Scale-bridging computational materials science: heterogeneous algorithms for heterogeneous platforms

Presented by Tim Germann with Lecture Notes by Amanda Peters Randles

February 23, 2012

1 Introduction

Materials science applications have often been some of the first applications run on each generation of supercomputer. They have not only provided great scientific insight, but have served as a testbed for exploring new computational approaches for tackling massive concurrency, resiliency, and data bottlenecks. Traditionally material science problems have been approached more along the lines of sequentially coupled length or time scales. The move toward the greater use of concurrent multiscale methods is crucial from both the application and the computer science perspectives, and maps well to the increasingly heterogenous and hierarchical nature of computer architectures. In this paper, we will discuss the state of the art in computational materials science and motivate the need for a shift to a co-design paradigm in which the algorithms, applications, and architectures are developed simultaneously.

In current materials science research, applications are hitting the bounds of single-scale models in both time and length scales. By coupling previous work completed by researchers focusing on specific scales, researchers may be able to tackle some of the larger unanswered questions in their fields. While it may not be feasible to do a fully atomistic scale model of systems consisting of many billions of atoms, a coupled approach can help to recover the relevant physics. For example, when studying issues such as fluid instability in which a heavy fluid lies on top of a lighter fluid, the region of interest may only be the interfacial layer. This layer where atomic resolution is needed may only account for a small fraction of the several billion atoms making up the fluids themselves. Fluid further from the interface is homogenous and could therefore be modeled using a continuum finite element (FE) method to recover all necessary attributes. This need for varying

resolution through the system is also seen in the case of a shockwave propagating through iron. Atomic resolution is not needed ahead of the shockwave or very far after. This disparity in resolution needs across a system is becoming more and more apparent as overall simulation size increases. A method for seamlessly coupling between scales in a single simulation is required.

In the following sections, we will motivate this need for multiscale methods and their coupling to next generation architectures. We will discuss the state of the art in sequential and concurrent materials science applications followed by a case study of the optimizing an application for a specific architecture. We will cover where we are and where we would like to go in multiscale materials science as we look ahead at the new architectures.

1.1 Coupling between the science and the computer architecture

As we look toward exascale computing, computer architectures are becoming increasingly heterogeneous and hierarchical, with greatly increased flop/byte ratios. The machines are becoming more communication dominated. The algorithms, programming models, and tools that will thrive in this environment must mirror these characteristics. Not only will the single program multiple data (SPMD) paradigm no longer be viable but the time scales of the simulations will necessitate changes to the applications.

SPMD bulk synchronous parallelism will not be optimal for next generation systems as the overhead associated with simply invoking a global synchronization across over one billion cores could be large. Beyond that, resiliency and fault tolerance start to become more pressing questions as we can no longer guarantee that the billion cores from one time step will be maintained to the following time step. It is becoming increasingly more important that MPI and/or the application has the ability to drop or replace nodes as well as to recover from soft and hard errors while anticipating faults. Traditional global checkpoint/restart is also being impractical as system size increases.

The time scale of the simulations also needs to be considered. For current single-scale Molecular Dynamics (MD) simulations, the time step can be on the order of one femtosecond and the memory-size of the processor dictates the number of atoms that can be run in one simulation. By simply growing the single-scale application with the size of the next generation supercomputer, we merely enable the overall size of the system that can be simulated; however, typically the goal is to run for many time steps. It is useless to increase the simulation to the scale of trillions of atoms if they can only be simulated for a few time steps. What is needed is the ability for the simulation to run long enough that the dynamics of interest can evolve. For example, when modeling a sound wave, one would at least

need it to propagate entirely through the material. Inherently, there is a tradeoff between the size of the system being modeled and the duration of time encapsulated by the simulation. Just adding more processors is not good enough.

For short range potentials, there is a tradeoff between the system size and the limit in which the application becomes communication bound. There is a point that the bookkeeping overhead for each time step actually overwhelms the time for completing that step and sets a limit for how fast the step can be finished. On current machines, the largest simulations are for millions or billions of atoms typically for tens of nanoseconds. As we look toward exascale computing, memory may increase by two orders of magnitude but current projections indicate that the number of processing elements per node will not increase in turn causing this time-scale problem to persist. In order to overcome it, the time scales of algorithms need to be extended or scales need to be coupled.

There is a need to introduce more detailed physics into computational material science applications in a way that escapes the traditional synchronous SPMD paradigm and exploits the exascale hardware.

2 State of the art in computational materials science

Currently the majority of material science applications are still single-scale applications meaning that they embody one regime like *ab initio* calculations. These applications receive some of the largest time allocations on today's supercomputers, so in order to understand the state of the art, it is important to review a few of these.

Moreover, when people refer to multiscale modeling in material science, they often mean the coupling of these models in a sequential manner. In this case, information is passed up a hierarchy of coupled length/time scales through a sequence of subscale models and parameters. Figure 1 shows on overview of the various single-scale approaches that may be coupled in such a manner. One great example was completed by Baron et al. focusing on a multiscale strength model and directly compares the methods from the *ab initio* up to continuum [1].

Sequential multiscale models often start at the low scale where *ab initio* calculations are used to calculate quantities such as force models or the equation of state for by solving the Schrodinger equation for materials under different constraints. This typically involves solving an eigenvalue problem in periodic basis sets with many fast fourier transforms (FFTs) or dense numerical algebra. This is followed by a classical molecular dynamics (MD) simulation that moves from the quantum regime to a scale on the order of microns and nanoseconds for larger processor counts. The MD models use the force fields calculated by the *ab initio* methods

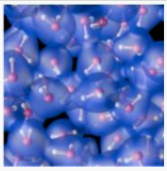
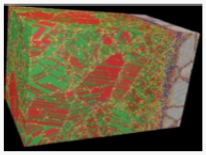
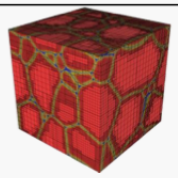
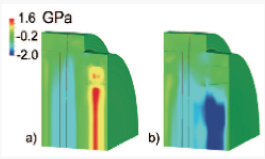
Ab-initio Methods	Molecular Dynamics	Phase-Field Modeling	Continuum Methods
Inter-atomic force model, equation of state	Defect and interface mobility, nucleation	Direct numerical simulation of multi-phase evolution	Multi-phase material response, experimental observables
			
Length/time: nm, ps Code: Qbox/LATTE Motif: Particles and wavefunctions, plane wave DFT with nonlocal norm-conserving, ScaLAPACK, BLACS, and custom parallel 3D FFTs Prog. Model: MPI	Length/time: m, ns Code: SPaSM/ddcMD Motif: Particles, domain decomposition, explicit time integration, neighbor and linked lists, dynamic load balancing, parity error recovery, and *in situ* visualization Prog. Model: MPI + Threads	Length/time: 100 m, s Code: AMPE/GL Motif: Regular and adaptive grids, implicit time integration, real-space and spectral methods, complex order parameter (phase, crystal, species) Prog. Model: MPI	Length/time: cm, ms Code: VP-FFT/ALE3d Motif: Regular and irregular grids, implicit time integration, 3D FFTs, polycrystal and simgle crystal plasticity, Prog. Model: MPI

Figure 1: A table showing how information is passed up a hierarchy of coupled length/time scales via a sequence of subscale models and parameters.

to study effects such as defects, growth, and interface mobility. Unlike the low level models that rely on meshes, MD modeling involves a set of particles that are propagated via a simple algorithm. This enables the exploration of the impacts of computational factors like load balancing and resiliency. When working with extremely large systems consisting of billions of atoms, new problems are introduced as the limits on visualization begin to be pushed. This raises the issue of how to analyze and visualize massive data sets *in situ* and emphasizes the overall need for data reduction. The ability to check point and restart also becomes strained as it is increasingly impractical to checkpoint a trillion atom system. This push on the computational demands have caused a close tie between the needs for next generation systems and the potential performance of even the single-scale applications.

