithm. The chare arrays for these phases will have fewer elements than the number of processors. OpenAtom has driven, and continues to drive, optimizations in the Charm++ runtime system for efficient support of such arrays. This requirement has also provided the initial basis for the later development of the UberIndex scheme, wherein all arrays span exclusive subsets of the processor allocation.

Many Multidimensional Chare Arrays The plethora of chare arrays and the total number of chare objects drove the development of robust and efficient support for the construction of many millions of objects on terascale machines.

5.5 Performance

OpenAtom scales well on the IBM Blue Gene architecture series, as shown in Figures 5.5 and 5.6. The high communication intensity of the algorithms benefits from the relatively balanced approach in the design of the Blue Gene series. Each figure shows the strong scaling performance of benchmark systems composed of molecules of liquid water with a 70 Rydberg cutoff at the Γ point, ranging from 8 to 256 molecules on the Blue Gene/L, and confined to 32 and 256 molecules on Blue Gene/P and Blue Gene/Q. All the plots uses log_2 on the X-axis and log_{10} on the Y-axis.

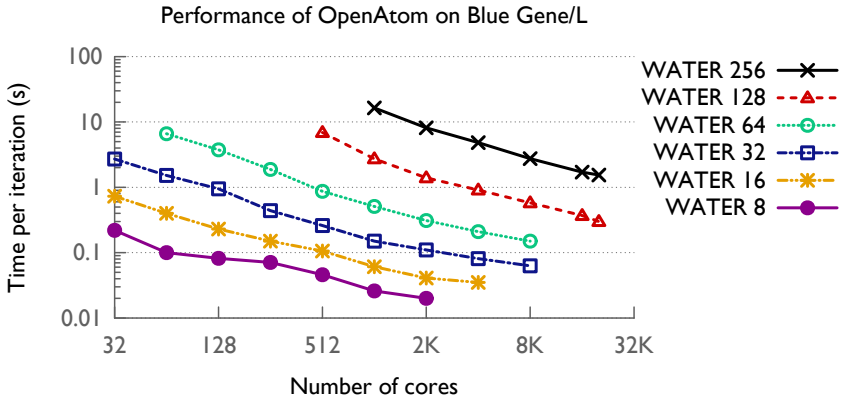

FIGURE 5.5: OpenAtom on Blue Gene/L (CO mode)

Figure 5.5 presents the time per iteration for six different water systems ranging from 8 to 256 molecules. All the runs were done in the co-processor (CO) mode which only uses one processor on each Blue Gene/L node for computation. To consider weak scaling performance, it must be noted that due to the dominance of $O(N^3)$ methods with increasing system size, doubling the number of molecules corresponds to an eight-fold increase in the amount of work. Therefore the corresponding order of magnitude relative time per step performance of these benchmarks represents good weak scaling. The largest 256 water molecules system scales well to 32,768 processors.

The left and right plots in Figure 5.6 show the scaling performance on Blue Gene/P and Blue Gene/Q respectivelty. The runs on Blue Gene/P were done in virtual node (VN) mode i.e. placing one process on each physical core. The runs on Blue Gene/Q were done in a similar on Blue Gene/Q refered to as the c16 mode. Once again, we see good scaling behavior for the larger benchmark system up to 32,768 cores of Blue Gene/Q.

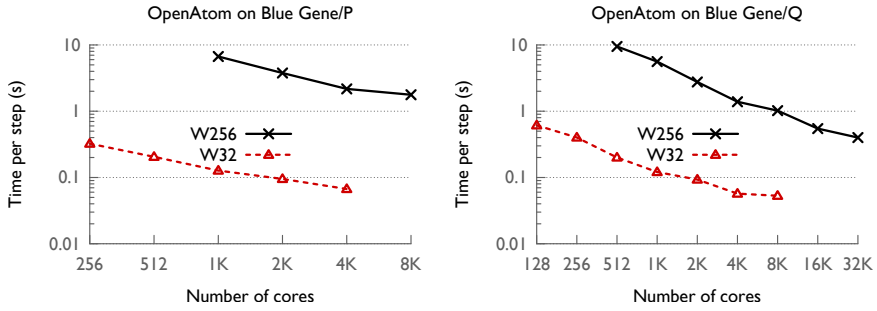

FIGURE 5.6: OpenAtom on Blue Gene/P (VN mode) and Blue Gene/Q (c16 mode)

5.6 Impact on Science & Technology

The OpenAtomm software suite [28] has been employed to gain insight into important systems spanning chemistry, biology, physics, materials science and engineering. Here, we discuss two application studies which illustrate the ability of the CPAIMD method to generate important insights into systems of high interest in S&T.

5.6.1 Carbon Based Materials for Photovoltaic Applications

One component of the green energy revolution involves the wide spread adoption of photovoltaic (PV) cells across an array of energy applications. However, breaking into the highly competitive energy market is quite difficult and a wholly economically driven adoption of solar cell technology requires breakthroughs that will lower cell cost, increase reliability and decrease the cost of installation. This of course requires, in turn, innovative exploratory research spanning all aspects of S&T.

There are several strategies that could be used to power the revolution [135, 83]. One involves the use of high quality crystalline silicon (c-Si) solar cells in conjunction with heat scavenging and solar concentrators; these high efficiency systems require harvesting every last bit of energy to offset the high (fixed) cost of c-Si. On the low end, organic PV cells can be cheaply ink-jet printed, need large areas due to low efficiency and are difficult to fabricate with the 25-30yr lifetimes required in some applications. An intermediate strategy is to build amorphous silicon (a-Si) solar cells which are stable to long times, and are sufficiently cheap to avoid concentrators and scavenging strategies, and would be adopted widely if costs could be dropped.

In Fig. 5.7, a mock-up of an a-Si solar cell is given. Unlike c-Si solar cells,

a top transparent conducting material or transparent conducting electrode (TCE) is required to conduct electrons away as doped a-Si mobilities are too low for the cell to function well. In current a-Si solar cell designs, the TCE is made of an Indium-Tin-Oxide (ITO) alloy. While these oxide materials perform very well, they are expensive both in terms of both materials(Indium) and processing (high temperature deposition of a metal oxide). It would be therefore an important advance if the ITO could be replaced by a cheaper, more easily processed material.

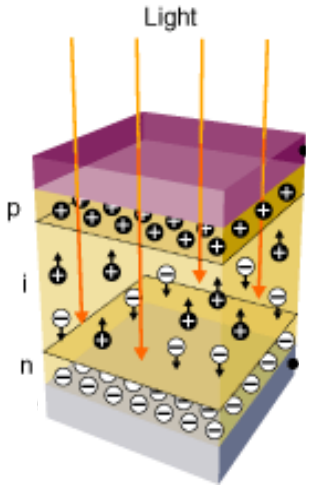

FIGURE 5.7: Mock up of a solar cell with a transparent top electrode, a PIN junction and a bottom electrode **(see Color Plate 5.)**

It has been suggested that graphene, a zero band gap insulator, which is a single carbon atom thick would, when doped, make a very effective TCE (see Fig. 5.8); that is, graphene is both thin and hence transparent and yet highly conductive when doped [3]. The recent development of a copper based process which yields high quality, large area graphene sheets [162] makes investigation of the properties of graphene in the context of solar cell TCE's very pertinent.

We have employed the CPAIMD technique as embodied in the OpenAtom software package to examine the physics and chemistry underlying the doping of graphene sheets. The basic physics we wish to observe is called rigid band doping. That is, the band structure (and hence density of states) does not alter in the low energy regime (near the Fermi level); the dopant merely

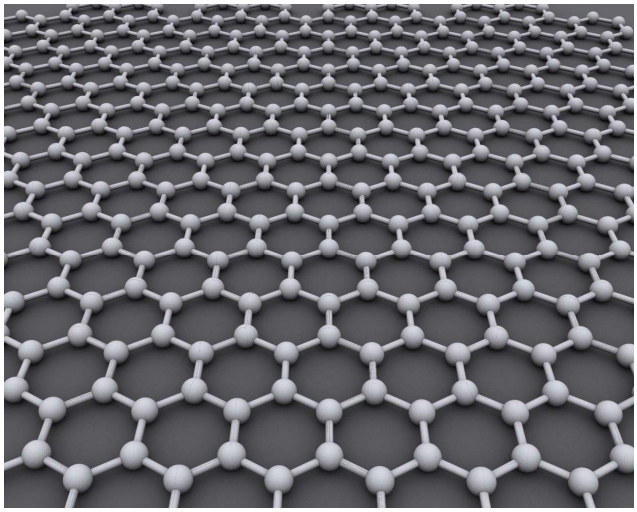

FIGURE 5.8: Graphene is a single continuous single atom thin isolated layer shaved from graphite **(see Color Plate 5.)**

serves to inject electrons/holes and hence shift the Fermi-level. Rigid band doping increases the number conduction pathways through the materials and hence the conductivity without diminution of charge carrier mobility through scattering; the intrinsic mobility of carriers in graphene is quite high.

In our studies of graphene doping, we considered a non-volatile dopant, $SbCl_5$, which would be expected to have long lifetime in a solar cell (Fig. 5.9). We observed that simply setting up a regular lattice of $SbCl_5$ molecules on the graphene surface did not cause a shift in the Fermi-level (e.g. no rigid band doping observed). That is, simply setting up a (guessed or presumed) structure and performing energy minimization did not properly predict the experimentally observed physics. We next allowed the molecules to evolve naturally according to Newton's equations using CPAIMD. We observed that the $SbCl_5$ molecules spontaneously dissociated following the disproportionation chemical reaction,

$$6SbCl_5 + C_84 \longrightarrow C_84^{2+}(2SbCl_6^-) + SbCl_3 + 3SbCl_5 \tag{5.13}$$

The open shell products p-dope induce rigid band doping of the sheet [195] (the system studied contains 6 molecular entities placed between two graphene layers containing 84 carbon atoms each).

