5-2019

Rejection Enhanced Off-Lattice Kinetic Monte Carlo

Hamza Mahmoud Ruzayqat
University of Tennessee, hruzayqa@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Hamza Mahmoud Ruzayqat entitled "Rejection Enhanced Off-Lattice Kinetic Monte Carlo." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mathematics.

Tim Schulze, Major Professor

We have read this dissertation and recommend its acceptance:

Ohannes Karakashian, Steven Wise, Haixuan Xu

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Rejection Enhanced Off-Lattice Kinetic Monte Carlo

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Hamza Mahmoud Ruzayqat

May 2019

This dissertation is dedicated to my parents, Mahmoud and Samira Ruzayqat, who have always loved and believed in me unconditionally and whose good examples have taught me to work hard for the things that I aspire to achieve. I am deeply grateful for their support, love and sacrifices. This work is dedicated to my beautiful wife, Arij, who has been a continual source of support, encouragement and inspiration during the challenges of graduate school and life and whom I am very happy to share life with. This work is also dedicated to my beloved daughter, Aya. I dedicate this work to my high school teachers especially my math teacher, Yassin Tarada, who was a real inspiration and whose talent is greatly responsible for my love of mathematics.

Acknowledgments

I would like to thank my committee chair and adviser Dr. Tim Schulze. Without his guidance and persistent help this dissertation would not have been possible. His support and encouragement are greatly appreciated. I would also like to thank Dr. Ohannes Karakashian, Dr. Steven Wise and Dr. Haixuan Xu for their help and their service on my committee. Thanks to Dr. Dmitri Schevarchov for many helpful discussions regarding this study. Special thanks goes to the National Science Foundation for their financial support through the NSF grant 1613729. Last and certainly not least I would like to thank my math department faculty for their teachings and support.

Abstract

We introduce a new kinetic Monte Carlo (KMC) algorithm for off-lattice simulation. In off-lattice KMC one needs to calculate the rates for all possible moves from the current state by searching the energy landscape for index-1 saddle points surrounding the current basin of attraction. We introduce a rejection scheme where the true rates are replaced by rate estimates. This is done by first associating each saddle point with a key atom defined to be the atom that moves the most or that corresponds to the largest energy change if the transition were to take a place, then constructing an estimate for the total rate associated with each atom by using a nearest-neighbor bond count. These estimates allow one to select a set of possible transitions, one of which is accepted or rejected based on a localized saddle point search focused on a particular atom. In principle, this allows a performance boost that scales with the number of particles in the system. We test the method on a growing two-species nanocluster with an emerging core-shell structure bound by Lennard-Jones potential. In addition to that, we give a detailed review for the dimer method used in this study to locate index-1 saddle points on the potential energy surface.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The overarching theme of the work presented here is the development of an enhanced rejection algorithm for off-lattice kinetic Monte Carlo (OLKMC) simulations. In this chapter, we breifly review the work that has been done in the area of OLKMC simulation and then give a detailed algorithm for the method. In chapter 2, we shed light on the dimer method [10, 13, 15, 34] for searching for index-1 saddle points on potential energy surfaces. In chapter 3, we introduce our rejection algorithm for OLKMC. In chapter 4, we demonstrate the validity of the new scheme by simulating the growth of a two-species cluster using the Lennard Jones potential. Finally, in chapter 5 we provide a discussion on the efficiency of the scheme and explore the prospects of developing this algorithm further down the line.

One of the essential goals in materials science and theoretical chemistry is to simulate the dynamical evolution of many systems at the atomic scale and be able to accurately calculate the time evolution of these dynamics. The interactions between atoms are usually obtained from solving the Schrödinger equation which describes electrons and nuclei in the system or from an interatomic potential function that depends on the atom's positions in space. The natural way to treat the motion of atoms in the system is to integrate the classical equations of motion for every atom in the system forward in time. This method is known as molecular dynamics (MD), a standard simulation technique. Most often, the classical equations of motion can easily be solved numerically, and therefore, if quantum dynamical effects are

ignored, the simulation of mechanical evolution is in principle very simple and is an accurate representation of what is going on in the real physical system. This is the main reason why the MD method is popular. However, an accurate integration of the classical equations of motion requires short time steps to account for atomic vibrations that happen at very high frequencies. For instance, a femtoseconds time step is needed to accurately simulate a displacement cascade that occurs in the first few picoseconds after a collision. This is an appropriate time step for atomic systems as it corresponds to the atomic vibrational frequency in the system. Accordingly, this limits the range of a simulation to nanoseconds, while important processes that occur after a cascade event often take place on a mesoscopic timescale.

The kinetic Monte Carlo (KMC) method overcomes this barrier and is often the method of choice when millisecond to hour time scales need to be accessed. KMC achieves this by exploiting the fact that the long-time dynamics are diffusive jumps that happen occasionally from state to state. The system spends most of its time randomly oscillating about a local minimum of its interatomic potential energy. The system then may surmount an energy barrier and transition from its current minimum state to another neighboring minimum state by passing through a saddle point of the potential energy (a transition state) that separates them (see Figure 1.1.) So, instead of following the systems trajectory, transitions between neighboring minima are directly performed. Each transition is chosen to be performed proportional to its transition rate.

KMC methods can be divided into two main categories: Lattice KMC and OLKMC. In the first category, the method is used to simulate systems with crystalline structures that assumes all atoms are sitting on a lattice, while OLKMC deals with systems that assume no restrictions on atom positions. In lattice-based KMC, lattice points and transition rates are defined at the beginning of the simulation. Then, during the simulation, energy barriers and diffusion rates are updated before every KMC event based on the local environments surrounding those atoms of interest. As a result, some important physical effects like lattice

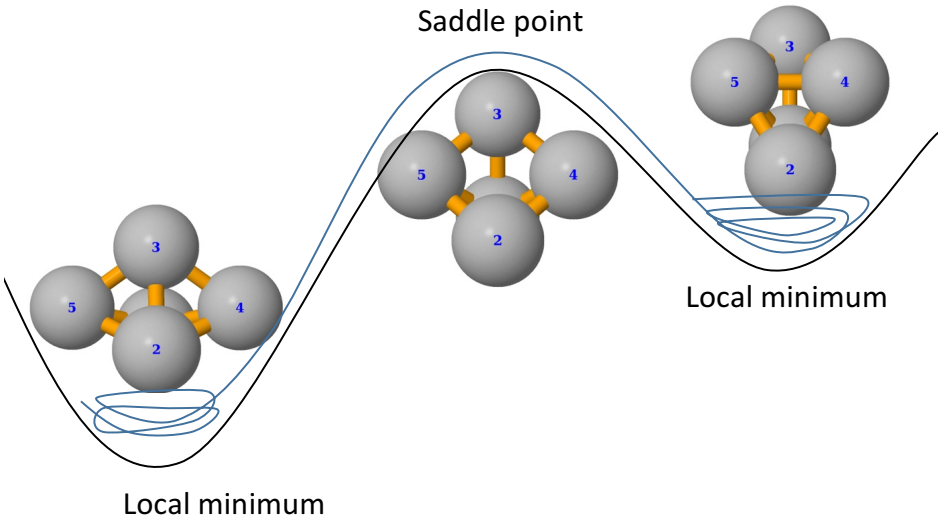


Figure 1.1: This figure is a 2d cartoon that shows two adjacent local minima and a saddle point that separates them for a system of five particles.

mismatch, which leads to elastic strain and dislocations (see Figure 1.2), are neglected in this model.

OLKMC, on the other hand, takes into account all these elastic strain effects. Several schemes have been used in the past to simulate epitaxial growth in 2D and 3D in the presence of strain. For example, Faux et al. [7, 35], Plotz et al. [28] and Much et al. [24] tried to simulate the early stages of strained-layer growth by molecular-beam epitaxy in '1+1' and '2+1' dimensions with the use of the Lennard-Jones interatomic potential. In another work [8], Guo et al. precomputed approximate saddle point locations based on unstrained lattice structures and used this approach to investigate impurity diffusion in a weakly strained FCC nanowire. Another approach to incorporating elastic effects into KMC can be found in [17, 20, 29, 30] where the rigid lattice of traditional KMC is replaced by a network of linear springs that are allowed to deform so as to minimize the system's potential energy. An important technique for accelerating this weakly off-lattice scheme was the development of a rejection-based algorithm that makes use of rate approximations that are similar to the models used in lattice-based simulation [31]. These weakly off-lattice approaches have been used to study the effect of lattice mismatch during heteroepitaxial film growth in [32, 33, 2].

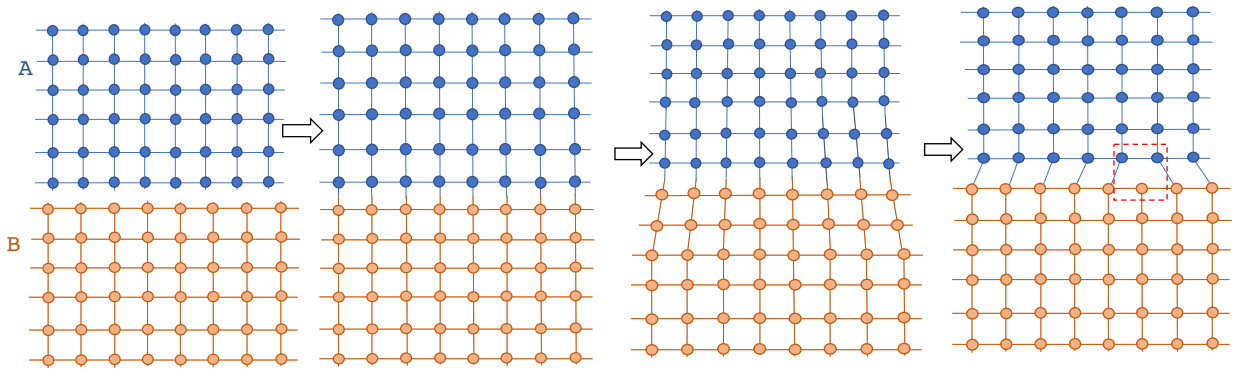


Figure 1.2: Illustration of two crystals A and B with different lattice constants resulting in elastic strain and dislocations near the interface between the two crystals.

However, this particular approach cannot capture effects due to large displacements, like the formation of dislocations and other lattice defects, nor effects due to the concerted movement of multiple atoms. None of the schemes cited above are considered fully off-lattice because of the restrictions assumed on atomic movement and most of them combine some aspects of lattice and off-lattice models together. For example, in [24] only a single atom is moved within the configuration space in order to locate approximate transition states on the potential energy surface. In [8], the model was restricted to transitions involving a single impurity atom diffusing in the interstices of a strained lattice.

Fully OLKMC was first proposed by Henkelman and Jónsson in [11]. In this method, the full list of saddle points surrounding the current basin of attraction must be found, allowing all atoms to move within the configuration space. These calculations need to be repeated before every KMC event. In that paper, the authors used the dimer method [10], which will be discussed in detail in the next chapter, to locate the surrounding saddle points. Fully general off-lattice simulations make use of either an empirical potential or an even more costly density functional theory calculation, seeking to exhaustively calculate the transition path to all of the neighboring states within the multi-particle configuration space. This makes a fully implemented off-lattice simulation an enormously complex task when compared to lattice based simulations, where rates can be precomputed and stored. So much so that

KMC simulation loses much of its utility and applications of these methods are limited to systems with only a few hundred atoms, simulated for much shorter times, and at much greater computational cost.

Several improvements have been proposed. Adaptive kinetic Monte Carlo [42] is a fully OLKMC method, which is very similar to that in [11], except one may recycle the previous visited transition states and reuse them. A dynamic stopping criterion is also provided to decide when enough saddle point searches have completed. Another approach is the so called off-lattice self-learning KMC [14, 26, 18], where a 2D in [14] and a 3D in [26, 18] off-lattice pattern-recognition schemes are used to study heteroepitaxial island diffusion. Local environment KMC [16] is a method for efficiently performing the previously mentioned off-lattice self-learning KMC. Another method that is mainly used to study defect evolution and interactions is the self-evolving atomistic KMC [40]. The idea behind this work is to freeze all atoms outside certain regions and search for saddle points using the dimer method with only atoms inside the region allowed to move. These are the regions where dynamical processes of interest may occur, and they are called active volumes. Active volumes are identified based on the presence of defects in the system. To enhance the efficiency of this scheme, the authors of [41] suggest using a multi-step procedure that begins with a relatively small active volume that is used to obtain an initial prediction of the transition states. Then, the active volume size is gradually increased to obtain more accurate values.

In the next section, we briefly review the components of a fully implemented, rejection-free OLKMC.

1.1 Off-Lattice Kinetic Monte Carlo

KMC simulation of crystal growth is motivated by observations of molecular dynamics simulations, relying on the transition state theory (TST) to provide an approximate model [36, 6, 11, 2]. The essential observation is that the system spends most of its time randomly oscillating within the N_p-particle, dN_p-dimensional configuration space about a

local minimizer $\mathbf{X}_i \in \mathbb{R}^{dN_p}$ of the system's potential energy, $U(\mathbf{X})$, with rare transitions between basins of attraction. For the system to move from basin i to basin j, it has to overcome a minimum energy barrier ΔU_{ij}. The harmonic approximation to TST [37, 38] estimates the rate R_{ij} at which the transition occurs as

$$R_{ij} = K \exp(-\Delta U_{ij}/k_B T), \tag{1.1}$$

where T is the temperature, k_B is Boltzmann's constant and K is defined below.

These observations give rise to an alternative model where the Newtonian dynamics is replaced by a Markov-chain, with the system making relatively rare, random transitions between states, represented by the local minima $\mathbf{X}_i$, at rates R_{ij} calculated from Eq. (1.1). More specifically the energy barrier

$$\Delta U_{ij} = U(\mathbf{X}_{ij}) - U(\mathbf{X}_i), \tag{1.2}$$

requires locating both the initial local minimum, $\mathbf{X}_i$, and the index-1 saddle point, $\mathbf{X}_{ij}$ (where $\nabla U = 0$ and all but one of the principal curvatures are positive), separating the basins of attraction. Note that these local minima and saddle points are, in principle, determined by the motion of all of the particles simultaneously within the configuration space. The prefactor K in the above equation Eq. (1.1) is defined as

$$K = \frac{\prod_{n=1}^{3N_p} \sqrt{\lambda_n^{(i)}}}{\prod_{n=1}^{3N_p-1} \sqrt{\lambda_n^{(ij)}}} \tag{1.3}$$

where $\left\{\lambda_n^{(i)}\right\}_{n=1}^{3N_p}$ and $\left\{\lambda_n^{(ij)}\right\}_{n=1}^{3N_p-1}$ are the eigenvalues of the hessian at $\mathbf{X}_i$, and $\mathbf{X}_{ij}$, respectively [9]. In our simulations we take K to be a constant scaled to one, a commonly used approximation.

After enumerating the full set of transition rates and relabeling them using a single index: $\{r_n \equiv R_{ij}^n\}$, a single iteration of an OLKMC simulation is described by:

Algorithm 1: Rejection-Free OLKMC

1. Calculate rates r_i for each transition accessible from the current configuration.
2. Calculate partial sums $P_n = \sum_{i=1}^{n} r_i$, $n = 1, 2, \ldots, N$.
3. Generate a uniformly distributed random number $r \in [0, P_N)$.
4. Locate interval I such that $P_{I-1} \leq r < P_I$.
5. Update the physical time $t \leftarrow t + \Delta t$ with $\Delta t = -\ln r'/P_N$, where r' is a uniformly distributed random number in $(0, 1]$.
6. Move the system to the designated transition state, perturb away from the current local minimum, and relax to the new configuration.