In the following sections we will give a high level overview of some of the key single-scale material science applications. For a wider overview of material science applications and their performance on the Blue Gene supercomputer see reference [2].

2.1 *Ab initio* Methods

Ab-initio methods are used to model particles and wave functions, often plane wave density functional theory (DFT) with non-local and norm-conserving forces. These codes often use ScaLAPACK, BLACS, and custom parallel 3D FFTs alongside MPI communication. The length scale dealt with typically in these simulations is on the order of nanometers while the time scale is on the order of picoseconds. Qbox is a strong example of one such application that has been shown to scale well on the large-scale supercomputers.

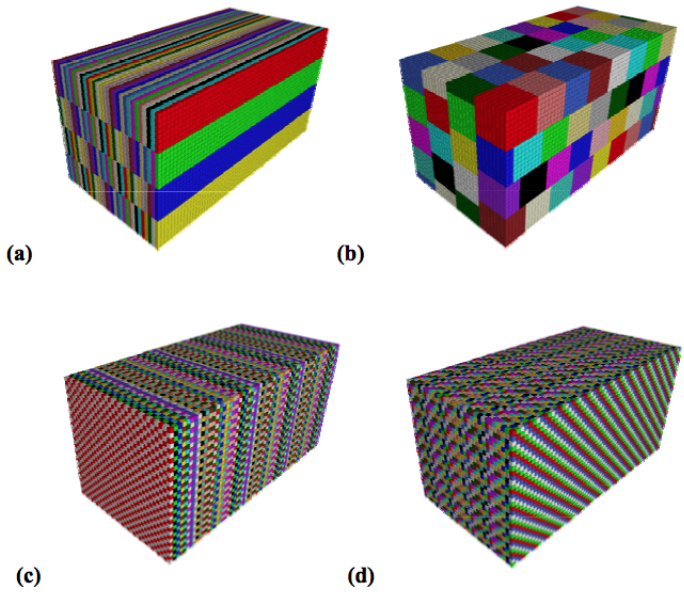


Figure 2: Ilustration of different node mappings for a 64k-node partition. Each color represents the nodes belonging to one 512-node column of the process grid. (a) default (b) compact (c) bipartite (d) quadpartite [3].

Qbox is a first-principles molecular dynamics (quantum electrons, classical nuclei) application based on the plane-wave, pseudopotential method for electronic structure calculations developed at Lawrence Livermore National Laboratory. Qbox implements first principles molecular dynamics (FPMD) within the DFT framework and has been used to simulate liquids and solids in extreme conditions. An effective potential is used to avoid solving for all of the electrons. It has been used for FPMD simulations of heavy metals like molybdenum or tantalum

for the evaluation of isolated defects in the metals. This application demonstrations several key issues that are met with in large-scale parallel applications. First, different parts of the equations dealt with have different ideal representations to make their solutions simpler. For example, the kinetic and potential terms are sparse in either the momentum or the real space making it ideal to go back and forth between the two representations. This necessitates frequent 3D FFTs, making optimal data layout and representation an issue especially for hybrid architectures. Secondly, there is complexity in maintaining orthogonality which leads to more linear algebra. Finally, the team developing Qbox demonstrated that optimal node mapping is non-obvious in this case which contributed to the 2006 Gordon Bell Peak Performance Award [3]. Initially they attempted the "compact" representation seen in Figure 2 b, in which the surface to volume ratio was minimized, but this actually proved to show a lower performance than the default node mapping. By leveraging a quadpartite mapping, as shown in Figure 2, Gygi et al. were able to increase their performance from 39.5 teraflops with the default mapping to 64.7 teraflops. This result is demonstrative of the shift from mathematically driven optimizations to data communication optimizations.

2.2 Dislocation Dynamics

ParaDis (Parallel Dislocation Simulator) is a large-scale dislocation dynamics simulation code to study the fundamental mechanics of plasticity that was developed originally at Lawrence Livermore National Laboratory [4]. In these simulations the plastic strength of materials is computed by tracing the evolution of dislocation lines over time with the goal of allowing scientists to gain insight into the nature of self-induced strengthening. By relying on the line-tracking model that ignores the material not impacted by the defect, the degrees of freedom are dramatically reduced. ParaDis is the state of the art in this domain with line defects discretized into nodes or segments and then in each time step of the algorithm, the forces that each exerts on the other are computed and each dislocation is propagated forward. In dislocation, the simulation starts with simple lines that stress the system, multiply, grow, and form junctions. The limit to the simulation is the dislocation density in the system that needs to be resolved. As it increases, the system becomes increasingly inhomogenous in its spatial distribution, resulting in load balancing challenges. Use of a minimal set of topological operators alongside recursive partitioning of the problem domain were used to maintain scalability [2]. In the early work on Blue Gene/L, a 1.8 speedup was achieved in going from 4000 to 8000 processors. Beyond that, however, the load balancing issues from the evolution of the dislocation structure inhibit the scaling performance.

2.3 Molecular Dynamics

In molecular dynamics (MD), the length and time scales can vary by quite a bit. Typically in materials sicence, scientists are concerned with the simulation of the movement and interaction of many particles on the length scale of meters in the span of nanoseconds. Common computational issues to be dealt with consist of domain decomposition, explicit time integration, neighbor and linked lists, dynamic load balancing, parity error recover, and *in situ* visualization. Applications in this domain often make use of MPI and threads for communication. Among the various MD applications, different domain decomposition strategies are used for in-node breakdown. ddcMD leverages particle-based decomposition whereas SPaSM uses the more traditional spatial breakdown. In some instances, such as the work by D.E. Shaw on Anton, decomposition is bond-based and there may be more processors than particles [5]. In this section, we'll touch on both ddcMD and SPaSM in more detail.

