University of Southampton

Faculty of Engineering and Physical Sciences

School of Chemistry

Enhancing conformational sampling by modifying the underlying velocity distribution: Digitally filtered Hybrid Monte Carlo

by

Can Simon Pervane

A thesis submitted for the degree of

Doctor of Philosophy

February, 2019

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
School of Chemistry

Doctor of Philosophy

Enhancing conformational sampling by modifying the underlying velocity distribution: Digitally filtered Hybrid Monte Carlo

by Can Simon Pervane

The rugged energy landscape of proteins often leads conventional molecular dynamics simulations to get stuck in local minima leading to inefficient sampling and slow convergence. Two main methods exist to improve sampling. First, by smoothing the underlying energy surface thereby encouraging barrier crossing, and second, by modifying the simulation velocities. There are many examples in the literature of the former, but velocity modification is a comparatively less well exploited approach. In a method previously developed in our group called Reversible Digitally Filtered Molecular Dynamics (RDFMD), we have been able to induce conformational changes in proteins by amplifying low-frequency molecular vibrations via application of a digital filter to the velocity set [1]. However, the application of a digital filter to the molecules, disrupts the equilibrium of the system, leading to incorrect ensemble averages. To sample a statistical ensemble and still benefit from the application of the digital filter a novel method called Digitally Filtered Hybrid Monte Carlo (DFHMC) is proposed. This method builds on the work of Momentum Enhanced Hybrid Monte Carlo (MEHMC) [2]. Low frequency motions are selectively enhanced via application of a specifically designed Digital Filter (as in RDFMD), but equilibrium is maintained using a Hybrid Monte Carlo approach.

In this thesis, the theory behind the DFHMC and the integration of the method into several optimised molecular dynamics packages is explained. The capability of sampling from the canonical ensemble and the enhancement in energy barrier crossing, and hence convergence, is demonstrated on a system with double well potential and with a curved transition path. As a model for the dihedral motions in proteins, the application of the DFHMC method to a single alanine dipeptide molecule in explicit and implicit water is discussed. We observed a reduction in sampling efficiency in DFHMC when the molecule has centre of mass rotation. Ideas on how to potentially overcome the rotation problem is discussed. And the Riemannian Manifold Hamiltonian Monte Carlo method [3] is introduced to investigate the application of the method to molecular systems.

Contents

List of Figures

List of Tables

List of Algorithms

Declaration of Authorship

I, Can Simon Pervane, declare that this thesis entitled “Enhancing conformational sampling by modifying the underlying velocity distribution: Digitally filtered Hybrid Monte Carlo” and the work presented in it are my own and has been generated by me as the result of my own original research. I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. None of this work has been published before submission.

Signed:

Date: 27 February 2019

Acknowledgments

I would like to express my very great appreciation to Prof. Jonathan W. Essex, my research supervisor, for his patient guidance, enthusiastic encouragement and useful critics of this research work.

My sincere thanks also go to Assoc. Prof. Michael R. Shirts offering me the visiting research opportunity in his group and leading me working on an exciting project to extend Gromacs to integrate HMC based algorithms.

I would also like to thank my labmates especially Khaled Maksoud for his motivating and inspiring theoretical discussions.

My grateful thanks are also extended to Domagoj Fijan for his help and discussion on the integration of the HMC and DFHMC methods into Gromacs and Plumed platform.

I wish to thank Dr. David Cortés Ortuño and Dr. Oliver Laslett for their friendship, support and stimulating discussions.

I would like to thank my family, whose love and guidance are with me in whatever I pursue. Most importantly, I wish to thank my loving and supportive wife Özge Pervane, who provide unending inspiration.

I would like to acknowledge the Institute for Complex Systems Simulation for creating a diverse and multidisciplinary research environment and providing financial support through the EPSRC Doctoral Training Centre grant (EP/G03690X/1).

ICSS

Nomenclature

NomGroup 1

$< A >$ Ensemble average of observable A

β inverse temprature

γ friction coefficient

G Mass Matrix in Riemannian Manifold, it is dependent on positions

M Mass Matrix

p Momentum Vector

q Position Vector

v Velocity Vector

dt timestep used in simulations

F Force

H Hamiltonian

K Kinetic Energy

k_B Boltzmann constant

N Number of atoms

T temperature, in Kelvin

U Potential Energy

Abbreviations

DF Digital Filter

DFHMC Digitally Filtered Hybrid Monte Carlo

fs femtosecond

HMC Hamiltonian/Hybrid Monte Carlo

MC Monte Carlo

MD Molecular Dynamics

MEHMC Momentum Enhanced Hybrid Monte Carlo

REMD Replica Exchange Molecular Dynamics

RMHMC Riemannian Manifold Hybrid Monte Carlo

Introduction

1 Motivation

Thermodynamical quantities of a molecular system obtained via experimental observations can be calculated from the atom's positions and velocities theoretically via two types of averages. The first is the time average which can be obtained by dynamical simulations such as Molecular Dynamics (MD). MD outputs a trajectory of the molecule's atomic velocities and coordinates; thus, it gives a time evolution of the molecule. The second type of average is the ensemble average which consists of a collection of atomic positions and velocities that follow a specific stationary distribution. The ensemble does not contain the information of time. To relate back the experimental measurement to the aforementioned averages, one can think that an experimental measurement is done on a concentration of molecules for some period of time. And a snapshot of that time evolution would show lots of frozen molecules having various atomic positions and velocities. On the other hand, if we could follow one molecule over that measurement time and take a snapshot at regular time intervals, we would again have a collection of various atomic positions and velocities. Thus, what we measure from the system is actually an average over time and/or also an average overall the number of molecules given by the concentration. In thermodynamical equilibrium, if the system has the ergodic property then the time average and the ensemble average will be the same. And therefore, the experimental measurement can be calculated via a time average such as with MD or an ensemble average.

Even though a system is ergodic in theory, it may take relatively long times for the time average to converge to the ensemble average. For biomolecules especially proteins, the energy landscape is very rugged. Thus via conventional MD simulations, the molecule under study tends to get stuck in a local minimum for long simulation times, which slows down the convergence of the time average. When the energy barriers in the rugged energy landscape are much bigger than the molecule's kinetic energy then the molecule gets stuck in a local minimum for so long that at some point it becomes unfeasible to sample the required phase space in the time period that we can reach with the current computer technologies. The simulation is bound to a small step size determined by the fastest periodic motion in the system, which is usually the hydrogen

bond vibration and is in the order of 10 fs [femtoseconds(fs) $=10^{-15}$ second(s)]. On the other hand, the motions of interest could be on the order of several nanoseconds, milliseconds, seconds or even longer. From a computational perspective, this means an increasing amount of iterations to be done to reach the desired time scale. In an MD simulation at each iteration, the force acting on each atom due to all the other atoms in the molecule needs to be evaluated. The evaluation time thus increases with bigger molecules having more atoms. Therefore, the simulation is not only bound by the total time to be simulated but also by the size of the system. Thus to overcome this computational limit or at least alleviate it, several enhanced sampling methods are developed.

The enhanced sampling methods may broadly be divided into methods where the underlying potential energy function is modified and in methods where the atomic velocities are explicitly modified. Some examples for methods that modify the underlying potential energy include Metadynamics, in which during the simulation the energy surface is filled with Gaussians preventing a configuration to be visited twice [4] and steered molecular dynamics (SMD) [5] where external forces are applied to the system to guide the molecule through the conformational space. Some examples of methods that modify explicitly the velocities includes simulated annealing [6] in which the temperature of the molecule is increased to overcome the energy barrier, Replica-Exchange molecular dynamics [7] which employs a set of replicated parallel simulations at different temperatures that exchanges at certain steps, Reversible Digital Filtering Molecular Dynamics (RDFMD) [1] which uses a digital filter to selectively enhance or suppress vibrational motions and Momentum Enhanced Hybrid Monte Carlo (MEHMC) [2], which uses a simple low pass filter to suppress high frequency motions and enhance low frequency motions. Although all of the above methods enhance the sampling of the molecular systems, not all preserve the equilibrium distribution and allowing to take an ensemble average.

In this study, the method to selectively enhance or suppress molecular vibrational modes presented in RDFMD is adopted, and this nonequilibrium method is extended or reformulated such that it samples a canonical ensemble, where ensemble averages and thermodynamical quantities can be calculated. To incorporate the filter application into a sampling method that generates equilibrium ensembles, a Hybrid Monte Carlo approach with a biased velocity distribution as used in the MEHMC method [2] is developed. The method is called Digitally Filtered Hybrid Monte Carlo (DFHMC) and is a multiphysics method merging Digital filtering Molecular Dynamics and Monte Carlo simulation techniques. Before applying the DFHMC method on proteins and biomolecules in general, the velocity biased HMC formulation is implemented and tested for fixed velocity biases on a simple 2D system to ensure biasing the velocities preserve the ensemble distribution. During this study, two possible biased velocity distributions that have similar acceptance rates are found to preserve the ensemble

distribution. Furthermore, the biased velocities obtained by filter application with regular updating of the bias values showed an enhanced sampling compared to the Hybrid Monte Carlo method [8] which essentially corresponds to having bias values of zero for all times.

To enable the DFHMC method to be used on Molecular systems efficiently, the HMC module including the DFHMC method is integrated into optimised and widely used MD packages. In this scope, a design developed previously in our lab that can pass and receive information from an MD package is extended to enable the usage of HMC based methods. However, the design was very slow, and the majority of computation was spent in writing and reading files from hard disk to communicate data between the MD engine and the HMC module. Therefore, another design that uses the package Plumed [9], which gives a much more efficient interface to control the information passed and received from the MD module is developed. The Plumed code was extended to include the functionality to receive and pass the velocities of the system from and to the MD engine, this was developed specifically to one MD engine called Gromacs [10]. Even though the implementation is computationally more efficient, the changes required makes the design and any new method development highly dependent on different MD engines. Therefore, the final version of the method was integrated into an MD engine called OpenMM [11], which allows to develop methods more independently from the lower abstraction layers of the MD engine and still provides a computationally efficient implementation.

Using the developed implementation of DFHMC and HMC via the OpenMM package, several systems are investigated. In particular, a double well potential system where the energy barrier is along a cartesian coordinate, and a rotated double well potential such that the energy barrier is along a linear combination of the cartesian coordinates are constructed and studied. A potential energy surface with a curved transition path is generated, and the enhancement properties of the DFHMC method is investigated. The DFHMC method is then applied to alanine dipeptide, as a model of the dihedral motions in proteins, and analysed under explicit and implicit water environment. We observe that the rotation of the molecule has a negative effect to the DFHMC method's sampling efficiency by comparing the results of alanine dipeptide in implicit water with and without center of mass rotation removed at each HMC step. Several possible concepts are suggested to tackle the drop in efficiency in the sampling caused by the molecule rotation. And one particular sampling method called Riemannian Manifold Hamiltonian Monte Carlo [3] developed for posterior sampling in Bayesian inference problems is introduced and the potential for application towards molecular systems is discussed.

2 Organization

The thesis is organised into five sections. Starting with the theory section to build the foundations for the DFHMC method. The theory section includes a brief introduction of the notion of an ensemble in statistical mechanics. It continues by introducing the Molecular Dynamics (MD) and Metropolis Monte Carlo (MC) methods as a way to generate molecular configurations that follow a stationary distribution. Thereafter signal processing methods such as Fourier Transform, Wavelet Transform are explained, which are used to analyse the frequency domain properties of simulation trajectories. Digital Filtering (DF) are explained, as a tool to be used to enhance sampling. Subsequently, enhanced sampling methods are introduced under two main categories, i.e., methods that change the potential or the kinetic energy of the system. Meta-Dynamics and Umbrella Sampling are discussed as examples of the former. Replica Exchange Molecular Dynamics and advanced methods that combine MD, MC and DF to achieve enhanced sampling are introduced as examples of the latter category. In this scope, the RDFMD and DFMD methods which combine MD with DF, the Hybrid Monte Carlo which combines MD with MC is explained. It follows by studying the method developed in this thesis, DFHMC which combines all three methods DF, MD, MC. Last, in this section a sampling method called Riemannian Manifold Hamiltonian Monte Carlo (RMHMC) that is used in the context of Bayesian inference is brought to attention to investigate the application of the method and/or the framework to be applied to molecular systems.

The next section focuses on the effect of the bimodal velocity distribution used in the DFHMC method to the simulation trajectory of the system. More specifically, it is investigated if certain frequencies in the system can be enhanced via the bimodal velocity distribution.

It is continued with the application section which introduces and presents the results of double well potential system, rotated double well system and a system with a curved transition path. It is followed by alanine dipeptide, where simulations in explicit and implicit water is carried out. Next, the effect of molecular rotation is analysed by studying alanine dipeptide in implicit water with and without center of mass rotation. Next, the RMHMC method is applied to a Bayesian logistic regression problem, and the applicability of the method to multimodal energy surfaces is studied.

The software integration of the HMC module with the MD packages are discussed in the code integration section. The evolution of the methods integration and the challenges encountered are explained, and the design of the final integration is presented.

Finally, the results are summarised in the Conclusion section. Suggestions for future work to build on this thesis is also given.

Theory

Chapter 1

1.1 Statistical Mechanics

The key element in statistical mechanics is the notion of the ensemble. An ensemble can be defined as the collection of accessible microstates subject to constraints specified by the system. The constraints of the system are usually given in terms of the macrostates of the system.
The microstate of a molecular system with N atoms is defined with the atomic position vector, $\boldsymbol{q} = (q_1, q_2, ..., q_{3N})$ and momentum vector, $\boldsymbol{p} = (p_1, p_2, ..., p_{3N})$
If the container volume where the molecules are held is constant and the number of atoms, N is fixed Then with the additional given constraint,

$$H(\boldsymbol{q}, \boldsymbol{v}) = E \tag{1.1}$$

where H denotes the Hamiltonian of the system and E is a constant, the ensemble is defined as,

$$\Omega(N, V, E) = \{(\boldsymbol{q}, \boldsymbol{p}) \ : \ H(\boldsymbol{q}, \boldsymbol{p}) = E\} \tag{1.2}$$

$|\Omega(N, V, E)|$ defines the total number of microstate that satisfies the constraint in Eq-1.1 with constant volume, V and number of atoms, N

Assuming that all the accessible microstates with the same total energy are equally likely then the probability of a microstate is equal to:

$$P(\boldsymbol{q}, \boldsymbol{p}) = \frac{1}{|\Omega(N, V, E)|} \tag{1.3}$$

The ensemble with the above probability, Eq-1.3 is also called the micro canonical ensemble [12].

Another example of constraint is where the systems temperature is constant T, together with constant volume, V and fixed number of atoms N.

It turns out that, under this constraint with the same assumption that all the accessible microstates with the same total energy are equally likely, the probability of finding

a microstate is equal to:

$$P(\boldsymbol{q},\boldsymbol{p}) = \frac{e^{-\beta H(\boldsymbol{q},\boldsymbol{p})}}{Z} \tag{1.4}$$

$$Z(\boldsymbol{q},\boldsymbol{p}) = \int e^{-\beta H(\boldsymbol{q},\boldsymbol{p})} d\boldsymbol{q} d\boldsymbol{p} \tag{1.5}$$

where Z is called the canonical partition function. And the ensemble is called the canonical ensemble [12].

The Hamiltonian of a molecular system can be written as the total energy, which is the sum of kinetic energy, $K(\boldsymbol{p})$ and potential energy, $U(\boldsymbol{q})$

$$H(\boldsymbol{q},\boldsymbol{p}) = K(\boldsymbol{p}) + U(\boldsymbol{q}) \tag{1.6}$$

Thus Eq-1.5 can be written as,

$$Z(\boldsymbol{q},\boldsymbol{p}) = \int e^{-\beta K(\boldsymbol{p})} d\boldsymbol{p} \int e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q} \tag{1.7}$$

A similar ensemble to the canonical ensemble is the iso-kinetic ensemble which can be obtained by having constant kinetic energy, together with constant volume and fixed number of atoms. The partition function for the iso-kinetic ensemble is as follows [13]

$$\begin{aligned} Z(\boldsymbol{q},\boldsymbol{p}) &= \int \delta(K(\boldsymbol{p}) - K_0) e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q} d\boldsymbol{p} \\ &= \int \delta(K(\boldsymbol{p}) - K_0) d\boldsymbol{p} \int e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q} \end{aligned} \tag{1.8}$$

where K_0 is a constant representing the kinetic energy. $\delta(K(\boldsymbol{p}) - K_0)$ is a delta function which evaluates to one when $K(\boldsymbol{p}) = K_0$ and zero otherwise. As with the canonical ensemble, the iso-kinetic ensemble has a separable partition function in terms of the positions and momentum. Comparing Eq-1.7 and Eq-1.8, it can be seen that the position part of the partition function is the same for iso-kinetic and canonical ensemble. However, the difference in the momentum part of the partition function in the canonical ensemble results in the kinetic energy thus the temperature to fluctuate, whereas in the iso-kinetic ensemble it is strictly constant [14].

In this thesis, the ensemble of interest is the canonical ensemble. The methods used and developed in this thesis simulate the system in the canonical ensemble.

Similar to separating the partition function the probability distribution of the canonical ensemble, Eq-1.4 can be written as,

$$\begin{aligned} P(\boldsymbol{q},\boldsymbol{p}) &= \frac{e^{-\beta K(\boldsymbol{p})}}{\int e^{-\beta K(\boldsymbol{p})} d\boldsymbol{p}} \frac{e^{-\beta U(\boldsymbol{q})}}{\int e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q}} \\ &= P(\boldsymbol{q}) P(\boldsymbol{p}) \end{aligned} \tag{1.9}$$

Using the probability distribution of the microstates one can construct averages that

represent thermodynamic quantities or observables, such as internal energy, temperature, pressure etc.
In general, the average of a quantity, $A(\boldsymbol{q}, \boldsymbol{p})$ that is a function of microstates can be expressed as an integral over the microstates as:

$$< A >= \int A(\boldsymbol{q}, \boldsymbol{p}) P(\boldsymbol{q}, \boldsymbol{p}) d\boldsymbol{q} d\boldsymbol{p} \tag{1.10}$$

However, often the average quantity of interest is a function of the atoms positions only, $A(\boldsymbol{q}, \boldsymbol{p}) = A(\boldsymbol{q})$. Hence the average in Eq-1.10, will contain integral over the positions only. The integral over momentum will integrate to one.

$$< A >= \int A(\boldsymbol{q}) P(\boldsymbol{q}) d\boldsymbol{q} \tag{1.11}$$

Additionally, the integral Z over the momentum can be calculated analytically. Thus, the partition function Z can be written in terms of the positions only. Therefore, even though an average quantity is a function of both positions and momentum, $A(\boldsymbol{q}, \boldsymbol{p})$ as long as it can be written in terms of the partition function, Z, the average can be obtained by knowing only the configurational ensemble. Thus, to get thermodynamic information out of the system from the microstates, the integral in Eq-1.10 or the partition function, Z in Eq-1.5 needs to be evaluated. An analytical solution to those integrals is restricted to a small number of systems. Therefore, various numerical methods are used to attempt to solve or approximate the integral for more complex systems with relatively high number of degrees of freedom. Often to solve the integral numerically, the first step is to be able to evaluate the probabilities of each microstate. However, the number of microstates grows rapidly with the degrees of freedom and a brute force approach where we would evaluate the probability of each microstate becomes very computationally expensive and infeasible.

The numerical methods discussed in this report, tackle this dimensionality problem by finding ways to generate a set of microstates from the relevant ensemble such that the number of microstates in the set is small enough so that the evaluation times are feasible and big enough to obtain more or less an accurate value for the probability of the microstates. In other words, the numerical methods will generate a subset of the complete microstate ensemble with the correct probability weights, such that the averages can be calculated with much fewer microstates.

1.2 Molecular Dynamics

Molecular Dynamics(MD) is a simulation method that generates not only points in phase space but also the time evolution or the trajectory of the system in phase space, via classical mechanics [15]. The method uses Newton's second law of motion to propagate the molecular system with respect to time through phase space.

Let $\boldsymbol{q}(t) = (q_1(t), q_2(t), ..., q_{3N}(t))$ and $\boldsymbol{p}(t) = (p_1(t), p_2(t), ..., p_{3N}(t))$ be the positions and momentum at time t, of a molecular system with N atoms respectively. Then the time evolution of the position vector can be written as follows:

$$m_i \frac{d^2 q_i}{dt^2} = F_i(\boldsymbol{q}) \quad i = 1, 2, ..., 3N \tag{1.12}$$

where the Force on atom i is related to the derivative of the systems Potential Energy with respect to the i^{th} atoms position.

$$F_i(\boldsymbol{q}) = -\frac{dU(\boldsymbol{q})}{dq_i} \tag{1.13}$$

and the momentum vector, $\boldsymbol{p}$ is defined as:

$$\frac{p_i}{m_i} = \frac{dq_i}{dt} \quad i = 1, 2, ..., 3N \tag{1.14}$$

Therefore Eq-1.12 can also be written as:

$$\frac{dp_i}{dt} = F_i(\boldsymbol{q}) \quad i = 1, 2, ..., 3N \tag{1.15}$$

As can be seen for a molecular system with N atoms, in total there will be $6N$ first order differential equations. In order to solve them the potential energy function of the system needs to be known.

The equations of motions from the Newton's classical mechanics can be reformulated within the Hamiltonian mechanics framework via the following Hamilton's equations Eq-1.16.

$$\begin{aligned} \frac{d\boldsymbol{p}}{dt} &= -\frac{\partial H}{\partial \boldsymbol{q}} \\ \frac{d\boldsymbol{q}}{dt} &= \frac{\partial H}{\partial \boldsymbol{p}} \end{aligned} \tag{1.16}$$

The Hamiltonian mechanics formulation makes it easier to investigate more general Hamiltonian functions.

The Newton equations can be obtained by letting the Hamiltonian be equal to the sum of the potential and kinetic energies of the system, where the kinetic and potential energies are a function of the momentum and positions respectively Eq-1.17.

$$H(\boldsymbol{q}, \boldsymbol{p}) = K(\boldsymbol{p}) + U(\boldsymbol{q}) \tag{1.17}$$

This type of a Hamiltonian is separable in momentum and position. And the standard molecular dynamics algorithms use this type of a Hamiltonian to calculate the trajectory of molecules. This is because the standard methods formulate the mechanics in cartesian coordinates.

However, if we would formulate the dynamics in the internal coordinates system,

the Hamiltonian would not be separable in momentum and positions. Even though the potential energy would only depend on the positions, the Kinetic energy would be a function of momentum and positions.

Similarly, a non-separable Hamiltonian is obtained, when the Mass matrix is made dependent on the positions of the system. This type of dependency has been used in the Riemannian Manifold Hamiltonian Monte Carlo sampling algorithm [3].

The algorithms to solve Hamilton's equations are called integrator algorithms, different types of algorithms are needed based on whether the Hamiltonian is separable in momentum and position or not. Commonly used integrator algorithms for separable and non-separable Hamiltonians are explained in the integrators section.

In Molecular Dynamics, the general form of the potential energy function with its parameters corresponding to different atoms or molecular systems is stored in files called Force-Fields. These are used in numerical simulations to obtain the force acting on an atom i, to numerically evaluate the corresponding differential equations mentioned above.

Furthermore, the total energy is conserved for a molecular system, where the force is given by Eq-1.13; thus the ensemble generated for this kind of system will be a micro-canonical one. However, quite often it is desirable to simulate the molecular system in different ensembles where some other quantities such as temperature, pressure, etc. are held fixed. Controlling the temperature and/or the pressure are generally the most used ones. To extend the MD to cover the temperature and/or pressure control thermostat and/or barostat algorithms are used respectively [16].

In this study, the simulations are done in an NVT / canonical ensemble, where the temperature of the system is conserved. In particular to control the temperature Langevin dynamics is used. Thus in the following subsection, the Langevin dynamics method is explained. An extensive review on thermostat algorithms can be found in [17]

1.2.1 Thermostats

Methods to control the temperature of the system can be categorized into temperature relaxation by stochastic dynamics [18], [19], [20], relaxation by stochastic coupling [16], constraining [21], [22], relaxation by weak coupling [23], relaxation by extended-system method [24], [25], [26]

In this study, we used temperature relaxation by stochastic dynamics, in particular, Langevin dynamics to achieve an NVT ensemble. Following, the Langevin Dynamics method is discussed

Langevin Dynamics

The temperature of a system is directly related to the velocities of the particles.

$$T = \frac{m < v^2 >}{3k_B} \tag{1.18}$$

Where, k_B is the Boltzmann constant, T is the absolute temperature, $< v^2 >$ the average of the squared particle velocities and m is the mass of the particles.

Thus any thermostat algorithm imposes control on the particle velocities. In Langevin dynamics the atomic velocities are controlled by modifying Newton's equation of motion Eq-1.12 as given in the below equation [18].

$$m_i \frac{d^2 q_i}{dt^2} = F_i(\boldsymbol{q}) - m_i \gamma \frac{dq_i}{dt} + R_i \quad i = 1, 2, ..., 3N \tag{1.19}$$

where γ is the friction coefficient, and R_i is the stochastic force.

The stochastic force in the Langevin dynamics implementation used in this study has the following form,

$$R_i = \sqrt{2 m_i \gamma k_B T \, dt}\epsilon \quad \epsilon \sim N(0, 1) \tag{1.20}$$

where dt is the integration time step, k_B is the Boltzmann constant, m_i is the particle's mass, γ is the friction coefficient, T is the simulation temperature and ϵ is an independent and identically distributed random variable (i.i.d) following a Gaussian distribution with zero mean and variance of one.

The trajectory generated by the Langevin dynamics follows a canonical distribution.

1.2.2 Force Fields

Force Fields in molecular dynamics define the interaction potentials between atoms, atomic groups and molecules. In general, the potential energy consists of two terms, bonded and non-bonded potentials. The bonded energy term is made up of atomic bonds, angles and dihedrals. The non-bonded term represents potentials due to electrostatic interactions, the van der Waals interactions and a repulsive force as a result of the Pauli-exclusion principle. The latter two potentials are often modeled by the Lennard-Jones(LJ) Potential. Thus the potential can be written as follows:

$$\begin{aligned} U &= U_{Bonded} + U_{Non-Bonded} \\ &= U_{Bond} + U_{Angle} + U_{Dihedral} + U_{Electrostatic} + U_{LJ} \end{aligned} \tag{1.21}$$

The following potential energy terms are written in SI units.

The atomic bond and angle potential terms can be approximated by a harmonic spring potential, which describes the change of bond length and atomic angles as oscillatory motions. The atomic bond potential term as a function of the bond length vector, $\boldsymbol{l} = (l_1, l_2, ..., l_N)$ where $N = \#Bonds$ is as follows:

$$U_{Bond}(\boldsymbol{l}) = \sum_{i=1}^{\#Bonds} \frac{1}{2} k_i^b (l_i - l_i^0)^2 \tag{1.22}$$

Where the summation is over all the bonds in the system, k_i^b is the spring constant

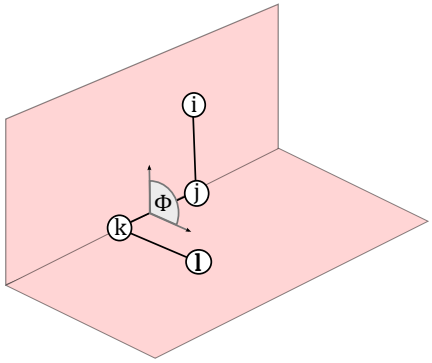
and l_i^0 is the equilibrium bond length of the i^{th} bond.

The atomic angle potential term as a function of the angle vector, $\boldsymbol{\theta} = (\theta_1, \theta_2, ..., \theta_N)$ where $N = \#Angles$ is as follows:

$$U_{Angle}(\boldsymbol{\theta}) = \sum_{i=1}^{\#Angles} \frac{1}{2} k_i^{\theta} (\theta_i - \theta_i^0)^2 \tag{1.23}$$

Where the summation is over all the angles in the system, k_i^{θ} is the spring constant and θ_i^0 is the equilibrium angle of the i^{th} angle.

The dihedral potential describes the torsional motion of a bond. A dihedral angle, can be defined as the angle between two planes formed by three atoms, ijk and jkl where i, j, k, l represent atoms such that i and j, j and k, k and l are bonded [Fig-1.1].



Dihedral Angle Illustration. Illustration of the Dihedral Angle, Φ

Figure 1.1

The often used functional form of the dihedral potential is as follows,

$$U_{Dihedral}(\boldsymbol{\phi}) = \sum_{i=1}^{\#Dihedrals} k_i^{\phi} \left[1 + cos(n\phi_i - \phi_i^0)\right] \tag{1.24}$$

Unlike the bond and angle potential terms, the dihedral potential can have multiple minima adjusted by the value "n". For instance, the dihedral term of a molecule with cis and trans symmetries would have two minima representing the two symmetries at $n = 2$.

The electrostatic interactions between atoms are represented by the Coulomb potential;

$$U_{Electrostatic}(\boldsymbol{r}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \tag{1.25}$$

where N is the number of atoms, ϵ_0 is the electric permitivity of free space, r_{ij} is the distance between atom i and j, and q_i, q_j are the electric charges of atom i and j respectively.

Last but not least, the Lennard-Jones(LJ) potential, which is also referred as the 6-12 potential is formalised as:

$$U_{LJ}(\boldsymbol{r}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right] \tag{1.26}$$

Where ϵ_{ij} is the depth of the LJ potential well and σ_{ij} is the finite distance where the LJ potential is zero for atoms i and j. The term with the power of 12 represents the repulsion due to the Pauli-Exclusion principle, and the term with the power of 6 models the attraction due to the van der Waals interactions. The LJ potential with its repulsion and attraction terms between atom i and j can be seen in Fiq-1.2

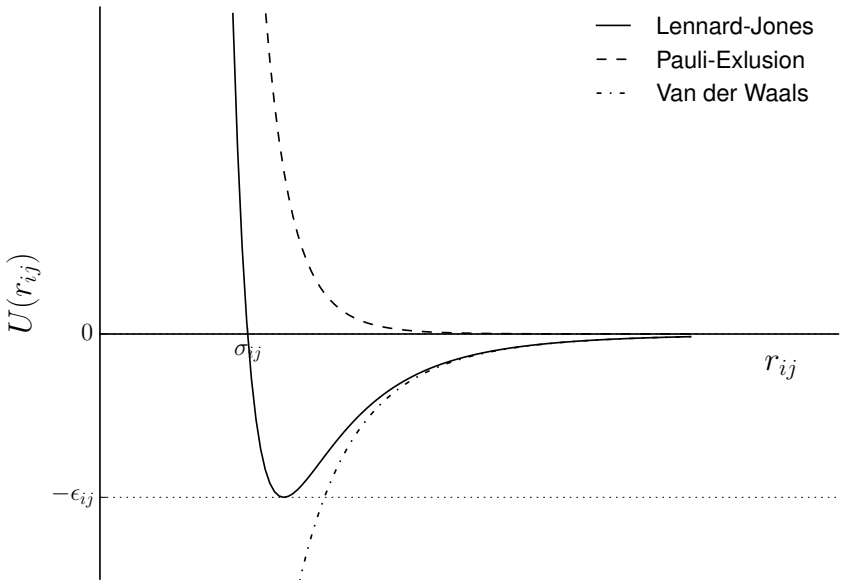


Figure 1.2 **Lennard-Jones Potential.** Lennard-Jones Potential with ϵ_{ij} and σ_{ij} parameters

To summarise, all the potential terms can be written as follows:

$$\begin{aligned} U = & \sum_{i=1}^{\#Bonds} \frac{1}{2} k_i^b (l_i - l_i^0)^2 \\ & + \sum_{i=1}^{\#Angles} \frac{1}{2} k_i^\theta (\theta_i - \theta_i^0)^2 \\ & + \sum_{i=1}^{\#Dihedrals} k_i^\phi \left[1 + cos(n\phi_i - \phi_i^0)\right] \\ & + \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \\ & + \sum_{i=1}^{N} \sum_{j=i+1}^{N} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right] \end{aligned} \tag{1.27}$$

1.2.3 Integrators

Integration algorithms are methods to calculate the derivative of a function numerically. They rely on taking the Taylor series expansion of the function of interest. For instance, the Taylor expansion of $f(x + dx)$ around x is,

$$f(x + dx) = \sum_{n=0}^{N} \frac{1}{n!} \frac{d^n f(x)}{dx^n} dx^n \tag{1.28}$$

If dx is small enough ($dx << 1$) the higher order terms in the Taylor series will make less contribution. Thus the function, $f(x + dx)$ can be approximated by truncating higher order terms in the Taylor series.

$$f(x + dx) = f(x) + \frac{df(x)}{dx} dx + \frac{1}{2} \frac{d^2 f(x)}{dx^2} dx^2 + \mathcal{O}(dx^3) \tag{1.29}$$

As can be seen from Eq-1.29, if we consider dx as the step of iteration, then the value of the function, f at the $x+dx$ step is formalised in terms of the values of the function, f and its derivatives at the previous step x. Similarly, the value of the function, f at step $x + 2dx$ can be written in terms of the values of f and its derivatives at $x + dx$ step. Thus the values of the function f, can be evaluated iteratively by a step size of dx. As a result, the change of f with respect to x will be obtained.
Since in Molecular Dynamics the goal is to find the change of the positions of the atoms with respect to time, the function f discussed above would be the position function q and the iteration step dx would be the time step dt. Thus Eq-1.29 can be reformulated as:

$$q_i(t + dt) = q_i(t) + \frac{dq_i(t)}{dt} dt + \frac{1}{2} \frac{d^2 q_i(t)}{dt^2} dt^2 + \mathcal{O}(dt^3), \quad i = 1, 2, ..., 3N \tag{1.30}$$

Similarly, the Taylor expansion for the momentum function, p of each atom can be written as:

$$p_i(t+dt) = p_i(t) + \frac{dp_i(t)}{dt}dt + \frac{1}{2}\frac{d^2p_i(t)}{dt^2}dt^2 + \mathcal{O}(dt^3), \quad i = 1, 2, ..., 3N \quad (1.31)$$

Where N is the number of atoms in the system.
Since the first derivative of the momentum Eq-1.15 and the second derivative of the position Eq-1.12, are related to the Force and the first derivative of the position is related to the momentum Eq-1.14, The Taylor expansions of the position, Eq-1.30 and momentum, Eq-1.31 can be written as:

$$q_i(t+dt) = q_i(t) + \frac{p_i(t)}{m_i}dt + \frac{F_i(t)}{2m_i}dt^2 + \mathcal{O}(dt^3), \quad i = 1, 2, ..., 3N \quad (1.32)$$

$$p_i(t+dt) = p_i(t) + F_i(t)dt + \mathcal{O}(dt^2), \quad i = 1, 2, ..., 3N \quad (1.33)$$

The calculation of the change of positions and momentum with time should follow Newton's equation of motion. Thus the integration methods discussed are numerical approximations to solve Newton's equation of motions or equivalently Hamilton's equations. The above method of integration is called the Euler integration method. Given the positions and momentum of each atom at time t, the trajectory of each atom's momentum and position can be calculated with an error in the order of dt^2 and dt^3 respectively. However, the Euler method is not able to fulfil some properties of the Hamilton equations, which makes the Euler method unsuitable for solving the equation of motion numerically.