In Step 5, $\Delta t = -\ln r'/P_N$ is an exponentially distributed random number sampled from the probability distribution $f(t) = P_N e^{-P_N t}$ of the time between successive events (see Figure 1.3.) The expected value of time between events (the mean time) is $\overline{\Delta t} = \int_0^\infty t f(t) dt = 1/P_N$. Steps 1 and 6 are tremendously costly compared to lattice-based simulations. So much so that KMC loses much of its utility in that the system's size and the number of iterations that can be simulated are greatly reduced. This is particularly so with the first step, as the number of saddle points grows rapidly with the number of particles in the system. Saddle point searches are typically done with some sort of eigenvector climbing algorithm, with a computational cost similar to minimization using nonlinear conjugate gradient. In this project we implement a version of the Dimer method introduced by Henkelman and Jónsson [10, 13, 15, 27].

In principle, the model requires locating all saddle points connected to the current basin of attraction. In practice, there is no way of knowing for certain when this has been achieved. This error is, in some sense, controllable, in that one can increase the number of attempts at finding new saddle points until a point of diminishing returns is acheived. Some recent

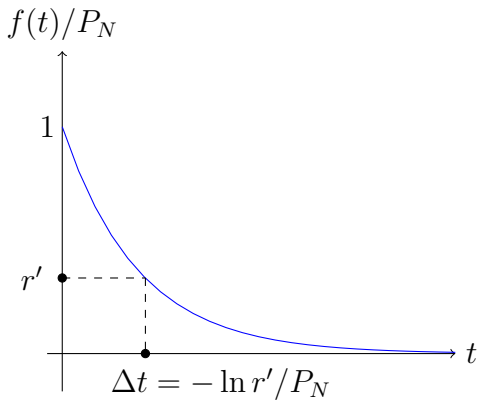


Figure 1.3: r' is a uniformly distributed random number generated in the interval (0,1] on the y-axis. The point on the t-axis that corresponds to r' is $\Delta t = -\ln r'/P_N$.

work [23] seeks to make a more exhaustive exploration of the local potential energy surface, but, as with so much of the work on OLKMC, seems to be limited to very small systems for the time being. Here, we follow a practice similar to that of Henkelman and Jónsson [11] and initiate a large number of searches by randomly perturbing the system about the current state. More specifically, let $\mathbf{X} = \{\boldsymbol{x}_i \in \mathbb{R}^3\}_{i=1}^{N_p}$. For each atom j, we perturb the entire system with a magnitude that decays with increasing distance from this particular atom:

$$\boldsymbol{x}_i^k = \boldsymbol{x}_i + \frac{k\sigma}{N_g}\hat{\boldsymbol{n}}_i^k \exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2\right) \qquad \text{for all } i = 1, \cdots, N_p, \tag{1.4}$$

where N_g is the number of initial guesses, $k = 1, \cdots, N_g$ is the guess number, $\hat{\boldsymbol{n}}_i^k$ is a random unit vector in $\mathbb{R}^3$ and σ is a length scale parameter that will be discussed later. This reflects the fact that most configuration changes are localized about a single particle or a small group of particles.

This procedure will find some saddle points that do not connect to the system's initial basin of attraction. Thus, upon finding a saddle point, one must requench the system starting from the newly found saddle point to ensure that the resulting saddle point is *connected*. One must also scan the list of previously acquired connected saddle points to prevent duplicates.

For the final step, we initialize a nonlinear conjugate-gradient minimization scheme near the chosen saddle point. We perturb this initial condition slightly in the direction away from

the initial configuration and monitor the progress with a strict descent requirement and maximum displacement threshold, with the aim of guiding the system into the neighboring basin of attraction.

Another difficulty with general OLKMC procedures is what is known as the "small barrier problem" [22]. Occasionally the system will reside in a basin of attraction with one or more extremely shallow minima, the crossing of which has little impact on the configuration. The small barrier means that the rate will be extremely high, and it is highly probable for the events associated with shallow barriers to be selected. When the reverse process also has a shallow barrier, this can lead to many wasted iterations as the system makes insignificant oscillations before a transition that fundamentally changes the configuration finally occurs. For this reason, we implement our OLKMC with a minimum barrier size chosen to reflect barriers that are typical for the surface motion of a single, loosely bonded atom, *e.g.* what one would refer to as an "adatom" in the context of epitaxial growth.

Chapter 2

Dimer Method

2.1 Introduction

In chemistry, critical points play a key role for the characterization of potential energy surfaces. In most cases, the determination of local minima is considered routine work where as the determination of saddle points is significantly more complex and is a nontrivial task even for small systems that consist of few particles. Several saddle point finding methods have been proposed, and they fall into two main categories: The first requires the knowledge of both the initial and final states of the system in order to find the connecting saddle point, and hence, the minimum energy path. The second category requires only the knowledge of the initial state of the system. For OLKMC, one wants to employ a single-ended method that starts the search from the current basin of attraction and finds all nearby saddle points.

Examples in the first category are the nudged-elastic band [12] and the string [4] methods. Examples of single-ended methods include the dimer method, which was presented by Henkelman and Jónsson in [10], eigenvector-following methods [25, 39], the Lanczos algorithm of the activation-relaxation technique which is known as "ART nouveau method" [21], and the biased gradient squared descent method [3]. However, comparisons show that the dimer method remains competitive especially when the system's size is larger [27].

The dimer method is Hessian-free, and so one only needs to evaluate the potential energy and its first derivatives. Therefore, the method is suitable for large clusters of particles. Moreover, the method can be efficiently parallelized to work on several processors. Modified versions of this method were proposed in [13, 15, 34]. These modifications will be described in detail in the next sections.

In this study, saddle points of interest are those of 1^{st} order (also called index 1 saddle points). Such points correspond to short lived transition states at which the system's potential energy is minimum in all directions but one. Also in this study, the negative gradient of the potential energy at a certain point is referred to as force at that point.

These rapid transition events, that happen between two long lived minimum states, play a fundamental role in understanding the dynamics of many systems in chemistry and physics, especially when methods like KMC are used to make a coarse description of such dynamics through certain stochastic simulations. Therefore, locating such states and computing their rates have attracted scientists' attention for many years.

2.2 Overview of Dimer Algortihm

Consider a system of N_p particles interacting through an empirical potential energy $U(\mathbf{X})$, where $\mathbf{X} = \{\boldsymbol{x}_i \in \mathbb{R}^3\}_{i=1}^{N_p}$ is a vector in $\mathbb{R}^{3N_p}$ and $\boldsymbol{x}_i = (x_{3i-2}, x_{3i-1}, x_{3i})$ are the corresponding Cartesian coordinates of particle i in three-dimensional space. The dimer method mainly uses finite difference approximations and Newton's method to move the system from an initial state on the potential energy surface U uphill toward an index 1 saddle point. The dimer consists of three points $\mathbf{X}_0$, $\mathbf{X}_1$ & $\mathbf{X}_2 \in \mathbb{R}^{3N_p}$ that lay on the same line, where $\mathbf{X}_0$ is its center and $\mathbf{X}_1$ & $\mathbf{X}_2$ are its endpoints. The distance between both $\mathbf{X}_1$, $\mathbf{X}_2$ and the dimer's center is ℓ, a short fixed distance. One can think of the dimer as a stencil in the sense that information at two points can be used to approximate the potential and gradient at the third point. Assuming the dimer is oriented in the direction of a unit vector $\mathcal{V} \in \mathbb{R}^{3N_p}$

(see Figure 2.1,) then the dimer's endpoints are given by

$$\mathbf{X}_1 = \mathbf{X}_0 + \ell\mathcal{V} \qquad \& \qquad \mathbf{X}_2 = \mathbf{X}_0 - \ell\mathcal{V}. \tag{2.1}$$

Initially, $\mathcal{V}$ and $\mathbf{X}_0$ will be chosen randomly, but later on during the search both vectors are repeatedly updated by performing the following two steps:

i. **Dimer Rotation**: In this step we search for the lowest curvature direction, $\mathcal{V}$, of the potential energy surface at $\mathbf{X}_0$, which corresponds to the lowest eigenvalue, $C_{\mathcal{V}}$, of the Hessian matrix $\mathbb{H} = [\partial^2 U(\mathbf{X})/\partial x_i \partial x_j]_{i,j=1}^{3N_p}$ at $\mathbf{X}_0$. After rotation, the eigen-pair $(\mathcal{V}, C_{\mathcal{V}})$ satisfies the following equation

$$\mathbb{H}(\mathbf{X}_0)\mathcal{V} = C_{\mathcal{V}}\mathcal{V}, \quad C_{\mathcal{V}} = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{3N_p}, \tag{2.2}$$

where $\{\lambda_i\}_{i=1}^{3N_p}$ is the set of eigenvalues of $\mathbb{H}(\mathbf{X}_0)$. In this step, we fix the dimer's midpoint $\mathbf{X}_0$ and rotate the dimer's axis in $\mathbb{R}^{3N_p}$ in order to find the direction at which the second derivative of the potential energy along the dimer's axis is minimized. This rotation will be in the direction of a net rotational force at one of the dimer's endpoints.

ii. **Dimer Translation**: Here the dimer's midpoint is not fixed anymore, however, its axis orientation is fixed in the direction of $\mathcal{V}$ that was found in the previous step. In this step, we translate the dimer a certain step length in a direction where the potential is maximum in the lowest curvature direction $\mathcal{V}$ and minimum in all other principal directions. One possible way is to translate the dimer in the direction of the Householder reflector of the force at $\mathbf{X}_0$.

Rotation and translation steps must be repeated in a certain way until the dimer's center $\mathbf{X}_0$ reaches an index 1 saddle point.

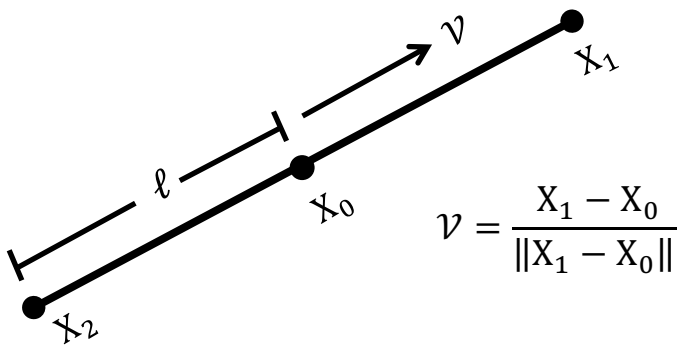


Figure 2.1: Definition of the various points of the dimer in the $3N_p$-dimensional space.

Denote by E_0, E_1 and E_2 the potential energy at $\mathbf{X}_0$, $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively. The dimer's energy is defined as

$$E \coloneqq E_1 + E_2. \tag{2.3}$$

The forces at the dimer's center and endpoints are $\mathbf{F}_0 = -\nabla U(\mathbf{X})|_{\mathbf{X}_0}$, $\mathbf{F}_1 = -\nabla U(\mathbf{X})|_{\mathbf{X}_1}$ and $\mathbf{F}_2 = -\nabla U(\mathbf{X})|_{\mathbf{X}_2}$ (see Figure 2.2.) However, in order to minimize the total number of energy and force evaluations needed for the whole search, one can make some interpolations. For instance, the force acting on $\mathbf{X}_0$ can be approximated as the average of the forces at the dimer's endpoints: $\mathbf{F}_0 = (\mathbf{F}_1 + \mathbf{F}_2)/2$ [10].

This approximation is good for some potential energy surfaces where evaluating the energy and the gradient can be calculated with high accuracy. But, in some cases such as quantum-chemical potential energies where evaluating energies and gradients cannot be done efficiently due to the numerical noise, it is better to evaluate the force at one of the dimer's endpoints and at its center then be approximated by a linear interpolation at the other endpoint. For such potentials, the system is converged to a saddle point within at least 3×10^{-4} bohr ($\sim 1.588 \times 10^{-4}$ Å) [13, 27], and therefore, to have this accuracy the force has to be calculated at the dimer's midpoint and one of the endpoints then approximated at the other endpoint. In our study we approximate the force at endpoint 2 and take it to be exact at the midpoint

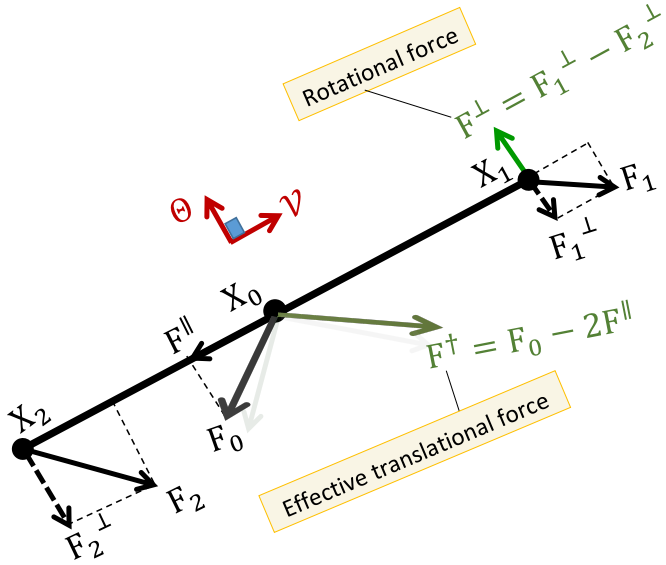


Figure 2.2: Definition of the various forces acting on the dimer's points.

and endpoint 1:

$$\mathbf{F}_2 \approx 2\mathbf{F}_0 - \mathbf{F}_1. \tag{2.4}$$

Taking the dot product with $\mathcal{V}$ in Eq. (2.2), we have

$$C_{\mathcal{V}} = \mathcal{V}^T \mathbb{H}(\mathbf{X}_0)\mathcal{V} = D^2_{\mathcal{V}} U(\mathbf{X})|_{\mathbf{X}_0} = -D_{\mathcal{V}}\mathbf{F}(\mathbf{X})|_{\mathbf{X}_0}, \tag{2.5}$$

where $D_{\mathcal{V}}$ and $D^2_{\mathcal{V}}$ are the first and second directional derivative of $U(\mathbf{X})$ in the direction of $\mathcal{V}$. Using the central finite difference formula

$$\begin{aligned} D_{\mathcal{V}}\mathbf{F}(\mathbf{X})|_{\mathbf{X}_0} &\approx \frac{1}{2\ell}[\mathbf{F}(\mathbf{X}_0 + \ell\mathcal{V}) - \mathbf{F}(\mathbf{X}_0 - \ell\mathcal{V})] \cdot \mathcal{V} \\ &= \frac{1}{2\ell}[\mathbf{F}_1 - \mathbf{F}_2] \cdot \mathcal{V}, \end{aligned} \tag{2.6}$$

the curvature is approximated as

$$C_{\mathcal{V}} \approx \frac{1}{2\ell}(\mathbf{F}_2 - \mathbf{F}_1) \cdot \mathcal{V}. \tag{2.7}$$

From Eq. (2.5) and the 2$^{\text{nd}}$ order central finite difference formula

$$D^2_{\mathcal{V}}U(\mathbf{X})|_{\mathbf{X}_0} \approx \frac{1}{\ell^2}[U(\mathbf{X}_0 + \ell\mathcal{V}) - 2U(\mathbf{X}_0) + U(\mathbf{X}_0 - \ell\mathcal{V})], \tag{2.8}$$

one can save two energy evaluations and approximate the dimer's energy as

$$E \approx 2E_0 + \ell^2 C_{\mathcal{V}}. \tag{2.9}$$

The energy at $\mathbf{X}_1$ can be approximated as

$$E_1 = U(\mathbf{X}_0 + \ell\mathcal{V}) \approx E_0 - \ell\mathbf{F}_0 \cdot \mathcal{V} + \frac{\ell^2}{2}C_{\mathcal{V}}. \tag{2.10}$$

2.3 Dimer Rotation

The dimer's energy in Eq. (2.9) is linearly proportional to the curvature along its axis. Therefore, minimizing the dimer's energy is equivalent to minimizing the curvature. This is done by rotating the dimer around its center until $\mathcal{V}$ becomes parallel to the direction along which $C_{\mathcal{V}}$ is minimal. In other words, to find the lowest eigenvalue of the Hessian matrix and its corresponding eigenvector at $\mathbf{X}_0$, the dimer is rotated a certain angle around its midpoint. In fact, this is a constrained minimization problem that can be formulated as

$$\text{Find } \mathbf{X}_1 \in \mathcal{N} \text{ such that } \mathbf{X}_1 = \underset{\mathbf{X}\in\mathcal{N}}{\arg\min}[U(\mathbf{X}) + U(2\mathbf{X}_0 - \mathbf{X})], \tag{2.11}$$

where $\mathcal{N}(\mathbf{X}_0, \ell) \subset \mathbb{R}^{3N_p}$ is a hemisphere centered at $\mathbf{X}_0$ and of radius ℓ. In subsection 2.3.1 we review the work done by Henkelman and Jónsson [10] to minimize the curvature along the dimer's axis. In subsection 2.3.2 & subsection 2.3.3 we review the modifications on the rotation step presented by Heyden and Keil [13] and Kästner and Sherwood [15], respectively.

