The Dissertation Committee for Samuel T. Chill
certifies that this is the approved version of the following dissertation:

Methods, Software, and Benchmarks for Modeling Long Timescale Dynamics in Solid-state Atomic Systems

Committee:

Graeme Henkelman, Supervisor

Ron Elber

Venkat Ganesan

Gyeong S. Hwang

Dimitri E. Makarov

Methods, Software, and Benchmarks for Modeling Long Timescale Dynamics in Solid-state Atomic Systems

by

Samuel T. Chill, B.S.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2014

Dedicated to my loving parents David and Susan.

Acknowledgments

I am deeply grateful for the opportunities that my advisor, Graeme Henkelman, has provided for me. Due to his support, I have been able to travel around the world and collaborate with many great scientists. I also wish to thank my wife Tara, my parents David and Susan, and my grandfather Leonard for their support of my education. Lastly I must thank my lab mates Rye Terrell, Matthew Welborn, Liang Zhang, Penghao Xiao, and Marco Howard for many fruitful and entertaining discussions.

Methods, Software, and Benchmarks for Modeling Long Timescale Dynamics in Solid-state Atomic Systems

Samuel T. Chill, Ph.D.
The University of Texas at Austin, 2014

Supervisor: Graeme Henkelman

The timescale of chemical reactions in solid-state systems greatly exceeds what may be modeled by direct integration of Newton's equation of motion. This limitation spawned the development of many different methods such as (adaptive) kinetic Monte Carlo (A)KMC, (harmonic) transition state theory (H)TST, parallel replica dynamics (PRD), hyperdynamics (HD), and temperature accelerated dynamics. The focus of this thesis was to (1) implement many of these methods in a single open-source software package (2) develop standard benchmarks to compare their accuracy and computational cost and (3) develop new long timescale methods.

The lack of a open-source package that implements long timescale methods makes it difficult to directly evaluate the quality of different approaches. It also impedes the development of new techniques. Due to these concerns we developed Eon, a program that implements several long timescale methods including PRD, HD, and AKMC as well as global optimization algorithms basin hopping, and minima hopping.

Standard benchmarks to evaluate the performance of local geometry optimization; global optimization; and single-ended and double-ended saddle

point searches were created. Using EON and several other well known programs, the accuracy and performance of different algorithms were compared. Important to this work is a website where anyone ma download the code to repeat any of the numerical experiments.

A new method for long timescale simulations is also introduced: molecular dynamics saddle search adaptive kinetic Monte Carlo (AKMC-MDSS). AKMC-MDSS improves upon AKMC by using short high-temperature MD trajectories to locate the important low-temperature reaction mechanisms of interest. Most importantly, the use of MD enables the development of a proper stopping criterion for the AKMC simulation that ensures that the relevant reaction mechanisms at the low-temperature have been found.

Important to the simulation of any material is knowledge of the experimental structure. Extended x-ray absorption fine structure (EXAFS) is a technique often used to determine local atomic structure. We propose a technique to quantitatively measure the accuracy of the commonly used fitting models. This technique reveals that the fitting models interpreted nanoparticles as being significantly more ordered and of much shorter bond length than they really are.

Table of Contents

Chapter 1

Introduction

This thesis is composed of work that I completed with my advisor, Graeme Henkelman during my graduate studies at the University of Texas at Austin. The primary focus is on the development of computational methods for the atomistic simulation of long timescale dynamics in solid-state systems. Each chapter of the dissertation is composed of a paper written with Graeme. Chapter 2 was written with the the help of my lab mates Matthew Welborn and Rye Terrell as well as our collaborators Jean-Claude Berthet, Andreas Pedersen and Hannes Jónsson from the University of Iceland. Chapter 4 was written in collaboration with Jacob Stevenson, Victor Ruehle, Cheng Shang and David Wales of the University of Cambridge as well as my lab mate Penghao Xiao. Chapter 5 was written with our experimental collaborators David Yancey, Rachel Anderson and Richard Crooks from the University of Texas as well as Anatoly Frenkel of Yeshiva University. The introduction will provide a brief background in order to motivate the following chapters.

1.1 Rare Event Dynamics

Chemical reactions in the solid state are considered rare events. This is because the shortest natural timescale of atomic motion – a vibrational period – is on the order of a tens to hundreds of femtoseconds, while interesting chemical reactions can can take anywhere from nanoseconds (vacancy diffusion), to

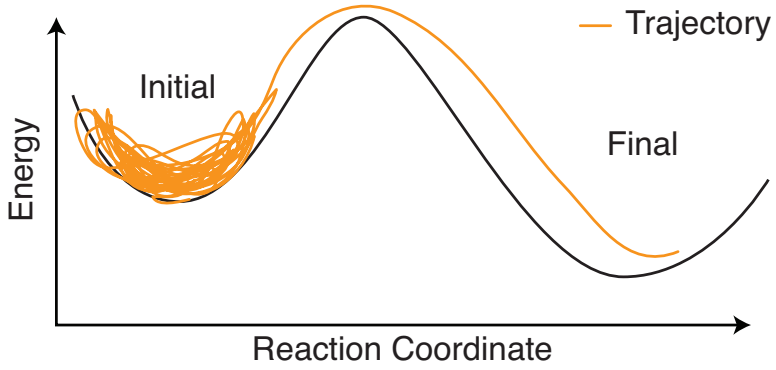

Figure 1.1: An artistic rendering of the separation of timescales between atomic vibrations and reactive events. The orange line represents a trajectory that starts in the "initial" energy basin and remains there for many vibrational periods before it escapes to the "final" energy basin.

milliseconds (surface catalysis), to years (material failure) to occur. A cartoon of rare event dynamics is shown in Figure 1.1.

The most straightforward method to simulate reactions is molecular dynamics (MD), where the classical equations of motion are integrated forward in time. In order for the simulation to be numerically stable, an integration time step of at most several femtoseconds must be used. Due to the enormous gap in the timescale of atomic vibrations and chemical reactions this means that an inordinate number of steps are typically required to witness a single reaction.

To give an example of just how long this calculation may take, a single time step (*i.e.* simulating 1×10^{-15} s) might take around 1 millisecond of CPU time using an empirical force-field with hundreds of atoms. This means that simulating 1 millisecond of time would take roughly 32 years!

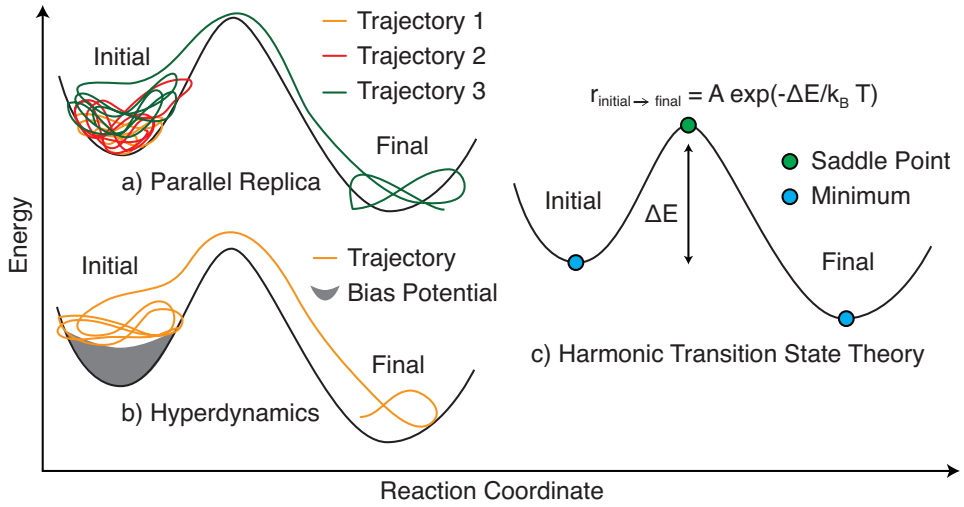

Figure 1.2: Three different methods for calculating the long timescale behavior in atomic systems are shown schematically.

1.2 Long Timescale Dynamics Methods

Many methods have been developed to reduce the computational work needed to model rare event dynamics. There exists a unifying theme amongst these techniques. Instead of modeling the intricate details of a single long atomic trajectory, the dynamics are simplified into model that depicts only the state-to-state time evolution. In this context, states are usually defined as meta-stable (long lived) regions of configuration space. A high-level overview of a few notable algorithms will be given in this section. Each of these methods are implemented in the Eon software presented in Chapter 2.

Parallel replica dynamics (PRD) is a method for parallelizing state-to-state molecular dynamics trajectories.[1] It does not reduce the amount of computation effort required, however, by running a set of independent replicas of the simulation on separate processors the total amount of time can be reduced linearly with the number of processors used. In order to make each

replica independent, a short MD trajectory is run on each processes to initialize it from a unique region of phase space. Figure 1.2(a) visualizes three separate trajectories run in parallel, one of which escapes. The total escape time can then be obtained by summing together the time each replica simulated.

Hyperdynamics (HD) reduces the computational effort to calculate the escape times in the state-to-state simulation by applying a bias potential that raises the energy of the energy basin, while leaving the energy of the transition state unaffected.[2] By raising the energy of the basin relative to the transition state, the effective energy barrier is lowered and the mean escape time is decreased. The true escape time on the unbiased potential energy surface can be recovered by multiplying the integration time step times an exponential correction factor at each step. The main challenge is the construction of a functional form for a bias potential that sufficiently increases the energy inside the energy basin but does not modify the transition state surface. An example of a HD trajectory is shown in Figure 1.2(b).

Transition state theory (TST) attempts to directly calculate a rate constant by identifying bottlenecks on the potential energy surface that the system must progress through for the reaction to occur.[3] These bottlenecks are referred to as transition states. In solids the bottleneck regions are small and are localized around saddle points on the potential energy surface (PES). The simplest form of TST is harmonic TST (HTST), where the PES around the saddle point is treated as a harmonic function. In HTST, the reaction rate can be expressed in terms that only involve the minimum and the saddle point; no MD trajectories are needed to calculate the rate. These scheme is shown graphically in Figure 1.2(c).

In Chapter 3, we present a new long timescale dynamics scheme

named molecular dynamics saddle search adaptive kinetic Monte Carlo (MDSS-AKMC), which is based upon HTST. MDSS-AKMC uses high temperature MD trajectories to locate saddle points. These saddle points are used with HTST to calculate reaction rates. The primary advancement of this technique is a novel stopping criteria to define when a sufficient number of saddle searches have been performed in our to guarantee that the important low energy (high rate) reaction mechanisms have been discovered.

In Chapter 4, a set of benchmarks is defined that measure the accuracy and computational cost of these algorithms. The Eon software package as well as other commonly used programs are compared and the relative merits of different algorithms are explored.

1.3 Local Structure Determination Using EXAFS

Key to the long timescale simulation of any material is an initial three dimensional atomic structure. One popular experimental technique that reveals local average structural information is extended x-ray absorption fine structure (EXAFS) spectroscopy. EXAFS spectra are collected by shining high energy x-rays upon a sample. These x-rays are high enough energy to excite core electrons into photo-electrons. The wave-like photo-electron then backscatters off the neighboring atoms. The interface pattern from the neighboring atoms causes the EXAFS signal to contain oscillations. These oscillations encode local structural information such as average coordination number, bond length, and disorder.

A non-linear fitting proceedure is used in order to extract structural information from EXAFS spectra. Key to this fitting process is the choice of a proper functional form to represent both the x-ray scattering and the

bond length distribution in the sample. The functional form of the EXAFS equation contains many terms, but can be roughly described as the sum of exponentially decaying sinusoids, whose amplitudes, frequencies, and decay rates are correlated with the atomic coordination numbers, bond lengths, and structural disorder respectively.

In Chapter 5, we introduce a new method to measure the accuracy of EXAFS fitting models and apply it to the study of Au_{147} nanoparticles. In the study, we show that the commonly used fitting models interpret nanoparticle structures as being smaller and more ordered than they really are. This is an important result when comparing theoretical calculations to the experimental EXAFS fits. We show the origin of these error in the fitting model and make suggestions on how use density functional theory calculations to improve the understanding of EXAFS spectra in highly disordered systems.

Chapter 2

EON: Software for Long Time Simulations of Atomic Scale Systems [1]

2.1 Abstract

The EON software is designed for simulations of the state-to-state evolution of atomic scale systems over time scales greatly exceeding that of direct classical dynamics. States are defined as collections of atomic configurations from which a minimization of the potential energy gives the same inherent structure. The time evolution is assumed to be governed by rare events, where transitions between states are uncorrelated and infrequent compared with the time scale of atomic vibrations. Several methods for calculating the state-to-state evolution have been implemented in EON including parallel replica dynamics, hyperdynamics, and adaptive kinetic Monte Carlo. Global optimization methods, including simulated annealing, basin hopping, and minima hopping are also implemented. The software has a client/server architecture where the computationally intensive evaluations of the interatomic interactions are calculated on the client-side and the state-to-state evolution is managed by the server. The client supports optimization for different computer architec-

[1] S. T. Chill, M. Welborn, R. Terrell, L. Zhang, J.-C. Berthet, A. Pedersen, H. Jónsson, G. Henkelman, *Model. Simul. Mater. Sci. Eng.* 22, 055002 (2014). M. Welborn, R. Terrell, L. Zhang, A. Pedersen, and J.-C. Berthet contributed to the software used in this chapter. A. Pedersen also ran the CuZr glass simulation. H. Jónsson and G. Henkelman supervised the work in this chapter.

tures to maximize computational efficiency. The server is written in Python so that developers have access to the high-level functionality without delving into the computationally intensive components. Communication between the server and clients is abstracted so that calculations can be deployed on a single machine, clusters using a queuing system, large parallel computers using a message passing interface, or within a distributed computing environment. A generic interface to the evaluation of the interatomic interactions is defined so that empirical potentials, such as in LAMMPS, and density functional theory as implemented in VASP and GPAW can be used interchangeably. Examples are given to demonstrate the range of systems that can be modeled, including surface diffusion and island ripening of adsorbed atoms on metal surfaces, molecular diffusion on the surface of ice, and global structural optimization of nanoparticles.

2.2 Introduction

Long time scale simulations of atomic systems pose a particular challenge in computational chemistry. A straightforward integration of the equations of motion typically requires too many integration steps to reach the time scale of interesting events. Since atomic vibrations occur on the femtosecond time scale, roughly 10^{15} time steps are required to directly model one second of molecular dynamics.

EON is a software package for performing accelerated dynamics simulations of diffusion in solids and reactions at surfaces. There are a number of established methods for calculating the long time scale properties of chemical and material systems and a primary goal of EON is to provide these methods in a single program so that they can be directly compared, the strengths of

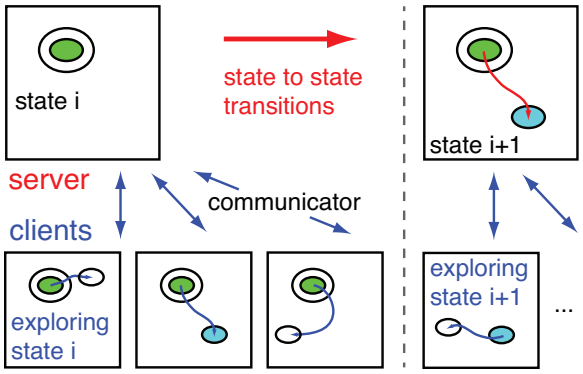

Figure 2.1: Computational framework of EON calculations.

different methods exploited, and where advantageous, combined into hybrid methods.

The methods implemented in EON share an underlying concept. They each assume that the system has stable states that correspond to minima on the potential energy surface (PES). The term 'stable' means that the average escape time from a state is significantly greater than the correlation time of a trajectory within the state. The transitions between states are then rare events in comparison to the vibrational relaxation time scale within any state.

The computational framework for the methods implemented in EON is illustrated in Figure 2.1. EON has a client-server architecture in which the server determines the state-to-state evolution of the system. Clients communicate with the server to receive tasks that involve exploring the configuration space around the current state. Results are reported back to the server so that the state-to-state evolution can proceed. For both receiving work and reporting results, communication is initiated by the client and the simulation data is transmitted in human readable text files.

The implementation of EON is illustrated in Figure 2.2. The client,

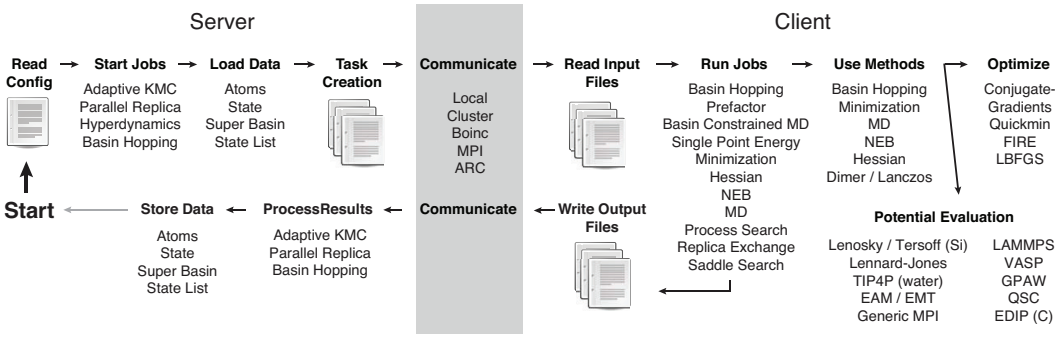

Figure 2.2: The Eon server interacts with Eon clients through communicators, providing them with independent tasks (e.g. saddle searches). The results are communicated back to the server where they are processed by higher-level algorithms (e.g. adaptive kinetic Monte Carlo). The various acronyms are defined in the text.

written in C++, performs computationally intensive tasks, while the server, written in Python, handles the bookkeeping associated with the higher-level algorithms. The client is used for any method that requires evaluation of the potential energy and force. Client-side methods include local geometry optimization, saddle point searches, molecular dynamics trajectories, Monte Carlo sampling, and normal mode analysis. The server implements the high-level methods using the tools provided by the client including parallel replica dynamics (PRD) [1], hyperdynamics [2] using the bond-boost form of the bias potential [4], adaptive kinetic Monte Carlo (AKMC) [5], basin hopping (BH) [6], and minima hopping [7].

The Eon architecture has several strengths. First and foremost, the client-server architecture allows for asynchronous cluster and distributed computing. Second, a single server can communicate with many different clients through a common interface, meaning that only part of the code has to be tailored to different platforms, either through compilation or linking to platform-

specific libraries. Finally, the server communicates with the client via text files so that a user can directly access the client's functionality without having to use the server.

2.3 Code Structure

2.3.1 Server

The EON server has several roles: managing the execution of client jobs, processing the results of client calculations, and transitioning between states of the system. Typically the server is run many times over the course of a calculation. Upon each invocation, the server processes the results of completed jobs, performs actions depending on the results that were received, and creates new jobs, in that order. EON will typically exit at this point. The one exception is when EON is run with the message passing interface (MPI) communicator, in which case the server will continue to run in a loop waiting for new results to process.

The EON server coordinates a set of independent tasks, which are executed in parallel by the clients. The primary mechanism of communication between client and server is via text files. The server creates the input files for each client calculation and processes the output files that the clients write. The results of the individual client calculations influence subsequent jobs; it is only through the server that the clients interact.

In a distributed computing environment, clients cannot be trusted to complete all of the work that they are given. In a heterogeneous computing environment, clients will typically leave and return to the network over the course of an EON simulation. Due to this constraint, server-client communication is stateless; no one client is relied upon to complete any given job. Instead,

enough work is created to keep all clients busy and the results are processed upon their completion. The server never waits for a particular client to finish a calculation.

The independence of the client jobs allows for flexibility in how the clients are executed. EON supports parallel calculations in a variety of contexts: on a single computer, on clusters using a queuing system, large parallel computers using MPI, and in a distributed computing environment. These contexts are explained more fully in Sec. 2.3.4.

The server also manages the state-to-state evolution in the system. For example, in the AKMC method the server uses the kinetic Monte Carlo (KMC) algorithm to move between states until an unexplored state is reached. At this point, client calculations are required to determine the mechanisms and rates of escape. In PRD the server waits for clients to run dynamics within a state and report transitions a new state. The server then updates the current state of the system and repeats the process.