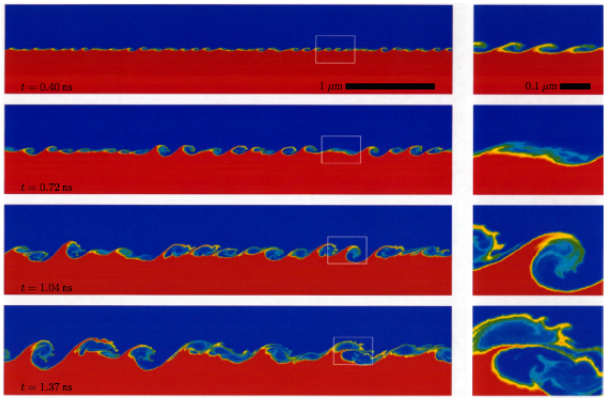


Figure 3: Evolution of Kelvin-Helmholtz instability modeled using molecular dynamics. The color indicates the local density, with red the density of copper and blue the density of aluminum. Only the region near the interface is shown. The fluid flows to the right at the top, and to the left at the bottom of each panel. The frames to the right enlarge the outlined rectangular region in the corresponding panel to the left. [6]

One MD application that has been shown to exhibit strong scaling across multiple platforms is ddcMD (domain decomposition Molecular Dynamics). This code was developed at Lawrence Livermore National Laboratory and was used in the papers that were awarded the Gordon Bell Performance Prize in 2005 and 2007 and one that was a finalist in 2009. In 2005, this application hit the milestone of achieving performance rates as high as 107 TFlops [7]. In 2007, the team achieved the

first micron-scale simulation of a Kelvin-Helmholtz instability using MD as shown in Figure 3. Advances focused on fault tolerance, kernel optimization, and parallel I/O efficiency [6]. The highly accurate Model-generalized pseudo-potential theory (MGPT) potentials are used. MGPT is a computational expensive potential that enables the avoidance of redundant communication and computation. Another key advancement made by the developers of this code, was the focus on parity error recovery. In MD applications, the memory footprint is very small as the state can be defined by simple atom positions and velocities. By periodically storing the current system state in memory, an in memory restart is enabled in the case of an unrecoverable parity error detection [8] .

Another particle-based molecular dynamics code is SPasM (Scalable Parallel Short-range Moleculare Dynamics). It is a classical molecular dynamics code developed at Los Alamos National Laboratory. Papers leveraging this code won the Gordon Bell Performance Prize in 1993 and the Gordon Bell Price/Performance Prize in 1998, and were finalists in both 2005 and 2008. Pairwise interactions are investigated via potentials such as Lennard-Jones or via the many-body embedded atom method (EAM). Finite-range interactions were modeled and *O(N)* computational scaling was achieved. This is a good example of strong spatial decomposition on both shared and distributed memory architectures. SPaSM has evolved over time through optimization for different architectures starting with the connection machine all the way up to LANL's RoadRunner. It's a simple MD algorithm where instead of decomposing the problem by particle, the developers divide by space among processors. This is a reasonable approach as the bulk of what materials are being modeled with this application are homogenous systems. There is a rapid search to find atoms that fall within the potential interacting range at the boundaries of the domains allowing further subdivision. As in ddcMD, there is a small memory footprint in which only the position and velocity of each atom is needed to store the state. This has enable simulations to push up to the first ever trillion-atom simulation. One of the main applications of this code has been to model the propagation of a shockwave through iron polycrystal as shown in Figure 4 [9]. As shown, the shockwave compresses the bcc lattice. Models such as these are used to study the mechanism of phase transformation and to assess both the kinetics needed by high length scales and the new mechanical properties of the new phase. In these large-scale simulations, it is particularly important to visualize the results *in situ* not only to identify new mechanisms, but also to assist in debugging. Often times, bugs that come from cross-processor boundary issues can be identified faster through viewing where the numerical problem originally occurs. To this end, throughout the development of this application, an effort was made to enable *in situ* visualization and analysis libraries to allow for runtime *steering* [10] [11]

In the case of the polycrystal model, it is a well known behavior that the

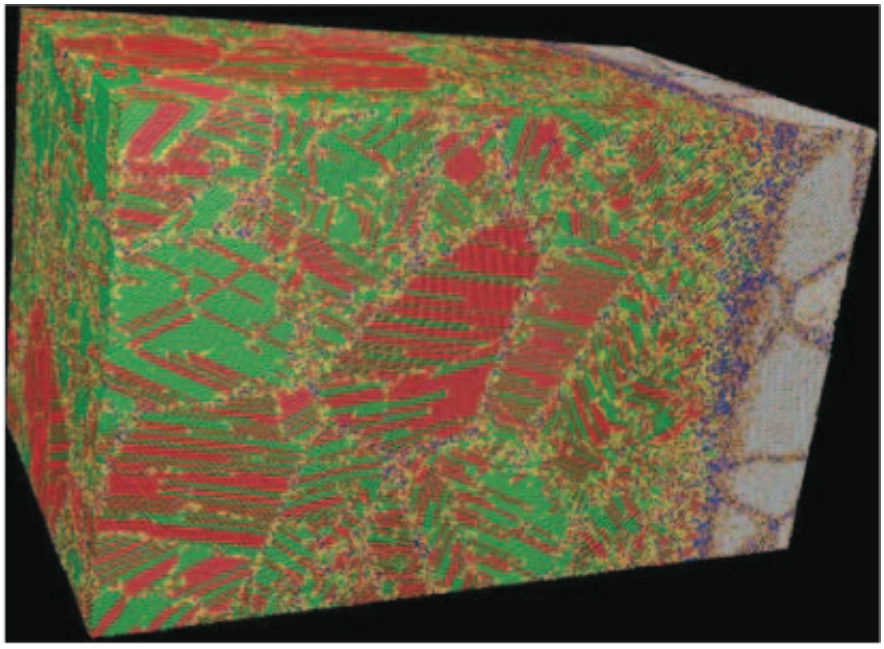


Figure 4: Simulation of an iron polycrystal subjected to a 39 GPa shock loading using the molecular dynamics application SPaSM. [9]

.

strength of the material depends on the grain size. In the large engineering scale limit, the mechanical strength of the material increases as the inverse square root of the grain size. This drives the simulation to the nanoscale but hits a limit as single atom materials are known to be weak. The tradeoff has shown the ideal length scale to be on the order of tens of nanometers. In order to model such a system at least 100 grains of 50 nanometers would be needed, leading to 10^9 atoms. A soundwave would take about a nanosecond to propagate through the material, but most simulations will likely need to run longer thus requiring millions to billions of time steps. This combination of length and time scales starts to hit the limits of what can be modeled with single scale material science applications.

3 State of the art in concurrent multiscale

In this section, we will briefly describe more commonly used techniques as well as the historical ones that have played an important role in the evolution of this field.