2.3.1 Modified Newton Method for Rotation within a Plane

Fixing the midpoint of the dimer while its endpoints are free to move will result in a rotational force orthogonal to $\mathcal{V}$,

$$\mathbf{F}^{\perp} = \mathbf{F}_1^{\perp} - \mathbf{F}_2^{\perp}, \text{ where } \mathbf{F}_i^{\perp} = \mathbf{F}_i - \mathbf{F}_i^{\parallel} \text{ and } \mathbf{F}_i^{\parallel} = (\mathbf{F}_i \cdot \mathcal{V})\mathcal{V} \text{ for } i = 1, 2, \tag{2.12}$$

which is the net rotational force acting on one of the endpoint (here we are taking it to be at endpoint 1.) It is worth mentioning that the components of the forces $\mathbf{F}_1$ and $\mathbf{F}_2$ that are parallel to the dimer's axis have no use in both rotational and translational steps. The rotational plane where the dimer is rotated is spanned by two orthonormal vectors $\mathcal{V}$ and $\mathbf{\Theta}$, where the latter is taken to be a unit vector parallel to the rotational force (see Figure 2.2)

$$\mathbf{\Theta} = \frac{\mathbf{F}^{\perp}}{\|\mathbf{F}^{\perp}\|}. \tag{2.13}$$

We will deal with the magnitude of $\mathbf{F}^{\perp}$ that is defined as

$$F \coloneqq \left\|\mathbf{F}^{\perp}\right\| = (\mathbf{F}_1^{\perp} - \mathbf{F}_2^{\perp}) \cdot \mathbf{\Theta}. \tag{2.14}$$

In the original paper [10], the authors suggest dividing by ℓ so that the force is independent of the distance separating the dimer's points.

In order to minimize its energy, we rotate the dimer by an angle $\hat{\phi}$ that could be approximated by one Newton's iteration as

$$\hat{\phi} \approx \theta_0 - \frac{F|_{\text{at } \theta_0}}{F'|_{\text{at } \theta_0}}, \tag{2.15}$$

where θ_0 is an initial guess. If we rotate the dimer in the rotational plane by a small angle $\Delta\theta$ and let $\theta_0 = \frac{1}{2}\Delta\theta$ and denote by $F(0)$ the scaled rotational force before rotation, $F(\frac{1}{2}\Delta\theta)$ and $F'(\frac{1}{2}\Delta\theta)$ the scaled rotational force and its derivative after rotation by an angle $\frac{1}{2}\Delta\theta$,

respectively, then we have the following approximations:

$$F(\Delta\theta/2) \approx \frac{F(\Delta\theta) + F(0)}{2}, \tag{2.16}$$

and its derivative, using central finite difference,

$$F'(\Delta\theta/2) \approx \frac{F(\Delta\theta) - F(0)}{\Delta\theta}. \tag{2.17}$$

Since the dimer is already rotated by $\Delta\theta/2$ far from θ_0, we only need to rotate it by

$$\begin{aligned} \phi_{\min} = \hat{\phi} - \frac{\Delta\theta}{2} &\approx \left(\frac{\Delta\theta}{2} - \frac{\Delta\theta}{2} \frac{F(\Delta\theta) + F(0)}{F(\Delta\theta) - F(0)} \right) - \frac{\Delta\theta}{2} \\ &= -\frac{\Delta\theta}{2} \frac{F(\Delta\theta) + F(0)}{F(\Delta\theta) - F(0)}. \end{aligned} \tag{2.18}$$

Unfortunately, this formula is not accurate and it overestimates the angle $\phi_{\min}$ [10]. However, expanding the potential energy U in the 2-dimensional plane of rotation to the second order around the dimer's midpoint will give a better approximation to $\phi_{\min}$. Using this expansion, we will be able to represent the curvature along the dimer's axis, $C_{\mathcal{V}}$, as a function of the rotational angle θ.

To have an analytic form that depends on the angle of rotation, we redefine the representation for the dimer's points using a new 2-dimensional coordinate system. This coordinate system uses two eigenvectors of the Hessian at $\mathbf{X}_0$. Define the plane of rotation as the plane spanned by two eigenvectors of the Hessian at $\mathbf{X}_0$. Assuming rotating the dimer a certain angle in this plane makes it oriented in the lowest curvature direction, then this direction must be one of the two eigenvectors spanning the plane. Let $\mathcal{X}$ and $\mathcal{Y}$ be these two eigenvectors with corresponding eigenvalues $C_{\mathcal{X}}$ and $C_{\mathcal{Y}}$, and let θ_0 be the angle between $\mathcal{X}$ and $\mathcal{V}(0)$ as shown in Figure 2.3, where $\mathcal{V}(0)$ is the dimer's axis before rotation, i.e. when $\theta = 0$. The origin in the new coordinate system is $\mathbf{X}_0$, and it is represented by $(0, 0) = 0\mathcal{X} + 0\mathcal{Y}$. For any point $\mathbf{X} \in \mathbb{R}^{3N}$ that lies in the rotational plane, it is represented

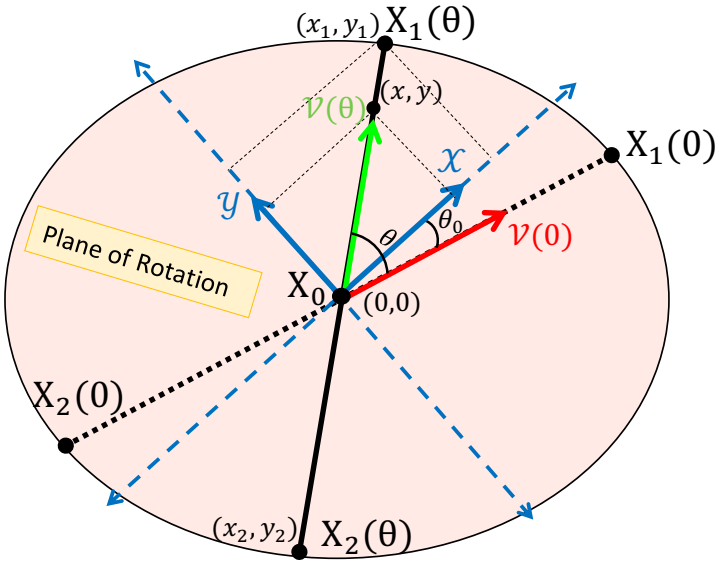


Figure 2.3: The 2-dimensional representation of the dimer's points in the plane of rotation that is spanned by two orthonormal eigenvectors $\mathcal{X}$ and $\mathcal{Y}$ of the Hessian matrix at $\mathbf{X}_0$.

as $(x, y) = x\mathcal{X} + y\mathcal{Y}$, where x and y are the projection of the 2-dimensional distance vector between the origin and $\mathbf{X}$ on $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Now suppose the dimer's axis is at an arbitrary angle θ, then the point $\mathbf{X}_1(\theta)$ is represented by

$$(x_1, y_1) = (\ell\mathcal{V}(\theta) \cdot \mathcal{X}, \ell\mathcal{V}(\theta) \cdot \mathcal{Y}) = (\ell\cos(\theta - \theta_0), \ell\sin(\theta - \theta_0)), \tag{2.19}$$

and similarly $\mathbf{X}_2(\theta)$ by

$$(x_2, y_2) = (\ell[-\mathcal{V}(\theta)] \cdot \mathcal{X}, \ell[-\mathcal{V}(\theta)] \cdot \mathcal{Y}) = (-\ell\cos(\theta - \theta_0), -\ell\sin(\theta - \theta_0)). \tag{2.20}$$

Moreover, the Hessian matrix at $\mathbf{X}_0$ is represented by

$$\mathbb{H}(0,0) = \begin{bmatrix} U_{\mathcal{X}\mathcal{X}}(0,0) & U_{\mathcal{X}\mathcal{Y}}(0,0) \\ U_{\mathcal{Y}\mathcal{X}}(0,0) & U_{\mathcal{Y}\mathcal{Y}}(0,0) \end{bmatrix}. \tag{2.21}$$

By Taylor's Theorem, the value of U at a point (x, y) on the dimer's axis after rotation by an angle θ is approximated by

$$\begin{aligned} U(x,y) &\approx U(0,0) + U_{\mathcal{X}}(0,0)x + U_{\mathcal{Y}}(0,0)y \\ &\qquad + \frac{1}{2}(x\mathcal{X} + y\mathcal{Y})^T \begin{bmatrix} U_{\mathcal{X}\mathcal{X}}(0,0) & U_{\mathcal{X}\mathcal{Y}}(0,0) \\ U_{\mathcal{Y}\mathcal{X}}(0,0) & U_{\mathcal{Y}\mathcal{Y}}(0,0) \end{bmatrix} (x\mathcal{X} + y\mathcal{Y}) \\ &= E_0 + U_{\mathcal{X}}(0,0)x + U_{\mathcal{Y}}(0,0)y + \frac{1}{2}(x\mathcal{X}^T + y\mathcal{Y}^T)\mathbb{H}(0,0)(x\mathcal{X} + y\mathcal{Y}) \\ &= E_0 + U_{\mathcal{X}}(0,0)x + U_{\mathcal{Y}}(0,0)y + \frac{1}{2}(C_{\mathcal{X}}x^2 + C_{\mathcal{Y}}y^2). \end{aligned} \tag{2.22}$$

Using the above expansion, Eq. (2.19) and Eq. (2.20), we have

$$\begin{aligned} E_1 = U(x_1, x_2) &= E_0 + \ell U_{\mathcal{X}}(0,0)\cos(\theta - \theta_0) + \ell U_{\mathcal{Y}}(0,0)\sin(\theta - \theta_0) \\ &\qquad + \frac{\ell^2}{2}[C_{\mathcal{X}}\cos^2(\theta - \theta_0) + C_{\mathcal{Y}}\sin^2(\theta - \theta_0)], \end{aligned}$$

$$\begin{aligned} E_2 = U(x_2, x_2) &= E_0 - \ell U_{\mathcal{X}}(0,0)\cos(\theta - \theta_0) - \ell U_{\mathcal{Y}}(0,0)\sin(\theta - \theta_0) \\ &\qquad + \frac{\ell^2}{2}[C_{\mathcal{X}}\cos^2(\theta - \theta_0) + C_{\mathcal{Y}}\sin^2(\theta - \theta_0)], \end{aligned}$$

and hence

$$\begin{aligned} E = E_1 + E_2 &= 2E_0 + \ell^2[C_{\mathcal{X}}\cos^2(\theta - \theta_0) + C_{\mathcal{Y}}\sin^2(\theta - \theta_0)] \\ &= 2E_0 + \ell^2[(C_{\mathcal{X}} - C_{\mathcal{Y}})\cos^2(\theta - \theta_0) + C_{\mathcal{Y}}] \\ &= 2E_0 + \ell^2[(C_{\mathcal{X}} - C_{\mathcal{Y}})\{\cos 2(\theta - \theta_0) + 1\}/2 + C_{\mathcal{Y}}]. \end{aligned} \tag{2.23}$$

Comparing Eq. (2.9) with Eq. (2.23), we have

$$C_{\mathcal{V}(\theta)} = \frac{(C_{\mathcal{X}} - C_{\mathcal{Y}})}{2}\cos 2(\theta - \theta_0) + \frac{C_{\mathcal{X}} + C_{\mathcal{Y}}}{2}. \tag{2.24}$$

The rotational force is

$$F(\theta) = -\frac{dE}{d\theta} = \ell^2(C_{\mathcal{X}} - C_{\mathcal{Y}}) \sin 2(\theta - \theta_0) \tag{2.25}$$

and the curvature along the dimer's axis is

$$F'(\theta) = \frac{d^2E}{d\theta^2} = -2\ell^2(C_{\mathcal{X}} - C_{\mathcal{Y}}) \cos 2(\theta - \theta_0). \tag{2.26}$$

We see that E is minimized or maximized when $\theta = \theta_0$ or $\theta = \theta_0 + \frac{\pi}{2}$, i.e. when the dimer is oriented in the direction of $\mathcal{X}$ or in the direction of $\mathcal{Y}$. If rotating the dimer by θ_0 (i.e. being in the direction of $\mathcal{X}$) gives negative curvature, then the dimer's axis must be rotated by $\pi/2$ to be in the direction of $\mathcal{Y}$, instead. To calculate θ_0, we rotate the dimer by a small angle $\Delta\theta$, then approximate F and F' at $\frac{1}{2}\Delta\theta$. The angle required so that the dimer's axis is in the direction of $\mathcal{X}$ is θ_0. But since the approximation starts at $\Delta\theta/2$, the dimer needs to be rotated from that point by $\phi_{\min} = \theta_0 - \Delta\theta/2$ to get to minimum or maximum. Thus, at $\theta = \Delta\theta/2$, we have

$$\begin{aligned} F\left(\Delta\theta/2\right) &= \ell^2(C_{\mathcal{X}} - C_{\mathcal{Y}}) \sin(\Delta\theta - 2\theta_0), \text{ and} \\ F'\left(\Delta\theta/2\right) &= 2\ell^2(C_{\mathcal{X}} - C_{\mathcal{Y}}) \cos(\Delta\theta - 2\theta_0) \end{aligned}$$

which implies that

$$\begin{aligned} \phi_{\min} = \theta_0 - \frac{1}{2}\Delta\theta &= \left(\frac{1}{2}\Delta\theta - \frac{1}{2}\arctan\frac{2F(\Delta\theta/2)}{F'(\Delta\theta/2)}\right) - \frac{1}{2}\Delta\theta \\ &= -\frac{1}{2}\arctan\frac{2F(\Delta\theta/2)}{F'(\Delta\theta/2)} \qquad (2.27) \\ &= -\frac{1}{2}\arctan\left[\Delta\theta\,\frac{F(\Delta\theta) + F(0)}{F(\Delta\theta) - F(0)}\right], \qquad (2.28) \end{aligned}$$

where F is as defined in Eq. (2.16).