2.3.2 Client

While the server is written in Python, the client is written in C++. An advantage of using a complied language for the client is that a self-contained executable is most easily used within a distributed computing platform [8]. A Python client would require the distribution of the Python runtime environment along with our code. Another reason for using C++ is that the client does most of the computational work, and it is not uncommon for Python to be 100 times slower than lower level languages.[9]

EON provides a number of reusable software components. For example, the numerical optimization routines operate on general objective functions.

This allows the same optimization code to be applied to a variety of problems, including local minimization, saddles searches, and nudged elastic band calculations. EON implements a number of first and second order optimization algorithms including quick-min (QM) [10], fast inertial relaxation engine (FIRE) [11], conjugate gradients (CG) [12], and limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) [13].

2.3.3 Potentials

EON makes use of a variety of different atomic interaction potentials, some of which are implemented within our code while others are accessed either by linking to a library or calling an external program. Specifically, EON works with the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [14], the Vienna Ab-initio Simulation Package (VASP) [15], and the Grid-based Projector-Augmented Wave (GPAW) [16] code.

The potentials that are included with EON are an embedded atom method (EAM) potential [17] with parameters for Al [18], quantum Sutton-Chen with parameters for FCC metals, effective medium theory from the ASAP package [19], pair potentials of the Lennard-Jones and Morse forms, Lenosky [20] and Tersoff [21] potentials for Si, the environment dependent interactive potential (EDIP) for carbon [22], and the TIP4P water model [23].

EON can use potentials from LAMMPS by linking EON to the LAMMPS library. The LAMMPS interface in EON issues commands to initialize the simulation by specifying the periodic boundary conditions and atom types. For each force call, EON updates the positions of the atoms in LAMMPS and calls the application programing interface (API) to update the energy and forces. Using the API, as opposed to calling the LAMMPS executable for each force

call, keeps the neighbor list in memory between force calls, and significantly improves the computational efficiency.

Finally, there is a MPI-based potential interface that communicates with an external MPI program to calculate the energy and forces. This potential type is implemented as a multiple program multiple data (MPMD) MPI job where the EON client is launched along with the external MPI program(s). Having a single executable for the server and clients is especially useful for running on supercomputers with *ab-initio* codes that do not have a library interface to be linked with EON. An example of such a program is VASP. We provide a small modification to VASP to work with EON so that VASP stays in memory between energy evaluations.[24] In our tests this approach is 60% faster than executing VASP for each force call, reusing the previous wavefunctions and charge density from disk.

2.3.4 Communicators

The client-server architecture gives EON flexibility in how the client program is executed. The details of how the client runs and communicates its results are abstracted. Decoupling the communication from the rest of the code allows for different parallelization schemes to be implemented with minimal effort. The different communicators are implemented as classes that handle how and where the client is executed. The interface to the class is simple, requiring only the data in the input files for the client. The specific communicator then ensures that the job is executed and returns the result, when completed, to the server. The class also provides methods to cancel jobs and retrieve the number of running jobs.

Our generic approach to running the computationally expensive part of

our program allows EON to run in the following modes.

Local: The simplest way to run EON is to run in serial or in parallel on a single computer. The server launches a user specified number of jobs in parallel and waits for them to complete before exiting.

Cluster: EON can directly submit jobs to a job queuing system such as Grid Engine, Torque, or the Portable Batch System (PBS). User supplied scripts enable the EON server to submit jobs, monitor running jobs, and delete jobs.

MPI: With the MPI communicator, the server and clients are bundled together as a single MPMD MPI executable. The server, instead of exiting after creating client jobs, polls the clients to determine when new jobs need to be created. When combined with the MPI potential interface, EON can run many independent density functional theory (DFT) calculations simultaneously.

BOINC: EON supports distributed computing using the Berkeley Open Infrastructure for Network Computing (BOINC) [8]. Running a distributed calculation requires a BOINC project server and an EON client that is linked with the BOINC library. EON is currently run as a BOINC project with over a thousand computers connected that has achieved a peak performance of four teraFLOPS.[25] It is possible to run many independent calculations using the same BOINC project with EON. The communicator uses a unique identifier to track which BOINC work units belong to which simulation. In this way, multiple users are able to run their own simulations within the same BOINC project.

A consideration when using BOINC for calculations is that the communication time between client and server should be shorter than the time to complete a client job. For calculations involving empirical potentials, a single client calculation can be faster than the time to process a work-unit through the distributed computing system. This condition causes the client computer to be idle while it waits for additional work. To solve this problem, we have implemented a mechanism for bundling a set of tasks together to be run as a single client job. Tuning the number of tasks in the bundle controls the BOINC work unit duration. Bundling a set of small jobs into work units also reduces the number of files which have to be processed by the server, an issue that becomes very important when thousands of machines are rapidly reporting their result to a single server.

ARC: An interface for the Advanced Resource Connector (ARC) middleware enables execution on resources connected to NORDUGRID [26]. The purpose of the interface is to enable to use of idle nodes on authorized resources. As only idle resources are to be requested, one initially registers how many idle nodes each cluster has, which is followed by a submission of the corresponding number of work units to the NORDUGRID queue on each of the clusters. The ARC middleware must be installed with the EON server to use this communicator.

2.4 Methods

2.4.1 Parallel replica dynamics and hyperdynamics

Parallel replica dynamics (PRD) is a method for parallelizing state-to-state molecular dynamics (MD) trajectories.[1] A set of M replicas of the simulation are initialized to run MD trajectories on separate processors from

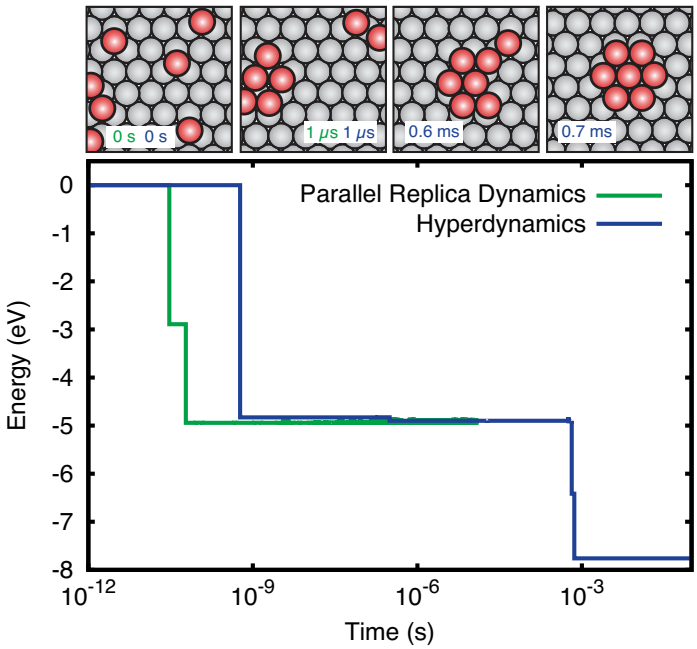

Figure 2.3: Snapshots from PRD and hyperdynamics simulations of seven Pt atoms randomly deposited on a (111) surface at 250 K. The final state, which is found using hyperdynamics on a time scale of minutes, shows the formation of a compact heptamer island. The atoms on the surface are shaded red.

the same initial state. Each replica is given its own unique initial random momenta and run with MD for a short dephasing time to ensure that it is uncorrelated from the other replicas in configuration space. After dephasing, the simulation clock is started and MD is performed until any replica has escaped from the initial state. All replicas then report the amount of time that has been simulated and the transition time is taken to be the cumulative simulation time from the replicas. Critical to this method is an algorithm to detect transitions between states. In EON we detect a transition by periodically minimizing the geometry and comparing to the initial minimized geometry.

The algorithm, as described, is not suitable for distributed computing

because it is not always possible to promptly report when a replica sees a transition, nor is it possible to query all the other replicas to determine how much simulation time had been accumulated when the transition occurred. To work with EON, all communication between replicas must be removed. This is accomplished in the distributed replica dynamics (DRD) implemented in EON by having each client run for a fixed length trajectory. The clients do not report back when they detect a reaction; instead, each replica does the same amount of work as all other clients, on average. Accordingly, when a client detects a transition it records the transition time and configuration and then continues to run dynamics for the full time length as if a transition had not occurred. It then returns the time of the transition and the new product state to the server. The server adds the transition time to the total simulation clock. Details of the validity and efficiency of the DRD algorithm will be presented elsewhere.[27] Since DRD is so similar in philosophy to PRD, we will refer to it hereafter simply as PRD.

As an example of a long time scale simulation modeled with PRD, we have chosen the formation of a compact Pt heptamer island on Pt(111) at 250 K using a Morse interatomic potential. Each PRD client was run for a 20 ps MD trajectory with a time step of 2 fs and a dephasing time of 1 ps. PRD was also combined with hyperdynamics to increase the accessible simulation time. We used the bond-boost form of bias potential [4] setting the magnitude of the total boost $\Delta V^{\max}$ to 0.5 eV, the stretch threshold q to 0.2, and the curvature parameter P_1 to 0.95. All other parameters were kept the same as the PRD calculation. The evolution of the system is shown in Figure 2.3. The initial coalescence of the adatoms occurs on a time scale of nanoseconds and the formation of the compact island occurs on a time scale of minutes, which

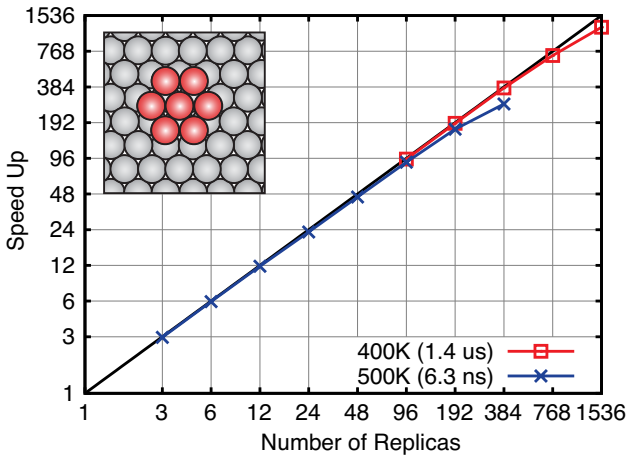

Figure 2.4: Timing data showing good linear speedup for PRD of a Pt heptamer island on Pt(111). Deviations from the linear scaling trend start when the number of replicas times the simulation time for each replica approaches the time scale of the transition.

can only be seen in the hyperdynamics simulation.

In the ideal case, PRD is able to reduce the wall clock time of a MD calculation linearly with respect to the number of processors used. To demonstrate that our code achieves this scaling, we measured the speedup of escaping from the compact Pt heptamer island state. Each PRD trajectory was run for 100 ps; reactive events were detected by minimizing every 10 ps. Figure 2.4 shows that the speed-up, in terms of wall clock time, required to find an escape time for the compact Pt heptamer island on Pt(111) increases linearly with the number of replicas. The slope of the speedup plot only deviates from unity when the total time simulated by the replicas in a reporting interval approaches the transition time. When there are M clients running 100 ps trajectories in parallel, this limit is reached for $M{\approx}100$ at 500 K and $M{\approx}1000$ at 400 K. In principle, the MD time simulated by each replica can be reduced, although here, 100 ps was chosen so that each calculation would take several

minutes, which is a suitable job length for distributed computing, as discussed in Sec. 2.3.4.

2.4.2 Adaptive kinetic Monte Carlo

KMC is a method used to model the state-to-state dynamics of chemical and material systems. An in-depth introduction and discussion of the history of KMC has been written by Voter.[28] KMC simulations are fundamentally different from MD in that they do not need to evaluate the potential energy or forces of the system, although in many cases, rates of reaction are derived from a PES. In KMC, a Markov chain of states is formed with transition probabilities proportional to the rates of reaction. The set of rates, known as a rate table, must be known *a priori* for all states that the system will encounter.

AKMC is a method to dynamically build a rate table during a KMC simulation.[5] For each new unique state that the system visits, searches are preformed to find low energy first-order saddle points on the PES. Typically these searches are carried out by minimum mode following algorithms where the minimum mode is estimated using the dimer method [29], Raleigh-Ritz minimization [30] as in the hybrid eigenvector following method [31], or the Lanczos method as in the activation relaxation technique nouveau [32]. Minimizations are carried out from the saddle point geometry to the two adjacent minimum energy configurations. Rates can be efficiently calculated for the forward and backward reactions using the harmonic approximation to transition state theory (HTST)

$$k_{\mathrm{HTST}} = \frac{\prod_i^{3N} \nu_i^{\mathrm{min}}}{\prod_i^{3N-1} \nu_i^{\ddagger}} \exp\left[-\left(E^{\ddagger} - E^{\mathrm{min}}\right)/k_{\mathrm{B}}T\right], \tag{2.1}$$

where N is the number of atoms, ν_i^{min} and $\nu_i^{\ddagger}$ are the positive (stable) normal

mode frequencies at the minimum and saddle, $E^{\min}$ and $E^{\ddagger}$ are the energies at the minimum and saddle point, k_{B} is Boltzmann's constant and T is the temperature.

Once the saddles of relevant energy have been located, the rate table for that state is considered complete. Details on how the confidence that the relevant saddles have been found is given in Ref. [33]. A KMC step is then taken to advance the simulation to the next state and to increment the simulation clock. If the next state is a previously visited state, then there already exists a rate table and no searches will need to be performed; otherwise saddle searches are needed to determine the new rate table.

An important feature of the AKMC implementation is systematic coarse graining of states to eliminate fast transitions. In Eon, simulations are coarse grained on-the-fly using the Monte Carlo with absorbing Markov chains (MCAMC) algorithm [34]. In this formalism, the mean first passage times and probabilities from one set of states to another are calculated exactly. The basic equations used in MCAMC will be reproduced here; a more detailed review can be found in Ref. [35]. An absorbing Markov chain may be written in the canonical form

$$\mathbf{M}_{(r+s)\times(r+s)} = \begin{pmatrix} \mathbf{T}_{s\times s} & \mathbf{R}_{s\times r} \\ \mathbf{0}_{r\times s} & \mathbf{I}_{r\times r} \end{pmatrix}, \tag{2.2}$$

where $\mathbf{T}$ is the matrix of probabilities to transition within the s transient states, and $\mathbf{R}$ is the matrix of probabilities to transition from the s transient states to the r absorbing states, and $\mathbf{I}$ is the identity matrix. The fundamental matrix is defined as

$$\mathbf{N} = \sum_{k=0}^{\infty} \mathbf{T}^k = (\mathbf{I} - \mathbf{T})^{-1}, \tag{2.3}$$

where the $\mathbf{N}_{ij}$ is the average number of times state j is visited before absorption, if the chain starts in state i. Using the fundamental matrix, it is possible

to calculate both the expected time until absorption, and the absorption probabilities

$$\mathbf{t} = \mathbf{N}\tau \tag{2.4}$$

$$\mathbf{B} = \mathbf{NR}, \tag{2.5}$$

where τ is a vector of the average escape times from each transient state, $\mathbf{t}$ is a vector where element i is the average time until absorption if the chain starts in state i, and $\mathbf{B}$ is a matrix whose ij entry corresponds to the probability that if the chain starts in state i it will be absorbed into state j.

While MCAMC gives exact times and probabilities between any set of states, it does not say which states should be grouped together in the transient subspace. Two heuristic algorithms have been developed for this classification. The first scheme is to count the number of times a transition has occurred between two states and when this number is greater than a specified threshold (typically tens of transitions) the two states are grouped together. A second scheme that is based on the energies of the saddles and minima is also implemented.[36] These different definitions of the transient space effect the resolution of the state-to-state description, but not the accuracy of the course grained simulation, since the MCAMC is exact for any choice of transient states.

Using AKMC as implemented in EON, atomistic and molecular systems have been simulated using forces and energies both from empirical potentials when available and DFT when higher accuracy is desired.

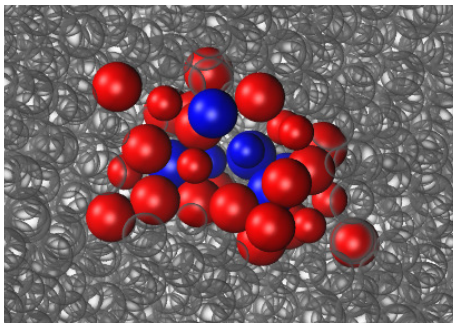

Figure 2.5: A diffusion event in a $Cu_{0.7}Zr_{0.3}$ glass crosses an energy barrier of 0.40 eV and lowers the structural energy by 0.79 eV. The atoms colored blue and red undergo displacements larger than 0.5 and 0.15 Å, respectively, during the event. The small atoms are Cu and the large atoms are Zr.

2.4.2.1 CuZr glass.

In a study of CuZr bulk metallic glasses the dynamical behavior at the μs time scale was simulated for a $Cu_{0.7}Zr_{0.3}$ alloy at 500 K. As this system is highly disordered, it is considered an intractable task either to propose a sufficiently complete table of events *a priori* or construct a standard on-lattice KMC simulation of the system. In this simulation, the super cell contained $\sim$1400 atoms and the number of accessible mechanisms found from each state ranged from 10 to 150. The atomic interactions were modeled applying an EAM potential as implemented in the LAMMPS code [37]. An example of the complexity of the reaction mechanisms is shown in Figure 2.5, where an annealing event, which lowered the structural energy by 0.79 eV, involved the displacement of 40 atoms more than 0.15 Å, of these, seven atoms were displaced more than 0.5 Å.

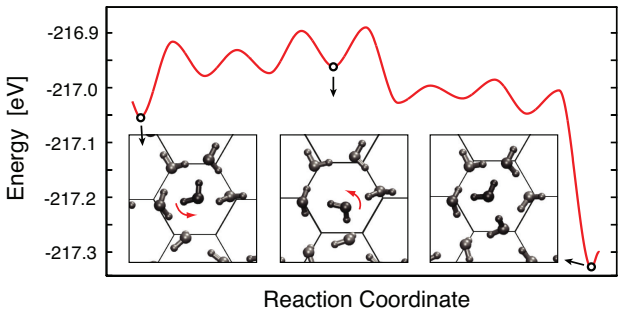

Figure 2.6: A set of reactions involving a water molecule on an ice surface. Seven events, overcoming an effective barrier of 0.14 eV results in a rearrangement of the proton order and the surface energy lowers by 0.28 eV and stabilize due to a more favorable morphology of the dangling hydrogen atoms.

2.4.2.2 H_2O on the surface of hexagonal ice.

A simulation of a molecular system modeled the diffusion of a water add-molecule on the basal (0001) surface of hexagonal ice. As such systems are of interest to the astronomy community, the simulations were conducted at low temperatures (<200 K) where reaction mechanisms with barriers of 0.3 eV become rare events. The average time for such an event at 100 K is on the order of minutes. In the present study the empirical TIP4P-flex potential [38] was applied. Figure 2.6 shows the structural rearrangements resulting from a series of six event, which lowered the system energy by 0.28 eV. For this sequence of events to occur, an effective barrier of 0.14 eV was overcome.

2.4.2.3 Breakup of a boron cluster in bulk silicon.

The break-up of a boron cluster in a bulk silicon lattice was modeled at 500 K using energies and forces from DFT. Boron is commonly used as a dopant for p-type silicon. The high B concentration required for nanoscale devices can lead to dopant clustering and deactivation. Thus, the kinetics

of dopant cluster formation and break-up is of interest to the semiconductor industry. The details of the DFT calculation and how the initial configuration was created have been reported previously.[39]

Here we show how on-the-fly coarse graining (MCAMC) [34] and the kinetic database (KDB) [39] can reduce the computational effort needed to model long time scales. We report a reaction pathway for the breakup of B_2I clusters, the discovery of which was enabled by using the geometry comparison routines in Eon to exploit the symmetry of the system to greatly reduce the number of identical states that needed to be explored. The KDB reduces the computational cost of finding new saddles in each state, by reusing information learned in previous states or simulations. Processes are added to the KDB using a minimal representation that includes only moving atoms and their immediate environment. The KDB is queried to provide suggestions of available saddle point geometries. These suggestions accelerate AKMC simulations by reducing the number of random searches needed to reach confidence that a sufficient rate table has been determined.