There are several methods for concurrent multiscale models. Lu and Kaxiras

provide a great review article on these [12]. As previously discussed, sequential multiscale techniques require a separation of length and timescales and prior knowledge of relevant physical processes. This works well if you have an idea *a priori* of what the relevant processes are and can develop models for them. For instance, when studying turbulence there is a strong coupling between the different time scales. For many systems like this, the physics is inherently multiscale, with a strong coupling between the behavior occurring at different length/time scales. In such cases, it is no longer possible to integrate out degrees of freedom via approximate models as one moves from finer to coarser scales. The models are useful for developing ways to do data reduction and work out how to identify what is the essential data and what is really important. In an MD simulation with billions or trillions of atoms, checkpointing all of them is unnecessary and costly when it may only be the interface atoms that matter and areas further from the interface could be reconstructed from an average state. Lu summarized it well saying, "Multiscale models are also useful for gaining physical insight ... [and] can be an effective way to facilitate the reduction andanalysis of data, which sometimes can be overwhelming." [12]

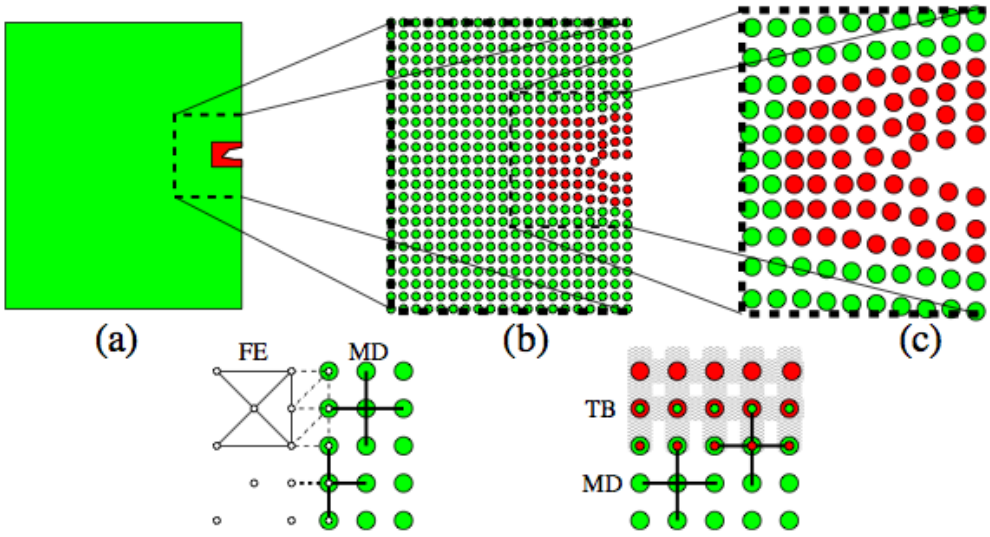


Figure 5: Crack propagation [12].

A common technique is referred to as the *onion* methods in which finer length scale model regions are embedded within coarser scale regions. A classic example is looking at fracture dynamics where you have a crack propagating through

a material as shown in Figure 5. Further away from the crack front, there is an elastic solid where a continuum model can be used but at the crack tip, the crack is propagating forward with individual bond breaking events requiring atomic resolution. The challenge is then how to couple the continuum region with the atomistic region while maintaining consistency between scales, with rigorous *handshaking* in overlap regions. One common way is that in the atomistic region, to use ghost atoms whose positions are determined by the FE in the boundary region and then the FE region has ghost cells from the MD. In this case each scale is simulated and then coupled through the boundary regions. In some cases this may be carried on past two scales to more scales for example by describing the bond breaking with tight bonding or quantum methods leading to multiple scales [12].

3.1 Quasicontinuum method

For quasi continuum (QC) methods, in regions of smoothly varying displacement (i.e. linear elastic deformation), full atomistic detail is replaced by representative atoms or *repatoms*. A fully atomistic representation is used near the crack or dislocation, but further away, where it is just an elastic material, the repatoms are used that describe the local elastic response to the material. As the simulation goes forward in time, the boundary regions may evolve and the size of the fully atomistic region my grow until it overwhelms the computer and the simulation terminates. A common theme with these early techniques is that they provide simple methods for static simulations making them optimal if you are trying to find a minimal energy configuration in a zero temperature static solution. However, adding dynamics or a finite temperature can pose a challenge. To read more about it please see reference [13].

3.2 Macroscopic, atomistic, *ab initio* dynamics (MAAD)

As opposed to the previously discussed QC method, macroscopic, atomistic *ab intio* dynamics (MAAD) is an example where dynamics were done. Three scales are coupled here: finite element, atomistic molecular dynamics, and quantum tight binding. This is shown in Figure 6. The tight binding is used ahead of the crack tip where bonds are breaking, MD around surrounds that and then FE is used for the furthest away regions. This is written as a Hamiltonian in which there are terms for each single scale and the challenge comes in at the handshaking regions between FE/MD and MD/TB. For quantum simulations, the issue is how to handle the dangling bonds with atoms you have carved out of the tight bonding model. For covalent systems like silicon, this can be done by adding pseudohydrogen atoms that solve the coordination of the silicon [14]. The coupling of continuum and

atomistic models for metals is still an open question.

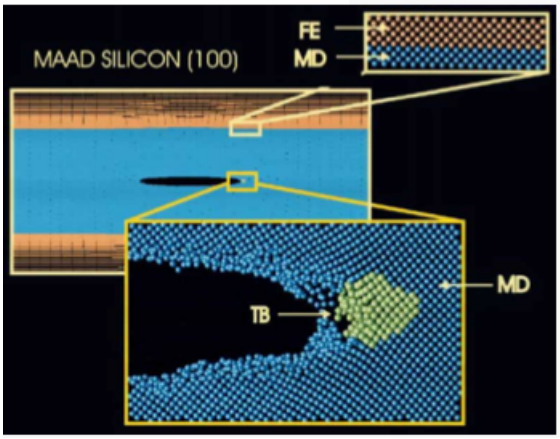


Figure 6: MAAD Silicon [14].

3.3 Coarse-grained molecular dynamics (CGMD)

In studying the behavior of these techniques, one major challenge is to avoid the introduction of spurious waves from the interface region when moving from the atomistic region and coarsening to a FE region. For instance, when simulating a sound wave that has a wavelength that is less than the length of FE cells, there is a problem as the wave moves from the atomistic region where it is supported to the FE region where it is not. There have been elegant solutions but they are very expensive and use particle history memory and are non-local in time and space so do not scale well. The presents a tradeoff between an approximate algorithm that scales well with this spurious wave reflection or one that scales poorly without any wave reflection.

One method that tries to deal with this is coarse-grained molecular dynamics (CGMD). This provides consistent transfer between the scales and has been shown to be successful in test cases to date. Addresses difficulties in a smooth transition between atomistic and continuum regions by replacing the continuum FE mesh with a continuum model developed by statistical coarse-graining. As the continuum mesh size approaches the atomistic scale, CGMD equations of motion become MD equations. As the behavior is based solely on the MD model, there are no continuum parameters, consistent treatment of phonon modes, and a smoother elastic wave propagation between regions. Furthermore, CGMD was designed for finite-temperature dynamics [15]. This method has shown a lot of promise but has

not really been extended.

3.4 Heterogenous Multiscale Method (HMM)

Another approach by W. E's group at Princeton is based on the heterogenous multiscale method (HMM) where instead of coupling from an energy perspective starting at MD and driving upward, it is driven by a macroscale solver like finite element or volume to give information as needed to drive the macroscale solver forward. As shown in the previous discussions, energy-based methods with coarse-grained Hamiltonians have several challenges such as the need to deal with time scales between the regions are still coupled, matching conditions at boundaries often either cause spurious reflections or are expensive and non-scalable, and finite temperature, dynamics simulations are difficult. The HMM philosophy is to use microscale models (e.g. MD) to supply missing data such as constitutive laws or kinetic relations for a macro-solver like FEM. This model is typically used for two types of problems. For *Type A problems* there are isolated defects treated via adaptive model refinement. For *Type B problems* there is on-the-fly computations of constitutive information [16].