2.3.2 First Modification on Dimer Rotation

Let $\mathcal{X}$ be a unit vector in the rotational plane. Let ϕ be the angle between $\mathcal{X}$ and the initial dimer's direction $\mathcal{V}$ (see Figure 2.4.) We can write $\mathcal{X}$ as

$$\mathcal{X}(\phi) = \cos\phi\,\mathcal{V} + \sin\phi\,\mathbf{\Theta}, \tag{2.29}$$

and thus the dimer's points as a function of ϕ are

$$\mathbf{X}_1 = \mathbf{X}_0 + \ell(\cos\phi\,\mathcal{V} + \sin\phi\,\mathbf{\Theta}), \text{ and} \tag{2.30}$$

$$\mathbf{X}_2 = \mathbf{X}_0 - \ell(\cos\phi\,\mathcal{V} + \sin\phi\,\mathbf{\Theta}). \tag{2.31}$$

If $\mathcal{X}(\phi)$ is the direction of the lowest curvature, then the curvature along it is given by

$$C_{\mathcal{X}} = \mathcal{X}^T \mathbb{H}(\mathbf{X}_0)\mathcal{X}. \tag{2.32}$$

Let $\{(\lambda_i, \mathcal{U}_i)\}_{i=1}^{3N_P}$ be the set of eigen-pairs for $\mathbb{H}(\mathbf{X}_0)$, where $\{\mathcal{U}_i\}_{i=1}^{3N_p}$ is an orthonormal set. Then, $\mathcal{X} = \sum_{i=1}^{3N_p}(\mathcal{X}\cdot\mathcal{U}_i)\mathcal{U}_i = \sum_{i=1}^{3N_p}[\cos\phi(\mathcal{V}\cdot\mathcal{U}_i) + \sin\phi(\mathbf{\Theta}\cdot\mathcal{U}_i)]\mathcal{U}_i$, and therefore,

$$\begin{aligned}
C_{\mathcal{X}} &= \sum_{i=1}^{3N_p}[\cos\phi(\mathcal{V}\cdot\mathcal{U}_i) + \sin\phi(\mathbf{\Theta}\cdot\mathcal{U}_i)]\mathcal{U}_i\,\mathbb{H}(\mathbf{X}_0)\sum_{j=1}^{3N_p}[\cos\phi(\mathcal{V}\cdot\mathcal{U}_j) + \sin\phi(\mathbf{\Theta}\cdot\mathcal{U}_j)]\mathcal{U}_j \\
&= \sum_{i=1}^{3N_p}\lambda_i[\cos\phi(\mathcal{V}\cdot\mathcal{U}_i) + \sin\phi(\mathbf{\Theta}\cdot\mathcal{U}_i)]^2 \\
&= \left(\sum_{i=1}^{3N_p}\lambda_i(\mathcal{V}\cdot\mathcal{U}_i)^2\right)\cos^2\phi + \left(\sum_{i=1}^{3N_p}\lambda_i(\mathcal{V}\cdot\mathcal{U}_i)(\mathbf{\Theta}\cdot\mathcal{U}_i)\right)\sin 2\phi \\
&\qquad + \left(\sum_{i=1}^{3N_p}\lambda_i(\mathbf{\Theta}\cdot\mathcal{U}_i)^2\right)\sin^2\phi.
\end{aligned}$$

Using the identities $\cos^2\phi = (\cos 2\phi + 1)/2$ and $\sin^2\phi = (1 - \cos 2\phi)/2$, we can write the

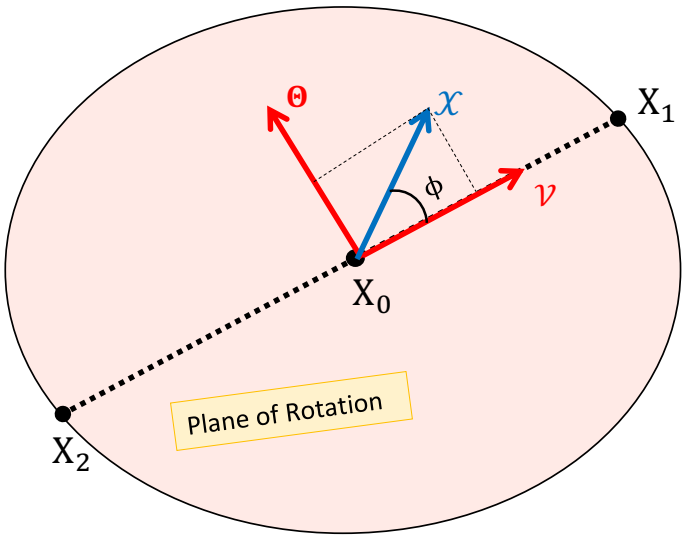


Figure 2.4: The rotational plane spanned by $\mathcal{V}$ and $\mathbf{\Theta}$. $\mathcal{X}$ is a vector in the rotational plane that is an angle ϕ far from the dimer's axis.

curvature as a Fourier series

$$C_{\mathcal{X}}(\phi) = \frac{a_0}{2} + a_1 \cos 2\phi + b_1 \sin 2\phi, \tag{2.33}$$

where a_0, a_1, and b_1 are constants that are determined by the eigenvalues and eigenvectors of the Hessian at $\mathbf{X}_0$ as well as the current configuration of the dimer's points. Notice that this equation is only valid when $\mathcal{X}$ is oriented in the lowest curvature direction. From Eq. (2.9), the dimer's energy when the dimer's axis is oriented in the direction $\mathcal{X}$ is

$$E(\phi) \approx 2E_0 + \ell^2 \left(\frac{a_0}{2} + a_1 \cos 2\phi + b_1 \sin 2\phi \right), \tag{2.34}$$

and hence the rotational force is

$$F(\phi) = -\frac{dE}{d\phi} =\approx 2\ell^2 (a_1 \sin 2\phi - b_1 \cos 2\phi) = -\ell^2 \frac{dC_{\mathcal{X}}}{d\phi}. \tag{2.35}$$

The angle required to minimize the curvature is

$$\phi_{\min} = \frac{1}{2} \arctan \frac{b_1}{a_1}. \tag{2.36}$$

Notice that

$$\begin{aligned}
\frac{dE}{d\phi} &= \frac{dE_1}{d\mathbf{X}_1} \cdot \frac{d\mathbf{X}_1}{d\phi} + \frac{dE_2}{d\mathbf{X}_2} \cdot \frac{d\mathbf{X}_2}{d\phi} \\
&= -\mathbf{F}_1(\phi) \cdot \frac{d\mathbf{X}_1}{d\phi} - \mathbf{F}_2(\phi) \cdot \frac{d\mathbf{X}_2}{d\phi} \\
&= -\mathbf{F}_1(\phi) \cdot \ell(-\sin\phi\, \mathcal{V}(0) + \cos\phi\, \boldsymbol{\Theta}(0)) - \mathbf{F}_2(\phi) \cdot \ell(\sin\phi\, \mathcal{V}(0) - \cos\phi\, \boldsymbol{\Theta}(0))
\end{aligned}$$

but $\boldsymbol{\Theta}(\phi) = -\sin\phi\, \mathcal{V}(0) + \cos\phi\, \boldsymbol{\Theta}(0)$, and so

$$\frac{dE}{d\phi}(\phi) = \ell[\mathbf{F}_2(\phi) - \mathbf{F}_1(\phi)] \cdot \boldsymbol{\Theta}(\phi). \tag{2.37}$$

From Eq. (2.35) and Eq. (2.37), the derivative of the curvature is

$$\frac{dC_{\mathcal{X}}}{d\phi}(\phi) = \frac{1}{\ell}[\mathbf{F}_2(\phi) - \mathbf{F}_1(\phi)] \cdot \boldsymbol{\Theta}(\phi). \tag{2.38}$$

Substituting $\phi = 0$ and $\phi = \phi_1$ (the choice of ϕ_1 will be discussed below) in Eq. (2.33) and Eq. (2.35) gives

$$b_1 = \frac{1}{2} \frac{dC_{\mathcal{X}}}{d\phi}(0) \tag{2.39}$$

$$a_1 = \frac{C_{\mathcal{X}}(0) - C_{\mathcal{X}}(\phi_1) + \frac{1}{2}\frac{dC_{\mathcal{X}}}{d\phi}(0) \sin 2\phi_1}{1 - \cos 2\phi_1} \tag{2.40}$$

$$= \frac{\frac{dC_{\mathcal{X}}}{d\phi}(0) \cos 2\phi_1 - \frac{dC_{\mathcal{X}}}{d\phi}(\phi_1)}{2 \sin 2\phi_1} \tag{2.41}$$

$$a_0 = 2(C_{\mathcal{X}}(0) - a_1). \tag{2.42}$$

The curvature at $\phi = 0$ and $\phi = \phi_1$ is calculated by Eq. (2.7)

$$C_{\mathcal{X}(\phi)} \approx \frac{1}{2\ell}[\mathbf{F}_1(\phi) - \mathbf{F}_2(\phi)] \cdot \mathcal{X}(\phi), \tag{2.43}$$

or by Eq. (2.10)

$$C_{\mathcal{X}(\phi)} \approx \frac{2[E_1(\phi) - E_0 + \ell \mathbf{F}_0 \cdot \mathcal{X}(\phi)]}{\ell^2}. \tag{2.44}$$

The angle ϕ_1 lies between 0 and $\pi/2$ and can be approximated using Eq. (2.27) as

$$\phi_1 \approx -\frac{1}{2}\arctan\frac{2F(0)}{F'(0)} = -\frac{1}{2}\arctan\frac{-2\ell^2\frac{dC_{\mathcal{X}}}{d\phi}(0)}{-\ell^2\frac{d^2C_{\mathcal{X}}}{d\phi^2}(0)} = -\frac{1}{2}\arctan\frac{2\frac{dC_{\mathcal{X}}}{d\phi}(0)}{\frac{d^2C_{\mathcal{X}}}{d\phi^2}(0)}.$$

We need to approximate the second derivative of the curvature along direction $\mathcal{X}$ at $\phi = 0$. Using Eq. (2.35), we have

$$\begin{aligned}
\ell^2\frac{d^2C_{\mathcal{X}}}{d\phi^2} &\approx \frac{d^2E}{d\phi^2} \\
&= \frac{d^2E_1}{d\phi^2} + \frac{d^2E_2}{d\phi^2} \\
&= \frac{d}{d\phi}\left(\frac{dE_1}{d\mathbf{X}_1}\cdot\frac{d\mathbf{X}_1}{d\phi}\right) + \frac{d}{d\phi}\left(\frac{dE_2}{d\mathbf{X}_2}\cdot\frac{d\mathbf{X}_2}{d\phi}\right) \\
&= \frac{d\mathbf{X}_1}{d\phi}^T \mathbb{H}(\mathbf{X}_1)\frac{d\mathbf{X}_1}{d\phi} + \frac{dE_1}{d\mathbf{X}_1}\cdot\frac{d^2\mathbf{X}_1}{d\phi^2} + \frac{d\mathbf{X}_2}{d\phi}^T \mathbb{H}(\mathbf{X}_2)\frac{d\mathbf{X}_2}{d\phi} + \frac{dE_2}{d\mathbf{X}_2}\cdot\frac{d^2\mathbf{X}_2}{d\phi^2} \\
&= (\ell\boldsymbol{\Theta}(\phi)^T)\mathbb{H}(\mathbf{X}_1)(\ell\boldsymbol{\Theta}(\phi)) - \mathbf{F}_1(\phi)\cdot(-\ell\mathcal{X}(\phi)) \\
&\qquad + (-\ell\boldsymbol{\Theta}(\phi)^T)\mathbb{H}(\mathbf{X}_2)(-\ell\boldsymbol{\Theta}(\phi)) - \mathbf{F}_2(\phi)\cdot(\ell\mathcal{X}(\phi)) \\
&= \ell^2\Big(\boldsymbol{\Theta}(\phi)^T[\mathbb{H}(\mathbf{X}_0 + \ell\mathcal{X}(\phi)) + \mathbb{H}(\mathbf{X}_0 - \ell\mathcal{X}(\phi))]\boldsymbol{\Theta}(\phi) \\
&\qquad + \frac{1}{\ell}[\mathbf{F}_1(\phi) - \mathbf{F}_2(\phi)]\cdot\mathcal{X}(\phi)\Big) \\
&= 2\ell^2\left(\boldsymbol{\Theta}(\phi)^T\mathbb{H}(\mathbf{X}_0)\boldsymbol{\Theta}(\phi) - C_{\mathcal{X}(\phi)}\right) \\
&= 2\ell^2(C_{\boldsymbol{\Theta}(\phi)} - C_{\mathcal{X}(\phi)}).
\end{aligned}$$

Consequently,

$$\frac{d^2C_{\mathcal{X}}}{d\phi^2}(0) \approx 2(C_{\boldsymbol{\Theta}(0)} - C_{\mathcal{X}(0)}). \tag{2.45}$$

In [13], the authors assume $\frac{d^2C_{\mathcal{X}}}{d\phi^2}(0) \approx 4|C_{\mathcal{X}(0)}|$, and hence,

$$\phi_1 \approx -\frac{1}{2}\arctan\frac{\frac{dC_{\mathcal{X}}}{d\phi}(0)}{2|C_{\mathcal{X}(0)}|}. \tag{2.46}$$

If this estimate of the rotational angle ϕ_1 is larger than a specified ϕ_{tol}, one calculates a_0, a_1, b_1, and the angle ϕ that is required to minimize the dimer's energy, otherwise, no rotation is performed. Several rotations, in different planes, must be performed to accurately minimize the curvature. In this study, the maximum number of rotaions is taken to be eight. We continue the rotations of the dimer until either ϕ_1 or ϕ_{min} less than ϕ_{tol} [15]. Suppose the current rotational plane is spanned by $\{\mathcal{V}_0, \boldsymbol{\Theta}_0\}$. Find $\phi_{\text{min}}^{(0)}$ using Eq. (2.36). If $|\phi_1|$ or $|\phi_{\text{min}}^{(0)}|$ less than ϕ_{tol} or $\left\|F(\phi_{\text{min}}^{(0)})\right\|$ less than F_{tol}, stop rotations, otherwise, find $\mathcal{V}_1$ and $\boldsymbol{\Theta}_1$ and repeat. A Conjugate gradient approach is more efficient than the steepest descent algorithm described above. Let $\mathbf{H}_0 = \mathbf{F}_0^{\perp}$, $\mathbf{G}_0 = \boldsymbol{\Theta}_0$ and $\boldsymbol{\Theta}_0^*$ be the vector resulting from rotating $\boldsymbol{\Theta}_0$ by $\phi_{\text{min}}^{(0)}$. Calculate the rotational force after rotation and denote it by $\mathbf{F}_0^{\perp *}$. If stopping criteria are not met, then the next conjugate direction is $\mathbf{G}_1 = \mathbf{H}_1/\|\mathbf{H}_1\|$, where $\mathbf{H}_1 = \mathbf{F}_1^{\perp} + \beta\,\|\boldsymbol{H}_0\|\,\boldsymbol{\Theta}_0^*$, and $\beta = \mathbf{F}_1^{\perp} \cdot (\mathbf{F}_1^{\perp} - \mathbf{F}_0^{\perp *})/\left\|\mathbf{F}_0^{\perp *}\right\|^2$. Find the angle required for minimization in the plane spanned by $\{\mathcal{V}_1, \mathbf{G}_1\}$. Repeat until any of the stopping criteria is met.

2.3.3 Second Modification on Dimer Rotation

In the previous section, it was mentioned that one rotation may is not enough to orientate the dimer in the lowest curvature direction. As a result, one needs to calculate the force at $\mathbf{X}_1$ after each rotation to use it as an input for the next one. This could be very costly

especially when the system's size is large. In this section we show how Kästner and Sherwood [15] approximate the force at $\mathbf{X}_1$ after each rotation.