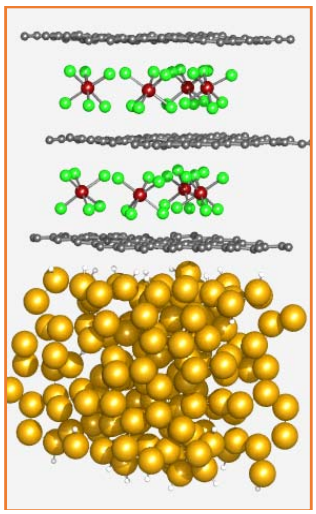

FIGURE 5.9: Graphene is a single continuous single atom think isolated layer shaved from graphite

The observed $SbCl_5$ disproportionation chemistry is catalyzed by graphene, itself, which functions as a metal-surface. There are antecedents in the graphite intercalation compound literature (e.g. graphite is a essentially an infinite number of A-B stacked graphene layers). However, in order to form intercalation compounds, extremely harsh conditions are applied to allow the compounds to enter/intercalate into the graphite lattice, making it unclear how the disproportionation reaction takes place. Our computations show clearly the mechanism involves a (metal) surface catalyzed charge transfer reaction that occurs spontaneously.

In order to further decrease the sheet resistance of the graphene, we have designed and tested a screen-printed busbar pattern experimentally. In order to determine the best metal for the design, we have used *ab initio* methods to study the physics and chemistry that leads to a high performance metal-graphene interface (e.g. emits a high tunneling current). Our current work involves studies of the a-Si-graphene interface which is key in developing an integrated solar cell solution.

5.6.2 Metal Insulator Transitions for Novel Devices

Although Moore's law continues in computer technology (chip features shrinking exponentially quickly with time), Dennard scaling, which allows clock frequency to increase concomitantly with feature size decrease, ceased abruptly in 2003 [242]. This halt is not due to a failure of engineering processes but occurs because CMOS, the current chip technology, has reached hard limits imposed by physics. Basically, preventing a charge carrier from crossing an electrostatically gated barrier requires in practice at least $\approx$ 1V line voltage at the operating temperatures and length scales of current technology. The computer industry therefore is in need of new approaches to digital switching that employ different physics.

Previous exploratory device research at IBM involved using electron-injection gated metal-insulator transitions (MIT) to provide the required switching (1=conducting, 0=insulating) physics [194]. The idea is to poise the channel material near the MIT, and provide just enough charge injection gating to push the system from the insulating to the conducting regime. This concept is, in fact, quite general and powerful, and not limited to charge injection mechanisms for the MIT.

We have also performed exploratory scientific research at IBM on pressure driven metal insulator transitions to study switching mechanisms of Phase Change Materials, specifically germanium doped Antimony, $\mathrm{Ge}_x Sb_{1-x}$ $x = 0.15$ [223]. In a typical application involving phase change materials, a heat pulse is applied to the conductive crystalline phase which creates an amorphous insulator and a more moderate annealing pulse is applied to the amorphous form to switch the material back to its crystalline state. This physics forms the basis of a non-volatile memory technology called Phase Change Memory. We have compared and contrasted pressure switching of the GeSb material [223]. Simulations of heating annealing and pressure annealing of the material yield similar end products. We concluded that the mechanism for the transition was a phenomena termed gap-driven amorphousization. As the crystalline material is heated or put under tensile stress, it begins to become favorable for the electronic band gap to open so as to lower free energy (as temperature or tensile stress has increased). As the gap opens, the bonding pattern changes (more 4-coordinate defects appear), the crystalline order decreases and the material evolves into an amorphous state. The amorphous state can be placed back into the crystal by applying compressive stress. The calculations were performed on 192 atom systems (29 Ge and 163 Sb) for very long times (100 picoseconds per quench).

We have recently explored combining the two approaches to create digital switches using pressure driven metal insulator transitions [191]. In order to make a switch as opposed to a memory, materials such as the intermediate valence compound SmSe which undergo a *continuous* MIT, a decrease of 4-orders in resistivity with the application of 1-2GPa of pressure, are used. The gating is accomplished through the application of a voltage across a piezoelectric

material. We have modeled our novel device which we term the Piezotronic Transistor(PET) theoretically and shown that it can be switched at very low line voltage 0.1V and yet maintain high speed (10GhZ). If the PET can be successfully fabricated at the nanoscale, it would represent an important new technology. We are currently pursuing the experimental embodiment of this device [192, 193].

5.7 Future Work

The introduction of replica style computations (Path Integrals,k-points, parallel tempering, and combinations thereof) has greatly increased the scale of machine, and the kind of simulation experiments, that the OpenAtom software can support. Path integrals allows computations of systems where nuclear quantum effects are important such as hydrogen exchange reactions for instances. Replica exchange permits increased sampling in systems with rough energy landscapes such as corregated surfaces and biomolecular configurations. Includling k-points allows increased accuracy even for large systems including metals which are important for the study of novel electronic devices. We are just beginning to explore these exciting new applications at present enable by OpenAtom.

The implementation of several highly desirable simulation features, are in the planning stages. Broadening the applicability of OpenAtom will be served by adding support for : GW-BSE, hybrid density functionals, fast super-soft pseudopotential techniques, localized basis sets, and CPAIMD-MM. GW-BSE will permit the study of excited properties of materials and bio-materials. The study of insulators is made more accurate by using hybrid DFT methods while super-soft pseudopotential techniques allow systems containing transistion and post-transition metals to be studied such as metalo-enzyme reaction centers and rare earth chalchogenide semiconductors. Localized basis sets will allow linear scale methods to be implemented increasing the system sizes that can be studied whilst CPAIMD-MM will permit a region treated with DFT methods to be embedded in a large bath of atoms treated more simply (empirical potential functions).

Lastly, the software infrastructure underlying OpenAtom will be upgraded. The expression of flow control will be improved by refactoring to use the higher level Charisma language, which has matured towards production over the course of this project. Refactoring the current planewise FFT decomposition into pencil form will improve strong scaling performance. A number of improvements in Charm++ infrastructure, such as the TRAM streaming module, will be leveraged to further improve performance. These changes are expected to improve both the performance and usability of the application

to expand its user community and extend the power of high performance computing to a wider variety of experimental challenges.

Acknowledgments

This work would not have been possible without the efforts of the many people who have contributed to the OpenAtom software over the years, including : Anshu Arya, Abhinav Bhatele, Eric Bohm, Chris Harrison, Marcelo Kuroda, Sameer Kumar, Glenn Martyna, Justin Meyer, Razvan Nistor, Esteban Pauli, Yan Shi, Edgar Solomonik, Mark Tuckerman, Ramkumar Vadali, Ramprasad Venkataraman, Da Wei, and Dawn Yarne.

This work began under the National Science Foundation grant ITR 0121357 and was continued under the Department of Energy grant DE-FG05-08OR23332. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-BOOK-608553).

This research used resources of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357.

Bibliography

[1] Chombo Software Package for AMR Applications. http://seesar.lbl.gov/anag/chombo.

[2] Leanmd code repository. git clone git://charm.cs.uiuc.edu/benchmarks/leanmd.

[3] A.and A. Bol A. Kasry, M. Kuroda, G.J. Martyna, and G.S. Tulevski. Chemical Doping of Large-Area Stacked Graphene Films for use as Transparent, Conducting Electrodes. *ACS Nano*, 4:3839, (2010).

[4] A. Adcroft, C. Hill, and J. Marshall. Representation of topography by shaved cells in a height coordinate ocean model. *Monthly Weather Review*, 125(9):2293–2315, 1997.

[5] L. Adhianto, S. Banerjee, M. Fagan, M. Krentel, G. Marin, J. Mellor-Crummey, and N. R. Tallent. Hpctoolkit: tools for performance analysis of optimized parallel programs http://hpctoolkit.org. *Concurr. Comput. : Pract. Exper.*, 22:685–701, April 2010.

[6] M.P. Allen and D.J. Tildesley. *Computer Simulations of Liquids*. Claredon Press, Oxford, (1989).

[7] R. J. Anderson. Tree data structures for n-body simulation. *SIAM J. Comput.*, 28:1923–1940, 1999.

[8] R. E. Angulo, V. Springel, S. D. M. White, A. Jenkins, C. M. Baugh, and C. S. Frenk. Scaling relations for galaxy clusters in the Millennium-XXL simulation. *ArXiv e-prints*, March 2012.

[9] Gabriel Antoniu, Luc Bouge, and Raymond Namyst. An efficient and transparent thread migration scheme in the PM^2 runtime system. In *Proc. 3rd Workshop on Runtime Systems for Parallel Programming (RTSPP) San Juan, Puerto Rico. Lecture Notes in Computer Science 1586*, pages 496–510. Springer-Verlag, April 1999.

[10] Amnon Barak, Shai Guday, and Richard G. Wheeler. The mosix distributed operating system. In *LNCS 672*. Springer, 1993.

[11] Kevin Barker, Andrey Chernikov, Nikos Chrisochoides, and Keshav Pingali. A Load Balancing Framework for Adaptive and Asynchronous Applications. In *IEEE Transactions on Parallel and Distributed Systems*, volume 15, pages 183–192, 2003.

[12] Kevin J. Barker and Nikos P. Chrisochoides. An Evaluation of a Framework for the Dynamic Load Balancing of Highly Adaptive and Irregular Parallel Applications. In *Proceedings of SC 2003*, Phoenix, AZ, 2003.

[13] J. Barnes and P. Hut. A Hierarchical O(NlogN) Force-Calculation Algorithm. *Nature*, 324:446–449, December 1986.

[14] C. Barrett, R. Beckman, K. Berkbigler, K. Bisset, B. Bush, K. Campbell, S. Eubank, K. Henson, J. Hurford, D. Kubicek, M. Marathe, P. Romero, J. Smith, L. Smith, P. Speckman, P. Stretz, G. Thayer, E. Eeckhout, and M. Williams. TRANSIMS: Transportation Analysis Simulation System. Technical Report LA-UR-00-1725, LANL, 2001.

[15] C. L. Barrett, K. Bisset, S. Eubank, M. V. Marathe, V.S. Anil Kumar, and Henning Mortveit. *Modeling and Simulation of Biological Networks*, chapter Modeling and Simulation of Large Biological, Information and Socio-Technical Systems: An Interaction Based Approach, pages 101–147. AMS, 2007.