3.5 Comparison

Miller and Tadmore provide a review that compares fourteen of these difference methods and analyzes their performance. They mention that none of these have been pushed to scale and summarize their findings saying, "Multiscale methods like the ones discussed in this review show much promise to improve the efficiency of atomistic calculations, but they have not yet fully realized this potential. This is in part because the focus to date has mainly been on development of the methodology as opposed to the large-scale application to materials problems. ... In order for multiscale methods to compete with, or eventually replace, atomistics it is necessary that the methods be implemented in 3D, parallel codes optimized to the same degree as atomistic packages." [17] For information on object kinetic Monte Carlo refer to [18] and for accelerated molecular dynamics methods refer to [19].

4 Case Study in Co-Design: Experience with SPaSM on LANL RoadRunner

The ability to address the large unanswered questions in materials science will continue to require the use of the largest supercomputers. In order to harness the power of such systems, it is impossible to develop the codes with ignorance to the

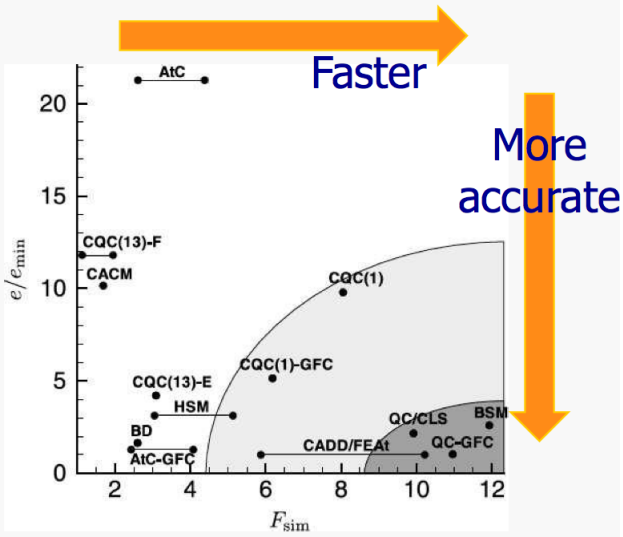


Figure 7: Performance and scalability of multiscale material methods. [17].

system's underlying architecture. The following case study demonstrates this by showing that making an architecture-centric re-design resulted in a 10x speedup of a large-scale MD application.

As we approach exascale computing, we have seen a trend toward hierarchical hybrid computing. Computational material science codes have been shown to perform extremely well on these types of architectures but can require careful attention. Motivated by the trend towards GPUs and other accelerators over the years, we're going to focus on one case study of optimizing a particular material science application, SPaSM which was discussed previously, to a large-scale hybrid supercomputer, LANL RoadRunner. This system was the first petaflop machine and was a hybrid cluster of clusters.

4.1 LANL RoadRunner

In the case of LANL RoadRunner, the choice was made to use the cell processor found in the Sony Playstation as the core accelerator. The cell processor somewhat resembles the CM5 connection machine which had 8 vector units and the peak performance of 1 gigaflop. The cell, however, is a 100 gigaflop processor with 8 synergistic processing units (SPEs). The question then was how to leverage these to work together, especially given that they used very little of the PowerPC features. For example, branch prediction had been stripped down. There was a one-to-one mapping of cell and opteron processors creating a truly hybrid archi-

tecture. This balance presented the challenge of taking applications developed for traditional systems and optimizing them for an architecture with heavy use of the new accelerator.

Drawing intuition from the gaming community which has been heavily reliant on accelerators for a while, the ideal paradigm would be to take the number of tasks needed to be completed in the Cell and write them in such a way that the data feed down to the SPE and up to the PPE (or even the CPU) can be overlapped. This enables the ammortization of the computation and ideally overlap direct memory address instructions with computations so that you can double or triple buffer the incoming data, data being worked on, and the outgoing data to hide computation time. In practice, this is much more complicated. In the case of the LANL Road-Runner, there are two different compilations needed for each part of the Cell and one for the opteron resulting in three compilers and different communication libraries. On top of that, there were two different types of byte ordering (Big and Little Endian).

4.2 SPaSM

SPaSM was originally written 20 years ago for the Connection Machine when both memory and computation were the bottle necks. At that time, communication could be viewed as cheap when there were only 32 megabytes for the SPARC-based node. For communication, initially there was the CM-5 fat tree network and then the Cray T3D and IBM Blue Gene/L both had a 3d torus. The 3D torus is ideal for 3D spatial decomposition when the bulk of the communication is nearest neighbor. Tuning for these networks, the algorithm was originally developed to minimize memory by ensuring that at any one time only the particles handled by that processor would be in memory along with one little subdomain from adjacent processor. Using this fine-grained parallelism, each MPI process would advance through subdomains in lockstep, buffering only one off-CPU cell using MPI Send() and MPI Receive() shown in the pseudo code in Figure 8.

The algorithm progresses by marching through the subdomains and calculating the interactions between pairs in the subdomain and those immediately adjacent while leveraging the synchronous send and receive to communicate as needed. As the computation is made faster, the overhead from the communication latency begins to dominate. In the last five to ten years, memory has become more available allowing MD applications to have memory to spare that can then be used to buffer the entire set of boundary cells. These are known as *ghost cells* or a *halo exchange* when you prefetch all neighboring cells ahead of time. While this method can lead to some redundant calculations as the boundary pairs will be calculated once on each processor, it is worth the tradeoff as the computation/communication ratio

```
for each subdomain in i:
     compute self-interactions (i,i)
     for each neighboring subdomain j in half-path:
                if half-path crosses processor boundary:
                        MPI_send_and_receive()
                compute interactions(i,j) = (j,i)
     end for
end for
```

Figure 8: Pseudo code showing the original algorithm for the force calculation in SPaSM.

has shifted. This method is shown in Figure 9.

```
get ghost cells from neighboring processors:
        MPI_send_and_receive()
for each subdomain i:
        compute self-interactions (i,i)
        for each neighbor j in the full path:
                compute interactions (i,j)
        end for
end for
```

Figure 9: Pseudo code showing the halo exchange algorithm for the force calculation in SPaSM.

In this initial approach, focus was placed on accelerating the most computationally intense piece of the code. Ninety-five percent of time was spent computing forces, so the effort was put on accelerating the force calculation on the Cell processor. In this model, the particle positions are acquired, communicated down to the Cell processor at which point the forces are calculated and communicated back up. The time steps are then integrated before the system checkpoints and continues on to the next time step. The SPEs compute the forces and then sit idle as the Opterons update the positions/velocities of the atoms and vice-versa. The resulting performance was only 2.5 times faster than the original code on base Opterons. To optimally use a hybrid system, the accelerator needs to be kept as active as possible. If the accelerator is left idle, performance is lost. This meant that the trading between between the Cell and Opteron processors was damaging the performance.