The Hamilton's equations and Newton's equation of motions are time reversible, i.e. if the momenta of all atoms are reversed at any time instant, then the system would trace back its trajectory in phase space. They are volume preserving, i.e. the volume formed by the infinitesimal change in position, $d\boldsymbol{q}$ and velocities $d\boldsymbol{p}$ in phase space does not change with time. They conserve the total energy of the system. Thus a good numerical integration method that solves Hamilton's equations should have those properties or at least approximate these properties as good as possible. Numerical integrators that approximate well these properties are also called symplectic integrators.

In practice, two algorithms called leapfrog, and velocity Verlet is mostly used to carry out the numerical integration of Hamilton's equations. These algorithms have a smaller error due to truncation and are also symplectic integrators [14]. Thus preserving the phase space volume and are time reversible. They do not strictly conserve the total energy, but the long time drift of the total energy is very small. Also compared to the change in potential and kinetic energy, the change in the total energy is minimal. Symplectic integrators conserve a pseudo Hamiltonian which approach to the true Hamiltonian in the limit of the integration time step, dt goes to zero [27]. Thus the

leapfrog and velocity Verlet methods approximate the energy conservation property quite well. These properties make the leapfrog and velocity Verlet algorithms ideal to be used for molecular dynamics and hybrid Monte Carlo simulations.

1.2.3.1 Leap Frog

The truncation errors in the Leap-Frog algorithm are dt^4 and dt^3 for the positions and momentum respectively.

$$\boldsymbol{p}(t+\frac{1}{2}dt) = \boldsymbol{p}(t-\frac{1}{2}dt) + dt\boldsymbol{F}(t) + \mathcal{O}(dt^3) \tag{1.34}$$

$$\boldsymbol{q}(t+dt) = \boldsymbol{q}(t) + dt\boldsymbol{M}^{-1}\boldsymbol{p}(t+\frac{1}{2}dt) + \mathcal{O}(dt^4) \tag{1.35}$$

1.2.3.2 Velocity Verlet

The truncation errors in the velocity Verlet algorithm are dt^4 and dt^2 for the positions and momentum respectively. Although the error of the momentum is higher for velocity Verlet than leapfrog, the obtained momentums are the full-time step momentums. Therefore, any quantity related to the momentums can be calculated for the full-time step, without any other extra calculation [28]

$$\boldsymbol{p}(t+\frac{1}{2}dt) = \boldsymbol{p}(t) + \frac{dt}{2}\boldsymbol{F}(t) \tag{1.36}$$

$$\boldsymbol{q}(t+dt) = \boldsymbol{q}(t) + dt\boldsymbol{M}^{-1}\boldsymbol{p}(t+\frac{1}{2}dt) + \mathcal{O}(dt^4) \tag{1.37}$$

$$\boldsymbol{p}(t+dt) = \boldsymbol{p}(t+\frac{1}{2}dt) + \frac{dt}{2}\boldsymbol{F}(t+dt) + \mathcal{O}(dt^2) \tag{1.38}$$

1.2.3.3 Generalised Verlet

The above mentioned integrator algorithms are suitable for separable Hamiltonians. If we apply the leapfrog or Verlet integrator algorithms to the non-separable Hamiltonian case directly the integrator becomes non-reversible. Thus to make the integrator time reversible for non-separable Hamiltonians the generalised Stoermer-Verlet algorithm can be used, which can be constructed from two semi-implicit also known as symplectic Euler integrators [29]. And the symplectic Euler is composed of one implicit and one explicit Euler integrator.

An explicit Euler integrator is a first order taylor expansion of $y(t+dt)$ evaluated at time t

$$y(t+dt) = y(t) + dt\frac{dy(t)}{dt} \tag{1.39}$$

An the implicit Euler,

$$y(t+dt) = y(t) + dt\frac{d}{dt}y(t+dt) \tag{1.40}$$

In the case of Hamiltonian dynamics, two different symplectic Euler integrators can be constructed

Symplectic Euler-1. Implicit Euler to update p and an explicit Euler to update q

$$\begin{aligned} \boldsymbol{p}(t+dt) &= \boldsymbol{p}(t) - dt\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t), \boldsymbol{p}(t+dt)) \\ \boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + dt\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t), \boldsymbol{p}(t+dt)) \end{aligned} \tag{1.41}$$

Symplectic Euler-2. Implicit Euler to update q and an explicit Euler to update p

$$\begin{aligned} \boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + dt\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t+dt), \boldsymbol{p}(t)) \\ \boldsymbol{p}(t+dt) &= \boldsymbol{p}(t) - dt\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t+dt), \boldsymbol{p}(t)) \end{aligned} \tag{1.42}$$

Generalised Stoermer-Verlet integrator is constructed by combining a half step update of Euler-1 with a half step update of Euler-2

$$\begin{aligned} \boldsymbol{p}(t+\frac{dt}{2}) &= \boldsymbol{p}(t) - \frac{dt}{2}\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t), \boldsymbol{p}(t+\frac{dt}{2})) \\ \boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + \frac{dt}{2}\left[\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t), \boldsymbol{p}(t+\frac{dt}{2})) + \frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t+dt), \boldsymbol{p}(t+\frac{dt}{2}))\right] \\ \boldsymbol{p}(t+dt) &= \boldsymbol{p}(t+\frac{dt}{2}) - \frac{dt}{2}\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t+dt), \boldsymbol{p}(t+\frac{dt}{2})) \end{aligned} \tag{1.43}$$

As can be seen, in the first two steps in the Generalised Stoermer-Verlet integrator is an implicit update, as the updated value depends on itself. Thus those steps need to be calculated recursively to find a stable updated value. The stable or fixed value can be obtained by a simple fixed point iteration method, where the implicit step is recalculated until the obtained value does not change.

It is also important to mention that the Generalized Stoermer-Verlet Integrator reduces to Verlet Integrator for separable Hamiltonians. Thus the method is, in fact, a generalized version of the previously explained velocity Verlet algorithm

for a separable Hamiltonian, i.e.

$$H(q, p) = U(q) + K(p) \tag{1.44}$$

where $U(q)$ is the potential and $K(p)$ is the kinetic energy of the system. The derivatives of the separable Hamiltonian is as follows,

$$\begin{aligned}\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q},\boldsymbol{p}) &= \frac{d}{d\boldsymbol{q}}U(\boldsymbol{q}) + \frac{d}{d\boldsymbol{q}}K(\boldsymbol{p}) \\ &= \frac{d}{d\boldsymbol{q}}U(\boldsymbol{q}) = -\boldsymbol{F}(\boldsymbol{q})\end{aligned} \tag{1.45}$$

$$\begin{aligned}\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q},\boldsymbol{p}) &= \frac{d}{d\boldsymbol{p}}U(\boldsymbol{q}) + \frac{d}{d\boldsymbol{p}}K(\boldsymbol{p}) \\ &= \frac{d}{d\boldsymbol{p}}K(\boldsymbol{p})\end{aligned} \tag{1.46}$$

If $K(\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^{\boldsymbol{T}}\boldsymbol{M}^{-1}\boldsymbol{p}$ then

$$\frac{d}{d\boldsymbol{p}}K(\boldsymbol{p}) = \boldsymbol{M}^{-1}\boldsymbol{p} \tag{1.47}$$

Thus when the Hamiltonian is separable, the derivatives given in the generalised Stoermer-Verlet integrator, Eq-1.43, will be as follows:

$$\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t),\boldsymbol{p}(t+\frac{dt}{2})) = -\boldsymbol{F}(\boldsymbol{q}(t)) \tag{1.48}$$

$$\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t+dt),\boldsymbol{p}(t+\frac{dt}{2})) = -\boldsymbol{F}(\boldsymbol{q}(t+dt)) \tag{1.49}$$

$$\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t),\boldsymbol{p}(t+\frac{dt}{2})) = \boldsymbol{M}^{-1}\boldsymbol{p}(t+\frac{dt}{2}) \tag{1.50}$$

$$\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t+dt),\boldsymbol{p}(t+\frac{dt}{2})) = \boldsymbol{M}^{-1}\boldsymbol{p}(t+\frac{dt}{2}) \tag{1.51}$$

substituting Eq-1.48, Eq-1.49, Eq-1.50, Eq-1.51 into Eq-1.43, we obtain.

$$\begin{aligned}\boldsymbol{p}(t+\frac{dt}{2}) &= \boldsymbol{p}(t) + \frac{dt}{2}\boldsymbol{F}(\boldsymbol{q}(t)) \\ \boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + \frac{dt}{2}\boldsymbol{M}^{-1}\left[\boldsymbol{p}(t+\frac{dt}{2}) + \boldsymbol{p}(t+\frac{dt}{2})\right] \\ \boldsymbol{p}(t+dt) &= \boldsymbol{p}(t+\frac{dt}{2}) + \frac{dt}{2}\boldsymbol{F}(\boldsymbol{q}(t+dt))\end{aligned} \tag{1.52}$$

By further simplifying the second update equation, in 1.52 we obtain the velocity Verlet integrator as given in Eq-1.36

$$\begin{aligned}
\boldsymbol{p}(t+\frac{dt}{2}) &= \boldsymbol{p}(t) + \frac{dt}{2}\boldsymbol{F}(\boldsymbol{q}(t)) \\
\boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + dt\boldsymbol{M}^{-1}\boldsymbol{p}(t+\frac{dt}{2}) \\
\boldsymbol{p}(t+dt) &= \boldsymbol{p}(t+\frac{dt}{2}) + \frac{dt}{2}\boldsymbol{F}(\boldsymbol{q}(t+dt))
\end{aligned} \tag{1.53}$$

1.3 Monte Carlo

The foundations of the Monte Carlo (MC) method used today for molecular systems, also called the Monte Carlo Metropolis method, was first laid by Metropolis et al. [30]. Starting from a configuration, $\boldsymbol{q}^0 = (q_1^0, q_2^0, ..., q_{3N}^0)$ the Monte Carlo Metropolis method generates a sequence of configurations via a transition kernel, $K(\boldsymbol{q^{n+1}}|\boldsymbol{q^n})$ with $n \geq 0$ that has the target distribution $P(\boldsymbol{q})$. The transition kernel is a conditional probability, that gives the probability to transition to a next configuration given the current configuration. The transition kernel can be written in terms of a proposal, $Q(\boldsymbol{q^{n+1}}|\boldsymbol{q^n})$ probablity, that in practice is easy to simulate and the acceptance, $A(\boldsymbol{q^{n+1}}, \boldsymbol{q^n})$ probability.

$$K(\boldsymbol{q^{n+1}}|\boldsymbol{q^n}) = Q(\boldsymbol{q^{n+1}}|\boldsymbol{q^n})A(\boldsymbol{q^{n+1}}, \boldsymbol{q^n}) \tag{1.54}$$

The criterion for the acceptance probability is determined via the detailed balance condition, which is sufficient to ensure that the sequence of configurations generated, form an ensemble or equivalently creates a thermodynamic equilibrium. The detailed balance condition can be written as follows:

$$P(\boldsymbol{x})Q(\boldsymbol{y}|\boldsymbol{x})A(\boldsymbol{y},\boldsymbol{x}) = P(\boldsymbol{y})Q(\boldsymbol{x}|\boldsymbol{y})A(\boldsymbol{x},\boldsymbol{y}) \tag{1.55}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are any molecular configurations. By rearranging Eq-1.55,

$$\frac{A(\boldsymbol{y},\boldsymbol{x})}{A(\boldsymbol{x},\boldsymbol{y})} = \frac{P(\boldsymbol{y})Q(\boldsymbol{x}|\boldsymbol{y})}{P(\boldsymbol{x})Q(\boldsymbol{y}|\boldsymbol{x})} \tag{1.56}$$

can be obtained. One choice of acceptance probability which fulfills the condition in Eq-1.56, is as follows:

$$A(\boldsymbol{y},\boldsymbol{x}) = min\left\{1, \frac{P(\boldsymbol{y})Q(\boldsymbol{x}|\boldsymbol{y})}{P(\boldsymbol{x})Q(\boldsymbol{y}|\boldsymbol{x})}\right\} \tag{1.57}$$

The aforementioned process of generating a sequence of random variates following a distribution $P(\boldsymbol{q})$ with the general formalism of the acceptance criterion, Eq-1.57, is called the Metropolis-Hastings method [31]. The MC Metropolis method is a special case of the Metropolis-Hastings method, where the proposal probability, $Q(\boldsymbol{y}|\boldsymbol{x})$ is symmetrical. In this context by the term symmetrical it is meant that, the proposal

probability of $\boldsymbol{y}$ given $\boldsymbol{x}$ is the same as $\boldsymbol{x}$ given $\boldsymbol{y}$.

$$Q(\boldsymbol{y}|\boldsymbol{x}) = Q(\boldsymbol{x}|\boldsymbol{y}) \tag{1.58}$$

Thus, Eq-1.57 reduces to the following;

$$A(\boldsymbol{y}, \boldsymbol{x}) = min\left\{1, \frac{P(\boldsymbol{y})}{P(\boldsymbol{x})}\right\} \tag{1.59}$$

For instance consider the Boltzmann distribution as the target,

$$P(\boldsymbol{x}) = \exp(-\beta U(\boldsymbol{x})) \tag{1.60}$$

where $\beta = 1/k_B T$ is the inverse temperature, and U is the potential energy of the molecular system. If the proposal probability is chosen such that, the coordinates of the atoms are given a random displacement, then the proposal probability will be symmetrical. Hence, the acceptance criterion for the MC Metropolis method can be used for this system.

$$A(\boldsymbol{y}, \boldsymbol{x}) = min\left\{1, \frac{\exp(-\beta U(\boldsymbol{y}))}{\exp(-\beta U(\boldsymbol{x}))}\right\} \tag{1.61}$$

According to the acceptance criterion a new configuration with a lower potential energy is always accepted whereas a configuration with a higher potential energy is accepted with the probability $\exp(-\beta[U(\boldsymbol{y}) - U(\boldsymbol{x})])$. After convergence is achieved, the configurations generated with this method associated with the target distribution in Eq-1.60, will form the canonical (NVT) ensemble.
Since the proposal probability is constructed by adding random displacements to the configuration, the probability of accepting a new configuration will decrease with increasing displacements and dimensionality of the configurations [32]. Thus to have a relatively high acceptance rate small displacements are required. On the other hand, having small displacements may result in sampling more locally, which will increase the required number of Monte Carlo steps to achieve global convergence. The MC steps required to get convergence will increase even more with the dimensions or degrees of freedom of the system, such that at some point it might get computationally too expensive to observe convergence.
One way to tackle this convergence problem is by choosing a proposal probability, that generates configurations in a wider region of the configuration space with a relatively high acceptance rate. One such proposal probability is generated via Molecular Dynamics steps. The method that uses Molecular dynamics as the update mechanism of its configuration is called the Hybrid Monte Carlo method [8]. A detailed theoretical background of this method and its extensions will be given in later sections.

A pseudocode to generate random variates from a target distribution for the general case of the proposal probability, the Metropolis-Hastings method, can be seen in

Algorithm-1.

Algorithm 1 Metropolis Hastings. Metropolis Hastings method

$x^0 \leftarrow$ Starting configuration
$n \leftarrow 0$
while $n < \#Steps$ **do**
    Generate $\boldsymbol{y} \sim Q(\boldsymbol{y}|\boldsymbol{x^n})$
    $accept \leftarrow \frac{P(\boldsymbol{y})Q(\boldsymbol{x^n}|\boldsymbol{y})}{P(\boldsymbol{x^n})Q(\boldsymbol{y}|\boldsymbol{x^n})}$
    $X \sim U(0,1)$
    if $X < accept$ **OR** $accept \geq 1$ **then**
        $x^{n+1} \leftarrow \boldsymbol{y}$
    else
        $x^{n+1} \leftarrow \boldsymbol{x^n}$
    end if
    $n \leftarrow n+1$
end while

1.4 Fourier Transform

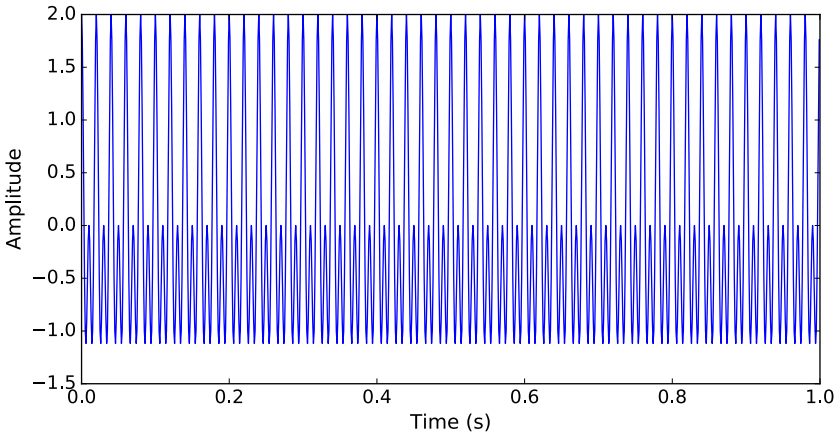
A signal can be represented as a weighted integral of complex exponentials or equivalently sine and cosine functions [33]. The representation of a function in terms of a linear combination of $sin(2\pi ft)$, $cos(2\pi ft)$ or $e^{i2\pi ft}$ periodic signals with different frequencies, f, is called the Fourier series expansion. Thus the Fourier expansion gives the frequency decomposition of a signal and can be used to find out the frequency spectrum of the signal. The coefficients of a signal's Fourier series expansion, which represent the contribution of each frequency to the signal, can be calculated numerically using the discrete Fourier transform. Using the discrete Fourier transform the frequency and phase spectrum of a signal can be obtained.

As an example, consider a superposition of two sin signals one with 50 Hz and the other with 100 Hz frequency Eq-1.62. The amplitudes of the two sin signals are equal and set to 1 unit.

$$x(t) = sin(2\pi 50t) + sin(2\pi 100t) \tag{1.62}$$

The corresponding graph of Eq-1.62 plotted between $0 - 1$ second, time interval can be seen in Figure-1.3

Taking the Fourier transform of signal $x(t)$ given in Eq-1.62, the frequency spectrum of the signal can be obtain Fig-1.4. As can be seen, the figure shows two peaks at 50Hz and 100 Hz with an equal height of 1. Thus the Fourier transform reveals the frequency components of the time domain signal. The height of the peaks corresponds to the amplitudes of each frequency component of the signal. Such a graph is also



Superposition of two sin signals. The graph shows the sum of two *sin* signals with 50 and 100 Hz frequency and an amplitude of 1 unit Eq-1.62. The signal is plotted between 0 – 1 second time interval

Figure 1.3

called the amplitude spectrum. If the height of the peaks is the square of the amplitude, then the graph is called the power spectrum.

The expansion of a signal in terms of the linear combination of the complex exponentials is the inverse Fourier transform.

$$x(t) = \int x(f) e^{i2\pi f t} df \tag{1.63}$$

where $x(f_k)$ are called the Fourier coefficients. For a discretised signal the Fourier coefficients can be calculated via the discrete Fourier transform,

$$x(f_k) = \frac{1}{N} \sum_{n=0}^{N-1} x(t_n) e^{-i\frac{2\pi}{N} nk} \tag{1.64}$$

The Fourier coefficients $x(f_k) = a_k + ib_k$ have a real and imaginary component from which the amplitude and phase for a given frequency can be calculated.

The amplitude of the k^{th} frequency is as follows,

$$A_k = \sqrt{a_k^2 + b_k^2} \tag{1.65}$$

The phase of the k^{th} frequency is as follows,

$$\phi_k = \arctan\left(\frac{b_k}{a_k}\right) \tag{1.66}$$

The frequency and phase spectrum can be generated via the amplitude and phase values, respectively. However, if the signal's frequency changes with time, then the Fourier transform will not give a good frequency representation of the signal, the time

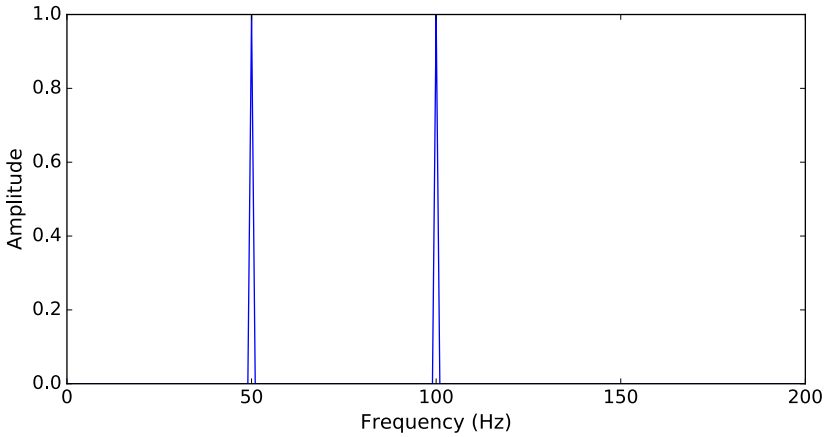


Figure 1.4 **Fourier transform.** The amplitude spectrum of the signal given in Eq-1.62. The graph shows the two frequencies of the signal 50, 100 Hz and the amplitudes

evolution of the frequency can not be captured.

As an example, a signal with time varying frequency is constructed Eq-1.67. The signal, similar to before is based on two sin signals with 50 and 100 Hz frequencies. However instead of taking the superposition of the two signals, the 50Hz signal is present on its own only between 0.1-0.3 seconds, and the 100 Hz signal is present between 0.7-1.0 seconds; otherwise, the signal takes a value of zero. A plot of this signal can be seen in Figure-1.6

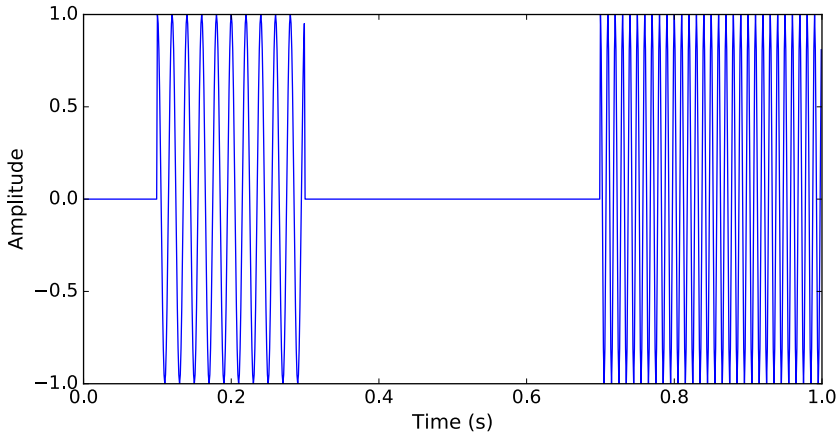
$$x(t) = \begin{cases} sin(2\pi 50t) & 0.1 \leq t < 0.3 \\ sin(2\pi 100t) & 0.7 < t \leq 1.0 \\ 0 & otherwise \end{cases} \tag{1.67}$$

As can be seen from the Fourier transform of the signal, the frequencies can be identified; however, the information of the time dependence of the frequencies is lost. Also, the amplitudes of the frequency spectrum are not reflective of the amplitudes of the time domain signal. Thus the standard Fourier transform is not suitable to study signals where the frequency changes with time, especially if the interest is to understand the time dependence of the frequencies of the signal.

1.5 Wavelet Transform

The term wavelet was coined by Jean Morlet who developed the Morlet wavelets, but the first wavelet was developed earlier by Alfred Haar which led to the Haar wavelets [34].

In wavelet transform the signal is written in terms of an infinite weighted sum over



Piecewise combination of two sin signals. The graph shows the combination of two sin signals with 50 and 100 Hz frequency and an amplitude of 1 unit. The 50Hz signal is present between 0.1-0.3 seconds and the 100 Hz signal is present between 0.7-1 seconds; otherwise, the signal takes the value zero Eq-1.67. The signal is plotted between 0−1 second time interval

Figure 1.5

wavelets that form a basis set similar to the complex exponentials used in Fourier transform. The difference from the basis used in Fourier transform is that wavelets are not only a localized function over frequencies but also of time allowing the exploration of the time dependent frequency property of the signal. Thus if a signal's frequency changes with time and we want to capture how the frequency of the signal evolves with time, then the wavelet transform can be used.

The signal $x(t)$ can be represented using wavelets ψ as follows.

$$x(t) = \int\int W(s,k)\psi\left(\frac{(t-k)}{s}\right) dsdk \tag{1.68}$$

where k and s within the wavelet represents the shift in time and frequency, respectively. The wavelet coefficients $W(s,k)$ for a discretised signal can be obtained via the wavelet transform

$$W(s,kdt) = \frac{dt}{\sqrt{s}}\sum_n x(ndt)\psi\left(\frac{(n-k)dt}{s}\right) \tag{1.69}$$

There are many different wavelets ψ, which define the functional forms and are generally chosen depending on the application and signal characteristics. Some well known wavelets are the Morlet [35], Haar [36], Daubechies [37] wavelets.

If we apply a wavelet transform to the signal in Eq-1.67, we can obtain the frequencies of the signal, but also the time dependence of the frequencies as can be seen in Figure-1.7. Thus the wavelet transform is ideal for studying the signal's frequency changes with time.

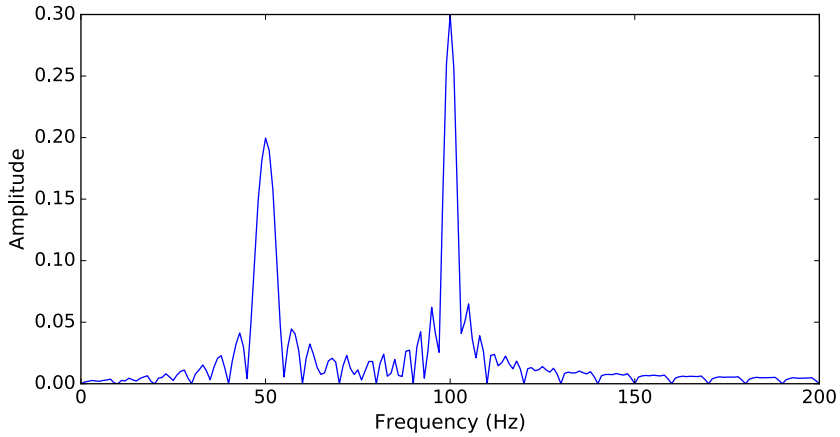


Figure 1.6 **Fourier transform.** The amplitude spectrum of the signal given in Eq-1.62. The transform can identify the signal's frequencies. However, the amplitudes are not representative of the actual amplitudes of the time domain signal. Also, no information can be obtained how the frequency changes with time

1.6 Digital Filters

The atoms in a molecule will vibrate due to the bonded interactions, explained in the force-fields section. The vibrations of the atoms can be thought as a superposition of simpler vibrations called the normal modes. In general, the number of normal modes for a molecule with N atoms is $3N - 6$. In case the molecule is linear, the number of normal modes will be $3N - 5$. To target a specific mode of vibration in a molecule a digital filter can be used. Thus, with a digital filter, it is possible to enhance or suppress specific vibrational modes for a specific atom or a collection of atoms in a molecule. The digital filter used in this study is applied to the atomic velocities. For $2m + 1$ number of filter coefficients, c_i the filtered velocity for the k^{th} time step is a linear combination of the velocities with weights c_i.

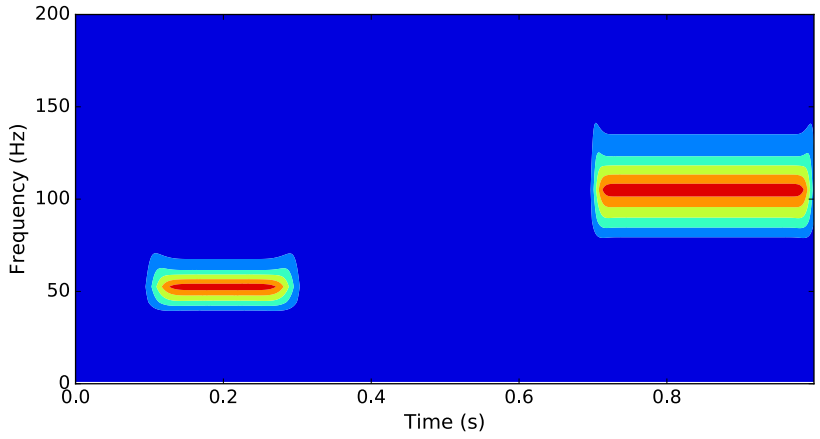
$$v^{filtered}(t_k) = \sum_{i=-m}^{m} c_i v(t_{i+k}) \tag{1.70}$$

The filter coefficients, c_i are determined such that the frequency response, $H(\omega)$,

$$H(\omega) = \sum_{i=-m}^{m} c_i e^{-j\omega i}, \quad where \quad j = \sqrt{-1} \tag{1.71}$$

approximates the desired frequency response, $D(\omega)$, by minimising ϵ

$$\epsilon = \int_{-\pi}^{\pi} |D(\omega) - H(\omega)|^2 \, d\omega \tag{1.72}$$



wavelet transform of signal. Time, frequency plot based on wavelet transform of the piece-wise *sin* signal given in Eq-1.67. Morlet wavelets are used as the basis function for the wavelet transform

Figure 1.7

The coefficients can be obtained via Eq. 1.71 and Eq. 1.72 to give Eq. 1.73

$$c_i = \frac{1}{2\pi}\int_{-\pi}^{\pi} D(\omega)e^{j\omega i}d\omega, \quad j = \sqrt{-1} \tag{1.73}$$

According to Eq-1.73, the filter coefficients can be obtained without the need to do an MD simulation. The only information required is the desired filter response. Thus, for a given filter response the filter coefficients can be calculated prior to the MD simulation, and then can be used to filter the desired frequencies from the atom's velocities.

In this study, the coefficients are obtained using the MATLAB's [38] $\boldsymbol{fircls()}$ function. The $\boldsymbol{fircls()}$ command is used as follows;

$$\boldsymbol{c} = \boldsymbol{fircls}(n,\ [0\ a\ b\ 1],\ [s1\ s2\ s3],\ [u1\ u2\ u3],\ [l1\ l2\ l3],\ 'text') \tag{1.74}$$

The number of coefficients is $n+1$, and $0,\ a,\ b,\ 1$ are the boundaries that partition the frequency range into three regions. The first region is between 0 and a, the second region is between a and b, and the last one is between b and 1. The parameters $s1,\ s2,\ s3$ are the scaling factor for each region respectively. Thus, the first region is multiplied by $s1$, the second by $s2$ and the third by $s3$. The last two parameter sets $[u1\ u2\ u3],\ [l1\ l2\ l3]$ are the constraints for the scaling factors. Thus the value of $s1$ is constrained between lower bound $l1$ and upper bound $u1$, and $s2,\ s3$ are constrained between $l2,\ u2$ and $l3,\ u3$ respectively. It has to be noted that the frequency values in the first set of parameters $[0\ a\ b\ 1]$ are normalised frequency. The normalised frequencies can be converted from the frequency in wavenumber (cm^{-1}) units as

follows;

$$f_{normalised} = \frac{f_{wavenumbers} \times c}{g(dt)} \tag{1.75}$$

Where c is the speed of light in the units of $cm\,s^{-1}$, and $g(dt)$ is the Nyquist critical frequency [39],

$$g(dt) = \frac{1}{2 \times dt} \tag{1.76}$$

here dt is the simulation time step. The Nyquist critical frequency is the maximum frequency that can be sampled for a time step of dt.