[16] C. L. Barrett, S. Eubank, and M. V. Marathe. An interaction based approach to computational epidemics. In *AAAI' 08: Proceedings of the Annual Conference of AAAI*, Chicago USA, 2008. AAAI Press.

[17] C. L. Barrett, H. B. Hunt III, M. V. Marathe, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns. Complexity of Reachability Problems for Finite Discrete Dynamical Systems. *J. Comput. Syst. Sci.*, 72(8):1317–1345, 2006.

[18] Christopher L. Barrett, Richard J. Beckman, Maleq Khan, V.S. Anil Kumar, Madhav V. Marathe, Paula E. Stretz, Tridib Dutta, and Bryan Lewis. Generation and analysis of large synthetic social contact networks. In M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, editors, *Proceedings of the 2009 Winter Simulation Conference*, Piscataway, New Jersey, December 2009. Institute of Electrical and Electronics Engineers, Inc.

[19] A. Basermann, J. Clinckemaillie, T. Coupez, J. Fingberg, H. Digonnet, R. Ducloux, J.-M. Gratien, U. Hartmann, G. Lonsdale, B. Maerten, D. Roose, and C. Walshaw. Dynamic load balancing of finite element applications with the DRAMA Library. In *Applied Math. Modeling*, volume 25, pages 83–98, 2000.

[20] Jerome Baudry, Emad Tajkhorshid, Ferenc Molnar, James Phillips, and Klaus Schulten. Molecular dynamics study of bacteriorhodopsin and the purple membrane. *Journal of Physical Chemistry B*, 105:905–918, 2001.

[21] A.D. Becke. Density-Functional exchange-energy approximation with correct assymptotic behavior. *Phys. Rev. A*, 38:3098, (1988).

[22] R. J. Beckman, K. A. Baggerly, and M. D. McKay. Creating synthetic baseline populations. *Transportation Research Part A: Policy and Practice*, 30(6):415–429, 1996.

[23] Milind Bhandarkar, L. V. Kale, Eric de Sturler, and Jay Hoeflinger. Object-Based Adaptive Load Balancing for MPI Programs. In *Proceedings of the International Conference on Computational Science, San Francisco, CA, LNCS 2074*, pages 108–117, May 2001.

[24] Abhinav Bhatele, Eric Bohm, and Laxmikant V. Kale. Optimizing communication for charm++ applications by reducing network contention. *Concurrency and Computation: Practice and Experience*, 23(2):211–222, 2011.

[25] Abhinav Bhatelé, Laxmikant V. Kalé, and Sameer Kumar. Dynamic topology aware load balancing algorithms for molecular dynamics applications. In *23rd ACM International Conference on Supercomputing*, 2009.

[26] Scott Biersdorff, Chee Wai Lee, Allen D. Malony, and Laxmikant V. Kale. Integrated Performance Views in Charm ++: Projections Meets TAU. In *Proceedings of The 38th International Conference on Parallel Processing (ICPP)*, pages 140–147, Vienna, Austria, September 2009.

[27] Keith Bisset, Ashwin Aji, Madhav Marathe, and Wu-chun Feng. High-performance biocomputing for simulating the spread of contagion over large contact networks. *BMC Genomics*, 13(Suppl 2):S3, 2012.

[28] E. Bohm, A. Bhatele, L.V. Kale, M.E. Tuckerman, S. Kumar, J.A. Gunnels, and G.J. Martyna. Fine-grained parallelization of the Car-Parrinello ab initio molecular dynamics method on the Blue Gene/L supercomputer. *IBM J. Res. Dev.*, 52 1/2:159–176, (2008).

[29] Eric Bohm, Abhinav Bhatele, Laxmikant V. Kale, Mark E. Tuckerman, Sameer Kumar, John A. Gunnels, and Glenn J. Martyna. Fine Grained Parallelization of the Car-Parrinello ab initio MD Method on Blue Gene/L. *IBM Journal of Research and Development: Applications of Massively Parallel Systems*, 52(1/2):159–174, 2008.

[30] Kevin J. Bowers, Edmond Chow, Huafeng Xu, Ron O. Dror, Michael P. Eastwood, Brent A. Gregersen, John L. Klepeis, Istvan Kolossvary, Mark A. Moraes, Federico D. Sacerdoti, John K. Salmon, Yibing Shan,

and David E. Shaw. Molecular dynamics—scalable algorithms for molecular dynamics simulations on commodity clusters. In *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, page 84, New York, NY, USA, 2006. ACM Press.

[31] Kevin J. Bowers, Edmond Chow, Huafeng Xu, Ron O. Dror, Michael P. Eastwood, Brent A. Gregersen, John L. Klepeis, Istvan Kolossvary, Mark A. Moraes, Federico D. Sacerdoti, John K. Salmon, Yibing Shan, and David E. Shaw. Scalable algorithms for molecular dynamics simulations on commodity clusters. In *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, New York, NY, USA, 2006. ACM Press.

[32] Brams. http://www.cptec.inpe.br/brams/, Janeiro 2009.

[33] S. Browne, J. Dongarra, N. Garner, K. London, and P. Mucci. A scalable cross-platform infrastructure for application performance tuning using hardware counters. In *Proceedings of Supercomputing'00*, Dallas, Texas, 2000.

[34] Robert K. Brunner and Laxmikant V. Kalé. Adapting to load on workstation clusters. In *The Seventh Symposium on the Frontiers of Massively Parallel Computation*, pages 106–112. IEEE Computer Society Press, February 1999.

[35] Robert K. Brunner and Laxmikant V. Kalé. Handling application-induced load imbalance using parallel objects. In *Parallel and Distributed Computing for Symbolic and Irregular Applications*, pages 167–181. World Scientific Publishing, 2000.

[36] G. T. Camacho and M. Ortiz. Computational modeling of impact damage in brittle materials. *Int. J. Solids Struct.*, 33:2899–2938, 1996.

[37] R. Car and M. Parrinello. Unified approach for molecular dynamics and density functional theory. *Phys. Rev. Lett.*, 55:2471, (1985).

[38] C. Cavazzoni, G.L. Chiarotti, S. Scandolo, E. Tosatti, M. Bernasconi, and M. Parrinello. Superionic and Metallic States of Water and Ammonia at Giant Planet Conditions. *Science*, 283:44, (1999).

[39] Sayantan Chakravorty and L. V. Kale. A fault tolerant protocol for massively parallel machines. In *FTPDS Workshop for IPDPS 2004*. IEEE Press, 2004.

[40] Sayantan Chakravorty and L. V. Kale. A fault tolerance protocol with fast fault recovery. In *Proceedings of the 21st IEEE International Parallel and Distributed Processing Symposium*. IEEE Press, 2007.

[41] K. Channakeshava, K. Bisset, M. Marathe, A. Vullikanti, and S. Yardi. High performance scalable and expressive modeling environment to study mobile malware in large dynamic networks. In *Proceedings of 25th IEEE International Parallel & Distributed Processing Symposium*, 2011.

[42] Karthik Channakeshava, Deepti Chafekar, Keith Bisset, Anil Vullikanti, and Madhav Marathe. EpiNet: A simulation framework to study the spread of malware in wireless networks. In *SIMUTools09*. ICST Press, March 2009. Rome, Italy.

[43] K.L. Chung, Y.L. Huang, and Y.W. Liu. Efficient algorithms for coding Hilbert curve of arbitrary-sized image and application to window query. *Information Sciences*, 177(10):2130–2151, 2007.

[44] A.J. Cohen, Paula Mori-Sanchez, and Weitao Yang. Insights into current limitations of density functional theory. *Science*, 321:792, (2008).

[45] M. C.Payne, M.P. Teter, D.C. Allan, T.A. Arias, and J.D. Joannopoulos. Iterative minimization techniques for ab initio total-energy calculations: molecular dynamics and conjugate gradients. *Rev. Mod. Phys.*, 64:1045, (1992).

[46] T.A. Darden, D.M. York, and L.G. Pedersen. Particle mesh Ewald. An N·log(N) method for Ewald sums in large systems. *JCP*, 98:10089–10092, 1993.

[47] M. Davis, G. Efstathiou, C. S. Frenk, and S. D. M. White. The evolution of large-scale structure in a universe dominated by cold dark matter. *Astrophys. J.*, 292:371–394, May 1985.

[48] M. Davis, G. Efstathiou, C. S. Frenk, and S. D. M. White. The evolution of large-scale structure in a universe dominated by cold dark matter. *Astrophys. J.*, 292:371–394, May 1985.

[49] W. Dehnen. Towards optimal softening in three-dimensional N-body codes - I. Minimizing the force error. *MNRAS*, 324:273–291, June 2001.

[50] S.W. deLeeuw, J.W. Perram, and E.R. Smith. Simulation of Electrostatic Systems in Periodic Boundary Conditions. I. Lattice Sums and Dielectric Constants. *Proc. R. Soc. London A*, 373:27, 1980.

[51] Department of Computer Science,University of Illinois at Urbana-Champaign, Urbana, IL. *The CHARM (5.9) programming language manual*, 2006.

[52] Department of Computer Science,University of Illinois at Urbana-Champaign, Urbana, IL. *The CONVERSE programming language manual*, 2006.

[53] Jayant DeSouza and Laxmikant V. Kalé. MSA: Multiphase specifically shared arrays. In *Proceedings of the 17th International Workshop on Languages and Compilers for Parallel Computing*, West Lafayette, Indiana, USA, September 2004.

[54] K. Devine, B. Hendrickson, E. Boman, M. St. John, and C. Vaughan. Design of Dynamic Load-Balancing Tools for Parallel Applications. In *Proc. Intl. Conf. Supercomputing*, May 2000.

[55] Karen D. Devine, Erik G. Boman, Robert T. Heaphy, Bruce A. Hendrickson, James D. Teresco, Jamal Faik, Joseph E. Flaherty, and Luis G. Gervasio. New challenges in dynamic load balancing. *Appl. Numer. Math.*, 52(2–3):133–152, 2005.

[56] J. Diemand, M. Kuhlen, P. Madau, M. Zemp, B. Moore, D. Potter, and J. Stadel. Clumps and streams in the local dark matter distribution. *Nature*, 454:735–738, August 2008.

