To my mother
ACKNOWLEDGMENTS

I thank Lauren, my wife, for putting up with me all these years, and my father for always being there for me. My colleagues in the Roitberg group, past and present, have provided numerous insights throughout the years and helped a physicist to learn at least a small part of chemistry and biology. I would like to thank my dissertation committee of Erik Deumens, Stephen Hagen, and Selman Hershfield, as well as my advisor and committee chair, Adrian Roitberg.
TABLE OF CONTENTS

ACKNOWLEDGMENTS............................................................................................................. 4
LIST OF TABLES.......................................................................................................................... 7
LIST OF FIGURES....................................................................................................................... 8
LIST OF ABBREVIATIONS........................................................................................................... 10
ABSTRACT .................................................................................................................................. 11
CHAPTER
1 INTRODUCTION ...................................................................................................................... 13
    Proteins..................................................................................................................................... 13
    Electronic Structure .................................................................................................................. 15
    Molecular Mechanics ............................................................................................................... 16
    Force Field Parameterization ................................................................................................. 17
    Molecular Dynamics............................................................................................................... 18
    Hybrid Simulations.................................................................................................................. 19
2 FORCE FIELD PARAMETERIZATION: DIHEDRAL FITTING ...................................................... 22
    Background............................................................................................................................ 22
    Linear least squares fitting of dihedral parameters ............................................................... 25
    Methods .................................................................................................................................. 29
        Validation........................................................................................................................... 30
        Chiral Molecule Parameterization ..................................................................................... 32
    Results ................................................................................................................................... 32
        Validation – butane .............................................................................................................. 32
        Validation – 1-butanol ......................................................................................................... 33
        Chiral Molecule Parameterization ..................................................................................... 35
    Discussion ............................................................................................................................... 36
3 FORCE FIELD PARAMETERIZATION: SAMPLING .................................................................. 45
    Background ............................................................................................................................. 45
    Methods .................................................................................................................................. 46
    Results and Discussion .......................................................................................................... 49
4 HYDROGEN MASS REPARTITIONING ..................................................................................... 63
    Background ............................................................................................................................. 63
    Methods .................................................................................................................................. 66
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Butane C-C-C-C (c3-c3-c3-c3 in GAFF) dihedral parameters</td>
<td>42</td>
</tr>
<tr>
<td>2-2</td>
<td>1-Butanol C-C-C-C (c3-c3-c3-c3 in GAFF) dihedral parameters</td>
<td>42</td>
</tr>
<tr>
<td>2-3</td>
<td>1-Butanol C-C-C-O (c3-c3-c3-oh in GAFF) dihedral parameters</td>
<td>43</td>
</tr>
<tr>
<td>2-4</td>
<td>C-c3-ca-ca dihedral parameters fitted via χ in amoxicillin</td>
<td>43</td>
</tr>
<tr>
<td>2-5</td>
<td>C-c3-ca-ca dihedral parameters fitted via χ in an amoxicillin isomer</td>
<td>44</td>
</tr>
<tr>
<td>3-1</td>
<td>RMSEs between fitted MM PES and QM PES for 1-butanol</td>
<td>61</td>
</tr>
<tr>
<td>3-2</td>
<td>RMSEs between fitted MM PES and QM PES for 1-butanol, using the MD cluster polar gridded sampling method</td>
<td>62</td>
</tr>
<tr>
<td>5-1</td>
<td>Number of basis functions for each studied system for the DZVP basis set, along with number of each angular momentum type</td>
<td>118</td>
</tr>
<tr>
<td>5-2</td>
<td>Runtimes and speedups for CopA and MshA</td>
<td>118</td>
</tr>
<tr>
<td>5-3</td>
<td>Speedups in the Coulomb gradients for the five studied systems, and energy difference between CPU and GPU single point calculations</td>
<td>119</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Test systems for the LLS method</td>
<td>38</td>
</tr>
<tr>
<td>2-2</td>
<td>Potential energy scans and error in scans for butane</td>
<td>39</td>
</tr>
<tr>
<td>2-3</td>
<td>2-D potential energy scans of C-C-C-C and C-C-C-O dihedrals in 1-butanol</td>
<td>40</td>
</tr>
<tr>
<td>2-4</td>
<td>Dihedral potential energy scans for amoxicillin and its isomer</td>
<td>41</td>
</tr>
<tr>
<td>3-1</td>
<td>Population density for 1-butanol</td>
<td>55</td>
</tr>
<tr>
<td>3-2</td>
<td>Configurations selected for the first MD cluster random sampling on 1-butanol with ( n = 10 ).</td>
<td>56</td>
</tr>
<tr>
<td>3-3</td>
<td>Configurations selected for MD cluster polar-gridded sampling on 1-butanol with ( r = 3 ) and ( \theta = 5 ).</td>
<td>56</td>
</tr>
<tr>
<td>3-4</td>
<td>Sample molecules used in testing of random configuration sampling for dihedral parameterization.</td>
<td>57</td>
</tr>
<tr>
<td>3-5</td>
<td>RMSEs between fitted MM PES and QM PES for 1-butanol</td>
<td>58</td>
</tr>
<tr>
<td>3-6</td>
<td>RMSEs between fitted MM PES and QM PES for various molecules using configurations randomly selected off of a grid</td>
<td>59</td>
</tr>
<tr>
<td>3-7</td>
<td>Two-dimensional RMSEs between fitted MM PES and QM PES for various molecules using configurations randomly selected off of an 18x18 point grid in dihedral space</td>
<td>60</td>
</tr>
<tr>
<td>3-8</td>
<td>RMSEs between fitted MM PES and QM PES for a 3-dimensional fit</td>
<td>61</td>
</tr>
<tr>
<td>4-1</td>
<td>Example of the HMR scheme used in this study applied to alanine dipeptide</td>
<td>80</td>
</tr>
<tr>
<td>4-2</td>
<td>Structures of systems studied</td>
<td>81</td>
</tr>
<tr>
<td>4-3</td>
<td>Population statistics of central ( \phi/\psi ) angles in the ((\text{Ala})_3) peptide for different time step and mass topology combinations</td>
<td>82</td>
</tr>
<tr>
<td>4-4</td>
<td>( \phi/\psi ) mask derived from norm/1 fs free energies</td>
<td>83</td>
</tr>
<tr>
<td>4-5</td>
<td>Fractional populations of the most populated rotamers in the ((\text{Ala})_3) peptide, averaged over ten 450 ns trajectories</td>
<td>84</td>
</tr>
<tr>
<td>4-6</td>
<td>Average transition rates between rotamers with highest transition frequency in ((\text{Ala})_3) peptide</td>
<td>85</td>
</tr>
</tbody>
</table>
Average potential energies with different mass topology / time step combinations for the (Ala)$_3$ peptide .......................................................... 86
Backbone RMSD to a crystal structure in HEWL for each trajectory type. ........ 87
Backbone RMSD and C$_\alpha$ RMS fluctuations of HEWL ................................ 88
Average potential energies with different mass topology / time step combinations for HEWL .......................................................... 89
Terminal-terminal distance for HEWL averaged over each type of trajectory .... 90
Projections of HEWL trajectories onto top 4 eigenvectors from PCA on the protein backbone of a common norm/1 fs trajectory ........................................ 90
Plot of c vs N (see text) averaged over each type of trajectory .................. 91
Evolution of calculated pK$_a$ values using constant pH MD on HEWL for two different trajectory types in three of the ten residues treated .................. 92
PMF for rotation about the dihedral $\chi$ in the MTS spin label, generated through umbrella sampling .......................................................... 93
Graphic representation of a round of the SHAKE algorithm ...................... 93
SIMT architecture and CUDA kernel execution model .............................. 113
Relative computation times from each part of LIO runtime for single-core optimized CPU version and GPU accelerated version .......................... 113
Example of implemented thread mapping for QM/MM and Coulomb Kohn-Sham matrix evaluation .......................................................... 114
Example of implemented thread mapping for QM/MM force evaluation ....... 114
Protein systems studied .......................................................................... 115
Systems studied in vacuum ..................................................................... 116
Scaling analysis of QM/MM gradient runtime versus MM region size ......... 116
Defined distances between the UDP-GlcNAc donor (right) and m-Ins-1P acceptor (left) in the MshA-catalyzed reaction ......................................... 117
Free energy profile and distances over the course of the reaction pathway in the MshA-catalyzed reaction .......................................................... 117
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFT</td>
<td>Density functional theory</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic algorithm</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics processing unit</td>
</tr>
<tr>
<td>HEWL</td>
<td>Hen egg white lysozyme</td>
</tr>
<tr>
<td>HF</td>
<td>Hartree-Fock</td>
</tr>
<tr>
<td>LLS</td>
<td>Linear least squares</td>
</tr>
<tr>
<td>MD</td>
<td>Molecular dynamics</td>
</tr>
<tr>
<td>MM</td>
<td>Molecular mechanics</td>
</tr>
<tr>
<td>PES</td>
<td>Potential energy surface</td>
</tr>
<tr>
<td>PMF</td>
<td>Potential of mean force</td>
</tr>
<tr>
<td>QM</td>
<td>Quantum mechanics</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular value decomposition</td>
</tr>
</tbody>
</table>
This document presents my work in the field of computational physical chemistry in the Roitberg group on a variety of projects that all have the overarching theme of improving and optimizing computational techniques in the field. Advances in computer hardware over the past decades have unlocked the ability to simulate much larger chemical systems for larger simulation times, with higher levels of physical theory, than were possible when the base algorithms and theoretical approximations that drive the core of our work were developed in the mid to late 20th century. This has led to the need to revisit and improve on existing techniques so that they might fully harness the computational power available.

Chapters 2 and 3 address optimizations to a particular problem in force field parameterization, the fitting of dihedral parameters. Chapter 2 presents a novel fitting procedure to analytically calculate the optimal set of dihedral force field parameters to reproduce a particular energy surface from high level theoretical data. Chapter 3 investigates the minimal number of high level calculations needed in order to derive a high quality dihedral parameter set.
Chapter 4 presents a simple method of accelerating molecular dynamics simulations by about a factor of two, called hydrogen mass repartitioning. By changing the system mass in a simulation, the highest frequency motions of the system can be slowed, allowing a stable increase in simulation time step.

Chapter 5 investigates the use of graphics processing unit accelerator hardware to speed up density functional theory calculations. In particular, the focus in the acceleration effort was in the use of full density functional theory in hybrid multiscale simulations in order to study enzyme catalysis.
CHAPTER 1
INTRODUCTION

Proteins

At the smallest scales of biological systems are found the proteins and nucleic acids, the two main classes of macromolecules that drive the chemistry of the living cell. Proteins in particular are involved in a variety of tasks within biological systems, including cell signaling,\(^1\) contributing to cell structure,\(^2\) and acting as enzymes in the biosynthesis pathway for important molecules within the cell.\(^3\) Knowledge of the mechanisms of the various roles played by these macromolecules, as well as the structure of the molecules themselves, is important in the understanding and control of the emergent biological activity of which they form the basis, such as disease, the body’s immune response, injury and recovery, and aging.

Computer simulations have developed over the past decades into a powerful complementary tool to experimental techniques in the study of protein structure and dynamics. Experimental methods such as X-ray crystallography,\(^4\) nuclear magnetic resonance (NMR),\(^5\) and circular dichroism (CD)\(^6\) have successfully been used to elucidate the three-dimensional structure of proteins as well as to provide a picture of relatively slower dynamics; however, their time resolutions range from picoseconds to nanoseconds, and include results from multiple molecules, and thus the data they return represents an averaged picture of the protein. Similarly, kinetics assays and calorimetric experiments can be used to investigate the kinetics and thermodynamics of enzyme catalyzed reactions, but they usually cannot provide a definitive answer to the exact mechanisms of the reaction. Computer simulations serve an increasingly important role in closing the gap here by allowing the study of finer length and time
scales than accessible by experiment. For example, protein simulations are often combined with experimental data to produce three-dimensional structures\textsuperscript{7,8} and have been used extensively in rational drug design and discovery.\textsuperscript{9} Computational techniques have been successfully employed to study the binding of ligands with proteins and the calculation of binding free energies,\textsuperscript{10–12} and to study the environment pH effect on protein activity.\textsuperscript{13–15} Simulations using high levels of theory are used to study enzyme activity, allowing the calculation of reaction free energy profiles and the elucidation of the reaction mechanism.\textsuperscript{16,17}

The present study is focused on the optimization of several techniques used in physics-based atomic level protein simulations. One of the major hurdles in the simulation of biological macromolecules is computational runtime. Many efficient techniques and algorithms have been developed over the past half-century or so to calculate molecular properties on protein systems. However, when modeling a protein system at the atomic level, and particularly when explicitly modeling the solvent around the protein, a typical system is composed of tens or hundreds of thousands of atoms. Even when using simple physics models, the interaction between each pair of atoms needs to be accounted for, meaning the problem size scales as at least $n^2$, where $n$ is the system size. On top of this, the calculation of free energies and other equilibrium properties does not involve the energy evaluation for a single configuration of the system, but rather the time or ensemble average taken over a large number of system snapshots, and thus a large number of energy evaluations. Therefore, the acceleration and optimization of protein simulations is an important problem and a very active field of research, with solutions coming in the form of further algorithmic improvements\textsuperscript{10,18–22}.
as well as acceleration using new hardware paradigms. In Chapters 2-4, this study presents algorithmic optimizations in the areas of molecular mechanics force field parameterization and in molecular dynamics for use in protein structure and dynamics studies. Chapter 5 presents an acceleration method using the relatively recent shift to hardware acceleration via graphics processing units for the study of enzyme catalysis.

**Electronic Structure**

An appropriate treatment of the atomistic simulation of a chemical system generally requires a quantum mechanical (QM) approach to model the system, directly modeling the wave function of the electrons. For most practical purposes, especially for the type of chemistry seen in biological systems, a QM treatment of equilibrium properties refers to the solution of the non-relativistic Schrödinger equation for the ground state of the electrons in the presence of a near-stationary (adiabatic) nuclear background, or the Born-Oppenheimer approximation. There are two main types of solutions in use for this problem, both of which generally use a variational type solution to the Schrödinger equation: Hartree-Fock and density functional theory. The Hartree-Fock (HF) method, which serves as the basis for many current QM methods, scales formally as $O(n^4)$, where $n$ is the number of basis functions for the wave function used. More accurate post-HF methods, like coupled cluster (CC), have increased formal computational complexity: $O(n^6)$ for CCSD and $O(n^8)$ for full CCSDT. Density functional theory (DFT) offers an efficient alternative to *ab initio* methods, though the full base scaling for a DFT method is still $O(n^4)$. Further approximations, such as the use of the Schwarz inequality, allow optimizations beyond these base limits. Many so-called linear scaling methods have been developed that allow efficient computation of the DFT problem. Still, the size of systems that can be treated with full QM has been
constrained until recently to hundreds of atoms. When considering problems that require multiple QM calculations (e.g., statistical dynamics simulations), full *ab initio* or DFT methods have, until recent years, not been practical. Semi-empirical methods, for example, offered a computationally cheaper alternative, at the price of chemical accuracy.

**Molecular Mechanics**

In order to be able to do atomistic simulations of large macromolecules, the typical model Hamiltonian uses molecular mechanics (MM), treating the system with a classical force field. In the force field model, the energy of the system is described using an empirical potential. The so-called “Class I” type force field used in the Amber\(^{31}\) and CHARMM\(^{32}\) programs is still predominantly used, especially in the study of biochemical systems:

\[
V(r) = \sum_{\text{bonds}} K_b (b - b_0)^2 + \sum_{\text{angles}} K_\theta (\theta - \theta_0)^2 + \sum_{\text{nombij}} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} + \frac{q_i q_j}{r_{ij}} \right) + \sum_{D \in \text{dihedrals}} \sum_{n=1}^{n_{\text{max}}} \frac{C_{nk}}{2} [1 + \cos(n \phi_k - \Delta_{nk})]
\]

This type of potential describes chemical bonds using a simple harmonic interaction (as well as angles between bonds), models non-bonded interactions with a short-range van der Waals term and long-range electrostatics with a simple point charge interaction, and includes a dihedral (four atom) term to pick up interactions not described by the other empirical terms. The “force field” in this case is comprised of the collection of constant parameters in the energy description (spring constants, equilibrium lengths and angles, point charge values, etc). The MM paradigm has allowed the efficient simulation of many aspects of macromolecule dynamics. With
modern computers and implementations, the classical force field approximation of MM makes feasible the routine evaluation of configuration energies for tens or even hundreds of thousands of atoms. When studying processes that do not involve large changes in the electronic density (e.g., large-scale fluctuations in the structure of a protein), an MM Hamiltonian is sufficient to accurately describe the energy landscape of the system, as long as the force field is accurately parameterized.

**Force Field Parameterization**

In order for a force field to make any physical sense, it must be parameterized against physically accurate data – this can include simulations of higher levels of theory (\textit{ab initio} or DFT) or experimental data. A force field is typically parameterized to reproduce certain desired observable molecular properties. As such, force fields are typically valid within a limited chemical space, leading to a different force field for carbohydrates, for lipids, for proteins and nucleic acids, and so on.

New force field parameters are constantly being published as new classes of molecules are studied\textsuperscript{33–37} and as existing parameter sets are improved and optimized to better agree with experimental and high level data.\textsuperscript{38–46} This is especially true for the "general" force fields (e.g., GAFF\textsuperscript{47} and CGenFF\textsuperscript{48}, which are supposed to apply to wide classes of molecules (e.g., drug-like molecules). While the exact parameterization process generally varies from author to author, standard techniques and philosophies, with respect to a specific force field, have been published.\textsuperscript{48,49} In general, the equilibrium values for the harmonic terms (bonds and angles) are closely based on experimental geometries, while the corresponding force constants can be linked to vibrational modes (from experimental and high level \textit{ab initio} data). There are standard
techniques for calculating new partial atomic charges (e.g., RESP\textsuperscript{50} and AM1-BCC)\textsuperscript{51,52} and for calculating Lennard-Jones parameters based on high level \textit{ab initio} data.