To show the effects of the KDB and MCAMC, three AKMC simulations were run: the first using both the KDB and MCAMC, the second using only the KDB, and a third using neither acceleration method. For each simulation, the AKMC confidence was set to 0.95, which corresponds to a stopping criterion in each state where no new event was added to the rate table within 20 consecutive searches. Each saddle search was initialized by displacing each degree of freedom of a B atom and all atoms within 2.6 Å by a random number drawn from a Gaussian distribution with a standard deviation of 0.2 Å.

The results are shown in Figure 2.7. The unit of computational time, a cluster-day, is defined as one day of CPU time provided by a local cluster

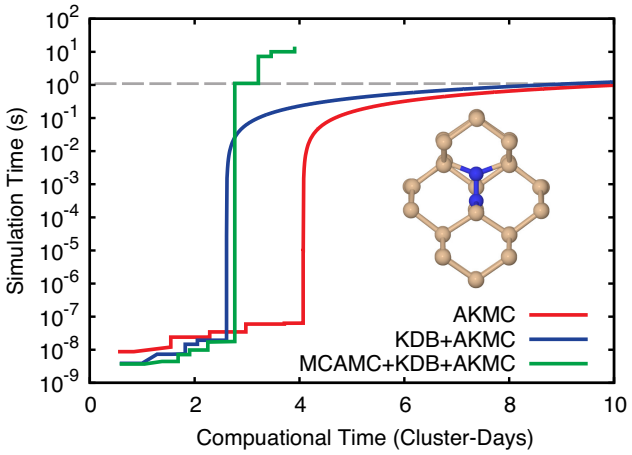

Figure 2.7: A comparison of how the kinetic database (KDB) and Monte Carlo with adsorbing Markov chains (MCAMC) reduce the computation time required to model the dynamics of a B_2I cluster in bulk Si with AKMC.

containing 39 nodes, each with eight Intel Xeon X5355 cores running at a clock rate of 2.66 GHz. The use of the KDB reduces the work needed to explore each state and here it saves more than a day of cluster time. However, without the use of MCAMC the KMC simulation becomes trapped in a set of states separated by low barriers. The mean number of steps required to escape this superbasin is 5×10^9. The number of KMC steps that could be performed per second was 3000, which means that it would have taken 20 cluster-days (on average) to escape from the superbasin each time the simulation entered it. The use of KDB and MCAMC allowed us to find the full B_2I dissociation pathway shown in Figure 2.8.

2.4.3 Basin Hopping

BH is an algorithm for determining global minimum energy structures.[40] In BH, a Monte Carlo (MC) simulation is performed on a transformed PES,

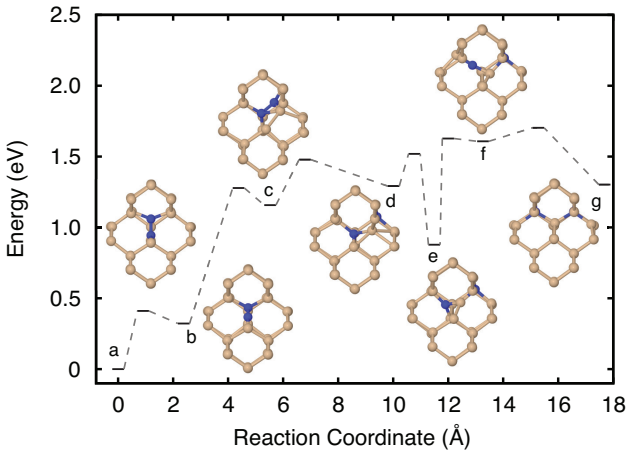

Figure 2.8: A reaction pathway for B_2I cluster rearrangement in Si (a-c) and dissociation (d-g) found in an AKMC simulation using forces and energies from DFT.

$\tilde{E}(\mathbf{X})$, obtained from an energy minimization of the atomic configuration $\mathbf{X}$. The standard Metropolis acceptance probability,

$$P_{\text{acc}} = \min[1, \exp(-(\tilde{E}(\mathbf{X}_{\text{new}}) - \tilde{E}(\mathbf{X}_{\text{old}}))/k_{\text{B}}T)], \tag{2.6}$$

is used where $\mathbf{X}_{\text{old}}$ and $\mathbf{X}_{\text{new}}$ are the configurations before and after each trial move. MC moves are made by displacing each coordinate by either a random uniform or Gaussian distribution. Our implementation also allows for swapping moves, where a pair of atoms of differing elements have their coordinates exchanged. The size of the displacement can also be dynamically updated during the simulation to reach a target acceptance ratio.[41]

There are several enhancements to BH that have been implemented including significant structure basin hopping (SSBH) [42] and basin hopping with occasional jumping (BHOJ) [43]. In SSBH all displacements are made from the local minimum of the previous displacement. In BHOJ, when a predetermined number of MC moves are rejected in a row, a fixed number

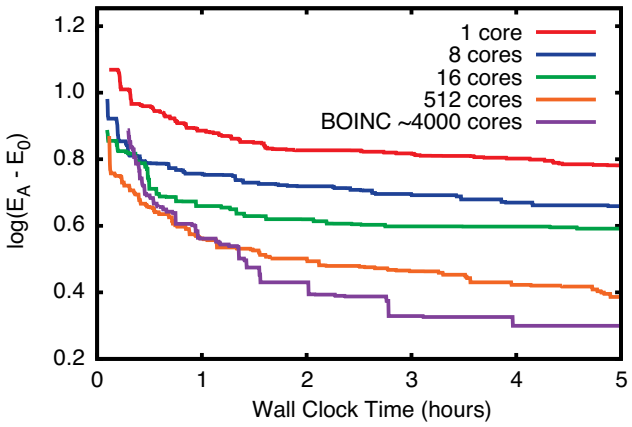

Figure 2.9: The difference between the lowest energy found by BH simulations (E_A) and the lowest known global minimum (E_0) plotted as a function of wall clock time. The amount of time taken to reach lower energies is reduced as more cores are utilized.

of MC moves are performed at infinite temperature. This gives the method a chance to escape from an energy funnel that doesn't contain the global minimum.

As an example, we show results from the global optimization of a $A_{42}B_{58}$ binary Lennard-Jones (LJ) cluster with energy given by:

$$E = 4 \sum_{i<j} \epsilon_{\alpha\beta} \left[\frac{\sigma_{\alpha\beta}}{r_{ij}}^{12} - \frac{\sigma_{\alpha\beta}}{r_{ij}}^{6} \right] \tag{2.7}$$

where α and β are the atom type of atoms i and j, respectively. We chose ϵ_{AA}=ϵ_{AB}=ϵ_{BB}=1, σ_{AA}=1, σ_{BB}=1.3, and σ_{AB}=(σ_{AA}+σ_{BB})/2. Previous studies using more sophisticated global optimization methods have reported a putative global minimum energy of -604.796307. [44]

To highlight the parallelization options available in EON we ran many BH calculations of LJ clusters, starting from a single random initial geometry, across different numbers of CPU cores: serially (1 core), on a local cluster of

Table 2.1: Comparison of the rate of escape from the compact Pt heptamer island state at 400 K using different long time scale methods. The PRD rates are based upon the mean-first-escape times from four trajectories.

Method	Escape Rate (1/s)	Force Calls
PRD	$(7.7 \pm 3.8)\times10^5$	1×10^9
PRD/hyperdynamics	$(6.1 \pm 2.3)\times10^5$	1×10^6
AKMC/HTST	8.7×10^5	2×10^4

computers (8-512 cores), and on a distributed network of machines (BOINC) with about 4000 cores.[25] Figure 2.9 shows the difference between the average lowest energy (E_A) and the energy of the global minimum (E_0) versus wall time for the different parallelization options. The lowest energy structure found in any of the runs was -604.136658.

2.5 Discussion

The motivation for developing the EON software is to make different approaches to long time scale simulations of atomic systems available in an integrated package that can make use of diverse computational resources. The common toolkit of optimizers, dynamics algorithms, saddle point finding methods, and interatomic potentials allows for a direct comparison of methods and makes it easier for the user to select the method which best fits the problem at hand. EON allows for the development of hybrid methods which can take advantage of the strengths and mitigate the weaknesses of the different algorithms.

An example comparison between methods is given in Table 2.1, in which the rate of escape from the compact heptamer island configuration (shown in Figure 2.4) is calculated at 400 K using three different long time scale methods:

PRD, PRD combined with hyperdynamics, and HTST as implemented in the AKMC method. Each of these methods exhibit linear speed-up with respect to the number of CPU cores used so that the wall-clock time to run each simulation is proportional to the number of force calls divided by the number of cores. On the previously mentioned 2.66 GHz Xeon X5355 processors, one force call takes about 8 ms on a single core, so that the HTST calculation takes 2.6 minutes on 1 core, the PRD/hyperdynamics calculation takes 16 minutes on 16 cores, and the PRD calculation takes 1.4 days on 64 cores. These wall clock times closely reflect the conditions under which the simulations were run.

The escape rate for the compact Pt heptamer island can be estimated with the fewest force calls using AKMC, where the transition mechanisms are found using saddle point searches and the rate of each mechanism is calculated using HTST as in Eq. 2.1. HTST is computationally efficient, but it relies on several assumptions that can contribute systematic errors to the escape rate. First, some relevant saddle points may not be identified by the search algorithm, resulting in an incomplete rate table; second, the harmonic approximation to the TST rate may not be accurate enough at the simulation temperature; and third, the TST approximation itself may contain errors due to dynamical recrossing events.

To test these approximations, a more accurate estimate of the rate was calculated using PRD, which relies on fewer assumptions. The equations of motion were integrated with a 2 fs time step and a dephasing time of 2 ps. The Andersen thermostat was used to sample from the NVT ensemble with soft collisions, rescaling 20% of the velocity of atoms on a time scale 20 fs.

Table 2.1 shows that the HTST rate is within the uncertainty of the rate calculated with PRD. At high enough temperature, the harmonic approximation

will break down and the HTST rate will become less accurate. However, when HTST holds, AKMC is an efficient method, being orders of magnitude less computationally demanding than PRD. The AKMC approach can also, in principle, be extended beyond HTST to variationally optimized hyper planar TST, and could, furthermore, be implemented to include dynamical corrections.[45] Understanding the tradeoff between computational cost and accuracy of TST calculations beyond the harmonic approximation are the subject of ongoing studies.

Hyperdynamics is a good compromise between PRD and HTST because it is substantially faster than PRD when a good bias potential is known, and the systematic errors which can be introduced with a poor choice of bias potential can be quantified. The PRD/hyperdynamics rate in Table 2.1 was calculated using the bond-boost bias potential with the parameters described in Sec. 2.4.1. The parameters were tuned so that the bias potential smoothly reached zero at the lowest energy saddle point (0.6 eV) with a maximum fractional bond stretch of 22%. The data in Table 2.1 shows a hyperdynamics rate in agreement with that of PRD and a thousand-fold gain in computational efficiency. It should be noted, however, that without the PRD calculation for reference, the hyperdynamics calculation would have to be repeated with more conservative settings to check for systematic errors introduced by the bias potential.

While much progress has been made in the development of algorithms for long time scale simulations, this is still an important challenge to further development and new methods are being proposed frequently. An essential aspect of this work should be systematic and careful benchmarking and comparison of the performance of the various approaches, both in terms of accuracy and computational effort. To further this endeavor, we are developing a community-

based website where authors can publish and compare their methods and codes on benchmark problems. The benchmark website can be accessed at http://optbench.org/.

The EON code is freely available under the GNU Public License version 3. EON can be obtained at http://theory.cm.utexas.edu/eon/.

2.6 Acknowledgements

The work in Austin was supported by the National Science Foundation (CHE-1152342), the Texas Advanced Computing Center, and the National Energy Research Scientific Computing Center. We gratefully acknowledge all the contributors of computer time to the EON BOINC project.

Chapter 3

Molecular Dynamics Saddle Search Adaptive Kinetic Monte Carlo [2]

3.1 Abstract

A method for accelerating molecular dynamics simulations in rare event systems is described. From each new state visited, high temperature molecular dynamics trajectories are used to discover the set of escape mechanisms and rates. This event table is provided to the adaptive kinetic Monte Carlo algorithm to model the evolution of the system from state to state. Importantly, an estimator for the completeness of the calculated rate table in each state is derived. The method is applied to three model systems: adatom diffusion on Al(100); island diffusion on Pt(111); and vacancy cluster ripening in bulk Fe. Connections to the closely-related temperature accelerated dynamics method of Voter and coworkers is discussed.

3.2 Introduction

Adaptive kinetic Monte Carlo (AKMC) is a method which applies dynamically-constructed rate tables to kinetic Monte Carlo (KMC) simulations.[5] For each unique state that the system visits, searches are performed on the po-

[2] S. T. Chill and G. Henkelman, *J. Chem. Phys.* 140, 214110 (2014). G. Henkelman supervised the work in this chapter.

tential energy surface (PES) to find low-energy first-order saddle points leading to adjacent states. Saddle searches have been carried out with minimum mode (min-mode) following algorithms such as the dimer method [29], Raleigh-Ritz minimization [30] as in the hybrid eigenvector following method [31], or the Lanczos method as in the activation relaxation technique (ART) nouveau [32]. Given the geometry of the saddle point, rates can be efficiently calculated for the forward and backward reactions using the harmonic approximation to transition state theory (HTST). In this paper, we compare the efficiency of min-mode following saddle searches to high temperature molecular dynamics (MD) saddle searches.

When using a min-mode following method, initial configurations are generated by displacing away from a minimum energy configuration. The choice of which degrees of freedom to displace (*e.g.* under-coordinated atoms) and the distribution of the displacement (*e.g.* a Gaussian distribution with a predetermined variance) need to be determined for each system under investigation. These parameters not only effect the computational efficiency of the algorithm, but also the accuracy of the resulting KMC simulation, as the distribution of initial configurations and the shape of the potential energy surface determine the probability that a particular saddle will be found. This makes it difficult to calculate the confidence that all of the important reactive events that are relevant at the simulation temperature have been found.

In a MD saddle search, the trajectory is confined to the initial potential energy basin by detecting when it escapes the basin and restarting it within the basin. The escape events can be detected by periodically performing a geometry optimization to determine if the trajectory is still in the initial energy basin. If it has exited, the trajectory is terminated and a nudged elastic band

(NEB) [46, 47] and/or min-mode following calculation is performed to locate the saddle point between the initial and final state basins.

An important advantage of using MD over min-mode following methods to find saddle points is that the probability of finding escape mechanisms with MD is directly proportional to their rates and their relative importance in the AKMC event table. At elevated MD temperatures, high entropy processes are overrepresented as compared to the temperature of interest, but this bias can be corrected within the HTST approximation. This is the strategy used by Voter and coworkers in their temperature accelerated dynamics (TAD) method [48], where escape events are found with high temperature MD trajectories and the escape times at the low temperature are determined from an Arrhenius extrapolation.

This work closely follows the TAD procedure for sampling possible escape pathways with high temperature MD. The difference is that we are not aiming to find just the first escape event at low temperature, instead, we want to find the entire set of escape pathways and rates that are accessible at the low temperature for use in AKMC. Key to the effective use of MD saddle search with AKMC (MDSS-AKMC) is an estimator for the completeness of the rate table. In past work, a confidence in the rate table found with min-mode following methods was based upon an assumed distribution for discovering saddle points, such as a uniform distribution.[33] This assumption can be a poor one. Even in well understood systems where the chosen initial displacement size and direction are close to optimal, it can be hundreds of times more likely to find one saddle than another, even when the two events have a similar rate.[29] With MD saddle searches, however, an unbiased error in the KMC rate catalog can be determined. Recently, Bhute and Chatterjee have shown how this can

be done using a maximum likelihood estimation of the total escape rate.[49, 50] Here, we use the HTST expression as in TAD to derive an estimator for the error in the total rate, and use this as a criterion for sufficient discovery of the rate catalogue to escape a state. In this way, we show how MDSS based AKMC can be done with higher accuracy and sometimes even more efficiently than when based upon min-mode following saddle searches. All numerical calculation were performed with the EON software.[51]

3.3 Error in the Escape Rate Due to an Incomplete Rate Catalog

The complete rate catalog $\mathbf{C}$ is the set of escape rate constants of all N possible escape processes from a potential energy basin. Technically this is a multiset as different processes may have the same rate constant. Initially, when dynamically building the rate catalog, none of the processes are known. As they are identified, they are added to the set of found events $\mathbf{F}$ and removed from the complementary set of missing events $\mathbf{M}$. The total escape rate, K, is defined as the sum of the escape rates, k_i, of each process at the low temperature of interest

$$K = \sum_{i=1}^{N} k_{i,T_{\text{low}}}. \tag{3.1}$$

In a KMC simulation, the probability of picking an event is proportional to its rate. Thus an appropriate error measure E for the rate catalogue $\mathbf{F}$ is the probability of picking one of the missing processes (in $\mathbf{M}$) in a KMC step based upon the complete catalogue $\mathbf{C}$,

$$E(\mathbf{F}) = 1 - \frac{1}{K} \sum_{k_i \in \mathbf{F}} k_{i,T_{\text{low}}} = \frac{1}{K} \sum_{k_i \in \mathbf{M}} k_{i,T_{\text{low}}}. \tag{3.2}$$

Under our assumption of first-order kinetics, the mean-first-escape-time for each process, τ_i, is exponentially distributed according to the rate k_i. Integrating the distribution up to time t yields the probability that the process has occurred by time t

$$p(t; k_i) = \int_0^t k_i \exp(-k_i \tau_i)\, \mathrm{d}\tau_i = 1 - \exp(-k_i t). \tag{3.3}$$

The probability of having found a particular set $\mathbf{F}$ of processes by time t in the high temperature MD simulation is

$$P(\mathbf{F}) = \prod_{k_i \in \mathbf{F}} p(t; k_{i,T_{\mathrm{high}}}) \prod_{k_i \in \mathbf{M}} 1 - p(t; k_{i,T_{\mathrm{high}}}). \tag{3.4}$$

Here, $k_{i,T_{\mathrm{high}}}$ are the rate constants at the high temperature. $P(\mathbf{F})$ represents the joint probability of having independently found the events in $\mathbf{F}$ and having not yet found the events in $\mathbf{M}$. Now we may express the average error at time t by averaging over all possible sets of processes that may be found

$$\overline{E}(\mathbf{C}) = \sum_{\mathbf{F} \in \mathcal{P}(\mathbf{C})} P(\mathbf{F}) E(\mathbf{F}) \tag{3.5}$$

$$= 1 - \frac{1}{K} \sum_{i=1}^{N} p(t; k_{i,T_{\mathrm{high}}}) k_{i,T_{\mathrm{low}}}, \tag{3.6}$$

where the $\mathcal{P}(\mathbf{C})$ represents the power set (the set of all subsets) of $\mathbf{C}$. A derivation of Eq. 3.6 is given in Appendix A. In the case that $k_{1,T_{\mathrm{high}}} = k_{2,T_{\mathrm{high}}} = \ldots = k_{N,T_{\mathrm{high}}} = k_{T_{\mathrm{high}}}$, Eq. 3.6 reduces to the simple form

$$\overline{E}(\mathbf{C}) = 1 - p(t; k_{T_{\mathrm{high}}}) = \exp(-k_{T_{\mathrm{high}}} t). \tag{3.7}$$

Note that there is no dependence upon N in this last expression. This means that if all of high temperature rate constants are equal then the uncertainty in the rate table can be expressed exactly using only that rate.