[57] H.-Q. Ding, N. Karasawa, and W. A. Goddard, III. The reduced cell multipole method for Coulomb interactions in periodic systems with million-atom unit cells. *Chemical Physics Letters*, 196:6–10, August 1992.

[58] P. Domingos and M. Richardson. Mining the Network Value of Customers. In *Proc. ACM KDD*, pages 57–61, 2001.

[59] Isaac Dooley. *Intelligent Runtime Tuning of Parallel Applications With Control Points*. PhD thesis, Dept. of Computer Science, University of Illinois, 2010. http://charm.cs.uiuc.edu/papers/DooleyPhDThesis10.shtml.

[60] D. J. Earl and M.W. Deem. Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.*, 7:3910–3916, (2005).

[61] D. Easley and J. Kleinberg. *Networks, Crowds and Markets: Reasoning About A Highly Connected World*. Cambridge University Press, New York, NY, 2010.

[62] G. Efstathiou, M. Davis, S. D. M. White, and C. S. Frenk. Numerical techniques for large cosmological N-body simulations. *Astrophys. J. Supp.*, 57:241–260, February 1985.

[63] S.N. Eliane, E. Araújo, W. Cirne, G. Wagner, N. Oliveira, E.P. Souza, C.O. Galvão, and E.S. Martins. The SegHidro Experience: Using the Grid to Empower a HydroMeteorological. In *Proceedings of the First International Converence on e-Science and Grid Computing (e-Science/05)*, pages 64–71, 2005.

[64] S. Eubank, H. Guclu, V. S. Anil Kumar, M. Marathe, A. Srinivasan, Z. Toroczkai, and N. Wang. Modelling disease outbreaks in realistic urban social networks. *Nature*, 429:180–184, 2004.

[65] A. E. Evrard. Beyond N-body - 3D cosmological gas dynamics. *MNRAS*, 235:911–934, December 1988.

[66] P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annalen der Physik*, 369:253–287, 1921.

[67] A. L. Fazenda, J. Panetta, P. Navaux, L. F. Rodrigues, D. M. Katsurayama, and L. F Motta. Escalabilidade de aplicação operacional em ambiente massivamente paralelo. In *Anais do X Simpósio em Sistemas Computacionais (WSCAD-SCC)*, pages 27–34, 2009.

[68] R.P. Feynman. *Statistical Mechanics.* Benjamin, Reading, (1972).

[69] B. Fitch, R. Germain, M. Mendell, J. Pitera, M. Pitman, A. Rayshubskiy, Y. Sham, F. Suits, W. Swope, T. Ward, Y. Zhestkov, and R. Zhou. Blue Matter, an application framework for molecular simulation on Blue Gene. *Journal of Parallel and Distributed Computing*, 63:759–773, 2003.

[70] Blake G. Fitch, Aleksandr Rayshubskiy, Maria Eleftheriou, T. J. Christopher Ward, Mark Giampapa, Michael C. Pitman, and Robert S. Germain. Molecular dynamics—blue matter: approaching the limits of concurrency for classical molecular dynamics. In *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, page 87, New York, NY, USA, 2006. ACM Press.

[71] IT Foster and BR Toonen. Load-balancing algorithms for climate models. In *Proceedings of Scalable High-Performance Computing Conference*, pages 674–681, 1994.

[72] Peter L. Freddolino, Anton S. Arkhipov, Steven B. Larson, Alexander McPherson, and Klaus Schulten. Molecular dynamics simulations of the complete satellite tobacco mosaic virus. *Structure*, 14:437–449, 2006.

[73] SR Freitas, KM Longo, MAF Silva Dias, R. Chatfield, P. Silva Dias, P. Artaxo, MO Andreae, G. Grell, LF Rodrigues, A. Fazenda, et al. The Coupled Aerosol and Tracer Transport model to the Brazilian developments on the Regional Atmospheric Modeling System (CATT-BRAMS). *Atmospheric Chemistry and Physics*, 9(8):2843–2861, 2009.

[74] C. S. Frenk, S. D. M. White, P. Bode, J. R. Bond, G. L. Bryan, R. Cen, H. M. P. Couchman, A. E. Evrard, N. Gnedin, A. Jenkins, A. M. Khokhlov, A. Klypin, J. F. Navarro, M. L. Norman, J. P. Ostriker, J. M. Owen, F. R. Pearce, U.-L. Pen, M. Steinmetz, P. A. Thomas, J. V. Villumsen, J. W. Wadsley, M. S. Warren, G. Xu, and G. Yepes. The Santa Barbara Cluster Comparison Project: A Comparison of Cosmological Hydrodynamics Solutions. *Astrophys. J.*, 525:554–582, November 1999.

[75] D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Academic Press, 1996.

[76] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1998.

[77] George Karypis and Vipin Kumar. Multilevel k-way Partitioning Scheme for Irregular Graphs. *Journal of Parallel and Distributed Computing*, 48:96–129 , 1998.

[78] T. C. Germann, K. Kadau, I. M. Longini, Jr., and C. A. Macken. Mitigation strategies for pandemic influenza in the United States. *Proc. of National Academy of Sciences*, 103(15):5935–5940, April11 2006.

[79] P. H. Geubelle and J. Baylor. Impact-induced delamination of composites: a 2d simulation. *Composites B*, 29(B):589–602, 1998.

[80] R. Gevaerd, S. R. Freitas, and K. M. Longo. Numerical simulation of biomass burning emission and trasportation during 1998 roraima fires. In *Proceedings of International Conference on Southern Hemisphere Meteorology and Oceanography (ICSHMO) 8*, 2006.

[81] S. Ghan, X. Bian, A. Hunt, and A. Coleman. The thermodynamic influence of subgrid orography in a global climate model. *Climate Dynamics*, 20(1):31–44, 2002.

[82] S. Ghan and T. Shippert. Load balancing and scalability of a subgrid orography scheme in a global climate model. *International Journal of High Performance Computing Applications*, 19(3):237, 2005.

[83] D.S. Ginley and D. Cahen. *Fundamentals of Materials for Energy and Environmental Sustainability*. Cambridge University Press, Cambridge, UK.

[84] Filippo Gioachin and Laxmikant V. Kalé. Dynamic High-Level Scripting in Parallel Applications. In *In Proceedings of the 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Rome, Italy, May 2009.

[85] Filippo Gioachin, Amit Sharma, Sayantan Chakravorty, Celso Mendes, Laxmikant V. Kale, and Thomas R. Quinn. Scalable cosmology simulations on parallel machines. In *VECPAR 2006, LNCS 4395, pp. 476-489*, 2007.

[86] Filippo Gioachin, Gengbin Zheng, and Laxmikant V. Kalé. Debugging Large Scale Applications in a Virtualized Environment. In *Proceedings of the 23rd International Workshop on Languages and Compilers for Parallel Computing (LCPC2010)*, number 10-11, Huston, TX (USA), October 2010.

[87] Filippo Gioachin, Gengbin Zheng, and Laxmikant V. Kalé. Robust Record-Replay with Processor Extraction. In *PADTAD '10: Proceedings of the 8th Workshop on Parallel and Distributed Systems: Testing, Analysis, and Debugging*, pages 9–19. ACM, July 2010.

[88] E Goldstein, A Apolloni, B Lewis, J C Miller, M Macauley, S Eubank, M Lipsitch, and J Wallinga. Distribution of vaccine/antivirals and the 'least spread line'; in a stratified population. *J R Soc Interface*, 7(46):755–64, 2010.

[89] R. Gould. Collective action and network structure. *American Sociological Review*, 58:182–196, 1993.

[90] F. Governato, C. Brook, L. Mayer, A. Brooks, G. Rhee, J. Wadsley, P. Jonsson, B. Willman, G. Stinson, T. Quinn, and P. Madau. Bulgeless dwarf galaxies and dark matter cores from supernova-driven outflows. *Nature*, 463:203–206, January 2010.

[91] F. Governato, B. Willman, L. Mayer, A. Brooks, G. Stinson, O. Valenzuela, J. Wadsley, and T. Quinn. Forming disc galaxies in ΛCDM simulations. *MNRAS*, 374:1479–1494, February 2007.

[92] S. L . Graham, P. B. Kessler, and M. K. McKusick. GPROF: a call graph execution profiler. *SIGPLAN 1982 Symposium on Compiler Construction*, pages 120–126, June 1982.

[93] M. Granovetter. Threshold Models of Collective Behavior. *American J. Sociology*, 83(6):1420–1443, 1978.

[94] L. Greengard. *The rapid evaluation of potential fields in particle systems.* PhD thesis, MIT, Cambridge, MA, USA., 1988.

[95] G. Grell and D. Devenyi. A generalized approach to parameterizing convection combining ensemble and data assimilation techniques. *Geophysical Research Letters*, 29(14):38–1, 2002.

[96] G. Grimmett. *Percolation.* Springer, 1989.

[97] A. Gursoy, L.V. Kale, and S.P. Vanka. Unsteady fluid flow calculations using a machine independent parallel programming environment. In R. B. Pelz, A. Ecer, and J. Hauser, editors, *Parallel Computational Fluid Dynamics '92*, pages 175–185. North-Holland, 1993.

[98] A. Haldane and R. May. Systemic risk in banking ecosystems. *Nature*, 469:351–355, 2011.

[99] M. Halloran, N. Ferguson, I. Longini S. Eubank, D. Cummings, B. Lewis, S Xu, C. Fraser, A. Vullikanti, T. Germann, D. Wagener, R. Beckman, K. Kadau, C. Barrett, C. Macken, D. Burke, and P. Cooley. Modeling targeted layered containment of an influenza pandemic in the united states. *PNAS*, 105(12):4639–4644, 2008.

[100] M. Elizabeth Halloran, Neil M. Ferguson, Stephen Eubank, Ira M. Longini, Derek A. T. Cummings, Bryan Lewis, Shufu Xu, Christophe Fraser, Anil Vullikanti, Timothy C. Germann, Diane Wagener, Richard Beckman, Kai Kadau, Chris Barrett, Catherine A. Macken, Donald S. Burke, and Philip Cooley. Modeling targeted layered containment of an influenza pandemic in the united states. *Proceedings of the National Academy of Sciences*, 105(12):4639–4644, March 2008.