Let $\mathbf{F}'$ denote the force at the point $\mathbf{X}'$ that is shown in Figure 2.5, which can be found by interpolating between $\mathbf{X}_1(0)$ and $\mathbf{X}_1(\phi_1)$ (we do not need to calculate $\mathbf{X}'$ nor $\mathbf{F}'$.) By linear interpolation between the pairs $(\mathbf{X}_1(0), \mathbf{F}_1(0))$ and $(\mathbf{X}_1(\phi_1), \mathbf{F}_1(\phi_1))$, we have

$$\begin{aligned}\mathbf{F}' &\approx \mathbf{F}_1(0)\left(1 - \frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}\right) + \mathbf{F}_1(\phi_1)\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|} \\ &= \mathbf{F}_1(0) + [\mathbf{F}_1(\phi_1) - \mathbf{F}_1(0)]\,\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}.\end{aligned}$$

By linear extrapolation between the pairs $(\mathbf{X}_0, \mathbf{F}_0)$ and $(\mathbf{X}', \mathbf{F}')$, we have

$$\mathbf{F}_1(\phi_{\min}) \approx \mathbf{F}_0 + [\mathbf{F}' - \mathbf{F}_0]\frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|}.$$

Substituting the formula for $\mathbf{F}'$ in the above expression

$$\begin{aligned}\mathbf{F}_1(\phi_{\min}) &= \mathbf{F}_0 + \frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|}\left[\mathbf{F}_1(0) + [\mathbf{F}_1(\phi_1) - \mathbf{F}_1(0)]\,\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|} - \mathbf{F}_0\right] \\ &= \left(1 - \frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|}\right)\mathbf{F}_0 + \frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|}\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}\mathbf{F}_1(\phi_1) \\ &\quad + \frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|}\left(1 - \frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}\right)\mathbf{F}_1(0). \qquad (2.47)\end{aligned}$$

From Figure 2.5, we see that

$$\begin{aligned}\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\sin\phi_{\min}} &= \frac{\|\mathbf{X}' - \mathbf{X}_0\|}{\sin\theta} = \frac{\|\mathbf{X}' - \mathbf{X}_1(\phi_1)\|}{\sin(\phi_1 - \phi_{\min})}, \\ \frac{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}{\sin\phi_1} &= \frac{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_0\|}{\sin\theta} = \frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\sin\theta}.\end{aligned}$$

Thus,

$$\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|} = \frac{\|\mathbf{X}' - \mathbf{X}_0\|}{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}\frac{\sin\phi_{\min}}{\sin\phi_1},$$

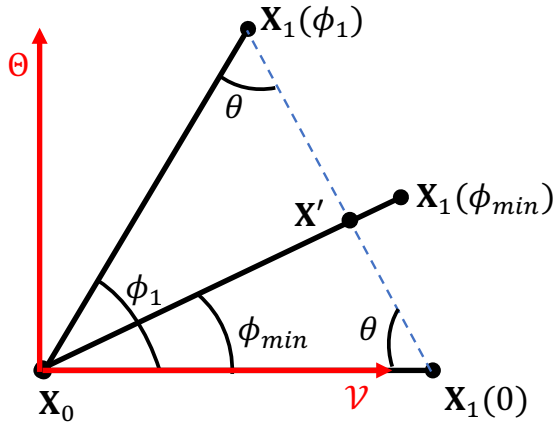


Figure 2.5: The dimer's endpoint $\mathbf{X}_1$ before rotation, after rotating by ϕ_1 and after rotating by $\phi_{\min}$.

$$\frac{\|\mathbf{X}_1(\phi_{\min}) - \mathbf{X}_0\|}{\|\mathbf{X}' - \mathbf{X}_0\|} = \frac{\|\mathbf{X}_1(\phi_1) - \mathbf{X}_1(0)\|}{\|\mathbf{X}' - \mathbf{X}_1(\phi_1)\|} \frac{\sin(\phi_1 - \phi_{\min})}{\sin\phi_1}$$

and

$$\frac{\|\mathbf{X}' - \mathbf{X}_1(0)\|}{\|\mathbf{X}' - \mathbf{X}_1(\phi_1)\|} = \frac{\sin\phi_{\min}}{\sin(\phi_1 - \phi_{\min})}.$$

Substituting in Eq. (2.47), we have

$$\begin{aligned}
\mathbf{F}_1(\phi_{\min}) &\approx \left(1 - \frac{\sin(\phi_{\min}) + \sin(\phi_1 - \phi_{\min})}{\sin\phi_1}\right)\mathbf{F}_0 + \frac{\sin\phi_{\min}}{\sin\phi_1}\mathbf{F}_1(\phi_1) \\
&\quad + \frac{\sin(\phi_1 - \phi_{\min})}{\sin\phi_1}\mathbf{F}_1(0) \\
&= \left(1 - \cos\phi_{\min} - \sin(\phi_{\min})\tan\left(\frac{\phi_1}{2}\right)\right)\mathbf{F}_0 + \frac{\sin\phi_{\min}}{\sin\phi_1}\mathbf{F}_1(\phi_1) \\
&\quad + \frac{\sin(\phi_1 - \phi_{\min})}{\sin\phi_1}\mathbf{F}_1(0). \qquad (2.48)
\end{aligned}$$

The force at $\mathbf{X}_2$ is approximated as in Eq. (2.4). This approximation of the force saves one gradient calculation per dimer rotation.

2.4 Dimer Translation

After rotation, the second step in the dimer method is to translate the dimer whilst its orientation is fixed in a direction that gets the dimer's midpoint closer to an index 1 saddle point. The component of the force $\mathbf{F}_0$ acting on the dimer's midpoint in the direction of $\mathcal{V}$ is $\mathbf{F}^{\parallel} = (\mathbf{F}_0 \cdot \mathcal{V})\,\mathcal{V}$. However, since the force $\mathbf{F}_0$ is the negative of the energy gradient at $\mathbf{X}_0$, $\mathbf{F}^{\parallel}$ will pull the dimer in the negative direction of $\mathcal{V}$ towards a minimum. Therefore, we need to modify the force so that $\mathbf{F}^{\parallel}$ pulls the dimer in the direction of $\mathcal{V}$. We define the translational force $\mathbf{F}^{\dagger}$ as the Householder reflection of $\mathbf{F}_0$ (see Figure 2.2) given by

$$\mathbf{F}^{\dagger} = \mathbf{F}_0 - 2\mathbf{F}^{\parallel}. \tag{2.49}$$

If the dimer is started from a convex region, which is the neighborhood of a local minimum, there is a chance that the Householder vector will be along a contour line of the potential energy. In such case, the translational force will tend to pull the dimer uphill very slowly, or even possibly the dimer become trapped there forever. This problem will mostly be encountered at the beginning of a search. Therefore, to escape the convex region faster, the following translational force is considered

$$\mathbf{F}^{\dagger} = \begin{cases} -\mathbf{F}^{\parallel} & \text{if } C_{\mathcal{V}} > 0 \\ \mathbf{F}_0 - 2\mathbf{F}^{\parallel} & \text{if } C_{\mathcal{V}} < 0 \end{cases}. \tag{2.50}$$

The dimer will be moved in the direction of $\mathbf{F}^{\dagger}$ a distance S so that the magnitude of the translational force is smaller, and presumably if we keep doing that after each rotation step, the translational force will eventually vanish. Let $\mathbf{D} = \mathbf{F}^{\dagger}/\left\|\mathbf{F}^{\dagger}\right\|$. If the dimer is in a convex region, then it is moved a fixed distance $\Delta^{\max}$ in the direction of $\mathbf{D}$. Otherwise, the dimer is moved a small fixed distance Δ along the unit vector $\mathbf{D}$ and the translational force is calculated at $\mathbf{X}_0^{\star} = \mathbf{X}_0 + \Delta\mathbf{D}$ and denoted by $\mathbf{F}^{\dagger\star}$. The magnitude of the translational force

and the curvature along the direction of $\mathbf{D}$ at $(\mathbf{X_0} + \mathbf{X_0^\star})/2$ are approximated as follows

$$F = \frac{(\mathbf{F}^\dagger + \mathbf{F}^{\dagger\star}) \cdot \mathbf{D}}{2} \qquad \text{and} \qquad C = \frac{(\mathbf{F}^{\dagger\star} - \mathbf{F}^\dagger) \cdot \mathbf{D}}{\Delta}. \tag{2.51}$$

Then, the distance is given by

$$S = \min\left\{\frac{\Delta}{2} - \frac{F}{C}, \Delta^{\max}\right\}, \tag{2.52}$$

where $\Delta^{\max}$ is chosen to prevent the dimer from stepping too far and is usually between 0.05 and 0.2. In the non-convex region, we are trying to find an approximation for one of the zeros of the function $\mathbf{F}^\dagger \cdot \mathbf{D}$ using only one Newton step. Since the convergence of Newton's method depends on the initial guess, which is here $(\mathbf{X_0} + \mathbf{X_0^\star})/2$, the method may fail to converge and as a result the dimer method fails to find a saddle point.

In our study, we do one translation in the direction of $\mathbf{D}$, but if that failed to find a zero of $\mathbf{F}^\dagger \cdot \mathbf{D}$, we try another initial guess for Newton's method but now with $\mathbf{X}_0^\star = \mathbf{X}_0 - \Delta\mathbf{D}$. We have found that this increases the total number of found saddle points. The dimer method will converge faster if one moves the dimer in a conjugate direction to the previous direction, which was found in the translational step before doing the rotations, instead of moving it in the direction $\mathbf{D}$ found in the current step.

Once a saddle point is found, one has to perturb the system in the opposite direction of $\mathcal{V}$ found in the last step and then quench to check if the found saddle point is connected to the original minimum or not. If the minimization process failed or the minimum found is not the original minimum, one must perturb in the direction $\mathcal{V}$ and try minimizing again. Due to inaccuracy one could pass the saddle point by a small distance, therefore, $\mathcal{V}$ will be pointing out from an adjacent local minimum to the saddle point with a direction opposite to what is expected. For minimization, we do few steps of steepest descent method with small fixed steps, then we use Polak-Ribière conjugate gradient method.

Chapter 3

Rejection Scheme for Off-Lattice Kinetic Monte Carlo

3.1 Introduction

The main idea behind the new scheme is to partition the set of all transitions out of the current state into at most N_p categories, where N_p is the number of particles in the system, and to do this in a way that allows the rates to be found using localized searches centered on a particular particle. This is accomplished by first associating each transition with the particle that moves the largest distance when the system is moved from the initial state to the transition state. This retains the essential simplification of lattice-based KMC models based on single-particle moves while allowing for more complicated, multi-particle moves. Rate estimates for all moves associated with a given atom are then constructed based on local environments in a way that also mimics the bond-counting approach in a typical lattice-based model. These rate estimates reflect the intuition that loosely coordinated atoms are much more likely to reconfigure than fully coordinated atoms in the interior of a crystal. Together, the partitioning and rate estimates allow one to select a candidate event without doing the costly saddle point search for each atom in the system. In principle, this allows for an $O(N_p)$ improvement in performance. In the next section we introduce the rejection scheme.

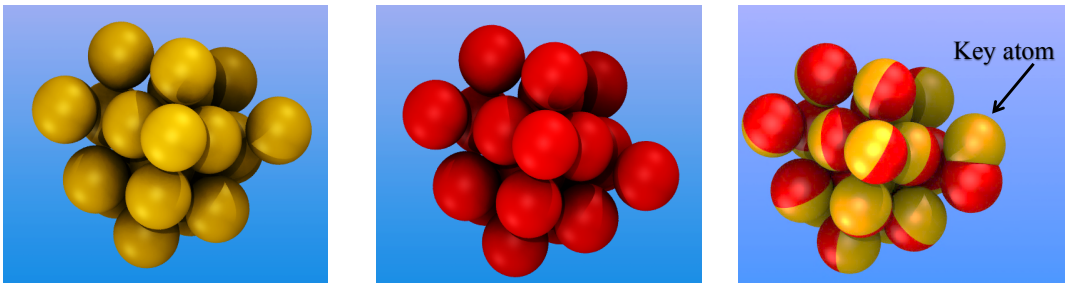
3.2 Rejection Algorithm

In a rejection algorithm, one samples a rate distribution formed from upper bounds on the actual rates, $\hat{r}_n \geq r_n$, and rejects the selected move with the appropriate frequency so that a stochastic process with the original rate distribution is formed. The simplest examples of this use a single global upper bound $\hat{r} \geq r_n, \forall n$. One can then select a candidate event using a randomly selected integer $1 \leq n \leq N$, where N is the total number of events. This avoids all but one of the rate calculations in Step 1 of Algorithm 1, but at the expense of additional random number generation when events are rejected.

The overall efficiency of a rejection scheme,

$$\mathcal{E} = \frac{\text{accepted trials}}{\text{attempts}},$$

can be very low when the rate distribution has a wide range, as is often the case for KMC simulations due to the exponential dependence of rates on the energy barrier ΔU. For lattice-based simulations, rejection-free KMC is therefore often superior because the cost of random number generation is high compared to calculating rates. The expense of rate calculations in OLKMC suggests that even an inefficient rejection scheme may be superior to the rejection-free scheme outlined in Algorithm 1. There is, however, a fundamental difficulty in implementing rejection for OLKMC in that determining the number and description of the events to be sampled relies on the same costly saddle-point searches required for the rejection-free algorithm. Below, we introduce a means of circumventing this need by partitioning the set of possible transitions so that each transition is associated with a uniquely defined *key* atom. In addition to making a rejection scheme viable, we will see that this also makes it more efficient by tailoring rate estimates to local environments. To this end, we will borrow the notion of a bond-counting formula from lattice based simulation, with an eye toward using this as a rate estimate rather than a rate model. This same strategy was used effectively in the weakly off-lattice models for strain mentioned earlier in chapter 1 [31, 32, 33].



(a) Local minimum.

(b) Connected-key saddle point.

(c) The saddle point and the local minimum snapshots being merged.

Figure 3.1: Illustration on how connected-key saddles are defined when a local saddle point search is started from a perturbed configuration with the perturbations being centered on the key atom.