Meanwhile, the dihedral terms tend to be the least physically motivated in the force field. They serve primarily as correction terms, picking up as much of the error inherent in the force field approximation (due to neglect of quantum many-body effects, polarization, etc.) as possible. Due to this role as corrections, the optimal values of the dihedral parameters depend rather strongly on the other force field parameters, particularly the non-bonded terms. As such, they tend to be computed in the last step of the parameterization process. Dihedral parameters are often used to optimize existing force fields by fitting them to a target data set; relative conformational energies are a common target. Force fields are frequently re-parameterized, or specialized for a particular class of molecules, by varying the parameters of important dihedral angles to better agree with high-level \textit{ab initio} conformational energies about those angles.\textsuperscript{39,40,45,53,54} In Chapters 2 and 3, we look at methods of optimizing this dihedral fitting procedure to be faster and more accurate.

**Molecular Dynamics**

With a Hamiltonian in hand, be it based on theoretically sound quantum mechanics, or on an empirically built force field, the simulation of a chemical system next needs a statistical method to sample from an ensemble that allows the equilibrium properties of the system to be calculated. The two main methods here are Monte Carlo and molecular dynamics (MD). Whereas Monte Carlo methods directly sample from the entire energy landscape of the Hamiltonian being used at each step, MD works in small local movements about the landscape based on classical equations of motion. Monte Carlo tends to be a preferable method in less dense systems when large barrier
crossings are a concern, but MD tends to be favored when simulating larger, denser systems (liquids) with many atomic collisions and many potential system movements. In addition, MD directly introduces a time parameter, allowing the direct calculation of transport properties. Thus, in this study, we are focused on MD as the method of choice.

MD effectively works by numerically integrating Newton’s equations of motion:

\[ F = -\nabla V = ma \]

The forces used are derived from the gradients of the potential in use, be it a closed form force field potential, or the energy surface of the ground electronic state from an \textit{ab initio} or DFT method.

For conventional MD simulations of large biomolecules in explicit solvent with MM, the edge of the current simulation timescale available for routine access with modest computational resources lies roughly in the hundreds of nanoseconds, with the recent introduction of graphics processing unit (GPU) accelerated code pushing this limit towards microseconds.\textsuperscript{18,24,25,55} Running for this amount of simulation time still involves a substantial real time investment. In addition, multiple long trajectories are often required for an appropriate statistical analysis for large systems, and many processes of biological importance take place on the timescale of at least hundreds of microseconds.\textsuperscript{55,56} Thus, improving on this “sampling problem” of MD is an important challenge that has been faced with a variety of techniques (see, for example, a recent review by Christ et al.\textsuperscript{19} and the recent work by Hansen and Hünenberger,\textsuperscript{20} and references cited therein). In Chapter 4 we look at such a technique, which involves
changing the system mass in order to increase the size of the time step of the MD simulation and effectively accelerate the simulation by a factor of two.

**Hybrid Simulations**

We have mentioned two main types of simulations at this point, those that involve a large change in the electronic density during the simulation, and thus require QM to appropriately model the system, and those that do not, and so can use the computationally cheaper MM to simulate a large system. However, there are some cases where the system lies in between these two extremes, where a small part of the system undergoes a large change in electronic density, while the larger part stays electronically stable. In these cases, it would be advantageous to combine the two, which is where the method of QM/MM comes in. QM/MM provides a hybrid, multiscale method where the (smaller) reactive / electronically dynamic part of the system is modeled using computationally expensive *ab-initio* or DFT methods, while the (larger) electronically stable region is treated with MM. A particularly relevant example where this duality can be applied is in the study of enzymes and the reactions that they catalyze. When a protein acts as an enzyme, a small part of the protein, called the active site, is involved in a chemical reaction, while the rest of the protein is not. This division of active site and reactants / rest of protein and solvent maps very well onto the QM/MM paradigm. As such, with computer hardware gaining in power in the past decade and allowing the QM simulation of larger and larger systems, QM/MM is becoming a very popular technique to study enzymatic activity. In Chapter 5, we look towards QM/MM and its use in simulating enzyme activity, and a method of speeding up these types of simulations. Modern hardware advances have unlocked the ability to
simulate larger systems for longer times, and in particular GPUs have proven particularly powerful in the past years for accelerating QM calculations. \textsuperscript{26,58–64}
CHAPTER 2
FORCE FIELD PARAMETERIZATION: DIHEDRAL FITTING

Background

The base force field form used in many molecular mechanics programs (e.g, Amber, CHARMM, and OPLS) uses a truncated Fourier series to represent the dihedral potential energy for a single dihedral angle $\theta$:

$$V(\alpha, \delta, \theta) = \sum_{n=1}^{n_{\text{max}}} \alpha_n [1 + \cos(n\theta - \delta_n)]$$  \hspace{1cm} (2-1)

The multiplicities $n$ that are used vary from case to case, and in theory using more multiplicities allows for a closer fit. In practice, only the first few terms are used ($n = 1, 2, 3, 4, 6$) for symmetry considerations, decreased complexity, to avoid over-fitting, and to avoid introducing additional high-frequency motions into the simulation.66

Eq. 2-1 gives the 1-dimensional potential energy for a single dihedral, but the total dihedral potential energy function is a $D$-dimensional function, where $D$ is the total number of dihedrals in the molecule being studied. There are differing philosophies on how to combine the single dihedral energies into the total dihedral potential energy function. The simplest approach is to decouple the individual dihedrals and treat the total energy function as a sum of the individual potential energy functions:

$$V(A, \Delta, \theta) = \sum_{k} \sum_{n=1}^{n_{\text{max}}} A_{nk} [1 + \cos(n\theta_k - \Delta_{nk})]$$  \hspace{1cm} (2-2)

Here, we have written out the sum over each dihedral as the sum over $k$. $\theta_k$ represents the value of the $k$th dihedral angle, $A_{nk}$ is the force constant for the $k$th

dihedral angle and the $n$th multiplicity, and $\Delta_{nk}$ is the phase constant for the $k$th dihedral angle and the $n$th multiplicity. Alternatively, some models (e.g., CMAP)\textsuperscript{42,43} introduce lookup tables or cross terms into the force field to couple dihedrals. Our focus is on the simple case of independent Fourier terms for each dihedral (Equation 2-2).

The force field parameters are fit to the target data by optimizing a merit function. One of the most common merit functions used in the literature is the square difference between the potential energy surface (PES) of the original force field about the dihedral angle or angles of interest, and the PES of ab initio energies, taken at the same values of the dihedral angles.\textsuperscript{67,68} An additional term is often added to the difference in order to account for the shift in absolute energy levels between the force field and ab initio energies.\textsuperscript{21} The (equivalent) goal of these approaches is to minimize (in terms of $A$ and $\Delta$) a function of the form:

$$\chi^2 (A, \Delta, \{i\}) = \sum_i \left[ V(A, \Delta, \theta_i) - (E_{QM,i} - E_{MM,i}) \right]^2$$

(2-3)

This is done for some set of data points $\{i\}$ composed of $\theta_i$, the value for each dihedral angle in data point $i$, and $E_{QM,i}$ and $E_{MM,i}$, the ab initio energy and the original uncorrected force field energy, respectively, for the configuration $\theta_i$.

The phase constants $\Delta$ are often fixed to the symmetry positions of 0 or 180°. Some parameterization philosophies prescribe this constraint whenever transferability of parameters is a primary goal.\textsuperscript{48} It has been pointed out that the dihedral parameters are not necessarily transferable between isomers, and asymmetric $\Delta$ values are needed to properly represent the quantum mechanical energy surface for chiral molecules.\textsuperscript{21} Most of the monomers and ligands modeled in biochemical simulations are chiral, including amino acids, nucleosides, carbohydrates, and more than half of current drugs.\textsuperscript{69}
Therefore, a general dihedral parameterization scheme must allow for the elements of $\Delta$ to be specified as variable parameters, which leads to the dihedral function form (Equation 2-2) being non-linear in its parameters. Thus, the problem has traditionally been solved as a non-linear least squares fitting.

Authors have used both local and global optimization schemes to reach a solution for the minimum of Equation 2-3. Local schemes, like the Levenberg–Marquardt algorithm,\textsuperscript{70} and simplex\textsuperscript{71} or Newton/quasi-Newton minimization\textsuperscript{65,72} find a local minimum to $\chi^2$ relative to some preset starting point. When varying $\Delta$, Equation 2-3 is a complicated function in parameter space with multiple minima,\textsuperscript{21} so for this case local schemes cannot always find the optimal parameter set, even after fully converging. As local schemes cannot optimally solve the dihedral parameterization problem (with variable $\Delta$) in general, we will not consider them further. Global schemes such as genetic algorithms, systematic search,\textsuperscript{73} and Monte Carlo simulated annealing\textsuperscript{21,74–76} can find an approximate optimal parameter set after full convergence, even when varying $\Delta$, though, as will be shown for genetic algorithms, the required run time for convergence of global optimizers will typically not be negligible, and convergence still requires an arbitrary stopping condition that can lead to further uncertainty in the resulting parameters. In the case where the four atoms in the dihedral being parameterized do not include a chiral center, it is appropriate to restrain the $\Delta$ values to 0 or 180° due to symmetry. For this special case, it has been pointed out that the potential function (Equation 2-2) is linear in the parameters and thus is subject to general linear least-squares (LLS) fitting.\textsuperscript{21,41} This is a well-known problem and there exists an analytic solution for calculating the optimal parameter set $\mathbf{A}$ in this case. As
far as we are aware, an analytic LLS scheme has not previously been applied to the simultaneous parameterization of $A$ and $\Delta$, though. We will show that it is possible to treat the full dihedral parameterization problem of solving for $A$ and $\Delta$ in a single general LLS fitting scheme, solving analytically for the optimal parameter set.

**Linear least squares fitting of dihedral parameters**

The constant terms in each term of the sum in Equation 2-2 can be combined, which gives a single constant term (independent of $\theta$):

$$V(A, \Delta, \theta) = \sum_{k}^{n_{\text{max}}} \sum_{n} A_{nk} + \sum_{k}^{n_{\text{max}}} \sum_{n} A_{nk} \cos(n\theta_k - \Delta_{nk})$$

$$= C(A) + V'(A, \Delta, \theta)$$

$C$ can be dropped, as a constant in the energy expression does not affect the dynamics, and $V'$ can be used for fitting. The potential energy function, for fitting purposes, then becomes (dropping the prime):

$$V(A, \Delta, \theta) = \sum_{k}^{n_{\text{max}}} \sum_{n} A_{nk} \cos(n\theta_k - \Delta_{nk})$$

(2-4)

This change will not alter the set of parameters that best puts the force field PES in agreement with the ab initio PES. To avoid any numerical issues with the (usually large) inherent energy level difference between force field and ab initio energies, the sets $E_{QM,i}$ and $E_{MM,i}$ should be shifted so that their respective averages are zero before performing the fitting calculation.

Eq. 2-4 can be linearized by rewriting it in terms of a new set of parameters. Through a trivial trigonometric transformation, we first expand:
\[ V(A, \Delta, \theta) = \sum K \sum_n [A_{nk} \cos(\Delta_{nk}) \cos(n\theta_k) + A_{nk} \sin(\Delta_{nk}) \sin(n\theta_k)] \]

We then combine terms to create two new parameter sets:

\[ V(A', B', \theta) = \sum K \sum_n [A'_{nk} \cos(n\theta_k) + B'_{nk} \sin(n\theta_k)] \quad (2-5) \]

\[
\begin{align*}
    A'_{nk} & = A_{nk} \cos(\Delta_{nk}) \\
    B'_{nk} & = A_{nk} \sin(\Delta_{nk}) \\
    A_{nk} & = \sqrt{(A'_{nk})^2 + (B'_{nk})^2} \\
    \Delta_{nk} & = \tan^{-1}\left(\frac{B'_{nk}}{A'_{nk}}\right)
\end{align*}
\quad (2-6)
\]

(Note that the last two lines of Equation 2-6 are simply derived from the first two lines). Eq. 2-5 is linear in its parameters and subject to LLS fitting. By solving for the parameters \(A'\) and \(B'\), \(A\) and \(\Delta\) can then be recovered via Equation 2-6.

As previously stated, the linear least squares method is a well-known solution to fitting a linear function form to data in a least squares sense. A complete derivation of the method can be found elsewhere.\(^{77}\) We will simply present here the final solution for the so-called “normal equations” method as applied to the dihedral parameterization problem. As discussed below, this solution to the LLS problem is susceptible to large numerical errors in some cases, but it is intuitive and easily implemented. The fitting problem can be expressed as a linear system of equations:

\[ C \cdot a = k \quad (2-7) \]

Here, the vector \(a\) is the parameter set being solved for:

\[
a_n = \begin{cases} 
    A'_{nk} & \text{if } 1 \leq n \leq d \cdot n_{max} \\
    B'_{nk} & \text{if } n > d \cdot n_{max}
\end{cases}
\quad (2-8)\]
The matrix $\mathbf{C}$ is a geometric factor defined in terms of the configurations in the data set (the “predictor” matrix) and the vector $\mathbf{k}$ is defined in terms of the configurations and energies (the “response” vector):

$$
\begin{align*}
\mathbf{C}_{no} = \mathbf{C}_{on} = \\
\left\{ \\
\sum_{i} \cos(n'\theta_{ki})\cos(o'\theta_{ji}) & \quad \text{if } 1 \leq n, o \leq d \cdot n_{\text{max}} \\
\sum_{i} \cos(n'\theta_{ki})\sin(o'\theta_{ji}) & \quad \text{if } 1 \leq n \leq d \cdot n_{\text{max}} \text{ and } o > d \cdot n_{\text{max}} \\
\sum_{i} \sin(n'\theta_{ki})\sin(o'\theta_{ji}) & \quad \text{if } n, o > d \cdot n_{\text{max}}
\end{align*}
$$

$$
\mathbf{k}_{n} = \\
\left\{ \\
\sum_{i} \cos(n'\theta_{ki})(E_{QM,i} - E_{MM,i}) & \quad \text{if } 1 \leq n \leq d \cdot n_{\text{max}} \\
\sum_{i} \sin(n'\theta_{ki})(E_{QM,i} - E_{MM,i}) & \quad \text{if } n > d \cdot n_{\text{max}}
\end{align*}
$$

$\mathbf{C}$ is symmetric, and $\mathbf{C}$, $\mathbf{a}$, and $\mathbf{k}$ have dimension $2 \cdot d \cdot n_{\text{max}}$. In Equations 2-8 – 2-10, $d$ is the number of dihedrals being fit, $n_{\text{max}}$ is the number of multiplicities being fit, and $E_{QM,i}$, $E_{MM,i}$, and $\theta_{ki}$ are the \textit{ab initio} energy, original force field energy, and value for dihedral angle $k$, respectively, from the $i$th data point. The variables $n'$ and $k$ are defined in terms of the index $n$, with $o'$ and $j$ having analogous definitions in terms of the index $o$:

$$
\begin{align*}
\left\{ \\
n' = [(n - 1) \mod n_{\text{max}}] + 1 \\
k = \left(\left\lfloor \frac{(n - 1)}{n_{\text{max}}} \right\rfloor \mod d + 1 \right)
\end{align*}
$$

For a sufficient number of data points (more than the number of fitted parameters, $2 \cdot d \cdot n_{\text{max}}$), and non-pathological data, Equation 2-7 can be solved for $\mathbf{a}$ using standard matrix techniques. Again, $\mathbf{a}$ contains the terms from the alternate Fourier coefficients $\mathbf{A}'$ and $\mathbf{B}'$, which can be converted back to $\mathbf{A}$ and $\mathbf{A}$ via Equation 2-6.
It should be pointed out that this is a rather naïve solution to the general LLS problem, as it is susceptible to round off error, as well as unsolvable cases (i.e., when $C$ is singular, as can happen when using an insufficient number of data points). More sophisticated standard techniques (employing, e.g., QR decomposition or SVD) for solving the general LLS problem can be found elsewhere.\textsuperscript{77} We want to emphasize again that at least $2\cdot d\cdot n_{\text{max}}$ data points (that is, points on the dihedral “scan” used) are needed in this implementation, and typically more than this number are needed to properly sample the \textit{ab initio} energy surface. Using too small of a data set will return spurious results that do not represent the optimal parameter set.

A few modifications can be made to Equations 2-9 and 2-10 to allow for some oft-used features of the fitting process. Many parameterization schemes employ data set weighting, making certain data points count more towards the merit function than others (e.g., with Boltzmann weighting).\textsuperscript{41,46} If the weight for data point $i$ is $w_i$, then each occurrence of $\cos(n\theta_{ki})$ or $\sin(n\theta_{ki})$ in $k$ and each product of $\cos(n\theta_{ki})$ and/or $\sin(n\theta_{ki})$ in $C$ would be multiplied by $w_i$. Also, sometimes two or more dihedral angles are being fit that have exactly the same atom type sequence – for example, if pentane’s two C-C-C dihedrals were being simultaneously fit – and thus should be assigned the same dihedral parameters. This “equivalencing” of dihedrals can also be handled by replacing each occurrence of $\cos(n\theta_{ki})$ or $\sin(n\theta_{ki})$ in $C$ and $k$ by a sum of cosines or sines, where the sum is over dihedrals of the same type. It is even possible to incorporate harmonic restraints on the initial parameters in order to avoid obtaining unphysical parameters; these restraints add quadratic terms to $\chi^2$, and so lead to simple modifications of $C$ and $k$. We have a script on our website.
(http://www.clas.ufl.edu/users/roitberg/links.html, under “Software”) that implements these fitting schemes. As mentioned before, when the dihedral being fit contains no chiral centers the $\Delta$ values are typically restrained to 0 or 180°; thus, the script also makes this restraint possible. The phase constants can be removed from the fitting function, giving:

$$V(A, \theta) = \sum_{k} \sum_{n} A_{nk}\cos(n\theta_k)$$

If just the set $A$ is treated here, $\Delta_{nk}$ is implicitly fit to either 0 or 180°, with the result being expressed through the sign of $A_{nk}$ (0 if positive, 180° if negative). This case is specified in our script through an input flag.