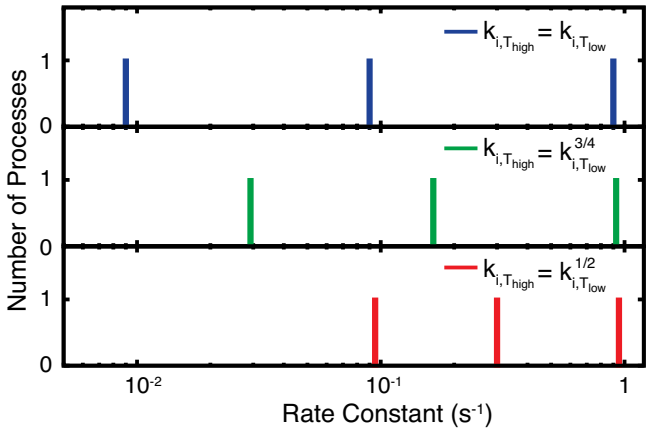

Figure 3.1: Distribution of rate constants $k_{i,T_{\text{low}}}$ in a model system. As the temperature is raised, the rate constants $k_{i,T_{\text{high}}}$ become closer.

3.4 Estimator of the Escape Rate Error

In an AKMC simulation, only the set of found processes **F** are known. The total rate K is unknown and therefore Eqs. 3.2 and 3.5 cannot be evaluated directly. Instead, we can construct an estimator for the average error using information from the set **F** and the MD time used to discover **F**,

$$X(\mathbf{F}) = 1 - \sum_{k_i \in \mathbf{F}} p(t; k_{i,T_{\text{high}}}) k_{i,T_{\text{low}}} / \sum_{k_i \in \mathbf{F}} k_{i,T_{\text{low}}}. \tag{3.8}$$

The assumption made in Eq. 3.8 is that the *average error* at time t from the known set of events **F** is a good estimator for the *error* from the complete set **C**. Another way of stating this approximation is that the events in **F** are characteristic of those in **C**.

The estimator $X(\mathbf{F})$ asymptotically approaches the average error, $\overline{E}(\mathbf{C})$, in two cases. The first is as t approaches infinity (*i.e.* when all processes have been found), where it reduces to Eq. 3.6. The second is as the MD temperatures approaches infinity for systems where each process has the same

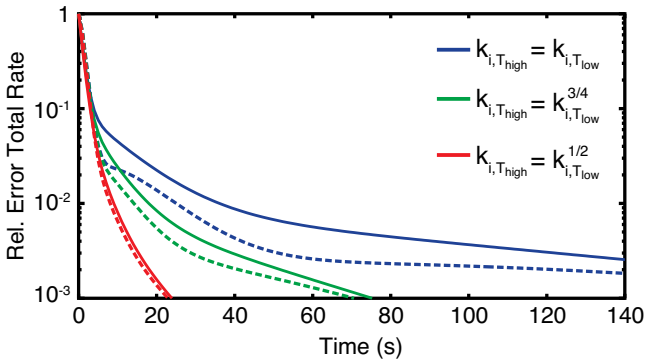

Figure 3.2: Comparison of the average error in the rate catalog $\overline{E}(\mathbf{C})$ (solid line) and the proposed estimator of the error $\overline{X}(\mathbf{F})$ (dashed line) for the three distributions of high temperature rates shown in Fig. 3.1.

entropic prefactor (*i.e.* when $k_{1,T_{\text{high}}} = k_{2,T_{\text{high}}} = \ldots = k_{N,T_{\text{high}}}$), where it reduces to Eq. 3.7.

In order to demonstrate the behavior of the estimator, we have chosen a simple model system with three processes: $k_{1,T_{\text{low}}} = 0.009$, $k_{2,T_{\text{low}}} = 0.09$, and $k_{3,T_{\text{low}}} = 0.9\ \text{s}^{-1}$. Three cases will be examined for the high temperature MD rates: $k_{i,T_{\text{high}}} = k_{i,T_{\text{low}}}$, $k_{i,T_{\text{high}}} = k_{i,T_{\text{low}}}^{3/4}$, $k_{i,T_{\text{high}}} = k_{i,T_{\text{low}}}^{1/2}$. These cases correspond to performing the MD sampling directly at the temperature of interest, at a 33% increase, and at a 100% increase in temperature. In this model, the prefactor of each process is considered to be the same. The rate constants for each case are shown in Fig. 3.1.

The quality of the estimator $X(\mathbf{F})$ is determined by comparing its average, $\overline{X}(\mathbf{C})$, to the exact average error $\overline{E}(\mathbf{C})$. $\overline{X}(\mathbf{C})$ is obtained in a similar manner to Eq. 3.5,

$$\overline{X}(\mathbf{C}) = \sum_{\mathbf{F} \in \mathcal{P}(\mathbf{C})} P(\mathbf{F}) X(\mathbf{F}). \tag{3.9}$$

Results from analytic evaluation of Eqs. 3.5 and 3.9 for the model system are

shown in Fig. 3.2. In the case that $k_{i,T_{\text{high}}} = k_{i,T_{\text{low}}}$, the MD rates are separated by an order of magnitude, which is far from the equal-rate case where the estimator is exact. At short time this leads to significant underestimation of the error because the fast process, which is found first, is not characteristic of the entire set, violating the assumption of the estimator. As the temperature is raised, the rate of finding the processes increases. Importantly, the spread between the high-temperature rates also decreases so that the error is accurately modeled by the estimator at all times.

3.5 Vacancy Cluster Formation in Iron

Systems that have been modeled using long time scale dynamics include materials which have been damaged by radiation. One such model system that has been used to compare long time scale methods is vacancy cluster formation in body centered cubic (bcc) Fe. The system was introduced by Fan et. al to demonstrate their autonomous basin climbing (ABC) algorithm.[52] In their calculation, the coalescence of vacancies into nano-voids was determined to occur on the time scale of hours at an initial temperature of 50°C and a heating rate of 0.01 K/s. Interestingly, a similar calculation was done by Brommer et. al using the kinetic activation relaxation technique (k-ART) [53] who calculated a time scale of milliseconds for the coalescence – a difference of eight orders of magnitude. This remarkable disagreement provides a strong motivation for developing benchmarks that can be used to compare the accuracy of different long timescale dynamics methods. As such, we define such a benchmark which is close to these previous calculations, which we then also use to test our error estimator in MDSS-AKMC.

The initial configuration for the benchmark has 50 randomly placed

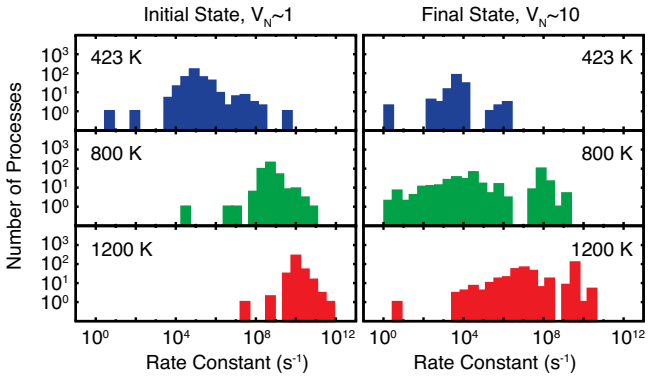

Figure 3.3: The effect of temperature on the HTST rate constants in a bcc Fe lattice with vacancies.

vacancies in a $10\times10\times10\,a_0$ supercell of bcc Fe, where $a_0 = 2.87$ Å is taken as the experimental lattice constant. Initially, the average vacancy cluster size, V_n is unity (or very close to unity). The state-to-state evolution of the system is followed in time based upon rate constants calculated using HTST with a fixed entropic prefactor of 5×10^{12} s^{-1} at a temperature of 423 K. A fixed prefactor was chosen to focus the benchmark to the efficiency of saddle point determination. States are defined as the set of points that minimize to the same geometry. The potential energy is evaluated with an embedded atom method (EAM) model, as parameterized by Ackland et. al.[54] The requirement of the benchmark is to determine the average time for the potential energy to decrease below -7763.5 eV. This final energy corresponds to an average vacancy cluster size $V_n > 9$.

The choice of T_{high} is important for the efficiency of MDSS-AKMC. Increasing T_{high} increases the rate at which processes are found. Too high, however, and a systematic error is introduced in $X(\mathbf{F})$ due to anharmonic corrections to the HTST rates in Eq. 3.8 and the loss of first-order kinetics. The second issue can be addressed by reaching local equilibrium before running high

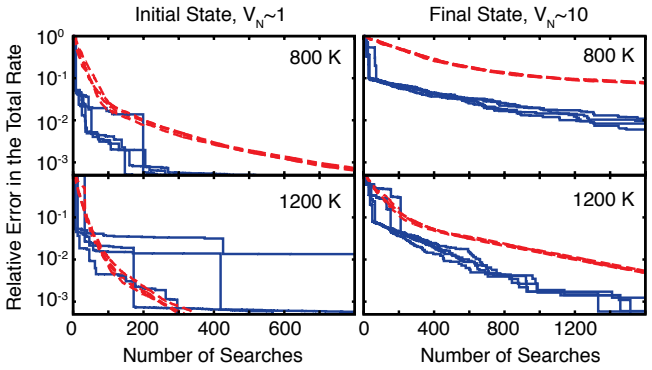

Figure 3.4: The relative error in the rate, $E(\mathbf{F})$, (blue solid lines) compared with the estimated error, $X(\mathbf{F})$, (red dashed lines) at 800 K (top) and 1200 K (bottom). Four realizations are superimposed to show the random variation between simulations.

temperature dynamics as in modified TAD.[55] Fig. 3.3 shows the spectrum of rate constants for an initial configuration ($V_n \sim 1$) and a final configuration ($V_n \sim 10$) at 423 K (T_{low}), 800 K, and 1200 K. In both states, MD at T_{low} cannot be used to sample transitions on the picosecond time scale of our saddle searches. At 800 K the temperature is sufficient to overcome the relatively low barriers of vacancy diffusion in the initial state. In the final state, however, when the vacancies have clustered, a higher temperature of 1200 K is necessary.

The accuracy of the estimator is shown in Fig. 3.4 as a function of T_{high}. In the initial state, $X(\mathbf{F})$ is a good (and safe) estimator of the error in the rate catalogue at 800 K. At 1200 K the harmonic approximation starts to break down, and the estimator loses accuracy. In the final state 1200 K is appropriate and the estimator is accurate. In order to overcome the high barriers at the end of the simulation, we choose T_{high} to be 1200 K. Tuning T_{high} appropriately for different states would be a natural improvement to the method.

Four independent MDSS-AKMC simulations were run with MD saddle

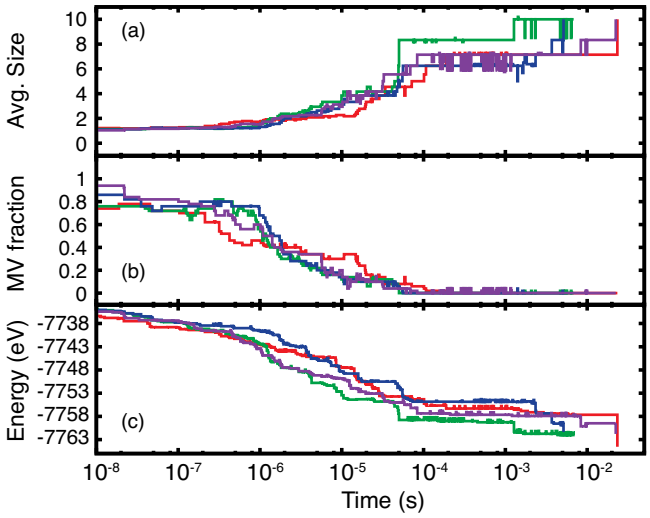

Figure 3.5: Four AKMC trajectories of 50 randomly distributed vacancies in a $10\times10\times10$ supercell of bcc Fe. (a) Average vacancy cluster size; (b) fraction of monovacancies (MV); and (c) potential energy of the minima along the trajectories.

searches performed in each state at 1200 K, until $X(\mathbf{F}) < 0.01$, which corresponds to a 99% confidence in the total escape rate. The average time taken to reach a potential energy of less than -7763.5 eV was calculated as 12 ± 5 ms. Figure 3.5 shows the average vacancy cluster size, fraction of defects that are monovacancies, and the potential energy as functions of time for each trajectory. While it is not possible to directly compare this time to previously reported ABC and k-ART simulations because of differences in the temperature profile, our calculated time scale for vacancy cluster formation are in much better agreement with k-ART than ABC.

3.6 Comparing the Efficiency of MD and Dimer Saddle Searches

Saddle searches based on MD have the advantage of allowing an error estimator of the escape rate from a state. This does not nessecarily mean, however, that MD is a computationally efficient way of finding saddle points. A numerical comparison of MD and dimer saddle searches is done for three systems: Al adatom diffusion on an Al(100) surface modeled with an an embedded atom model developed by Voter and Chen [18]; the motion of a compact Pt heptamer island on a Pt(111) surface modeled with a a Morse potential [56]; and vacancy cluster formation in bcc Fe, as described in Sec. 3.5.

Our metric for comparing the saddle search methods is the average relative error in the total escape rate. Here the escape rate is defined as the rate to exit from the initial potential energy basin and is obtained by averaging Eq. 3.2 over 50 runs vs. the number of potential energy (force) evaluations. All rates are calculated using HTST, with a constant vibrational prefactor of 5×10^{12} s^{-1}, at 300 K for the Al and Fe systems and at 700 K for the Pt system. The total escape rate K in Eq. 3.2 was evaluated by first running 20,000 high temperature MD saddle searches to obtain a rate catalog that was considered complete.

The initial distribution of configurations for the dimer saddle searches was tuned for each system based upon chemical intuition. In this way, *a priori* knowledge of likely reaction mechanisms can be used to reduce the computational effort. In each case, searches were initiated with displacements from the reactant minimum drawn from a Gaussian distribution in a subset of the Cartesian degrees of freedom. In the Al system, the adatom and its first coordination shell (15 degrees of freedom) were displaced with a standard

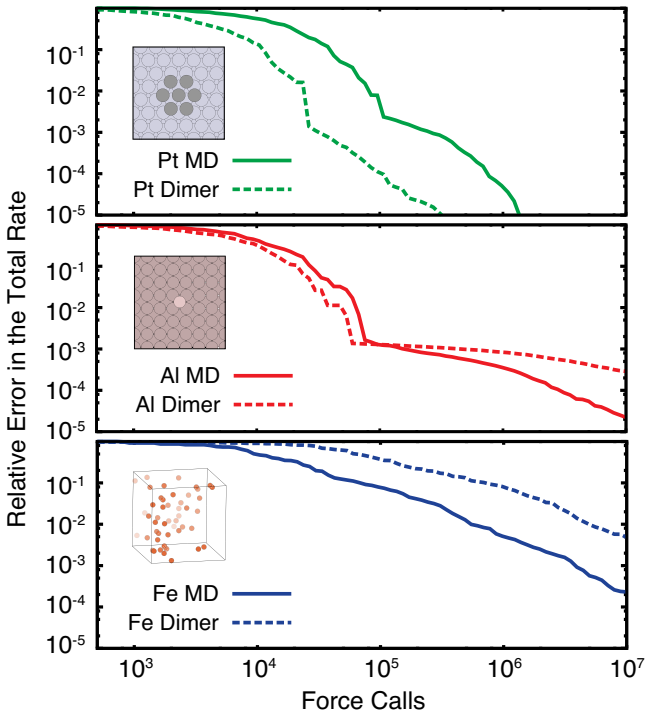

Figure 3.6: An efficiency comparison of dimer and MD saddle searches for three different systems: Pt heptamer diffusion; Al adatom diffusion; and vacancy cluster ripening in bcc Fe. The inset for the Fe system shows the initial distribution of vacancies.

deviation of $\sigma = 0.2$ Å. In the Pt system, all seven island atoms (21 degrees of freedom) were displaced by $\sigma = 0.1$ Å. In the Fe system a random Fe atom with coordination number less than eight was displaced, as well as all neighbors within 6 Å, by $\sigma = 0.2$ Å. For the MD saddle searches only the T_{high} parameter is required; temperatures of 1000, 1200, and 2000 K were chosen for the Al, Fe, and Pt systems respectively.

Differences in efficiency are shown for the three systems in Fig. 3.6. In the Pt system, the dimer searches significantly outperform the MD searches. This is due to the localized displacement scheme that effectively targets the

most important mechanism of island sliding.

In the Al system, diffusion mechanisms involve both the adatom and surface atoms. This makes it more difficult to construct an effective distribution for dimer search displacements. The performance of the two methods here is similar unless a highly accurate rate catalog is required. In this low-error regime, high-energy long range events involving many atoms must be found and these are hard to find with the dimer initiated with displacements localized around the adatom.

In the initial state of the Fe system, there are 337 atoms that neighbor the 50 vacancies. These under-coordinated atoms are targeted by the dimer searches. While each of the vacancies can diffuse, the escape rate is dominated by a single fast process involving two nearby vacancies. This outlier can be seen in the initial state spectrum of rates at 423 K in Fig. 3.3. Since a random selection of the correct under-coordinated atom to displace has a small probability, the MD search strategy is more efficient because it automatically finds the fast event with a high probability.

Each MD saddle search takes on average several times more force calls to find a saddle. In the Al, Fe, and Pt systems the MD saddle searches were six, two, and four times more expensive, respectively. Despite the increased cost per search, the MD method can still outperform dimer searches. There are a few reasons for this. First, dimer searches can wander to configurations of high energy where they are terminated, or to saddles which do not connect back to the initial state minimum (by steepest descent). Second, while dimer searches can be localized to active regions of configuration space, intuition may not be good enough to target the part of the system with the highest rates (as in the bcc Fe case). While MD has a higher overhead, per search, it is more

Table 3.1: The three fastest events at 300 K for Al adatom diffusion on a Al(100) surface.

Event	Prefactor	Barrier	HTST Rate (s^{-1})		MD Rate (s^{-1})
	(s^{-1})	(eV)	300 K	800 K	800 K
2-Atom Exchange	1.4×10^{13}	0.206	5.0×10^{9}	7.3×10^{11}	$(4.2 \pm 0.3)\times10^{11}$
Hop	5.8×10^{12}	0.377	2.7×10^{6}	2.4×10^{10}	$(7.7 \pm 0.8)\times10^{10}$
4-Atom Exchange	2.0×10^{14}	0.396	4.6×10^{7}	6.6×10^{11}	$(2.9 \pm 0.5)\times10^{10}$

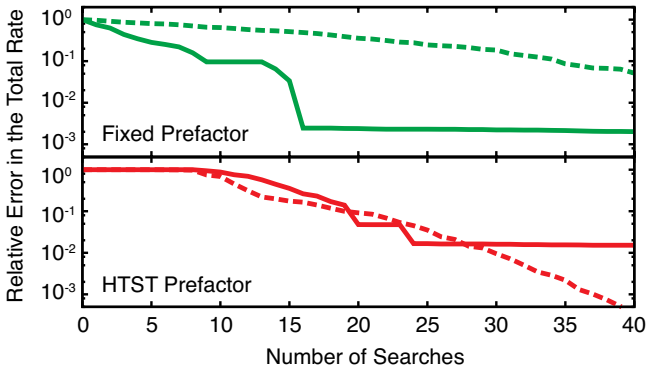

Figure 3.7: The average relative error in the rate for Al adatom diffusion on Al(100) at 300 K with MDSS performed at 800 K, $\overline{E}(\mathbf{C})$, (solid lines) compared with the average estimated error, $\overline{X}(\mathbf{C})$, (dashed lines) with a fixed harmonic prefactor of 5×10^{12} s^{-1} (top) and the Vineyard prefactor (bottom).

likely to find relevant saddles of high rate which are connected to the initial state. Combined with a good error estimation, MD searches can be preferable, particularly in cases where high accuracy is desired.

3.7 Discussion

In order to evaluate the estimator $X(\mathbf{F})$ the high temperature rates $k_{i,T_{\text{high}}}$ must be known. In the Fe vacancy cluster formation simulation, the estimator, when evaluated with these HTST rates, was found to be accurate

enough to be useful as a stopping criterion. However, it is not always the case that the HTST rate is a good estimate of the high temperature escape rate.