Unlike lattice-based models with predefined event catalogs, neither the number nor the nature of transitions at a given time step is known *a priori* in OLKMC. Acquiring this information requires an exhaustive saddle point search like that in Step 1 of Algorithm 1, a calculation that would defeat the purpose of the rejection algorithm. In order to extend the rejection scheme to OLKMC, we partition the set of connected saddle points into localized subsets by associating moves with their key atom. We define this as the atom whose position changes by the greatest magnitude when it is moved from the configuration of the current local minimum to the saddle point configuration, and refer to the associated set of saddle points as *connected key* saddles (see Figure 3.1). For a system with N_p particles this has the effect of partitioning the entire set of transitions into at most N_p subsets, each of which represents all of the moves for which one particular atom is the key atom (see Figure 3.2.) For moves that are essentially single atom hops, this will correctly associate the event with the hopping atom, while providing a natural generalization for more complicated, multi-atom moves. One now needs to over-estimate the sum of the rates for all configuration changes associated with a given key atom. If this can be done reliably, one can then choose a candidate event based on the estimated rates. After a candidate is chosen, one then calculates the total transition rate for that particular atom, accepting or rejecting the move

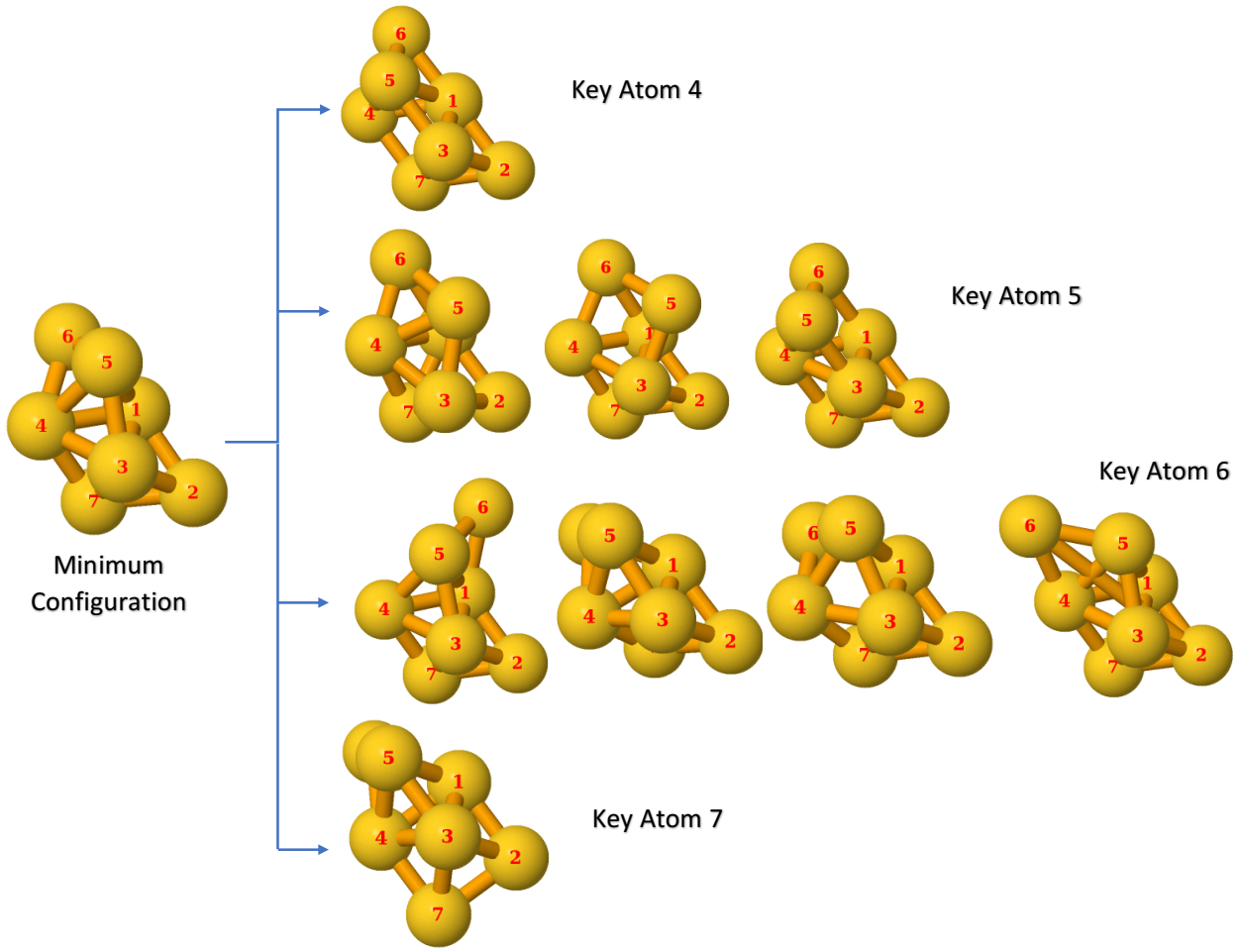


Figure 3.2: In this figure, nine connected saddle points were found surroundings the basin of the minimum configuration on the very left. The set of connected saddle points is partitioned into four subsets of connected key saddle points. Translation and rotation were prevented and that is why we did not find any saddles associated with the first three atoms.

with probability:

$$P(\text{acceptance}) = \frac{\text{true rate}}{\text{approximate rate}}.$$

This avoids the tremendously more costly need to calculate all of the rates for the entire system before selecting a move, and it does so with zero error as long as the estimates are upper bounds for the true rates.

The rate estimates we use attempt to model a lower bound for the smallest significant barrier (i.e. attempting to avoid the small barrier problem discussed above) $\check{U}$ and an upper bound $\hat{N}$ for the number of transitions that have this energy scale. To allow for greater flexibility in protecting against the estimate being lower than the true rate, we also include an additive constant $\hat{C}$ in the estimate:

$$\hat{r}_j = K\hat{N}_j \exp\left(-\check{U}_j/k_BT\right) + \hat{C}, \qquad j = 1, \cdots, N_p. \tag{3.1}$$

The energy barrier bound is based on a generalized notion of a nearest neighbor. For our off-lattice model, we define a nearest neighbor as an atom lying within a distance, d_1, slightly larger than the lattice spacing of a perfect crystal. Similarly, we define second and third neighbor distances d_2 and d_3. We use an estimate where $\check{U}_j$ is linear in the number of first, second and third nearest neighbors for each species.

Next, we provide an outline of the rejection algorithm. We assume that the system is initialized at time t to an arbitrary local minimum.

Algorithm 2: Rejection OLKMC

---

1. Calculate rate estimates $\hat{r}_j$, $j = 1, \cdots, N_p$, using Eq. (3.1).

2. Calculate partial sums $P_n = \sum_{j=1}^{n} \hat{r}_j$, $n = 1, 2, \ldots, N_p$.

3. Generate a uniformly distributed random number $r \in [0, P_{N_p})$.

4. Locate interval J such that $P_{J-1} \leq r < P_J$.

5. Update the physical time $t \leftarrow t + \Delta t$ with $\Delta t = -\ln r'/P_{N_p}$, where r' is a uniformly distributed random number in $(0, 1]$.

6. Perform *local* saddle point searches centered on atom J as follows:

   (a) For $k = 1, \cdots, N_g$ initiate dimer searches with initial guesses as in Eq. (1.4).

   (b) Sort through the resulting saddle points so that $\{\mathbf{X}_J^1, \mathbf{X}_J^2, \cdots, \mathbf{X}_J^M\}$ is the set of distinct connected key saddles.

7. Calculate the true rates $r_J^i = K \exp\left(-\Delta U_J^i / k_B T\right)$, $i = 1, 2, \ldots, M$ for moves in which atom J is the key atom.

8. Calculate partial sums $p_i = \sum_{n=1}^{i} r_J^n$, $i = 1, 2, \ldots, M$.

9. If $r - P_{J-1} > p_M$, reject the event; set $\hat{r}_J = p_M$; return to Step 2.

10. Otherwise, locate the interval I such that $p_{I-1} \leq r - P_{J-1} < p_I$.

11. Move the system to saddle-point configuration $\mathbf{X}_J^I$, perturb away from the current local minimum, and relax to the new local minimum.

The success of the method hinges on Steps 1 and 6. If the rate estimates in Step 1 are lower than the actual sum of rates, the algorithm is no longer equivalent to a fully implemented OLKMC, as the corresponding events will be undersampled. While this is undesirable, it has an effect similar to other sources of error inherent to OLKMC. This undersampling error can be monitored and controlled to some extent in that it will be detected a certain fraction of the time. If $\hat{R}_E$ is the sum of the rate estimates for the atoms with rate estimates that are too small, then the probability that one of these atoms is selected for a saddle search is $\frac{\hat{R}_E}{P_{N_p}}$. These instances can be counted and used as a metric for upwardly adjusting rate estimates. Note, however, that the selection of an undersampled event is not

an error—the error is a failure to select such events sufficiently often. The error can be more accurately measured by calculating

$$E = \sum_{i=1}^{N_p} \max(0, r_i - \hat{r}_i). \tag{3.2}$$

When events are undersampled we introduce an error with probability $\frac{E}{P_{N_p}+E}$ relative to a simulation with the minimal correction applied to the errant rates, *i.e.* $\hat{r}_i \leftarrow r_i$ whenever $\hat{r}_i < r_i$. The error cannot be computed on every iteration without losing the advantages of the rejection scheme, as one has to loop through the atom list and calculate the true rate for each atom and its rate estimate. However, it can be monitored for some small fraction of the time steps. While undersampling can be reduced or even eliminated by using sufficiently generous rate estimates, this comes at the expense of increased rejection due to oversampling. Specifically, we will reject an event with probability $1 - \mathcal{E} = 1 - R/P_{N_p}$, where R is the sum of the actual rates. In our simulations, we monitor both our undersampling error E and the efficiency $\mathcal{E}$

$$\mathcal{E} = \sum_{\substack{i=1 \\ \hat{r}_i \geq r_i}}^{N_p} r_i \Big/ \sum_{\substack{i=1 \\ \hat{r}_i \geq r_i}}^{N_p} \hat{r}_i. \tag{3.3}$$

In Step 6, we wish to calculate the sum of all the rates for which atom J is the key atom. We do this by perturbing the system about atom J as in Eq. (1.4) to create a list of initial configurations for the dimer search. It is possible that this will miss some moves which will introduce an error similar to that discussed above in the context of the rejection-free scheme. This error can be decreased by creating another list of initial configurations by perturbing the system about a neighboring atom. This will reduce the efficiency: As we make our search for connected key saddle points more exhaustive, we will produce costly duplicates and saddles associated with atoms other than the candidate atom.

In Step 9, p_M is the sum of the true transition rates for which the candidate atom is the key atom. When we reject a move where this atom is key we can set the estimated rate $\hat{r}_J$ to this sum, so that transitions associated with this atom will not be rejected on the next

trial. This normally does not save a significant amount of computation, as it is likely that a different key atom will be selected on the next iteration, but it will occur more often when there is a single loosely-bonded atom at the surface. In the cases where the same key atom is chosen on a subsequent trial, we can also re-use saddle point information obtained for the rejected trial. Similarly, on any rejected trial, we can retain the information about saddle points associated with atoms other than the candidate atom, although this will only have a slight impact on performance.

When parallel resources are available, we suggest performing both the full OLKMC and the rejection scheme by distributing the N_g dimer searches for each atom over the available processors, keeping in mind that one has to remove the duplicates before calculating the true rates. However, the boost in performance will still be $O(N_p)$. Suppose the computational cost to perform the rejection scheme on one processor is $O(N_g)$, then it is $O(N_p N_g)$ for the full scheme. When n processors are available, the cost for the rejection scheme is $O(N_g/n)$ and $O(N_p N_g/n)$ for the full scheme.

Chapter 4

Results

In this chapter we demonstrate the new algorithm by simulating the growth and evolution of a two-species cluster with an emerging core-shell structure. This is a rather challenging system compared to that typically studied using OLKMC. Indeed, the fully implemented OLKMC algorithm that is used for comparison purposes has to be abandoned once the system contains around seventy particles. We use a Lennard-Jones potential, modified for two interacting species, type A and type B [1, 19]. The total energy of a system of N_p particles interacting by the Lennard-Jones potential is given by

$$U(\mathbf{X}) = \sum_{i<j}^{N_p} \phi(r_{ij}), \quad \text{where } \phi(r_{ij}) = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right],$$

$\mathbf{X} \in \mathbb{R}^{3N_p}$ is the current configuration in three dimensions, r_{ij} is the distance between atoms i and j, $\phi(r_{ij})$ is the interaction potential, σ_{ij} is the distance at which $\phi(r_{ij})$ is zero and ϵ_{ij} is the chemical bond energy. We will use the Lorentz-Berthelot mixing rules [19] and take

$$\sigma_{ij} = \begin{cases} \sigma_A & \\ \sigma_B & \\ \frac{1}{2}(\sigma_A + \sigma_B), & \end{cases} \qquad \epsilon_{ij} = \begin{cases} \epsilon_A & \text{if both atoms are type A,} \\ \epsilon_B & \text{if both atoms are type B,} \\ \sqrt{\epsilon_A \epsilon_B}, & \text{if the atoms are different.} \end{cases}$$

We omit any truncation of the potential, but for larger systems we could employ the standard practice of introducing a cutoff radius that is often chosen to be three or four times a typical bond-spacing. To remove translational and rotational degrees of freedom, one particle is constrained to the origin, a second to a line passing through the origin, and a third to a plane containing this line. This is equivalent to adjusting the frame of reference to satisfy these constraints.

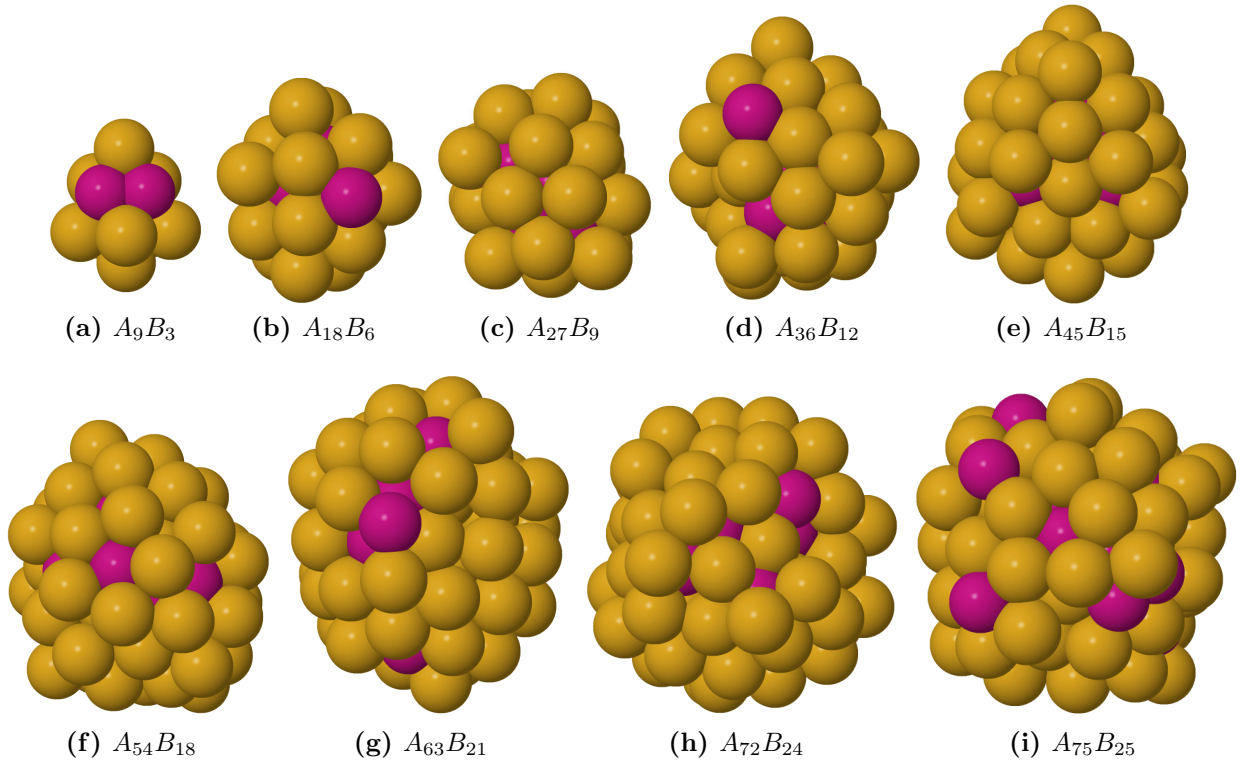
For a particle of type A or B, we denote its nearest neighbors of species α by $n_k^{(\alpha)}$ where $k = 1, 2, 3$ correspond to the first, second and third nearest neighbors, respectively. In Eq. (3.1), we take $\hat{N}_i = 4\sum_{i=1}^{3}\sum_{\alpha\in\{A,B\}} n_i^{(\alpha)}$, $\hat{C} = 1.955$ and the energy barrier bound $\check{U}_i = -1.2(B_1 + B_2)$, where

$$B_1 = \epsilon_\alpha(\lambda_1 n_1^{(\alpha)} + \lambda_2 n_2^{(\alpha)} + \lambda_3 n_3^{(\alpha)}),$$
$$B_2 = \sqrt{\epsilon_\alpha \epsilon_\beta}(\lambda_1 n_1^{(\beta)} + \lambda_2 n_2^{(\beta)} + \lambda_3 n_3^{(\beta)} + \lambda_4),$$

with $\alpha = \text{species}(i)$, $\beta \neq \alpha$, $\lambda_1 = \lambda_2 = 0.5$, $\lambda_3 = 0.8$ and $\lambda_4 = -5.9$.