**Methods**

All molecular mechanics (MM) calculations were performed using the Amber 12 suite\textsuperscript{78}. We used the nonbonding, bond, and angle parameters of GAFF\textsuperscript{47} in all calculations. Atomic charges were calculated using the AM1-BCC method in the antechamber program in Amber.\textsuperscript{51,52,79} It is possible to start from any reasonable “initial guess” or starting point (i.e., values used in generating the initial MM energy surface) for the dihedral parameters being fit. Neglecting round-off differences, any starting point will lead to the same parameter set; however, the starting point must be taken into account when calculating the final parameter set. In our tests, we used GAFF dihedral parameters to generate the initial MM surfaces. Using Equation 2-6, the GAFF dihedral force constants $A_{GAFF}$ and phase constants $\Delta_{GAFF}$ are converted to the sets $A'_{GAFF}$ and $B'_{GAFF}$. The parameters obtained from solving Equation 2-7 will be denoted $A'_{fit}$ and
The final presented dihedral parameter sets for the LLS method \( (A_{LLS} \text{ and } \Delta_{LLS}) \) are then obtained by converting, via Equation 2-6, \( A'_{LLS} \) and \( B'_{LLS} \), which are given by:

\[
A'_{LLS} = A'_{GAFF} + A'_{fit}
\]
\[
B'_{LLS} = B'_{GAFF} + B'_{fit}
\]

Optimized geometries and quantum mechanics (QM) energies were calculated using Gaussian 09. The QM optimized geometries were then used to calculate the MM PES for each molecule, by performing a single point MM energy calculation for each structure. As has been pointed out, a further constrained MM optimization usually should be performed to produce the MM PES. However, our main goal here is to investigate the LLS fitting method, not to produce extremely reliable parameters for the test cases.

**Validation**

In order to validate our implementation of the LLS fitting method and compare the method to a traditional optimization technique, we first calculate dihedral parameters for two simple, achiral molecules: butane (Figure 2-1A) and 1-butanol (Figure 2-1B). This is done with the presented LLS method and with a genetic algorithm (GA) applied in the “traditional” way. By this, we mean that the GA merit function is Equation 2-3 with \( V(A, \Delta, \theta_i) \) given by the original non-linear form (Equation 2-2) so that \( A \) and \( \Delta \) are directly being varied. These two cases are compared in terms of the resulting parameter sets and run times for each method.

Geometry optimizations and single point calculations were performed at the MP2/6-31G* level on butane and 1-butanol. The 1-dimensional QM PES of butane was created from a 72-point relaxed scan of the C-C-C-C dihedral angle, with the points
spaced by 5°. The QM PES of 1-butanol was calculated from a 2-dimensional 72-by-72 point grid (5184 data points). We note that all other dihedrals besides the ones being fit were allowed to move freely during the optimizations, including the C-C-O-H dihedral of 1-butanol. This dihedral in particular led to small “jumps” during the scan, as it moved over a large range between two consecutive steps of the scan, which can be seen as discontinuities in the resulting energy maps. A more appropriate data collection scheme would be to take the free dihedrals into better account, either restraining them or using them as a target in the fit. However, these discontinuities have a minimal effect on the fit for this simple case, and again our focus is on the fitting procedure itself, and not on deriving high quality parameters for 1-butanol.

The pyevolve\textsuperscript{81} Python module was used for the GA fit. The normal equations solution was implemented in Perl, and all matrix calculations were done with the Math::MatrixReal module. All fitting calculations were done on a quad core laptop PC with Intel i7 processors and 8 GB of memory. One parameter set for each molecule, fitting both $\mathbf{A}$ and $\mathbf{\Delta}$, was calculated with the GA, denoted as GAFF\textsubscript{GA}. For LLS, two sets of parameters for each molecule were calculated. In one set (GAFF\textsubscript{LLS,var}), both $\mathbf{A}$ and $\mathbf{\Delta}$ were fit as in GAFF\textsubscript{GA}, and in the other (GAFF\textsubscript{LLS,fix}), $\mathbf{\Delta}$ was fixed to be either 0 or 180° and only $\mathbf{A}$ was fit. Butane and 1-butanol are both achiral, so the latter case would be preferred for calculating physically accurate parameters, if one were actually modeling butane or 1-butanol. Four terms in the Fourier series for each dihedral ($n_{\text{max}} = 4$) were fit.
Chiral Molecule Parameterization

In order to investigate a realistic case where variable $\Delta$ would be needed, we also derived dihedral parameters for a linker dihedral in the drug amoxicillin (DrugBank ID DB01060) (Figure 2-1C). The dihedral $\chi$ contains the chiral center $C_3$, and as previously stated, dihedral parameters are not expected to be transferable between isomers, especially for dihedrals near a chiral center. Thus, a separate parameterization routine was run for $\chi$ in the isomer of amoxicillin about $C_3$ (Figure 2-1D). For each case, QM calculations were done at the MP2/6-31G*//HF/6-31G* level of theory. The 1-dimensional QM PES for each molecule was created from an 18-point relaxed scan of $\chi$, with the points spaced by 20°. For the two isomers, LLS was used to parameterize $\chi$, again with variable phases (GAFF$_{\text{LLS, var}}$) and fixed phases (GAFF$_{\text{LLS, fix}}$). The presented parameters were applied to the two C-C-C-C (c-c3-ca-ca in GAFF) dihedrals about the $C_2$-$C_3$ bond in order to generate the new energy profiles, though there are really six dihedrals (and three unique dihedral types) about the bond that contribute to the profile, and in a real parameterization session, the parameters should be split up over the other two types as well. Again, four terms in the Fourier series for each dihedral ($n_{\text{max}} = 4$) were fit.

Results

Validation – butane

For butane, the GAFF$_{\text{LLS}}$ sets and GAFF$_{\text{GA}}$ yielded nearly identical results (Table 2-1). The difference between GAFF$_{\text{LLS, var}}$ and GAFF$_{\text{LLS, fix}}$ was negligible; there were no major degrees of freedom unaccounted for during the dihedral scan, so the inherent symmetry was maintained even when the phases were allowed to vary. The RMSE between the MM PES and the QM PES was improved identically in all of the fitting
cases. Figure 2-2 shows a comparison of the resulting potential energy scans for both methods to the *ab initio* scan.

The GA started finding reasonable solutions after 50 generations, which took about a second to run. However, it was necessary to run longer for the force constants in the two sets of parameters to agree within the order of $10^{-3}$ kcal/mol; it was sufficient to run the GA for 1000 generations with a population of 200, which took 25.3 s to complete. The LLS calculation took 0.1 s to complete.

**Validation – 1-butanol**

For the 1-butanol test, a large slowdown in the GA method was apparent. The GA took about 400 generations in this case to start finding optimal solutions due to the doubling of the parameter space dimensionality, and each generation took longer to run with the increase in data points. To run the same number of generations as the previous example (1000 generations with population 200), took about 45 minutes, while the LLS calculation still took a negligible amount of time to complete (0.8 s). It should be pointed out that this particular scan (72 x 72 grid) is a much higher resolution than is needed to obtain numerically accurate parameters in this case, which contributed to the long run time of the GA. We uniformly reduced the grid down to 18 x 18 and obtained similar parameters, usually within $10^{-2}$ kcal/mol for the force constants. Still, data sets of thousands of points would start to be necessary if three or more dihedrals were being simultaneously treated.

Figure 2-3 shows the 2-D potential energy scans for the *ab initio* energies, the original GAFF energies, the GAFF$_{GA}$ energies, and the GAFF$_{LLS,var}$ energies. Tables 2-2 and 2-3 show that, again, the derived parameter sets were very similar among all the fitting methods, and the RMSE for the MM PES with fitted parameters was reduced by a
similar amount for each case, with GAFF_{LLS, var} giving the closest fit by a slight margin. However, in this case, allowing $\Delta$ to take on arbitrary values in GAFF_{LLS, var} introduced a slight asymmetry which is not present in the achiral structure of 1-butanol. In general, these kinds of artifacts can appear due to biases in the configurations chosen in the data set and from not taking into account other major degrees of freedom during the relaxed scan. In this case, the data set is sufficiently high resolution and uniformly distributed over dihedral space that we can conclude that data set bias is not the cause of the artifacts. The most likely reason for these asymmetric $\Delta$ values is the aforementioned C-C-O-H dihedral which was not taken into account during the scan. The discontinuities it introduced into the energy surfaces from “jumping” during the scan obviously introduced a bias towards one region of the energy surface which led to an asymmetric potential giving a numerically better fit. We predict that if this dihedral were also treated in a new 3-dimensional fit, these artifacts would vanish. This was a minimal effect; compared to GAFF_{LLS, fix}, the calculated values of $\Delta$ are nearly identical, and the $\Delta$ values in GAFF_{LLS, var} do not differ very much from 0 or 180°. The biggest offender was the $\delta_2$ term for the C-C-C-C dihedral; however, the corresponding force constant is effectively zero, and so any phase constants calculated along with it should be considered spurious anyway. Every other phase constant was within 15° of 0 or 180°. Still, we repeat our earlier warning: these parameters (including the more reasonable GAFF_{LLS, fix} set) are not meant to be taken as physically realistic parameters for modeling butane or 1-butanol. Besides the presence of artifacts, isolated dihedral scans, without taking into further account environment or solvation effects, are of limited usefulness anyway, as the optimal dihedral potential is very sensitive to the surrounding
environment. The most significant result for our purposes here is the fact that $\text{GAFF}_{\text{LLS, var}}$ and $\text{GAFF}_{\text{GA}}$ are effectively the same (if inaccurate) parameter set (neglecting the spurious $\delta_2$ values), with $\text{GAFF}_{\text{LLS, var}}$ being a slightly better numerical fit calculated in a negligible run time.

**Chiral Molecule Parameterization**

Table 2-4 shows the resulting parameters and residual RMSEs when fitting the C-C-C-C parameters in the $\chi$ dihedral in amoxicillin with the LLS method, and Figure 2-4A shows the resulting energy profiles of $\chi$ for the various parameter sets and *ab initio* energies. In this case, there is a significant difference between $\text{GAFF}_{\text{LLS, var}}$ and $\text{GAFF}_{\text{LLS, fix}}$. Clearly, $\text{GAFF}_{\text{LLS, fix}}$ was not able to improve significantly on $\text{GAFF}$, which does not have any parameters for this dihedral. $\text{GAFF}$ overestimated the rotation barriers for this dihedral by 2-3 kcal/mol, and by fixing $\Delta$ to 0 or 180° no improvement was made as the profile minima lie close to 120° and 300°. The values of $\Delta$ in $\text{GAFF}_{\text{LLS, var}}$, with the exception of the smallest $n=3$ term, vary significantly from 0 or 180°, which brings the MM energy profile into much closer agreement with the QM profile, reducing the RMSE by about a factor of five.

Table 2-5 and Figure 2-4B show the results of parameterizing the same dihedral in the amoxicillin isomer. Exactly the same disparity between $\text{GAFF}_{\text{LLS, var}}$ and $\text{GAFF}_{\text{LLS, fix}}$ as for amoxicillin is seen here. As predicted, the parameters derived for amoxicillin are not directly transferable to its isomer. The $\Delta$ values are approximately identical between the two cases, but the optimal values of $\Delta$, apart from the small $n=3$ term, are shifted by about 120±20° between the isomers.
We caution once more that the exact parameters derived here are not necessarily physically relevant parameters for modeling amoxicillin or its isomer. Looking at the profiles, it is clear that the fitted parameters are picking up small features of the QM surface which are not significant and should not necessarily be represented in the corresponding single dihedral potential in the force field; different choices of the periodicities or placing restraints on some parameters could help alleviate this issue. Also, the values of the other dihedrals along the linking chain between the ring systems were not accounted for. Movements in these dihedrals during the relaxed scan contributed to the features in the QM surface to which the dihedral parameters were overfit. An interesting avenue for future work would be to investigate simultaneously parameterizing the four dihedrals along this linking chain and study how taking the full multidimensional QM energy surface into account affects the individual dihedral parameters. An appropriate parameterization of these dihedrals would additionally take into account the environment in which the drug is being modeled.

Discussion

Using LLS to solve the dihedral problem is clearly superior to any optimization technique, in general. However, the real improvement in run time and parameter quality over the traditional techniques will be seen when simultaneously treating multiple dihedrals while varying the phases. As shown, variable phases are clearly needed when modeling dihedrals near a chiral center. As the majority of molecules of biochemical interest are chiral, variable phases are an important consideration for the dihedral parameterization process. The remaining question here is then the extent of the utility of multidimensional fits for dihedral parameterization. The literature mostly employs 2-dimensional fits (e.g., φ/ψ fits in amino acids), and these are mostly used in
conjunction with coupling methods like CMAP. Parameters will not be as transferable when treating more dihedrals simultaneously, but a much better representation of the true QM PES for a particular model could outweigh other considerations. The main barrier to simultaneously treating many dihedrals is the need for more \textit{ab initio} calculations in order to properly sample the QM PES. If the usual “dihedral scan” notion is directly translated to higher dimensions, than the number of \textit{ab initio} calculations grows \textit{exponentially} with the number of dihedrals being fit. However, the number of data points that the LLS method needs to be overdetermined is \textit{linear} in the number of dihedrals being fit, though it is still not clear how many additional points will be needed to appropriately sample the QM PES. Thus, in Chapter 3, we investigate the use of methods of data collection for dihedral parameterization other than a complete multidimensional grid, and study how many \textit{ab initio} calculations are really needed to find the best representation of the QM PES in terms of the dihedral potential.
Figure 2-1. Test systems for the LLS method: A) butane, B) 1-butanol, C) amoxicillin, and D) isomer of amoxicillin. A 1-dimensional scan was performed on butane and used to fit the C-C-C-C dihedral parameters, and for 1-butanol a 2-dimensional scan was used to simultaneously fit the C-C-C-C and C-C-C-O dihedral parameters. For the two isomers, a 1-dimensional scan of the dihedral marked χ in C was performed to fit the corresponding C-C-C-C parameters.
Figure 2-2. Potential energy scans and error in scans for butane. A) C-C-C-C dihedral potential energy scans in butane. Comparison of \textit{ab initio} (MP2/6-31G*) energies and force field energies with GAFF, GAFF$_{GA}$, and GAFF$_{LLS,var}$ parameter sets. The averages for each energy curve were set to zero before fitting was done. B) Residual errors between the scans with the three parameter sets and the \textit{ab initio} scan.
Figure 2-3. 2-D potential energy scans of C-C-C-C and C-C-C-O dihedrals in 1-butanol. Energies are in kcal/mol. The scan was performed over a 72 x 72 grid, with a spacing of 5°. Energies were evaluated for MP2/6-31G* optimized geometries, and the individual plots are labeled with the type of single point energy evaluation (MP2/6-31G*, GAFF, GAFFGA, or GAFFLLS, var parameters).
Figure 2-4. Dihedral potential energy scans for amoxicillin and its isomer. A) Amoxicillin scan and B) isomer scan about the dihedral χ. Energies were calculated for HF/6-31G* optimized structures, with MP2/6-31G*, GAFF, GAFF\textsubscript{LLS, var}, and GAFF\textsubscript{LLS, fix} energies.
Table 2-1. Butane C-C-C-C (c3-c3-c3-c3 in GAFF) dihedral parameters in GAFF, GAFFGA, and GAFFLLS parameter sets, and RMSE between force field energies and \textit{ab initio} energies.

<table>
<thead>
<tr>
<th>Force constants (kcal/mol)</th>
<th>GAFF</th>
<th>GAFFGA</th>
<th>GAFFLLS, var</th>
<th>GAFFLLS, fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>A₁</td>
<td>0.200</td>
<td>0.071</td>
<td>0.071</td>
<td>0.074</td>
</tr>
<tr>
<td>A₂</td>
<td>0.250</td>
<td>0.048</td>
<td>0.048</td>
<td>0.045</td>
</tr>
<tr>
<td>A₃</td>
<td>0.180</td>
<td>0.347</td>
<td>0.347</td>
<td>0.350</td>
</tr>
<tr>
<td>A₄</td>
<td>0.000</td>
<td>0.171</td>
<td>0.170</td>
<td>0.173</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phase constants (degrees)</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>δ₁</td>
<td>180.000</td>
<td>0.548</td>
<td>0.511</td>
<td>0.000</td>
</tr>
<tr>
<td>δ₂</td>
<td>180.000</td>
<td>-179.547</td>
<td>-179.711</td>
<td>180.000</td>
</tr>
<tr>
<td>δ₃</td>
<td>0.000</td>
<td>-0.221</td>
<td>0.010</td>
<td>0.000</td>
</tr>
<tr>
<td>δ₄</td>
<td>0.000</td>
<td>0.084</td>
<td>0.011</td>
<td>0.000</td>
</tr>
</tbody>
</table>

| RMSE (kcal/mol)           | 0.302    | 0.073  | 0.073        | 0.073        |

Table 2-2. 1-Butanol C-C-C-C (c3-c3-c3-c3 in GAFF) dihedral parameters in GAFF, GAFFGA, and GAFFLLS parameter sets, and RMSE between \textit{ab initio} energies and force field energies calculated with the corresponding dihedral parameter set.