As an example, consider a superposition of two sin waves with frequencies of 1 and 10 Hz respectively.

$$signal = sin(2\pi t) + sin(20\pi t) \tag{1.77}$$

If the time step to sample this signal is $dt = 0.01$s then the Nyquist frequency is,

$$f_{Nyquist} = \frac{1}{2x0.01} = 50Hz \tag{1.78}$$

and using the Nyquist frequency, the normalised frequencies of the signal's frequency components can be calculated as,

$$\begin{aligned} f^{1Hz}_{normalised} &= \frac{f^{1Hz}}{f_{Nyquist}} \\ &= \frac{1}{50} = 0.02 \end{aligned} \tag{1.79}$$

$$\begin{aligned} f^{10Hz}_{normalised} &= \frac{f^{10Hz}}{f_{Nyquist}} \\ &= \frac{10}{50} = 0.2 \end{aligned} \tag{1.80}$$

To filter out the higher frequency vibration from the signal Eq 1.77, a low pass filter can be designed using Matlab's $fircls$ function as follows,

$$\boldsymbol{c = fircls}(1000,\ [0\ 0.15\ 1],\ [1\ 0],\ [1.01\ 0.01],\ [0.09\ \ -0.01],\ 'text') \tag{1.81}$$

This will generate 1001 coefficients such that when multiplied with the signal as in 1.70, the amplitude of signals with normalised frequencies between $[0, 0.15]$ will get multiplied by one and between $[0.15, 1]$ will get multiplied by zero. The filter response obtained for the designed filter with 1001 coefficients can be seen in Fig 1.8, which approximates the desired filter response quite well. Hence, all the signals that have normalised frequencies greater than 0.15 will be filtered out [Fig 1.9].

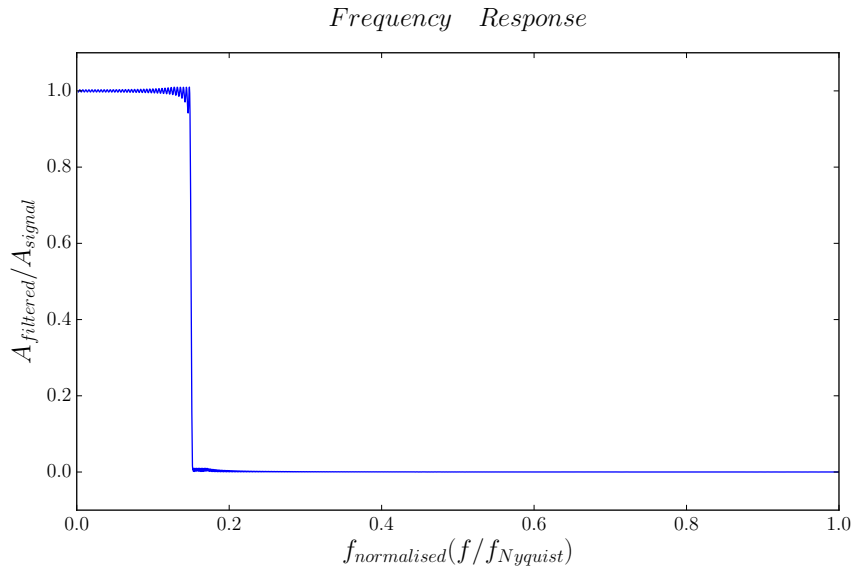


Figure 1.8

Frequency Response. The frequency response of the designed filter given by MATLAB command 3.24. $A_{filtered}$ is the amplitude of the filtered sin signal and A_{signal} is the amplitude of the sin signal

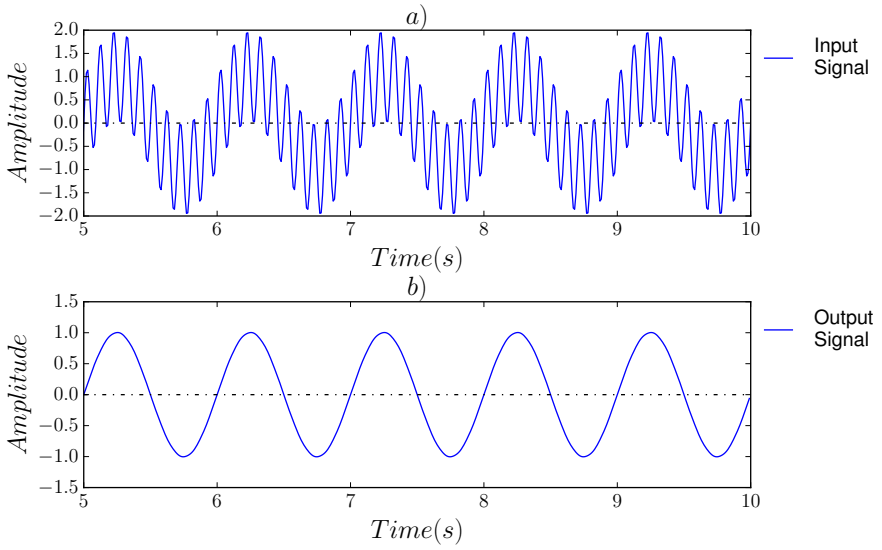


Figure 1.9

Signal-Low Pass Filter. a) Input signal composed of the superposition of 1 and 10 Hz sin-waves b) Output signal (1 Hz) after low-pass filter (3.24) is applied to the input signal

1.7 Enhanced Sampling Methods

Enhanced sampling methods can be classified broadly into two groups: ones that bias the potential energy and the ones that modify the underlying velocity distribution or kinetic energy. Other methods exist that fall outside this categorization. For instance, SWARM-MD, which neither modifies the velocities nor the potential energy of the system but enhances the sampling by using multiple different starting conformations of the molecule [40], or Markov State Models [41], [42] which also uses different conformations and their transition probabilities to enhance the sampling. However, in the scope of this study, the given classification is sufficient to understand the methods discussed.

In the following subsections, umbrella sampling [43], [44] and Metadynamics methods [4], [45] are discussed as examples for methods that change the underlying potential energy. And for methods that change the underlying kinetic energy the replica exchange molecular dynamics [7] method and the method developed in this thesis called DFHMC is discussed.

Furthermore, to build the theory of the DFHMC method, the DFMD [46] and HMC methods [8] are explained in more detail.

1.7.1 Methods that change the underlying potential energy

These methods usually try to obtain a uniform distribution along some chosen variables. Thus creating a flat surface and thereby enhancing the sampling along the chosen variables. However it was also shown that a non-uniform target distribution gives a better sampling perfomance compared to a uniform target distribution [47]. The chosen variables are a function of the atomic coordinates and are called collective variables (CV's). Some common CV's are dihedral angles and distances between atoms. The underlying assumption in those methods is that the efficient sampling along the chosen CV's captures the relevant configurational changes in the system. Thus the choice of the CV's is the most important part of those methods to obtain a well sampled simulation.

Two well known methods that fall into this category are umbrella sampling and metadynamics. They differ in their approach to flatten the distribution along the specified CV's

Below we will show what the desired bias potential is to flatten the distribution, and in the following subsection for umbrella sampling and metadynamics we discuss the approach they use to obtain the desired bias potential.

In a canonical or NVT ensemble, the configurational distribution is given by the Boltzmann distribution.

$$P(\boldsymbol{q}) = \frac{\exp(-\beta U(\boldsymbol{q}))}{\int \exp(-\beta U(\boldsymbol{q}))d\boldsymbol{q}} \tag{1.82}$$

where $U(\boldsymbol{q})$ is the potential energy of the system and $\beta = 1/k_B T$ is the inverse temperature.

Let us define the collective variable (CV) as $\boldsymbol{s}(\boldsymbol{q}) = \{s_1(\boldsymbol{q}), s_2(\boldsymbol{q}), ..., s_3(\boldsymbol{q})\}$. The distribution along the CV's is,

$$P(\boldsymbol{s}) = \int \delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})] P(\boldsymbol{q}) d\boldsymbol{q} = \frac{\int \delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})] \exp(-\beta U(\boldsymbol{q})) d\boldsymbol{q}}{\int \exp(-\beta U(\boldsymbol{q})) d\boldsymbol{q}} \tag{1.83}$$

where the dirac-delta function $\delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})]$ ensures the integral is only evaluated for any $\boldsymbol{q}$ that satisfies the relation $\boldsymbol{s} = \boldsymbol{s}(\boldsymbol{q})$

the potential energy of the system is modified by adding a bias potential that is a function of the collective variables.

$$U^b(\boldsymbol{q}) = U^u(\boldsymbol{q}) + V(\boldsymbol{s}) \tag{1.84}$$

where $V(\boldsymbol{s})$ is the bias potential, the superscripts u and b refers to the unbiased and biased systems respectively

The distribution in the biased system is,

$$P^b(\boldsymbol{q}) = \frac{\exp(-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})])}{\int \exp(-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]) d\boldsymbol{q}} \tag{1.85}$$

and the distribution along the CV's in the biased system is,

$$\begin{aligned} P^b(\boldsymbol{s}) &= \int \delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})] P^b(\boldsymbol{q}) d\boldsymbol{q} \\ &= \frac{\int \delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})] \exp(-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]) d\boldsymbol{q}}{\int \exp(-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]) d\boldsymbol{q}} \\ &= e^{-\beta V(\boldsymbol{s})} \frac{\int \delta[\boldsymbol{s} - \boldsymbol{s}(\boldsymbol{q})] e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q}}{\int e^{-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]} d\boldsymbol{q}} \end{aligned} \tag{1.86}$$

By using Eq-1.83 and Eq-1.86 we obtain the following,

$$P^b(\boldsymbol{s}) = e^{-\beta V(\boldsymbol{s})} P^u(\boldsymbol{s}) \frac{\int e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q}}{\int e^{-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]} d\boldsymbol{q}} \tag{1.87}$$

$$\begin{aligned} \frac{\int e^{-\beta U(\boldsymbol{q})} d\boldsymbol{q}}{\int e^{-\beta[U(\boldsymbol{q}) + V(\boldsymbol{s})]} d\boldsymbol{q}} &= \frac{1}{< e^{-\beta V(\boldsymbol{s})} >_u} \\ &= \frac{1}{\int e^{-\beta V(\boldsymbol{s})} P^u(\boldsymbol{s}) d\boldsymbol{s}} \end{aligned} \tag{1.88}$$

The free energy along the CV is defined as,

$$F^u(\boldsymbol{s}) = -\frac{1}{\beta} \log P^u(\boldsymbol{s}) \tag{1.89}$$

thus

$$P^u(\boldsymbol{s}) = e^{-\beta F^u(\boldsymbol{s})} \tag{1.90}$$

using the definition of the free energy Eq-1.86 can be rewritten as follows,

$$\begin{aligned} P^b(\boldsymbol{s}) &= \frac{e^{-\beta V(\boldsymbol{s})} e^{-\beta F^u(\boldsymbol{s})}}{\int e^{-\beta V(\boldsymbol{s})} e^{-\beta F^u(\boldsymbol{s})} d\boldsymbol{s}} \\ &= \frac{e^{-\beta [F^u(\boldsymbol{s}) + V(\boldsymbol{s})]}}{\int e^{-\beta [F^u(\boldsymbol{s}) + V(\boldsymbol{s})]} d\boldsymbol{s}} \end{aligned} \tag{1.91}$$

Thus to make the distribution of the biased system along the CV's flat the bias potential, $V(\boldsymbol{s})$, should be equal to the negative of the free energy along the CV's of the unbiased system.

$$V(\boldsymbol{s}) = -F^u(\boldsymbol{s}) \tag{1.92}$$

Umbrella Sampling

Umbrella sampling developed by Torrie and Valleau [43] commonly adds a harmonic potential along the CV's centered around a given value of the CV to restrict the sampling to the vicinity of the given CV value.

$$V(\boldsymbol{s}) = \frac{k}{2}(\boldsymbol{s} - \boldsymbol{s}_0)^2 \tag{1.93}$$

Having a high k only points close to s_0 will be explored. Multiple simulations with varying s_0 are conducted to sample the configurations along $\boldsymbol{s}$. Since the different bias potentials are not dependent on each other, the simulations can be conducted in parallel. Afterwards, individual trajectories are combined using a reweighting method. The often used reweighting method in conjunction with umbrella sampling is the weighted histogram analysis method (WHAM) [48], which is an extension of the multiple histogram technique [49]. An alternative to WHAM to combine the multiple simulation trajectories in umbrella sampling is the umbrella integration method [50].

The parameter k and the step size between consecutive CV values are usually chosen such that the simulation gives a sampling from a flat surface.

MetaDynamics

In metadynamics developed by Laio and Parrinello [51], [4] to enhance the simulation, the bias potential is taken in the following form:

$$V(\boldsymbol{s}) = -(1 - \frac{1}{\gamma})F(\boldsymbol{s}) \tag{1.94}$$

Thus if $\gamma = 1$ then $V(\boldsymbol{s}) = 0$ which leads to an unbiased (original) systems

when $\gamma \to \inf$ then $V(\boldsymbol{s}) \to -F(\boldsymbol{s})$ which leads to a flat distribution along $\boldsymbol{s}$

In practice $F(\boldsymbol{s})$ is not known apriori, thus in Metadynamics the bias potential is

adaptively estimated to asymptotically converge to Eq-1.97. This results in having dependent bias potentials; thus the simulations for different bias potentials along a given CV cannot be run in parallel as was the case with Umbrella Sampling. However multiple CV's can be run in parallel, and an exchange between the biases can be made as per the biased exchange method [52]

The metadynamics implementation uses the following bias potential

$$V(\boldsymbol{s}, t) = \sum_{k\tau<t} W(k\tau) exp\left(-\sum_{i=1}^{d} \frac{(s_i - s_i(q(k\tau)))^2}{2\sigma_i^2}\right) \tag{1.95}$$

where τ is the Gaussian deposition stride and $k = 0, 1, 2...$ such that $k\tau$ results in discrete time points with stride τ. σ_i is the standard deviation of the Gaussian for the i^{th} CV, $W(k\tau)$ is the height of the Gaussian. In standard metadynamics the height of the deposited Gaussians are constant, resulting in the estimated free energy to fluctuate around the actual free energy. In Well Tempered Metadynamics the Gaussian heights are adjusted at each deposition based on the following equation [53]

$$W(k\tau) = W_0 exp\left(-\frac{V(s(q(k\tau)), k\tau)}{k_B \Delta T}\right) \tag{1.96}$$

Where k_B is the Boltzmann constant, W_0 a constant initial Gaussian height, ΔT is a parameter with temperature dimensions. With the rescaled Gaussian height, the bias potential converges smoothly but not exactly to the actual free energy. The deviation from the actual free energy can be observed when calculating the long time limit of the bias potential.

$$V(\boldsymbol{s}, t \to \text{inf}) = -\frac{\Delta T}{T - \Delta T} F(s) + C \tag{1.97}$$

where T is the temperature of the system. Thus in the long time limit the method samples an ensemble at temperature $T + \Delta T$, rather than the actual temperature T. When $\Delta T = 0$ then the bias potential $V(\boldsymbol{s}, t \to \text{inf})$ is constant, and the method is equal to a standard MD simulation. When $\Delta T \to \text{inf}$ then the bias potential $V(\boldsymbol{s}, t \to \text{inf}) \to -F(s) + C$ which equates to the standard metadynamics method.

1.7.2 Methods that change the underlying kinetic energy

Replica Exchange Molecular Dynamics

Replica exchange MD developed by Sugita and Okamoto [7] is an enhanced sampling method that falls into the category of biasing the underlying Kinetic Energy of the system.

The methods covered in this study that generate an equilibrium sampling and modify the underlying kinetic energy all uses a Markov Chain Monte Carlo methodology with the Metropolis-Hastings acceptance criterion.

In REMD the state is made up of M replicas of the molecular system simulated at

different temperatures. Each replica's position and momentum vectors evolve via an NVT MD simulation. Since each replica is simulated at a different temperature, there is always only one replica at each temperature. Hence the replica and the temperatures form a one-to-one mapping. For instance, if there are M number of replicas, then there will be M number of corresponding temperatures. If the replicas and temperatures are indexed as $i = 1, ..., M$ and $m = 1, ..., M$ respectively, one particular mapping can be represented as:

$$\begin{pmatrix} 1 & 2 & 3 & ... & M \\ 1 & 2 & 3 & ... & M \end{pmatrix} \tag{1.98}$$

If the first row corresponds to the temperature indices and the second row to the replica indices, then this particular mapping indicates that the first replica is simulated at the first temperature, the second replica at the second temperature etc. If we want to represent the second replica to be simulated at the third temperature, and the third replica to be simulated at the second temperature, with all the other replicas mapped to the temperature the same as in Eq-1.98, the representation would be as follows:

$$\begin{pmatrix} 1 & 2 & 3 & ... & M \\ 1 & 3 & 2 & ... & M \end{pmatrix} \tag{1.99}$$

more generically any mapping of temperatures to replicas can be represented with a permutation function f. Thus the replicas are defined as a permutation function of temperatures.

$$f = \begin{pmatrix} 1 & 2 & 3 & ... & M \\ f(1) & f(2) & f(3) & ... & f(M) \end{pmatrix} \tag{1.100}$$

Where $f(m)$ would represent the replica at temperature m. Thus the state can be defined as a vector $X = (x_1^{f[1]}, ..., x_M^{f[M]})$ where the superscript and subscript refers to replica and temperature respectively, f is the permutation function of the temperatures, which identifies the replicas and $x = (q, p)$ encapsulates the positions and momentum of the system.

Each replica follows a canonical ensemble as they are simulated in an NVT simulation. Since replicas are independent of each other, the ensemble of state X that capture all replicas at once will be the product of the distribution of each replica Eq-1.101

$$P(X) = P(x_1^{[f(1)]})P(x_2^{[f(2)]})...P(x_M^{[f(M)]}) \sim \exp\left(-\sum_i^M \beta_i H(q^{[f(i)]}, p^{[f(i)]})\right) \tag{1.101}$$

Where the Hamiltonian, $H(q, p)$ is the sum of kinetic, $K(p)$ and potential, $U(q)$ energy.

The replicas are simulated in parallel at different temperatures. In regular intervals, a replica is randomly chosen and is swapped with another replica closest in temperature. The swap in the replica can be represented as a change in the state $X \rightarrow X'$, which correspond to changing the permutation function from f to f'

$$X = (..., x_m^{[f(m)]}, ..., x_n^{[f(n)]}, ...) \rightarrow X' = (..., x_m^{[f'(m)]'}, ..., x_n^{[f'(n)]'}, ...) \tag{1.102}$$

Where the f and f' permutation functions are different only at temperatures m and n as follows

$$\begin{cases} f(m) = i \rightarrow f'(m) = j \\ f(n) = j \rightarrow f'(n) = i \end{cases} \tag{1.103}$$

by substituting the permutation function into Eq-1.102, we obtain

$$X = (..., x_m^{[i]}, ..., x_n^{[j]}, ...) \rightarrow X' = (..., x_m^{[j]'}, ..., x_n^{[i]'}, ...) \tag{1.104}$$

When the replicas are swapped, the momentum of the replicas needs to be adjusted to reflect the new temperature. That is why in Eq-1.102 and Eq-1.104 the replica states within X' is represented as x' to denote that the momentum components are changed. More explicitly the change of momentum after the exchange of replica i, j is shown in Eq-1.105,

$$\begin{cases} x_m^{[i]} = (q^{[i]}, p^{[i]})_m \rightarrow x_n^{[i]'} = (q^{[i]}, p^{[i]'})_n \\ x_n^{[j]} = (q^{[j]}, p^{[j]})_n \rightarrow x_m^{[j]'} = (q^{[j]}, p^{[j]'})_m \end{cases} \tag{1.105}$$

In REMD the following momentum assignments are used, when the replicas are swapped to a different temperature.

$$\begin{cases} p^{[i]'} \rightarrow \sqrt{\dfrac{T_n}{T_m}} p^{[i]} \\ p^{[j]'} \rightarrow \sqrt{\dfrac{T_m}{T_n}} p^{[j]} \end{cases} \tag{1.106}$$

The new state is then accepted or rejected based on the metropolis acceptance criterion.

$$A(X, X') = min \left\{ 1, \frac{P(X')Q(X' \rightarrow X)}{P(X)Q(X \rightarrow X')} \right\} \tag{1.107}$$

where $P(X)$ is the probability of state X as given in Eq-1.101. $Q(X \rightarrow X')$ denotes the proposal probability of changing the state from $X \rightarrow X'$. The mechanism of changing the states are done through permutations, where only one replica is exchanged with another. This mechanism is symmetric. Thus the proposal probability from $X \rightarrow X'$ and $X' \rightarrow X$ is equal, hence the ratio of the proposal probabilities is one.

$$\frac{Q(X' \to X)}{Q(X \to X')} = 1 \tag{1.108}$$

Therefore the acceptance criterion can be written as the ratio of the state probabilities

$$A(X, X') = min\left\{1, \frac{P(X')}{P(X)}\right\} \tag{1.109}$$

If we substitute eq-1.101, into the acceptance criterion relation. And expand the hamiltonian as the sum of kinetic and potential energy. We obtain,

$$\frac{P(X')}{P(X)} = exp\left(\begin{array}{c} -\beta_m\left[U(q^{[j]}) + K(p^{[j]'})\right] - \beta_n\left[U(q^{[i]}) + K(p^{[i]'})\right] \\ +\beta_m\left[U(q^{[i]}) + K(p^{[i]})\right] + \beta_n\left[U(q^{[j]}) + K(p^{[j]})\right] \end{array}\right) \tag{1.110}$$

Note that the hamiltonians of states with temperatures other than m and n cancel out. Since the momentums are transformed according to eq-1.106, the kinetic energy of p' can be written in terms of the kinetic energies of p.

$$\begin{cases} K(p^{[i]'}) = \dfrac{T_n}{T_m}K(p^{[i]}) = \dfrac{\beta_m}{\beta_n}K(p^{[i]}) \\ K(p^{[j]'}) = \dfrac{T_m}{T_n}K(p^{[j]}) = \dfrac{\beta_n}{\beta_m}K(p^{[j]}) \end{cases} \tag{1.111}$$

where $\beta = 1/k_B T$, with k_B being the Boltzmann constant. By substituting eq-1.111 into eq-1.110

$$\frac{P(X')}{P(X)} = exp\left(\begin{array}{c} -\beta_m\left[U(q^{[j]}) + \dfrac{\beta_n}{\beta_m}K(p^{[j]})\right] - \beta_n\left[U(q^{[i]}) + \dfrac{\beta_m}{\beta_n}K(p^{[i]})\right] \\ +\beta_m\left[U(q^{[i]}) + K(p^{[i]})\right] + \beta_n\left[U(q^{[j]}) + K(p^{[j]})\right] \end{array}\right) \tag{1.112}$$

By reorganising Eq-1.112, we can see that the kinetic energies cancel out, and we obtain

$$\frac{P(X')}{P(X)} = exp\left(-[\beta_n - \beta_m](U(q^{[i]}) - U(q^{[j]}))\right) \tag{1.113}$$

by substituting Eq-1.113 back into the acceptance criterion, Eq-1.109, we obtain

$$A(X, X') = min\{1, \exp(-\Delta)\} \tag{1.114}$$

where Δ,

$$\Delta = [\beta_n - \beta_m](U(q^{[i]}) - U(q^{[j]})) \tag{1.115}$$

The exchange of replicas of different temperatures allows configurations at lower

temperatures to be simulated at higher temperatures where it is more probable to overcome an energy barrier. The new sampled configuration can then swapped back to the lower temperature simulations, hence enhancing the configurational sampling.

Variations/extensions of Replica Exchange Molecular Dynamics can be found in literature. [54], [55]

1.7.3 Digitally Filtered MD - (DFMD)

The DFMD method developed by Phillips et al. [46], scales the velocity components of selected atoms according to a filter response to enhance or suppress specific vibrational frequencies in the molecular system during an MD simulation and can be used to overcome energy barriers in a molecule to enhance the sampling. The method combines the theory of Molecular Dynamics with Digital Filtering explained in the previous sections. The filter is applied via Eq-1.70 in the form of Eq-1.116 to a buffer filled with velocity components of selected atoms, $B = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_{|B|}\}$. The buffer is filled by NVE MD simulation, with a number of steps equal to the number of filter coefficients. Thus $\boldsymbol{v_i} = (v_i(t_0), v_i(t_1), ..., v_i(t_{2m}))$ is an ordered list where $2m+1$ is equal to the number of filter coefficients, $\boldsymbol{c} = (c_{-m}, c_{-m+1}, ..., c_0, ..., c_{m-1}, c_m)$. The filter type used in this method is called a linear phase filter which has the property of having symmetrical filter coefficients, $c_i = c_{-i}$. This symmetry of filter coefficients ensures that the filter response has no imaginary component and the relative phase of the data point in the middle, the phase of $v_i(t_m)$ is the same as the phase of the filtered velocity at the $(m+1)^{th}$ time step, $v_i^{filtered}(t_m)$.

$$v_k^{filtered}(t_m) = \sum_{i=-m}^{m} c_i v_k(t_{i+m}), \quad k = 1, 2, ..., |B| \tag{1.116}$$

For instance, assume the desired filter response has been obtained accurate enough by using $2m+1 = 2001$, filter coefficients. Then to apply the filter, an NVE MD simulation with 2001 steps will be carried out where at every step the velocity components of the selected atoms are stored in the buffer. The coordinates of each atom are also stored during the MD simulation. The filter is applied to the buffer via Eq-1.116 from which we obtain the filtered velocity at the $(m+1)^{th} = 1001^{th}$ time step of the buffer filling MD simulation. A normal MD simulation which continues until the next filter application is then employed, with the starting velocities set to the filtered velocities. To ensure the velocities and the positions are in phase, the positions at the 1001^{th} time step are used as the initial conditions.

It turns out that DFMD has difficulties in targeting low-frequency vibrations, in systems with rapid energy dissipation like in condensed phase systems [46]. To overcome the problem in DFMD a method called Reversible Digitally Filtered Molecular Dynamics or in short RDFMD has been developed by Phillips et al. [1]. The RDFMD method differs from DFMD in the way of filling the buffer velocities for the second and suc-

cessive filter applications. In DFMD after a filter application to apply the next filter, a forward MD simulation is done to fill the next buffer. This leads to having a fixed number of MD steps present between two successive filter applications determined by the number of filter coefficients. Thus in systems where the energy dissipation takes fewer time steps than the half of the number of filter coefficients, $(2m+1)/2$, the introduced energy by the filter will be dissipated before the next filter is applied. This, on the other hand, restricts the multiple filter applications to introduce energy to the system gradually, which for instance could be important to observe a smooth conformational transition. The RDFMD method tackles this issue by filling the buffer with backwards and forwards MD simulations instead of just doing forward MD simulation. Thus, the time to which the filtered velocities correspond can be adjusted by changing the number of forwards and backwards MD steps. This, in turn, will enable the adjustment of the number of MD steps to be present between successive filter applications enabling the application of a filter before the energy introduced by a previous filter dissipates.

It has to be noted that both DFMD and RDFMD disturb the equilibrium distribution, $P(\boldsymbol{q})$ of the system by the filter application. Thus even though an enhanced configurational sampling can be achieved, thermodynamic averages cannot be calculated correctly. Thus in order to sample from an equilibrium distribution and still benefit from the digital filtering, a method called Digitally Filtered Hybrid Monte Carlo in short DFHMC is developed in this study. This method combines the notion of digital filtering with the theory of Hybrid Monte Carlo. Thus, we will start the next subsection by introducing the Hybrid Monte Carlo method and then extending it to explain the DFHMC method.

1.7.4 Hybrid Monte Carlo - (HMC)

Hybrid Monte Carlo (HMC) first proposed for simulation of lattice field theory by Duane et al. [8] was later applied to molecular systems by Mehlig et al. [56].
HMC is a method to generate a sequence of configurations/states following a target distribution, $P(\boldsymbol{q})$ by using the Metropolis-Hastings method, where the proposed configurations are obtained via molecular dynamics steps. Since molecular dynamics in principle is deterministic, a pair of configuration and velocity vector, $(\boldsymbol{x}, \boldsymbol{v})$ will determine uniquely the configuration at a later step. Therefore the proposal probability, $Q(\boldsymbol{q}^{'}|\boldsymbol{q})$ will be equal to the probability of getting the velocity, $\boldsymbol{v}$ such that the pair, $(\boldsymbol{q}, \boldsymbol{v})$ will go to the proposed configuration, $\boldsymbol{q}^{'}$ via molecular dynamics for a certain MD step.

$$Q(\boldsymbol{q}^{'}|\boldsymbol{q}) = P_{vel}(\boldsymbol{v}) \tag{1.117}$$

The propability distribution of the velocity vector, $\boldsymbol{v} = (v_1, v_2, ..., v_{3N})$, where N denotes the number of atoms in the system, is mostly chosen to be a multivariate

normal distribution, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean, $\boldsymbol{\mu} = 0$ and covariance matrix, $\boldsymbol{\Sigma} = k_B T \boldsymbol{M}^{-1}$

$$P_{vel}(\boldsymbol{v}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{W} \exp\left(-\frac{\boldsymbol{v}^T \boldsymbol{M} \boldsymbol{v}}{2k_B T}\right) \tag{1.118}$$

where W is the normalisation factor making the integral of the probability distribution function over the whole space equal to 1. k_B is the Boltzmann constant, T is the temperature of the system and $\boldsymbol{M}$ is the diagonal $3Nx3N$ mass matrix,

$$\boldsymbol{M}_{(3Nx3N)} = \begin{pmatrix} m_1 & 0 & 0 & . & . & . & . & . & . & 0 \\ 0 & m_1 & 0 & . & . & . & . & . & . & 0 \\ 0 & 0 & m_1 & . & . & . & . & . & . & 0 \\ 0 & 0 & 0 & m_2 & . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & m_2 & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & m_2 & . & . & . & 0 \\ & . & & . & & & . & & & \\ & . & & . & & & . & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & . & m_N & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & . & . & m_N & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & . & . & . & m_N \end{pmatrix} \tag{1.119}$$

where, m_i is the mass of the i^{th} atom, $i = 1, 2, ..., N$
Since $\boldsymbol{M}$ is a diagonal matrix,

$$\frac{1}{2}\boldsymbol{v}^T \boldsymbol{M} \boldsymbol{v} = \frac{1}{2} \sum_{i=1}^{3N} M_{ii} {v_i}^2 \tag{1.120}$$

which is equal to the total kinetic energy, $E(\boldsymbol{v})$. Therefore Eq-1.118 can be rewritten as,

$$P(\boldsymbol{v}) = \exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right) \tag{1.121}$$

Furthermore, the joint probability of the atom's velocities can be written as the product of the probabilities of each atom's velocity component as a result of $\boldsymbol{M}$ being diagonal

$$\begin{aligned} P_{vel}(\boldsymbol{v}) &= P_{vel}(v_1, v_2, ..., v_{3N}) \\ &= P_{vel}(v_1) P_{vel}(v_2) ... P_{vel}(v_{3N}) \end{aligned} \tag{1.122}$$

Thus each atom's velocity component in a molecular system can be obtained independently via,

$$P_{vel}(v_i) = \frac{1}{W_i} \exp\left(-\frac{M_{ii} v_i^2}{2k_B T}\right), \quad i = 1, 2, ..., 3N \tag{1.123}$$

Given that the target distribution is, $P(\boldsymbol{q})$, and the proposal distribution, $P(\boldsymbol{v})$ is as in Eq-1.121, then the Metropolis-Hasting acceptance criterion, Eq-1.57 can be written as,

$$A(\boldsymbol{q}^{'}, \boldsymbol{q}) = min\left\{1, \frac{P(\boldsymbol{q}^{'})\exp\left(-\frac{E(\boldsymbol{v}^{'})}{k_B T}\right)}{P(\boldsymbol{q})\exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right)}\right\} \tag{1.124}$$

For a canonical ensemble the target distribution, $P(\boldsymbol{q})$ is

$$P(\boldsymbol{q}) = \frac{1}{Z}\exp\left(-\frac{U(\boldsymbol{q})}{k_B T}\right) \tag{1.125}$$

Hence Eq-1.124, will be,

$$A(\boldsymbol{q}^{'}, \boldsymbol{q}) = min\left\{1, \frac{\exp\left(-\frac{U(\boldsymbol{q}^{'})}{k_B T}\right)\exp\left(-\frac{E(\boldsymbol{v}^{'})}{k_B T}\right)}{\exp\left(-\frac{U(\boldsymbol{q})}{k_B T}\right)\exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right)}\right\} \tag{1.126}$$

Noting that $U(\boldsymbol{q}) + E(\boldsymbol{v}) = H(\boldsymbol{q}, \boldsymbol{v})$, where H is the total energy of the system, Eq-1.126 becomes,

$$A(\boldsymbol{q}^{'}, \boldsymbol{q}) = min\left\{1, \exp\left(-\frac{\Delta H}{k_B T}\right)\right\} \tag{1.127}$$

Where $\Delta H = H(\boldsymbol{q}^{'}, \boldsymbol{v}^{'}) - H(\boldsymbol{q}, \boldsymbol{v})$

In addition, it can be noted that the normalisation factor of the target distribution, Z and of the velocity distribution, W cancels out in the acceptance criterion. Thus knowing the target distribution up to a multiplicative constant is enough to sample from the exact target distribution.

A pseudo code to perform Hybrid Monte Carlo can be seen in Algorithm-2

1.7.5 Digitally Filtered HMC - (DFHMC)

The DFHMC method, which is proposed in this study as another way to generate random variates from a target distribution, with a special emphasis on molecular systems, uses the method of digital filtering and HMC. The method tries to bias the velocities such that the MD updates are more likely to overcome a high energy barrier relative to the system's average kinetic energy. Such a biased velocity approach used in conjunction with HMC called Momentum Enhanced Hybrid Monte Carlo (MEHMC) was developed by Andricioaei et al. [2].