Deposition is modeled by adding an additional rate, r_{dep}, to the rate table in Steps 1 and 2. When a deposition occurs, the species is selected so that the ratio is three A-particles for every B particle, and the appropriate particle is placed at a randomly selected solid angle a distance d from the origin. The coordinates of this particle are relaxed toward a local minimum by steepest descent while constraining the remaining particles. After the constrained minimum is reached, the full system is relaxed by a conjugate gradient search, with care taken to monitor for descent or large moves, so that the particle settles into a local minimum without significantly disturbing the prior configuration.

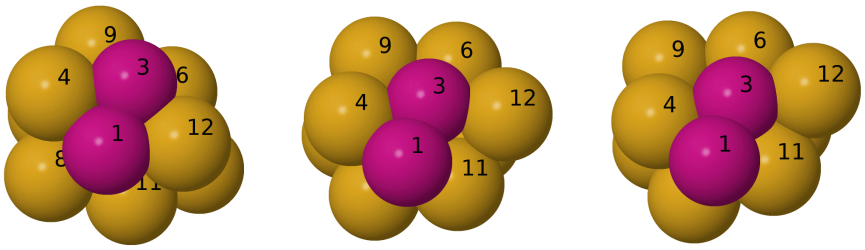
Figure 4.1 shows nine snapshots of the growth process, with a new snapshot selected after 12 particles have been added to the system. The larger particles are shown in gold and tend to evolve toward the outer shell. This tendency is increased if we slow the deposition rate, allowing more diffusive transitions between deposition events.



(a) A_9B_3 **(b)** $A_{18}B_6$ **(c)** $A_{27}B_9$ **(d)** $A_{36}B_{12}$ **(e)** $A_{45}B_{15}$

(f) $A_{54}B_{18}$ **(g)** $A_{63}B_{21}$ **(h)** $A_{72}B_{24}$ **(i)** $A_{75}B_{25}$

Figure 4.1: Snapshots at different times during the growth of a two species cluster, $A_{75}B_{25}$, with random deposition at rate = 0.85. The Lennard-Jones parameters are taken to be $\epsilon_A = \epsilon_B = 0.25$ and $\sigma_A = 1.3$ & $\sigma_B = 1$. Both the dimer and the conjugate gradient algorithms are terminated once the L^2 norm of the gradient is less than 10^{-3} or the maximum number of iterations is achieved. The view in these snapshots is chosen so that B atoms appear clearly.



(a) Minimum 1 **(b)** Saddle Point **(c)** Minimum 2

Figure 4.2: Example of a concerted move that includes the movement of all atoms except the constrained atom at the origin.

These simulations represent a significant challenge for OLKMC. When the cluster is small, moves are highly concerted, with the transitions often resulting in all of the particles moving a significant distance as shown in Figure 4.2.

In Table 4.1, we record the efficiency $\mathcal{E}$ and the error E of Algorithm 2, as defined in the previous section, for clusters in Figure 4.1, along with the scaled physical time, CPU time in hours, and the number of hops at which these clusters are obtained. We see that the efficiency is close to 1/2 for the larger clusters, meaning that, on average, we reject about every other candidate event. Note that the error, as defined above, is not bounded by 1, and so the error values in the table do not represent percentages. The error is around 0.04 for the larger clusters. As described earlier, this can be improved upon, but at the cost of increased rejection, by providing larger rate estimates.

Table 4.1: The first two columns show the efficiency and error of Algorithm 2 for clusters in Figure 4.1, calculated via the rate estimate formula in Eq. (3.1) with $N_g = 200$ per atom. The last three columns are the scaled physical time, CPU time, and the number of hops at which these clusters were formed, respectively.

Subfig.	Efficiency $\mathcal{E}$	Error E	Scaled Phys. time	CPU time (hrs)	No. of hops
(a)	0.065	0.000	9.433	0.0985	139
(b)	0.25	0.007	23.680	2.436	872
(c)	0.32	0.000	42.796	13.812	2073
(d)	0.44	0.016	54.611	25.739	3081
(e)	0.44	0.011	71.050	63.756	4715
(f)	0.40	0.009	82.940	86.683	6153
(g)	0.45	0.028	94.000	142.453	7463
(h)	0.53	0.036	110.726	257.608	9796
(i)	0.54	0.041	114.679	309.312	10359

In Figure 4.3, we plot the number of events executed as a function of the scaled physical time for three realizations of the full OLKMC and the rejection-based algorithm as a means of demonstrating their near equivalence. Again, if one could achieve exhaustive saddle-point searches and strict bounds for the rate estimates, the two algorithms are stochastically equivalent. In Figure 4.4, we plot the CPU time as a function of the scaled physical time for this same set of realizations as a way of demonstrating the increased speed of the rejection scheme. For cluster sizes around fifty-five particles, the rejection algorithm is about ten times faster, and when they are around sixty-five particles, it is thirty times faster. This factor will continue to increase with larger cluster sizes, as the local search regions become a smaller fraction of the entire domain.

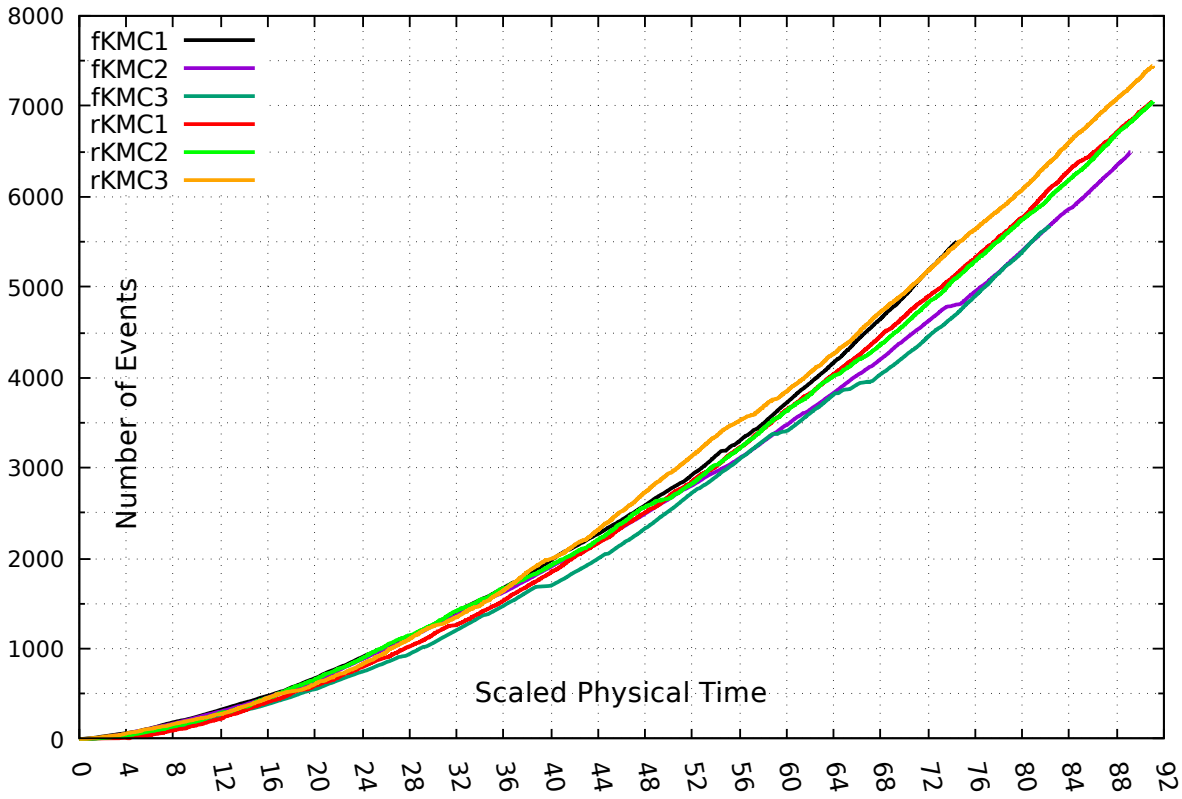


Figure 4.3: As a way of demonstrating the near equivalence of the two algorithms, this figure shows the number of events executed as a function of scaled physical time for three realizations of each algorithm. Different colors correspond to different seeds used for the random number generator.

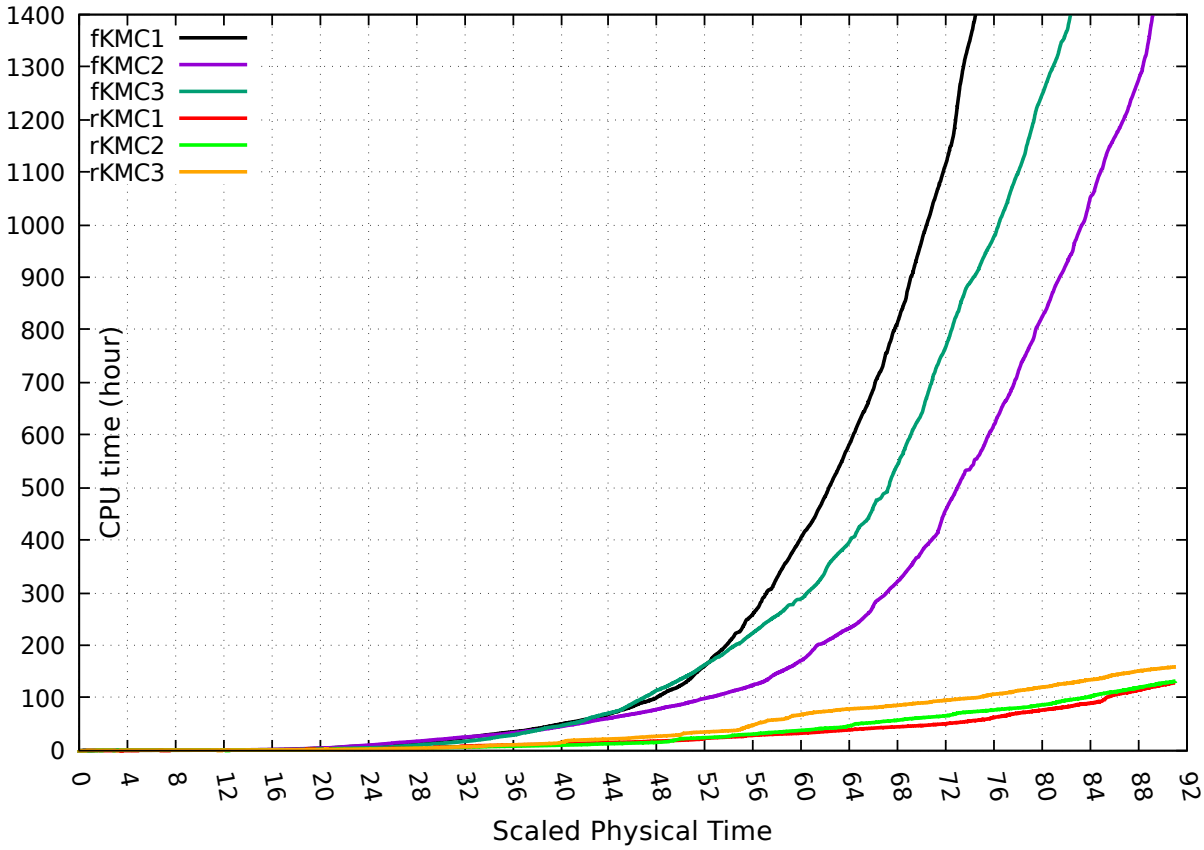


Figure 4.4: This graph shows the time needed by one CPU to perform both rejection and rejection-free (full) OLKMC schemes. The data plotted here is for the same realizations in Figure 4.3. When the system size is about fifty-five particles, the ratio of the black curve to the red and green curves is about ten and when it is about sixty-five particles, the ratio is about thirty.

Chapter 5

Conclusion

In this study we have demonstrated the viability and potential for using a rejection scheme to accelerate OLKMC. The main idea is to partition the set of rates that must be found in a way that allows for a local search procedure. If this can be achieved, one can expect an $O(N_p)$ boost in performance. The partitioning is accomplished by identifying moves with the atom that moves the furthest in the transition, and rate estimates rely on a notion of bond-counting similar to what one finds in the lattice-based KMC literature. There remain many unexplored variations on both of these approaches. For example, one could partition moves based on energy changes rather than distance moved, and rate estimates could incorporate prior information similar to the way saddle-point reconvergence/recycling has been used in other OLKMC work [11, 5, 42], especially when the size of the system is large. We also hope to come up with a perturbation scheme that will lead to a higher percent of connected key saddle points found around the basin of attraction.