<table>
<thead>
<tr>
<th>Force constants (kcal/mol)</th>
<th>GAFF</th>
<th>GAFFGA</th>
<th>GAFFLLS, var</th>
<th>GAFFLLS, fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>A₁</td>
<td>0.200</td>
<td>0.201</td>
<td>0.214</td>
<td>0.214</td>
</tr>
<tr>
<td>A₂</td>
<td>0.250</td>
<td>0.071</td>
<td>0.002</td>
<td>0.001</td>
</tr>
<tr>
<td>A₃</td>
<td>0.180</td>
<td>0.103</td>
<td>0.112</td>
<td>0.109</td>
</tr>
<tr>
<td>A₄</td>
<td>0.000</td>
<td>0.142</td>
<td>0.132</td>
<td>0.130</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phase constants (degrees)</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>δ₁</td>
<td>180.000</td>
<td>-176.857</td>
<td>179.209</td>
<td>180.000</td>
</tr>
<tr>
<td>δ₂</td>
<td>180.000</td>
<td>116.177</td>
<td>-75.661</td>
<td>0.000</td>
</tr>
<tr>
<td>δ₃</td>
<td>0.000</td>
<td>-11.176</td>
<td>12.913</td>
<td>0.000</td>
</tr>
<tr>
<td>δ₄</td>
<td>0.000</td>
<td>-7.356</td>
<td>-11.629</td>
<td>0.000</td>
</tr>
</tbody>
</table>

| RMSE (kcal/mol)           | 0.981    | 0.491  | 0.450        | 0.500        |
Table 2-3. 1-Butanol C-C-C-O (c3-c3-c3-oh in GAFF) dihedral parameters in GAFF, GAFF_{GA}, and GAFF_{LLS} parameter sets, and RMSE between \textit{ab initio} energies and force field energies calculated with the corresponding dihedral parameter set.

<table>
<thead>
<tr>
<th></th>
<th>GAFF</th>
<th>GAFF_{GA}</th>
<th>GAFF_{LLS, var}</th>
<th>GAFF_{LLS, fix}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Force constants (kcal/mol)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$A_1$</td>
<td>0.000</td>
<td>0.856</td>
<td>0.843</td>
<td>0.825</td>
</tr>
<tr>
<td>$A_2$</td>
<td>0.000</td>
<td>0.224</td>
<td>0.272</td>
<td>0.272</td>
</tr>
<tr>
<td>$A_3$</td>
<td>0.156</td>
<td>0.880</td>
<td>0.908</td>
<td>0.880</td>
</tr>
<tr>
<td>$A_4$</td>
<td>0.000</td>
<td>0.234</td>
<td>0.261</td>
<td>0.251</td>
</tr>
<tr>
<td>Phase constants (degrees)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\delta_1$</td>
<td>0.000</td>
<td>169.312</td>
<td>168.122</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_2$</td>
<td>0.000</td>
<td>177.000</td>
<td>178.418</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_3$</td>
<td>0.000</td>
<td>13.267</td>
<td>14.270</td>
<td>0.000</td>
</tr>
<tr>
<td>$\delta_4$</td>
<td>0.000</td>
<td>24.185</td>
<td>15.702</td>
<td>0.000</td>
</tr>
<tr>
<td>RMSE (kcal/mol)</td>
<td>0.981</td>
<td>0.491</td>
<td>0.450</td>
<td>0.500</td>
</tr>
</tbody>
</table>

Table 2-4. C-c3-ca-ca dihedral parameters fitted via $\chi$ in amoxicillin for GAFF, GAFF_{LLS, var}, and GAFF_{LLS, fix} parameter sets, and RMSE between \textit{ab initio} energies and force field energies calculated with the corresponding dihedral parameter set.

<table>
<thead>
<tr>
<th></th>
<th>GAFF</th>
<th>GAFF_{LLS, var}</th>
<th>GAFF_{LLS, fix}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Force constants (kcal/mol)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$A_1$</td>
<td>0.000</td>
<td>0.128</td>
<td>0.019</td>
</tr>
<tr>
<td>$A_2$</td>
<td>0.000</td>
<td>0.691</td>
<td>0.120</td>
</tr>
<tr>
<td>$A_3$</td>
<td>0.000</td>
<td>0.085</td>
<td>0.100</td>
</tr>
<tr>
<td>$A_4$</td>
<td>0.000</td>
<td>0.173</td>
<td>0.110</td>
</tr>
<tr>
<td>Phase constants (degrees)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\delta_1$</td>
<td>0.000</td>
<td>-103.747</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_2$</td>
<td>0.000</td>
<td>-101.034</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_3$</td>
<td>0.000</td>
<td>-3.038</td>
<td>0.000</td>
</tr>
<tr>
<td>$\delta_4$</td>
<td>0.000</td>
<td>-55.354</td>
<td>0.000</td>
</tr>
<tr>
<td>RMSE (kcal/mol)</td>
<td>1.008</td>
<td>0.220</td>
<td>0.986</td>
</tr>
</tbody>
</table>
Table 2.5. C-c3-ca-ca dihedral parameters fitted via $\chi$ in an amoxicillin isomer for GAFF, GAFF$_{LLS,var}$, and GAFF$_{LLS,fix}$ parameter sets, and RMSE between $ab$ initio energies and force field energies calculated with the corresponding dihedral parameter set.

<table>
<thead>
<tr>
<th></th>
<th>GAFF</th>
<th>GAFF$_{LLS,var}$</th>
<th>GAFF$_{LLS,fix}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Force constants (kcal/mol)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$A_1$</td>
<td>0.000</td>
<td>0.090</td>
<td>0.073</td>
</tr>
<tr>
<td>$A_2$</td>
<td>0.000</td>
<td>0.667</td>
<td>0.318</td>
</tr>
<tr>
<td>$A_3$</td>
<td>0.000</td>
<td>0.083</td>
<td>0.092</td>
</tr>
<tr>
<td>$A_4$</td>
<td>0.000</td>
<td>0.150</td>
<td>0.062</td>
</tr>
<tr>
<td><strong>Phase constants (degrees)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\delta_1$</td>
<td>0.000</td>
<td>157.136</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_2$</td>
<td>0.000</td>
<td>119.390</td>
<td>180.000</td>
</tr>
<tr>
<td>$\delta_3$</td>
<td>0.000</td>
<td>-3.712</td>
<td>0.000</td>
</tr>
<tr>
<td>$\delta_4$</td>
<td>0.000</td>
<td>69.699</td>
<td>0.000</td>
</tr>
<tr>
<td><strong>RMSE (kcal/mol)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.973</td>
<td>0.177</td>
<td>0.848</td>
</tr>
</tbody>
</table>
CHAPTER 3
FORCE FIELD PARAMETERIZATION: SAMPLING

Background

In Chapter 2, we discussed the typical method for calculating the dihedral parameters of choosing the set of parameters which brings the system of interest’s MM configuration potential energy surface (PES), with respect to the dihedral angles of interest, into best agreement with the corresponding QM PES. The measure of "agreement" between MM and QM PES is typically a least squares merit function, given by eq 2-3. The optimal set of dihedral parameters is that which minimizes Equation 2-3. We discussed in Chapter 2\textsuperscript{82} that the single optimal set of force constants $A$ and phase constants $\Delta$ that minimizes Equation 2-3 can be calculated analytically via the linear least squares (LLS) method. With the LLS method of fitting, over 100 periodicities for a single dihedral could quickly be fit as long as a sufficient number of data points is used; however, we will stick to using $n_{\text{max}} = 5$ here in keeping with the traditional approach (see Background in Chapter 2).

The predominant approach used in the literature to select the configurations to be used in Equation 2-3 is a single torsion scan. By this, we mean that parameters for one dihedral angle are calculated at a time, and the set of configurations used in the minimization of Equation 2-3 are derived from an equally spaced relaxed potential energy scan on that dihedral angle coordinate. Occasionally, this will be extended to a 2-dimensional case,\textsuperscript{37,43} where two dihedrals are simultaneously parameterized and the configurations used are a 2D grid in dihedral space, generated via a 2D relaxed scan. Simultaneous parameterizations of more than two dihedrals are rare, though in theory they should give a better representation of the full QM PES.
Extending the scanning technique to a general $d$-dimensional case (i.e., simultaneously parameterizing $d$ dihedrals), we see that the number of configurations needed in Equation 2-3 will be $p^d$, where $p$ is the number of scan points used along each dihedral coordinate. Typical numbers of scan points used in 1D scans in the literature are in the range 18-36 (spaced every 10-20° along the dihedral). So, the number of configurations used, which ultimately means the number of ab initio geometry optimizations needed, quickly blows up after just a few dihedrals are considered. With just three dihedrals, and $p = 18$, the number of QM calculations needed approaches 6000, which is inconvenient for a single parameterization session for small fragments of molecules, and infeasible for larger model compounds.

Our focus here is an alternative method of selecting the configurations used during the dihedral parameterization process, with the aim of minimizing the number of QM calculations needed while still deriving a parameter set that approaches the quality of a “full scan” derived set. We will show that by sampling configurations over the full multidimensional dihedral space, even using a very simple selection method, the number of ab initio calculations required is approximately linear in $d$, as opposed to the exponential behavior of the traditional full scan method. Our goal here is to show that doing multidimensional fits of dihedral parameters with three or more dihedrals is feasible, and not as limited by the ab initio cost as it would initially seem.

**Methods**

Before choosing a method of sampling, it is important to consider what our metric for a successful parameterization is. We will use the root mean square error (RMSE) between the fitted MM PES and the ab initio QM PES as a metric in this paper - the lower the final RMSE, the better. For our considerations, we will calculate the RMSE
over the entire PES, with equal weighting; however, this is not the only option. One of the published philosophies on dihedral fitting has been that, if you must choose an area of the PES to reproduce more exactly than others, the lower energy areas around the minima should take precedence. If this is the goal of the parameterization, then the metric should be appropriately modified - perhaps weighted by a Boltzmann-like term, or alternatively taken over only the low energy areas of the PES.

The first test in this study is to apply the results of doing this kind of energy-weighted sampling to the entire PES. To this end, we ran 2 ns of MD on 1-butanol in vacuum and selected configurations out of the resulting trajectory data, which are implicitly energy-weighted by their relative frequency. In order to ensure a spread in the resulting configurations, and with an eye towards automating this whole procedure, we first ran an RMSD-based clustering algorithm on the MD trajectories - 9 distinct clusters were obtained (Figure 3-1). We then randomly chose \( n \) configurations out of each cluster, where we used \( n = 10 \) and 20, corresponding to 90 and 180 total configurations, respectively. Figure 3-2 shows the configurations selected for the first of the \( n = 10 \) runs. We ran each case twice, obtaining two random sets of configurations for each \( n \).

The results from this MD cluster random sampling approach showed a significant dependence on the exact nature of the random set of configurations chosen, so in order to reduce this dependence we used a gridded approach to increase the spread between the chosen configurations. We applied a polar grid around each minima, and forced a configuration to be chosen out of each bin, in addition to the structure closest to the centroid of the whole cluster. Specifically, we chose the configuration in the bin that was closest in RMSD to the centroid of the members of the bin (Figure 3-3), thus
eliminating all randomness from the selection procedure. We varied the number of radial ($r$) and angular ($\theta$) bins, with $r$ ranging from 1 to 3, and $\theta$ ranging from 3 to 8 - thus the total number of configurations used varied from 36 ($r = 1, \theta = 3$) to 225 ($r = 3, \theta = 8$).

In order to improve the RMSE over the full PES, we next moved on to sampling configurations out of the full dihedral space, without bias. We ran the following sampling and parameterization procedure for 7 separate fits over 5 small molecules (Figure 3-4), which corresponded to three 1D fits, three 2D fits, and one 3D fit. For the examples we show, configurations were selected off of a grid obtained from a full QM scan, which we performed in order to have a benchmark for comparison. However, in general, performing a full scan, even in terms of the MM energies, is not required. Configurations that are selected during the sampling can be generated on the fly by doing a partial relaxed MM scan - which represents a gentle twisting of the molecule about the dihedrals - from some initial structure to the desired structure which represents the sampled point in dihedral space. This is the general procedure that we use when running the parameterization process from scratch, without a reference full scan. In order to generate the curves of RMSE vs number of configurations (Figures 3-5 – 3-8), we generated random sets of configurations, and iteratively built them up, adding one (for the 1D scans) or two (for the 2D and 3D scans) new random configurations at a time. We then ran the fitting procedure on the current set of configurations at each step. This was done 20 times for each system being parameterized, representing 20 unique sampling and parameterization runs.

All MM calculations, including the MD run and all single point MM energy evaluations, were done with the AMBER 12 program. The clustering of the 1-butanol
trajectory was performed with the MMTSB Toolkit kclust routine. Atom typing and charge calculation (AM1-BCC) were performed with the antechamber program, and initial force field parameters for all systems in the study were from GAFF. All QM calculations (full relaxed scans and single point energy evaluations) were performed with the Gaussian 09 program. The fitting of 1-butanol from the MD trajectory sampling was done with QM data at the HF/6-31G* level of theory. All other fits performed by sampling randomly over all of dihedral space used MP2/6-31G*/HF/6-31G*/HF/6-31G data. The actual fitting of dihedral parameters via minimization of Equation 3-2 was performed with our LLS fitting script.

Results and Discussion

Table 3-1 shows the resulting RMSEs between the fitted MM PES and the HF/6-31G* QM PES for the first case studied - random selection of \( n \) configurations from each of the 9 clusters derived from the 1-butanol MD trajectory data, resulting in \( 9n \) configurations. As the MD cluster sampling methods (random and polar-gridded) are biased to lower energy configurations, we examined the resulting RMSE between the fitted MM PES and QM PES for points around the energy minima, as well as the RMSE for the full PES. “Around the minima” is defined here as within a 20° distance (in dihedral space) of each energy minimum. The reference configurations for calculating the RMSEs in both cases (near minima and full PES) were taken from a full 72x72 2D scan.

Even with just 2 runs for each case, we can already see that there is a significant dependence on the exact random set of configurations chosen. The full PES RMSE for the \( n = 10 \) case varies by about 0.45 kcal/mol between the two runs, and the \( n = 20 \) case RMSE varies by 0.19 kcal/mol. In addition, none of the runs provides a significant
improvement on the full PES over the default (GAFF) parameter set. The best run, \( n = 10 \) #1, gives an RMSE of 0.795 kcal/mol compared to 0.839 for GAFF, and the other three actually make the fitted MM PES agree less with the QM PES than the GAFF parameters. The situation is only slightly better when considering configurations around the minima. The four runs either are about the same as the GAFF RMSE or only slightly better, and do not come close to the full scan RMSE near the minima. On top of it all, there does not seem to be a significant improvement in the RMSEs between the \( n = 10 \) and \( n = 20 \) cases, meaning that adding 90 extra configurations to the fits did not improve anything. We immediately conclude that selecting configurations randomly out of an MD trajectory, even slightly spread over clusters, is an insufficient method of sampling configurations for the dihedral parameterization process.

When looking at Figure 3-2, it is evident that some of the configurations within a given cluster are perhaps too close. We might guess that this lack of spread (small RMSD) between the configurations within each cluster is what led to the abysmal performance during parameterization. If two configurations are too close in dihedral space, they add the same "information" to the fit as a single configuration at that point. So, our next step is to enforce spread amongst the configurations chosen within a given cluster by applying a polar gridding scheme (see Methods). Table 3-2 shows the resulting RMSEs for various numbers of bins; \( r \) is the number of radial bins and \( \theta \) is the number of angular bins, so the total number of configurations for a given run is \( 9 \times (r \times \theta + 1) \).

We can immediately see an improvement over the MD cluster random sampling method, starting with grids of 2 radial bins and 3 angular bins or more (corresponding to
63 or more configurations). Also, we can see the solutions mostly converge after $\theta = 5$ for $r = 2$ and 3. After these points, the fit is giving parameter sets better than GAFF both near the minima and for the full PES. Adding more configurations past these points also generally improves the parameters, though they do not quite reach the RMSE of the full scan. When we consider that the $r = 3, \theta = 5$ run represents using only 144 configurations, compared to 5184 for the full scan, we see that we have already accounted for most of the correct dihedral potential with an order of magnitude fewer configurations. If one were just interested in reproducing the low energy portion of the PES, we can see RMSEs near the minima that are improvements over GAFF with just 54 configurations used.

With the MD cluster polar gridded sampling method, we have a sufficient method of selecting configurations for dihedral parameterization, especially when considering just the low energy portion of the PES. However, we might consider if we can do even better at improving the full PES RMSE by lifting the constraint of selecting configurations randomly out of an MD trajectory, and randomly selecting configurations out of the entire dihedral space. This eliminates the energy bias, and leads to a purely unbiased sampling method. The physical justification here would be that we are introducing a better sampling of the barriers to rotation of the selected dihedral angles, and can expect to get a better picture of the shape of the barriers themselves. We use what we have learned from the MD cluster methods - namely, that it is a good idea to enforce some kind of spread in the configurations chosen - and decide to randomly select configurations off of a $d$-dimensional grid in dihedral space.
We will first look at 1-butanol, and compare back to the MD cluster methods. We used a 72x72 grid generated from a full 2D scan of 1-butanol. We performed 20 runs of iteratively building up sets of configurations and performing the parameterization process with each new set (see Methods). Figure 3-5 shows the results (full PES RMSE) for sets with up to 100 configurations.

The gridded random sampling method shows a further improvement over the MD cluster methods with respect to the full PES. After just 30 configurations, this method shows consistent improvements over GAFF, and the RMSE consistently lowers as more configurations are added until about 70 configurations, at which point the converged solution's RMSE is within about 0.1 kcal/mol of the full scan solution. The fact that we are improving on the GAFF parameters after just 30 \textit{ab initio} calculations should be stressed. With just 30 random configurations, we are already getting a good sense of what the full 2-dimensional PES looks like and properly reproducing it through the dihedral parameters. After about 60 or 70 configurations we have basically zeroed in on precisely the same solution as the full 5184 configuration scan, independent of the exact 60 or 70 configurations we have chosen off of the grid.

Before moving on, we want to see how the gridded random sampling method performs with respect to the "near minima" RMSE. We will simply note that at 50 configurations the average "near minima" RMSE is 0.610 ± 0.060 kcal/mol, and also consistently improves as more configurations are added. This still represents an improvement over the MD cluster polar gridded sampling method at a comparable number of configurations. It would seem that the gridded random sampling method applied over all of dihedral space is an overall better method of selecting configurations.
than either of the MD cluster methods, even when considering the part of the PES that those methods are biased towards. We conclude that the larger spread in the configurations sampled due to sampling over all of dihedral space improves the fit for any one part of the PES, as well as for the PES as a whole.