[101] R. Halstead. Multilisp: A Language for Concurrent Symbolic Computation. *ACM Transactions on Programming Languages and Systems*, October 1985.

[102] Tsuyoshi Hamada, Tetsu Narumi, Rio Yokota, Kenji Yasuoka, Keigo Nitadori, and Makoto Taiji. 42 tflops hierarchical n-body simulations on gpus with applications in both astrophysics and turbulence. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, SC '09, pages 62:1–62:12, New York, NY, USA, 2009. ACM.

[103] Richard Hamming. *Numerical Analysis for Scientists and Engineers*. 1973.

[104] K. Heitmann, P. M. Ricker, M. S. Warren, and S. Habib. Robustness of Cosmological Simulations. I. Large-Scale Structure. *Astrophys. J. Supp.*, 160:28–58, September 2005.

[105] L. Hernquist, F. R. Bouchet, and Y. Suto. Application of the Ewald method to cosmological N-body simulations. *Astrophys. J. Supp.*, 75:231–240, February 1991.

[106] L. Hernquist and N. Katz. TREESPH - A unification of SPH with the hierarchical tree method. *Astrophys. J. Supp.*, 70:419–446, June 1989.

[107] D. Hilbert. Über die stetige abbildung einer linie auf ein flächenstück. *Mathematische Annalen*, 38:459–460, 1891.

[108] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. New York: McGraw-Hill, 1981.

[109] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136:B864, 1964.

[110] Chao Huang and Laxmikant V. Kale. Charisma: Orchestrating migratable parallel objects. In *Proceedings of IEEE International Symposium on High Performance Distributed Computing (HPDC)*, July 2007.

[111] Chao Huang, Orion Lawlor, and L. V. Kalé. Adaptive MPI. In *Proceedings of the 16th International Workshop on Languages and Compilers for Parallel Computing (LCPC 2003), LNCS 2958*, pages 306–322, College Station, Texas, October 2003.

[112] Chao Huang, Gengbin Zheng, Sameer Kumar, and Laxmikant V. Kalé. Performance Evaluation of Adaptive MPI. In *Proceedings of ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming 2006*, March 2006.

[113] J. JáJá. *An introduction to parallel algorithms*. Addison Wesley Longman Publishing Co., Inc. Redwood City, CA, USA, 1992.

[114] Pritish Jetley, Filippo Gioachin, Celso Mendes, Laxmikant V. Kale, and Thomas R. Quinn. Massively parallel cosmological simulations with ChaNGa. In *Proceedings of IEEE International Parallel and Distributed Processing Symposium 2008*, 2008.

[115] Pritish Jetley, Lukasz Wesolowski, Filippo Gioachin, Laxmikant V. Kalé, and Thomas R. Quinn. Scaling hierarchical n-body simulations on gpu clusters. In *Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '10, Washington, DC, USA, 2010. IEEE Computer Society.

[116] Xiangmin Jiao, Gengbin Zheng, Phillip A. Alexander, Michael T. Campbell, Orion S. Lawlor, John Norris, Andreas Haselbacher, and Michael T. Heath. A system integration framework for coupled multiphysics simulations. *Engineering with Computers*, 22(3):293–309, 2006.

[117] J. M. Jiménez, B. L. Lewis, and S. Eubank. Hospitals as complex social systems: agent-based simulation of hospital-acquired infections. In *Proceedings of 2nd International Conference on Complex Sciences: Theory and Applications*, 2012.

[118] John A. Board Jr., Laxmikant V. Kale, Klaus Schulten, Robert D. Skeel, , and Tamar Schlick. Modeling biomolecules: Large sclaes, longer durations. *IEEE Computational Science & Engineering*, 1:19–30, Winter 1994.

[119] Rashmi Jyothi, Orion Sky Lawlor, and L. V. Kale. Debugging support for Charm++. In *PADTAD Workshop for IPDPS 2004*, page 294. IEEE Press, 2004.

[120] L. V. Kale. Application oriented and computer science centered HPCC research. pages 98–105, 1994.

[121] L. V. Kale and Milind Bhandarkar. Structured Dagger: A Coordination Language for Message-Driven Programming. In *Proceedings of Second International Euro-Par Conference*, volume 1123-1124 of *Lecture Notes in Computer Science*, pages 646–653, September 1996.

[122] L. V. Kale and Sanjeev Krishnan. A comparison based parallel sorting algorithm. In *Proceedings of the 22nd International Conference on Parallel Processing*, pages 196–200, St. Charles, IL, August 1993.

[123] L. V. Kale and Sanjeev Krishnan. A comparison based parallel sorting algorithm. In *Proceedings of the 22nd International Conference on Parallel Processing*, pages 196–200, St. Charles, IL, August 1993.

[124] L. V. Kale and Sanjeev Krishnan. Charm++: Parallel Programming with Message-Driven Objects. In Gregory V. Wilson and Paul Lu, editors, *Parallel Programming using C++*, pages 175–213. MIT Press, 1996.

[125] L. V. Kale, B. H. Richards, and T. D. Allen. Efficient parallel graph coloring with prioritization. In *Lecture Notes in Computer Science*, volume 1068, pages 190–208. Springer-Verlag, August 1995.

[126] Laxmikant Kale, Anshu Arya, Abhinav Bhatele, Abhishek Gupta, Nikhil Jain, Pritish Jetley, Jonathan Lifflander, Phil Miller, Yanhua Sun, Ramprasad Venkataraman, Lukasz Wesolowski, and Gengbin Zheng. Charm++ for productivity and performance: A submission to the 2011 HPC class II challenge. Technical Report 11-49, Parallel Programming Laboratory, November 2011.

[127] Laxmikant Kale, Anshu Arya, Nikhil Jain, Akhil Langer, Jonathan Lifflander, Harshitha Menon, Xiang Ni, Yanhua Sun, Ehsan Totoni, Ramprasad Venkataraman, and Lukasz Wesolowski. Migratable objects + active messages + adaptive runtime = productivity + performance a submission to 2012 HPC class II challenge. Technical Report 12-47, Parallel Programming Laboratory, November 2012.

[128] Laxmikant Kalé, Robert Skeel, Milind Bhandarkar, Robert Brunner, Attila Gursoy, Neal Krawetz, James Phillips, Aritomo Shinozaki, Krishnan Varadarajan, and Klaus Schulten. NAMD2: Greater scalability for parallel molecular dynamics. *Journal of Computational Physics*, 151:283–312, 1999.

[129] Laxmikant V. Kalé. The virtualization model of parallel programming : Runtime optimizations and the state of art. In *LACSI 2002*, Albuquerque, October 2002.

[130] Laxmikant V. Kalé. Performance and productivity in parallel programming via processor virtualization. In *Proc. of the First Intl. Workshop on Productivity and Performance in High-End Computing (at HPCA 10)*, Madrid, Spain, February 2004.

[131] Laxmikant V. Kalé, Sameer Kumar, Gengbin Zheng, and Chee Wai Lee. Scaling molecular dynamics to 3000 processors with projections: A performance analysis case study. In *Terascale Performance Analysis Workshop, International Conference on Computational Science(ICCS)*, Melbourne, Australia, June 2003.

[132] Laxmikant V. Kale, Gengbin Zheng, Chee Wai Lee, and Sameer Kumar. Scaling applications to massively parallel machines using projections performance analysis tool. In *Future Generation Computer Systems Special Issue on: Large-Scale System Performance Modeling and Analysis*, volume 22, pages 347–358, February 2006.

[133] L.V. Kalé and S. Krishnan. CHARM++: A Portable Concurrent Object Oriented System Based on C++. In A. Paepcke, editor, *Proceedings of OOPSLA'93*, pages 91–108. ACM Press, September 1993.

[134] L.V. Kalé and Amitabh Sinha. Projections: A preliminary performance tool for charm. In *Parallel Systems Fair, International Parallel Processing Symposium*, pages 108–114, Newport Beach, CA, April 1993.

[135] S.A. Kalogirou. *Solar Energy Engineering: Processes and Systems*. Academic Press, Waltham, MA USA.

[136] George Karypis and Vipin Kumar. METIS: Unstructured graph partitioning and sparse matrix ordering system. University of Minnesota, 1995.

[137] George Karypis and Vipin Kumar. Parallel multilevel k-way partitioning scheme for irregular graphs. In *Supercomputing '96: Proceedings of the 1996 ACM/IEEE conference on Supercomputing (CDROM)*, page 35, 1996.

[138] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the Spread of Influence Through a Social Network. In *Proc. ACM KDD*, pages 137–146, 2003.

[139] D. Kempe, J. Kleinberg, and E. Tardos. Influential Nodes in a Diffusion Model for Social Networks. In *Proc. ICALP*, pages 1127–1138, 2005.

[140] C.H. Koelbel, D.B. Loveman, R.S. Schreiber, G.L. Steele Jr., and M.E. Zosel. *The High Performance Fortran Handbook*. MIT Press, 1994.

[141] W. Kohn and L.J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133, 1965.

[142] C. Koziar, R. Reilein, and G. Runger. Load imbalance aspects in atmosphere simulations. *International Journal of Computational Science and Engineering*, 1(2):215–225, 2005.

[143] Sanjeev Krishnan and L. V. Kale. A parallel adaptive fast multipole algorithm for n-body problems. In *Proceedings of the International Conference on Parallel Processing*, pages III 46 – III 50, August 1995.

[144] Rick Kufrin. Perfsuite: An Accessible, Open Source Performance Analysis Environment for Linux. In *In Proceedings of the Linux Cluster Conference*, 2005.

[145] Sameer Kumar. *Optimizing Communication for Massively Parallel Processing*. PhD thesis, University of Illinois at Urbana-Champaign, May 2005.

[146] Sameer Kumar, Chao Huang, Gheorghe Almasi, and Laxmikant V. Kalé. Achieving strong scaling with NAMD on Blue Gene/L. In *Proceedings of IEEE International Parallel and Distributed Processing Symposium 2006*, April 2006.

[147] V. Kumar. *Introduction to parallel computing*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2002.