This led to a *Cell-centric* redesign. One of the first steps was to adjust the data layout to optimize for the computation on the Cell processor versus the communication. While an array of atoms is optimal for communication, a structure of arrays allows streaming and vectorization on the Cell processor. This notion of

data layout from the cell-centric viewpoint epitomizes the goals of the redesign. Efforts were made to put as much work as possible on the Cell processor and to hide the data transfer time with work that could be done on data that was already local. By overlapping local data computation with the transfer, the Opterons are left idle more often. This idle time could be leveraged to enable more *in situ* visualization and checkpointing that took place during the time for the computation on the Cell processor. The Opteron owned all off-node communication while the Cell owned all compute-intensive parts of the application and ran with minimal idle time. These changes resulted in a 10x speedup achieving 369 Tflop/s which was 28% of peak [20].

5 Discussion

Single-scale computational materials science codes have been useful not only for gaining scientific insight, but also as testbeds for exploring new approaches for tackling evolving computational challenges. These including massive (nearly million-way) concurrency, an increased need for fault and power management, and data bottlenecks. It is no longer enough to simply port existing code to the next generation of systems. The current technology revolution is a tremendous opportunity to fundamentally rethink our applications and algorithms. Scale-bridging methods are crucial from both the application and computer science perspectives, and map well to the increasingly heterogeneous and hierarchical nature of new computer architectures. Preparations for the exascale (10^18 operations/second) era are underway by initiating an early and extensive collaboration between domain scientists, computer scientists, and hardware manufacturers i.e., computational co-design in which the applications, algorithms, and architectures are developed concurrently.

The goal is to introduce more detailed physics into computational material science applications in a way that escapes the traditional synchronous SPMD paradigm and exploits the exascale hardware. Sub-scale models could be used to drive forward the macro-scale models. In this case, coarse-scale simulations dynamically spawn tightly coupled and self-consistent fine-scale simulations as needed. One advantage is that this approach has relatively independent work units. For example, if in a set of cells, each needs a response to be calculated that could be a quantity from a Molecular Dynamics or a phase-field calculation, the needed model could be spawned off to be computed in a contained and independent way. This method is heterogenous with different length scales allowing multiple instances of different single-scale simulations, thus addressing the concurrency challenge by having 1000 million-way tasks instead of 1 billlion-way task. Current research has already achieved million-way parallelization demonstrating that coupling in this manner is

feasible today.

In this paper, we strived to motivate the need for a shift to a co-design paradigm in which the algorithms, applications, and architectures are taken into account simultaneously. Next generation multiscale materials science applications must take into account the underlying architectures of the systems being used in order to fully exploit their potential. By leveraging architectural information and concurrent coupling between scaling, they can begin to address outstanding questions in the field.

6 Glossary

CGMD–Coarse Grained Molecular Dynamics
ddcMD–domain decomposition Molecular Dynamics code developed at Lawrence Livermore National Laboratory
DFT–Density Funcitonal Theory
EAM–Embedded Atom Model
FE– Finite Element Method
FPMD–First Principles Molecular Dynamics
HMM–Heterogenous Multiscale Method
MAAD–Macroscopic, atomistic, *ab initio* dynamics
MD–Molecular Dynamics
MGPT–Model-generalized pseudo-potential theory
ParaDis–Parallel Dislocation Simulator, large-scale dislocation dynamics simulation code developed at Lawrence Livermore National Laboratory
Qbox–FPMD application developed at Lawrence Livermore National Laboratory
QC–Quasi continuum method
SPaSM–Scalable Parallel Short-range Molecular Dynamics, classical molecular dynamics code developed at Los Alamos National Laboratory
SPE–synergistic processing unit, component of the Cell processor
SPMD– Single Process Multiple Data

References

[1] NR Barton, JV Bernier, R. Becker, A. Arsenlis, R. Cavallo, J. Marian, M. Rhee, H.S. Park, BA Remington, and RT Olson. A multiscale strength model for extreme loading conditions. *Journal of Applied Physics*, 109:073501, 2011.

(abstract) We present a multiscale strength model in which strength depends on pressure, strain rate, temperature, and evolving dislocation density. Model construction employs an information passing paradigm to span from the atomistic level to the continuum level. Simulation methods in the overall hierarchy include density functional theory, molecular statics, molecular dynamics, dislocation dynamics, and continuum based approaches. Given the nature of the subcontinuum simulations upon which the strength model is based, the model is particularly appropriate to strain rates in excess of 104?s?1. Strength model parameters are obtained entirely from the hierarchy of simulation methods to obtain a full strength model in a range of loading conditions that so far has been inaccessible to direct measurement of material strength. Model predictions compare favorably with relevant high energy density physics (HEDP) experiments that have bearing on material strength. The model is used to provide insight into HEDP experimental observations and to make predictions of what might be observable using dynamic x-ray diffraction based experimental methods.

[2] George Almasi, Gyan Bhanot, Alan Gara, Manish Gupta, James Sexton, Bob Walkup, Vasily V. Bulatov, Andrew W. Cook, Bronis R. de Supinski, James N. Glosli, Jeffrey A. Greenough, Francois Gygi, Alison Kubota, Steve Louis, Thomas E. Spelce, Frederick H. Streitz, Peter L. Williams, Robert K. Yates, Charles Archer, Jose Moreira, and Charles Rendleman. Scaling physics and material science applications on a massively parallel blue gene/l system. In *Proceedings of the 19th annual international conference on Supercomputing*, ICS '05, pages 246–252, New York, NY, USA, 2005. ACM.

A great paper discussing early experiences with several physics and material science applications on the IBM Blue Gene/L supercomputer.

[3] F. Gygi, E.W. Draeger, M. Schulz, B.R. De Supinski, J.A. Gunnels, V. Austel, J.C. Sexton, F. Franchetti, S. Kral, C.W. Ueberhuber, et al. Large-scale electronic structure calculations of high-z metals on the bluegene/l platform. In *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, pages 45–es. ACM, 2006.

(abstract) First-principles simulations of high-Z metallic systems using the Qbox code on the BlueGene/L supercomputer demon-

strate unprecedented performance and scaling for a quantum simulation code. Specifically designed to take advantage of massively-parallel systems like BlueGene/L, Qbox demonstrates excellent parallel efficiency and peak performance. A sustained peak performance of 207.3 TFlop/s was measured on 65,536 nodes, corresponding to 56.5% of the theoretical full machine peak using all 128k CPUs.

[4] V.V. Bulatov, L.L. Hsiung, M. Tang, A. Arsenlis, M.C. Bartelt, W. Cai, J.N. Florando, M. Hiratani, M. Rhee, G. Hommes, et al. Dislocation multi-junctions and strain hardening. *Nature*, 440(7088):1174–1178, 2006.

(abstract) At the microscopic scale, the strength of a crystal derives from the motion, multiplication and interaction of distinctive line defects called dislocations. First proposed theoretically in 1934 to explain low magnitudes of crystal strength observed experimentally, the existence of dislocations was confirmed two decades later. Much of the research in dislocation physics has since focused on dislocation interactions and their role in strain hardening, a common phenomenon in which continued deformation increases a crystal's strength. The existing theory relates strain hardening to pair-wise dislocation reactions in which two intersecting dislocations form junctions that tie the dislocations together. Here we report that interactions among three dislocations result in the formation of unusual elements of dislocation network topology, termed multi-junctions. We first predict the existence of multi-junctions using dislocation dynamics and atomistic simulations and then confirm their existence by transmission electron microscopy experiments in single-crystal molybdenum. In large-scale dislocation dynamics simulations, multi-junctions present very strong, nearly indestructible, obstacles to dislocation motion and furnish new sources for dislocation multiplication, thereby playing an essential role in the evolution of dislocation microstructure and strength of deforming crystals. Simulation analyses conclude that multi-junctions are responsible for the strong orientation dependence of strain hardening in body-centred cubic crystals.