For example, the three fastest diffusion events in the Al adatom on Al(100) system at 300 K are a 2-atom exchange mechanism, where the adatom pushes a substrate atom up onto the surface; a hop mechanism, where the adatom moves directly to a neighboring site; and a 4-atom exchange mechanism where the adatom pushes three substrate atoms so that one surfaces three sites away. The rates of these events as calculated by HTST and MD are shown in Table 3.1. The HTST rate of the 4-atom exchange mechanism is 22 times greater than what is observed in a direct MD simulation. The high HTST rate means that the estimator underestimates the error on the characteristic timescale of this event. Figure 3.7 shows the average value of the error estimator compared to the true average error. Two cases are considered: a constant prefactor for all events of 5×10^{12} s^{-1} and a Vineyard harmonic prefactor calculated by diagonalizing the dynamical matrix, which in turn is calculated by finite difference. [57] In the case of the constant prefactor the predicted high temperature rates significantly underestimate the true rate, which results in an overly conservative estimator. With the harmonic prefactor, the error estimator is accurate until the relative error reaches the contribution of the 4-atom exchange, at about 1% of the total rate. At this point the estimator diverges from the true error. A promising future direction is to obtain a more accurate true rate from the statistics of the high temperature MD trajectory.

MDSS-AKMC is similar to Voter's TAD method, but there are significant differences. In TAD, high temperature MD is used to find escape events from a state and the time at which that event would have occurred at a lower temperature of interest is extrapolated from HTST. Once confidence has been

reached that the first event at low temperature has been found, the transition is taken and the processes is repeated in the new state. One might then ask why one should do the additional work in MDSS-AKMC to reach confidence for the rate catalogue. First, an advantage of doing AKMC with the rate catalogue is that it is based upon rates calculated at the low temperature, and does not rely on an extrapolation based upon the HTST approximation at the high temperature, as in TAD. In principle the AKMC rates can be made as accurate as desired, for example using dynamical corrections to TST. Second, MDSS-AKMC can be augmented with computational strategies that efficiently recover known reaction mechanisms, including saddle recycling [33] and the kinetic database [39]. Third, AKMC allows for efficient coarse-graining of fast rates through the Monte Carlo with absorbing Markov chains approach [58]. The relative strengths of TAD and MDSS-AKMC, as well as the possibility of a hybrid approach, will be the subject of future studies.

3.8 Conclusion

We have described a method to determine the events that go into an AKMC rate catalog using high temperature MD saddle searches. In simulations of surface and bulk diffusion, this MDSS-AKMC method is shown to be efficient for the calculation of long time scale dynamics in comparison to AKMC based upon dimer saddle searches.

3.9 Acknowledgements

This work was supported by the National Science Foundation under grant CHE-1152342 and the Welch Foundation under grant F-1841. We grate-

fully acknowledge computational resources at the Texas Advanced Computing Center and the generous people who contributed computing time through the EON project [25], and Gideon Simpson for helpful discussions.

3.10 Appendix

Here we give the details of how Eq. 3.6 is derived from the definition given in Eq. 3.5.

Proof.

$$\overline{E}(\mathbf{C}) = \sum_{\mathbf{F}\in\mathcal{P}(\mathbf{C})} P(\mathbf{F})E(\mathbf{F}) \tag{3.5}$$

$$= \frac{1}{K} \sum_{\mathbf{F}\in\mathcal{P}(\mathbf{C})} \sum_{k_i\in\mathbf{M}} k_{i,T_{\text{low}}} \prod_{k_j\in\mathbf{F}} p(t;k_{j,T_{\text{high}}}) \prod_{k_j\in\mathbf{M}} 1 - p(t;k_{j,T_{\text{high}}}) \tag{3.10}$$

$$= \frac{1}{K} \sum_{n_1=0}^{1} \cdots \sum_{n_N=0}^{1} \sum_{i=1}^{N} (1-n_i) k_{i,T_{\text{low}}} \prod_{j=1}^{N} p(t;k_{j,T_{\text{high}}})^{n_j} (1-p(t;k_{j,T_{\text{high}}}))^{1-n_j} \tag{3.11}$$

$$= \frac{1}{K} \Bigg[k_{N,T_{\text{low}}} (1-p(t;k_{N,T_{\text{high}}})) \sum_{n_1=0}^{1} \cdots \sum_{n_{N-1}=0}^{1} \prod_{j=1}^{N-1} p(t;k_{i,T_{\text{high}}})^{n_j} (1-p(t;k_{i,T_{\text{high}}}))^{1-n_j}$$
$$+ \sum_{n_1=0}^{1} \cdots \sum_{n_{N-1}=0}^{1} \sum_{i=1}^{N-1} k_i n_i \prod_{j=1}^{N-1} p(t;k_{i,T_{\text{high}}})^{n_j} (1-p(t;k_{i,T_{\text{high}}}))^{1-n_j} \Bigg] \tag{3.12}$$

$$= \frac{1}{K} k_{N,T_{\text{low}}} (1-p(t;k_{N,T_{\text{high}}})) + \overline{E}(\{k_1,\ldots,k_{N-1}\}) \tag{3.13}$$

$$= \frac{1}{K} \sum_{i=1}^{N} (1-p(t;k_{i,T_{\text{high}}})) k_{i,T_{\text{low}}} \tag{3.14}$$

$$= 1 - \frac{1}{K} \sum_{i=1}^{N} p(t;k_{i,T_{\text{high}}}) k_{i,T_{\text{low}}} \tag{3.6}$$

Eq. 3.10 follows from the definitions of $E(\mathbf{F})$ and $P(\mathbf{F})$. In Eq. 3.11, the sum over the power set of $\mathbf{C}$ has been re-written as N sums over indicator variables (n_i) in order to enumerate all subsets of $\mathbf{C}$. In Eq. 3.12, the N^{th} sum has been explicitly evaluated for $n_N = 0$ and $n_N = 1$. The factor $\sum_{n_1=0}^{1} \cdots \sum_{n_{N-1}=0}^{1} \prod_{j=1}^{N-1} p(t; k_{i,T_{\text{high}}})^{n_j} (1 - p(t; k_{i,T_{\text{high}}}))^{1-n_j}$ is equal to one as it represents the sum of probabilities of all the ways to find any subset of $\{k_1, \ldots, k_{N-1}\}$. Eq. 3.13 is a recursion relation for $\overline{E}(\mathbf{C})$ used to give Eq. 3.14, which is equivalent to Eq. 3.6. □

Chapter 4

Benchmarks for characterization of minima, transition states and pathways in atomic systems

4.1 Abstract

A set of benchmark systems is defined to compare different computational approaches for characterizing local minima, transition states, and pathways in atomic and molecular systems. Comparisons between several commonly used methods are presented. The strengths and weaknesses are discussed, as well as some implementation details that are important for achieving the best performance. All of the benchmarks and methodology are provided in an online database to make the implementation details available and the results reproducible. While this paper provides a snapshot of the benchmark results, the online framework is structured to be dynamic and incorporate new methods and codes as they are developed.

4.2 Introduction

Computational tools for geometry optimization of potential energy surfaces (PESs) are ubiquitous in the field of computational chemistry and molecular and materials science. The most appropriate and efficient tools are generally determined by a few limited comparisons between available methods

and codes, rather than a systematic consensus of the strengths and weaknesses of different methods. There are several reasons for this. One is that the efficiency of different methods can vary with the system in question, so it is may be difficult to draw general conclusions in terms of the performance. Another issue is that different research groups have their own codes, often involving methods that they have developed, and are naturally biased towards using them. Furthermore, the barriers associated with distributing and supporting codes for others to use, as well as understanding and implementing different methods, makes systematic comparisons difficult. Finally, the community has not emphasized the importance of standard benchmarks. Such a benchmark database, containing codes and results, will facilitate comparisons between methods and implementations, as well as making it easier to draw general conclusions regarding performance. The benefit will be both for the community of developers as well as for users who want to understand the similarities and differences of available methods.

The performance of geometry optimization algorithms can be sensitive to the form of the underlying objective function. In this regard, our focus in the present contribution is specifically on systems in chemistry and materials science, where we want to find atomic structures that are stationary points on the PES. Local minima are usually the first structures to be considered, because they provide information about the thermodynamically stable states of the system. We are also interested in the kinetics that determine the pathways between the stable states. Within the context of transition state theory (TST) the first-order saddle points that connect states via intrinsic reaction coordinate, minimum energy paths (MEPs), are also important.[59] Accordingly, we also investigate methods for finding these transition states and the approximate

steepest-descent paths that connect them, from which energy barriers and reaction rates can be determined.

This manuscript is structured as follows: First, we define our choice of benchmark systems and codes to test, and perform a comparison of minimization methods to find local minima. Second, we compare global optimization methods, involving a long sequence of displacements from local minima and minimizations, designed to find globally low-energy structures. Third, we investigate single-ended saddle point finding methods to find a nearby saddle from a given initial position. Fourth, we compare double-ended saddle point finding methods, which attempt to locate a saddle between a specified initial and final position. Finally, we look at methods to find all the low energy saddles that lead from an initial minimum in order to calculate the rate of escape from harmonic TST. The paper concludes with a discussion of the results and information about the online benchmark database.

4.3 Benchmark systems

Four different atomic clusters and condensed phase systems are compared in these benchmarks: (a) a cluster of 38 particles interacting through a pairwise Lennard-Jones potential (LJ_{38}); (b) a two-component LJ cluster with 100 particles (BLJ_{100}); (c) a seven-atom heptamer island supported on a low energy (111) surface of a face-centered cubic (FCC) material interacting via a pairwise Morse potential; and (d) a bulk FCC system for the same Morse potential. Figure 4.1 illustrates these four systems.

The LJ potential employs the standard 6-12 form,[60] appropriate for

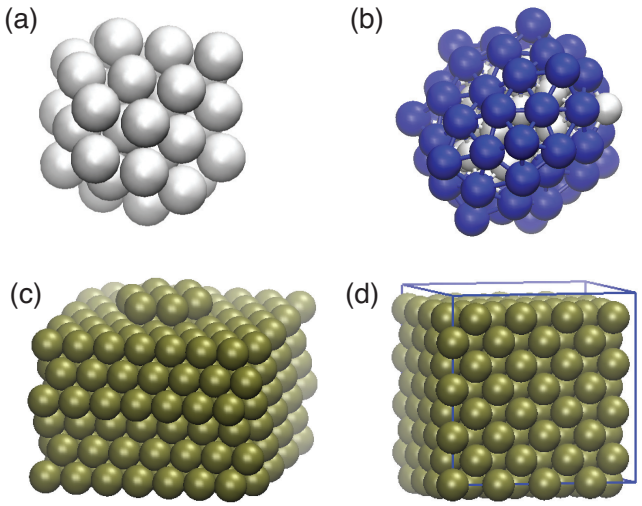

Figure 4.1: The four benchmark systems: (a) Lennard-Jones 38-particle cluster, LJ_{38}; (b) two-component LJ_{100} cluster; (c) heptamer island on a (111) surface; and (d) bulk Pt as described by a Morse potential.

describing the interaction between noble gas atoms,

$$V_{\mathrm{LJ}} = 4\epsilon \sum_{i<j} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right], \tag{4.1}$$

where $r_{ij} = \|r_j - r_i\|$ and r_i is the position vector for atom i. The Morse potential also takes a pairwise additive form,[61]

$$V_{\mathrm{Morse}} = \left[\sum_{i<j} A e^{-2a(r_{ij}-r_0)} - 2A e^{-a(r_{ij}-r_0)} \right], \tag{4.2}$$

where A=0.7102 eV, a=1.6047 Å^{-1}, and r_0=2.8970 Å were chosen to fit bulk Pt[56].

There are a couple of differences between these systems that can affect the performance of the methods. The LJ_{38} and LJ_{100} systems are clusters in the gas phase, where rotations and translations need to be explicitly accounted

for. The LJ potential is also quite 'stiff' in terms of producing relatively large forces and curvatures when atoms approach each other. The Morse potential in the present parameterisation is 'softer' and somewhat easier for quasi-Newton optimizers (those that build up a local approximation to the curvature) to handle. The surface system has frozen atoms, which automatically prevent translation and rotation. These differences affect the performance of the different optimizers, especially the searches for transition states.

4.4 Method implementations

Four in-house codes, PELE[62], OPTIM[63], GMIN[64], and EON[65] were compared, as well as ASE[66], and SCIPY[67]. These programs implement a variety of optimizers, including limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS),[13] conjugate gradients (CG), [12, 68] fast inertial relaxation engine (FIRE), [11] the damped dynamics 'quick-min' (QM) method,[10] and steepest-descent (SD). Double-ended transition state algorithms include the climbing-image nudged elastic band (CI-NEB) [46, 47] and doubly nudged elastic band (D-NEB) [69] methods; single-ended min-mode following methods include the hybrid eigenvector-following (EF), [31, 70] dimer, [29, 71, 72] and Lanczos algorithms, as in the activation-relaxation technique (ARTn). [32] Global optimization was performed using basin-hopping (BH) [40].

The details of these methods will not be discussed here; they are fully documented in the references. Instead, we will mention any important implementation details, as well as qualitative differences between the methods which explain their relative performance.

4.5 Results

4.5.1 Local Optimization

In the first local optimization benchmark, a set of 1000 structures for the LJ_{38} cluster are minimized until the magnitude (L^2 norm) of the force is less than 10^{-2} reduced units. The initial structures were generated using a cluster-growing algorithm.[73]

The local optimization benchmark demands that the programs load each of these structures and minimize the energy for the LJ potential until the magnitude of the force is less than 10^{-2} reduced units. The average number of gradient evaluations of the potential (force calls) is reported, as well as the minimum and maximum number of force calls to find any single minimum. Note that we are comparing force-based optimizers, so it is assumed that the force is evaluated at every iteration; the energy is typically evaluated as well, so each force call also counts these calculations. Table 4.1 summarizes a selected set of benchmark results; when codes have similar performance, only one characteristic value is reported. Figure 4.2 shows how the different algorithms compare in a typical run. A full set of results for these tests and all the other benchmarks reported here can be found online at `http://optbench.org/`. A comparison of the performance for typical starting configuration is shown in Figure 4.2.

In this local optimization benchmark, LBFGS is the clear winner, with CG (the other quasi-Newton optimizer) in second place, followed by the other methods. This result is not surprising;[74] both LBFGS and CG employ numerical curvature information to accelerate optimization within a harmonic approximation of the potential. CG can be thought of as having memory of a single curvature, whereas LBFGS builds up an approximate inverse Hessian

Table 4.1: Minimization of LJ_{38} configurations until $\|\text{Force}\| < 10^{-2}$.

Code	Method	Force calls		
		Avg	Min	Max
OPTIM	LBFGS	176	90	421
PELE	LBFGS	177	88	408
EON	LBFGS	181	90	405
ASE	LBFGS	355	166	9317
EON	CG	453	207	1154
EON	FIRE	645	207	2963
EON	QM	3523	667	9929
EON	SD	4901	1355	9982

with a longer memory. In our implementation, the performance of LBFGS gradually improves with the memory, and a default value of 20 cycles works well, though additional speed gains can be made with larger values. Perhaps even more important, however, is that LBFGS provides a step length, whereas CG gives only a direction. Some implementations of CG use a bracketing approach to find a minimum along this direction; the implementation in EON evaluates the curvature along the direction with a second force evaluation and does a Newton's step to approximate the location of the zero in the force along the line. This formulation requires a second force call at each step compared to LBFGS, which is evident from the results in Table 4.1, with CG taking approximately twice as long to converge as LBFGS.

The non-quasi-Newton methods, FIRE, QM, and SD, are all slower. The potential advantage of these methods is stability,[74] although stability of the optimizers was not an issue for this benchmark.

The same relative performance for the different optimizers is reflected in a second benchmark, namely the minimization of 1000 structures randomly

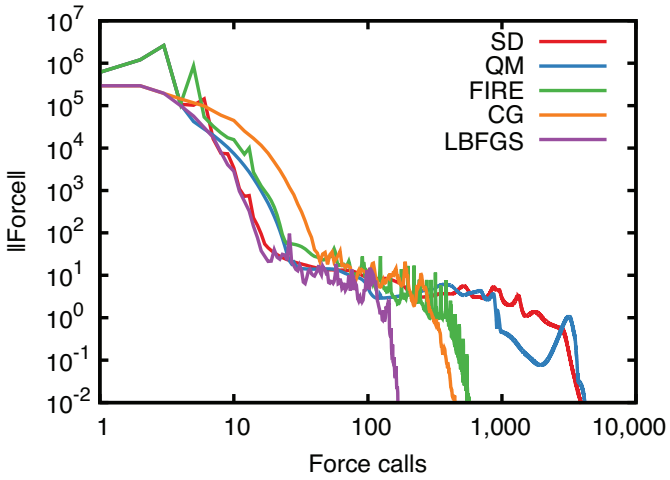

Figure 4.2: Performance of the different optimizers for LJ_{38} as implemented in the EON code.

displaced from a bulk crystalline FCC structure, modeled with a Morse interatomic potential. Again, LBFGS is the fastest, followed by CG and then the methods that do not use approximate curvature information. The LBFGS implementation in ASE is a little slower because it uses a fixed initial Hessian (here 25 eV/Å^2) as opposed to the LBFGS implementations in EON, OPTIM, and PELE, where the initial Hessian is updated at each step based upon the the curvature between the current and previous configurations.

A few comments should be made about our implementations of these optimizers. First, all methods employed a 'max move' parameter; if the optimizers ever try to make a step larger than this maximum, the size is reduced in magnitude to the limit. Our limit is generally set between 0.1–0.2 Å, or a fraction of a LJ distance unit. Another important issue is that the two quasi-Newton methods, LBFGS and CG, can reach ill-conditioned states. When the optimizers operate outside the harmonic region around a minimum, negative Hessian eigenvalues can be introduced into the memory in LBFGS.

Table 4.2: Minimization of a bulk FCC solid until $\|\text{Force}\| < 10^{-3}$ eV/Å.

Code	Method	Force calls		
		Avg	Min	Max
PELE	LBFGS	45	23	73
OPTIM	LBFGS	46	21	80
EON	LBFGS	52	35	82
ASE	LBFGS	62	36	117
EON	CG	106	67	183
EON	FIRE	156	107	212
EON	SD	196	95	360

Following the LBFGS algorithm blindly would take the system to a maximum for such directions. There are different ways to deal with this situation: in EON a 'max move' step is taken parallel to the gradient and the memory is reset; in PELE, OPTIM, and GMIN, the step is inverted so that the algorithm is forced to move down the potential, and the memory is retained. The LBFGS implementations listed simply accept the LBFGS step or do a minimalistic backtracking linesearch, but none of them, by default, use a full linesearch. CG algorithms can also become frustrated if the search direction becomes linearly dependent upon search direction at the previous step. It is essential to check for this condition, which is done implicitly in the Polak-Ribiere update formula [75] and explicitly as in the Powell reset criterion [76].

4.5.2 Global Optimization

Two benchmarks are defined for global optimization; the first is for the same LJ_{38} cluster considered above for the local optimization test, and the second is a two-component BLJ_{100} cluster. The BLJ cluster is composed of 42 A particles and 48 B particles with LJ parameters $\epsilon_{AA} = \epsilon_{AB} = \epsilon_{BB} = 1$,

Table 4.3: Global optimization of LJ_{38} clusters.

Code	Method	Force Calls (thousands)		
		Avg	Min	Max
GMIN	BH-csm	6	0.3	16
GMIN	BH-sym	23	0.6	141
GMIN	BH	266	4.5	916
EON	BH	508	4.6	1796
PELE	BH	522	9.5	2534

Table 4.4: Global optimization of a two-component BLJ_{100} cluster. This reports the average minimum energy energy reached after 2,000,000 force calls. Previous studies[77] have reported a putative global minimum energy of -604.80 reduced energy units.