[148] Akhil Langer, Jonathan Lifflander, Phil Miller, Kuo-Chuan Pan, , Laxmikant V. Kale, and Paul Ricker. Scalable Algorithms for Distributed-Memory Adaptive Mesh Refinement. In *Proceedings of the 24th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD 2012). To Appear*, New York, USA, October 2012.

[149] Ilya Lashuk, Aparna Chandramowlishwaran, Harper Langston, Tuan-Anh Nguyen, Rahul Sampath, Aashay Shringarpure, Richard Vuduc, Lexing Ying, Denis Zorin, and George Biros. A massively parallel adaptive fast-multipole method on heterogeneous architectures. In *SC '09: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, pages 1–12, New York, NY, USA, 2009. ACM.

[150] Orion Lawlor, Sayantan Chakravorty, Terry Wilmarth, Nilesh Choudhury, Isaac Dooley, Gengbin Zheng, and Laxmikant Kale. Parfum: A parallel framework for unstructured meshes for scalable dynamic physics applications. *Engineering with Computers*, 22(3-4):215–235, September 2006.

[151] Orion Lawlor, Hari Govind, Isaac Dooley, Michael Breitenfeld, and Laxmikant Kale. Performance degradation in the presence of subnormal floating-point values. In *Proceedings of the International Workshop on Operating System Interference in High Performance Applications*, September 2005.

[152] Orion Sky Lawlor. *Impostors for Parallel Interactive Computer Graphics*. PhD thesis, University of Illinois at Urbana-Champaign, December 2004.

[153] Orion Sky Lawlor and L. V. Kalé. Supporting dynamic parallel object arrays. *Concurrency and Computation: Practice and Experience*, 15:371–393, 2003.

[154] D. Lea and W. Gloger. A memory allocator. `http://web.mit.edu/sage/export/singular-3-0-4-3.old/omalloc/Misc/dlmalloc/malloc.ps`, 2000.

[155] C. Lee, W. Yang, and R.G. Parr. Development of the Calle-Salvetti correlation energy into a functional of the electron density. *Phys. Rev. B*, 37:785, (1988).

[156] Chee Wai Lee. *Techniques in Scalable and Effective Parallel Performance Analysis*. PhD thesis, Department of Computer Science, University of Illinois, Urbana-Champaign, December 2009.

[157] J. K. Lenstra, D. B. Shmoys, and E. Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Math. Program.*, 46(3):259–271, 1990.

[158] J. Leskovec, L. Adamic, and B. Huberman. The Dynamics of Viral Marketing. *ACM Trans. on the Web*, 1(1), 2007.

[159] J.R. Levine. *Linkers and Loaders*. Morgan-Kauffman, 2000.

[160] Bryan Lewis, Stephen Eubank, Allyson M Abrams, and Ken Kleinman. In silico surveillance: evaluating outbreak detection with simulation models. *BMC medical informatics and decision making*, 13(1):12, January 2013.

[161] G. F. Lewis, A. Babul, N. Katz, T. Quinn, L. Hernquist, and D. H. Weinberg. The Effects of Gasdynamics, Cooling, Star Formation, and Numerical Resolution in Simulations of Cluster Formation. *Astrophys. J.*, 536:623–644, June 2000.

[162] X. Li, W. Cai, J. An, S. Kim, J. Nah, D. Yang, R. Piner, A. Velamakanni, I. Jung, E. Tutuc, S.K. Banerjee, L. Colombo, and R.S. Ruoff. Large-Area Synthesis of High-Quality and Uniform Graphene Films on Copper Foils. *Science*, 324:1312, (2009).

[163] X. Liu and G. Schrack. Encoding and decoding the Hilbert order. *Software, practice & experience*, 26(12):1335–1346, 1996.

[164] Kwan-Liu Ma, Greg Schussman, Brett Wilson, Kwok Ko, Ji Qiang, and Robert Ryne. Advanced visualization technology for terascale particle accelerator simulations. In *Supercomputing '02: Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, pages 1–11, Los Alamitos, CA, USA, 2002. IEEE Computer Society Press.

[165] Paulo W. C. Maciel and Peter Shirley. Visual navigation of large environments using textured clusters. In *Proceedings of the 1995 symposium on Interactive 3D graphics*, pages 95–ff. ACM Press, 1995.

[166] Sandhya Mangala, Terry Wilmarth, Sayantan Chakravorty, Nilesh Choudhury, Laxmikant V. Kale, and Philippe H. Geubelle. Parallel adaptive simulations of dynamic fracture events. *Engineering with Computers*, 24:341–358, December 2007.

[167] Achla Marathe, Bryan Lewis, Christopher Barrett, Jiangzhuo Chen, Madhav Marathe, Stephen Eubank, and Yifei Ma. Comparing effectiveness of top-down and bottom-up strategies in containing influenza. *PloS one*, 6(9):e25149, 2011.

[168] Achla Marathe, Bryan Lewis, Jiangzhuo Chen, and Stephen Eubank. Sensitivity of household transmission to household contact structure and size. *PloS one*, 6(8):e22461, 2011.

[169] Dominik Marx, Mark E. Tuckerman, and M. Parrinello. The nature of the hydrated excess proton in water. *Nature*, 601:397, (1999).

[170] L. Mayer, T. Quinn, J. Wadsley, and J. Stadel. Formation of Giant Planets by Fragmentation of Protoplanetary Disks. *Science*, 298:1756–1759, November 2002.

[171] M. McPherson, L. Smith-Lovin, and J. Cook. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, 27:415–444, 2001.

[172] Chao Mei, Yanhua Sun, Gengbin Zheng, Eric J. Bohm, Laxmikant V. Kalé, James C.Phillips, and Chris Harrison. Enabling and scaling biomolecular simulations of 100 million atoms on petascale machines with a multicore-optimized message-driven runtime. In *Proceedings of the 2011 ACM/IEEE conference on Supercomputing*, Seattle, WA, November 2011.

[173] Chao Mei, Gengbin Zheng, Filippo Gioachin, and Laxmikant V. Kalé. Optimizing a Parallel Runtime System for Multicore Clusters: A Case Study. In *TeraGrid'10*, number 10-13, Pittsburgh, PA, USA, August 2010.

[174] Esteban Meneses, Greg Bronevetsky, and Laxmikant V. Kale. Dynamic load balance for optimized message logging in fault tolerant hpc applications. In *IEEE International Conference on Cluster Computing (Cluster) 2011*, September 2011.

[175] Esteban Meneses, Celso L. Mendes, and Laxmikant V. Kale. Team-based message logging: Preliminary results. In *3rd Workshop on Resiliency in High Performance Computing (Resilience) in Clusters, Clouds, and Grids (CCGRID 2010).*, May 2010.

[176] J. Michalakes. MM90: a scalable parallel implementation of the Penn State/NCAR Mesoscale Model (MM5). *Parallel Computing*, 23(14):2173–2186, 1997.

[177] John Michalakes, Josh Hacker, Richard Loft, Michael O. McCracken, Allan Snavely, Nicholas J. Wright, Tom Spelce, Brent Gorda, and Robert Walkup. Wrf nature run. In *Proceedings of SuperComputing*, pages 1–6, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[178] Phil Miller, Aaron Becker, and Laxmikant Kal. Using shared arrays in message-driven parallel programs. *Parallel Computing*, 38(12):66 – 74, 2012.

[179] J. Minkel. The 2003 northeast blackout–five years later. *Scientific American*, 2008. 13 August 2008, `http://www.scientificamerican.com/article.cfm?id=2003-blackout-five-years-later.`

[180] M.Levy. Universal variational functionals of electron densities, first-order density matrices, and natural spin-orbitals and solution of the v-representability problem. *Proc. Natl. Acad. Sci. U.S.A.*, 76:6062, (1979).

[181] P. R. Monge and N. S. Contractor. *Theories of Communication Networks*. Oxford University Press, USA, 2003.

[182] B. Moore, F. Governato, T. Quinn, J. Stadel, and G. Lake. Resolving the Structure of Cold Dark Matter Halos. *Astrophys. J. Lett.*, 499:L5–+, May 1998.

[183] E. Moretti. Social learning and peer effects in consumption: Evidence from movie sales. *Review of Economic Studies*, 78:356–393, 2011.

[184] H. Mortveit and C. Reidys. *An Introduction to Sequential Dynamical Systems*. Springer, New York, NY, 2007.

[185] Martin Mundhenk, Judy Goldsmith, Christopher Lusena, and Eric Allender. Complexity of finite-horizon markov decision process problems. *JACM*, 47(4):681–720, July 2000.

[186] National Institutes of Health, 2009. http://www.nigms.nih.gov/Initiatives/MIDAS/.

[187] J. F. Navarro, C. S. Frenk, and S. D. M. White. A Universal Density Profile from Hierarchical Clustering. *Astrophys. J.*, 490:493, December 1997.

[188] NDSSL. Synthetic data products for societal infrastructures and proto-populations: Data set 2.0. Technical Report NDSSL-TR-07-003, NDSSL, Virginia Polytechnic Institute and State University, Blacksburg, VA, 24061, 2007.

[189] M. Newman. The structure and function of complex networks. *SIAM Review*, 45, 2003.

[190] M.E. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 2002.

[191] D.M. Newns, B.G. Elmegreen, X.-H. Liu, and G.J. Martyna. High Response Piezoelectric and Piezoresistive Materials for Fast, Low Voltage Switching: Simulation and Theory of Transduction Physics at the Nanometer-Scale. *Adv. Mat.*, 24:3672, 2012.

[192] D.M. Newns, B.G. Elmegreen, X.-H. Liu, and G.J. Martyna. High Response Piezoelectric and Piezoresistive Materials for Fast, Low Voltage Switching: Simulation and Theory of Transduction Physics at the Nanometer-Scale. *Adv. Mat.*, 24:3672, 2012.

[193] D.M. Newns, B.G. Elmegreen, X.-H. Liu, and G.J. Martyna. The piezoelectronic transistor:A nanoactuator-based post-CMOS digital switch with high speed and low power. *MRS Bulletin*, 37:1071, 2012.

[194] D.M. Newns, J.A. Misewich, A. Gupta C.C. Tsuei, B.A. Scott, and A. Schrott. Mott Transition Field Effect Transistor. *Appl. Phys. Lett.*, 73:780, (1998).