[5] D.E. Shaw, R.O. Dror, J.K. Salmon, JP Grossman, K.M. Mackenzie, J.A. Bank, C. Young, M.M. Deneroff, B. Batson, K.J. Bowers, et al. Millisecond-scale molecular dynamics simulations on anton. In *Proceedings of the Con-*

ference on High Performance Computing Networking, Storage and Analysis, page 65. ACM, 2009.

SC09 Gordon Bell Winner and Best Paper Winner discussing results for molecular dynamics (MD) simulations of biomolecular systems on Anton, a recently completed special-purpose supercomputer. A strong example of the co-design method.

[6] JN Glosli, DF Richards, KJ Caspersen, RE Rudd, JA Gunnels, and FH Streitz. Extending stability beyond cpu millennium: a micron-scale atomistic simulation of kelvin-helmholtz instability. In *Proceedings of the 2007 ACM/IEEE Conference on Supercomputing*, page 58. ACM, 2007.

Gordon Bell Winner. (abstract) We report the computational advances that have enabled the first micron-scale simulation of a Kelvin-Helmholtz (KH) instability using molecular dynamics (MD). The advances are in three key areas for massively parallel computation such as on BlueGene/L (BG/L): fault tolerance, application kernel optimization, and highly efficient parallel I/O. In particular, we have developed novel capabilities for handling hardware parity errors and improving the speed of interatomic force calculations, while achieving near optimal I/O speeds on BG/L, allowing us to achieve excellent scalability and improve overall application performance. As a result we have successfully conducted a 2-billion atom KH simulation amounting to 2.8 CPU-millennia of run time, including a single, continuous simulation run in excess of 1.5 CPU-millennia. We have also conducted 9-billion and 62.5-billion atom KH simulations. The current optimized ddcMD code is benchmarked at 115.1 TFlop/s in our scaling study and 103.9 TFlop/s in a sustained science run, with additional improvements ongoing. These improvements enabled us to run the first MD simulations of micron-scale systems developing the KH instability.

[7] F.H. Streitz, J.N. Glosli, M.V. Patel, B. Chan, R.K. Yates, B.R. de Supinski, J. Sexton, and J.A. Gunnels. 100+ tflop solidification simulations on bluegene/l. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, 2008.

(abstract) We investigate solidication in tantalum and uranium systems ranging in size from 64,000 to 524,288,000 atoms on the IBM BlueGene/L computer at LLNL. Using the newly developed ddcMD code, we achieve performance rates as high as 103 TFlops,

with a performance of 101.7 TFlop sustained over a 7 hour run on 131,072 cpus. We demonstrate superb strong and weak scaling. Our calculations are signicant as they represent the rst atomic-scale model of metal solidication to proceed, without nite size e?ects, from spontaneous nucleation and growth of solid out of the liquid, through the coalescence phase, and into the onset of coarsening. Thus, our simulations represent the rst step towards an atomistic model of nucleation and growth that can directly link atomistic to mesoscopic length scales.

[8] DF Richards, JN Glosli, B. Chan, MR Dorr, EW Draeger, J.L. Fattebert, WD Krauss, T. Spelce, FH Streitz, MP Surh, et al. Beyond homogeneous decomposition: scaling long-range forces on massively parallel systems. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, page 60. ACM, 2009.

Gordon Bell Finalist. (abstract) With supercomputers anticipated to expand from thousands to millions of cores, one of the challenges facing scientists is how to effectively utilize this ever-increasing number. We report here an approach that creates a heterogeneous decomposition by partitioning effort according to the scaling properties of the component algorithms. We demonstrate our strategy by developing a capability to model hot dense plasma. We have performed benchmark calculations ranging from millions to billions of charged particles, including a 2.8 billion particle simulation that achieved 259.9 TFlop/s (26% of peak performance) on the 294,912 cpu JUGENE computer at the Jlich Supercomputing Centre in Germany. With this unprecedented simulation capability we have begun an investigation of plasma fusion physics under conditions where both theory and experiment are lacking–in the strongly-coupled regime as the plasma begins to burn. Our strategy is applicable to other problems involving long-range forces (i.e., biological or astrophysical simulations). We believe that the flexible heterogeneous decomposition approach demonstrated here will allow many problems to scale across current and next-generation machines.

[9] RE Rudd, TC Germann, BA Remington, and JS Wark. Metal deformation and phase transitions at extremely high strain rates. *MRS bulletin*, 35(12):999–1006, 2010.

(abstract) The powerful lasers being constructed for inertially confined fusion generate enormous pressures extremely rapidly. These extraordinary machines both motivate the need and provide the means to study materials under extreme pressures and loading rates. In this frontier of materials science, an experiment may last for just 10s of nanoseconds. Processes familiar at ambient conditions, such as phase transformations and plastic flow, operate far from equilibrium and show significant kinetic effects. Here we describe recent developments in the science of metal deformation and phase transitions at extreme pressures and strain rates. Ramp loading techniques enable the study of solids at high pressures (100s of GPa) at moderate temperatures. Advanced diagnostics, such as in situ x-ray scattering, allow time-resolved material characterization in the short-lived high-pressure state, including crystal structure (phase), elastic compression, the size of microstructural features, and defect densities. Computer simulation, especially molecular dynamics, provides insight into the mechanisms of deformation and phase change.

[10] T.C. Germann, K. Kadau, and P.S. Lomdahl. 25 tflop/s multibillion-atom molecular dynamics simulations and visualization/analysis on bluegene/l. In *In Proceedings of IEEE/ACM Supercomputing 05*. Citeseer, 2005.

(abstract) We demonstrate the excellent performance and scalability of a classical molecular dynamics code, SPaSM, on the IBM BlueGene/L supercomputer at LLNL. Simulations involving up to 160 billion atoms (micron-size cubic samples) on 65,536 processors are reported, consistently achieving 24.425.5 Tflop/s for the commonly used Lennard-Jones 6-12 pairwise interaction potential. Two extended production simulations (one lasting 8 hours and the other 13 hours wall-clock time) of the shock compression and release of porous copper using a more realistic many-body potential are also reported, demonstrating the capability for sustained runs including on-the-fly parallel analysis and visualization of such massive data sets. This opens up the exciting new possibility of using atomistic simulations at micron length scales to directly bridge to mesoscale and continuum-level models.

[11] P.S. Lomdahl and D.M. Beazley. Molecular dynamics on the connection machine. *Los Alamos Science*, 2:44–57, 1994.

A strong overview of work exploiting a CM-5 machine for molecular dynamics simulations.

[12] E. Lu. G, Kaxiras. An overview of multiscale simulations of materials. *Handbook of Theoretical and Computational Nanotechnology*, 2005.