The only theoretical difference between the HMC and the velocity biased methods DFHMC and MEHMC is in the definition of the probability of the velocity vector, $P_{vel}(\boldsymbol{v})$.

Recall that in the HMC method, $P_{vel}(\boldsymbol{v})$ was equal to a multivariate normal distribution and could be written as a product of the probabilities of each atom's velocity compo-

Algorithm 2 Hybrid Monte Carlo. Hybrid Monte Carlo

$n \leftarrow 0$
$x^0 \leftarrow$ Starting Configuration
while $n < \#Steps$ **do**
    Generate $\boldsymbol{v} \sim \mathcal{N}(0, k_B T \boldsymbol{M}^{-1})$
    $H \leftarrow \frac{1}{2}\boldsymbol{v^T M v} + U(\boldsymbol{x^n})$
    $\boldsymbol{x}', \boldsymbol{v}' \leftarrow MDupdate(\boldsymbol{x^n}, \boldsymbol{v}, \#MDSteps)$
    $H' \leftarrow \frac{1}{2}(\boldsymbol{v}')^{\boldsymbol{T}} \boldsymbol{M v}' + U(\boldsymbol{x}')$
    $\Delta H \leftarrow H' - H$
    $accept \leftarrow \exp\left(-\frac{\Delta H}{k_B T}\right)$
    $X \sim U(0, 1)$
    if $X < accept$ **OR** $accept \geq 1$ **then**
        $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x}'$
    else
        $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x^n}$
    end if
    $n \leftarrow n + 1$
end while

nent, $P_{vel}(v_i)$ for $i = 1, 2..., 3N$, which were defined in Eq-1.123.
In the MEHMC and DFHMC methods, the probabilities of the velocity vector are given by a multivariate bimodal distribution with a mathematical form as seen below,

$$P_{vel}(\boldsymbol{v}) = \frac{1}{W}\left[\exp\left(-\sum_{i=1}^{3N}\frac{m_i(v_i - b_i)^2}{2k_B T}\right) + \exp\left(-\sum_{i=1}^{3N}\frac{m_i(v_i + b_i)^2}{2k_B T}\right)\right], \quad i = 1, 2, ..., 3N \tag{1.128}$$

In principle, a single-variate bimodal distribution can also be used for the probabilities $P_{vel}(v_i)$ for $i = 1, 2..., 3N$.

$$P_{vel}(v_i) = \frac{1}{W_i}\left[\exp\left(-\frac{m_i(v_i - b_i)^2}{2k_B T}\right) + \exp\left(-\frac{m_i(v_i + b_i)^2}{2k_B T}\right)\right], \quad i = 1, 2, ..., 3N \tag{1.129}$$

The single-variate distribution $P_{vel}(v_i)$ is essentially a sum of two normal distributions with the same standard deviation but different mean values, one having a mean value $-b_i$ and the other the value b_i. The graph of the distribution can be seen in Fig-1.10. A bimodal distribution introduces the parameter b_i, which allows adjusting the location of the modes. It thus can be used to increase or decrease the most probable velocity value for each degree of freedom separately. In the DFHMC method, the b_i parameters of the velocity distribution are obtained by a digital filter application based on a pre-designed filter response. To enhance sampling, the filter is designed to filter

out high frequency vibration from the velocity trajectory. Thus only degrees of freedom with low frequency vibration will have a non zero filtered velocity value. Then to increase the velocity along the degrees of freedom with low frequency vibration, the filtered velocities are used as the b_i parameters of the velocity distribution. The symmetrical shape of the bimodal distribution results in increasing velocities in both positive and negative direction with equal probabilities.

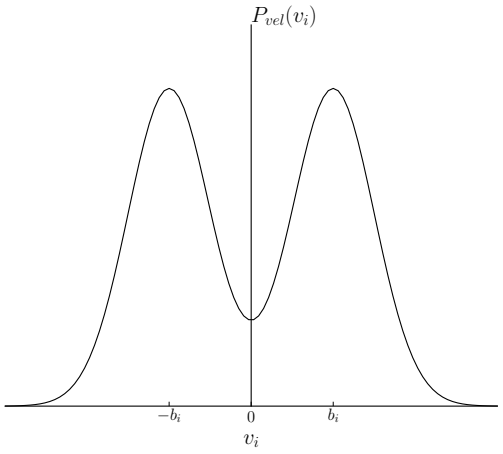


Figure 1.10 **Bimodal Distribution.** The velocity propability distribution for the i^{th} velocity degree of freedom.

Depending on the velocity distribution used (single or multivariate) the corresponding acceptance criteria needs to be applied to maintain the detailed balance or, in other words, the thermodynamic equilibrium.
Before deriving the acceptance criteria, let us reformulate the velocity distributions to obtain terms that will make the acceptance criteria look more similar to the ones from the HMC method.

Reformulating the single-variate velocity distribution:

The Eq-1.129 can also be written as,

$$P_{vel}(v_i) = \frac{1}{W_i} \exp\left(-\frac{m_i v_i^2}{2k_B T}\right)\left[\exp\left(\frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i b_i}{k_B T}\right)\right] \tag{1.130}$$

The joint probability of the velocity vector, $P_{vel}(\boldsymbol{v})$ will be taken as the product of $P_{vel}(v_i)$ for $i = 1, 2, ...3N$

$$\begin{aligned} P_{vel}(\boldsymbol{v}) &= \prod_{i=1}^{3N} P_{vel}(v_i) \\ &= \prod_{i=1}^{3N} \exp\left(-\frac{m_i v_i^2}{2k_B T}\right) \prod_{i=1}^{3N} P_b(v_i) \end{aligned} \tag{1.131}$$

where,

$$P_b(v_i) = \frac{1}{W_i}\left[\exp\left(\frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i b_i}{k_B T}\right)\right] \tag{1.132}$$

The first term of the product, $P_{vel}(\boldsymbol{v})$ in Eq-1.131, can be written as,

$$\begin{aligned} \prod_{i=1}^{3N} \exp\left(-\frac{m_i v_i^2}{2k_B T}\right) &= \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i^2}{2k_B T}\right) \\ &= \exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right) \end{aligned} \tag{1.133}$$

where $E(\boldsymbol{v})$ is the total kinetic energy.
Thus the single-variate velocity distribution can be written as;

$$P(\boldsymbol{v}) \sim \exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right) \prod_{i=1}^{3N} \left[\exp\left(\frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i b_i}{k_B T}\right)\right] \tag{1.134}$$

Reformulating the multi-variate velocity distribution:

The Eq-1.128 can be written as,

$$P_{vel}(\boldsymbol{v}) = \frac{1}{W} \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i^2}{2k_B T}\right) \left[\exp\left(\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right)\right] \tag{1.135}$$

As in Eq-1.133 the first exponential term in the above equation (Eq-1.135), can be written in terms of the total kinetic energy, $E(\boldsymbol{v})$. Thus the multi-variate velocity distribution can be written as;

$$P_{vel}(\boldsymbol{v}) \sim \exp\left(-\frac{E(\boldsymbol{v})}{k_B T}\right) \left[\exp\left(\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right)\right] \tag{1.136}$$

Note that both in the single and multi-variate distributions we wrote the final reformulated velocity distribution as a proportion and left out the normalisation constant, W. This is because of the normalisation constant or any multiplicative constant to the velocity distribution will cancel out when carrying out the acceptance criteria via the Metropolis-Hasting Algorithm.

Thus, for a target distribution that generates the canonical configurational ensemble the Metropolis-Hastings acceptance criterion can be written as follows;
For the Single-Variate Distribution

$$A(\boldsymbol{q}^{'}, \boldsymbol{q}) = min\left\{1, \exp\left(-\frac{\Delta H}{k_B T}\right) \prod_{i=1}^{3N} \left[\frac{\exp\left(\frac{m_i v_i^{'} b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i^{'} b_i}{k_B T}\right)}{\exp\left(\frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i b_i}{k_B T}\right)}\right]\right\} \quad (1.137)$$

For the Multi-Variate Distribution

$$A(\boldsymbol{q}^{'}, \boldsymbol{q}) = min\left\{1, \exp\left(-\frac{\Delta H}{k_B T}\right) \frac{\exp\left(\sum_{i=1}^{3N} \frac{m_i v_i^{'} b_i}{k_B T}\right) + \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i^{'} b_i}{k_B T}\right)}{\exp\left(\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\sum_{i=1}^{3N} \frac{m_i v_i b_i}{k_B T}\right)}\right\} \quad (1.138)$$

where $\Delta H = H(\boldsymbol{q}^{'}, \boldsymbol{v}^{'}) - H(\boldsymbol{q}, \boldsymbol{v})$, the change in the total energy. Note that for $\boldsymbol{b} = (b_1, b_2, ..., b_{3N}) = 0$, the velocity biased method will recover the acceptance criteria given by the HMC method as in Eq-1.127.

So far the DFHMC and MEHMC methods are the same. The difference in those methods are in the way how they determine the bias vector, $\boldsymbol{b}$ in the velocity distribution function (Eq-1.134, Eq-1.136). The value of b_i, in the MEHMC method, corresponds to the average velocities (momenta) taken over a specified past MD step. Depending on the MD step used the averaging procedure can be thought as a simple low pass filter, which results in a higher b_i value for atoms with low-frequency motion.

In DFHMC the $\boldsymbol{b}$ also corresponds to some kind of filtered velocity but instead of an arithmetic average, a linear combination of the velocities with predefined weights is used to obtain a finer way of filtering frequencies. The digital filter as in Eq-1.70 is used for DFHMC in the following form,

$$b_i = \sum_{j=-m}^{m} c_j v_i(t_{j+m}), \quad i = 1, 2, ..., 3N \quad (1.139)$$

where the set of velocities, $\{v_i(t_0), v_i(t_1), v_i(t_2), ...v_i(t_{2m})\}$ are obtained by an MD simulation with $2m + 1$ number of steps. The filtered velocity, b_i calculated via Eq-1.139 is the filtered velocity at the m^{th} step of the $2m + 1$ step MD simulation. The $2m+1$ number of coefficients are chosen such that, they create the desired frequency response as explained in the digital filter section. Although any kind of filter response can be used, to accomplish an enhanced sampling over the configuration space, a filter response that enhances the low frequency motions is preferred.

A pseudo code to perform digitally filtered Hybrid Monte Carlo using the single vari-

ate and multi variate velocity distribution can be seen in Algorithm-3 and Algorithm-4 respectively.

Algorithm 3 DFHMC - Single Variate. Digitally Filtered HMC algorithm-1: Single-variate distribution

$x^0 \leftarrow$ Starting Configuration
$n \leftarrow 0$
while $n < \#Steps$ **do**
  if $n \mod T_{filter}$ **then**
    Generate $\boldsymbol{v} \sim \mathcal{N}(0, k_B T \boldsymbol{M}^{-1})$
    $\boldsymbol{b} \leftarrow FilterVelocities(\boldsymbol{v}, \boldsymbol{x^n}, \boldsymbol{c})$
  end if
  Generate $\boldsymbol{v} \sim \prod_{i=1}^{3N} \mathcal{B}(b_i, k_B T / M_{ii})$
  $H \leftarrow \frac{1}{2}\boldsymbol{v^T M v} + U(\boldsymbol{x^n})$
  $\boldsymbol{x}', \boldsymbol{v}' \leftarrow MDupdate(\boldsymbol{x^n}, \boldsymbol{v}, \#MDSteps)$
  $H' \leftarrow \frac{1}{2}(\boldsymbol{v}')^{\boldsymbol{T}} \boldsymbol{M v}' + U(\boldsymbol{x}')$
  $\Delta H \leftarrow H' - H$
  $accept \leftarrow \exp\left(-\frac{\Delta H}{k_B T}\right) \prod_{i=1}^{3N} \left[\frac{\exp\left(\frac{m_i v_i' b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i' b_i}{k_B T}\right)}{\exp\left(\frac{m_i v_i b_i}{k_B T}\right) + \exp\left(-\frac{m_i v_i b_i}{k_B T}\right)} \right]$
  $X \sim U(0, 1)$
  if $X < accept$ **OR** $accept \geq 1$ **then**
    $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x}'$
  else
    $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x^n}$
  end if
  $n \leftarrow n + 1$
end while

1.7.6 Riemannian Manifold HMC - (RMHMC)

In Riemannian manifold HMC [3] the mass matrix is made dependent on the positions of the system, $M = G(q)$, where M or G is a $3N \times 3N$ matrix with N being the number of atoms in the system. The kinetic energy in the Riemannian manifold is given as follows,

$$E(p) = \frac{1}{2} p^T G(q)^{-1} p + \frac{1}{2} \log\left((2\pi)^D |G(q)|\right) \tag{1.140}$$

The second term in the kinetic energy ensures that the marginal distribution $P(q) \sim \exp\{-U(q)\}$ follows the NVT ensemble distribution of the positions. The Hamiltonian of the systems is:

$$H(q, p) = U(q) + \frac{1}{2} \log\Big((2\pi)^{3N} |G(q)|\Big) + \frac{1}{2} p^T G(q)^{-1} p \tag{1.141}$$

Algorithm 4 DFHMC - Multi variate. Digitally Filtered HMC algorithm-2: Multivariate distribution

$x^0 \leftarrow$ Starting Configuration
$n \leftarrow 0$
while $n < \#Steps$ **do**
  if $n \mod T_{filter}$ **then**
    Generate $\boldsymbol{v} \sim \mathcal{N}(0, k_B T \boldsymbol{M}^{-1})$
    $\boldsymbol{b} \leftarrow FilterVelocities(\boldsymbol{v}, \boldsymbol{x^n}, \boldsymbol{c})$
  end if
  Generate $\boldsymbol{v} \sim \mathcal{B}(\boldsymbol{b}, k_B T / \boldsymbol{M}^{-1})$
  $H \leftarrow \frac{1}{2}\boldsymbol{v^T M v} + U(\boldsymbol{x^n})$
  $\boldsymbol{x}', \boldsymbol{v}' \leftarrow MDupdate(\boldsymbol{x^n}, \boldsymbol{v}, \#MDSteps)$
  $H' \leftarrow \frac{1}{2}(\boldsymbol{v}')^{\boldsymbol{T}} \boldsymbol{M v}' + U(\boldsymbol{x}')$
  $\Delta H \leftarrow H' - H$
  $accept \leftarrow \exp\left(-\frac{\Delta H}{k_B T}\right) \frac{\exp\left(\frac{(\boldsymbol{v}')^{\boldsymbol{T}} \boldsymbol{Mb}}{k_B T}\right) + \exp\left(-\frac{(\boldsymbol{v}')^{\boldsymbol{T}} \boldsymbol{Mb}}{k_B T}\right)}{\exp\left(\frac{\boldsymbol{v^T Mb}}{k_B T}\right) + \exp\left(-\frac{\boldsymbol{v^T Mb}}{k_B T}\right)}$
  $X \sim U(0, 1)$
  if $X < accept$ **OR** $accept \geq 1$ **then**
    $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x}'$
  else
    $\boldsymbol{x^{n+1}} \leftarrow \boldsymbol{x^n}$
  end if
  $n \leftarrow n + 1$
end while

To calculate the marginal distribution of q, i.e. $P(q)$, the integral of the joint distribution $P(q,p)$ is taken with respect to the momentum p.

$$\begin{aligned} P(q) &\sim \int \exp\{-H(q,p)\}dp \\ &= \exp\{-U(q)\}\exp\left\{-\frac{1}{2}\log\left((2\pi)^{3N}|G(q)|\right)\right\}\int \exp\left\{\frac{1}{2}p^T G(q)^{-1}p\right\}dp \\ &= \exp\{-U(q)\}\frac{1}{\sqrt{(2\pi)^{3N}|G(q)|}}\int \exp\left\{\frac{1}{2}p^T G(q)^{-1}p\right\}dp \end{aligned} \tag{1.142}$$

since the Gaussian integral is equal to $\sqrt{(2\pi)^{3N}|G(q)|}$ the term below is equal to 1

$$\frac{1}{\sqrt{(2\pi)^{3N}|G(q)|}}\int \exp\left\{\frac{1}{2}p^T G(q)^{-1}p\right\}dp = 1 \tag{1.143}$$

Therefore the marginal distribution of q, is equal to the NVT ensemble distribution.

$$P(q) \sim \exp\{-U(q)\} \tag{1.144}$$

The dynamics of the system according to Hamilton's equation q-1.16 is defined by,

$$\begin{aligned} \frac{d\boldsymbol{p}}{dt} &= -\frac{\partial H}{\partial \boldsymbol{q}} \\ &= -\frac{dU(q)}{dq} - \frac{1}{2}G(q)^{-1}\frac{dG(q)}{dq} + \frac{1}{2}p^T G(q)^{-1}\frac{dG(q)}{dq}G(q)^{-1}p \end{aligned} \tag{1.145}$$

$$\begin{aligned} \frac{d\boldsymbol{q}}{dt} &= \frac{\partial H}{\partial \boldsymbol{p}} \\ &= G(q)^{-1}p \end{aligned} \tag{1.146}$$

Since the Hamiltonian of the Riemannian manifold is not separable as a sum of a function of q and p the Hamilton equations can be solved numerically via the Generalized Verlet Integrator explained in section 1.2.3.3

$$\begin{aligned} \boldsymbol{p}(t+\frac{dt}{2}) &= \boldsymbol{p}(t) - \frac{dt}{2}\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t),\boldsymbol{p}(t+\frac{dt}{2})) \\ \boldsymbol{q}(t+dt) &= \boldsymbol{q}(t) + \frac{dt}{2}\left[\frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t),\boldsymbol{p}(t+\frac{dt}{2})) + \frac{d}{d\boldsymbol{p}}H(\boldsymbol{q}(t+dt),\boldsymbol{p}(t+\frac{dt}{2}))\right] \\ \boldsymbol{p}(t+dt) &= \boldsymbol{p}(t+\frac{dt}{2}) - \frac{dt}{2}\frac{d}{d\boldsymbol{q}}H(\boldsymbol{q}(t+dt),\boldsymbol{p}(t+\frac{dt}{2})) \end{aligned} \tag{1.147}$$

Inputting the Derivatives of the Hamiltonian as given in Eq-1.145 and Eq-1.146 into the Generalized Verlet Integrator, Eq-1.147, the 3 step update given below is

obtained.

The first part is a half step momentum update, which needs to be solved self-consistently, i.e., given a starting momentum, the step is repeatedly evaluated either until the momentum does converge to a stable value or until a predefined fixed number of iterations have been done.

$$\begin{aligned}\boldsymbol{p}^{t+\frac{dt}{2}} = \boldsymbol{p}^{t} - \frac{dt}{2}\Bigg[&\frac{dU(\boldsymbol{q}^{t})}{d\boldsymbol{q}} + \frac{1}{2}G(\boldsymbol{q}^{t})^{-1}\frac{dG(\boldsymbol{q}^{t})}{d\boldsymbol{q}} \\ &- \frac{1}{2}(\boldsymbol{p}^{t+\frac{dt}{2}})^{T}G(\boldsymbol{q}^{t})^{-1}\frac{dG(\boldsymbol{q}^{t})}{d\boldsymbol{q}}G(\boldsymbol{q}^{t})^{-1}\boldsymbol{p}^{t+\frac{dt}{2}}\Bigg]\end{aligned} \tag{1.148}$$

The second part consists of a full step position update, which also needs to be solved self-consistently

$$\boldsymbol{q}^{t+dt} = \boldsymbol{q}^{t} + \frac{dt}{2}\left[G(\boldsymbol{q}^{t})^{-1}\boldsymbol{p}^{t+\frac{dt}{2}} + G(\boldsymbol{q}^{t+dt})^{-1}\boldsymbol{p}^{t+\frac{dt}{2}}\right] \tag{1.149}$$

The final part is again a half step momentum update but does not require to be solved self consistently.

$$\begin{aligned}\boldsymbol{p}^{t+dt} = \boldsymbol{p}^{t+\frac{dt}{2}} - \frac{dt}{2}\Bigg[&\frac{dU(\boldsymbol{q}^{t+dt})}{d\boldsymbol{q}^{t+dt}} + \frac{1}{2}G(\boldsymbol{q}^{t+dt})^{-1}\frac{dG(\boldsymbol{q}^{t+dt})}{d\boldsymbol{q}^{t+dt}} \\ &- \frac{1}{2}(\boldsymbol{p}^{t+\frac{dt}{2}})^{T}G(\boldsymbol{q}^{t+dt})^{-1}\frac{dG(\boldsymbol{q}^{t+dt})}{d\boldsymbol{q}^{t+dt}}G(\boldsymbol{q}^{t+dt})^{-1}\boldsymbol{p}^{t+\frac{dt}{2}}\Bigg]\end{aligned} \tag{1.150}$$

In contrast to the conventional velocity Verlet or leapfrog integrators, in the above integrator, the momentum and positions update is also affected by the change in the redefined mass matrix, G with respect to the positions.

Velocity effect on systems power spectrum

Chapter 2

2.1 Spectral Analysis

In the literature, it was shown that during an MD simulation resetting the atomic velocities to the filtered velocities obtained through digital filtering is able to put energy into the selected frequency range [46], [1]. However, this way of redistributing the energy makes the process non-equilibrium. Instead by using the filtered velocities as a parameter for a distribution where the velocities are drawn from, the process can be made equilibrium by using an acceptance-rejection criterion to form a Metropolis-Hastings algorithm.

Before applying the Metropolis-Hastings algorithm to form the DFHMC, we analyse whether drawing the velocities from a bimodal distribution with parameters as the filtered velocities can still put energy into the selected frequencies. Since the velocities will be drawn from a distribution, the direction of the drawn velocity vectors will not necessarily be the same. Therefore we would expect that not every time the energy will go into the selected frequency. Thus, the question we are interested in is whether the energy in the selected frequency can be increased by using the bimodal distribution with the filtered velocities as parameters compared to drawing the velocities from a normal distribution.

The systems that are used all have separated frequencies making it easy to target a specific frequency. Energy is introduced into the lowest frequency by designing a filter to remove all frequencies except the lowest one. The direction of the lowest frequency vibration is chosen to be different for each system to answer the following questions:

Can energy be introduced via a bimodal distribution

1. when the direction of the target vibration is along a cartesian coordinate
2. when the direction of the target vibration is along an angle
3. When the direction of the target vibration is along a dihedral angle

Essentially, the second and third questions investigate how the energy distributes when the vibrations are along a nonlinear combination of the cartesian coordinates.

2.1.1 DoubleWell

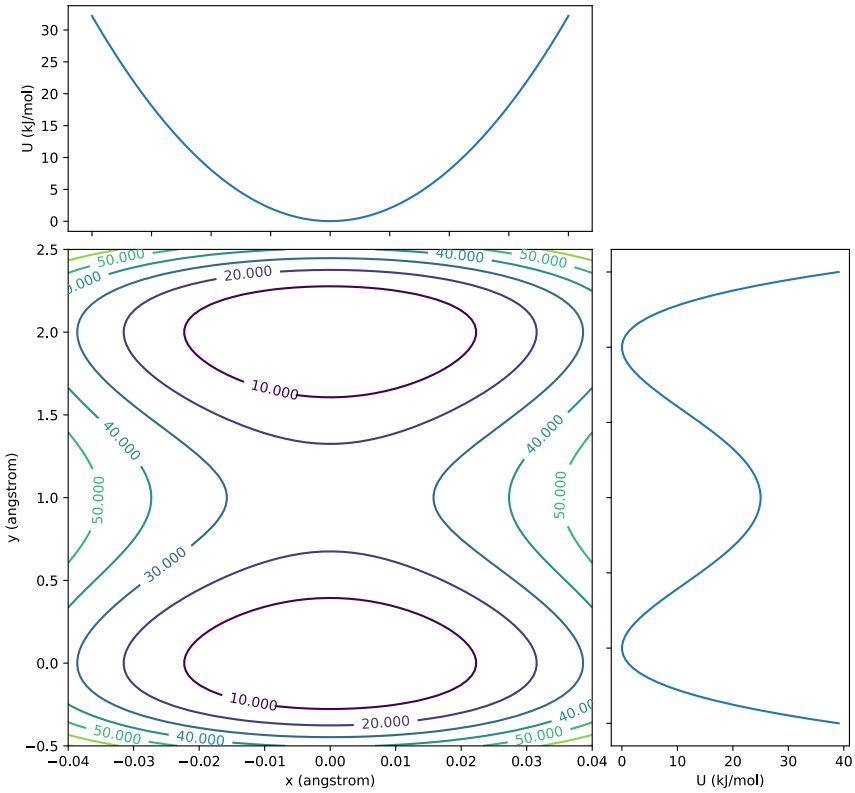


Figure 2.1 **Double Well potential surface along** x, y**.** The graph shows the equal potential energy lines, isoline graph of the double well energy surface. On the top and right side of the figure the projections of the surface on to the y and x axis are drawn.

The first and simplest system is the Double Well potential system Eq-2.1, and will be used further in the application chapter.

$$U(x, y, z) = K_{xz}(x^2 + z^2) + K_y y^2 (y - b)^2 \tag{2.1}$$

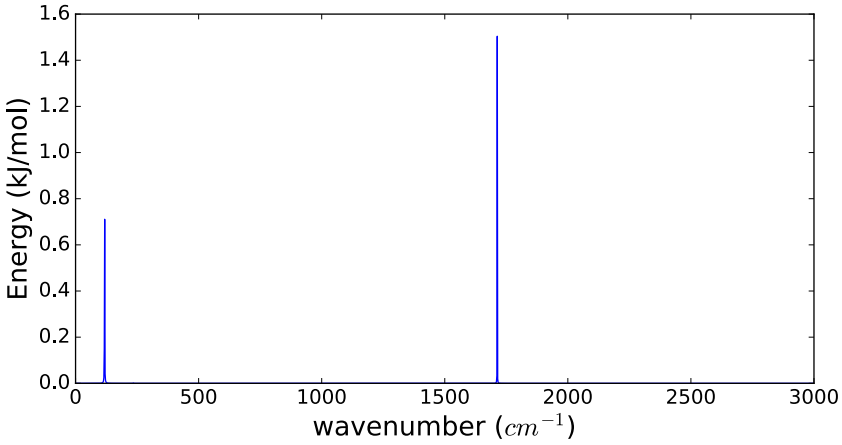
The parameters of the double well potantial are choosen to be as follows:

$K_{xz} = 40227.6254\ kJ/(molAngstrom^2)$, $K_y = 25\ kJ/(molAngstrom^4)$, $b = 2\ Angstrom$

The energy surface of the double well potential with the given parameters can be seen in Fig-2.1.

The system's vibrations are along the cartesian coordinates. As can be seen from the spectral density Fig-2.2, the low and high frequencies are well separated, and a low pass filter with the band being between $0 - 1000 cm^{-1}$ can be used. Note that

$1000cm^{-1}$ as the band end is used as one arbitrary option that is able to separate the low and high frequency. Thus any frequency that is between the low and high frequency can be used as the band end.



Fourier Spectrum of Double Well potential system. Frequencies are displayed in wavenumber units. MDStep = 10^4, dt= 2fs, temperature= 300K

Figure 2.2

Fig-2.3, shows the distribution of the percentage of energy that is in the selected frequency with respect to the total energy. The percentage is calculated from the spectral density by summing the energies between the selected frequency range, dividing by the total energy and multiplying by 100, Eq-2.2.

$$PF = 100x\frac{\sum_{i=0}^{f} energy(i)}{\sum_{i=0}^{f_{Nyquist}} energy(i)} \tag{2.2}$$

Since the starting velocities are drawn from a distribution and the molecular dynamics simulation conserves the total energy, the amplitudes of the spectral densities obtained from the MD simulation will vary depending on the drawn velocities. Therefore repeated simulations have been done to generate a histogram for the percentage energies. Fig-2.3 compares the histogram of the percentage of selected frequency range's energies for bimodal velocity distribution and the normal velocity distribution. As can be seen, when the filtered velocities are used as a parameter in the bimodal distribution, a significantly higher percentage of energy goes into the selected frequency region compared to using the normal distribution. An important parameter that influences the degree of increase of energy into the selected frequency range is the filter scaling parameter. After the filtered velocities are obtained, they can be scaled by a scalar, which will not change the direction of the filtered velocity vector but it would increase its magnitude. if the filtered velocities are small compared to the standard deviation of the modes that make up the bimodal distribution the directional

information would be lost in the drawn velocities. Thus to preserve the directional information the filtered velocities can be scaled such that the standard deviation is not big enough to merge the two modes in the bimodal distribution. In the case of the Double Well system, a scaling parameter of 2 has been used, which resulted in a significant increase in the percentage energy of the selected frequency range.

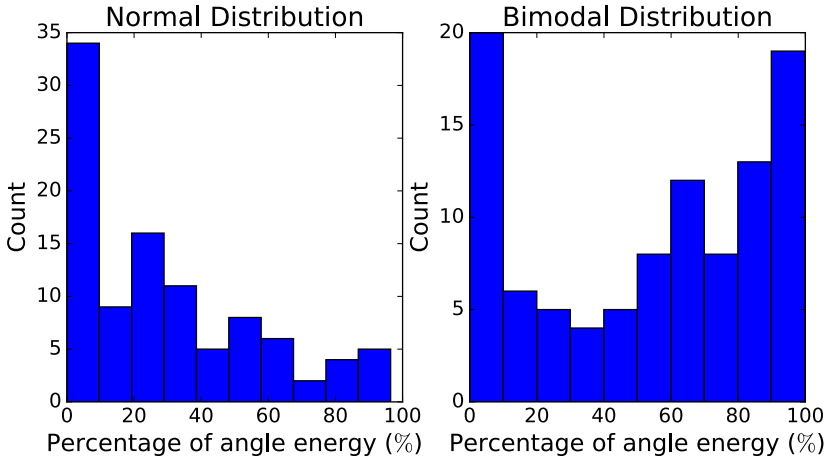
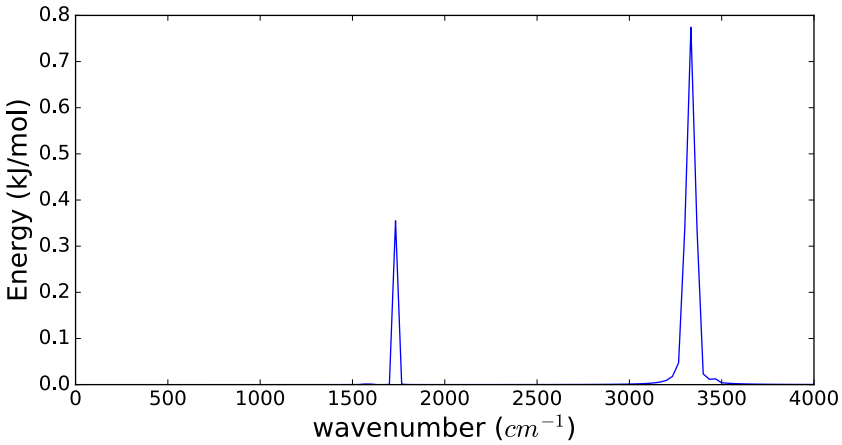


Figure 2.3 **DoubleWell: Percentage of low frequency energy for a Normal vs Bimodal velocity distribution.** The figure compares the percentage of energy that goes into the lower frequency motion when a normal and/or bimodal velocity distribution is used to initialise the velocities in an MD simulation

2.1.2 Water

The Water system, consists of 3 atoms. It has two bonds and one angle. The low frequency vibration is from the angle bending and is well separated from the bond stretching vibrations, Fig-2.4. Energy to the lowest frequency vibration, i.e. angle bending can be introduced by designing a low pass filter with the band between 0 – 2500 cm^{-1}.



Fourier Spectrum of Water molecule. Frequencies are displayed in wavenumber units. MD-Step = 10^4, dt= 0.1fs, temperature= 300K

Figure 2.4

Fig-2.5 shows the histogram of the percentage of the angle bending energy for simulation with normal and bimodal velocity distribution. As can be seen, higher percentages are reached when the bimodal distribution is used. Thus using the filtered velocities together with the bimodal distribution is able to introduce energy into an angle bending motion. The shown increase in the percentage of energy is obtained by using a filter scaling factor of 10.

When the filtered velocity is used as a parameter for the bimodal distribution, the degree and ability to enhance the slow vibration highly depends on the magnitude of the variance of the modes in the bimodal distribution. Thus if the magnitude of the filtered velocities is smaller than the magnitude of the mode's variance in the bimodal distribution than the procedure acts like an HMC implementation.