We will now present the parameterization results for a few other small molecules to see how well the gridded random sample method performs with different systems and different sized fits. Figure 3-6 shows the RMSE vs number of configurations results for the 1D scans of the H-S-S-H and O=CH-NH-CH₃ fragments and the C-C-C-C dihedral of butane. The gridded random sample method shows a consistent behavior between the three systems, with all three showing mostly converged solutions by 20 configurations. Butane shows a slightly larger spread in the solutions due to the fact that the configuration sets were taken off of a 72 point grid, while for the other two systems, a 36 point grid was employed.

Figure 3-7 shows the results for two more 2-dimensional fits, performed on different sections of the same molecule (CH₃-CH₂-S-S-CH₂-OH); one fit was simultaneously performed on the C-C-S-S and C-S-S-C dihedrals, while the other was simultaneously performed on the C-S-S-C and S-S-C-O dihedrals. Configurations were selected off of an 18x18 point grid.

The C-S-S-C/S-S-C-O parameterization had a similar behavior to the 1-butanol case, with convergence mostly by 60-70 configurations. However the C-C-S-S/C-S-S-C parameterization was not quite as well behaved. While the fit was improving on GAFF by 60 configurations, the solutions were just starting to converge towards the full scan result at 100 configurations.
Finally, we performed a full 3-dimensional scan on the previous system, simultaneously optimizing all three dihedrals (C-C-S-S/C-S-S-C/S-S-C-O) and selecting configurations off of an 18x18x18 point grid in dihedral space. Figure 3-8 shows the resulting RMSEs for up to 200 configurations. The "GAFF point" occurs at about 60 configurations and the solutions settle down mostly after 100-120 configurations, which is a significant improvement over doing the full 18x18x18 = 5832 configuration scan.

By studying all of these test cases, we can begin to draw some general conclusions. First of all, we will point out that we need at least $2^d n_{\text{max}}$ configurations or data points before we can run the LLS fitting routine. This is the number of parameters being fit and thus is the minimum number of data points that will ensure that the linear system of equations that results from minimizing Equation 2-3 is overdetermined. Now looking through Figures 3-5 – 3-8, we can say further that, with one exception, twice this minimum number (i.e. $4^d n_{\text{max}}$ or 20 for the 1D cases, 40 for the 2D cases, and 60 for the 3D case) gets us past the "GAFF point" where we are improving on the default GAFF parameters and can feel comfortable that we are calculating reasonable parameters from the fitting routine. Three times the minimum number gives us reasonable parameters without exception. So, we make the general recommendation that when simultaneously parameterizing $d$ dihedrals, the absolute minimum number of configurations that should be used in the fitting routine be $4^d n_{\text{max}}$, preferably $6^d n_{\text{max}}$. Data points added beyond this number will generally improve the RMSE further.
Figure 3-1. Population density for 1-butanol. A) Population numbers, on a 1°-spaced grid, of 2 ns MD trajectory on 1-butanol, with respect to the two dihedral angles C-C-C-C and C-C-C-O. B) and C) Population numbers of two most populous clusters from the above MD data; MMTSB clustered the rest of the trajectory into the seven other obvious choices.
Figure 3-2. Configurations selected for the first MD cluster random sampling on 1-butanol with $n = 10$.

Figure 3-3. Configurations selected for MD cluster polar-gridded sampling on 1-butanol with $r = 3$ and $\theta = 5$, showing the greater spread in selected configurations compared to the MD cluster random sampling case.
Figure 3-4. Sample molecules used in testing of random configuration sampling for dihedral parameterization.
Figure 3-5. RMSEs between fitted MM PES and QM PES for 1-butanol, using configurations randomly selected off of a 72x72 grid in dihedral space, averaged over 20 separate sets iteratively built up from 20 to 100 configurations.
Figure 3-6. RMSEs between fitted MM PES and QM PES for various molecules using configurations randomly selected off of a grid (A) O=CH-NH-CH$_3$ and (B) H-S-S-H fragments and (C) butane, 72 point grid for butane or 36 point for others) in dihedral space, averaged over 20 separate sets iteratively built up from 10 to 35 configurations.
Figure 3-7. Two-dimensional RMSEs between fitted MM PES and QM PES for various molecules using configurations randomly selected off of an 18x18 point grid in dihedral space (A) C-C-S-S and C-S-S-C and B) C-S-S-C and S-S-C-O dihedrals from the top right molecule in Figure 4), averaged over 20 separate sets iteratively built up from 20 to 100 configurations.
Figure 3-8. RMSEs between fitted MM PES and QM PES for a 3-dimensional fit of the top right molecule in Figure 4, using configurations randomly selected off of an 18x18x18 point grid in dihedral space, averaged over 20 separate sets iteratively built up from 60 to 200 configurations.

Table 3-1. RMSEs between fitted MM PES and QM PES for 1-butanol, using \( n \) configurations randomly selected out of each of the 9 clusters for the 1-butanol MD data. Also shown are the RMSEs for the default (GAFF) parameter set, and the RMSE for the parameter set fit by using a full 72x72 scan (5184 configurations). RMSEs in kcal/mol.

<table>
<thead>
<tr>
<th></th>
<th>RMSE (near minima)</th>
<th>RMSE (full PES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n = 10 ) run #1</td>
<td>0.804</td>
<td>0.795</td>
</tr>
<tr>
<td>( n = 10 ) run #2</td>
<td>0.725</td>
<td>1.237</td>
</tr>
<tr>
<td>( n = 20 ) run #1</td>
<td>0.824</td>
<td>1.065</td>
</tr>
<tr>
<td>( n = 20 ) run #2</td>
<td>0.708</td>
<td>0.871</td>
</tr>
<tr>
<td>GAFF</td>
<td>0.800</td>
<td>0.839</td>
</tr>
<tr>
<td>Full scan (72x72) fit</td>
<td>0.449</td>
<td>0.426</td>
</tr>
</tbody>
</table>
Table 3-2. RMSEs between fitted MM PES and QM PES for 1-butanol, using the MD cluster polar gridded sampling method, with $r$ radial bins and $\theta$ angular bins about each minima. RMSEs are in kcal/mol.

<table>
<thead>
<tr>
<th>$r$</th>
<th>$\theta$</th>
<th>RMSE (near minima)</th>
<th>RMSE (full PES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>0.916</td>
<td>4.613</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>0.893</td>
<td>5.323</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>0.723</td>
<td>1.914</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>0.726</td>
<td>1.218</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>0.626</td>
<td>1.000</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>0.752</td>
<td>2.019</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1.116</td>
<td>4.316</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0.698</td>
<td>0.849</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0.629</td>
<td>0.908</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.642</td>
<td>0.581</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>0.673</td>
<td>0.599</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>0.650</td>
<td>0.664</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>0.672</td>
<td>0.635</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.580</td>
<td>0.791</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.654</td>
<td>0.636</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>0.676</td>
<td>0.673</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>0.625</td>
<td>0.590</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>0.598</td>
<td>0.575</td>
</tr>
</tbody>
</table>
CHAPTER 4
HYDROGEN MASS REPARTITIONING

Background

The length of a MD simulation is defined in units of the time step employed in the integration of the equations of motion. The size of the time step of a MD simulation is constrained by the timescale of the highest frequency motions in the system. These are typically bond vibrations involving hydrogen atoms; the typical limit given for running unrestrained MD simulations is 1 fs due to these fast vibrations. Increasing the time step beyond this number potentially makes the simulation unstable, with a higher probability of instabilities appearing the longer that the simulation is run. The 1 fs limit is commonly combated with the introduction of restraints on the high frequency bonds. In Amber, the most common simulation setup involves the application of the SHAKE algorithm to non-water bonds involving hydrogens, and the analytic SETTLE algorithm to water bonds and angles, with a 2 fs time step. This 2 fs limit is due to limitations in the SHAKE algorithm itself.

Apart from simulation stability issues, there are also formal issues with increasing the time step. Amber, for instance, uses the leapfrog integrator method for integrating the classical equations of motion in the MD simulation. The velocity Verlet / leapfrog family of integrators has an error, when taken over the entire trajectory, that is second order in the time step. Practically, what this means is that different time steps lead to different trajectories, and the larger the time step, the further the computed trajectory deviates from the “correct” one prescribed by Newtonian mechanics. This is only a

---
problem if the ensemble sampled from the frames of the trajectory is significantly altered from the original ensemble, thus changing any observables calculated over the whole trajectory. In fact, symplectic integrators, like velocity Verlet and leapfrog, have been shown to give the exact solution to a so-called “shadow Hamiltonian” which can be expressed as a Taylor expansion in the time step.\textsuperscript{87–89} This allows one to, in theory, analytically express the error introduced into the system energy by the size of the time step, and predict the effect on the sampled ensemble. In practice, this type of analysis has been used to predict the energy drift seen in NVE simulations.\textsuperscript{87} However, employing a thermostat in an NVT or NPT simulation, as is more widely used, tends to obfuscate the problem. For instance, NVT simulations do not usually suffer from the same type of energy drift seen in NVE simulations.\textsuperscript{90} This is dependent on the type of thermostat algorithm employed; in some sense, with some thermostats, accumulating discretization errors in the total system energy is “fixed” by siphoning out excess energy introduced by the integrator into the heat bath. For some thermostats, conserved energies can be defined that allow the same type of time step error analysis as with microcanonical simulations. Still, the formal description of the discretization error becomes more difficult with a thermostat. There has been some work to this end, particularly with the use of Langevin dynamics as a thermostat.\textsuperscript{91,92} In this manuscript, we take a more empirical approach in showing that the longer time step MD trajectories we employ do not introduce significant discretization errors, and do not attempt to provide a formal theoretical framework for the method.

With these two limitations (simulation stability and formal discretization error) in mind, we turn to hydrogen mass repartitioning (HMR) as a method to increase the time
step of the MD simulation by a factor of two, while keeping within the bounds set by both limitations. The idea of changing atomic masses in order to speed up MD simulations has been around since at least the 1970s.\textsuperscript{93} The main idea behind the method is that equilibrium thermodynamic averages of observables do not depend on the exact mass distribution of the system. For the canonical ensemble, these averages take the form:

$$
\langle A \rangle = \frac{\int A e^{-\beta H} \, dp \, dx}{\int e^{-\beta H} \, dp \, dx}
$$

If $A = A(x)$ and the Hamiltonian is separable in position and momentum, as is the case with MD with classical force fields without magnetic terms, then the momentum dependent parts of the integrals on the top and bottom can be separated out and thus cancel. Assuming that the force field does not contain any mass-dependent terms, the remaining expression is independent of the mass distribution of the system.

Some early work actually focused on employing a general mass tensor to the kinetic energy part of the Hamiltonian, with entries calculated from a normal modes analysis of the system,\textsuperscript{94} which has branched into a separate line of inquiry that we will not focus further on here. Over the next decades, there were a few studies\textsuperscript{95–99} done that involved simply increasing the masses of the hydrogen atoms (or in some cases every atom) in the system in order to slow down their motion. The results from these studies were promising, displaying the ability to speed up MD simulations by about a factor of 10, seemingly. However, Feenstra et al.\textsuperscript{22} pointed out that increasing the total system mass by selectively changing atomic masses in this way actually scales the time of the simulation, so that reported simulation lengths are off by a factor related to the ratio of system masses between the original and changed systems. In this seminal
study, Feenstra et al. prescribed the repartitioning of mass amongst the atoms in the system, so that the total system mass is kept constant. In other words, the mass of the light hydrogen atoms is increased while the heavy atoms’ masses are decreased by the same amount. This study became the basis for several implementations of the HMR technique, particularly in the GROMACS\textsuperscript{100} and ACEMD\textsuperscript{101} programs. In the years since, there have been few works building on this initial survey; one of the more recent studies by Rao and Spichty did not use HMR, in fact, but rather applied the older method of increasing the mass of only the solute hydrogens.\textsuperscript{102} The longest simulation that was run in the study of Feenstra et al. was 1 ns. With modern hardware, it is possible to study the effects of long time step HMR MD on trajectories of hundreds of nanoseconds, which is our focus here. It is important to show that a lengthened time step using HMR does not introduce significant errors in long simulations, and can indeed keep the simulation stable over long timescales.

**Methods**

**Mass Repartitioning Scheme**

A slightly different mass repartitioning scheme than used in previous studies was employed in our investigation. The scheme of Feenstra et al. scales all hydrogen masses by a factor of 4, while decreasing the masses of the bonded heavy atoms by enough to keep the total system mass constant. Due to technical reasons with the GPU code in Amber, we use a scaling factor of 3. This factor is also more satisfactory when applied to methyl groups, as a scaling factor of 4 leads to the methyl carbon having a lower mass (3 amu) than the methyl hydrogens (4 amu). Figure 4-1 shows how this scheme is applied to alanine dipeptide. Three different mass topology schemes were studied: normal masses (“norm”), HMR applied to only the solute atoms (“repart”), and
HMR applied to the solute and solvent atoms (“repart_water”). The application of the analytic SETTLE algorithm, and the total absence of high frequency intramolecular movement in the TIP3P model, means that the water molecules can already be simulated at a higher time step than 2 fs without affecting simulation stability; how this affects discretization error does not have a clear a priori answer.

All MD simulations were performed with the CUDA accelerated Amber 1424,31,103,104 program. Peptide and protein force field parameters were obtained from the ff12SB force field (a newer revised version of the ff99SB39,49 force field), and the original MTS spin label parameters came from GAFF47 with charges derived from a RESP50 fit using an HF/6-31G* electrostatic potential calculated using the Gaussian 09 program.80 The five linking dihedrals in the spin label model, including χ, were reparameterized via a simultaneous five-dimensional fit with MP2/6-31G*/HF/6-31G* data using our previously reported dihedral parameterization method.82 All trajectories were analyzed with the cpptraj105 program in Amber.

**Peptide**

The first system that will be presented is a small capped peptide, (Ala)₃ (Figure 4-2A). The structure was built and solvated with the LEaP Amber program with an octahedral box, leading to the addition of 1106 TIP3P106 water residues, for a total of 3360 total atoms. Due to the small size of the solute, stable MD trajectories can be run for this system with time steps of up to 4 fs with normal masses and 5 fs with HMR applied. We will explore further what happens past these time steps in Section IV. Ten trajectories were run for each time step of 1, 2, 3, and 4 fs for the “norm” mass topology, and for time steps of 1, 2, 3, 4, and 5 fs for the “repart” and “repart_water” topology types, giving 140 total trajectories. Each trajectory was independently equilibrated,
consisting of minimization, 200 ps of heating to 300 K, and 100 ps of density equilibration with a constant pressure simulation. From the equilibrated structure, each trajectory was run for 450 ns of NVT production time, with energies and structures saved every 0.48 ps. In all production runs in this study, all covalent hydrogen bonds were restrained with SHAKE/SETTLE, and Langevin dynamics with a weak coupling (1 ps$^{-1}$ collision frequency) was used as a thermostat.

**Protein – Structure**

The initial HEWL structure (Figure 4-1B) was obtained from a crystal structure (Protein Data Bank ID: 4LYT). With LEaP, eight Cl$^{-}$ counterions were added to neutralize the protein’s total charge; the resulting structure was solvated with an octahedral box, with 5736 TIP3P water residues added for a total of 19152 atoms. With the protein, only trajectories with time steps up to 3 fs for normal masses and 4 fs with HMR were stable. Fifteen trajectories were run for time steps of 1, 2, and 3 fs for the norm topology and for time steps of 1, 2, 3, and 4 fs for the repart and repart_water topologies, giving 165 total trajectories. For the protein, a common equilibrated structure was used as a starting point for all trajectories, equilibrated through a similar process to the peptide, with 2 ns of additional NVT equilibration time. Each trajectory was then run for 160 ns of production time in the NVT ensemble, with energies and structures saved every 12 ps.

**Protein - pK$_a$**

The option to run constant pH MD simulations$^{108}$ was recently added to the GPU accelerated Amber code. We also investigated the effect of long time step HMR trajectories on the calculation of pK$_a$ values for titratable residues in HEWL using this feature. To this end, we ran two sets of trajectories (2 fs time step with norm mass
topology and 4 fs time step with the repart topology) with ten residues specified as targets for attempted protonation state changes in the Generalized Born implicit solvent model. These were run for pH values from 0 to 7 in half pH increments, with 20 independent trajectories run for each pH value starting from a common equilibrated structure. Each trajectory was run for 50 ns, with protonation state changes attempted every three MD steps. Titration curves for each trajectory set were derived through the calcpka program in Amber. pK_a values were then calculated by fitting the modified Hill equation to the titration curve:

\[ f_d = \frac{1}{1 + 10^{n(pK_a - pH)}} \]

This fit was performed by algebraically minimizing the least squares term of \( f_d \) to the data.

**Spin Label**

The MTS spin label has been used to study protein dynamics (e.g., flap movement in the HIV-1 protease)\(^{109}\) in EPR experiments and has previously been the target for force field parameterization efforts,\(^{37}\) as accurately characterizing the dynamics of the label, particularly movement in the flexible chain linking the spin label to the protein, is important for interpreting EPR measurements. The model we use for the spin label is a single capped cysteine residue attached to the spin label via a disulfide bond (Figure 4-1C). The model’s initial structure was built with the Vega ZZ\(^{110}\) modeling software, and solvated in LEaP with 848 TIP3P water residues, for a total atom count of 2593. The resulting system was equilibrated using the same process as for the peptide. The barrier to rotation about the linking dihedral centered on the disulfide bond (labeled \( \chi \) in Figure 4-1C) was then analyzed via umbrella sampling. We used 120 windows of
7.5×10^4 biased MD steps, with the centers of bias for the windows equally spaced over the full 360° range of rotation. Two trajectory types were run for this test: 2 fs time step (150 ps per window) with the norm topology and 4 fs time step (300 ps per window) with the repart topology. The weighted histogram analysis method (WHAM)\textsuperscript{111} was then used to remove the effect of the bias in the individual windows and produce a single PMF for the dihedral.

**Results**

**Peptide**

We studied the dynamics of the peptide in terms of the three sets of φ/ψ angles along the length of the molecule. Figure 4-3 shows the accumulated population statistics for the middle (i.e., spanning the second alanine residue) φ/ψ angles of the peptide, integrated over all other degrees of freedom, taken over all 10 trajectories for each time step/topology combination.