Code	Method	Lowest Energy		
		Avg	Min	Max
GMIN	BH/KL	-594.68	-602.71	-587.35
EON	BH	-584.58	-596.13	-584.58

$\sigma_{AA} = 1$, $\sigma_{BB} = 1.3$, and $\sigma_{AB} = (\sigma_{AA}+\sigma_{AB})/2$. The first example is a relatively simple test in which the benchmark requires all entries to report the number of force calls, on average, required to find the global minimum from 100 initial structures. The initial structures were generated as in the local optimization benchmark, except that they were initially minimized. The second benchmark is significantly harder because of the two components and the frustration in the energy landscape caused by the different sizes of the particles. Since the algorithms are not expected to locate the global minimum within the number of allowed steps, the benchmark reports the average lowest energy, along with the minimum and maximum values, found from 100 initial structures after 2,000,000 force calls.

Both EON and PELE use standard Cartesian trial moves and LBFGS to minimize trial structures; their performance is comparable. The GMIN code is better optimized for this system, using angular moves for surface atoms[78], as well as symmetrized moves (BH-sym and BH-csm)[79]. However, the best results obtained so far (Table 4.4) correspond to a deterministic search scheme to locate the optimal chemical ordering.[80] This procedure is based on Kernighan and Lin's (KL)[81] heuristic for partitioning graphs.

4.5.3 Single-Ended Saddle Point Searches

Single-ended saddle point finding methods start from a single configuration on the PES and aim to converge to a nearby saddle point. Almost all such methods rely on the determination of an uphill direction along which the potential is a local maximum, while minimizing in the space perpendicular to that direction. A common choice is to use the local lowest curvature mode as the maximization direction; this class of methods is referred to as 'min-mode following' methods. Overall translation and rotation can be projected out in this procedure.[82]

The performance of the min-mode following algorithms can be understood by decomposing them into their two components: (a) identifying the lowest (non-zero) curvature mode and (b) optimizing to the saddle point. We compare three strategies for estimating the lowest curvature mode: minimsation of a Rayleigh-Ritz ratio [30] (hybrid EF methods[31]); rotation of a dimer (dimer method); and the Lanczos, algorithm as employed in ARTn method. In fact, the Rayleigh-Ritz approach used in hybrid EF methods is based upon the same finite-difference gradient of the Rayleigh quotient that is used in the dimer method, as we have recently reviewed.[83] Rayleigh-Ritz minimization and

Table 4.5: Determination of the lowest curvature modes for points near saddles in LJ_{38}. The eigenvector corresponding to the smallest non-zero Hessian eigenvalue is considered converged when the dot product with the exact vector is greater than 0.99.

Code	Method	Force calls		
		Avg	Min	Max
OPTIM	RR	25	13	58
PELE	RR	25	12	61
TSASE	Lanczos	25	13	54
TSASE	dimer (BFGS)	27	13	65
EON	Lanczos	28	13	73
TSASE	dimer (CG)	29	13	80
EON	dimer (BFGS)	39	21	92
EON	dimer (CG)	51	16	274

dimer rotations both correspond to minimization of the same force equations, though the details of implementation can be different.

Table 4.5 compares these strategies by choosing points near saddles in the LJ_{38} cluster and seeing how many force calls are required to determine the lowest eigenvector direction. These methods all use only first derivatives. The lowest (non-zero) curvature mode could be found by diagonalizing the Hessian matrix, but this approach is often significantly slower, especially for larger systems. No initial information about the mode with the smallest non-zero Hessian eigenvalue is provided; the eigenvector corresponding to the unique negative eigenvalue is randomly initialized. Both OPTIM, PELE and EON use LBFGS to as the optimizer in the EF and dimer methods.

The second aspect of single-ended transition state methods is convergence to a saddle, which is achieved by maximizing the potential along the chosen uphill direction and minimization in all other directions. Again, there

Table 4.6: Determination of a transition state from a starting point near a saddle in LJ_{38}. Convergence is defined by $\|\text{Force}\| < 10^{-3}$.

Code	Method	Force calls			Failed
		Avg	Min	Max	
OPTIM	EF	173	64	534	0
PELE	EF	207	58	2294	0
EON	Lanczos	234	65	1635	0
EON	dimer	528	92	3581	0

are different strategies. At each iteration in hybrid EF, the system is moved uphill along the negative mode according to a Newton's method type step, [84, 85] the system is then minimized in the space perpendicular to that mode, and the eigenvector corresponding to the uphill direction is reconverged. In the dimer method, the force is inverted along the lowest mode and followed uphill with a standard (force-based) optimizer. Table 4.6 shows the relative performance of the methods for finding a saddle starting from a point between pairs of adjacent minima in LJ_{38}.

A drawback of the Lanczos algorithm is that several force calls are required to determine that the lowest eigenvector is sufficiently converged. This is because convergence of the eigenvector is measured by a change in the estimated lowest eigenvalue. In contrast, the hybrid EF and dimer methods use the root mean square gradient for the Rayleigh-Ritz ratio or the rotational force on the eigenvector to determine convergence, and in fact, to determine if any steps or rotations are required. Particularly near the saddle, when the lowest mode is found to sufficient accuracy, the hybrid EF and dimer methods can outperform Lanczos by avoiding any refinement of the uphill direction.

This effect is illustrated in a slightly different benchmark, using the Pt-

Table 4.7: Determination of a saddle for the Pt-heptamer island. Convergence is defined by $\|\text{Force}\| < 10^{-4}$ eV/Å.

Code	Method	Force calls			Failed
		Avg	Min	Max	
OPTIM	EF	71	43	143	0
PELE	EF	84	53	147	0
EON	Lanczos	113	78	468	0
EON	dimer	138	90	260	0

heptamer island system, where we start with an initial point near a saddle, but this time the reactant minimum structure is also supplied. Information about a minimum is typically known, for example when single-ended search methods are used to find saddles connected to an initial minimum. The vector between the reactant and the initial search point can be a reasonable guess for the uphill direction if it does not contain any components of Hessian eigenvectors corresponding to overall translation or rotation. Using this information, the performance of the different methods is similar, as shown in Table 4.7.

When fewer force calls are required to find the uphill direction, the advantage of the dimer and hybrid EF methods is apparent.

4.5.4 Double-Ended Saddle Searching

A second class of saddle point methods involves double-ended searches, in the sense that they find a saddle (or a set of saddles and intermediate minima) between specified reactant and product endpoints. To benchmark these methods, we choose pairs of minima known to be separated by a single transition state and see how quickly the corresponding saddle can be found. Table 4.8 shows the results for the LJ_{38} cluster using selected pairs of adjacent

Table 4.8: Determination of a transition state between adjacent minima in the LJ_{38} cluster with a convergence condition of $\|\text{Force}\| < 10^{-3}$.

Code	Method	Force calls			Failed
		Avg	Min	Max	
OPTIM	DNEB+EF	131	66	260	0
EON	Lanczos	250	85	1103	0
EON	dimer	337	89	2079	0
PELE	DNEB+EF	444	189	1415	2
EON	CI-NEB(5)	827	407	2347	7
EON	CI-DNEB(5)	861	407	2187	5

minima. An issue for this system, and all gas-phase molecules, is that overall translation and rotation should be removed. This condition can be achieved (to some extent) by explicitly projecting out these modes in the NEB methods, but the chain-of-states approaches can still suffer from elongation of the path due to displacements between the images along the low or zero-frequency normal modes. In extreme cases, this elongation can lead to slow convergence, low resolution of images around the saddle, and a failure to converge. In this LJ_{38} benchmark, we have chosen a maximum number of force calls (500 iterations) where NEB calculations that start to develop problems, for example, due to long paths through intermediate minima, are considered failures.

An example of a 'failed' NEB calculation is shown in Fig. 4.3. While we know that the minima in question are indeed connected by a single transition state, (found using approximate steepest-descent following small displacements along the eigenvector corresponding to the unique negative Hessian eigenvalue at the transition state), convergence of the NEB from an initial linear interpolation leads to intermediate minima and an elongated path. Using a single-ended search method to find the first transition state, adding more images, or breaking

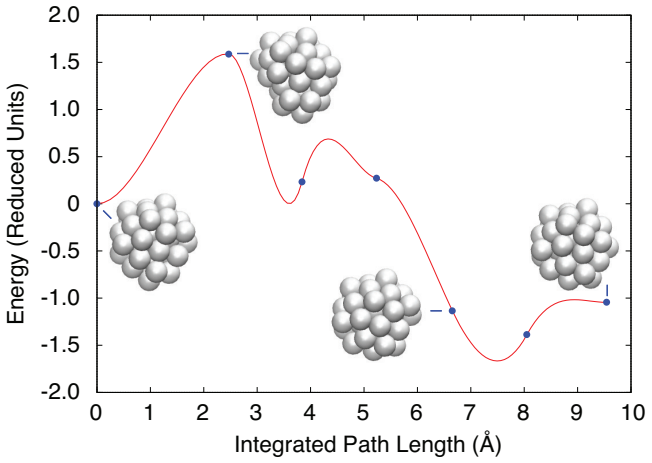

Figure 4.3: Potential energy profile for a NEB calculation that failed to converge in the LJ_{38} system due to the presence of an intermediate minimum and path elongation.

up the band into separate NEB calculations between all local minima, are sensible ways of recovering from this problem.

The single-ended searches are not as sensitive to complex pathways, since they search only for a nearby saddle instead of a path. It is still possible to use them with information from two endpoints. In fact, the dimer and Lanczos methods work very efficiently when initialized half way between two adjacent minima, using the vector between as an initial guess for the negative mode. In PELE and OPTIM the double and single-ended approaches are combined. First a DNEB calculation is performed to find an image in the vicinity of the saddle. The double-nudging helps to keep the band short and smooth, as well as helping to prevent elongation of the path. Instead of a fixed number of images, a constant image density was used. When the force on the highest energy image drops below a loose force threshold (here $\|\text{Force}\| < 0.1$) the algorithm switches to hybrid EF and the transition state is tightly refined.

Table 4.9: Determination of a saddle between adjacent minima for the Pt-heptamer island to a tolerance of $\|\text{Force}\| < 10^{-4}$ eV/Å.

Code	Method	Force calls			Failed
		Avg	Min	Max	
OPTIM	DNEB+EF	103	59	194	0
EON	CI-NEB(1)	110	29	594	0
EON	Lanczos	165	72	308	0
EON	dimer	182	74	322	0
EON	CI-NEB(3)	248	86	761	3
PELE	DNEB+EF	380	141	2267	3
EON	CI-NEB(5)	391	162	797	0

Switching to hybrid EF avoids having to converge the DNEB images, and also improves the stability of the single-ended methods, which is particularly important for gas phase clusters.

For the Pt-heptamer island diffusion test, frozen atoms at the bottom of the slab hold the surface in place and automatically eliminate rotation and translational degrees of freedom. Removing these modes that correspond to zero frequencies improves the relative performance of the NEB method and reduces the need for double-nudging. Another factor in favor of the NEB is that the paths chosen to investigate, 60 low energy diffusion mechanisms from the compact Pt-heptamer, are better approximated by linear interpolation compared to the LJ_{38} cluster.

Table 4.9 shows the number of force calls required to find the transition state along the path, which works efficiently for only a single image CI-NEB calculation, labeled as CI-NEB(1). The dimer and Lanczos algorithms also converge, but with some extra work because the eigenvector corresponding to the negative eigenvalue is optimized instead of being held fixed, as in the single

image CI-NEB calculation. Realistically, one would not generally use a single image for an NEB calculation, so we include the three-image NEB, CI-NEB(3), as a more realistic indication of the cost of the method.

4.6 Discussion and Conclusions

The aim of this work is to establish a set of benchmarks for geometry optimization, transition state searches, and characterization of pathways in atomic, molecular and condensed matter systems. The benchmarks can be accessed at `http://optbench.org/`. All entries are provided with the corresponding source code and scripts to run the benchmark. Hence the results should be reproducible and meaningful comparisons are possible. The results presented here compare a limited number of methods and codes, and include representative examples from the online database. A longer-term aim is to have a dynamic site, which is updated as new methods and codes are developed. Contributions are welcome, either by email or via the subversion (SVN) repository.

4.7 Acknowledgements

This work was supported by the National Science Foundation (CHE-1152342), the Texas Advanced Computing Center at the University of Texas at Austin, the EPSRC, and the European Research Council. We would like to thank Rye Terrell for helpful discussions.

Chapter 5

Using Density Functional Theory and EXAFS to Determine Nanoparticle Structure

5.1 Abstract

A method for quantifying the accuracy in extended X-ray absorption fine structure (EXAFS) fitting models is presented. It is used used to study the structure of bare Au_{147} nanoparticles as well as particles bound with thiol ligands, used to systematically vary disorder in the atomic structure. The accuracy of the fitting models is determined by comparing two distributions of bond lengths: (1) a direct average over a molecular dynamics (MD) trajectory using forces and energies from density functional theory and (2) fits to the theoretical EXAFS spectra generated from that same trajectory. Both Gaussian and third-cumulant expansion EXAFS fitting models are used to characterize the first-shell Au-Au bond length distribution. The Gaussian model is found to significantly underestimate the coordination number, disorder, and bond length. The third-cumulant expansion model yields accurate predictions for bond lengths, but incorrectly predicts a decrease in particle size and little change in the disorder with increasing thiol ligands. A direct analysis of the MD data shows that the particles become much more disordered with ligand binding and the high disorder in the nanoparticles is incorrectly interpreted by the EXAFS models. These theoretical results are realized experimentally for Au nanoparticles of comparable size, synthesized with a dendrimer encapsulation

technique. We finally show that a combination of experimental EXAFS analysis with candidate models from density functional theory is a promising strategy for a more accurate determination of nanoparticle structures.

5.2 Introduction

Extended X-ray absorption fine structure (EXAFS) is a powerful tool for the determination of local atomic structure. The use of *in situ* EXAFS analysis is particularly useful for studying catalysts because changes in the local atomic structure at the surface of a catalyst under reaction conditions can dramatically change reaction rates. While EXAFS is not surface-sensitive, catalytic particles on the scale of 1−2nm have a similar number of surface as bulk atoms, so that surface structure contributes significantly to the total signal.

The analysis of EXAFS scattering is well-established for bulk materials in which the local atomic structure is repeated throughout the material. There is more uncertainty about the appropriate use and accuracy of standard EXAFS fitting models to determine the structure of nanoparticles. The challenge associated with nanoparticles is that they have more disorder in structure which is correlated to the position in the particle (*e.g.* a contraction of bonds at the surface of the particle). The interference pattern in an EXAFS signal emphasizes the ordered component of the structure so that disorder can be underestimated, or as we will show here, convoluted with other fitting parameters such interatomic distances in the particle or the average coordination number of the atoms in the particle, which is used to infer the particle size. Quantification of systematic errors caused by inappropriate EXAFS fitting models is the focus of this paper.

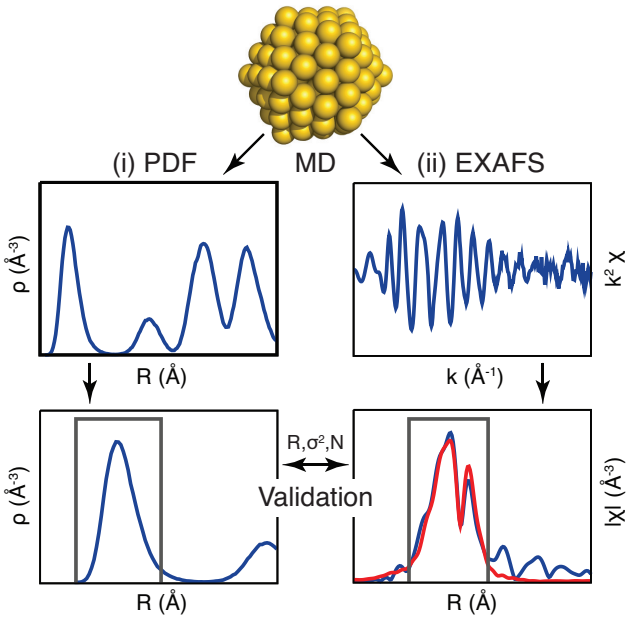

Figure 5.1: The scheme for testing the self-consistency of an EXAFS model: (i) a dynamical trajectory of, in this case, a nanoparticle is simulated and the distribution of bond lengths (PDF) is collected, and (ii) the EXAFS spectra is modeled as an average over the trajectory and subsequently fit using the same EXAFS equations. The resulting parameters, including the average bond length, R, number of neighbors, N, and the Debye-Waller factor, σ^2 are compared to the direct MD simulation.

Key to our understanding of systematic errors in EXAFS fitting is a computational technique designed specifically to test the self-consistency of the EXAFS analysis, as illustrated in Fig. 5.1. Candidate nanoparticles structures are generated and modeled using density functional theory (DFT). Molecular dynamics (MD) trajectories give an ensemble average of structures from which pair-correlation data between the atoms is collected. Analysis of the pair distribution function (PDF), generated from the MD trajectory, allows for a direct calculation of average number of neighbors N, the average bond length R, and a measure of the disorder in the bond length distribution σ^2, quantified by the Debye-Waller factor. We take this PDF data and the structural parameters derived directly from it as the target for EXAFS analysis. We then simulate the EXAFS spectra by averaging the scattering over each structure in the ensemble, as modeled by the FEFF6 software. Precise simulation of experimental data is not essential because we then use the same EXAFS equations used in FEFF6, as provided by FEFFIT, to determine best-fit values of N, R, and σ^2. Comparison with the corresponding values from the structural models used to generate the EXAFS spectra gives us a measure of how appropriate the fitting model is, and a quantification of any systematic errors.

Importantly, this approach does not rely on the accurate simulation of EXAFS spectra as compared to what is observed in an experiment. It is important, however, to connect with experiment in order to understand if systematic errors in EXAFS fits are being made in the determination of nanoparticle structures. Accordingly, we compare our theory to experimental EXAFS measurements of Au nanoparticles and show that a standard EXAFS fit using a bulk reference model with a normal distribution of first-neighbor bonds causes significant errors in the fitted parameters. The systematic error

is shown to increase with disorder, induced by the binding of thiol groups to the nanoparticle surface.

A fitting model using the cumulant expansion method is also examined. Here the Au first-neighbor bond length distribution includes anharmonic corrections to the effective pair potential. This allows the model to account for the non-Gaussian disorder (both static and dynamic) that arises in nanoparticle systems and offers a significant improvement over the Gaussian model in predicting mean bond lengths.

We end by suggesting that DFT modeling can be used in concert with EXAFS to improve the determination the structure of nanoparticle. Since EXAFS does not provide a unique structure, DFT can help to evaluate the relative weight of candidate structures from the calculated total energy. Additionally, explicit theoretical modeling of the EXAFS spectra from many candidate structures, and a subsequent comparison to experiment, can be used to determine which theoretical structures are consistent with the experimental data.

5.2.1 EXAFS Modeling

EXAFS spectra reveal local atomic structure around the X-ray absorbing atoms. The functional form of the spectrum contains many terms, but can be roughly described as exponentially decaying sinusoids, whose amplitudes, frequencies, and decay rates are correlated with coordination numbers, bond lengths, and structural disorder respectively. The most general model for EXAFS single scattering describes the positions of the neighboring atoms using

a probability distribution function

$$\chi(k) = \sum_i \frac{N_i S_0^2 F_i(k)}{kR_i^2} \times \int P(r_i) \frac{\exp(-2r_i/\lambda_i(k))}{kr_i^2} \sin(2kr_i + \delta_i(k))\, \mathrm{d}r_i. \quad (5.1)$$

Here $P(r_i)$ is the probability of finding a neighbor atom between r_i and $r_i + \mathrm{d}r_i$; i is a path index which represents different scattering paths (*e.g.* Au-Au or Au-S single scattering); k is the photo-electron wavenumber; R_i is the average bond length; N_i is average coordination number; S_0^2 is the amplitude reduction factor; $\lambda_i(k)$ is the mean free path of the photo-electron; $F_i(k)$ is the effective scattering amplitude; and $\delta_i(k)$ is the effective phase shift.

Theoretical standards[86] can be used to determine the effective scattering amplitude ($F_i(k)$), the phase shift ($\delta_i(k)$), and the photo-electron mean free path ($\lambda_i(k)$). The amplitude reduction factor (S_0^2) and the correction to the energy origin (ΔE_0) can be determined by fits to experimental standards (*e.g.* a metal foil) or recently by the use of theoretical standards. This leaves $P(r_i)$ as the only unknown in Eq. 5.1.