[195] R. Nistor, D.M. Newns, and G.J. Martyna. Understanding the doping mechanism in graphene-based electronics: The role of chemistry. *ACS Nano*, 5:3096, (2011).

[196] A. Odell1, A. Delin1, B. Johansson, N. Bock, M. Challacombe, and A. M. N. Niklasson. Higher-order symplectic integration in Born.Oppenheimer molecular dynamics. *J. Chem. Phys.*, 131:244106, (2009).

[197] Committee on Modeling Community Containment for Pandemic Influenza and Institute of Medicine. *Modeling Community Containment for Pandemic Influenza: A Letter Report*. The National Academies Press, Washington D.C., 2006.

[198] Ehsan Totoni Osman Sarood, Phil Miller and L. V. Kale. 'Cool' Load Balancing for High Performance Computing Data Centers. In *IEEE Transactions on Computer - SI (Energy Efficient Computing)*, September 2012.

[199] J. P. Ostriker and P. J. E. Peebles. A Numerical Study of the Stability of Flattened Galaxies: or, can Cold Galaxies Survive? *Astrophys. J.*, 186:467–480, December 1973.

[200] Douglas Z. Pan and Mark A. Linton. Supporting reverse execution for parallel programs. *SIGPLAN Not.*, 24(1):124–129, 1989.

[201] J. P. Perdew, K. Burke, and M. Ernzerhof. Generalized Gradient Approximation Made Simple. *Phys. Rev. B*, 77:386, (1996).

[202] P. Perzyna. Fundamental problems in viscoplasticity. *Advances in applied mechanics*, 9(C):243–377, 1966.

[203] James C. Phillips, Gengbin Zheng, Sameer Kumar, and Laxmikant V. Kalé. NAMD: Biomolecular simulation on thousands of processors. In *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, pages 1–18, Baltimore, MD, September 2002.

[204] Planck Collaboration, N. Aghanim, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. Balbi, A. J. Banday, R. B. Barreiro, J. G. Bartlett, E. Battaner, K. Benabed, J.-P. Bernard, M. Bersanelli, H. Böhringer, A. Bonaldi, J. R. Bond, J. Borrill, F. R. Bouchet, H. Bourdin, M. L. Brown, C. Burigana, R. C. Butler, P. Cabella, J.-F. Cardoso, P. Carvalho, A. Catalano, L. Cayón, A. Chamballu, R.-R. Chary, L.-Y. Chiang, G. Chon, P. R. Christensen, D. L. Clements, S. Colafrancesco, S. Colombi, A. Coulais, B. P. Crill, F. Cuttaia, A. Da Silva, H. Dahle, R. J. Davis, P. de Bernardis, G. de Gasperis, G. de Zotti, J. Delabrouille, J. Démoclès, F.-X. Désert, J. M. Diego, K. Dolag, H. Dole, S. Donzelli, O. Doré, M. Douspis, X. Dupac, T. A. Enßlin, H. K. Eriksen, F. Finelli, I. Flores-Cacho, O. Forni, P. Fosalba, M. Frailis, S. Fromenteau, S. Galeotta, K. Ganga, R. T. Génova-Santos, M. Giard, J. González-Nuevo, R. González-Riestra, K. M. Górski, A. Gregorio, A. Gruppuso, F. K. Hansen, D. Harrison, A. Hempel, C. Hernández-Monteagudo, D. Herranz, S. R. Hildebrandt, A. Hornstrup, K. M. Huffenberger, G. Hurier, T. Jagemann, J. Jasche, M. Juvela, E. Keihänen, R. Keskitalo, T. S. Kisner, R. Kneissl, J. Knoche, L. Knox, H. Kurki-Suonio, G. Lagache, A. Lähteenmäki, J.-M. Lamarre, A. Lasenby, C. R. Lawrence, S. Leach, R. Leonardi, A. Liddle, P. B. Lilje, M. López-Caniego, G. Luzzi, J. F. Macías-Pérez, D. Maino, N. Mandolesi, R. Mann, F. Marleau, D. J. Marshall, E. Martínez-González, S. Masi, M. Massardi, S. Matarrese, F. Matthai, P. Mazzotta, P. R. Meinhold, A. Melchiorri, J.-B. Melin, L. Mendes, A. Mennella, M.-A. Miville-Deschênes, A. Moneti, L. Montier, G. Morgante, D. Mortlock, D. Munshi, P. Naselsky, P. Natoli, H. U. Nørgaard-Nielsen, F. Noviello, S. Osborne, F. Pasian, G. Patanchon, O. Perdereau, F. Perrotta, F. Piacentini, E. Pierpaoli, S. Plaszczynski, P. Platania, E. Pointecouteau, G. Polenta, N. Ponthieu, L. Popa, T. Poutanen, G. W. Pratt, J.-L. Puget, J. P. Rachen, R. Rebolo, M. Reinecke, M. Remazeilles, C. Renault, S. Ricciardi, T. Riller, I. Ristorcelli, G. Rocha, C. Rosset, M. Rossetti, J. A. Rubiño-Martín, B. Rusholme, M. Sandri, G. Savini, B. M. Schaefer, D. Scott, G. F. Smoot, J.-L. Starck, F. Stivoli, R. Sunyaev, D. Sutton, J.-F. Sygnet, J. A. Tauber, L. Terenzi, L. Toffolatti, M. Tomasi, M. Tristram, L. Valenziano, B. Van Tent, P. Vielva, F. Villa, N. Vittorio, B. D. Wandelt, J. Weller, S. D. M. White, D. Yvon, A. Zacchei, and A. Zonca. Planck intermediate results. I. Further validation of

new Planck clusters with XMM-Newton. *Astronomy and Astrophysics*, 543:A102, July 2012.

[205] S. J. Plimpton and B. A. Hendrickson. A new parallel method for molecular-dynamics simulation of macromolecular systems. *J Comp Chem*, 17:326–337, 1996.

[206] Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.*, 117(1):1–19, 1995.

[207] C. Power, J. F. Navarro, A. Jenkins, C. S. Frenk, S. D. M. White, V. Springel, J. Stadel, and T. Quinn. The inner structure of ΛCDM haloes - I. A numerical convergence study. *Monthly Notices of the Royal Astronomical Society*, 338:14–34, January 2003.

[208] D. Reed, J. Gardner, T. Quinn, J. Stadel, M. Fardal, G. Lake, and F. Governato. Evolution of the mass function of dark matter haloes. *MNRAS*, 346:565–572, December 2003.

[209] D.K. Remler and P.A. Madden. Molecular Dynamics without effective potentials via the Car-Parrinello approach. *Mol. Phys.*, 70:921, (1990).

[210] E. R. Rodrigues, P. O. A. Navaux, J Panetta, and C. L. Mendes. A new technique for data privatization in user-level threads and its use in parallel applications. In *ACM 25th Symposium On Applied Computing (SAC), Sierre, Switzerland*, 2010.

[211] Eduardo R. Rodrigues, Philippe O. A. Navaux, Jairo Panetta, Alvaro Fazenda, Celso L. Mendes, and Laxmikant V. Kalé. A comparative analysis of load balancing algorithms applied to a weather forecast model. In *Proceedings of 22nd IEEE International Symposium on Computer Architecture and High Performance Computing*, Petrópolis - Brazil, 2010.

[212] Eduardo R. Rodrigues, Philippe O. A. Navaux, Jairo Panetta, Celso L. Mendes, and Laxmikant V. Kalé. Optimizing an MPI weather forecasting model via processor virtualization. In *Proceedings of IEEE International Conference on High Performance Computing (HiPC 2010)*, Goa - India, 2010.

[213] D. Romero, B. Meeder, and J. Kleinberg. Differences in the Mechanics of Information Diffusion Across Topics: Idioms, Political Hashtags, and Complex Contagion on Twitter. In *Proceedings of the 20th International World Wide Web Conference (WWW 2011)*, 2011.

[214] Michiel Ronsse and Koen De Bosschere. RecPlay: a fully integrated practical record/replay system. *ACM Trans. Comput. Syst.*, 17(2):133–152, 1999.

[215] H.G. Rotithor. Taxonomy of dynamic task scheduling schemes in distributed computing systems. In *Proceedings of IEE: Computers and Digital Techniques*, volume 141, pages 1–10, 1994.

[216] J. J. Ruan, T. R. Quinn, and A. Babul. The Observable Thermal and Kinetic Sunyaev-Zel'dovich Effect in Merging Galaxy Clusters. *ArXiv e-prints*, April 2013.

[217] Ruth Rutter. Run-length encoding on graphics hardware. Master's thesis, University of Alaska at Fairbanks, 2011.

[218] J. K. Salmon and M. S. Warren. Skeletons from the treecode closet. *Journal of Computational Physics*, 111:136–155, March 1994.

[219] Yanhua Sun Sameer Kumar and L. V. Kale. Acceleration of an Asynchronous Message Driven Programming Paradigm on IBM Blue Gene/Q. In *Proceedings of 26th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Boston, USA, May 2013.

[220] Osman Sarood and Laxmikant V. Kalé. A 'cool' load balancer for parallel applications. In *Proceedings of the 2011 ACM/IEEE conference on Supercomputing*, Seattle, WA, November 2011.

[221] Martin Schulz, Jim Galarowicz, Don Maghrak, William Hachfeld, David Montoya, and Scott Cranford. Open|speedshop: An open source infrastructure for parallel performance analysis. *Scientific Programming*, 16(2-3):105–121, 2008.

[222] Melih Sener, Johan Strumpfer, John A. Timney, Arvi Freiberg, C. Neil Hunter, and Klaus Schulten. Photosynthetic vesicle architecture and constraints on efficient energy harvesting. *Biophysical Journal*, 99:67–75, 2010.

[223] D. Shakhvorostov, R.A. Nistor, L. Krusin-Elbaum, G.J. Martyna, D.M. Newns, B.G. Elmegreen, X. Liu, Z.E. Hughesa, S. Paul, C. Cabral, S. Raoux, D.B. Shrekenhamerd, D.N. Basovd, Y. Songe, and M.H. Mueser. Evidence for electronic gap-driven metal-semiconductor transition in phase-change materials. *PNAS.*, 106:10907–10911, (2009).