As can be seen in Figure 4-3, the Ramachandran populations are remarkably similar. The RMS error between the norm/1 fs populations and the other topology / time step combinations are (in units of 10^{-4} % population) for the norm topology, 2-4 fs: 1.46, 1.57, 1.82; for the repart topology, 1-5 fs: 1.35, 1.93, 1.61, 2.21, 2.30; for the repart_water topology, 1-5 fs: 1.36, 1.62, 1.44, 3.26, 1.59. It is somewhat surprising that the same level of agreement extends to the 4 fs (norm topology) and 5 fs (repart and repart_water topologies) trajectories, as these time steps are higher than usually prescribed by NVE energy drift studies. We also point out here that the real walltimes to run the longer time step simulations for the same amount of simulation time (450 ns) were shorter. The 5 fs trajectories ran in about a fifth of the time as the 1 fs trajectories.
The main comparison of interest, the norm/2 fs trajectories to the repart/ 4 fs (or repart_water/ 4 fs) trajectories, represents an approximate doubling of throughput.

For a more in-depth investigation of the sampling in each trajectory type, we defined three semi-stable conformations for each alanine residue: beta sheet (β) (combining the canonical beta sheet and poly-proline II conformations), right handed alpha helix (αR), and left handed alpha helix (αL). We investigated the relative sampling and kinetics among the 27 unique rotamers defined by the permutations of these three local conformations in each of the three alanine residues. A definition of “semi-stable” was derived based on free energies calculated over the ten norm/1 fs trajectories, for the central φ/ψ coordinates. A cut-off of 2 kcal/mol was applied to this two-dimensional free energy surface to define a φ/ψ “mask” (Figure 4-4); every configuration in φ/ψ space for that particular residue that had a free energy less than the cut-off was considered semi-stable. For each trajectory frame, the mask was applied to each alanine residue, and if the conformation of all three residues fell within the mask, that frame was marked as a semi-stable rotamer and categorized into one of the 27 rotamer states based on the conformation of the individual alanine residues. Figure 4-5 shows the fractional populations for the four most populated of the 27 rotamers, averaged over the ten trajectories for each trajectory type. Figure 4-6 shows average transition rates between rotamers for the transitions with the highest frequency. A transition was counted as soon as the peptide moved from one rotamer conformation to another; i.e., the given rates include transitions between the two rotamers along any pathway that did not contain another rotamer conformation.
The plots clearly show that there are no significant differences between the various topology/time step trajectory types in terms of relative sampling. The main systematic variation that is noticeable in the population numbers is a slight downward drift with increasing time step, which is most easily visible in the highly populated N-(β-β-β)-C rotamer. With increasing time step, the peptide spends slightly less time in the stable rotamer conformations. No systematic variation is observed from the application of HMR. Two trends are immediately apparent when comparing transition rates for all but the slowest transition shown. For one, there is a gradual increase in transition rate with time step. Also, the repart_water topology trajectories show a consistently lower rate for all transitions (approximately 10% lower on average). In other words, observed rates of interconversion among the rotamers were damped when applying HMR to the solvent. This agrees with observations from Feenstra, et al., which attribute a decrease in conformational space sampling efficiency when applying HMR to an increase in viscosity. In the repart trajectories, viscosity is not affected by HMR, and thus a similar sampling efficiency to the norm trajectories is observed. This conservation of sampling efficiency comes at the price of increased discretization error, at least in the solvent dynamics.

Typically, discretization error is more accurately monitored by calculating energy drift in an NVE simulation. As previously mentioned, constant temperature simulations do not express discretization error in the same way as NVE simulations; there is no energy drift during the course of the trajectory. In terms of energetics, the most noticeable difference between trajectories with different time steps is the average potential energy; there is a shift towards higher energy with an increase in time step.
This phenomenon has been noted before\textsuperscript{102}. We reiterate that this is solely due to discretization error from the time step, and repartitioning masses with HMR does not lead to further energy increase. Figure 4-7 shows the average potential energies for each trajectory type, taken over all ten trajectories for that type.

The repart\_water topology type shows the most stable behavior with increasing time step, with a consistently lower potential energy for all time steps compared to the other two topology types. This trend seems to indicate that errors in the water librational motion are the main contributor to the shift in energy surface being explored. The repart\_water topology damps the water librational motion with HMR while the norm and repart topologies do not, which would explain the consistently lower potential energy seen for this topology.

**Protein – Structure**

Figure 4-8 shows a histogram of the total backbone RMSD to the 4LYT crystal structure for each of the trajectory types, taken over all 15 trajectories of each type. Figure 4-9A shows the per-residue RMSD to the same common structure, while Figure 4-9B shows the RMS fluctuations of the alpha carbon in each residue to its average position over all trajectories of that type.

As with the peptide, there are no significant deviations in structure or flexibility observed by the application of HMR or with the different time steps employed. The per-residue RMSD is particularly similar, while the RMS fluctuations show a bit more variation around the flexible random loops about residues 70 and 100; this can be attributed to a combination of lack of sampling and of biases in the different average structures being used for comparison. Figure 4-10 shows the average potential energy for each trajectory type taken over all 15 trajectories.
The relative shifts here are about six times higher than observed in the peptide, which corresponds roughly to the ratio of the system sizes. The relative behavior of the different topologies with respect to increase in time step is similar to the behavior seen with the peptide; in particular, once again, the repart_water topology shows the least increase in system energy with increasing time step.

To get a more specific view of the conformations being sampled in the protein system, Figure 4-11 shows the profile of the distance between the termini of the protein for each trajectory type. Again, no significant difference between the various trajectory types is observed. Another useful tool for exploring collective motions in proteins is the application of principal component analysis (PCA) to the backbone motion; this type of analysis has been previously employed to compare trajectories with different advanced sampling techniques (e.g., accelerated molecular dynamics\textsuperscript{18}). The PCA functionality in cpptraj was used for the following analysis. PCA provides collective modes of the protein, ordered by their contribution to the variance of the motion over the whole trajectory. For each trajectory, PCA gives a set of eigenvectors corresponding to each mode obtained by diagonalizing the covariance matrix over the trajectory. It is expected that the eigenvectors between the different trajectory types, even from one trajectory to the next within the same type, will not be identical, as they are very sensitive to relatively small differences in the fluctuations from one trajectory to the next. However, these principal components (PCs) provide a worthwhile comparison via projecting the protein’s motion over a trajectory onto a particular mode. We selected a single set of eigenvectors obtained from PCA on a single norm/1 fs trajectory, and used those as a basis set for comparing between trajectories. Figure 4-12 shows the projections onto
the first four eigenvectors from this set for each trajectory type. We again point out that
the observed structure of the protein is independent of both the mass distributions and
of the time steps employed in our tests.

Although eigenvectors obtained from PCA on one trajectory differ from those obtained from a different trajectory (even between trajectories of the same type), it is worthwhile to compare how well eigenvectors from one trajectory cover the space of a PC from a different trajectory. To this end, we investigated the behavior of eigenvector coverage versus number of eigenvectors in order to compare the various trajectory types, where we define eigenvector coverage, $c$, as:

$$c = \sum_{i} \left( \vec{v}_{o} \cdot \vec{e}_{i} \right)^2$$

Here, $\vec{v}_{o}$ is the first eigenvector from the trajectory being used as a basis for comparison and the set $\vec{e}_{i}$ are the first $N$ eigenvectors of the trajectory being compared. Figure 4-13 shows a plot of $c$ vs $N$, averaged for each type of trajectory, using the first eigenvector from the previous reference set as $\vec{v}_{o}$. Again, the results show that using HMR and an increased time step does not affect the conformational ensemble, while increasing throughput (by a factor of two, when comparing the norm/2 fs to the repart/4 fs and repart_water/4 fs trajectories).

**Protein - pK$_a$**

Figure 4-14 shows the evolution of the calculated pK$_a$ values for three of the ten targeted HEWL residues over the run time of the trajectories, in terms of invested processor time. Note that the actual walltime for these calculations was less by a factor of 15 as the different pH values were run in parallel.
The two most interesting aspects to these plots are that the norm/2 fs and repart/4 fs trajectories converge to the same pK$_a$ within error, and that the repart/4 fs trajectories converge approximately twice as fast as the norm/2 fs trajectories, which indicates that about half of the real calculation time is required to arrive at the same result. We note that in the case of constant pH MD, one important degree of freedom, namely the protonation state, is not being accelerated with the application of HMR with long time steps. This means that there are some cases where pK$_a$ calculations as shown here will not necessarily converge more quickly from applying longer time steps to the configuration space sampling portion of the constant pH MD. However, this is a separate issue from the utility of long time step HMR MD.

**Spin Label**

Figure 4-15 shows the PMF calculated via WHAM for the linker dihedral, for the norm/2 fs and repart/4 fs trajectory types. The profiles are practically identical, in particular the relative energies of the ±90° conformations and the barrier between them. The results here and with the pK$_a$ calculations indicate that the increase in time step with HMR applied conserves relative conformational energies. Although the average sampled energy surface lies slightly above the original one, the relative global energetics of the studied systems appear to be unaffected.

**Discussion**

Using several examples, we have shown that HMR allows us to increase the MD time step without changing relative conformational sampling or kinetics of the trajectory. One further detail that is worth some comment is in the mixing of HMR with constraint algorithms, particularly SHAKE. In our tests on our largest system (HEWL), using the default Amber SHAKE settings, we observed that SHAKE failure typically occurred
before any significant problems from energy conservation breaking down arose. Here, we would like to focus on the mechanics by which HMR preserves simulation stability beyond SHAKE’s capabilities. SHAKE is used to effectively freeze the fast bond vibrations involving hydrogen atoms in the solute, so it would initially seem odd that HMR applied to the solute hydrogens provides any further stabilization over the application of SHAKE by itself. As previously mentioned, the 2 fs time step limit that comes with the use of SHAKE derives from limitations within the SHAKE algorithm itself, and is not a fundamental limit due to high frequency motions like the original 1 fs limit. Beyond 2 fs, for large systems, there is a high chance of SHAKE failing due to the algorithm’s limitations (this is somewhat dependent on the exact tolerance settings used, so the discussion that follows assumes default Amber SHAKE settings). The source of this limitation can be understood by briefly exploring how the SHAKE algorithm works. In what follows, we will consider the case of a single round of SHAKE involving only two atoms (e.g., a Cα/Hα pair), referring to Figure 4-16.

During an iteration of SHAKE, as implemented in Amber, the two atoms (A and B) are moved back to the target distance (R₀) along the axis connecting them in the previous MD step (z). We consider two cases from the perspective of atom A (i.e., atom A does not move): for the first, the SHAKE algorithm is able to successfully move atom B back to the original distance. This case is shown in green in Figure 4-16. However, it is possible that atom B moves far enough in a given direction from A that it is not possible to move the atom back to R₀ along the axis z (shown in grey). What happens at this point is implementation-dependent. For example, in the CPU version of Amber 14, the simulation will exit in this case with an error. However, in the GPU version, for
efficiency reasons, the simulation will continue even though the SHAKE iteration did not successfully converge. With the usual tolerance settings on large systems, this quickly leads to instabilities that manifest as the simulation “blowing up.” Obviously, with an increase in time step, the distance moved by each atom between each run of SHAKE is increased on average, leading to a larger probability of the “grey” case. The application of HMR damps this increase in average instantaneous displacement from MD step to MD step in the hydrogen atoms, leading to a decrease in the probability of this SHAKE failure.

For each topology, there was a “border” time step, where the simulation experienced SHAKE instabilities after more than a few MD steps, but still on the order of tens of nanoseconds. For instance, for HEWL, the border was 4 fs for norm and 5 fs for repart and repart_water. In this system, the norm/4 fs trajectories ran at most until 14 ns before experiencing SHAKE instabilities, with most failing before 10 ns, while the repart/5 fs and repart_water/5 fs trajectories ran for at most one nanosecond before failing. The role of HMR in extending the stability of systems (with respect to SHAKE) to longer time steps is in pushing this “border” of SHAKE instability to longer time steps without affecting conformational space.

In this part of the study, we have examined a technique that allows MD simulations, as well as advanced sampling techniques based on MD like umbrella sampling and constant pH MD, to be sped up by about a factor of two. We have also explored in some small detail how HMR stabilizes the system on top of the SHAKE algorithm. As shown, the application of HMR with long time step MD trajectories does not introduce significant additional error compared to the “usual” case of 2 fs trajectories.
with normal masses. Additionally, HMR does not place any significant overhead on the user; it is trivial to incorporate mass repartitioned topologies into existing simulation setups.

One interesting remaining point for further study is more fully exploring the relative utility of the “repart” topology type versus the “repart_water” type; i.e., what advantages and disadvantages there are to applying HMR to the solvent atoms with the increased time step. Our results here indicate that the time step can be doubled in a stable fashion without repartitioning the masses of the TIP3P model, which avoids increasing the viscosity of the system and thus decreasing the sampling efficiency of the trajectory. This is due to the application of the analytic SETTLE algorithm in this water model; this algorithm does not suffer the same limitations as the iterative SHAKE. However, leaving the water molecules unchanged while increasing the time step leads to more sampling of unfavorable configurations due to the less accurate integration of the librational movement in the water molecules and to higher stress in the solvent distribution. If the protein sampling is truly unaffected, as our results seem to indicate, then this would be the preferred simulation setup for protein studies as solute sampling efficiency is then truly doubled without the viscosity damping effect. A possible alternative to this approach could be to apply an overall mass scaling to only the solvent molecules after applying HMR, decreasing the total solvent mass and thus lowering the system viscosity. Lin and Tuckerman\textsuperscript{112} recently investigated a similar method as applied to peptide simulations. Although, as previously mentioned, the time scale of the solvent motion would be affected, it is possible that this type of setup would allow an accelerated sampling of the protein motion without the viscosity effect.
Figure 4-1. Example of the HMR scheme used in this study applied to alanine dipeptide; masses in amu shown for the original system before mass repartitioning (a) and after applying HMR (b). Note that the total system mass is not changed by applying HMR.
Figure 4-2. Structures of systems studied: A) Capped \((\text{Ala})_3\); B) HEWL protein; C) MTS spin label model.
Figure 4-3. Population statistics of central $\phi/\psi$ angles in the (Ala)$_3$ peptide for different time step and mass topology combinations. Axis scale shown in lower left plot. Note that configurations which accounted for less than $10^{-3} \%$ of the total time were culled from the plot, and that all plots are shown on the same color scale.
Figure 4-4. $\varphi/\psi$ mask derived from norm/1 fs free energies. Regions of $\varphi/\psi$ space colored black were considered semi-stable, and the mask was applied to each alanine residue in the peptide to define global rotamer conformations, based on the three shown local conformations ($\beta$, $\alpha_R$, $\alpha_L$).
Figure 4-5. Fractional populations of the most populated rotamers in the (Ala)$_3$ peptide, averaged over ten 450 ns trajectories. Norm/1 fs average shown as grey line for reference.
Figure 4-6. Average transition rates between rotamers with highest transition frequency in (Ala)$_3$ peptide. Norm/1 fs average shown as grey line for reference.
Figure 4-7. Average potential energies with different mass topology / time step combinations for the (Ala)$_3$ peptide, relative to the norm/1 fs average, averaged over ten trajectories of each trajectory type. The error bars show the standard deviation in the mean over the ten trajectories.
Figure 4-8. Backbone RMSD to a crystal structure in HEWL for each trajectory type.
Figure 4-9. Backbone RMSD and \( \alpha \) RMS fluctuations of HEWL - A) Per-residue backbone RMSD to a crystal structure in HEWL. B) Per-residue \( \alpha \) RMS fluctuations against the average structure of each trajectory type.
Figure 4-10. Average potential energies with different mass topology / time step combinations for HEWL, relative to the norm/1 fs average, averaged over 15 trajectories of each type.
Figure 4-11. Terminal-terminal distance for HEWL averaged over each type of trajectory.

Figure 4-12. Projections of HEWL trajectories onto top 4 eigenvectors from PCA on the protein backbone of a common norm/1 fs trajectory.
Figure 4-13. Plot of $c$ vs $N$ (see text) averaged over each type of trajectory. Note that error bars are shown for averaging over 15 trajectories for each trajectory type, except for norm/1 fs for which 14 trajectories are used (one normal trajectory is used as the reference).
Figure 4-14. Evolution of calculated pK\(_a\) values using constant pH MD on HEWL for two different trajectory types in three of the ten residues treated: A) GLU7, B) HIS15, and C) ASP87. Error bars are from averaging over 20 independent trajectories in each pH for each trajectory type.
Figure 4-15. PMF for rotation about the dihedral $\chi$ in the MTS spin label, generated through umbrella sampling.

Figure 4-16. Graphic representation of a round of the SHAKE algorithm. The distance between the two atoms (labeled A and B) at the end of the previous MD step is $R_0$. Two cases are shown: in one, the atom B moves a small distance in the current MD step to $B_i$, and is moved along the original z axis back to a distance of $R_0$ at $B_i'$. In the other, the atom B moves a farther distance to $B_{i,\text{fail}}$ at which point it is not possible to move back to $R_0$ along the axis z.
CHAPTER 5
QM/MM SIMULATIONS USING GPU-ACCELERATED DFT

Background

QM versus MM

The force field approach of MM becomes inappropriate when considering, for example, chemical reactions. Force fields are parameterized to properly describe the bond energy only near equilibrium lengths, as they usually use a simple harmonic term to describe it. In order to properly model the energy of the system as a chemical bond is being broken and reformed, a QM Hamiltonian is required. However, as mentioned in the Introduction, full QM based energy calculations with \textit{ab initio} or density functional theory (DFT) methods have typically been computationally intractable for models beyond what would be considered a small system, on the order of 100 atoms.

So-called semi-empirical methods, including AM1, PM3, and DFTB, have somewhat bridged the gap between MM and QM. By using formal approximations or parameterizing certain integrals (corresponding to the calculation of certain matrix elements) in the QM calculation, these methods avoid the full computational cost of \textit{ab initio} and DFT methods. However, semi-empirical methods still involve matrix diagonalization and are not capable of being applied to systems of the size of biological macromolecules, and they do not have the chemical generality of pure \textit{ab initio} methods or DFT.