In order to extract structural information from $P(r_i)$, a fitting model must be assumed. A second order Taylor series expansion of Eq. 5.1 about the mean bond length R_i produces a Gaussian model for the bond length distribution. This is an appropriate model in systems with a low to moderate level of disorder. The resulting EXAFS equation is

$$\chi_2(k) = \sum_i \frac{N_i S_0^2 F_i(k)}{kR_i^2} \exp(-2R_i/\lambda_i(k)) \times \exp(-2k^2\sigma_i^2) \sin(2kR_i + \delta_i(k)). \quad (5.2)$$

The variance of the Gaussian bond length distribution, σ_i^2, is also called the Debye-Waller factor (DWF).

In systems with a high level of disorder the Gaussian model is invalid and a higher order Taylor series expansion must be used. A third order Taylor series expansion introduces a term, C_3 (known as the third cumulant of the distribution), which is a measure of how skewed the bond length distribution is from a Gaussian shape. The resulting EXAFS equation is

$$\chi_3(k) = \sum_i \frac{N_i S_0^2 F_i(k)}{k R_i^2} \exp(-2R_i/\lambda_i(k)) \times \exp(-2k^2\sigma_i^2) \sin(2kR_i - \frac{4k^3}{3}C_3 + \delta_i(k)). \quad (5.3)$$

For a more detailed derivation of the cumulant expansion method see References [87, 88]

5.3 Methods

5.3.1 Experimental Details

Au_{147} dendrimer-encapsulated nanoparticles (DENs) were synthesized as previous described[89, 90] using sixth-generation amine-terminated (G6-NH_2) poly(amidoamine) dendrimers at a concentration of 2.0 µM. The Au_{147} DENs were modified with differing amounts of 2-mercaptoethanol (2ME). The resulting samples are referred to as $Au_{147}@S_n$, where n is the ratio of 2ME : Au_{147} DENs. In this study, values of n = 0, 12, 24, 50, and 72 were investigated.

The $Au_{147}@S_n$ solutions were frozen in liquid N_2 and freeze-dried. For EXAFS analysis, the dried $Au_{147}@S_n$ DENs were mixed with BN and pressed under 1 metric ton of pressure to form a pellet. EXAFS experiments were carried out at the National Synchrotron Light Source (Brookhaven National Laboratory) at beam line X18B. Au L3-edge data were collected at 25 °C in

transmission mode using gas ionization detector chambers. Au foil data were collected simultaneously with the sample data in reference mode.

5.3.2 EXAFS Fitting Details

The Gaussian and cumulant expansion fitting models described in Eqs. 5.2 and 5.3 were used for experimental and theoretical EXAFS data in order to determine N, R, σ^2, and C_3 (for the non-Gaussian model) for the first shell Au-Au pair distribution function for bulk and nanoparticle gold. Fitting models arising from higher order Taylor series expansions of the pair distribution function that include a fourth cumulant term (C_4) were not considered as these models gave a worse fit (increase in the reduced chi-squared) to the experimental dataset. For gold nanoparticles with thiol ligands bound to the surface, the Au-S pair distribution function was modeled as Gaussian.

The effective scattering amplitudes, $F_i(k)$ and effective phase shifts, $\delta_i(k)$ were calculated using FEFF6. The amplitude reduction factor (S_0^2) and the correction to the energy origin (ΔE_0) were determined by a fit to the experimental Au foil.

All nanoparticle fits were performed simultaneously, with the Au-S bond length and Au-S DWF constrained to be the same for each particle and the correction to the energy origin fixed to the experimental bulk foil value. This fitting procedure was carried out using the IFEFFIT software package. [91, 92]

5.3.3 Computational Details

DFT was used to simulate the equilibrium structures of the bulk Au and 147-atom Au nanoparticles either bare or with n thiol groups bound to the surface. For each system, a 10 ps molecular dynamics (MD) simulation

was performed at 300 K to obtain representative geometries for structural and EXAFS analysis. We used the Vienna *ab initio* simulation package (VASP) code [93] with electron correlation treated within the generalized gradient approximation using the PBEsol functional, [94] which is a modified form of Perdew-Burke-Ernzerhof (PBE) [95] with reduced gradient dependence that improves lattice parameters and surface energies in solids. Core electrons were described with the projector augmented-wave method. [96, 97] Kohn-Sham wave functions for the valence electrons were expanded in a plane wave basis set with an energy cutoff of 200 eV.

Bulk Au was modeled with a 4×4×4 face centered cubic (FCC) supercell (64 atoms) with a lattice constant of 4.078 Å. The Au_{147} nanoparticle was modeled as an icosahedron structure with 20 (111) facets, and the 2ME molecule was modeled as a (-S-H) ligand to reduce the computational cost as compared to modeling the entire molecule. Nanoparticles were isolated in a cubic box with side lengths of 28 Å to avoid artificial interactions between periodic images.

Theoretical EXAFS signals were simulated using an approach similar to that reported previously.[98] The Au L3-edge EXAFS spectra were calculated from the MD trajectories by averaging the signal arising from each Au atom in the particle. Each MD simulation was allowed to thermalize for at least 4 ps and the per-configuration EXAFS spectra were calculated from snapshots of the trajectory at intervals of 20 fs for at least 4 ps giving at least 200 independent configurations in the canonical average. All neighboring atoms up to 6 Å away from each photo-absorbing atom were included in the scattering calculations. Once the theoretical EXAFS signal was produced, the correction to the energy origin that was found for the experimental EXAFS (4.11 eV) was applied to the theoretical data to align the experimental and theoretical data in k-space.

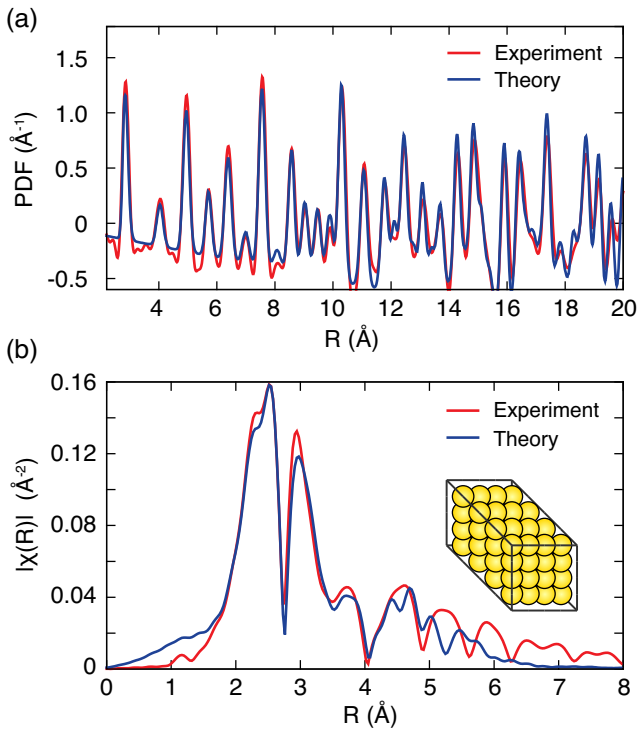

Figure 5.2: Comparison of bulk Au (a) DFT-PDF and (b) DFT-EXAFS to the corresponding experimental data.

The Au-Au PDFs were also produced from similarly sampled snapshots of the DFT-MD trajectories. Theoretical values for Au-Au coordination number, bond length, and bond disorder were calculated by considering all Au-Au bonding pairs shorter than 3.3 Å.

5.4 Results and discussion

5.4.1 Au bulk

Synchrotron based techniques, including XRD-PDF and EXAFS, are adept for determining the structure of bulk materials. We demonstrate this

Table 5.1: Derived structural properties of bulk Au including the average bond length R, coordination number N, and Debye-Waller factor σ^2.

Method	R (Å)	σ^2 (10^{-3} Å^2)
PDF expt.	2.854	9.8
PDF theory	2.859	7.9
EXAFS expt.	2.875(24)	9.1(3)
EXAFS theory	2.859(6)	8.2(8)

using a bulk Au system. Figure 5.2 shows the excellent agreement between (a) the simulated pair distribution function (PDF) and as measured with XRD-PDF and (b) the EXAFS spectra as simulated from a DFT-MD trajectory and measured.

Table 5.1 shows good agreement between the different methods in the physical values of bulk Au derived from the data.

5.4.2 $\mathbf{Au_{147}@S_n}$ nanoparticles

$Au_{147}@S_n$ nanoparticles were synthesized with a dendrimer template technique, characterized using both XRD-PDF and EXAFS as described in the experimental details section. Modeling was done with DFT-MD, from which the PDF and EXAFS spectra were calculated for comparison with experiment.

Figure 5.3 shows experimental XRD-PDF data of the particles with that of our MD-DFT calculations of an icosahedral particle. The icosahedral shape was found to be a low-energy structure of the Au particle in vacuum according to our DFT calculations. In order to make appropriate comparison to experiment, we fit an envelope function (grey) to capture the distribution of particles in the experiment. Interestingly, the first few peaks show good agreement, but the overall structure over the length-scale of a nanometer

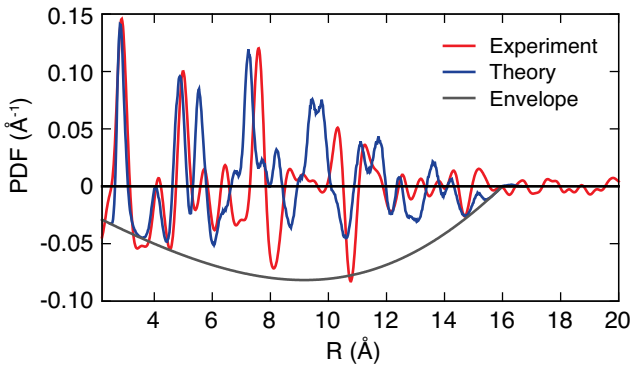

Figure 5.3: Measured and calculated pair distribution function (PDF) for a Au_{147} nanoparticle.

does not agree. Comparison of the XRD-PDF data with the bulk spectra in Fig. 5.2(a) shows that the nanoparticle structure is more consistent with being FCC rather than the icosahedral structure favored by DFT.

Ideally, we would be able to consider an FCC particle instead of icosahedral, but unfortunately as described by DFT, Au FCC cuboctahedral and truncated octahedral structures are unstable and revert to an icosahedral-like structure on a time scale of picosecond under room temperature MD. Fortunately, it is the first-neighbor peak which is important for EXAFS and this is adequately reproduced by the icosahedral model.

Two other points can be made about working with a theoretical model which may not be in agreement with experiment. First, we are interested in the range of structural models which are consistent with EXAFS data, and so finding consistency between a reasonable yet possibly incorrect model is interesting in and of itself. Second, the central result in this paper, which is that a standard Gaussian EXAFS analysis of a nanoparticle can result in a parameters which are biased towards the ordered bulk, is based upon our self-

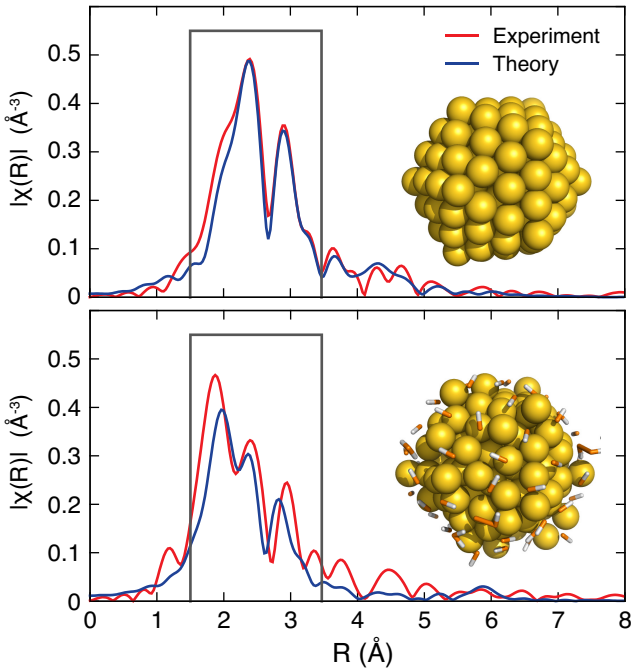

Figure 5.4: Comparison between theoretically simulated and experimentally measured EXAFS spectra of bare Au_{147} and $Au_{147}@S_{72}$ with 72 thiol ligands bound to the nanoparticle surface.

consistent theoretical approach and not upon a comparison between experiment and theory.

The simulated EXAFS spectra from the Au_{147} particle is shown in Fig. 5.4(a). Within the radial window of the first-neighbor shell, there is remarkable agreement with the experimental EXAFS data, especially given that there is no fitting involved. The agreement is consistent with the similar first-neighbor peak found in the MD-PDF and XRD-PDF data sets.

We can view the Au_{147} nanoparticle as likely having a somewhat more disordered structure as compared to bulk Au. To increase this disorder, we have introduced thiol ligands to the surface. Figure 5.4(b) again shows a consistent

Table 5.2: Fitted parameters to a Gaussian and 3rd order cumulant expansion first-neighbor models of both experimental and theoretical (from DFT-MD) EXAFS spectra of Au_{147} and $Au_{147}@S_{72}$.

System	N	R (Å)	σ^2 (10^{-3} Å^2)	C_3 (10^{-4} Å^3)
Au_{147} expt.	8.7(11)	2.816(5)	11.8(11)	–
Au_{147} theory	8.6(22)	2.829(10)	12.5(24)	–
Au_{147} expt.	9.0(10)	2.838(10)	12.0(10)	5.3(21)
Au_{147} theory	8.8(9)	2.859(9)	12.5(10)	7.3(21)
Au_{147} DFT-MD	9.38	2.871	17.3	11.8
$Au_{147}@S_{72}$ expt.	6.0(11)	2.825(6)	13.2(17)	–
$Au_{147}@S_{72}$ theory	4.3(11)	2.812(9)	12.0(23)	–
$Au_{147}@S_{72}$ expt.	6.0(10)	2.811(14)	13.2(15)	−3.5(31)
$Au_{147}@S_{72}$ theory	5.3(6)	2.878(8)	12.4(8)	13.5(19)
$Au_{147}@S_{72}$ DFT-MD	7.58	2.901	26.8	20.8

EXAFS spectra from our DFT-MD model (inset) and the experiment. Both of these spectra can be fit using a standard approach of assuming a Gaussian distribution of first-neighbor bond lengths. The resulting fits look quite good and the parameters derived from the experimental EXAFS spectra and the theoretical EXAFS spectra simulated from the DFT-MD data are reasonably consistent. The specific values of the coordination number N, the bond length R, and Debye-Waller factor σ^2 are reported in Table 5.2.

While there is reasonable agreement between the experimental EXAFS spectra and the one based upon our DFT model, this is not enough to say that the model is correct. In fact there is a wide range of structural models which are consistent with a nanoparticle EXAFS spectra. Furthermore, we will argue next that a standard EXAFS fitting model can have inherent and undetectable bias towards the model that is used for the fit.

The way we can quantify bias in the EXAFS fit is by our self-consistent

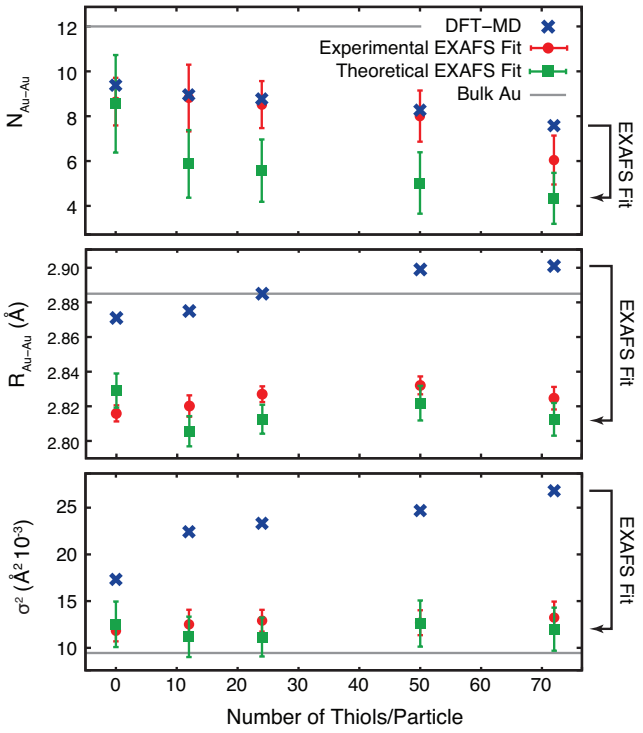

Figure 5.5: DFT-MD and Gaussian EXAFS fit parameters characterizing the first-neighbor shell of Au in Au_{147} and $Au_{147}@S_n$.

analysis presented in Fig. 5.1. First, the DFT-MD trajectories are analyzed in terms of their first-neighbor distributions to extract average values of N, R, and σ^2. These values are the reference that a subsequent EXAFS analysis should reproduce. Second, FEFF is used to generate an EXAFS spectrum from the DFT-MD trajectory, and this spectrum is fit using a Gaussian model to give values of N, R, and σ^2 which can be compared to the DFT-MD standard.

Figure 5.5 shows the EXAFS Gaussian fit parameters in comparison to those obtained by direct MD-DFT. What is remarkable, is that DFT-MD trajectory shows significantly larger values of N, R, and σ^2 as compared to those based upon a fit to the EXAFS spectra generated from the same trajectory data.

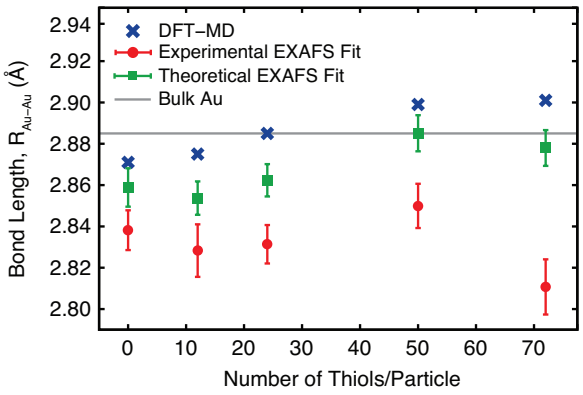

Figure 5.6: DFT-MD and 3rd cumulant EXAFS mean bond lengths in the first-neighbor shell of Au in Au_{147} and $Au_{147}@S_n$.

The standard EXAFS analysis indicates that the particles are smaller and more ordered than they really are. Also, remarkably, the fitted parameters from the DFT-MD simulated EXAFS are much closer to those seen in experiment, as compared to those taken directly from the DFT-MD data. This indicates that a standard fit to the experimental EXAFS data of these nanoparticle can give structural parameters that are unknowingly many standard deviations away from the actual values.

In order to account for the large static disorder present in the nanoparticle system, a third order cumulant expansion fitting model for the Au-Au first-shell was also considered. The third cumulant (C_3) represents how skewed, from a Gaussian distribution, the bond length distribution is. Table 5.2 compares N, R, σ^2, and C_3 for both fitting models for the theoretical and experimental Au_{147} and $Au_{147}@S_{72}$ particles. The reference DFT-MD values show that there is significant skew towards longer bond lengths (positive C_3) in the bare Au_{147} particle and even larger skew for $Au_{147}@S_{72}$. The primary improvement provided by this additional degree of freedom in the fitting model

is an enhanced estimate of the bond length. These estimates are plotted for all thiol concentrations in Figure 5.6. The bond length estimates in the fit to the theoretical data are quite good and show the correct trend.

It is relevant to point out that the theoretical fit has one "advantage" over the experimental fit in regards to determining the bond length. In the theoretical fit, the correction to the energy origin is exact, since it was fixed to the value used to generate the EXAFS data. This was done because when ΔE_0 was allowed to float in the fit, it was determined to be $-4\,\mathrm{eV}$ below the known value and thus the mean bond length was significantly underestimated. To be consistent, the experimental fit also had a fixed energy origin during the fit, however, the effect of having it float during the fitting process was insignificant. Since ΔE_0 is highly correlated with the bond length in the fit, using the correct value (as opposed to letting it float in the fit) increases the accuracy of the predicted bond length for the theoretical data.