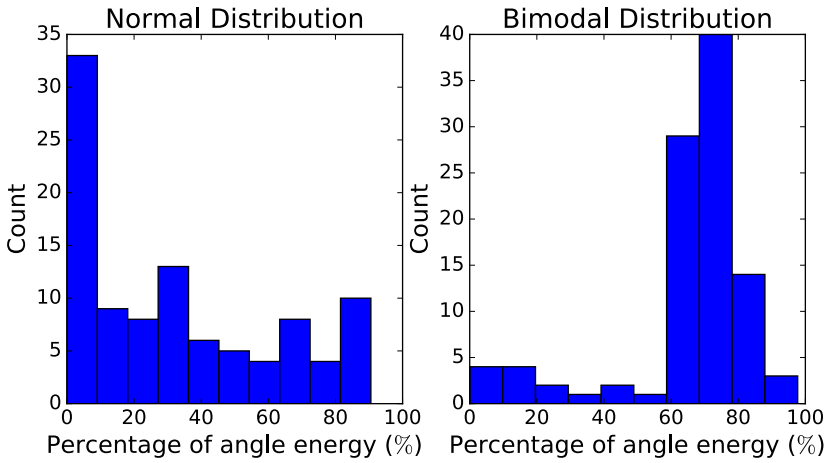
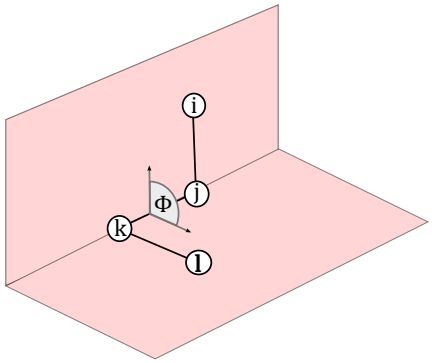


Figure 2.5 **Water: Percentage of angle Energy for a Normal vs Bimodal velocity distribution.** The figure compares the percentage of energy that goes into the abgle bending motion when a normal and/or bimodal velocity distribution is used to initialise the velocities in an MD simulation

2.1.3 Dihedral

The dihedral systems consist of 4 atoms, it has 3 bonds, 2 angles and 1 dihedral angle Fig-2.6. The parameters of the potential energy are adjusted such that the dihedral, angle and bond vibrations are well separated in the spectral density, Fig-2.8. Energy to the dihedral motion which is the lowest frequency vibration can be introduced by designing a low pass filter with the band being between $0 - 500\ cm^{-1}$.



Dihedral Angle Illustration. Illustration of the Dihedral Angle, Φ

Figure 2.6

The dihedral potential is a triple cosine potential Eq-2.3, and has been used before to simulate a butane molecule [57]. In the interval $0 - 360$ degrees it has 2 degenerate local minima and one global minima, Fig-2.7. The parameters $V_1 = 6.362\ kJ/mol$, $V_2 = -1.317,\ kJ/mol\ V_3 = 13.405\ kJ/mol$ which creates approximately 15, 10 kJ/mol height energy barriers.

$$V_{dihedral} = 0.5V_1(1 + cos(\phi)) + 0.5V_2(1 - cos(2\phi)) + 0.5V_3(1 + cos(3\phi)) \quad (2.3)$$

The angle potential has the following form and has the same parameters ($k_{angle} = 10063.34 kJ/molradian^2$, $\theta_0 = 112$ degree) for both angles

$$V_{angle} = 0.5k_{angle}(\theta - \theta_0)^2 \quad (2.4)$$

The bond potential is also in harmonic form and has the same parameters ($k_{bond} = 20086.8 kJ/molangstrom^2$, $r_0 = 1.53 angstrom$) for all bonds

$$V_{bond} = 0.5k_{bond}(r - r_0)^2 \quad (2.5)$$

Similar to before the histogram of the percentage of dihedral energy is plotted for simulations with bimodal and normal velocity distributions. As before, it can be seen from Fig-2.9 that the simulation with bimodal distribution together with the filtered

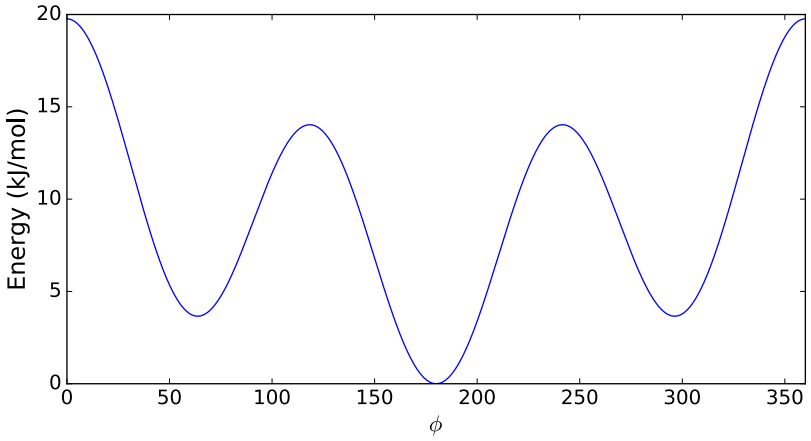
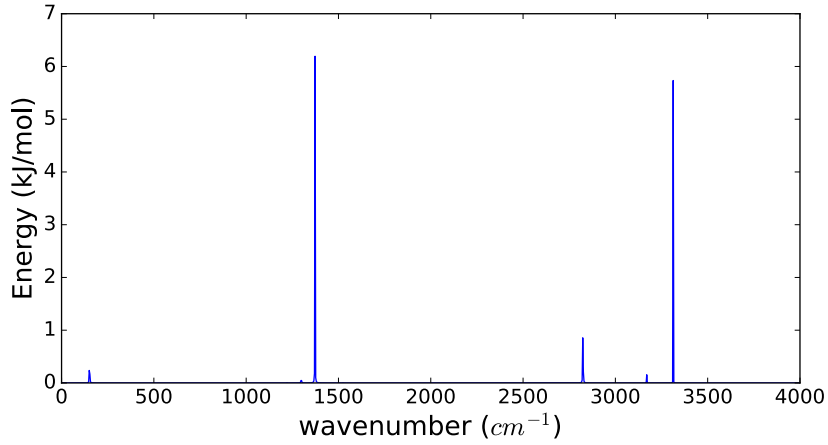


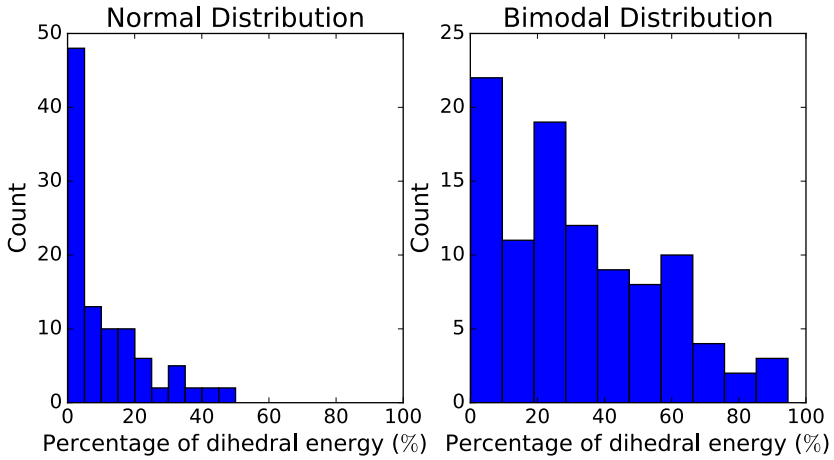
Figure 2.7 **dihedral potential - triple cosine.** triple cosine function to represent the dihedral potential. for mathematical form see Eq-2.3

velocities increases the percentage energy that goes into the dihedral significantly compared to the simulation with the normal velocity distribution. Thus using the filtered velocities as the parameters for the bimodal distribution makes it possible to add energy to the dihedral angle. The shown increase in the percentage of energy is obtained by using a filter scaling factor of 10.



Fourier Spectrum of Dihedral system. Frequencies are displayed in wavenumber units. MDStep = 10^4, dt= 2fs, temperature= 300K

Figure 2.8



Percentage of angle Energy for HMC and DFHMC. Frequencies are displayed in wavenumber units. MDStep = 10^4, dt= 0.1fs, temperature= 300K

Figure 2.9

2.1.4 Alanine dipeptide

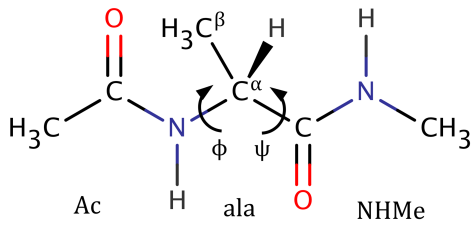


Figure 2.10 **Alanine di peptide, Molecule.** Molecular representation of alanine di peptide

Alanine dipeptide a molecular system with 22 atoms, is a relatively simple molecule to model the protein backbone dihedral motions Fig-2.10. The system is also studied in the application section.

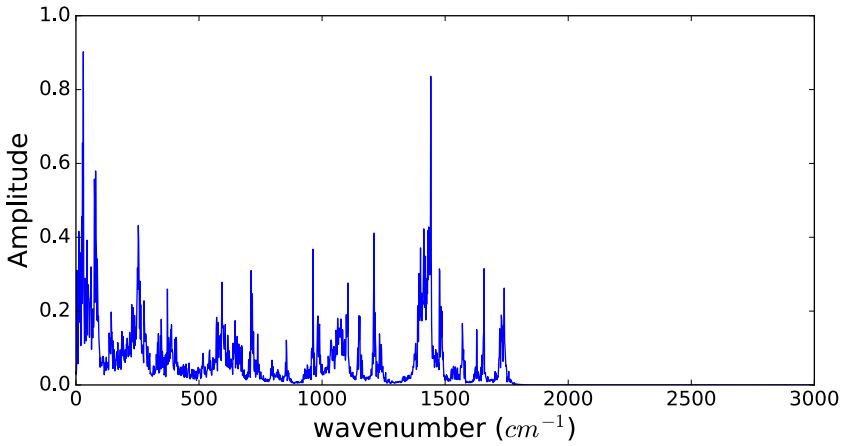
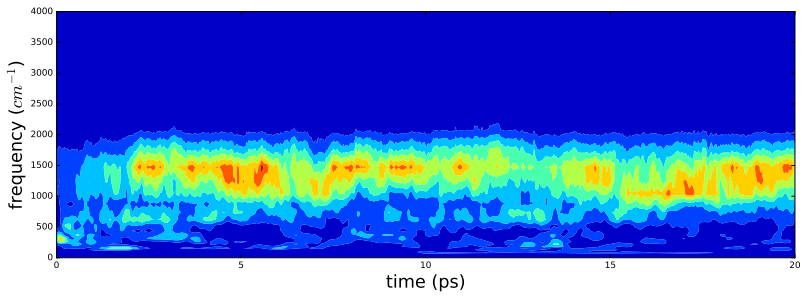


Figure 2.11 **FFT of alaine di peptide via MD simulation.** Frequencies are displayed in wavenumber units. MDStep = 10^4, dt= 2fs, temperature= 300K

In this system, the low frequency motion that causes the simulation to get stuck in local minima comes from the ϕ and ψ dihedral angles. However different than the previous system the frequency spectrum of alanine dipeptide Fig-2.11 does not have a clear separation of low and high frequency vibrational modes and thus specifically targeting just the ϕ and ψ dihedral motions is difficult. In the RDFMD study, it is shown that a filter that targets between $0 - 100cm^{-1}$ can capture 75% of ϕ and

almost 40% of ϕ while the other degrees of freedom are not affected significantly [1]. Therefore in our study, we apply a low pass filter that enhances frequencies between $0 - 100cm^{-1}$. Due to the higher number of degrees of freedom and a more complex frequency spectrum, the wavelet spectrum is analysed to inspect the effect of the bimodal velocity distribution.

Fig-2.12 shows the wavelet spectrum of alanine dipeptide where the initial velocities are drawn from a bimodal distribution with the bias values obtained through the application of a $0 - 100cm^{-1}$ digital filter. As can be seen, at the beginning of the simulation there is an increase in the intensity of lower frequency motions. However, it quickly dissipates into higher frequency vibrational modes.



Wavelet spectrum of alanine di peptide. Frequencies are displayed in wavenumber units. MDStep = 10^4, dt= 2fs, temperature= 300K

Figure 2.12

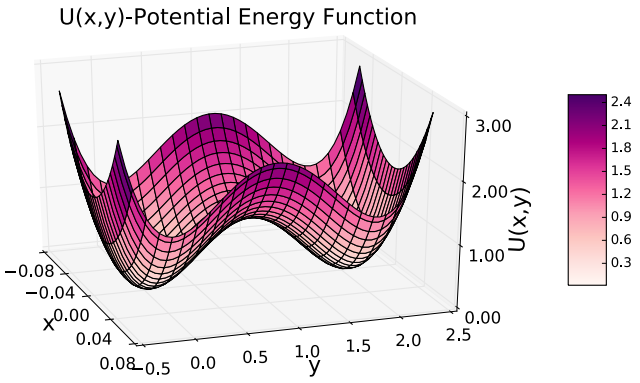
To summarize, using the filtered velocities within the bimodal distribution can introduce more energy to the selected frequencies compared to a normal distribution. Being able to add energy towards the angle and dihedral, supports the assumption that the procedure can add energy along a direction that is a nonlinear combination of the cartesian coordinates. These observations motivate to construct a Metropolis-Hastings algorithm by combining the bimodal distribution with an acceptance criterion to satisfy the detailed balance condition and hence create an equilibrium process which we call the DFHMC method. The next chapter investigates the DFHMC method on different systems with varying complexities and compares it with the standard HMC method.

Applications

Chapter 3

3.1 Double Well - 2D

Before applying the DFHMC method to complex high dimensional systems such as molecules, the HMC with the modified velocity distribution is studied in a two-dimensional system, where the bias value (b) of the bimodal velocity distribution is kept fixed. The aim is to make sure that the method with the biased velocity distribution conserves the target distribution. As explained in the theory section, two different velocity distribution functions with the corresponding acceptance criteria may be used to bias the velocities. In this test case, the system is investigated under the canonical ensemble.



2D Double Well Potential Energy $U(x, y)$. Potential energy function, $U(x, y)$ for the 2 dimensional double well system with respect to x and y. The energy barrier is 1, and the two minima are at position $(0, 0)$ and $(0, 2)$

Figure 3.1

A two-dimensional system with the following potential energy function is used as a simple testing model. The same energy surface was also used in the paper explaining the MEHMC method [2]

$$U(x, y) = 500x^2 + y^2(y - 2)^2 \tag{3.1}$$

The energy function is such that the motion in the x direction is much faster than along the y direction. The energy function, $U(x, y)$ has two minimum states, one at $(x, y) = (0, 0)$ and the other at $(x, y) = (0, 2)$ seperated by a barrier of height $U = 1.0$. The potential energy function can be seen in Figure 3.1.

Since the system is tested under the canonical ensemble the target distribution or in other words the probability of being in a state, $\boldsymbol{q} = (x, y)$ is as follows:

$$P(x, y) = \frac{\exp\left(\frac{-U(x,y)}{k_B T}\right)}{\int_{-\infty}^{\infty} \exp\left(\frac{-U(x,y)}{k_B T}\right) dx dy} \tag{3.2}$$

Since $U(x, y) = U(x) + U(y)$, the joint probability can be written as the product of the probabilities, $P(x)$ and $P(y)$. Hence they are independent

$$\begin{aligned} P(x, y) &= \frac{\exp\left(\frac{-500x^2}{k_B T}\right)}{\int_{-\infty}^{\infty} \exp\left(\frac{-500x^2}{k_B T}\right) dx} \frac{\exp\left(\frac{-y^2(y-2)^2}{k_B T}\right)}{\int_{-\infty}^{\infty} \exp\left(\frac{-y^2(y-2)^2}{k_B T}\right) dy} \\ &= P(x)P(y) \end{aligned} \tag{3.3}$$

Thus the histogram of the samples x (y) obtained from simulation should generate the distribution $P(x)$ ($P(y)$).

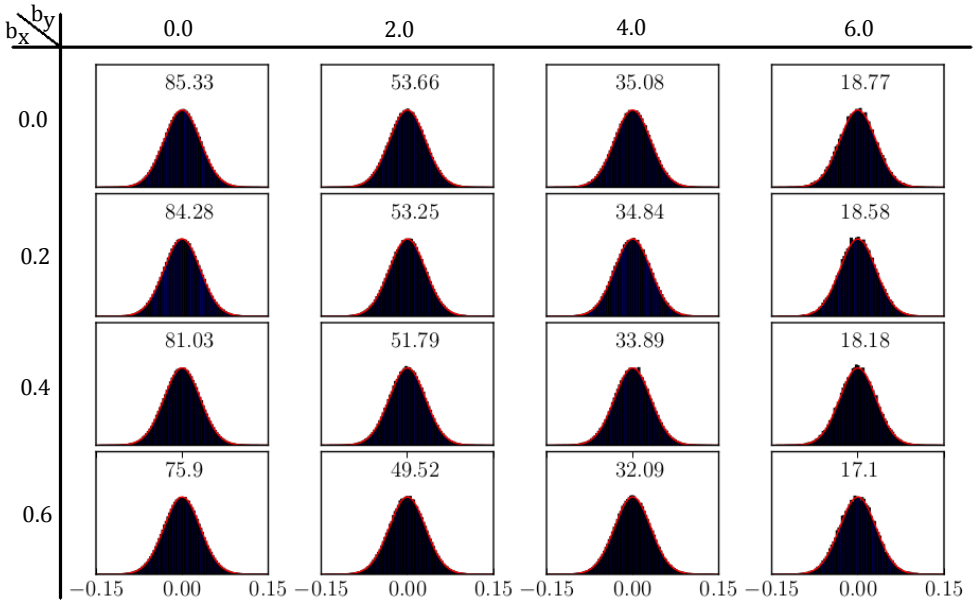
3.1.1 HMC with Bimodal Distribution

To show numerically that the bimodal velocity distribution with a given bias value and its corresponding acceptance criterion can generate the equilibrium distribution, we run multiple simulations for different fixed bias values and analyse the distributions generated.

The probability distribution, $P(x)$ and $P(y)$ for $k_B T = 1$ is obtained for several bias, b values as histograms from the configurations generated by the simulation and plotted together with the probability distribution structured in a matrix format for the HMC methods with single-variate [Figure 3.2, 3.4] and multi-variate [Figure 3.3, 3.5] velocity distributions.

As can be seen from the Figures 3.2, 3.4, 3.3, 3.5 the samples obtained using the several bias values for both of the HMC methods with the different velocity distribution fits reasonably well with the theoretical probability distribution along x and y coordinates. It can also be seen that the fit gets worse when the acceptance ratio gets smaller. Thus, for smaller acceptance ratios more HMC steps would be needed for a better fit.

To get a more quantitative measure, the PMF's(Potential of Mean Force) along the x and y coordinates are calculated. The PMF of some variable x is defined as follows:



2D Double Well system-Histograms for $P(x)$, single-variate bimodal distribution. Red Curve: $P(x)$ through numerical integration. Black bars: Histogram with 100 bins of the sample x obtained via the HMC method with **single-variate** bimodal velocity distribution for the listed bias values. The values inside of the figures are the acceptance ratio(%) for the corresponding simulation. Simulation parameters: #HMC Steps= 10^6, #MD Steps= 10, MD time step $dt = 0.04$, $k_BT = 1.0$

Figure 3.2

$$PMF(x) = -k_BT\,ln\left[\int_{-\infty}^{\infty} \exp\left\{-\frac{U(x,y)}{k_BT}\right\} dy\right] \tag{3.4}$$

And in our case where $U(x,y) = U(x) + U(y)$, the PMF of x, will be equal to $U(x) + c$.

$$PMF(x) = \underbrace{-k_BT\,ln\left[\exp\left\{\left(-\frac{U(x)}{k_BT}\right)\right\}\right]}_{U(x)} + \underbrace{ln\left[\int_{-\infty}^{\infty} \exp\left\{\left(-\frac{U(y)}{k_BT}\right)\right\} dy\right]}_{c:\ \ Constant} \tag{3.5}$$

To compare the PMF's from simulation with the theory, the minimum of the PMF is set to zero, to make the constant, c equal for both simulation and theory. The comparisons between the PMF's are done using two measures. The first measure is the RMSD (Root Mean Square Deviation) between the PMF's. If the values of PMF from the simulation for the set of x, is given by $PMF_{sim}(x) = s$, and the PMF from numerical integration for the same set x is $PMF_t(x) = t$ then the RMSD between the PMF's is as follows:

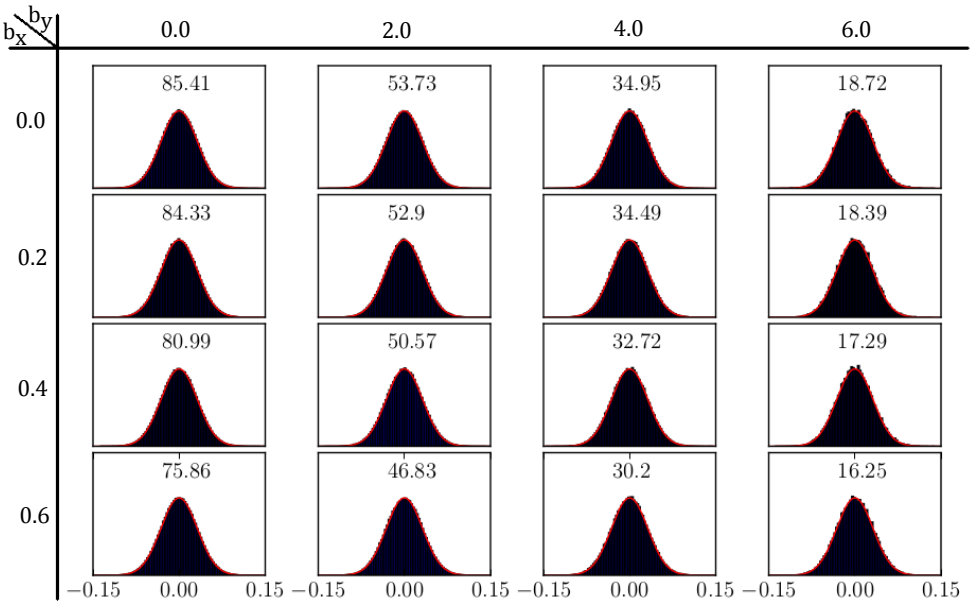


Figure 3.3 **2D Double Well system-Histograms for $P(x)$, multi-variate bimodal distribution.** Red Curve: $P(x)$ through numerical integration. Black bars: Histogram with 100 bins of the sample x obtained via the HMC method with **multi-variate** bimodal velocity distribution for the listed bias values. The values inside of the figures are the acceptance ratio(%) for the corresponding simulation. Simulation parameters: #HMC Steps= 10^6, #MD Steps= 10, MD time step $dt = 0.04$, $k_B T = 1.0$

$$RMSD(s,t) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_i - t_i)^2} \tag{3.6}$$

where n is the number of samples.

The RMSD is expected to get closer to 0 when the PMF values for the simulation get closer to the numerically integrated PMF.

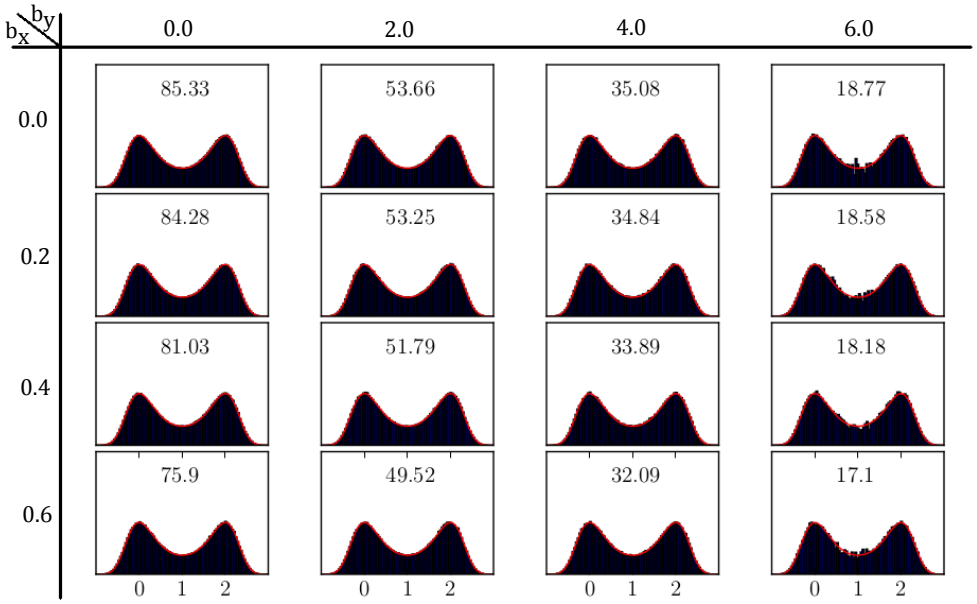
The second measure is the Pearson product-moment correlation coefficient (PCC) between both PMF's, and is defined as follows:

$$PCC(s,t) = \frac{\sum_{i=1}^{n}(s_i - \bar{s})(t_i - \bar{t})}{\sqrt{\sum_{i=1}^{n}(s_i - \bar{s})^2}\sqrt{\sum_{i=1}^{n}(t_i - \bar{t})^2}} \tag{3.7}$$

The PCC is expected to reach 1 if the PMF values come closer.

The PCC and the RMSD values with respect to the number of HMC steps for various bias,b values can be seen in Figure 3.6

According to the Figure 3.6, 3.7, for all bias values the PCC gets closer to 1 and the RMSD gets closer to 0 with increasing HMC steps. According to the graph, it also seems that the local convergence is slower for higher bias values. Nevertheless,



2D Double Well system-Histograms for $P(y)$, single-variate bimodal distribution. Red Curve: $P(y)$ through numerical integration. Black bars: Histogram with 100 bins of the sample y obtained via the HMC method with **single-variate** bimodal velocity distribution for the listed bias values. The values inside of the figures are the acceptance ratio(%) for the corresponding simulation. Simulation parameters: #HMC Steps= 10^6, #MD Steps= 10, MD time step $dt = 0.04$, $k_BT = 1.0$

Figure 3.4

the PMF values from the various simulations with different biases converges to the correct PMF obtained through numerical integration for the same set of input values. However, since the PMF is evaluated for the same input values as obtained from simulation, the RMSD and the PCC gives a measure only for the local convergence. If a simulation gets trapped in a local minimum, the local convergence measures might give a result that shows a good fit with the correct PMF. In such a situation it is clear that the simulation did not sample the whole domain of the underlying ensemble. Thus, the simulation would not be globally converged. Therefore, as a measure for the global convergence, an ensemble average can be used. To test the global convergence along x and y separately ensemble averages formed from only one coordinate are used.

An average value, A that only depends on x, can be calculated by

$$< A >= \int_{-\infty}^{\infty} A(x)P(x)dx \tag{3.8}$$

and similarly, an average value, B that depends only on y, can be calculated by the integral with respect to y only.

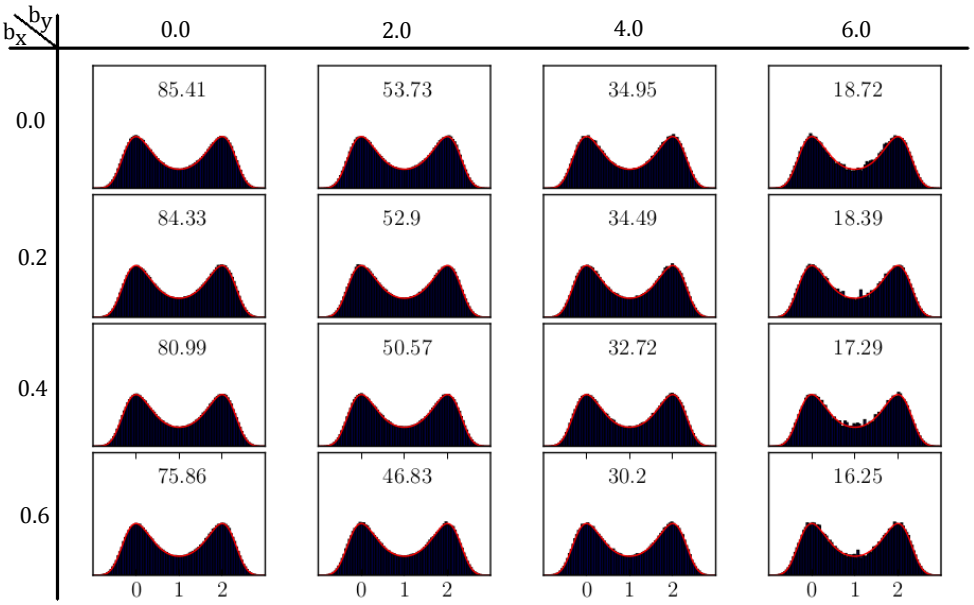


Figure 3.5 **2D Double Well system-Histograms for $P(y)$, multi-variate bimodal distribution.** Red Curve: $P(y)$ through numerical integration. Blue bars: Histogram with 100 bins of the sample y obtained via the HMC method with **multi-variate** bimodal velocity distribution for the listed bias values. The values inside of the figures are the acceptance ratio(%) for the corresponding simulation. Simulation parameters: #HMC Steps= 10^6, #MD Steps= 10, MD time step $dt = 0.04$, $k_BT = 1.0$

$$< B >= \int_{-\infty}^{\infty} B(y)P(y)dx \tag{3.9}$$

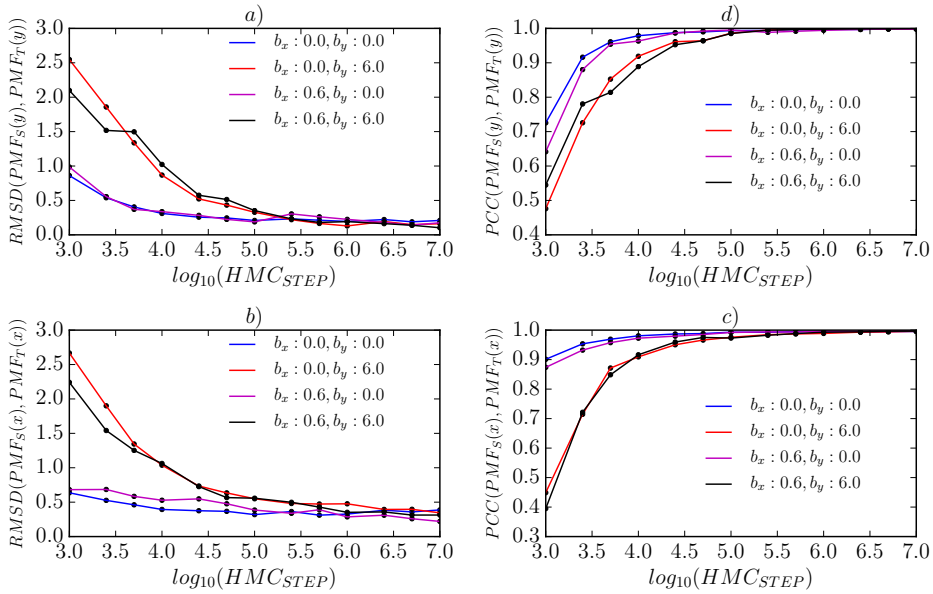
In order to interpret quantitatively how well HMC and the HMC with the modified velocity distribution for different bias velocities samples from the target distribution in both dimensions separately, ensemble averages of x^2 and y^2 are taken as a measure. Theoretically, the ensemble average of x^2 can be calculated as follows,

$$< x^2 >= \frac{\int_{-\infty}^{\infty} x^2 \exp\left(\frac{-500x^2}{k_BT}\right) dx}{\int_{-\infty}^{\infty} \exp\left(\frac{-500x^2}{k_BT}\right) dx} \tag{3.10}$$

The integral on the denominator is a Gaussian integral,

$$A = \int_{-\infty}^{\infty} \exp\left(\frac{-500x^2}{k_BT}\right) dx = \sqrt{\frac{\pi k_BT}{500}} \tag{3.11}$$

And the integral on the nominator can be obtained by taking the second derivative of the Gaussian integral, A with respect to x. Since,



2D double wel system-RMSD and PCC of PMF's. RMSD of PMF's along $\boldsymbol{a})x$ and $\boldsymbol{b})y$ coordinates. PCC of PMF's along $\boldsymbol{c})x$ and $\boldsymbol{d})y$ coordinates with respect to the log_{10} of the HMC Steps. Obtained using the single variate velocity distribution. Simulation Parameters: #HMC Steps= 10^7, #MD Steps= 10, MD time step $dt = 0.04$, $k_BT = 1.0$

Figure 3.6

$$\begin{aligned}\int \frac{d^2}{dx^2}\left[\exp\left(\frac{-500x^2}{k_BT}\right)dx\right] &= \frac{d^2}{dx^2}\int \exp\left(\frac{-500x^2}{k_BT}\right)dx \\ &= \frac{d^2}{dx^2}\sqrt{\frac{\pi k_BT}{500}} = 0\end{aligned} \tag{3.12}$$

The LHS (left hand side) of Eq 3.12 can be evaluated as:

$$\begin{aligned}\int \frac{d^2}{dx^2}\left[\exp\left(\frac{-500x^2}{k_BT}\right)dx\right] &= \left(\frac{1000}{k_BT}\right)^2\int_{-\infty}^{\infty} x^2\exp\left(\frac{-500x^2}{k_BT}\right)dx \\ &- \frac{1000}{k_BT}\int_{-\infty}^{\infty}\exp\left(\frac{-500x^2}{k_BT}\right)dx\end{aligned} \tag{3.13}$$

Since the LHS of Eq 3.13 is also equal to zero, the nominator of Eq 3.10 will be;

$$\int_{-\infty}^{\infty} x^2\exp\left(\frac{-500x^2}{k_BT}\right)dx = \frac{1000}{k_BT}\left(\frac{k_BT}{1000}\right)^2\int_{-\infty}^{\infty}\exp\left(\frac{-500x^2}{k_BT}\right)dx \tag{3.14}$$

Thus, the average in Eq-3.10,

$$< x^2 >= \frac{k_BT}{1000} \tag{3.15}$$

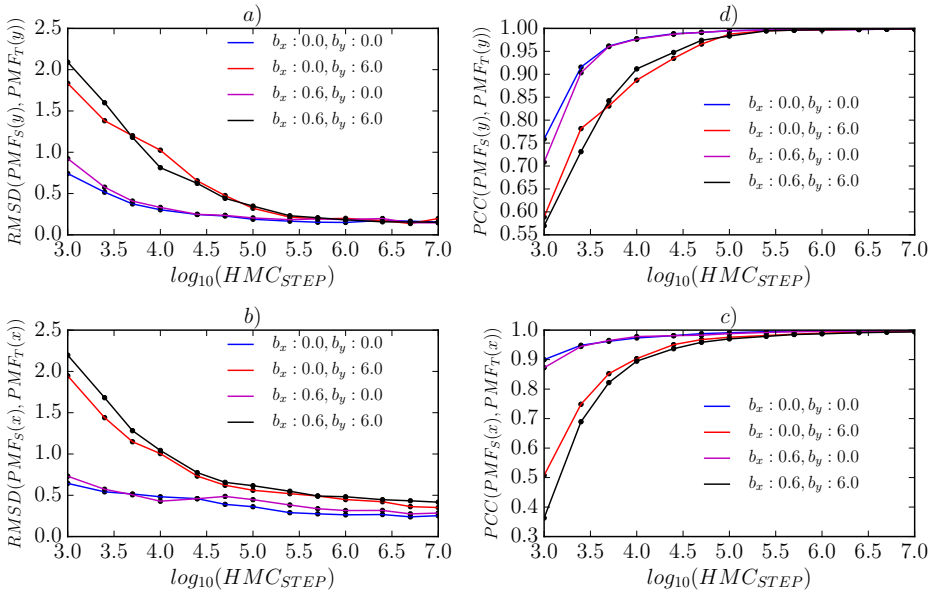


Figure 3.7

2D Double Well system-RMSD and PCC of PMF's. RMSD of PMF's along $\boldsymbol{a)}x$ and $\boldsymbol{b)}y$ coordinates. PCC of PMF's along $\boldsymbol{c)}x$ and $\boldsymbol{d)}y$ coordinates with respect to the log_{10} of the HMC Steps. Obtained using the Multivariate velocity distribution. Simulation parameters were the same as in caption Fig-3.6, Parameters: #HMC Steps= 10^7, #MD Steps= 10, MD time step $dt = 0.04$, $k_BT = 1.0$

For $k_BT = 1.0$, the ensemble average $< x^2 >= 0.001$
On the other hand the ensemble average of y^2,

$$< y^2 >= \frac{\int_{-\infty}^{\infty} y^2 \exp\left(\frac{-y^2(y-2)^2}{k_BT}\right) dy}{\int_{-\infty}^{\infty} \exp\left(\frac{-y^2(y-2)^2}{k_BT}\right) dy} \tag{3.16}$$

is calculated numerically. For $k_BT = 1.0$ the ensemble average of y^2 is,

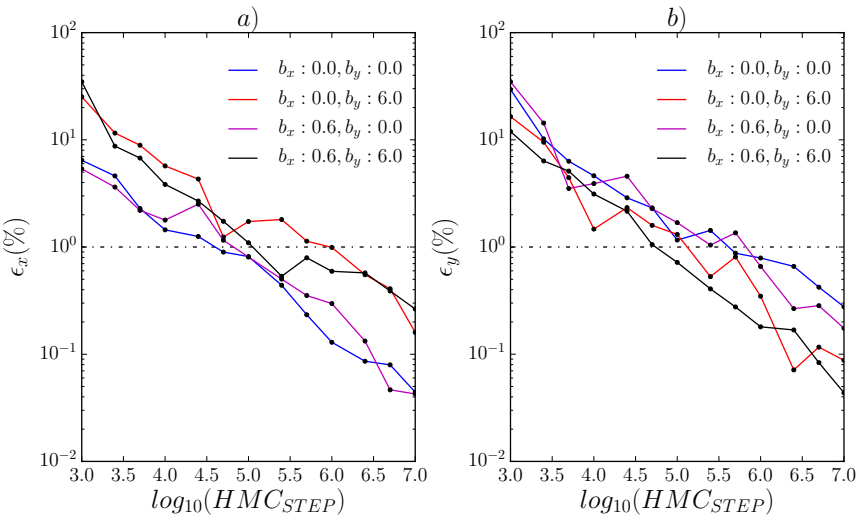
$$< y^2 >= 1.83274 \tag{3.17}$$

The root mean percentage error between the simulated and the theoretical ensemble average has been calculated for four simulations with the same simulation parameters but different random number seeds as follows and plotted with respect to the number of HMC steps in Figure 3.8.

$$\epsilon = \sqrt{\frac{1}{4}\sum_{i=0}^{4}\left(\frac{< z >_i - < Z >}{< Z >}\right)^2} \tag{3.18}$$

where $< z_i >$ represents the ensemble average of x^2 or y^2 depending on the co-

ordinate of interest for the i^{th} simulation. And $< Z >$ represents the theoretically calculated ensemble average of the corresponding coordinate.