In the case of studies involving enzyme catalysis, the system being modeled involves only a small region, including the protein active site and substrates, where a chemical reaction is occurring. Meanwhile, the rest of the system remains relatively electronically stable, with no large changes in the electron density occurring. In these
cases, a QM Hamiltonian is needed to model the small active region, but it is not feasible to apply full QM over the entire system, which could include tens of thousands of atoms. It is desirable to apply QM only to the active region that requires it, while using MM to model the rest of the system. Thus, the method of QM/MM has been developed to treat significant changes in electronic density within a large chemical background.\textsuperscript{57} The QM/MM method is increasingly becoming a popular technique to model enzyme catalysis. The Nobel Prize in chemistry for 2013 was awarded “for the development of multiscale models for complex chemical systems” to the original developers of QM/MM: Arieh Warshel, Michael Levitt, and Martin Karplus.

**QM/MM**

In the QM/MM methodology, the system being modeled is split into two regions or subsystems: the QM region where changes in the electronic density are more accurately modeled using a quantum Hamiltonian, and the MM region where the other degrees of freedom (background) are treated using computationally cheap force field methods. There are many ways of treating the interaction between the two subsystems, which has led to a variety of QM/MM methods.

In general, the QM/MM method needs to account for three main types of interactions between the two subsystems: short range van der Waals type interactions, electrostatic interactions, and, if the boundary between the two regions passes through a bond, the energy due to cut bonds. The van der Waals interaction is typically handled in the force field calculation and treated in the typical fashion of MM. There are a variety of methods to handle bond cuts in QM/MM. More complex techniques explicitly account for the type of bond being cut (its hybridization, etc), and will potentially rearrange charges to more properly account for nearby electrostatics. Perhaps the
simplest technique is the link atom approach, where the cut bond is simply capped with an added atom (“link atom”) in the QM region, typically a hydrogen atom. The electrostatic interaction is usually handled in one of two ways: mechanical embedding or electronic embedding. With mechanical embedding, the electrostatic interaction is handled in the force field calculation. Typically, the atomic centers of the QM region are assigned point charge values using a method like AM1-BCC or RESP, and the electrostatic interaction is then handled as a simple point charge interaction. With electronic embedding, the electrostatic interaction is handled in the \textit{ab initio} or DFT calculation. In this case, a new term is added to the QM Hamiltonian, a sum of one-electron integrals that accounts for the energy of interaction between the QM region electron density and the atomic center point charges in the MM region. Mechanical embedding has the advantage of being simpler to implement and computationally cheaper, while not properly modeling polarization of the electron density in the QM region. Electronic embedding more correctly accounts for polarization of the electron density, while being more computationally intensive and also requiring that the software being used to perform the \textit{ab initio} or DFT calculation supports QM/MM.

Though QM/MM allows the application of QM methods on a relatively small part within a large system, when using QM/MM with full \textit{ab initio} or DFT methods, the QM calculation is still the major computational bottleneck. Thus, the main focus of this work is in accelerating the QM calculation (in particular, using full DFT) for use with QM/MM molecular dynamics (MD) simulations of enzyme catalysis. When attempting to accelerate QM calculations for QM/MM in these cases, there are some unique design aspects to consider. We consider QM subsystems that are typically small, not larger
than 100 atoms. At each new MD step, the initial guess for the electron density is based on the density at the end of the last step; this usually means a smaller number of self-consistent field (SCF) iterations per MD step compared to single-point calculations. Thus, initialization has more relative importance.

**GPUs**

In this work, we look at one method of accelerating DFT calculations for use in QM/MM simulations. The use of graphical processor units (GPUs) in scientific computing has seen an explosion in popularity over the past decade. The introduction of simplified programming libraries, such as Nvidia’s CUDA (the platform used in this study), has led to the use of GPUs in many complex computational problems. Writing a CUDA kernel (user-defined code that runs on the GPU) is similar to writing C code, with a few extra keywords and library calls. The streaming SIMD/SIMT (single instruction, multiple data/thread) architectures that GPUs employ can be harnessed to accelerate QM calculations, which involve large data parallelism - the same mathematical operation being applied simultaneously over a large data structure. There have been several works over the past decade which have applied GPU porting to speed up a variety of electronic structure calculations.²⁶,₅₈–⁶⁴

NVIDIA GPUs are composed of multiple streaming multiprocessors (SMs) which independently schedule thread execution. Threads for kernel execution are specified by the user in blocks (maximum of 1024 threads/block in NVIDIA GPU “compute capability” (version) 3.x), and the kernel is called with a grid of thread blocks. Each block is sent to an SM (each SM can run multiple blocks simultaneously), and the SM splits up the block into warps (32 threads in compute capability 3.x) which are scheduled for execution by a warp scheduler (Figure 5-1).
NVIDIA GPU memory is also set up in a hierarchical fashion. The host (CPU) sends data to the device’s (GPU) global memory space (or constant or texture memory space), stored in device RAM, which is persistent across multiple kernel executions. Each block also has an on-chip shared memory space which is coherent across all threads in a block. Each thread will have access to its own registers, and also has an off-chip local memory space.

There are a few general statements that can be made to give a flavor for the type of considerations that go into accelerating scientific code with GPUs. Device-to-host (and vice-versa) data transfer is expensive, and should be minimized, and done asynchronously with other tasks when possible. Global memory access from kernel code is expensive, and should be replaced by shared memory access whenever possible (reading from global memory once and putting the data into shared memory for further calculations is a typical access pattern). However, it is important to take into account that the number of blocks that can be run on a single SM at a time is limited by available shared memory and registers. When global memory access is necessary, it should be coalesced as much as possible (i.e., threads in a warp should access sequential aligned memory). Shared memory is split into modules, or banks, and bank conflicts need to be avoided (different threads shouldn’t access the same shared memory bank in the same instruction). Diverging execution paths for threads in the same warp should be avoided whenever possible, and latency hiding should be enforced as much as possible (e.g., a single thread doing more work can sometimes hide the latency of scheduling a new thread).
LIO

Our collaborators from the Estrin group in Buenos Aires reported previously on efforts to introduce GPU acceleration in DFT calculations with the LIO software\textsuperscript{23} (http://www.github.com/nanolebrero/lio). LIO was designed for use in QM/MM MD simulations. In particular, LIO is currently designed for use with Amber 14’s sander program in order to run MD.

The largest computational cost in a DFT calculation is due to the calculation of the exchange-correlation terms; this was the focus of the Estrin group’s previous acceleration efforts. After moving this part of the DFT calculation to run on GPU, they showed that the overall runtime of MD with LIO (compared to the older CPU version) was sped up by a factor of four. After accelerating the exchange-correlation calculation, the next bottlenecks involved in a MD step using LIO involve the calculation of the gradients of the QM/MM and Coulomb interactions (Figure 5-2). For systems with large MM subsystems - several thousands of atoms - the QM/MM interaction in particular dominates the computation time in LIO. Thus, porting the calculation of these terms to GPU is the main focus for this work.

Theory

DFT gives the energy of the QM subsystem in terms of functionals of the electron density:

\[ E[\rho] = T_\text{s}[\rho] + V_{\text{ne}}[\rho] + \frac{1}{2} \int \int \frac{\rho(\vec{r}_1)\rho(\vec{r}_2)}{r_{12}} \, d\vec{r}_1 \, d\vec{r}_2 + E_{\text{xc}}[\rho] \]  

(5-1)

The terms on the right hand side are the electron kinetic energy, the density-nuclear interaction energy, the density-density Coulomb energy, and the exchange-correlation energy. The total energy for the QM subsystem also includes the nuclear-
nuclear interaction energy, which is independent of the density. With QM/MM, the energy of the whole system is decomposed into three terms:

\[ E_{\text{system}} = E_{\text{QM}} + E_{\text{MM}} + E_{\text{QM/MM}} \] (5-2)

The \( E_{\text{QM}} \) and \( E_{\text{MM}} \) terms are the energies for the QM and MM subsystems, respectively, calculated without considering the other subsystem. \( E_{\text{QM}} \) is given by Equation 5-1 with the nuclear-nuclear interaction, while \( E_{\text{MM}} \) would come from a pure force field calculation. \( E_{\text{QM/MM}} \) represents the interaction energy of the two subsystems and can be calculated in a variety of ways, which define different QM/MM methods. We use the QM/MM method of Amber, which uses link atoms and a van der Waals interaction directly between the subsystems, and treats the electrostatic interaction with the electronic embedding method: the interaction between MM point charges and QM density are treated in the DFT calculation itself similarly to \( V_{\text{ne}}[\rho] \) in Equation 5-1, up to a certain cut-off outside the QM subsystem:

\[ E_{\text{QM/MM,es}} = \sum_i q_i \int \frac{\rho(\vec{r})}{|\vec{r} - \vec{r}_i|} d\vec{r} \] (5-3)

In Equation 5-3, the sum is over MM point charges within the specified cutoff, with charge \( q_i \) and position \( \vec{r}_i \).

LIO employs basis sets of contracted Cartesian Gaussian functions. The electron density can be expanded:

\[ \rho(\vec{r}) = \sum_i |\psi_i(\vec{r})|^2 \]

\[ \psi_i(\vec{r}) = \sum_j c_i^j \chi_j(\vec{r}) \]

The \( \psi \) are the molecular orbitals (MO) while the \( \chi \) are the basis functions:
\[ \chi_j(\vec{r}) = (x - x_0)^{n_x^j}(y - y_0)^{n_y^j}(z - z_0)^{n_z^j} \sum_{m} k_m^j e^{-\alpha_m(\vec{r} - \vec{r}_j)^2} \] (5-4)

In terms of the density matrix \( p_{jk} \) and the basis functions, the electron density is written as:

\[ \rho(\vec{r}) = \sum_{j} \sum_{k \geq j} p_{jk} \chi_j(\vec{r}) \chi_k(\vec{r}) \] (5-5)

The total system energy can be described with the density matrix and the Kohn-Sham (KS) matrix:

\[ E[p] = \sum_{j} \sum_{k \geq j} p_{jk} KS_{jk} \] (5-6)

Thus, the QM/MM electrostatic energy can be written as:

\[ E_{QM/MM,es} = \sum_{i} q_i \sum_{j} \sum_{k \geq j} p_{jk} \int \frac{\chi_j(\vec{r}) \chi_k(\vec{r})}{|\vec{r} - \vec{r}_i|} d\vec{r} \] (5-7)

The evaluation of the integrals:

\[ \int \frac{\chi_j(\vec{r}) \chi_k(\vec{r})}{|\vec{r} - \vec{r}_i|} d\vec{r} \]

using contracted Cartesian Gaussian basis functions has been well studied, and is one of the simpler molecular integral types. We use the Obara-Saika\(^{113}\) (OS) recursive method to evaluate them (Equations A19-20 in OS) as well as their gradients (Equations A24-26 in OS).

The evaluation of the Coulomb terms in Equation 5-1 typically represents a major computational bottleneck for \textit{ab initio} or DFT calculations; these terms cause the base \( N^4 \) scaling. LIO uses two approximations to reduce the scaling of this term to \( N^2 \). One is to use the Shwarz inequality, employing a computationally cheap prefactor calculation to discard negligible terms. The other is the use of an auxiliary basis set in which the
density is expressed as a linear expansion. For the Coulomb calculations, the linear density basis is fit to the MO basis at each SCF iteration, cutting down the scaling by another order. Thus, the largest sets of integrals to be evaluated are three-center, two-electron rather than four-center:

$$\int \int \frac{\psi_i(\vec{r}_1)\psi_j(\vec{r}_1)\psi_k(\vec{r}_2)}{|\vec{r}_1 - \vec{r}_2|} d\vec{r}_1 d\vec{r}_2$$ (5-8)

Here, $\psi_i$ and $\psi_j$ are MO basis functions while $\psi_k$ is an auxiliary density basis function. The evaluation of these integrals and their gradients can be simply derived from the full Obara-Saika recursion relation (Equations 39 and 44 in OS). Basis sets have been developed specifically for this method, with an MO basis and corresponding auxiliary basis.

**Methods**

**QM/MM - Energies**

The following energy and gradient calculations have been implemented in LIO for any combination of s, p, and d angular momentum basis functions. We employed a basic mapping from Equation 5-6 to CUDA threads in which each thread was responsible for calculating and summing all of the MM charge interaction terms for a single pair of basis functions. By "basis function," we refer to a "primitive" function, not the whole contraction (single term in Equation 5-4 rather than the full sum). We denote a primitive Gaussian function by $\varphi_{jm}$ so that:

$$\psi_j(\vec{r}) = \sum_m \varphi_{jm}(\vec{r})$$

Thus each thread calculates at least one partial term in the KS matrix:
Each thread is mapped to an index \((j,k,m,n)\), which refers to a single pair of basis function radial parameters \((k_m^j, \alpha_m^k)\) in Equation 5-4). We found it more efficient to have each thread calculate all of the terms corresponding to the overlap of the two sub-shells that the basis functions \((j,k,m,n)\) belong in. This corresponds to the blocks in the density (or KS) matrix with basis functions belonging in the same sub-shell (1x1 block for two s functions, 3x1 block for one p and one s function, 3x3 block for two p functions, and so on). Therefore, the index \((j,k,m,n)\) refers to one of these blocks and not to a single density matrix element. Figure 5-3 shows the example matrix elements that particular threads are responsible for in a simple KS with only s and p type functions. We use the nomenclature \(p_{jk}^{1}\) to refer to the set of density matrix elements a particular thread maps to and \(\varphi_{jm}^{1}\) as one of the sets of basis functions (same radial parameters, differing angular parameters) to which each thread is mapped. For example, for two s functions (s-s thread type), \(p_{jk}^{1}\) is a single element, while for a p and an s function pair (p-s thread type), \(p_{jk}^{1}\) is three elements. Thus, each thread calculates its own partial set of KS elements:

\[
KS_{jk,mn} = \sum_i q_i \int \frac{\varphi_{jm}(\vec{r})\varphi_{kn}(\vec{r})}{|\vec{r} - \vec{r}_i|} d\vec{r}
\] (5-9)

The evaluation of the terms in the inner sum of Equation 5-10 differs based on the angular momentum type of the two basis functions. Different thread types (s-s, p-s, p-p, and so on) require different code to evaluate their energy term. One way to handle this would be to have a simple branch in the main GPU kernel that has a thread check.
what type it is, and then branch off to evaluate the appropriate code. This can possibly lead to some minor warp divergence, and also does not allow for as much optimization by the compiler. We found the best performance results by building separate kernels for each thread type that are compiled separately; C++ templates and pre-processor directives allow this to be done without increasing the size of the code base. We applied a further optimization to this process using CUDA "streams", which allow for the asynchronous launching of different kernels. Each kernel type is assigned its own stream, and all kernels are launched at the same time, allowing them to be scheduled on the GPU simultaneously. We found this led to a 2-3x speedup in the overall calculation time for test systems (compared to running the separate kernels synchronously).

Another design consideration is the application of the Schwarz inequality. This optimization is more important for the Coulomb terms, but LIO also applies it to the QM/MM terms. The use of the Schwarz inequality effectively eliminates pairs (j,k) of contractions from the total energy calculation (Equation 5-6) that have negligible overlap. This is done by evaluating a prefactor based on the arguments of the exponentials in Equation 5-4; if the resulting overlap is below the set threshold, then the individual energy term (Equation 5-10) does not need to be evaluated for that basis function pair. One way to accomplish this on the GPU would be to launch a thread for every basis function pair, having the GPU thread calculate the prefactor and check if its function pair has sufficient overlap to continue. If it does not, then the thread would exit. Once again, this method can lead to warp divergence issues, depending on the ordering of the density matrix. More importantly, for larger systems, with standard overlap
thresholds, most of the basis function pairs do not meet the Schwarz inequality, and thus many threads are launched that do nothing but quit immediately, while still requiring memory to be allocated on the device ahead of time. We found it more efficient (especially memory efficient) to do the Schwarz inequality check on the CPU before the GPU kernels are launched. Afterwards, the data structures on the GPU are set up for only valid function pairs. This effectively reduces the inner double sum of Equation 5-6 to a single sum over only valid function pairs, corresponding to GPU threads that will actually contribute to the total energy.

The main input data structures needed on the GPU for the QM/MM terms are the function parameters \( k_m^j \), \( \alpha_m \), and \( \vec{r}_j \) (Equation 5-4), and the MM charges \( q_i \) and positions. The main output data structure is the (KS) matrix of energy terms (contracted to use significant function pairs). Each thread calculates the energy contribution of a pair of primitive functions, over the entire overlapping block of the corresponding sub-shells. The final energy terms to go into the KS matrix need to be constructed by summing over threads from the same contracted function. The threads for a particular contraction are not guaranteed to fall in the same warp, or even the same thread block, so care needs to be taken here. We use a parallel reduction algorithm to combine the individual terms efficiently.

**QM/MM – Forces**

We refer to the angular momentum parameters in Equation 5-4 \( (n_x^j, n_y^j, n_z^j) \) by the vector \( \mathbf{n}^j \). Thus, the gradients required to calculate forces are determined using the relationship of (primitive) Cartesian Gaussian functions:

\[
\frac{\partial}{\partial r_i} \varphi_{jm}(r, \mathbf{n}^l) = 2\alpha_m \varphi_{jm}(r, \mathbf{n}^l + \mathbf{1}_i) - n_i^j \varphi_{jm}(r, \mathbf{n}^l - \mathbf{1}_i)
\]
Here, \( \mathbf{1}_i \) is a vector with component \( i \) equal to 1 and every other component 0.

The gradient contribution of the overlap of two s functions requires the evaluation of p-s terms, a p-s gradient requires evaluation of d-s and p-p terms, and so on. As mentioned previously, the QM/MM gradient calculations represent the current major computational bottleneck in LIO. The thread mapping scheme for the calculation of gradients we employ here is essentially identical to that for the energies. Each thread maps to a pair of primitive basis functions. Each thread calculates partial gradients for one or two atoms in the QM region (the function centers) and for every MM atom. Figure 5-4 shows a graphical representation of this mapping. Using an efficient reduction to sum the MM atom forces is particularly important here, as the entire thread block calculates partial forces for each MM atom, which need to be summed together and put into the global array.