Bibliography

[1] Allen, M. P. and Tildesley, D. J. (1987). *Computer Simulation of Liquids*, volume 18. Oxford University Press, New York. 38

[2] Boateng, H. A., Schulze, T. P., and Smereka, P. (2014). Approximating Off-Lattice Kinetic Monte Carlo. *Multiscale Model. Simul.*, 12(1):181–199. 3, 5

[3] Duncan, J., Wu, Q., Promislow, K., and Henkelman, G. (2014). Biased gradient squared descent saddle point finding method. *J. Chem. Phys.*, 140(19). 10

[4] E, W., Ren, W., and Vanden-Eijnden, E. (2002). String method for the study of rare events. *Phys. Rev. B*, 66(5):52301. 10

[5] El-Mellouhi, F., Mousseau, N., and Lewis, L. J. (2008). Kinetic activation-relaxation technique: An off-lattice self-learning kinetic Monte Carlo algorithm. *Phys. Rev. B - Condens. Matter Mater. Phys.*, 78(15). 45

[6] Eyring, H. (1935). The Activated Complex in Chemical Reactions. *J. Chem. Phys.*, 3(2):107–115. 5

[7] Gaynor, G., Carolina, N., Carson, C. L., Carolina, N., Hall, C. K., and Bernholc, J. (1990). Computer simulation studies. 42(5):2914–2922. 3

[8] Guo, W., Schulze, T. P., and Weinan, E. (2007). Simulation of impurity diffusion in a strained nanowire using off-lattice KMC. *Commun. Comput. Phys.*, 2(1):164–176. 3, 4

[9] Hänggi, P., Talkner, P., and Borkovec, M. (1990). Reaction-rate theory: Fifty years after Kramers. *Rev. Mod. Phys.*, 62(2):251–341. 6

[10] Henkelman, G. and Jónsson, H. (1999). A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives. *J. Chem. Phys.*, 111(15):7010–7022. 1, 4, 7, 10, 13, 15, 16, 17

[11] Henkelman, G. and Jónsson, H. (2001). Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *J. Chem. Phys.*, 115(21):9657–9666. 4, 5, 8, 45

[12] Henkelman, G., Uberuaga, B. P., and Jónsson, H. (2000). Climbing image nudged elastic band method for finding saddle points and minimum energy paths. *J. Chem. Phys.*, 113(22):9901–9904. 10

[13] Heyden, A., Bell, A. T., and Keil, F. J. (2005). Efficient methods for finding transition states in chemical reactions: Comparison of improved dimer method and partitioned rational function optimization method. *J. Chem. Phys.*, 123(22):224101. 1, 7, 11, 13, 15, 25

[14] Kara, A., Trushin, O., Yildirim, H., and Rahman, T. S. (2009). Off-lattice self-learning kinetic Monte Carlo: application to 2D cluster diffusion on the fcc(111) surface. *J. Phys. Condens. Matter*, 21(8):084213. 5

[15] Kästner, J. and Sherwood, P. (2008). Superlinearly converging dimer method for transition state search. *J. Chem. Phys.*, 128(1). 1, 7, 11, 15, 25, 26

[16] Konwar, D., Bhute, V. J., and Chatterjee, A. (2011). An off-lattice, self-learning kinetic Monte Carlo method using local environments. *J. Chem. Phys.*, 135(17):174103. 5

[17] Lam, C.-H., Lee, C.-K., and Sander, L. (2002). Competing Roughening Mechanisms in Strained Heteroepitaxy: A Fast Kinetic Monte Carlo Study. *Phys. Rev. Lett.*, 89(21):216102. 3

[18] Latz, A., Brendel, L., and Wolf, D. E. (2013). Erratum: A three-dimensional self-learning kinetic Monte Carlo model: Application to Ag(111) (Journal of Physics Condensed Matter (2012) 24 (485005)). *J. Phys. Condens. Matter*, 25(5). 5

[19] Lorentz, H. A. (1881). Ueber die Anwendung des Satzes vom Virial in der kinetischen Theorie der Gase. *Ann. Phys.*, 248(1):127–136. 38

[20] Lung, M. T., Lam, C. H., and Sander, L. M. (2005). Island, pit, and groove formation in strained heteroepitaxy. *Phys. Rev. Lett.*, 95(8):86102. 3

[21] Malek, R. and Mousseau, N. (2000). Dynamics of lennard-jones clusters: A characterization of the activation-relaxation technique. *Phys. Rev. E*, 62(6 Pt A):7723–8. 10

[22] Miron, R. A. and Fichthorn, K. A. (2004). Multiple-time scale accelerated molecular dynamics: Addressing the small-barrier problem. *Phys. Rev. Lett.*, 93(12):128301. 9

[23] Mitchell, I., Irle, S., and Page, A. J. (2016). A global reaction route mapping-based kinetic Monte Carlo algorithm. *J. Chem. Phys.*, 145(2):24105. 8

[24] Much, F., Ahr, M., Biehl, M., and Kinzel, W. (2002). A kinetic Monte Carlo method for the simulation of heteroepitaxial growth. In *Comput. Phys. Commun.*, volume 147, pages 226–229. 3, 4

[25] Munro, L. and Wales, D. (1999). Defect migration in crystalline silicon. *Phys. Rev. B*, 59(6):3969–3980. 10

[26] Nandipati, G., Kara, A., Shah, S. I., and Rahman, T. S. (2012). Off-lattice pattern recognition scheme for kinetic Monte Carlo simulations. *J. Comput. Phys.*, 231(9):3548–3560. 5

[27] Olsen, R. A., Kroes, G. J., Henkelman, G., Arnaldsson, A., and Jónsson, H. (2004). Comparison of methods for finding saddle points without knowledge of the final states. *J. Chem. Phys.*, 121(20):9776–9792. 7, 10, 13

[28] Plotz, W. M., Hingerl, K., and Sitter, H. (1992). Monte Carlo simulation of epitaxial growth. *Phys. Rev. B*, 45(20):12122–12125. 3

[29] Russo, G. and Smereka, P. (2006a). A Multigrid-Fourier Method for the Computation of Elastic Fields with Application to Heteroepitaxy. *Multiscale Model.*, 5(1):130–148. 3

[30] Russo, G. and Smereka, P. (2006b). Computation of strained epitaxial growth in three dimensions by kinetic Monte Carlo. *J. Comput. Phys.*, 214(2):809–828. 3

[31] Schulze, T. P. and Smereka, P. (2009). An energy localization principle and its application to fast kinetic Monte Carlo simulation of heteroepitaxial growth. *J. Mech. Phys. Solids*, 57(3):521–538. 3, 31

[32] Schulze, T. P. and Smereka, P. (2011). Simulation of Three-Dimensional Strained Heteroepitaxial Growth Using Kinetic Monte Carlo. *Commun. Comput. Phys.*, 10(5):1089–1112. 3, 31

[33] Schulze, T. P. and Smereka, P. (2012). Kinetic Monte Carlo simulation of heteroepitaxial growth: Wetting layers, quantum dots, capping, and nanorings. *Phys. Rev. B - Condens. Matter Mater. Phys.*, 86(23). 3, 31

[34] Shang, C. and Liu, Z. P. (2010). Constrained broyden minimization combined with the dimer method for locating transition state of complex reactions. *J. Chem. Theory Comput.*, 6(4):1136–1144. 1, 11

[35] Spjut, H. and Faux, D. A. (1994). Computer simulation of strain-induced diffusion enhancement of Si adatoms on the Si(001) surface. *Surf. Sci.*, 306(1-2):233–239. 3

[36] Truhlar, D. G., Hase, W. L., and Hynes, J. T. (1983). Current status of transition-state theory. *J. Phys. Chem.*, 87(15):2664–2682. 5

[37] Vineyard, G. H. (1957). Frequency factors and isotope effects in solid state rate processes. *J. Phys. Chem. Solids*, 3(1-2):121–127. 6

[38] Voter, A. F. and Doll, J. D. (1985). Dynamical corrections to transition state theory for multistate systems: Surface self-diffusion in the rare-event regime. *J. Chem. Phys.*, 82(1):80–92. 6

[39] Wales, D. J. and Doye, J. P. K. (2003). Stationary points and dynamics in high-dimensional systems. *J. Chem. Phys.*, 119(23):12409–12416. 10

[40] Xu, H., Osetsky, Y. N., and Stoller, R. E. (2011). Simulating complex atomistic processes: On-the-fly kinetic Monte Carlo scheme with selective active volumes. *Phys. Rev. B - Condens. Matter Mater. Phys.*, 84(13):1–4. 5

[41] Xu, H., Osetsky, Y. N., and Stoller, R. E. (2012). Self-evolving atomistic kinetic MonteCarlo: Fundamentals and applications. *J. Phys. Condens. Matter*, 24(37). 5

[42] Xu, L. and Henkelman, G. (2008). Adaptive kinetic Monte Carlo for first-principles accelerated dynamics. *J. Chem. Phys.*, 129(11). 5, 45

Appendices

Appendix A

Potentials

A Lennard-Jones Potential

$$U(\boldsymbol{x}) = \sum_i \sum_{j>i} \phi(r_{ij}) = \frac{1}{2} \sum_i \sum_{j \neq i} \phi(r_{ij}), \text{ where } \phi(r_{ij}) = 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right],$$

$\boldsymbol{x} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) \in \mathbb{R}^{3N}$ where $\boldsymbol{x}_i = (x_i^1, x_i^2, x_i^3)$ is the position of atom i in $\mathbb{R}^3$ and $r_{ij} = \sqrt{(x_i^1 - x_j^1)^2 + (x_i^2 - x_j^2)^2 + (x_i^3 - x_j^3)^2}$ is the distance between atoms i and j, $\phi(r_{ij})$ is the interaction potential, σ_{ij} is the finite distance at which the interaction potential is zero and ϵ_{ij} is the chemical bond energy. The bond length is $r_{ij}^m = \sqrt[6]{2}\sigma_{ij}$. The first and second derivatives of the interaction potential with respect to the distance r_{ij} are

$$\phi'(r_{ij}) = 6\eta_{ij} \left[\frac{1}{r_{ij}^7} - \frac{2\sigma_{ij}^6}{r_{ij}^{13}} \right] \quad \text{and} \quad \phi''(r_{ij}) = 6\eta_{ij} \left[\frac{26\sigma_{ij}^6}{r_{ij}^{14}} - \frac{7}{r_{ij}^8} \right],$$

where $\eta_{ij} = 4\epsilon_{ij}\sigma_{ij}^6$. We also have

$$\frac{\partial r_{ij}}{\partial x_k^n} = \frac{(x_i^n - x_j^n)(\delta_{ik} - \delta_{jk})}{r_{ij}},$$

where $\delta_{\alpha\beta}$ is the Kronecker delta. The gradient vector is denoted by $\boldsymbol{g}(\boldsymbol{x}) = (\boldsymbol{g}_1, \cdots, \boldsymbol{g}_N)$, where $\boldsymbol{g}_i = (g_i^1, g_i^2, g_i^3)$. We calculate the first derivative of U as

$$\begin{aligned}
\frac{\partial U}{\partial x_k^n} &= \frac{1}{2}\sum_i \sum_{j\neq i} \frac{\partial \phi(r_{ij})}{\partial x_k^n} \\
&= \frac{1}{2}\sum_i \sum_{j\neq i} \phi'(r_{ij}) \frac{(x_i^n - x_j^n)(\delta_{ik} - \delta_{jk})}{r_{ij}} \\
&= \frac{1}{2}\sum_i \sum_{j\neq i} \phi'(r_{ij}) \frac{(x_i^n - x_j^n)\delta_{ik}}{r_{ij}} + \frac{1}{2}\sum_i \sum_{j\neq i} \phi'(r_{ij}) \frac{(x_j^n - x_i^n)\delta_{jk}}{r_{ij}} \\
&= \frac{1}{2}\sum_{j\neq k} \phi'(r_{kj}) \frac{(x_k^n - x_j^n)}{r_{kj}} + \frac{1}{2}\sum_{i\neq k} \phi'(r_{ik}) \frac{(x_k^n - x_i^n)}{r_{ik}} \\
&= \sum_j \phi'(r_{kj}) \frac{(x_k^n - x_j^n)}{r_{kj}} (1 - \delta_{kj}) \\
&= 6\sum_j \eta_{kj} \left[\frac{1}{r_{kj}^8} - \frac{2\sigma_{kj}^6}{r_{kj}^{14}} \right] (x_k^n - x_j^n)(1 - \delta_{kj})
\end{aligned}$$

Hence, the k^{th} component of the gradient is

$$\boldsymbol{g}_k = \nabla_{\boldsymbol{x}_k} U(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) = \sum_{n=1}^{3} \frac{\partial U}{\partial x_k^n} \hat{\boldsymbol{e}}_n = 6\sum_{\substack{j=1 \\ j\neq k}}^{N} \eta_{kj} \left[\frac{1}{r_{kj}^8} - \frac{2\sigma_{kj}^6}{r_{kj}^{14}} \right] (\boldsymbol{x}_k - \boldsymbol{x}_j). \tag{A.1}$$

The $3N \times 3N$ Hessian matrix H has components of the form $H_{k\ell}^{nm} = \partial^2 U / \partial x_k^n \partial x_\ell^m$, where $k, \ell = 1, \cdots, N$ and $m, n = 1, 2, 3$. The second derivative of U is calculated as

$$\begin{aligned}
\frac{\partial^2 U}{\partial x_k^n \partial x_\ell^m} &= 6\frac{\partial}{\partial x_k^n} \left(\sum_j \eta_{\ell j} \left[\frac{1}{r_{\ell j}^8} - \frac{2\sigma_{\ell j}^6}{r_{\ell j}^{14}} \right] (x_\ell^m - x_j^m)(1 - \delta_{\ell j}) \right) \\
&= 6\sum_j \eta_{\ell j} \Bigg\{ \left[\frac{1}{r_{\ell j}^8} - \frac{2\sigma_{\ell j}^6}{r_{\ell j}^{14}} \right] (\delta_{k\ell} - \delta_{kj})(1 - \delta_{\ell j})\delta_{mn} + \\
&\qquad \left[\frac{28\sigma_{\ell j}^6}{r_{\ell j}^{15}} - \frac{8}{r_{\ell j}^9} \right] \frac{(x_\ell^n - x_j^n)(\delta_{\ell k} - \delta_{jk})}{r_{\ell j}} (x_\ell^m - x_j^m)(1 - \delta_{\ell j}) \Bigg\} \\
&= 6\sum_j \eta_{\ell j}(1 - \delta_{\ell j})(\delta_{k\ell} - \delta_{kj}) \Bigg\{ \left[\frac{1}{r_{\ell j}^8} - \frac{2\sigma_{\ell j}^6}{r_{\ell j}^{14}} \right] \delta_{mn} +
\end{aligned}$$

$$\left[\frac{28\sigma_{\ell j}^6}{r_{\ell j}^{16}} - \frac{8}{r_{\ell j}^{10}}\right](x_\ell^n - x_j^n)(x_\ell^m - x_j^m)\Bigg\}$$

$$= 6\delta_{k\ell}\sum_{\substack{j=1\\ j\neq\ell}}^{N}\eta_{\ell j}\Bigg\{\left[\frac{1}{r_{\ell j}^8} - \frac{2\sigma_{\ell j}^6}{r_{\ell j}^{14}}\right]\delta_{mn} + \left[\frac{28\sigma_{\ell j}^6}{r_{\ell j}^{16}} - \frac{8}{r_{\ell j}^{10}}\right](x_\ell^n - x_j^n)(x_\ell^m - x_j^m)\Bigg\}$$

$$- 6(1-\delta_{k\ell})\eta_{\ell k}\Bigg\{\left[\frac{1}{r_{\ell k}^8} - \frac{2\sigma_{\ell k}^6}{r_{\ell k}^{14}}\right]\delta_{mn} + \left[\frac{28\sigma_{\ell k}^6}{r_{\ell k}^{16}} - \frac{8}{r_{\ell k}^{10}}\right](x_\ell^n - x_k^n)(x_\ell^m - x_k^m)\Bigg\} \quad \text{(A.2)}$$

The Hessian is a real symmetric matrix that looks like:

$$\left[\begin{array}{ccc|ccc|c}
H_{11}^{11} & H_{11}^{12} & H_{11}^{13} & H_{12}^{11} & H_{12}^{12} & H_{12}^{13} & \\
H_{11}^{21} & H_{11}^{22} & H_{11}^{23} & H_{12}^{21} & H_{12}^{22} & H_{12}^{23} & \cdots \\
H_{11}^{31} & H_{11}^{32} & H_{11}^{33} & H_{12}^{31} & H_{12}^{32} & H_{12}^{33} & \\
\hline
 & & & H_{22}^{11} & H_{22}^{12} & H_{22}^{13} & \\
 & \vdots & & H_{22}^{21} & H_{22}^{22} & H_{22}^{23} & \cdots \\
 & & & H_{22}^{31} & H_{22}^{32} & H_{22}^{33} & \\
\hline
 & \vdots & & & \vdots & & \ddots
\end{array}\right]$$

Vita

Hamza Ruzayqat was born and raised in a small town in Hebron city in Palestine. He attended elementary and high school in his town, Taffouh. After graduating high school, Hamza enrolled in Birzeit University where he received a bachelors in physics and a minor in mathematics. It was at Birzeit University that Hamza made the decision to pursue higher education in America and study towards a PhD in mathematics. He was admitted to the Math Department at the University of Tennessee, Knoxville where he received teaching and research assistantships for the duration of his graduate studies. It was at UTK that Hamza developed a love for research and teaching. Hamza's research under the supervision of Dr. Tim Schulze was on the rejection off-lattice kinetic Monte Carlo method. Hamza has a very diverse teaching experience both as an instructor and a teaching assistant here in the US and abroad.