A useful summary article of the state of the art of multiscale modeling in material science. They discuss differing methods, classify them into spatial and temporal regimes, and analyze the strengths and weaknesses.

[13] E.B Tadmor and R.E. Miller. Quasicontinuum method: The original source for information, publications, and downloads. `http://www.qcmethod.org`.

A website providing more background on the Quasicontinuum method.

[14] J.Q. Broughton, F.F. Abraham, N. Bernstein, and E. Kaxiras. Concurrent coupling of length scales: methodology and application. *Physical Review B*, 60(4):2391, 1999.

(abstract) A strategic objective of computational materials physics is the accurate description of specific materials on length scales approaching the meso and macroscopic. We report on progress towards this goal by describing a seamless coupling of continuum to statistical to quantum mechanics, involving an algorithm, implemented on a parallel computer, for handshaking between finite elements, molecular dynamics, and semiempirical tight binding. We illustrate and validate the methodology using the example of crack propagation in silicon.

[15] R.E. Rudd and J.Q. Broughton. Coarse-grained molecular dynamics: Nonlinear finite elements and finite temperature. *Physical Review B*, 72(14):144104, 2005.

A strong overview of Coarse-grained molecular dynamics. In this paper, they discuss both the formulation and application of CGMD.

[16] X. Li and E. Weinan. Multiscale modeling of the dynamics of solids at finite temperature. *Journal of the Mechanics and Physics of Solids*, 53(7):1650–1685, 2005.

(abstract) We develop a general multiscale method for coupling atomistic and continuum simulations using the framework of the heterogeneous multiscale method (HMM). Both the atomistic and the continuum models are formulated in the form of conservation laws of mass, momentum and energy. A macroscale solver, here the finite volume scheme, is used everywhere on a macrogrid; whenever necessary the macroscale fluxes are computed using the microscale model, which is in turn constrained by the local macrostate of the system, e.g. the deformation gradient tensor, the mean velocity and the local temperature. We discuss how these constraints can be imposed in the form of boundary conditions. When isolated defects are present, we develop an additional strategy for defect tracking. This method naturally decouples the atomistic time scales from the continuum time scale. Applications to shock propagation, thermal expansion, phase boundary and twin boundary dynamics are presented.

[17] R.E. Miller and EB Tadmor. A unified framework and performance benchmark of fourteen multiscale atomistic/continuum coupling methods. *Modelling and Simulation in Materials Science and Engineering*, 17:053001, 2009.

(abstract) A partitioned-domain multiscale method is a computational framework in which certain key regions are modeled atomistically while most of the domain is treated with an approximate continuum model (such as finite elements). The goal of such methods is to be able to reproduce the results of a fully atomistic simulation at a reduced computational cost. In recent years, a large number of partitioned-domain methods have been proposed. Theoretically, these methods appear very different to each other making comparison difficult. Surprisingly, it turns out that at the implementation level these methods are in fact very similar. In this paper, we present a unified framework in which fourteen leading multiscale methods can be represented as special cases. We use this common framework as a platform to test the accuracy and efficiency of the fourteen methods on a test problem; the structure and motion of a Lomer dislocation dipole in face-centered cubic aluminum. This problem was carefully selected to be sufficiently simple to be quick to simulate and straightforward to analyze, but not so simple to unwittingly hide differences between methods.

The analysis enables us to identify generic features in multiscale methods that correlate with either high or low accuracy and either fast or slow performance.

[18] C. Domain, CS Becquart, and L. Malerba. Simulation of radiation damage in fe alloys: an object kinetic monte carlo approach. *ournal of Nuclear Materials*, 335(1):121–145, 2004.

(abstract) The reactor pressure vessel (RPV) steels used in current nuclear power plants embrittle as a consequence of the continuous irradiation with neutrons. Among other radiation effects, the experimentally observed formation of copper-rich defects is accepted to be one of the main causes of embrittlement. Therefore, an accurate description of the nucleation and growth under irradiation of these and other defects is fundamental for the prediction of the mechanical degradation that these materials undergo during operation, with a view to guarantee a safer plant life management. In this work we describe in detail the object kinetic Monte Carlo (OKMC) method that we developed, showing that it is well suited to investigate the evolution of radiation damage in simple Fe alloys (Fe, Fe-Cu) under irradiation conditions (temperature, dose and dose-rate) typical of experiments with different impinging particles and also operating conditions. The still open issue concerning the method is the determination of the mechanisms and parameters that should be introduced in the model in order to correctly reproduce the experimentally observed trends. The state-of-the-art, based on the input from atomistic simulation techniques, such as ab initio calculations, molecular dynamics (MD) and atomic kinetic Monte Carlo, is critically revised in detail and a sensitivity study on the effects of the choice of the reaction radii and the description of defect mobility is conducted. A few preliminary, but promising, results of favorable comparison with experimental observations are shown and possible further refinements of the model are briefly discussed.

[19] A.F. Voter, F. Montalenti, and T.C. Germann. Extending the time scale in atomistic simulation of materials. *Annual Review of Materials Research*, 32(1):321–346, 2002.

(abstract) Obtaining a good atomistic description of diffusion dynamics in materials has been a daunting task owing to the time-

scale limitations of the molecular dynamics method. We discuss promising new methods, derived from transition state theory, for accelerating molecular dynamics simulations of these infrequent-event processes. These methods, hyperdynamics, parallel replica dynamics, temperature-accelerated dynamics, and on-the-fly kinetic Monte Carlo, can reach simulation times several orders of magnitude longer than direct molecular dynamics while retaining full atomistic detail. Most applications so far have involved surface diffusion and growth, but it is clear that these methods can address a wide range of materials problems.

[20] T. C. Germann, K. Kadau, and S. Swaminarayan. 369 tflop-s molecular dynamics simulations on the petaflop hybrid supercomputer roadrunner'. *Concurr. Comput. : Pract. Exper.*, 21:2143–2159, December 2009.

(abstract) We describe the implementation of a short-range parallel molecular dynamics (MD) code, SPaSM, on the heterogeneous general-purpose Roadrunner supercomputer. Each Roadrunner TriBlade compute node consists of two AMD Opteron dual-core microprocessors and four IBM PowerXCell 8i enhanced Cell microprocessors (each consisting of one PPU and eight SPU cores), so that there are four MPI ranks per node, each with one Opteron and one Cell. We will briefly describe the Roadrunner architecture and some of the initial hybrid programming approaches that have been taken, focusing on the SPaSM application as a case study. An initial evolutionary port, in which the existing legacy code runs with minor modifications on the Opterons and the Cells are only used to compute interatomic forces, achieves roughly a 2? speedup over the unaccelerated code. On the other hand, our revolutionary implementation adopts a Cell-centric view, with data structures optimized for, and living on, the Cells. The Opterons are mainly used to direct inter-rank communication and perform I-O-heavy periodic analysis, visualization, and checkpointing tasks. The performance measured for our initial implementation of a standard LennardJones pair potential benchmark reached a peak of 369 Tflop-s double-precision floating-point performance on the full Roadrunner system (27.7% of peak), nearly 10? faster than the unaccelerated (Opteron-only) version.