**Coulomb**

With the use of an auxiliary density basis set, the calculation of the Coulomb terms is very similar to the QM/MM terms. The evaluation of the individual integrals is more complicated, but the memory access patterns and thread mapping are nearly identical, with the auxiliary basis set replacing the MM charges in function. Therefore, much of the code base for calculating the QM/MM energy terms and forces can be reused in evaluating the Coulomb terms in LIO.

**Results**

**Speedup**

Calculations were performed on two different hardware setups: a desktop with an Intel i7 processor and Nvidia Tesla K20c GPU, and the NCSA Blue Waters system (AMD 6276 "Interlagos" CPUs and Tesla K20X GPUs). All DFT calculations were
performed with LIO and all QM/MM MD calculations were performed with LIO and Amber 14, which were connected with the recent external QM/MM interface in sander used with external electronic structure programs. All calculations use the PBE functional with the DZVP MO basis set and DGA1 auxiliary basis set. We refer to CPU calculations with LIO as those that compute the QM/MM and Coulomb terms on the CPU, though the exchange-correlation terms are still calculated on the GPU. All CPU calculations were performed on a single core.

To analyze the improvement in computation runtime of the QM/MM terms, we look at two reactive protein systems. The first is the soluble domain of CopA from *Archaeoglobus fulgidus* (Figure 5-5A), an ATPase involved in copper transport across cell membranes, with a QM region size of 34 atoms. The second is MshA from *Corynebacterium glutamicum* (Figure 5-5B), which is involved in the pathway of mycothiol synthesis in *Actinobacteria*, with a QM region size of 64 atoms. Table 5-1 gives the number of basis functions, and breakdown by angular momentum type, for each of the systems we look at here.

Table 5-2 shows the speedup in the calculation of QM/MM gradients for CopA and MshA on the K20 desktop. The CopA system had 2954 atoms in the MM region here while the MshA QM region used an electrostatic cutoff (qmcut in sander) of 16 Å resulting in 3885 MM atoms in the calculation. Speedups ranging from 70 to 100 times are seen in the individual force and energy calculations for the QM/MM terms, while the overall SCF and gradient calculations are accelerated by about a factor of 3.

We also examined the speedup in the Coulomb calculations for the CopA and MshA systems, as well as three further systems in vacuum – caffeine (24 atoms), a
modified heme group (48 atoms), and carotene (92 atoms) (Figure 5-6). Table 5-3 shows timings for the Coulomb terms in these five systems. We see speedups in the Coulomb gradients ranging from 50 to 100, while the overall SCF and gradient time is sped up by about a factor of 1.5 in the vacuum systems where there are no QM/MM interactions. The GPU acceleration leads to the runtime for QM/MM and Coulomb terms to be negligible relative to the other parts of the total DFT calculation. Table 5-3 also shows the energy differences (in kcal/mol) found for single point calculations between the CPU and GPU versions. None of the observed differences were significant (all below a tenth of a kcal/mol); this was to be expected as all QM/MM and Coulomb calculations are still performed in full double precision on the GPU.

We used the MshA system further to see how the accelerated QM/MM calculation performs over varying qmcut values, which lead to different numbers of atoms in the MM region. Figure 5-7 shows QM/MM gradient runtime for varying MM region sizes (different qmcut values). The gradient runtime shows a linear trend with MM region size, however if we look at the scale that the runtime increase by (going up to about 1 second at 10000 atoms compared to a total MD step time of about 36 seconds), we can see that the QM/MM force calculations runtimes are now negligible compared to the rest of the QM calculation up to large MM regions.

**MshA Reaction Barrier**

MshA is a glycosyltransferase that is involved in the biosynthesis of mycothiol, a compound found in *Actinobacteria* (for instance, *Mycobacterium tuberculosis*) that serves an important role in protecting the bacterium from the immune system.\(^{117}\) MshA catalyzes the transfer of a sugar moiety, N-acetyl-glucosamine (GlcNAc), from a donor, UDP-GlcNAc, to an acceptor, 1-L-my-o-inositol-1-phosphate (m-Ins-1P).\(^{3}\) This step of
the mycothiol synthesis pathway has been identified as a potential drug target, so understanding this reaction and possible inhibition mechanisms could have future applications in drug design. Studying this reaction with LIO also provides a validation of the GPU porting beyond single point comparisons into real free energy calculations from long MD trajectories generated with the QM/MM and Coulomb forces being computed on the GPU. All calculations in this section were performed on the Blue Waters system.

Figure 5-8 shows the distances used to define the reaction coordinate (RC) in a recent study by our collaborators,118 which describes the progression of the reaction from reactants (around -2 Å in RC) to products (around +2 Å in RC). The reaction as modeled involves two bonds being broken and reformed: a proton transfer from m-Ins-1P to UDP, and the nucleophilic attack of the m-Ins-1P breaking the UDP-GlcNAc bond and forming a new glycosidic linkage to form the product. The RC describes only the second mechanism, and thus in the QM/MM MD simulations, only that bond is forced to break and reform through restraints. The proton transfer is left unrestrained, allowed to proceed naturally during the reaction pathway.

We investigated the reaction through 1-dimensional umbrella sampling. Different windows in the umbrella sampling were defined by different values of the RC. The two distances d1 and d2 were restrained with a force constant of 200 kcal/mol so that the value d1-d2 would equal the current target RC, while the rest of the system was allowed to evolve unrestrained. An initial reaction pathway was generated by equilibrating each window (starting from the product side at +2.56 Å RC) for 0.5 ps, and then beginning the next window (closer to the reactant side) with the structure from the previous window. The structures on the product side of the pathway were stabilized by a hydrogen bond
between GlcNAc and m-Ins-1P. This bond is not favorable at the reactant side of the pathway, and a hydrogen bond between m-Ins-1P and protein residue ASN 25 is preferred, so the extra GlcNAc-m-Ins-1P bond is not seen in structures below -1.5 Å RC. However, this GlcNAc-m-Ins-1P hydrogen bond was not reconnected when generating a pathway from the reactant side to the product side, given the limited sampling we are able to do. Thus, generating the pathway by starting from the product side led to more stability in the transition region. In total, we used 48 windows spaced by 0.1 Å. There were also unfavorable interactions between hydrogens in m-Ins-1P and GlcNAc and nearby protein residues. Thus, we applied restraints to these hydrogens of force constant 800 kcal/mol to their bonded oxygens. An electrostatic cutoff of 16 Å around the QM region was used leading to around 3900 MM atoms (with fluctuations from step to step). All MD simulations were performed with the Langevin thermostat and periodic boundary conditions on the system. Each window was equilibrated for at least another 2 ps, and then the last 4 ps of the trajectories were taken as production data. The weighted histogram analysis method (WHAM) was used to generate the free energy profile from the umbrella trajectories.

Figure 5-9 shows the free energy of the system over the entire reaction pathway. A reactant to transition state barrier of about 17 kcal/mol is observed. This is in good agreement with previous experimental data. Observing the evolution of the distances d3 and d4 also supports the proton transfer occurring late in the reaction.

**Discussion**

We have shown results for the developed GPU ports of the QM/MM and Coulomb calculations in LIO that reveal a substantial speedup in simulation time. All calculations were performed with double precision on the GPU, which led to no
significant numerical errors compared to the CPU version. In addition, we have shown
the new GPU ports of LIO used to make a real calculation with the MshA system, with
results in agreement with experiment. In closing, we have a few remarks regarding
future directions with this work, and a small example of the need for full DFT in high
quality QM/MM simulations.

Our collaborators originally performed the 1-dimensional umbrella sampling of
the MshA system using the semi-empirical DFTB method.\textsuperscript{118} As a result, they found
reaction barriers ranging from 20 to 45 kcal/mol. It is evident that, in this case at least, a
semi-empirical method was not sufficient to properly model the reaction under study.
However, the time it would have taken to perform the entire umbrella sampling with the
version of LIO before the GPU porting of the QM/MM and Coulomb terms would have
been untenable, or at least would have further limited the amount of sampling that could
be done. At 0.5 ps per window, with 48 windows, at least 24 ps of simulation time were
needed to prepare the reaction pathway before further equilibration and production were
performed (the production can be done completely in parallel across all windows,
especially on a large system like Blue Waters). At 3-4 ps/day, the current version of LIO
takes a little under a week of wall time to generate the initial pathway. The previously
reported version of LIO would have taken more than a month just to generate the
pathway. As we can see, these kinds of acceleration efforts allow systems to be
studied with higher levels of theory with sufficient sampling in a reasonable amount of
real time.

Currently, the GPU port of LIO supports only running on a single GPU. Our
collaborators have begun expanding the exchange-correlation GPU port to allow the
use of multiple GPUs at once, to further accelerate the calculation. This type of multi-GPU approach can also be used in the QM/MM and Coulomb terms, which is an interesting avenue for future work. These terms can be easily parallelized, as each thread block currently acts independently, and only an extra reduction step would be needed to combine results across GPUs.
Figure 5-1. SIMT architecture and CUDA kernel execution model. Threads are specified by the user in a grid of blocks. On the SMs, blocks are split up into warps to be executed concurrently.

Figure 5-2. Relative computation times from each part of LIO runtime for single-core optimized CPU version and GPU accelerated version. Taken from Figure 9 of Reference 23.
Figure 5-3. Example of implemented thread mapping for QM/MM and Coulomb Kohn-Sham matrix evaluation. Three example threads are shown: thread 1, for example, calculates part of the matrix element for an overlap of two s type functions. The Kohn-Sham matrix shown has five basis functions, three s type and two p type.

Figure 5-4. Example of implemented thread mapping for QM/MM force evaluation.
Figure 5-5. Protein systems studied. A) CopA from *Archaeoglobus fulgidus*. B) MshA from *Corynebacterium glutamicum*.
Figure 5-6. Systems studied in vacuum. A) Caffeine, B) heme group / carbon monoxide complex with no lateral chains, and C) carotene.

Figure 5-7. Scaling analysis of QM/MM gradient runtime versus MM region size.
Figure 5-8. Defined distances between the UDP-GlcNAc donor (right) and m-Ins-1P acceptor (left) in the MshA-catalyzed reaction. The reaction coordinate used is defined as $d_1 - d_2$.

Figure 5-9. Free energy profile and distances over the course of the reaction pathway in the MshA-catalyzed reaction.
Table 5-1. Number of basis functions for each studied system for the DZVP basis set, along with number of each angular momentum type. Total in this case means total number of contracted Gaussian functions, while the count for each angular momentum type gives the number of sub-shells (e.g., each p function counted represents three contracted Gaussian functions, and so on).

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>s</th>
<th>p</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caffeine</td>
<td>230</td>
<td>62</td>
<td>28</td>
<td>14</td>
</tr>
<tr>
<td>CopA</td>
<td>364</td>
<td>106</td>
<td>46</td>
<td>20</td>
</tr>
<tr>
<td>Heme</td>
<td>523</td>
<td>130</td>
<td>65</td>
<td>33</td>
</tr>
<tr>
<td>MshA</td>
<td>591</td>
<td>165</td>
<td>72</td>
<td>35</td>
</tr>
<tr>
<td>Carotene</td>
<td>756</td>
<td>276</td>
<td>80</td>
<td>40</td>
</tr>
</tbody>
</table>

Table 5-2. Runtimes and speedups for CopA and MshA. The total times given are the time to reach SCF convergence and calculate all gradients to be used for force computation for the given structure. Note that total speedups include effect of QM/MM and Coulomb acceleration. All times in seconds.

<table>
<thead>
<tr>
<th></th>
<th>CopA</th>
<th>MshA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
<td>QM/MM energy</td>
</tr>
<tr>
<td>CPU</td>
<td>37.66</td>
<td>3.22</td>
</tr>
<tr>
<td>GPU</td>
<td>14.14</td>
<td>0.04</td>
</tr>
<tr>
<td>Speedup</td>
<td>x 2.7</td>
<td>x 73.1</td>
</tr>
</tbody>
</table>
Table 5-3. Speedups in the Coulomb gradients for the five studied systems, and energy difference between CPU and GPU single point calculations. Note that total speedups for CopA and MshA include effect of QM/MM and Coulomb acceleration, while other systems include only the effect of Coulomb acceleration. Also note that the GPU test for carotene resulted in significantly fewer SCF iterations to reach convergence compared to the CPU test, thus the discrepancy between total time saved and time saved from the Coulomb calculation. All times in seconds and energy differences in kcal/mol.

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>GPU</th>
<th>Speedup</th>
<th>SP Energy Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coulomb gradient</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Caffeine</td>
<td>2.67</td>
<td>0.06</td>
<td>x 48.5</td>
<td>0.006</td>
</tr>
<tr>
<td>Total</td>
<td>10.43</td>
<td>7.16</td>
<td>x 1.5</td>
<td></td>
</tr>
<tr>
<td>CopA</td>
<td>6.08</td>
<td>0.10</td>
<td>x 63.3</td>
<td>0.025</td>
</tr>
<tr>
<td>Total</td>
<td>37.66</td>
<td>14.14</td>
<td>x 2.7</td>
<td></td>
</tr>
<tr>
<td>Heme</td>
<td>22.55</td>
<td>0.19</td>
<td>x 116.9</td>
<td>0.031</td>
</tr>
<tr>
<td>Total</td>
<td>109.3</td>
<td>75.21</td>
<td>x 1.5</td>
<td></td>
</tr>
<tr>
<td>MshA</td>
<td>21.34</td>
<td>0.22</td>
<td>x 98.8</td>
<td>0.075</td>
</tr>
<tr>
<td>Total</td>
<td>108.93</td>
<td>36.97</td>
<td>x 2.9</td>
<td></td>
</tr>
<tr>
<td>Carotene</td>
<td>27.21</td>
<td>0.28</td>
<td>x 97.5</td>
<td>0.014</td>
</tr>
<tr>
<td>Total</td>
<td>178.70</td>
<td>108.69</td>
<td>x 1.6</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 6
CONCLUSIONS

In this study, we have examined several methods of optimizing techniques within the field of computational chemistry. We have looked at an efficient computational method to fit the dihedral parameters of a force field, and simple ways to minimize the number of expensive QM calculations needed to do these fits. We have examined a simple way to speed up MD simulations by a factor of two. Finally, we have looked at the use of GPUs to speed up DFT calculations allowing the use of full QM in QM/MM studies of enzymes.

Two main problems face the researcher using MD with MM as a tool to study atomic level dynamics of large biomolecular systems; these are typically called the “force field problem” and the “sampling problem.” The force field problem refers to inaccuracies in the coarse molecular mechanics level of theory used in classical force fields. Modern force fields are constantly being optimized to more closely represent physical reality. In order for the coarse MM model to have any validity, high quality parameters derived through a rigorous process and validated against high level QM and experimental data are necessary. However, validation stretches far beyond the initial release; as new classes of systems are studied, and as hardware and algorithmic improvements allow the treatment of larger models with QM calculations, force fields need to frequently be updated to best reflect the latest knowledge in the field. Many of the challenges faced in force field parameterization do not have a clear, exact solution, and so there exists a multitude of techniques to approximate one. Often, a targeted physical property does not have a closed functional form in terms of the force field parameters, or the functional form is truly non-linear (e.g., vibrational spectra,
geometries, bonds and angles). In these cases, non-linear fitting techniques or heuristic optimizations are needed in order to arrive at a parameter set that best fits the target data. In Chapters 2 and 3 we have shown that the usual method of deriving the dihedral parameters, through a least squares fit to conformational energies with a truncated Fourier series, is somewhat of a special case in that it does have an “easy”, exact solution, and that a simple sampling method combined with this exact solution can more quickly lead to higher quality force field parameters.

The “sampling problem” refers to the infeasibility of running simulations long enough to sufficiently sample long-timescale (i.e., much longer than microseconds) events that form the basis of many important biological processes. This problem is constantly being chipped away at with incremental improvements in hardware, and this, combined with algorithmic improvements and increased use of parallelism (e.g., GPUs) continues to unlock routine access to longer timescales for study with MD. We introduced one such algorithmic improvement, HMR, in Chapter 4.

When turning to higher levels of physical theory in atomistic simulations, the force field problem is moot, but the sampling problem becomes many orders of magnitude worse. Semi-empirical methods like DFTB and AM1 somewhat bridge the gap, easing the sampling problem, but as mentioned these methods are not an appropriate model for some systems and some processes. QM/MM allows the application of high levels of theory within a larger molecular system, but the QM computation still serves as a bottleneck here. Thus, improving sampling with high levels of physical theory, as we investigated in Chapter 5, is crucial to the improvement of atomistic simulations. Not only does this apply to the direct study of molecular
properties with DFT or *ab initio* theories, but it also has the potential to improve the quality of force fields used in MM macromolecular studies, as the production of high level QM data plays an important role in developing high quality force fields, while acting as a computational bottleneck in that area as well.
LIST OF REFERENCES


(26) Ufimtsev, I. S.; Martínez, T. J. **2008**.


(120) Ferreria, M.; Darago, J. P.; González Lebrero, M. C.; Mocskos, E. E. *Submitt. to Elsevier* **2016**.
BIOGRAPHICAL SKETCH

Chad Hopkins was born at Cape Canaveral Hospital in Cocoa Beach, Florida. He attended elementary school in Melbourne, Florida. During the 4th grade, he moved with his parents to the Florida Keys so that his father could open his own business in Key West, The Bed and Chair Depot. He spent the rest of his school years through high school in the Keys, attending Key West High School, where he met his future wife, Lauren. He moved back to Melbourne to attend the Florida Institute of Technology, majoring in computer science. During his first year, he discovered a passion for science, particularly physics, which led him to add a second major in physics. He graduated in the fall of 2009. After some time as a software engineer in Melbourne, he was accepted to graduate school at the University of Florida. Chad and Lauren moved to Gainesville in the fall of 2010, and got married in the summer of 2011. Chad received his Ph.D. in physics in the spring of 2017.