2D Double Well system-root mean squared percentage error of ensemble average for single variate velocity distribution. Root mean squared percentage error of $\boldsymbol{a)}$ $< x^2 >$ and $\boldsymbol{b)}$ $< y^2 >$ in a log scale with respect to the log_{10} of HMC Step. Obtained using the Singlevariate velocity distribution. Simulation parameters were the same as in caption Fig-3.6, Parameters: #HMC Steps= 10^7, #MD Steps= 10, MD time step $dt = 0.04$, $k_B T = 1.0$

Figure 3.8

As can be seen from Figure 3.8, 3.9 the root mean percentage error for all simulations with different bias values along both coordinates drops with increasing HMC steps. After a million (10^6) HMC steps the error in averages drops below 1%. Hence, it can be said that for all simulations with various bias values the ensemble averages are relatively well obtained and are able to sample correctly the configurational canonical ensemble. In the given simulation time, it can also be seen that the simulation with a bias in the y coordinate (slow dimension) results on average in a smaller percentage error for the $< y^2 >$ ensemble average but a higher percentage error for the $< x^2 >$ ensemble average than the simulations without the bias in that coordinate.

To sum up, the HMC method with the modified velocity distribution, where we introduced a bias to the velocities is shown to generate samples that follow the target distribution. From the local convergence figures, Figure 3.6 it can be observed that the simulations with a bias value different than zero converge more slowly locally. However, according to the global convergence measure, Figure 3.8 the simulations with a bias in the slow dimension, y seems to result in smaller percentage error for the average related to the y coordinate, $< y^2 >$ than the simulations without the bias in that coordinate, but seems to perform worse for the $< x^2 >$ average. For a conclusive

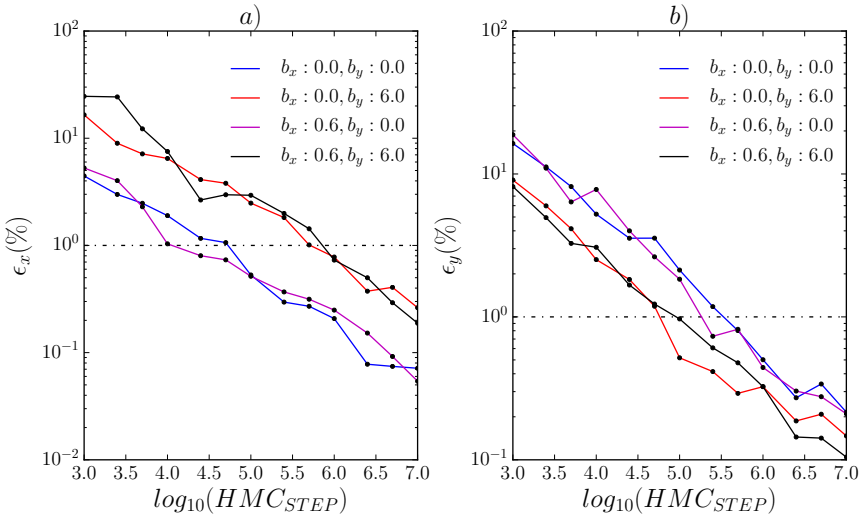


Figure 3.9

2D Double Well system - root mean squared percentage error of ensemble average for multivariate velocity distribution. Root mean squared percentage error of $\boldsymbol{a)}$ $< x^2 >$ and $\boldsymbol{b)}$ $< y^2 >$ in a log scale with respect to the log_{10} of HMC Step. Obtained using the Multivariate velocity distribution. Simulation parameters were the same as in caption Fig-3.6, Parameters: #HMC Steps= 10^7, #MD Steps= 10, MD time step $dt = 0.04$, $k_B T = 1.0$

decision on the performance effect of the bias values more simulation experiments needs to be constructed at lower $k_B T$ values where the HMC without any bias needs relatively long HMC steps to converge globally. Thus the next subsection analyses the performance of the DFHMC method where the bias values are obtained by filtering a specified frequency window, on the same surface but with a lower $k_B T$ value.

3.1.2 DFHMC vs HMC

In this section, we analyse and compare the convergence performance of the proposed method DFHMC to the standard HMC method. The comparison is made by analysing the x^2, y^2 ensemble averages and energy barrier crossing along the (y) coordinate. The kT value is reduced to 0.1 which will lower the temperature of the simulation and makes it difficult for a standard HMC or MD simulation to overcome the energy barrier along the y direction which has a height of 1 kJ/mol.

For the DFHMC method, a filter is designed via the following MATLAB function,

$$\boldsymbol{c} = \boldsymbol{fircls}(1000,\ [0\ 0.15\ 1],\ [1\ 0],\ [1.01\ 0.01],\ [0.09\ -0.01],\ 'text') \quad (3.19)$$

which, gives the frequency response graph shown in the digital filter section Fig-1.8, The normalised frequency 0.15 is determined such that it separates the high (x direction) and low frequencies (y direction) present in the system. Thus, the frequency values of the slow and high frequencies need to be known. In this 2D system, the high-frequency motion is along the x direction. The frequency can be calculated from the spring constant which is equal to 1000. Thus, the high frequency is equal to:

$$\begin{aligned} f &= \frac{1}{2\pi}\sqrt{\frac{k}{m}} \\ &= \frac{1}{2\pi}\sqrt{\frac{1000}{1}} = 5.03 \end{aligned} \quad (3.20)$$

Since the MD time step used in the DFHMC simulation is chosen to be 0.04, the normalised frequency of $f = 5.03$ is,

$$\begin{aligned} f_{normalised} &= fx2xdt \\ &= 5.03x2x0.04 = 0.4 \end{aligned} \quad (3.21)$$

To determine the low-frequency value, a Fourier transform over a molecular dynamics trajectory has been taken to obtain the frequency spectrum. From this frequency spectrum, the low-frequency values are determined. Certainly, the high-frequency value can also be obtained this way.

As can be seen from the Fig-3.10, any normalised frequency value that is bigger than 0.1 and smaller than 0.4 would separate the high frequencies from the low-frequency values. Thus, the choice of 0.15 as the separation value satisfies the requirement.

To analyse the global convergence of DFHMC with the given filter design, the root mean square percent error of the x^2 and y^2 ensemble averages to the exact values at low temperature ($kT = 0.1$), are plotted for the DFHMC and HMC simulation Fig-3.11.
The exact values at the temperature of $kT = 0.1$ are calculated via Eq-3.1.1 for $<$

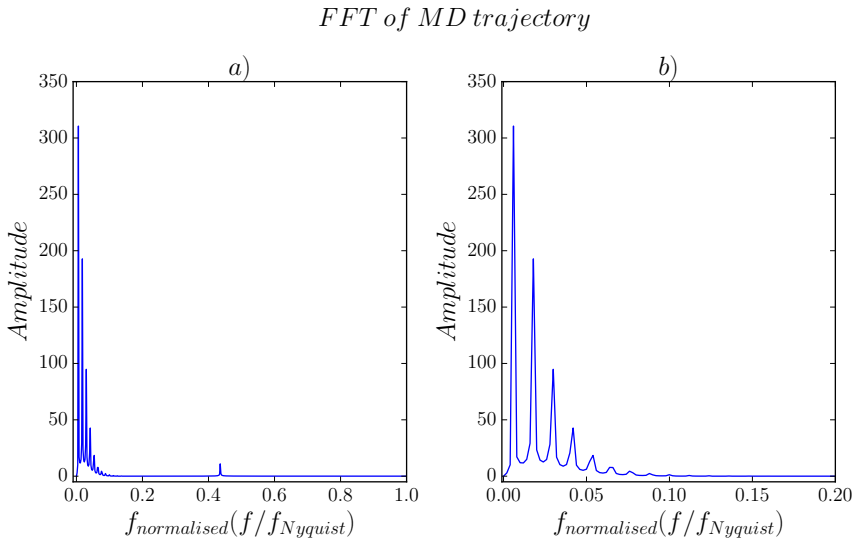


Figure 3.10 **2D Double Well system-FFT of MD trajectory.** FFT of the linear combination of the MD trajectories along x and y dimensions. **a)** full positive frequency range **b)** frequecncy range between 0.0 and 0.2. Simulation Parameters: #MD Steps= 1000, MD time step $dt = 0.04$, starting position $(x_0, y_0) = (0.0, 1.0)$, This starting position correspond to the top of the energy barrier allowing the NVE simulation to have enough energy to explore all local minima, thereby giving a better representation of the overal frequency spectrum.

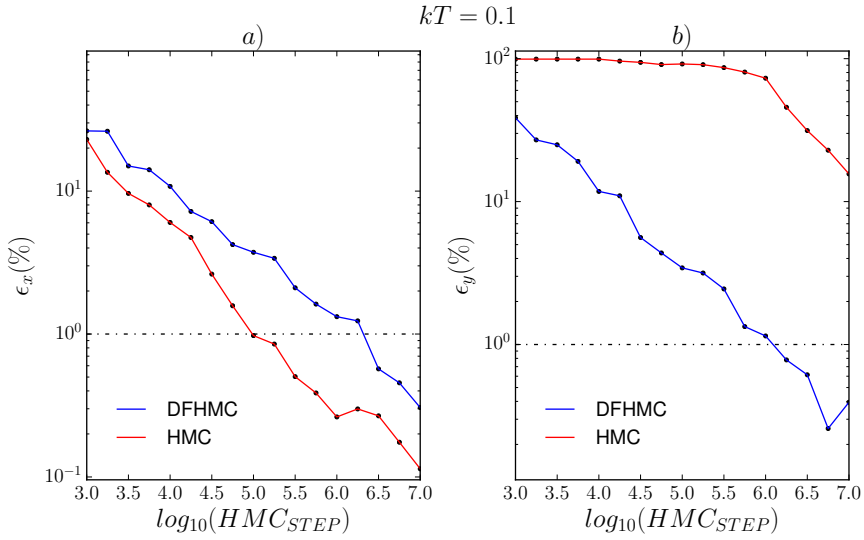
$x^2 >$ and Eq-3.16 using numerical integration for $< y^2 >$ to be,

$$\begin{aligned} < x^2 > &= 0.0001 \\ < y^2 > &= 1.97252 \end{aligned} \tag{3.22}$$

As can be seen from Fig-3.11 the error in the fast direction x converges to below 1% for both HMC and DFHMC, the error in the slow direction (y) is much smaller in DFHMC than in HMC. While DFHMC is below the 1% error region, the HMC is still above 10% of error. The small error in DFHMC compared to HMC can be explained by the system crossing the barrier more frequently Fig-3.12.

Note that in Fig-3.12 the total HMC steps plotted on the x-axis, are different for the HMC and DFHMC simulation. While for the HMC simulation a barrier crossing event is observed in the order of 10^6 HMC steps, for the DFHMC a plot with a max range of 10^6 would make the transitions step not visible. Therefore, the range is decreased to the order of 10^5 to see the transitions better on the figure.

To summarise, we observe that the use of the bimodal velocity distribution together with the appropriate acceptance criterion conserves the equilibrium distribution of the system. From the results of the simulations at lower $kT = 0.1$ values compared to the energy barrier height, i.e. 1, we noted that the simulations that use the bimodal velocity distribution with the bias obtained via a digital filter converge much more



2D Double Well system-root mean squared percentage error of ensemble average. Root mean squared percentage error of $\boldsymbol{a)}$ $< x^2 >$ and $\boldsymbol{b)}$ $< y^2 >$ over 10 independent trials, in a log scale with respect to the log_{10} of HMC Step. Obtained using the Multivariate velocity distribution for $k_B T = 0.1$. Simulation Parameters: #HMC Steps= 10^7, #MD Steps= 10, MD time step $dt = 0.04$, Digital Filter application frequency was every 10 HMC steps.

Figure 3.11

quickly to the ensemble average along the slow direction (y)compared to the standard Hybrid Monte Carlo method. On the other hand, the convergence on the ensemble average along the fast direction (x) is slower with the DFHMC method compared to the HMC method. Suggesting that the DFHMC method is better in exploring the surface globally; however, it is not as good in exploring the surface locally. The enhancement in the global convergence of the DFHMC method is especially prominent when the energy barrier is high enough such that the standard HMC method has difficulties to overcome it.

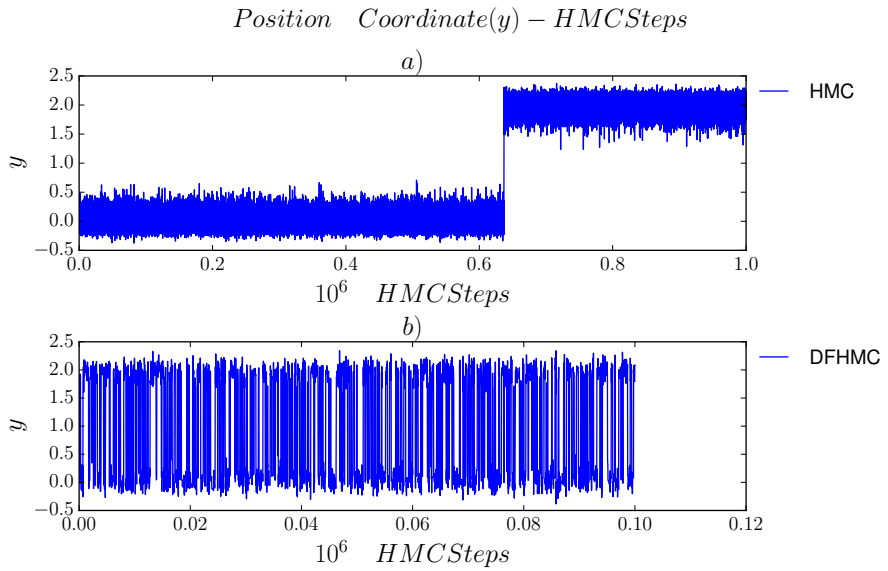
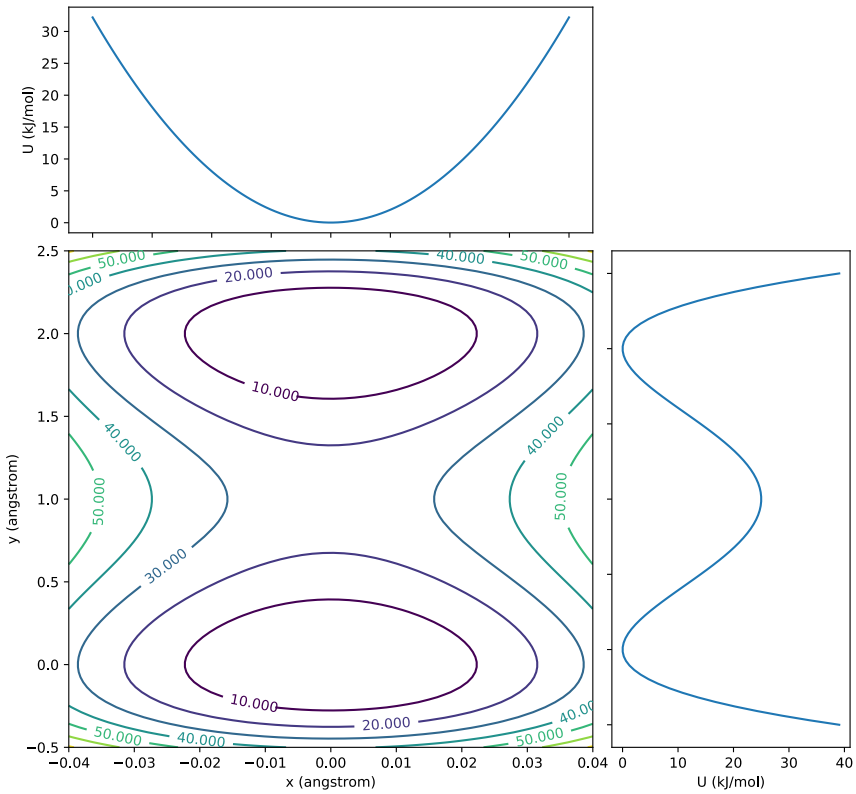


Figure 3.12 **2D Double Well system-Positions in the y dimension as a function of HMC steps.** The change of positions along the slow dimension y with respect to HMC steps for **a)** HMC **b)** DFHMC simulations. Simulation Parameters: #HMC Steps= 10^6 for HMC 10^5 for DFHMC, #MD Steps= 10, MD time step $dt = 0.04$, $k_B T = 0.1$, Digital Filter application frequency was every 10 HMC steps.

3.2 Double Well - 3D



3D Double Well potential surface along x, y**.** The graph shows the equal potential energy lines, isoline graph of the double well energy surface. On the top and right side of the figure the projections of the surface on to the y and x axis are drawn.

Figure 3.13

The 2D potential simulations investigated in the previous sections were performed on a code that was developed from scratch using python. The code is suitable to test and design algorithms such as DFHMC, but due to the use of non-optimised algorithms and non-parallelisation, it is not suitable to run on 3D systems with multiple particles such as molecules. Thus to be able to apply DFHMC on molecules, DFHMC is integrated into the optimized Molecular Dynamics open source platform OpenMM [11] for more details on the code integration see section 4

Before applying the DFHMC on Molecules using OpenMM, we test the DFHMC implementation on a 3D Double Well potential Eq-3.26.

$$U(x, y, z) = K_{xz}(x^2 + z^2) + K_y y^2 (y - b)^2 \tag{3.23}$$

As can be seen, the 3D potential has the same structure as the 2D double well

potential investigated in the previous section. To make the 2D double well potential a 3D potential, a harmonic potential is added along the z direction. Thus along x and z there is a harmonic potential, and in y direction there is a double well potential. Since the potential is separable along all the x, y, z coordinates, the ensemble averages of $< y^2 >$ and $< x^2 >$ will be the same as in 2D case calculated in 2D Double Well section.

Using this 3D Double Well potential in OpenMM, we can check if the DFHMC integration to OpenMM is correct by comparing the outcome of the 3D Double Well system with the 2D double well system.

The parameters of the double well potential are chosen to be as follows:

$K_{xz} = 40227.6254\ kJ/(molAngstrom^2)$, $K_y = 25\ kJ/(molAngstrom^4)$, $b = 2\ Angstrom$

The energy surface of the double well potential with the given parameters can be seen in Fig-3.13. The energy barrier is along the y direction, and the barrier height is equal to $25kJ/mol$ as can be seen from the projection of the surface on to the y axis. The local minima of the system is located at $(0, 0, 0)$ and $(0, 2, 0)$ angstrom.

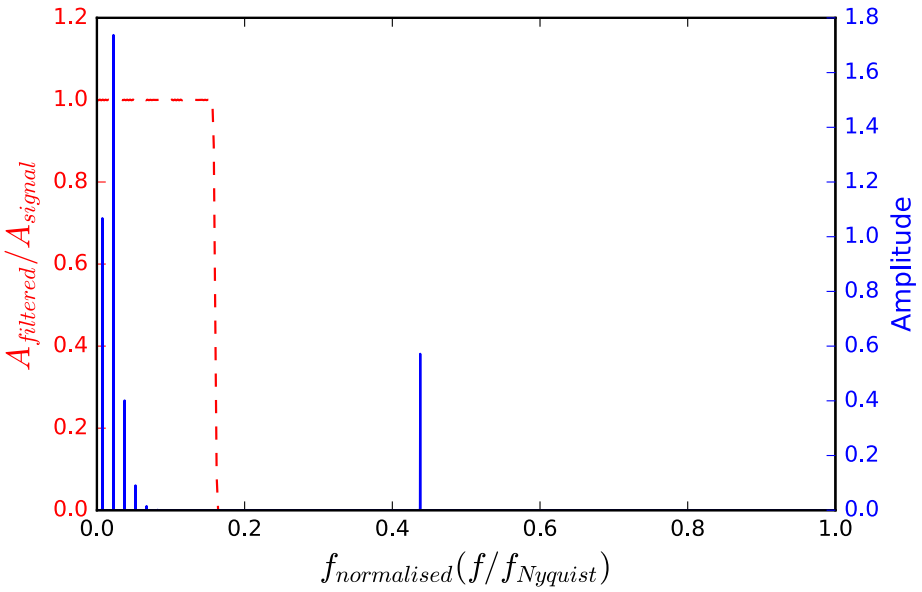


Figure 3.14 **Double Well 3D Power spectrum of MD trajectory and Frequency Response of designed filter.** Power Spectrum (in blue and solid line) of NVE MD velocity trajectory. Frequency Response (in red, dashed line) of the designed low pass filter that removes high frequency from the signal. Simulation paramaters: MD Steps = 10^4, MD timestep $dt = 4fs$, starting position $(x_0, y_0, z_0) = (0.0, 1.0, 0.0)$ angstroms.

The frequency characteristics of a system moving in such a double well potential are obtained by running an NVE constant total energy simulation and then calculating the average of the Fourier transform of the velocity trajectories along each degree of

freedom. The starting point of the NVE simulation as in the previous section is chosen to be on top of the energy barrier, i.e., (0, 1, 0) angstrom, to ensure the trajectory reaches all local minima and thereby generating a better representation of the frequency spectrum. As can be seen from Fig-3.14, the high and low frequency signals are well separated and can be easily targeted with a low pass filter design.

To be used in the DFHMC method a filter is designed that suppresses the high frequency motion in the system, the following Matlab function is used to generate the filter coefficients.

$$\boldsymbol{c} = \boldsymbol{fircls}(1000,\ [0\ 0.16\ 1],\ [1\ 0],\ [1.01\ 0.01],\ [0.09\ \ -0.01],\ 'text') \quad (3.24)$$

which, gives the frequency response graph shown in red and dashed line in Fig-3.14. As can be seen from the figure, the designed filter is suitable to separate the low and high frequency vibrations in the system shown as a solid blue line on the same graph. The normalised frequency 0.16 at $dt = 4fs$, corresponds to $667cm^{-1}$ in wavenumber units.

$$\begin{aligned} f &= f_{normalised} \times f_{Nyquist} \\ &= \frac{f_{normalised}}{2dt} \\ &= \frac{0.16}{2 \times 4} \quad fs^{-1} = \frac{0.16}{8c} \quad cm^{-1} \\ &= \frac{0.16}{8 \times 3 \times 10^{-5}} \quad cm^{-1} \\ &= 667 \quad cm^{-1} \end{aligned} \quad (3.25)$$

where c is the speed of light in cm/fs units, $3 \times 10^{-5}\ cm/fs$

Thus a filter that enhances frequencies between $0 - 667cm^{-1}$ and suppresses any other frequency range is used in the DFHMC method for the double well potential system.

Fig-3.15 shows the potential of mean force (PMF) along the slow degree of freedom, y, for DFHMC and HMC simulation results. The solid line is the numerically obtained PMF and is used as the benchmark. As can be seen, both HMC and DFHMC follow the numerically obtained PMF quite well, indicating that the methods conserve equilibrium.

To assess the global convergence speed of the two methods, as in the previous section of the 2D double well potential, the ensemble average of y^2 is calculated at varying HMC steps and the percentage error from the benchmark average is plotted in Fig-3.16. The average used as the benchmark is calculated via numerical integration to be $< y^2 >= 0.0197\ angstrom^2$

As can be seen from Fig-3.16, the DFHMC method converges to the ensemble

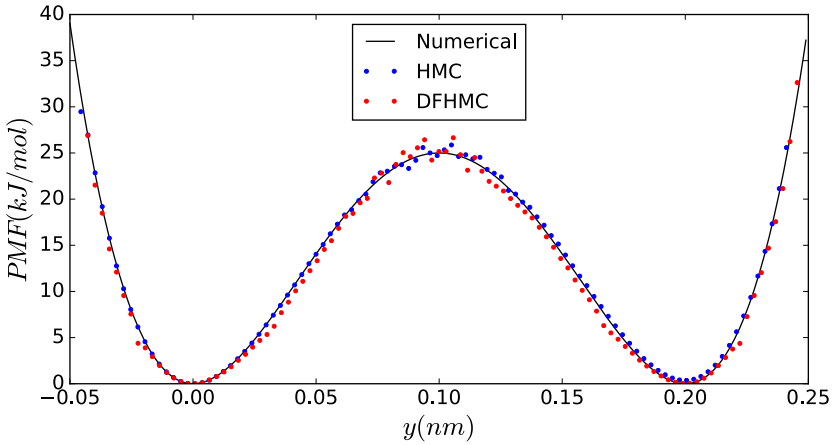
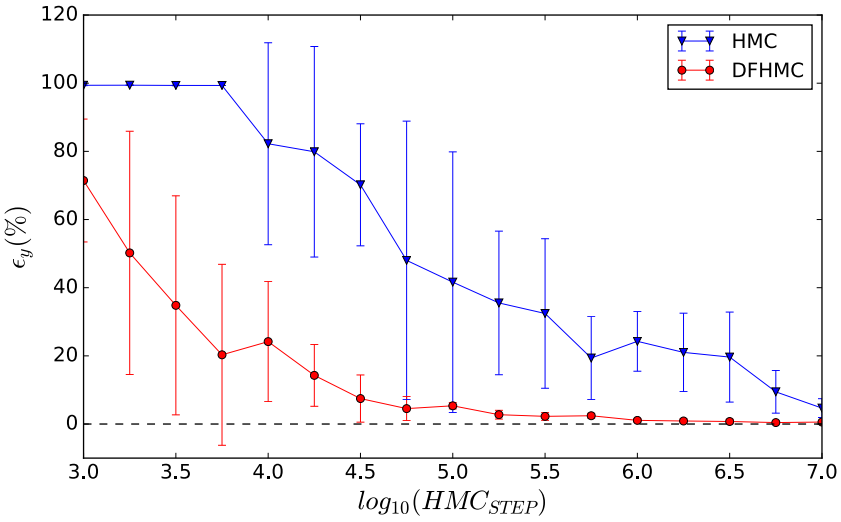


Figure 3.15 **PMF along y coordinate for the Double Well potential.** Solid black line is the PMF ontained via numerical integration. The blue and red dots represents the result obtained via HMC and DFHMC simulation respectively. Simulation Parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

average much faster than the standard HMC method. Indicating that the DFHMC method is enhancing the simulation to cross the energy barrier. The variations across repeated simulations for the DFHMC also decreases faster compared to the HMC, as seen by the smaller error bars.

To qualitatively assess the rate of energy barrier crossing, the trajectory of the slow degree of freedom,y, of a representative simulation for HMC and DFHMC is generated in Fig-3.17. As can be seen from the figure, the rate of energy barrier crossing, i.e. moving from the local minima around $y = 0$ to $y = 2 \quad angstrom = 0.2 \quad nm$ and vice versa is much higher in DFHMC compared to HMC. The HMC simulation spends more time around the local minima before transitioning between them, via crossing the energy barrier.

To summarise, The double well potential in the 3D setup is simulated via the HMC and DFHMC methods. Consistent with our observations from the previous section of the 2D double well system, the DFHMC method converges faster to the ensemble average compared to the HMC method. Both HMC and DFHMC methods conserve the equilibrium distribution. Thus the performance is consistent with the observed result for the 2D Double Well potential analysed via the non-optimised python code in the DoubleWell- 2D section, which indicates that the integration of the DFHMC method into the OpenMM molecular dynamics package works as expected. Using the OpenMM implementation the DFHMC and HMC application can be scaled to more complex and bigger systems in a computationally efficient way. The observed enhancement of the DFHMC method on the double well potential shows us that the DFHMC is capable of adding energy along a slow degree of freedom that is aligned



Convergence of y^2 ensemble average for Double Well system. Mean absolute percentage error of $< y^2 >$ over 4 independent simulations with respect to the log_{10} of HMC Steps. The error bars represent the standard deviation of the percentage error ($PE \pm \sigma_{PE}$) from the 4 independent simulation repeats. Simulation parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

Figure 3.16

along a cartesian coordinate. In the next two sections, we investigate the behaviour of the DFHMC on surfaces where the slow degree of freedom is along a linear combination of the cartesian coordinates and on a system where the slow degree of freedom changes direction such as a surface with a curved transition path.