[224] S. Shende and A. D. Malony. The TAU Parallel Performance System. *International Journal of High Performance Computing Applications*, 20(2):287–331, Summer 2006.

[225] S.A. Shevlin, A. Curioni, and W. Andreoni. Ab Initio Design of High-k Dielectrics: $La_xY_{1-x}AlO_3$. *Phys. Rev. Lett.*, 94:146401, (2005).

[226] S. Shingu, H. Takahara, H. Fuchigami, M. Yamada, Y. Tsuda, W. Ohfuchi, Y. Sasaki, K. Kobayashi, T. Hagiwara, S. Habata, et al. A 26.58 tflops global atmospheric simulation with the spectral transform method

on the earth simulator. In *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, pages 1–19. IEEE Computer Society Press, 2002.

[227] D. Siegel. Social networks and collective action. *Americal Journal of Political Science*, 53:122–138, 2009.

[228] A. Sinha and L.V. Kalé. Information Sharing Mechanisms in Parallel Programs. In H.J. Siegel, editor, *Proceedings of the 8th International Parallel Processing Symposium*, pages 461–468, Cancun, Mexico, April 1994.

[229] Marc Snir. A note on n-body computations with cutoffs. *Theory of Computing Systems*, 37:295–318, 2004.

[230] Edgar Solomonik and Laxmikant V. Kale. Highly Scalable Parallel Sorting. In *Proceedings of the 24th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, April 2010.

[231] R. Souto, RB Avila, POA Navaux, MX Py, N. Maillard, T. Diverio, HC Velho, S. Stephany, AJ Preto, J. Panetta, et al. Processing mesoscale climatology in a grid environment. In *Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid–CCGrid*, 2007.

[232] V. Springel. The cosmological simulation code GADGET-2. *MNRAS*, 364:1105–1134, December 2005.

[233] V. Springel, J. Wang, M. Vogelsberger, A. Ludlow, A. Jenkins, A. Helmi, J. F. Navarro, C. S. Frenk, and S. D. M. White. The Aquarius Project: the subhaloes of galactic haloes. *MNRAS*, 391:1685–1711, December 2008.

[234] V. Springel, S. D. M. White, A. Jenkins, C. S. Frenk, N. Yoshida, L. Gao, J. Navarro, R. Thacker, D. Croton, J. Helly, J. A. Peacock, S. Cole, P. Thomas, H. Couchman, A. Evrard, J. Colberg, and F. Pearce. Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature*, 435:629–636, June 2005.

[235] J. Stadel, D. Potter, B. Moore, J. Diemand, P. Madau, M. Zemp, M. Kuhlen, and V. Quilis. Quantifying the heart of darkness with GHALO - a multibillion particle simulation of a galactic halo. *MNRAS*, 398:L21–L25, September 2009.

[236] J. G. Stadel. *Cosmological N-body Simulations and their Analysis*. PhD thesis, Department of Astronomy, University of Washington, March 2001.

[237] Yanhua Sun, Gengbin Zheng, Chao Mei Eric J. Bohm, Terry Jones, Laxmikant V. Kalé, and James C.Phillips. Optimizing fine-grained communication in a biomolecular simulation application on cray xk6. In *Proceedings of the 2012 ACM/IEEE conference on Supercomputing*, Salt Lake City, Utah, November 2012.

[238] Yanhua Sun, Gengbin Zheng, L. V. Kale, Terry R. Jones, and Ryan Olson. A uGNI-based Asynchronous Message-driven Runtime System for Cray Supercomputers with Gemini Interconnect. In *Proceedings of 26th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Shanghai, China, May 2012.

[239] Emad Tajkhorshid, Aleksij Aksimentiev, Ilya Balabin, Mu Gao, Barry Isralewitz, James C. Phillips, Fangqiang Zhu, and Klaus Schulten. Large scale simulation of protein mechanics and function. In Frederic M. Richards, David S. Eisenberg, and John Kuriyan, editors, *Advances in Protein Chemistry*, volume 66, pages 195–247. Elsevier Academic Press, New York, 2003.

[240] Emad Tajkhorshid, Peter Nollert, Morten Ø. Jensen, Larry J. W. Miercke, Joseph O'Connell, Robert M. Stroud, and Klaus Schulten. Control of the selectivity of the aquaporin water channel family by global orientational tuning. *Science*, 296:525–530, 2002.

[241] Claudia Taylor, Achla Marathe, and Richard Beckman. Same influenza vaccination strategies but different outcomes across us cities? *International Journal of Infectious Diseases*, 14(9):e792 – e795, 2010.

[242] T.N. Theis and P.M. Solomon. In Quest of the "Next Switch": Prospects for Greatly Reduced Power Dissipation in a Successor to the Silicon Field-Effect Transistor. *Proceedings of the IEEE*, 98:2005, (2010).

[243] GJ Tripoli and WR Cotton. The Colorado State University three-dimensional cloud/mesoscale model. Technical Report 3, Atmos, 1982.

[244] M. Tuckerman, G. Martyna, M.L. Klein, and B.J. Berne. Efficient Molecular Dynamics and Hybrid Monte Carlo Algorithms for Path Integrals. *J. Chem. Phys.*, 99:2796, (1993).

[245] Ramkumar V. Vadali, Yan Shi, Sameer Kumar, L. V. Kale, Mark E. Tuckerman, and Glenn J. Martyna. Scalable fine-grained parallelization of plane-wave-based ab initio molecular dynamics for large supercomputers. *Journal of Comptational Chemistry*, 25(16):2006–2022, Oct. 2004.

[246] J. W. Wadsley, J. Stadel, and T. Quinn. Gasoline: a flexible, parallel implementation of TreeSPH. *New Astronomy*, 9:137–158, February 2004.

[247] R.L. Walko, L.E. Band, J. Baron, T.G.F. Kittel, R. Lammers, T.J. Lee, D. Ojima, R.A. Pielke Sr, C. Taylor, C. Tague, et al. Coupled atmosphere–biophysics–hydrology models for environmental modeling. *Journal of Applied Meteorology*, 39(6), 2000.

[248] Yuhe Wang and John Killough. A new approach to load balance for parallel compositional simulation based on reservoir model over-decomposition. In *2013 SPE Reservoir Simulation Symposium*, 2013.

[249] M. S. Warren and J. K. Salmon. A parallel hashed oct-tree n-body algorithm. In *Proceedings of the 1993 ACM/IEEE conference on Supercomputing*, Supercomputing '93, pages 12–21, New York, NY, USA, 1993. ACM.

[250] M.S. Warren, J.K. Salmon, D.J. Becker, M.P. Goda, T. Sterling, and W. Winckelmans. Pentium pro inside: I. a treecode at 430 gigaflops on asci red, ii. price/performance of $50/mflop on loki and hyglac. In *Supercomputing, ACM/IEEE 1997 Conference*, page 61, nov. 1997.

[251] S. D. M. White, C. S. Frenk, and M. Davis. Clustering in a neutrino-dominated universe. *Astrophys. J. Lett.*, 274:L1–L5, November 1983.

[252] X.-P. Xu and A. Needleman. Numerical simulation of fast crack growth in brittle solids. *Journal of the Mechanics and Physics of Solids*, 42:1397–1434, 1994.

[253] M. Xue, K.K. Droegemeier, and D. Weber. Numerical Prediction of High-Impact Local Weather: A Driver for Petascale Computing. *Petascale Computing: Algorithms and Applications*, pages 103–124, 2007.

[254] Jae-Seung Yeom, Abhinav Bhatele, Keith Bisset, Eric Bohm, Abhishek Gupta, Laxmikant V. Kale, Madhav Marathe, Dimitrios S. Nikolopoulos, Martin Schulz, and Lukasz Wesolowski. Overcoming the scalability challenges of contagion simulations on Blue Waters. Technical Report 13-057, NDSSL, Virginia Bioinformatics Institute at Virginia Tech, 2013.

[255] Y. B. Zeldovich and R. A. Sunyaev. The Interaction of Matter and Radiation in a Hot-Model Universe. *Astrophysics & Space Science*, 4:301–316, July 1969.

[256] Gongpu Zhao, Juan R. Perilla, Ernest L. Yufenyuy, Xin Meng, Bo Chen, Jiying Ning, Jinwoo Ahn, Angela M. Gronenborn, Klaus Schulten, Christopher Aiken, and Peijun Zhang. Mature HIV-1 capsid structure by cryo-electron microscopy and all-atom molecular dynamics. *Nature*, 497:643–646, 2013. doi:10.1038/nature12162.

[257] Gengbin Zheng. *Achieving high performance on extremely large parallel machines: performance prediction and load balancing.* PhD thesis, Department of Computer Science, University of Illinois at Urbana-Champaign, 2005.

[258] Gengbin Zheng, Abhinav Bhatele, Esteban Meneses, and Laxmikant V. Kale. Periodic Hierarchical Load Balancing for Large Supercomputers. *International Journal of High Performance Computing Applications (IJHPCA)*, March 2011.

[259] Gengbin Zheng, Orion Sky Lawlor, and Laxmikant V. Kalé. Multiple flows of control in migratable parallel programs. In *2006 International Conference on Parallel Processing Workshops (ICPPW'06)*, pages 435–444, Columbus, Ohio, August 2006. IEEE Computer Society.

[260] Gengbin Zheng, Xiang Ni, and L. V. Kale. A Scalable Double In-memory Checkpoint and Restart Scheme towards Exascale. In *Proceedings of the 2nd Workshop on Fault-Tolerance for HPC at Extreme Scale (FTXS)*, Boston, USA, June 2012.

[261] Gengbin Zheng, Lixia Shi, and Laxmikant V. Kalé. FTC-Charm++: An In-Memory Checkpoint-Based Fault Tolerant Runtime for Charm++ and MPI. In *2004 IEEE International Conference on Cluster Computing*, pages 93–103, San Diego, CA, September 2004.

[262] Gengbin Zheng, Terry Wilmarth, Praveen Jagadishprasad, and Laxmikant V. Kalé. Simulation-based performance prediction for large parallel machines. In *International Journal of Parallel Programming*, volume 33, pages 183–207, 2005.