While the bond length estimates improve significantly with the use of the 3rd order cumulant expansion model, the estimates for σ^2 and N did not improve substantially across the dataset. This means that if only EXAFS fitting data were used to analyze the $Au_{147}@S_n$ series one would reach a qualitatively incorrect conclusion: that the addition of thiol did not increase the disorder and decreased the particle size (coordination number). However, the DFT-MD data shows a 55% increase in disorder from Au_{147} to $Au_{147}@S_{72}$ and a small decrease in coordination number due only to extreme lengthening of the Au-Au bonds beyond the 3.3 Å "first-neighbor" cutoff distance.

The range of models which are consistent with the same EXAFS data can be illustrated by comparing the first-neighbor bond length distribution from DFT-MD and from the parameters fit to the simulated EXAFS spectra

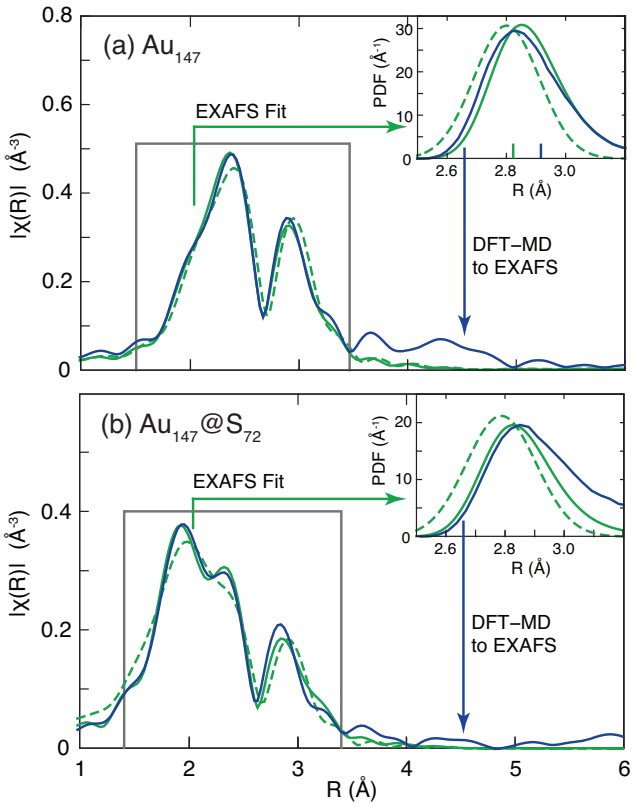

Figure 5.7: Fits to the theoretical R-space EXAFS as well as the resulting bond length distributions are compared to the results from DFT-MD.

based upon the same DFT-MD data. Figure 5.7 shows this comparison for Au_{147} and $Au_{147}@S_n$. The contrasting fit parameters from Fig. 5.5 are reflected in the distributions, with the Gaussian EXAFS analysis missing the fat tails at both short but especially long bond lengths seen in the DFT-MD simulations, and instead giving a more ordered, narrow distribution with a shorter average bond length.

5.5 Conclusion

In conclusion, we have presented a self-consistent method for measuring the error in EXAFS fitting models and applied it to the study of disordered $Au_{147}@S_n$ nanoparticles. We show through the use of DFT-MD simulations, that EXAFS fitting models can significantly underestimate particle size and disorder. Since the simulated EXAFS compares well with the experimental $Au_{147}@S_n$ DENs, it is reasonable to believe that physical systems exist with this level high level of disorder. Thus the Gaussian fitting model should be used with caution when fitting nanoparticle data and trends in disorder and coordination number should be carefully compared with other sources of data such as DFT simulations in order to interpret the results of the fit.

5.6 Acknowledgments

We gratefully acknowledge support from the U.S. Department of Energy under Contract DE-FG02-09ER16090 and the Robert A. Welch Foundation (Grants F-0032 and F-1841). The computational work was done at the National Energy Research Scientific Computing Center and the Texas Advanced Computing Center.

Bibliography

[1] A. F. Voter. Parallel replica method for dynamics of infrequent events. *Phys. Rev. B*, 57:R13985–R13988, 1998.

[2] A. F. Voter. Hyperdynamics: Accelerated molecular dynamics of infrequent events. *Phys. Rev. Lett.*, 78:3908–3911, 1997.

[3] H. Eyring. The activated complex in chemical reactions. *J. Chem. Phys.*, 3:107–115, 1935.

[4] R. A. Miron and K. A. Fichthorn. Accelerated molecular dynamics with the bond-boost method. *J. Chem. Phys.*, 119:6210–6216, 2003.

[5] G. Henkelman and H. Jónsson. Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *J. Chem. Phys.*, 115:9657–9666, 2001.

[6] D. J. Wales and J. P. K. Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *J. Phys. Chem. A*, 101:5111–5116, 1997.

[7] S. Goedecker. Minima hopping: An efficient search method for the global minimum of the potential energy surface of complex molecular systems. *J. Chem. Phys.*, 120:9911–9917, 2004.

[8] David P. Anderson. Boinc: A system for public-resource computing and storage. In *5th IEEE/ACM International Workshop on Grid Computing*, pages 4–10, 2004.

[9] Xing Cai, Hans Petter Langtangen, and Halvard Moe. On the performance of the python programming language for serial and parallel scientific computations. *Scientific Programming*, 13:31–56, 2005.

[10] H. Jónsson, G. Mills, and K. W. Jacobsen. Nudged elastic band method for finding minimum energy paths of transitions. In B. J. Berne, G. Ciccotti, and D. F. Coker, editors, *Classical and Quantum Dynamics in Condensed Phase Simulations*, pages 385–404. World Scientific, Singapore, 1998.

[11] E. Bitzek, P. Koskinen, F. Fähler, M. Moseler, and P. Gumbsch. Structural relaxation made simple. *Phys. Rev. Lett.*, 97:170201–1–4, 2006.

[12] M. R. Hestenes and E. Steifel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436, 1952.

[13] J. Nocedal. Updating quasi-Newton matrices with limited storage. *Math. Comput.*, 35:773–782, 1980.

[14] S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comp. Phys.*, 117:1–19, 1995.

[15] G. Kresse and J. Hafner. Ab initio molecular dynamics for liquid metals. *Phys. Rev. B*, 47:R558–R561, 1993.

[16] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen. Real-space grid implementation of the projector augmented wave method. *Phys. Rev. B*, 71:035109, 2005.

[17] M. S. Daw and M. I. Baskes. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B*, 29:6443–6453, 1984.

[18] A. F. Voter and S. P. Chen. Accurate interatomic potentials for Ni, Al and Ni_3Al. *Mat. Res. Soc. Symp. Proc.*, 82:175–180, 1987.

[19] K. Jacobsen, P. Stoltze, and J. K. Nørskov. A semi-empirical effective medium theory for metals and alloys. *Surf. Sci.*, 366:394–402, 1996.

[20] T. J. Lenosky, B. Sadigh, E. Alonso, V. V. Bulatov, T. Diaz de la Rubia, J. Kim, A. F. Voter, and J. D. Kress. Highly optimized empirical potential model of silicon. *Modelling Simul. Mater. Sci. Eng.*, 8:825, 2000.

[21] J. Tersoff. Empirical interatomic potential for silicon with improved elastic properties. *Phys. Rev. B*, 38:9902, 1988.

[22] J. F. Justo, M. Z. Bazant, E. Kaxiras, V. V. Bulatov, and S. Yip. Interatomic potential for silicon defects and disordered phases. *Phys. Rev. B*, 58:2539, 1998.

[23] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79:926–935, 1983.

[24] Vtsttools. http://theory.cm.utexas.edu/vtsttools, 2012.

[25] The EON distributed computing project can be found at http://eon.ices.utexas.edu/.

[26] Nordugrid. http://www.nordugrid.org/arc/, 2012.

[27] L. Zhang, S. T. Chill, and G. Henkelman. Distributed replica dynamics. in preparation, 2013.

[28] A. F. Voter. Introduction to the kinetic Monte Carlo method. In K. E. Sickafus, E. A. Kotomin, and B. P. Uberuaga, editors, *Radiation Effects*

in Solids, NATO Science Series II: Mathematics, Physics and Chemistry, pages 1–23. Springer Netherlands, 2007.

[29] G. Henkelman and H. Jónsson. A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives. *J. Chem. Phys.*, 111:7010–7022, 1999.

[30] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

[31] L. J. Munro and D. J. Wales. Defect migration in crystalline silicon. *Phys. Rev. B*, 59:3969, 1999.

[32] R. Malek and N. Mousseau. Dynamics of Lennard-Jones clusters: A characterization of the activation-relaxation technique. *Phys. Rev. E*, 62:7723–7728, 2000.

[33] L. Xu and G. Henkelman. Adaptive kinetic Monte Carlo for first-principles accelerated dynamics. *J. Chem. Phys.*, 129:114104, 2008.

[34] M. A. Novotny. Monte carlo algorithms with absorbing markov chains: Fast local algorithms for slow dynamics. *Phys. Rev. Lett.*, 74:1–5, Jan 1995.

[35] Charles M. Grinstead and J. Laurie Snell. *Introduction to Probability*. AMS, 2nd edition, 2003.

[36] A. Pedersen, J. C. Berthet, and H. Jónsson. Simulated annealing with coarse graining and distributed computing. *Lect. Notes Comput. Sc.*, 7134:34–44, 2012.

[37] M. I. Mendelev, D. J. Sordelet, and M. J. Kramer. Using atomistic computer simulations to analyze x-ray diffraction data from metallic glasses. *J. Appl. Phys.*, 102:043501, 2007.

[38] C. P. Lawrence and J. L. Skinner. Flexible tip4p model for molecular dynamics simulation of liquid water. *Chem. Phys. Lett.*, 372:842–847, 2003.

[39] R. Terrell, M. Welborn, S. T. Chill, and G. Henkelman. Database of atomistic reaction mechanisms with application to kinetic monte carlo. *J. Chem. Phys.*, 137:015105, 2012.

[40] D. J. Wales and H. A. Scheraga. Global optimization of clusters, crystals, and biomolecules. *Science*, 285:1368–1372, 1999.

[41] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Oxford Science Publications, 1987.

[42] Ronald P White and Howard R Mayne. An investigation of two approaches to basin hopping minimization for atomic and molecular clusters. *Chemical Physics Letters*, 289(5 - 6):463 – 468, 1998.

[43] Masao Iwamatsu and Yutaka Okabe. Basin hopping with occasional jumping. *Chemical Physics Letters*, 399(4â“6):396 – 400, 2004.

[44] Michael Sicher, Stephan Mohr, and Stefan Goedecker. Efficient moves for global geometry optimization methods and their application to binary systems. *J. Chem. Phys.*, 134:044106, 2011.

[45] H. Jónsson. Simulation of surface processes. *Proc. Nat. Acad. Sci.*, 108:944–949, 2011.

[46] G. Henkelman and H. Jónsson. Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points. *J. Chem. Phys.*, 113:9978–9985, 2000.

[47] G. Henkelman, B. P. Uberuaga, and H. Jónsson. A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *J. Chem. Phys.*, 113:9901–9904, 2000.

[48] M. R. Sørensen and A. F. Voter. Temperature-accelerated dynamics for simulation of infrequent events. *J. Chem. Phys.*, 112:9599–9606, 2000.

[49] V. J. Bhute and A. Chatterjee. Accuracy of a Markov state model generated by searching for basin escape pathways. *J. Chem. Phys.*, 138:084103, 2013.

[50] V. J. Bhute and A. Chatterjee. Building a kinetic Monte Carlo model with a chosen accuracy. *J. Chem. Phys.*, 138:244112, 2013.

[51] S. T. Chill, M. Welborn, R. Terrell, L. Zhang, J.-C. Berthet, A. Pedersen, H. Jónsson, and G. Henkelman. Eon: Software for long time scale simulations of atomic scale systems. *Model. Simul. Mater. Sci. Eng.*, in press, 2012.

[52] Y. Fan, A. Kushima, S. Yip, and B. Yildiz. Mechanism of void nucleation and growth in bcc Fe: Atomistic simulations at experimental time scales. *Phys. Rev. Lett.*, 106:125501, 2011.

[53] P. Brommer and N. Mousseau. Comment on "mechanism of void nucleation and growth in bcc fe: Atomistic simulations at experimental time scales". *Phys. Rev. Lett.*, 108:219601, 2012.

[54] G. J. Ackland, M. I. Mendelev, D. J. Srolovitz, S. Han, and A. V. Barashev. Development of an interatomic potential for phosphorus impurities in α-iron. *J. Phys.: Condens. Matter*, 16:S2629, 2004.

[55] David Aristoff and Tony Lelièvre. Mathematical analysis of temperature accelerated dynamics. *Multiscale Model. Simul.*, 12(1):290–317, 2014.

[56] D. W. Bassett and P. R. Webber. Diffusion of single adatoms of platinum, iridium and gold on platinum surfaces. *Surf. Sci.*, 70:520, 1978.

[57] G. H. Vineyard. Frequency factors and isotope effects in solid state rate processes. *J. Phys. Chem. Solids*, 3:121–127, 1957.

[58] M. A. Novotny. Monte Carlo algorithms with absorbing Markov chains: Fast local algorithms for slow dynamics. *Phys. Rev. Lett.*, 74:1–5, 1995.

[59] J. N. Murrell and K. J. Laidler. Symmetries of activated complexes. *Trans. Faraday. Soc.*, 64:371–377, 1968.

[60] J. E. Jones and A. E. Ingham. On the calculation of certain crystal potential constants, and on the cubic crystal of least potential energy. *Proc. R. Soc. A*, 107:636–653, 1925.

[61] P. M. Morse. Diatomic molecules according to the wave mechanics. ii. vibrational levels. *Phys. Rev.*, 34:57, 1929.

[62] https://github.com/pele-python/pele, 2013.

[63] http://www-wales.ch.cam.ac.uk/OPTIM/, 2013.

[64] http://www-wales.ch.cam.ac.uk/GMIN/, 2013.

[65] http://theory.cm.utexas.edu/eon/, 2013.

[66] https://wiki.fysik.dtu.dk/ase/, 2013.

[67] http://www.scipy.org, 2013.

[68] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C: The art of scientific computation.* Cambridge University Press, Cambridge, 2nd edition, 1992.

[69] S. A. Trygubenko and D. J. Wales. A doubly nudged elastic band method for finding transition states. *J. Chem. Phys.*, 120:2082–2094, 2004.

[70] Y. Kumeda, D. J. Wales, and L. J. Munro. Transition states and rearrangement mechanisms from hybrid eigenvector-following and density functional theory. application to $C_{10}H_{10}$ and defect migration in crystalline silicon. *Chem. Phys. Lett.*, 341:185–194, 2001.

[71] A. Heyden, A. T. Bell, and F. J. Keil. Efficient methods for finding transition states in chemical reactions: Comparison of improved dimer method and partitioned rational function optimization method. *J. Chem. Phys.*, 123:224101–1–14, 2005.

[72] J. Kästner and P. Sherwood. Superlinearly converging dimer method for transition state search. *J. Chem. Phys.*, 128:014106–1–6, 2008.

[73] Lesley D. Lloyd, Roy L. Johnston, and Said Salhi. Strategies for increasing the efficiency of a genetic algorithm for the structural optimization of nanoalloy clusters. *J. Comput. Chem.*, 26(10):1069–1078, 2005.

[74] Daniel Asenjo, Jacob D. Stevenson, David J. Wales, and Daan Frenkel. Visualizing basins of attraction for different minimization algorithms. *J. Phys. Chem. B*, 2013.

[75] E. Polak and G. Ribiere. Note sur la convergence de methodes de directions conjuguées. *Rev. Franç. Informat. Rech. Operationnelle*, 16:35–43, 1969.

[76] M. J. D. Powell. Restart procedures for the conjugate-gradient method. *Math. Program.*, 12:241–254, 1977.

[77] István Kolossváry and Kevin J. Bowers. Global optimization of additive potential energy functions: Predicting binary lennard-jones clusters. *Phys. Rev. E*, 82:056711, Nov 2010.

[78] D. J. Wales and J. P. K. Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *J. Phys. Chem. A*, 101(28):5111–5116, 1997.

[79] Mark T. Oakley, Roy L. Johnston, and David J. Wales. Symmetrisation schemes for global optimisation of atomic clusters. *Phys. Chem. Chem. Phys.*, 15:3965–3976, 2013.

[80] D. Schebarchov and D. J. Wales. Communication: A new paradigm for structure prediction in multicomponent systems. *J. Chem. Phys.*, 139(22):–, 2013.

[81] B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *Bell Syst. Tech. J.*, 49(2):291–307, 1970.

[82] M. Page and J. W. McIver. on evaluating the reaction path hamiltonian. *J. Chem. Phys.*, 88:922, 1988.

[83] Yi Zeng, Penghao Xiao, and Graeme Henkelman. Unification of algorithms for minimum mode optimization. *J. Chem. Phys.*, 140(4):–, 2014.

[84] D. J. Wales. Basins of attraction for stationary-points on a potential-energy surface. *J. Chem. Soc. Faraday Trans.*, 88:653–657, 1992.

[85] D. J. Wales. Locating stationary-points for clusters in cartesian coordinates. *J. Chem. Soc. Faraday Trans.*, 89:1305–1313, 1993.

[86] J. Mustre de Leon, J. J. Rehr, S. I. Zabinsky, and R. C. Albers. Ab initio curved-wave x-ray-absorption fine structure. *Phys. Rev. B*, 44:4146–4156, Sep 1991.

[87] A. I. Frenkel and J. J. Rehr. Thermal expansion and x-ray-absorption fine-structure cumulants. *Phys. Rev. B*, 48:585–588, Jul 1993.

[88] Eveline Bus, Jeffrey T. Miller, A. Jeremy Kropf, Roel Prins, and Jeroen A. van Bokhoven. Analysis of in situ exafs data of supported metal catalysts using the third and fourth cumulant. *Phys. Chem. Chem. Phys.*, 8:3248–3258, 2006.

[89] David F. Yancey, Liang Zhang, Richard M. Crooks, and Graeme Henkelman. Au@pt dendrimer encapsulated nanoparticles as model electrocatalysts for comparison of experiment and theory. *Chem. Sci.*, 3:1033–1040, 2012.

[90] David F. Yancey, Samuel T. Chill, Liang Zhang, Anatoly I. Frenkel, Graeme Henkelman, and Richard M. Crooks. A theoretical and experimental examination of systematic ligand-induced disorder in au dendrimer-encapsulated nanoparticles. *Chem. Sci.*, 4:2912–2921, 2013.

[91] Matthew Newville. EXAFS analysis using *FEFF* and *FEFFIT*. *Journal of Synchrotron Radiation*, 8(2):96–100, Mar 2001.

[92] B. Ravel and M. Newville. *ATHENA*, *ARTEMIS*, *HEPHAESTUS*: data analysis for X-ray absorption spectroscopy using *IFEFFIT*. *Journal of Synchrotron Radiation*, 12(4):537–541, Jul 2005.

[93] G. Kresse and J. Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Comput. Mater. Sci.*, 6:15–50, 1996.

[94] J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou, and K. Burke. Restoring the density-gradient expansion for exchange in solids and surfaces. *Phys. Rev. Lett.*, 100:136406, 2008.

[95] J. P. Perdew, K. Burke, and M. Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77:3865–3868, 1996.

[96] P. E. Blöchl. Projector augmented-wave method. *Phys. Rev. B*, 50:17953, 1994.

[97] G. Kresse and D. Joubert. From ultrasoft pseudopotentials to the projector augmented wave method. *Phys. Rev. B*, 59:1758, 1999.

[98] Otello Maria Roscioni, Nicholas Zonias, Stephen W. T. Price, Andrea E. Russell, Tatiana Comaschi, and Chris-Kriton Skylaris. Computational prediction of l3 exafs spectra of gold nanoparticles from classical molecular dynamics simulations. *Phys. Rev. B*, 83:115409, Mar 2011.