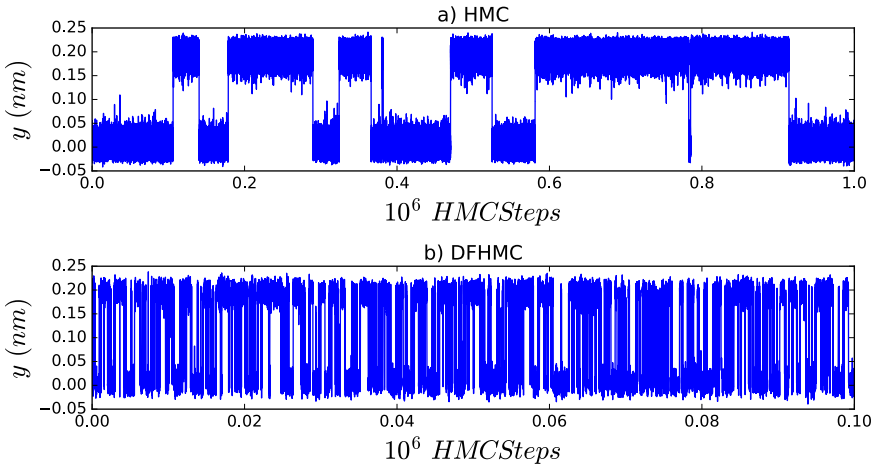
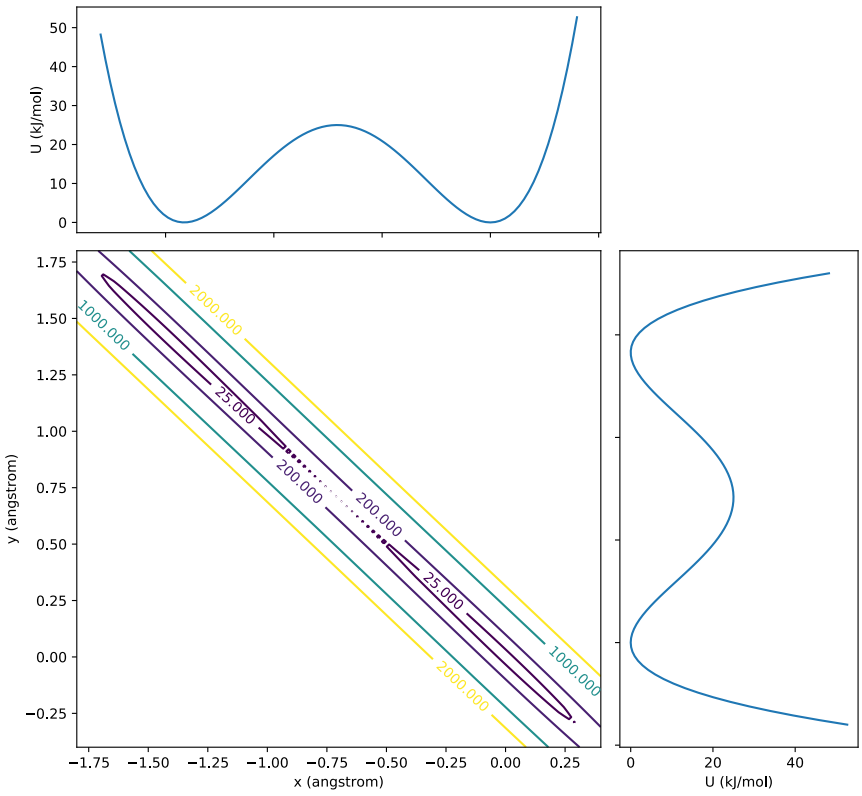


Figure 3.17 **Trajectory of y coordinate for the Double Well system.** The simulation trajectory of y for DFHMC and HMC simulation. The first 10^5 and 10^6 HMCSteps are shown for DFHMC and HMC respectively. Simulation parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

3.3 Double Well - 3D Rotated

The analysed Double Well system in the previous sections have the potential energy barrier along the y cartesian axis. Thus for DFHMC to enhance the sampling the filtered velocity needs to have values close to zero along x and non-zero along the y direction. In this section, we test if DFHMC can also generate the right filtered velocities when the potential energy barrier is along a linear combination of the cartesian coordinates.



3D Double Well potential surface along x, y**, The coordinates are rotated by** 45 **degrees.** The graph shows the equal potential energy lines, isoline graph of the double well energy surface where the coordinates are rotated by 45 degrees. On the top and right side of the figure the projections of the surface on to the y and x axis are drawn.

Figure 3.18

To analyse how DFHMC performs on a landscape where the energy barrier is along a linear combination of the cartesian coordinates, the 3D double well potential is rotated on the $x - y$ plane by an angle of $\theta = 45$ degree. The same double well equation in the previous chapter, Eq-3.26, is written as a function of a 45 degree rotated cartesian coordinates, x_{rot} and y_{rot}

$$U(x, y, z) = K_{xz}(x_{rot}^2 + z^2) + K_y y_{rot}^2 (y_{rot} - b)^2 \tag{3.26}$$

The rotational transformation of x and y to x_{rot} and y_{rot} can be obtained as follows,

$$\begin{aligned} x_{rot} &= xcos(\theta) + ysin(\theta) \\ y_{rot} &= -xsin(\theta) + ycos(\theta) \end{aligned} \tag{3.27}$$

Aligned with the previous section, the parameters are chosen to be as follows: $K_{xz} = 40227.6254\ kJ/(molAngstrom^2)$, $K_y = 25\ kJ/(molAngstrom^4)$, $b = 2$ $Angstrom$, $theta = 45$

The energy surface created by this transformation can be seen in Fig-3.18.The energy barrier is along $y = -x$ axis and has a height of $25kJ/mol$. The projections of the energy surface on to the x and y coordinates can be seen on the top and right of the figure respectively. The local minima of the system is located at $(0, 0, 0)$ and $(-1.41, 1.41, 0)$ Angstrom. To overcome the barrier the filter would need to add energy along the $y = -x$ direction.

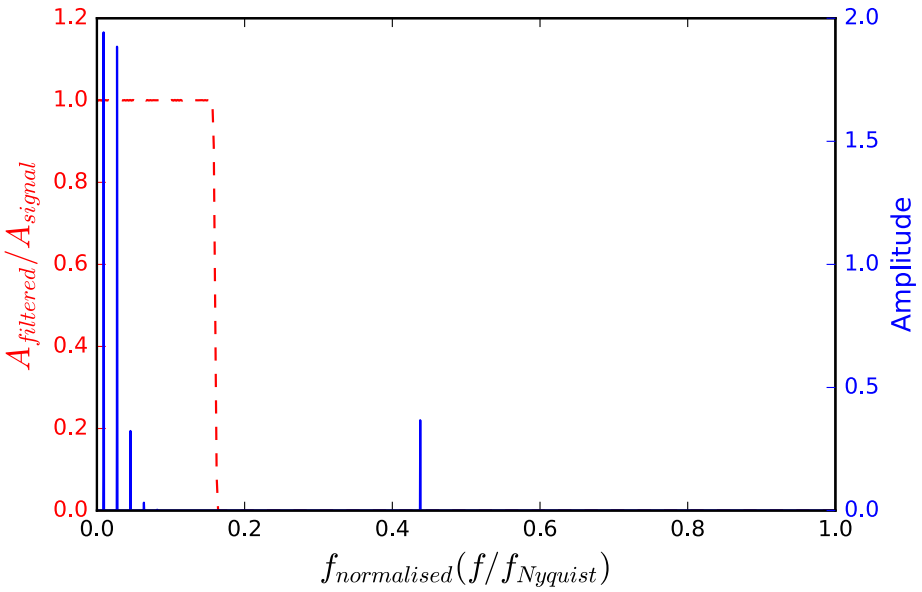
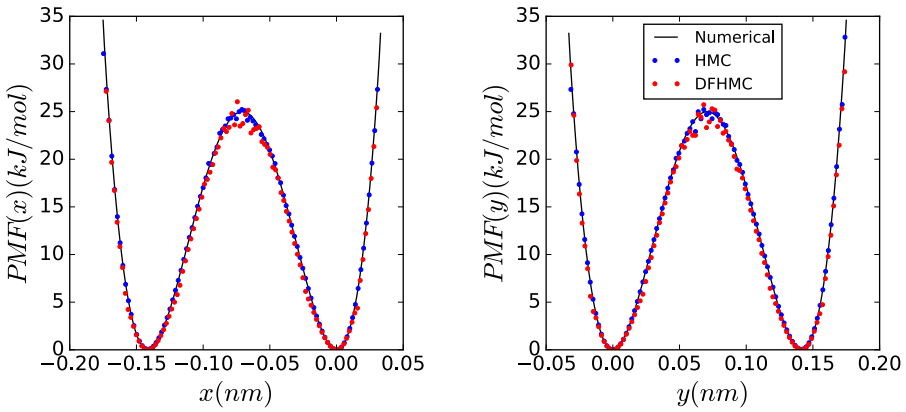


Figure 3.19 **45 degree Rotated Double Well 3D Power spectrum of MD trajectory and Frequency Response of designed filter.** Power Spectrum (in blue and solid line) of NVE MD velocity trajectory. Frequency Response (in red, dashed line) of the designed low pass filter that removes high frequency from the signal. Simulation paramaters: MD Steps = 10^4, MD timestep $dt = 4$ femtosecond, starting position $(x_0, y_0, z_0) = (-0.707, 0.707, 0.0)$ Angstroms.

To obtain the frequency characteristics of this system, as done previously an NVE simulation is run, and the average of the Fourier transform applied to the velocity

trajectory along each degree of freedom is calculated. The starting position of the NVE simulation is chosen to be the top of the energy barrier,i.e., $(-0.707, 0.707, 0.0)$, this allows the simulation to visit all local minima and to capture better the various frequencies available in the trajectories caused by the curvature of the surface. As the curvature of the surface is not changed compared to the surface in the previous section, as expected the rotation of the potential does not cause any changes in the observed frequencies of the system [Fig-3.19], thus the same filter coefficients can be used for this rotated system.

the DFHMC simulation is run using a filter that enhances the frequencies between $0 - 0.16$ normalised frequency at $4fs$, or equivalently the range $0 - 667$ in cm^{-1}.



PMF along x and y coordinates for the rotated Double Well potential. Solid black line is the PMF obtained via numerical integration. The blue and red dots represent the results obtained via HMC and DFHMC simulation respectively. Simulation Parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

Figure 3.20

Fig-3.20 shows the potential of mean force (PMF) along x and y coordinates as obtained from the HMC and DFHMC simulation results. The solid back line is used as the benchmark and is calculated using numerical integration. Similar to the previous section we observe that DFHMC and HMC match reasonably well with the benchmark values. This indicates that the DFHMC preserves the equilibrium distribution.

To assess the global convergence rate of the two methods the ensemble average of y^2 and x^2 is calculated at varying HMC steps and the percentage error from the benchmark average is plotted in Fig-3.21. The average used as the benchmark is calculated via numerical integration to be $< x^2 >= 0.0098\ angstrom^2$ and $< y^2 >= 0.0098\ angstrom^2$

As can be seen from Fig-3.21, the convergence of the DFHMC to the true ensemble average is faster than the standard HMC method. This indicates that the DFHMC method is able to enhance the sampling along a direction that is a linear combination of

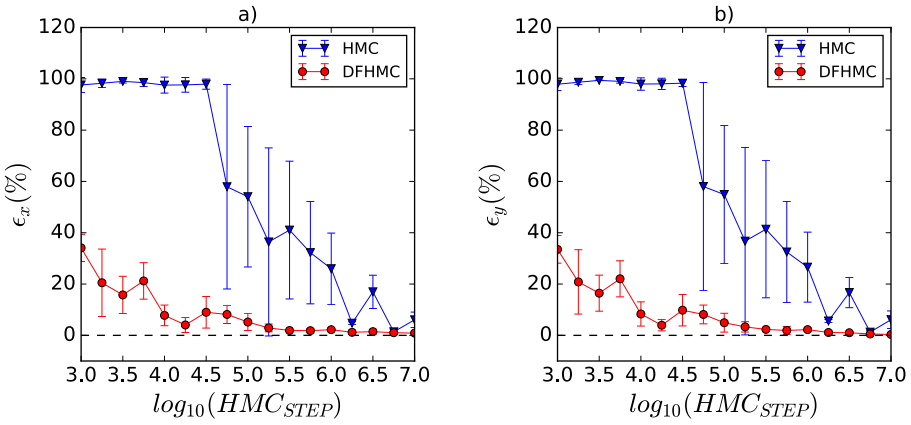


Figure 3.21 **Convergence of x^2, y^2 ensemble average for the rotated Double Well system.** Mean absolute percentage error of **a)** $< x^2 >$ and **b)** $< y^2 >$ over 4 independent simulations with respect to the log_{10} of HMC Steps

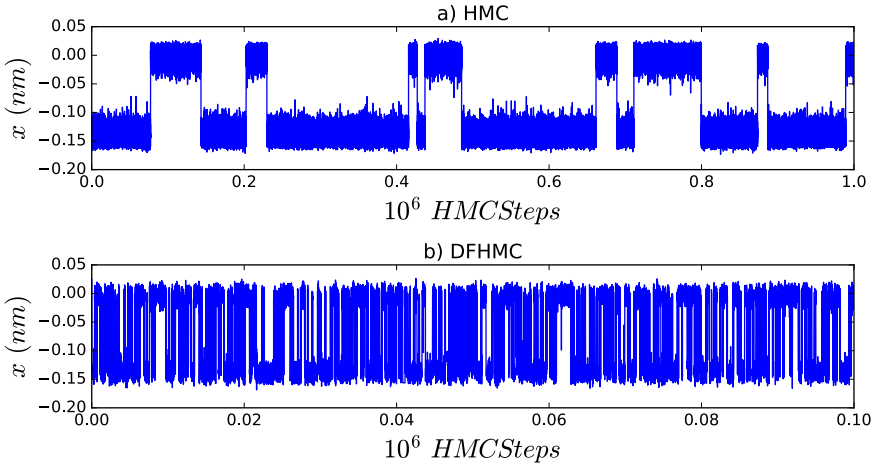
the cartesian coordinates. The variations across repeated simulations for the DFHMC are smalled and decreases faster compared to the HMC, as seen by the smaller error bars.

A representative trajectory along x and y obtained from the HMC and DFHMC simulation is shown in Fig-3.22, Fig-3.23. Similar to the double well system in the previous section where the energy barrier is aligned along the y coordinate, in the rotated double well system the DFHMC jumps between the local minima in a faster rate compared to HMC. The HMC method spends more time sampling from the local minima region before crossing the energy barrier.

To compare the performance of the Double Well system with the rotated Double Well system, y^2_{rot} ensemble average is calculated for the rotated Double Well system. This way the ensemble average for both of the systems will be the same and the convergence of this ensemble average can be compared.

As can be seen from Fig-3.24, The DFHMC method converge significantly faster than the HMC method for both the non-rotated and rotated double well potential system. Furthermore, the speed of convergence is nearly the same for the two different systems for both the DFHMC and HMC method. This implies that the DFHMC method is able to add energy along a direction that is a linear combination of the cartesian coordinates, and the convergence performance is not affected by the rotational transformation of the energy surface.

To understand how DFHMC and HMC work in the cases where the energy barrier is along a cartesian coordinate and along a linear combination of the cartesian coordinates, the direction of the velocity vector that is picked at each hybrid Monte Carlo steps are analysed. To measure the direction of the drawn velocities at the beginning



Trajectory of x coordinate for the Double Well system. The simulation trajectory of x for DFHMC and HMC simulation. The first 10^5 and 10^6 HMCSteps are shown for DFHMC and HMC respectively. Simulation parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

Figure 3.22

of each HMC step the angle α is introduced as follows,

$$\alpha = arctan(v_y/v_x) \tag{3.28}$$

Where v_y and v_x represents the velocities along x and y coordinates respectively. αrepresents the angle of the velocity vector (projected on the $x-y$ surface) that it makes with the x axis. The angle α captures only the axis of the velocity vector. To also capture the sign of the direction the quadrant in which the velocity vector falls in, can be identified by looking at the signs of v_x and v_y. Thus,

if $v_x > 0$ and $v_y > 0$ then it is in the 1^{st} Quadrant
if $v_x < 0$ and $v_y > 0$ then it is in the 2^{nd} Quadrant
if $v_x < 0$ and $v_y < 0$ then it is in the 3^{rd} Quadrant
if $v_x > 0$ and $v_y < 0$ then it is in the 4^{th} Quadrant

The α angle and the quadrants are calculated from the velocity trajectory obtained from the HMC and DFHMC simulations, that are run before for the convergence analysis. The velocity trajectory corresponds to the velocities obtained after the result of the acceptance and rejection criterion. Thus the direction of velocities obtained from the trajectory is expected to be distorted based on the energy surface by the molecular dynamics runs within each HMCstep. However, if the DFHMC method is significantly enhancing the simulation by increasing the rate of barrier crossing, we would also expect that the probability of the velocity trajectories to be aligned with the direction of the barrier to be much higher for the DFHMC than the HMC method.

As can be seen from Fig-3.25 and Fig-3.26 the α angle of the velocity at each HMC step is very close to a uniform distribution for the standard HMC method. The de-

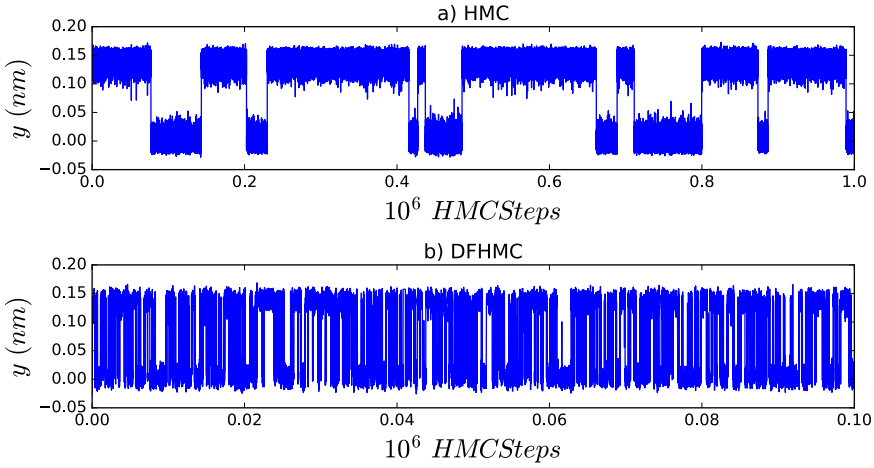
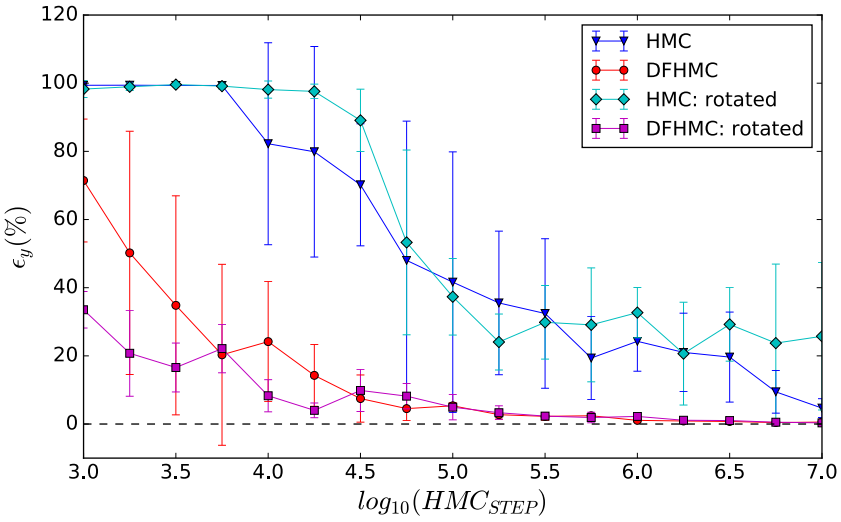


Figure 3.23 **Trajectory of y coordinate for the Double Well system.** The simulation trajectory of y for DFHMC and HMC simulation. The first 10^5 and 10^6 HMCSteps are shown for DFHMC and HMC respectively. Simulation parameters: HMCStep = 10^7, MDStep = 30, dt = $4fs$, scl=5, nstfilter = 100, temperature = $300K$

viation from the uniform distribution is caused by the molecular dynamics simulation runs within each HMC step. It can be seen that the distortions to the uniform distributions make the velocity direction more favorable when it is closer to the direction of the energy barrier. However, the difference in the probability of the velocity directions is still very small compared to the distribution seen for the DFHMC method. In the DFHMC method the angle is more likely to be around -90 and 90 degrees when energy barrier is along $y = 0$ line and -45 degrees when it is along $y = -x$. The α angle being 90 or -90 degrees corresponds to the velocity being in the direction of the y axis which is also the direction of the energy barrier. And similarly α being -45 degrees corresponds to the velocity being on the $y = -x$ line which also aligns with the direction of the energy barrier for the rotated system. This shows that the DFHMC is able to increase the velocities along the direction of the energy barrier compared to other possible directions. On the other hand the HMC method does not add energy to a particular direction; instead, it adds energy to any direction with equal probability.

The quadrant information for the double well system where the energy barrier is along the y axis is not very useful in capturing the differences between the HMC and DFHMC method Fig-3.27. As per the definition, the quadrants does not include the cases where the direction aligns on an axis. Therefore on a system where the energy barrier is along the y axis, we would expect that the velocity direction to be on the y axis, as validated by the α angle distribution graph Fig-3.25. Thus making the quadrants bar chart uniform. On the other hand for the system where the energy barrier is along $y = -x$ axis the Fig-3.28 shows that the DFHMC method generates with



Convergence of the percentage error of the y^2 ensemble average. Compares the convergence of the percentage error in the y^2 ensemble average between HMC and DFHMC methods for the rotated (barrier is along $y = -x$) and non-rotated (barrier is along $y = 0$) doublewell systems. Simulation paramaters: HMCSteps= 10^7, dt= 4 fs

Figure 3.24

higher probability velocities that fall into the 2^{nd} and 4^{th} quadrant whereas the standard HMC method does not have a preference in which quadrant the velocities will be in. As in the α angle distribution graphs Fig-3.25, Fig-3.26 we can see a distortion in the uniformity for the HMC, which is caused by the molecular dynamics simulations run within each HMC step.

To summarise, It has been shown that on a system where the energy barrier is along the cartesian coordinate and a linear combination of the cartesian coordinates the DFHMC method is able to enhance the sampling much better than the HMC method. Furthermore, the convergence performance of DFHMC and HMC did not change significantly by the rotation of the cartesian coordinates, given the energy surface has the same curvatures. It is observed that the direction of velocities generated by DFHMC aligns along the energy barrier with higher probability, whereas in HMC there is no significant preference of the direction of the velocities.

In the next section, the performance of the DFHMC method is investigated on a surface where the energy barrier is along a curved path. Thus the capability of the DFHMC method to adjust the direction of the biased velocity is tested in this setup.

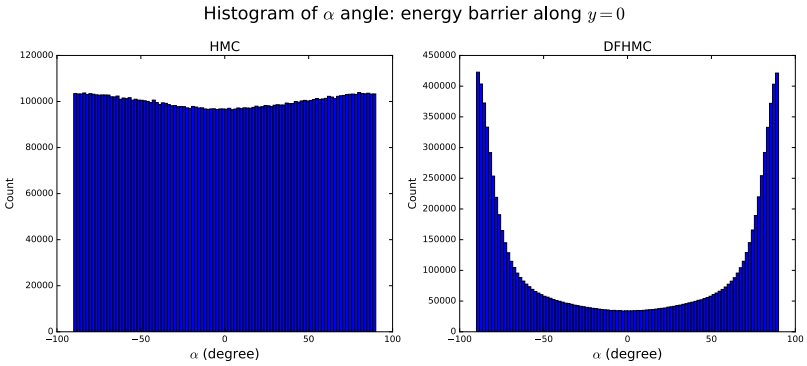


Figure 3.25 **Histogram of α angle: energy barrier along $y = 0$.** The total number of velocities generated is equal to HMCSteps= 10^7. The system id the double well potential where the barrier is along the *y* axis

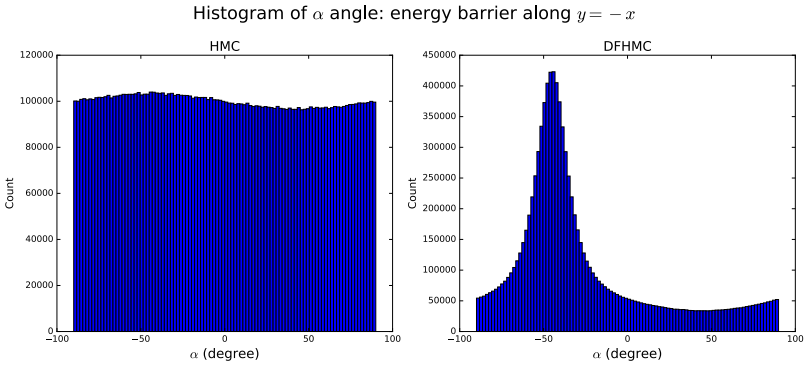
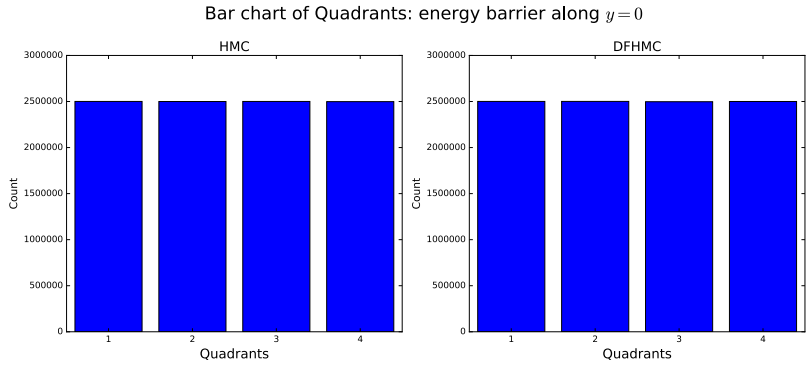
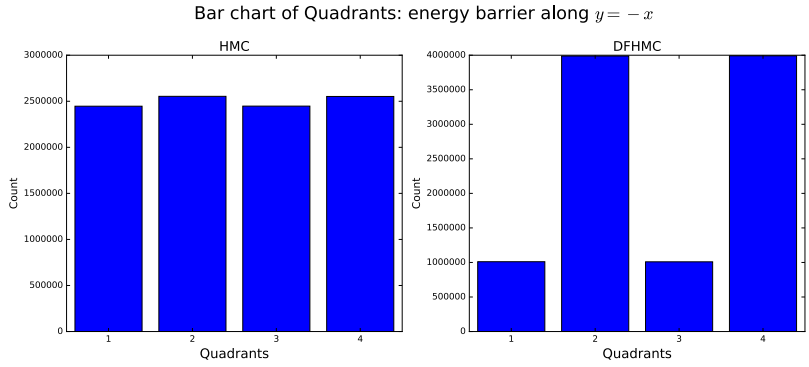


Figure 3.26 **Histogram of α angle: energy barrier along $y = -x$.** The total number of velocities generated is equal to HMCSteps= 10^7



Barchart of Quadrants: energy barrier along $y = 0$**.** The total number of velocities generated is equal to HMCSteps= 10^7

Figure 3.27



Barchart of Quadrants: energy barrier along $y = -x$**.** The total number of velocities generated is equal to HMCSteps= 10^7

Figure 3.28

3.4 Curved Potential

In this section, we investigate the sampling performance of the DFHMC method on a surface with a curved transition path. The aim is to find out whether the DFHMC method is able to enhance the simulation in systems where the direction along the energy barrier changes.

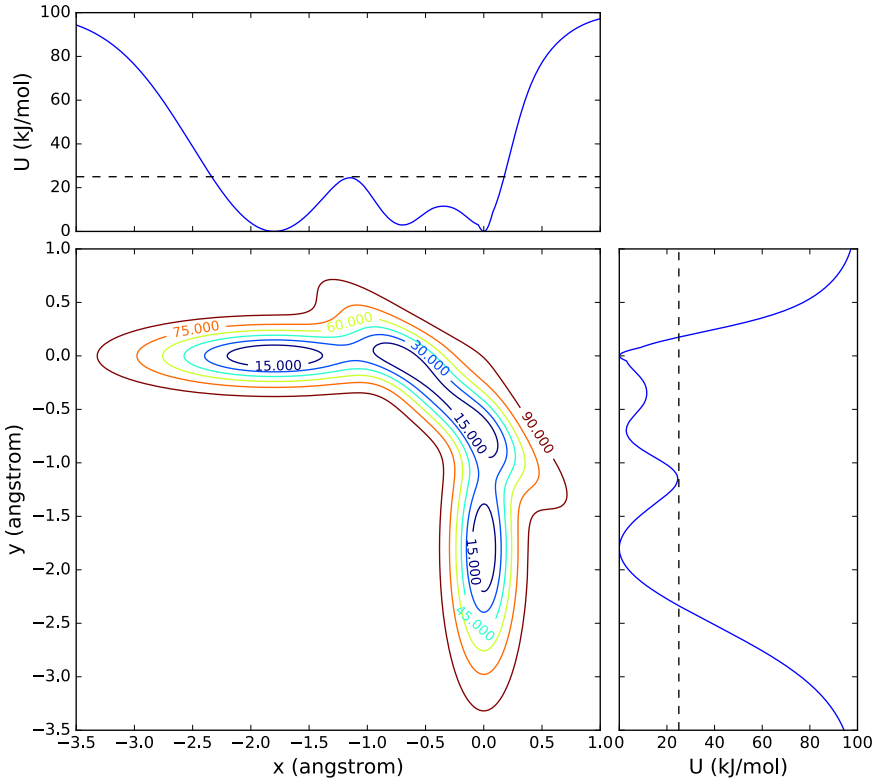


Figure 3.29 **Potential energy surface with curved transition path.** The graph shows the equal potential energy lines, isoline graph of the curved energy surface. On the top and right side of the figure the projections of the surface on to the y and x axis are drawn.

The curved potential is generated by combining three, two dimensional Gaussian distributions, a, b and c as given below. The constant, 99.99 is added to the potential to align the minimum of the potential surface approximately to zero.

$$U(x, y) = -85a - 100b - 100c + 99.99 \tag{3.29}$$

The Gaussians are defined as follows,

$$
\begin{aligned}
a &= e^{-16(x_{rot}+0.5)^2 - y_{rot}^2} \\
b &= e^{-16x^2-(y+1.8)^2} \\
c &= e^{-(x+1.8)^2-16y^2}
\end{aligned}
\tag{3.30}
$$

The Gaussians b and c are centered at $(x, y) = (0, -1.8)$ and $(x, y) = (-1.8, 0)$ coordinates. b is narrower along x compared to y where as c is narrower along y compared to x. The gaussian a is narrower along $y = x$ axis compared to $y = -x$. To make a narrower along $y = x$, a rotational transformation is done on x and y to be used on the gaussian a.

$$
\begin{aligned}
y_{rot} &= -x\sin(\theta) + y\cos(\theta) \\
x_{rot} &= x\cos(\theta) + y\sin(\theta)
\end{aligned}
\tag{3.31}
$$

where $\theta = \frac{45\pi}{180}$ to rotate x, y coordinates by 45 degrees. This allows adjusting the width of the Gaussian a along $y = x$ and/or $y = -x$ axis. The obtained energy surface can be seen in Fig-3.29. The surface has two minima where the simulation has the potential to get stuck, one is located at $(x, y) = (0, -1.8)$ and the other at $(x, y) = (-1.8, 0)$. For the simulation to escape these minima a barrier of height $25kJ/mol$ needs to be crossed as can be seen from the projections of the surface onto x and y coordinates.

The surface is designed to have a wider curvature along the transition path compared to other directions. Thus the low frequency motion of a particle moving on this surface will be along the transition path. To understand the frequency characteristics of a particle moving on this surface and NVE (constant total energy) simulation is run where the starting position is at $(x, y, z) = (0, -1.2, 0)$. This makes the starting potential of the system equal to $24.25kJ/mol$ together with the Kinetic energy the total energy is enough for the simulation to explore all the minima Fig-3.30, thereby allowing us to capture a good representation of the possible frequencies that can occur on this surface.

Fig-3.31, shows the frequency spectrum calculated as the average Fourier transform of the NVE simulation's velocity trajectory along each degree of freedom. It can be seen that there is a clear separation between the low and high frequency motions. And the low frequency motion can be enhanced by a low pass filter with the band $0 - 0.08$ as a normalised frequency at timestep $6fs$, which corresponds to the range $0 - 222\ cm^{-1}$ in wavenumbers Eq-3.32.

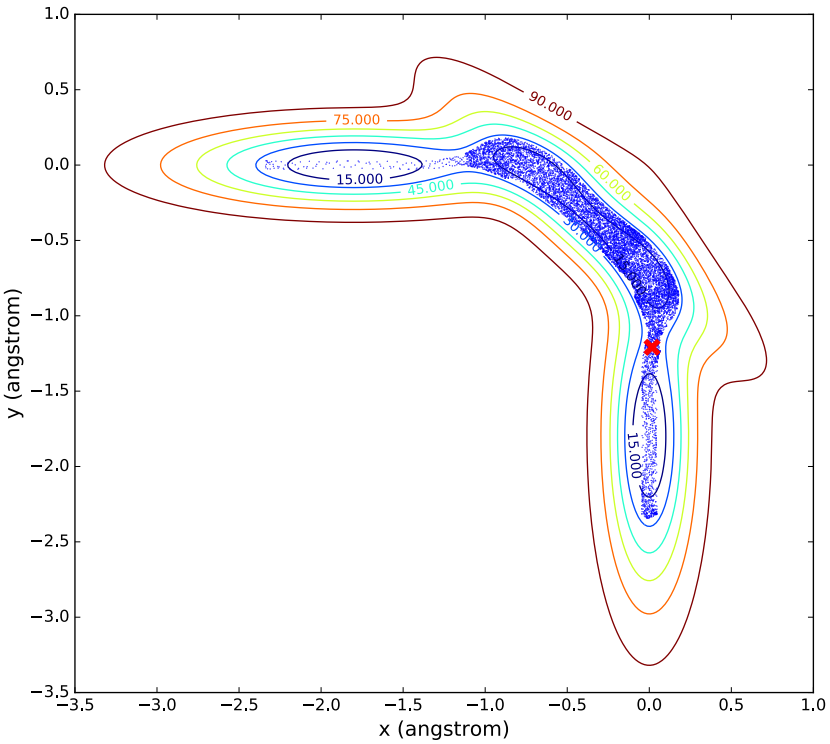
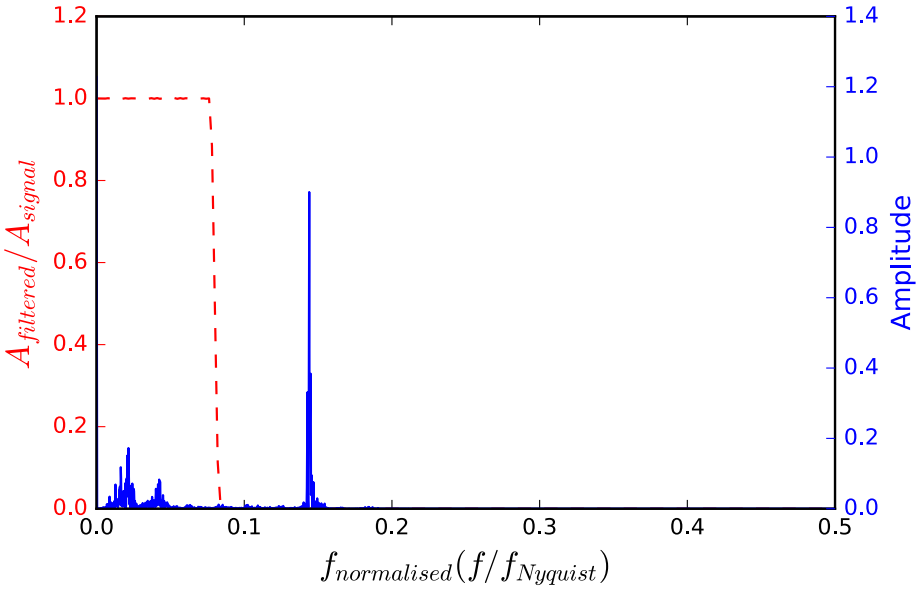


Figure 3.30 **trajectory of an NVE simulation on curved system.** The dots shows the positions explored by the NVE simulation. The red cross shows the starting position. Simulation paramaters: MD Steps = 10^4, MD timestep dt = 6 femtosecond, starting position $(x_0, y_0, z_0) = (0.0, -1.2, 0.0)$ angstroms, temperature=300K

$$
\begin{aligned}
f &= f_{normalised} \times f_{Nyquist} \\
&= \frac{f_{normalised}}{2dt} \\
&= \frac{0.08}{2 \times 6} \quad fs^{-1} = \frac{0.08}{12c} \quad cm^{-1} \\
&= \frac{0.08}{12 \times 3 \times 10^{-5}} \quad cm^{-1} \\
&= 222 \quad cm^{-1}
\end{aligned}
\tag{3.32}
$$

where c is the speed of light in cm/fs units, 3×10^{-5} cm/fs

To test the performance of the DFHMC method in this curved surface, four independent HMC and DFHMC simulations are run. The DFHMC simulation is used with a filter that enhances the frequencies between $0 - 222$ cm^{-1} and the filtered velocities are scaled by a factor of $scl = 2$ and updated every 100 HMC step. The



Power Spectrum curved system. Power Spectrum (in blue and solid line) of NVE MD velocity trajectory. Frequency response (in red, dashed line) of the designed low pass filter ($f_{normalised} = 0.08$) that removes high frequency from the signal. Simulation parameters: MD Steps = 10^4, MD timestep dt = 6 femtosecond, starting position $(x_0, y_0, z_0) = (0.0, -1.2, 0.0)$ angstroms.

Figure 3.31

simulations are run at $300K$, with $12fs$ timestep, 10 MD steps within each HMC step and 10^7 HMCSteps. Fig-3.32 shows a representative HMC and DFHMC simulation where the PMF along y and x is calculated, the solid black line is the PMF result obtained from numerical integration and is used as the benchmark. In general, both HMC and DFHMC match quite well with the benchmark results. However, we observe a small deviation of DFHMC from the benchmark at the well and the barrier between $-0.1 < x < 0nm$ and $-0.1 < y < 0nm$, indicating the DFHCM simulation did not sample well that region. Looking at the surface in Fig-3.29, $-0.1 < x < 0$ nm and $-0.1 < y < 0$ nm region corresponds to the transition state, where the energy barrier to transition to one of the lower energy minima is relatively low. Thus the DFHMC method does not spend much time in the transition state; instead, it transitions to the lower state thereby reducing the sampling performance of the transition state itself. Fig-3.33 and Fig-3.34 shows the trajectories obtained from the HMC and DFHMC simulations along x and y coordinates, and it confirms that the DFHMC has a much higher rate of transitioning between states compared to HMC thus enhancing global sampling. However, this also results in the DFHMC spending less time in sampling the local state thereby reducing the local sampling performance. This behavior of DFHMC is also consistent with the observations made on the double well and rotated double well system explained in the previous sections.

Fig-3.35 shows the sampling reached at 10^4 HMCStep for a representative simula-

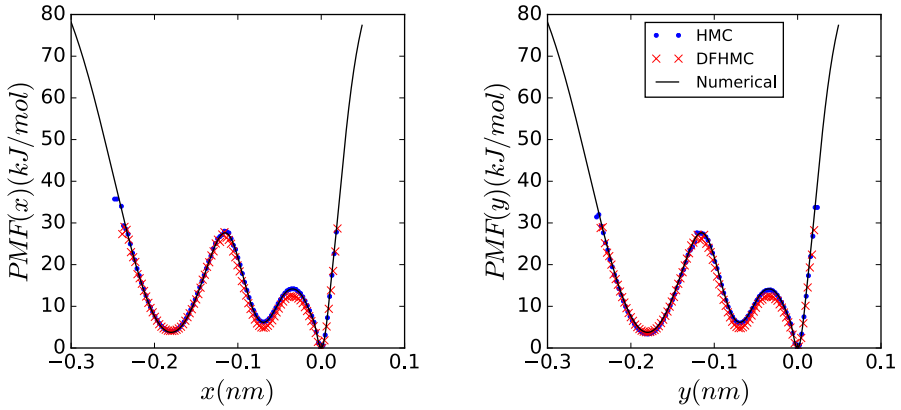


Figure 3.32 **PMF along x and y coordinates for the curved potential system.** Solid black line is the PMF obtained via numerical integration. The blue and red dots represents the results obtained via HMC and DFHMC simulation respectively. Simulation parameters: HMCStep = 10^7, MDStep = 10, dt = $12fs$, scl=2, nstfilter = 100, temperature = $300K$

tion of HMC and DFHMC. The starting positions for both HMC and DFHMC are the same and lies in one of the local minima shown as the red cross in the figure. It can be seen that where the DFHMC is able to cross the energy barriers and explore all local minima states the HMC method is still sampling the minima in the vicinity of the starting position. Thus the DFHMC method is able to adapt to the directional change of the energy barrier and add energy along the curved transition path.

To assess the global convergence, the ensemble averages $< x^2 >$ and $< y^2 >$ are calculated at several HMCSteps for four independent HMC and DFHMC simulations. Fig-3.36 shows the average calculated over 4 independent simulations of the ensemble average $< x^2 >$ and $< y^2 >$ with respect to $log_{10}(HMC_{STEP})$. As can be seen from the figure in both $< x^2 >$ and $< y^2 >$ the DFHMC method converges much faster to a stable value compared to the HMC simulation. Note that the HMC simulation does not update the ensemble average value significantly for a relatively long time indicating that the simulation is trapped in a local minimum.

To summarise, the DFHMC and HMC methods are simulated on a system with a curved transition path. The DFHMC method has shown enhanced global sampling compared to the HMC method. The convergence rate for DFHMC is much higher than the HMC method. Both methods generated the equilibrium distribution reasonably well. However, the DFHMC method showed a slight difference from the benchmark PMF around the transition state indicating the local sampling of DFHMC suffers from the performance increase in the global sampling.

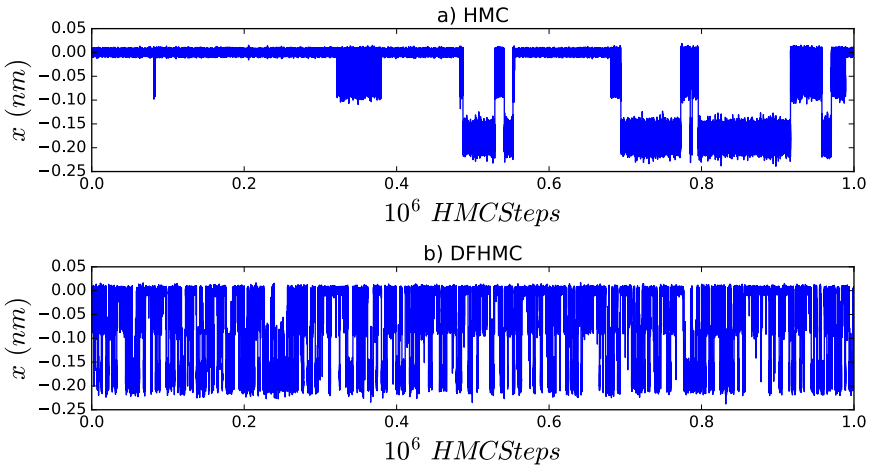


Figure 3.33 **Trajectory of x coordinate for HMC and DFHMC simulation on curved potential.** The simulation trajectory of x for DFHMC and HMC simulation. The first 10^6 HMCSteps are shown for DFHMC and HMC respectively. Simulation parameters: HMCStep = 10^7, MDStep = 10, dt = $12fs$, scl=2, filter update = every 100 HCMSteps, temperature = $300K$

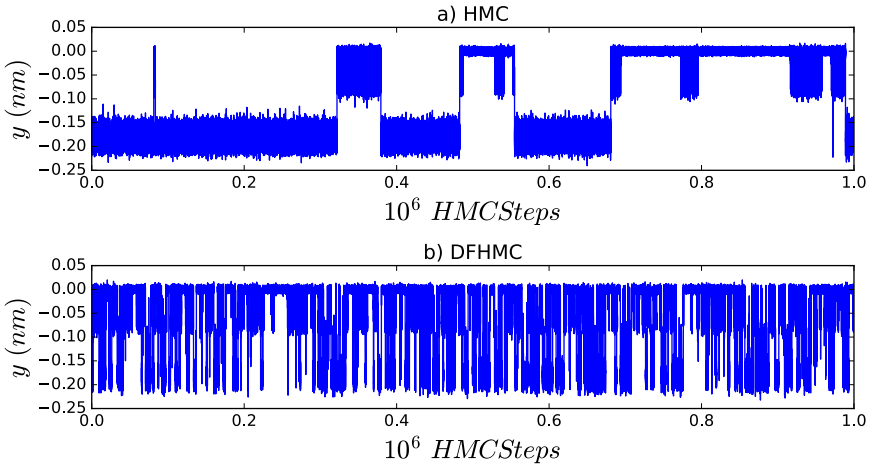


Figure 3.34 **Trajectory of y coordinate for HMC and DFHMC simulation on curved potential.** The simulation trajectory of y for DFHMC and HMC simulation. The first 10^6 HMCSteps are shown for DFHMC and HMC respectively. Simulation parameters: HMCStep = 10^7, MDStep = 10, dt = $12fs$, scl=2, filter update = every 100 HCMSteps, temperature = $300K$

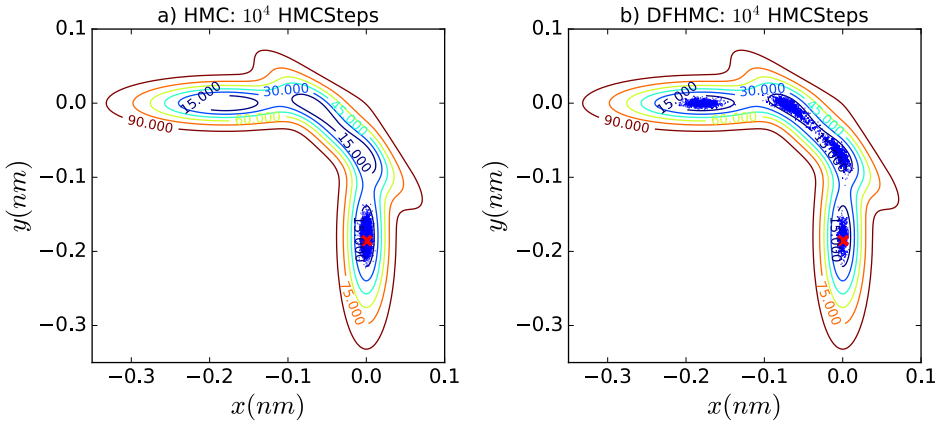


Figure 3.35 **HMC and DFHMC trajectory at 10^4 HMC step for curved potential.** Shows the extend of sampling occured by **a)** HMC and **b)** DFHMC at 10^4 HMC steps. the red cross shows the starting positions

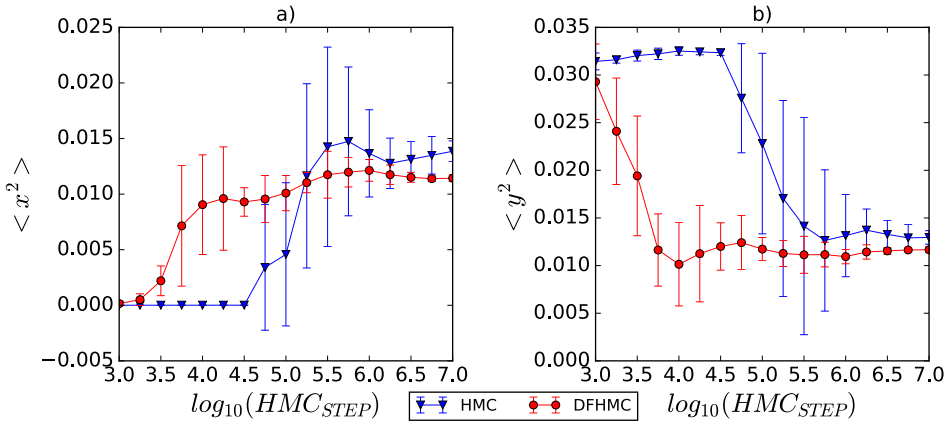
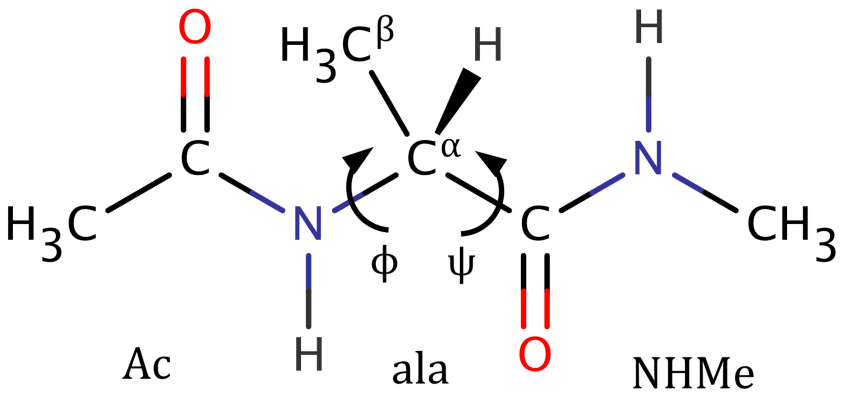


Figure 3.36 **Convergence of the x^2 and y^2 ensemble average for the curved potential system.** Compares the convergence of the x^2, y^2 ensemble average between HMC and DFHMC methods for the curved potential system. Simulation paramaters: HMCSteps= 10^7, dt= 12 fs, MDSteps=10, temperature=300K, additional parameters for DFHMC: $scl = 2$, filter update = every 100 HMCsteps, filter=0 – $222cm^{-1}$ low pass filter.

3.5 Alanine dipeptide

As mentioned before the DFHMC method has an application on biomolecular systems especially on proteins. Thus, alanine dipeptide, which represents a peptide/protein much better but still is relatively simple is chosen as a test molecule. Several other enhanced sampling methods such as Momentum Enhanced HMC [2], Metadynamics [4] also used alanine dipeptide for testing their methods at some stage.
Alanine dipeptide, as shown in Figure-3.37 consists of 22 atoms and two peptide bonds corresponding to the ϕ and ψ dihedral angles. The molecule in itself is not a peptide, but the presence of the two peptide bonds captures some of the backbone properties of peptides/proteins. The dihedral angles on the peptide bonds especially the ϕ angle, for this system are the slow degrees of freedom which cause the system to get trapped in local minima for relatively long simulation times.



Alanine dipeptide, molecule. Molecular representation of alanine dipeptide

Figure 3.37

In this section, the sampling performance of DFHMC on alanine dipeptide molecule in explicit and implicit water is analysed. The system in implicit water is more prone to develop a centre of mass rotation, whereas in the explicit water the water molecules may reduce the centre of mass rotation of the molecule by providing friction. Therefore to analyse the effect of the rotation on the sampling performance of DFHMC, the alanine dipeptide in the implicit water is further investigated by constructing a DFHMC and HMC simulation where at each HMC step the centre of mass rotation

	Equilibration Simulations			
	NPT (Non H Restraint)	NPT (CARestraint)	NPT	NVT
MDStep	10^5	1.5×10^5	10^6	10^6
dt (fs)	2	2	2	2
temperature (K)	300	300	300	300
pressure (atm)	1	1	1	-
frictioncoeff (ps^{-1})	1	1	1	1
nstout	100	100	100	100

Table 3.1 **Simulation parameters for equilibration, alanine dipeptide explicit water.**

and translation is removed from the velocities.

3.5.1 Explicit Water

A new simulation system has been constructed by adding water molecules to the Alanine dipeptide. Thus before running any production simulation the system has been run through an equilibration process.

To equilibrate the water and hydrogen atoms in the system, an NPT simulation has been run where apart from hydrogen and water molecules harmonic position restraints are applied to all atoms. Then to equilibrate atoms apart from the backbone of the molecule, an NPT simulation with a harmonic position restraint only applied to backbone CA (carbon alpha) atom has been run. After that, an NPT simulation followed by an NVT simulation has been run without any restraints to complete the equilibration process.

The parameters used for each simulation in the equilibration process is given in Table-3.1

For the production NVT, HMC and DFHMC simulations are run. see Table-3.2 for simulation parameters.

A metadynamics simulation has been run to be used as the benchmark for obtaining the potential of mean force along ϕ and ψ dihedral angles. The parameters for the metadynamics simulations are given in Table-3.3

Fig-3.38 shows the potential of mean force (PMF) calculated along the ϕ and ψ dihedral angles. The result from MetaDynamics simulation is used as the benchmark. Overall the PMF of NVT, HMC and DFHMC matches well with the benchmark. However specific regions especially along the ϕ dihedral angle is not converged fully to the benchmark distribution. Indicating the need for longer simulations for NVT, HMC and DFHMC to fully equilibrate. In this particular simulation, we can also see that the DFHMC is better in crossing the energy barrier between $20 - 25 kJ/mol$ along the ϕ

	Production Simulations		
	NVT	HMC	DFHMC
MDStep	80×10^6	80	80
dt (fs)	2	2	2
temperature (K)	300	300	300
frictioncoeff (ps^{-1})	1	-	-
HMCStep	-	10^6	10^6
nstfilter	-	-	100
filter (cm^{-1})	-	-	100
atom (index)	-	-	4,6,8,14,16
scl	-	-	2
nstout	80	1	1

Simulation parameters for production, alanine dipeptide explicit water. 4^{th}, 6^{th}, 8^{th}, 14^{th}, 16^{th} atoms are part of the ϕ and ψ dihedral angles

Table 3.2

dihedral. We also see that the NVT implementation is sampling very well compared to the HMC and DFHMC methods. Langevin dynamics is used as the NVT implementation and is run for 80×10^6 Steps. Even though the number of step in NVT is 80 times higher than the HMC and DFHMC simulations, it equates to roughly the same number of force calculations thereby having a similar CPU/GPU run time. The Langevin implementation falls into the stochastic simulation category like HMC and DFHMC, thus having more steps to sample may result in similar if not better sampling with HMC and DFHMC.

Fig-3.39 shows one example of the sampling achieved by the DFHMC and HMC on the ϕ and ψ dihedral space. We can observe that the DFHMC has more extensive sampling compared to HMC. However, the sampling performance difference is quite small compared to the performance gains observed in the systems analysed in the previous sections.

Similar to the previous section, an ensemble average of ϕ^2 and ψ^2 is calculated at different HMC steps and are averaged over four independent simulations that have different starting conditions. Fig-3.40, shows the convergence of ensemble averages with respect to HMC step for the HMC and DFHMC simulations. Although the ensemble average metric does not change much from the converged value, we can see along the ψ dihedral angle that there is not much difference in the convergence speed whereas along ϕ we see a slightly better convergence performance for DFHMC.

To summarise, In this section, the alanine dipeptide system is equilibrated in an explicit water environment. Independently repeated simulation with different starting conditions are run for HMC and DFHMC methods. Although the difference is small, we observe that the DFHMC is sampling slightly better than the HMC method, espe-

	Metadynamics Simulations
MDStep	2×10^7
dt (fs)	2
temperature (K)	300
frictioncoeff (ps^{-1})	1
Pace	500
Height	1.2
Sigma	0.35
biasfactor	10
CV	ϕ, ψ
nstout	100

Table 3.3 **Simulation parameters for MetaDynamics, alanine dipeptide explicit water.**

cially along the ϕ dihedral angle. We also observed that the Langevin implementation of the NVT simulation performed very well.

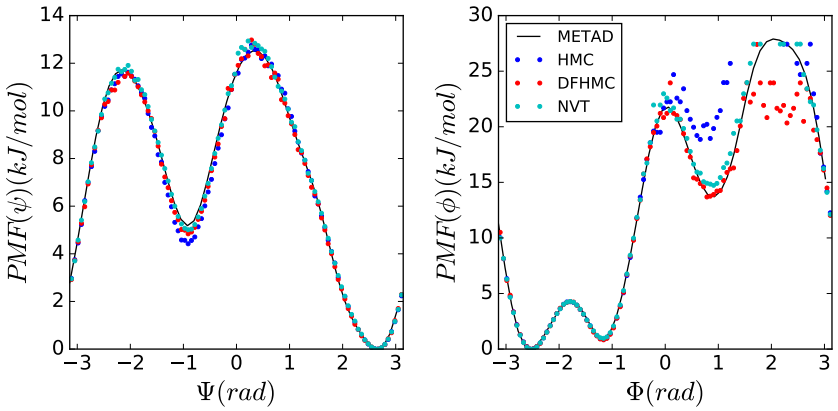


Figure 3.38

PMF along ϕ and ψ dihedral angles for alanine dipeptide molecule simulated in explicit water. Solid black line is the PMF ontained via Metadynamics simulation. The blue and red dots represents the result obtained via HMC and DFHMC simulation respectively. And the dots in cyan represent the result from NVT simualtion. Simulation Parameters: for Metadynamics see Table-3.3 and for NVT, HMC, DFHMC see Table-3.2

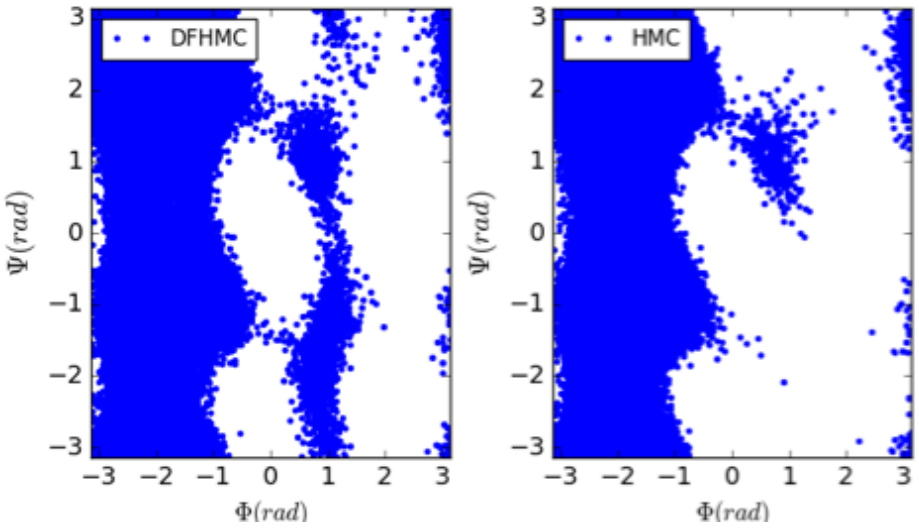


Figure 3.39

ϕ - ψ plot of alanine dipeptide in explicit water - DFHMC vs HMC.

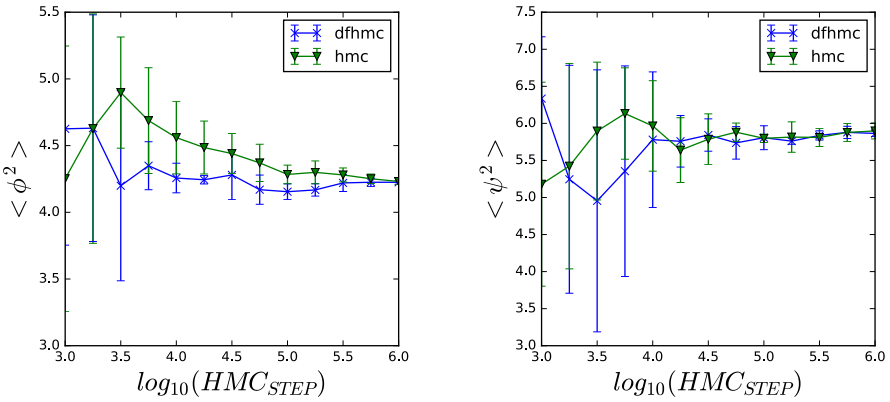


Figure 3.40 **Convergence of the ϕ^2 and ψ^2 ensemble average for alanine dipeptide in explicit water.** Compares the convergence of the ϕ^2, ψ^2 ensemble average between HMC and DFHMC methods for the alanine dipeptide in explicit water system. The graph is obtained by averaging over 4 independent simulation repeats and the error bars represent the standard deviation of the ensemble average. Simulation paramaters: see Table-3.2

	Production Simulations		
	NVT	HMC	DFHMC
MDStep	48×10^7	80	80
dt (fs)	2	2	2
temperature (K)	300	300	300
frictioncoeff (ps^{-1})	1	-	-
HMCStep	-	6×10^6	6×10^6
nstfilter	-	-	1000
filter (cm^{-1})	-	-	100
atom (index)	-	-	4,6,8,14,16
scl	-	-	2
nstout	80	1	1

Simulation parameters for production, alanine dipeptide implicit water. 4^{th}, 6^{th}, 8^{th}, 14^{th}, 16^{th} atoms are part of the ϕ and ψ dihedral angles

Table 3.4

3.5.2 Implicit Water

In this section, we investigate the DFHMC method on the same alanine dipeptide molecule but in a setup with a smaller number of degrees of freedom, where the explicit waters are modeled by Generalized Born implicit water model. In this setup the physical presence of the water molecules are eliminated but, the effect of the water on the alanine dipeptide's equilibrium distribution is approximately conserved.

Since no explicit water is present in this setup, the equilibration protocol used in the previous section is not needed. Thus production simulation is run for NVT, HMC and DFHMC simulation directly. For each method, four independent simulations are run. The simulation parameters used are defined in Table-3.4

A metadynamics simulation has been run to be used as the benchmark for obtaining the potential of mean force along ϕ and ψ dihedral angles. The parameters for the metadynamics simulations are given in Table-3.5

The convergence of the simulations to the equilibrium distribution is measured by comparing the PMF along ϕ and ψ dihedral angle against a benchmark. For the benchmark, a metadynamics simulation is run. Fig-3.41 shows the PMF along ϕ and ψ dihedral angle. The simulations for the alanine dipeptide in implicit water are run for longer steps compared to the simulation run in explicit water see Table-3.4, and Table-3.2 for reference. Thus the simulations are converged to the equilibrium (benchmark PMF) much better. However in energies between $25 - 30 kJ/mol$ along the ϕ dihedral the simulations result are still noisy indicating the sampling in that region has not been converged yet. The result for HMC, NVT and DFHMC also match quite well with each other, and we observe all methods deviate by roughly the same amount from the benchmark at the higher energy local minima between $4 - 6 kJ/mol$ along ψ di-

	Metadynamics Simulations
MDStep	10^7
dt (fs)	2
temperature (K)	300
frictioncoeff (ps^{-1})	1
Pace	500
Height	1.2
Sigma	0.35
biasfactor	10
CV	ϕ, ψ
nstout	100

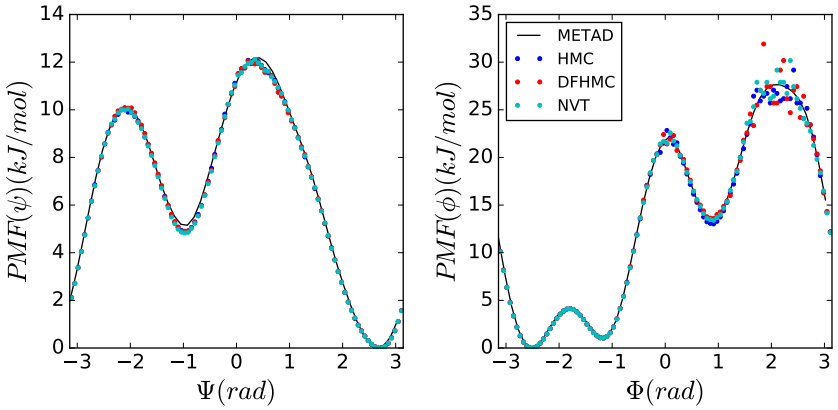
Table 3.5 **Simulation parameters for MetaDynamics, alanine dipeptide implicit water.**

hedral, either indicating the benchmark is not quite reached equilibration or the HMC, NVT and DFHMC methods have not been converged yet. However, since the differences are small and the DFHMC is consistent with the HMC and NVT which are known to generate the equilibrium canonical distribution, we can conclude that the DFHMC is also able to generate the equilibrium distribution.

Fig-3.42 shows the sampling of ϕ and ψ dihedral angle for one HMC and DFHMC simulation. It can be seen that the sampling of DFHMC is slightly higher if not equal. Compared to the alanine dipeptide in explicit simulation, the performance difference of DFHMC with respect to HMC is even smaller.

To check the convergence rate more quantitatively the ensemble average of ϕ^2 and ψ^2 is calculated at different HMC steps for the HMC and DFHMC simulation over four repeated simulation runs. Fig-3.43 shows the convergence rate of the ensemble averages. The convergence rate for HMC and DFHMC are very similar. The HMC method stabilises earlier compared to DFHMC as can be seen by the decreasing error bars of the HMC method. Although the overall difference is small we observe that the HMC method has a better convergence performance.

In summary, The HMC and DFHMC simulations of the alanine dipeptide molecule in implicit water are able to generate the equilibrium NVT distribution. However, compared to the alanine dipeptide explicit system the enhancement of DFHMC versus HMC observed in the implicit water system is significantly lower. We note that the overall convergence performance for HMC is better compared to the DFHMC method in terms of the speed of generating stable results as observed by the decreasing error bars. Thus we can conclude that the DFHMC method performs worse in the implicit water system compared to the explicit water system. To explain a possible reason for the observed decline, we hypothesise that the observed difference in enhancement



PMF along ϕ and ψ dihedral angles for alanine dipeptide molecule simulated in implicit water. Solid black line is the PMF ontained via Metadynamics simulation. The blue and red dots represents the result obtained via HMC and DFHMC simulation respectively. And the dots in cyan represent the result from NVT simulation. Simulation Parameters: for Metadynamics see Table-3.5 and for NVT, HMC, DFHMC see Table-3.4

Figure 3.41

between the implicit and explicit water system is due to the difference of the centre of mass rotation of the alanine dipeptide molecule. The molecule in explicit water is less prone to centre of mass rotation due to the friction between the water molecules, and thus the added energy to the centre of mass rotation can be dissipated much faster and easier. At each HMC step, a new velocity is drawn, and some portion will contribute to the centre of mass rotation of the system. Since the whole system in the implicit water case is the alanine dipeptide molecule, rotational motion of the centre of mass is conserved within each HMC step due to the conservation of angular momentum. The rotation of the molecule will make the obtained filtered velocities to be out of phase with the coordinates, and thus the direction of the slow degree of freedom cannot be identified with the velocities. As a result, the extra energy we put in the system could be added in any direction, making the procedure in principle the same as the HMC simulation. Thus the benefit of the DFHMC to add energy into the direction of the energy barrier will negatively be affected if not lost by the rotation. Even though this makes the DFHMC the same as the HMC in terms of the direction of the energy added, the amount of energy that is added is generally higher in DFHMC due to the non zero velocity bias and the scaling parameter. Thus the DFHMC would add more energy to the wrong direction and hence would be expected to perform worse compared to HMC, which we also observed in the alanine dipeptide implicit simulation results.

In the next section, the same alanine dipeptide implicit water system is simulated with HMC and DFHMC, but at each HMC step, the centre of mass rotation and trans-

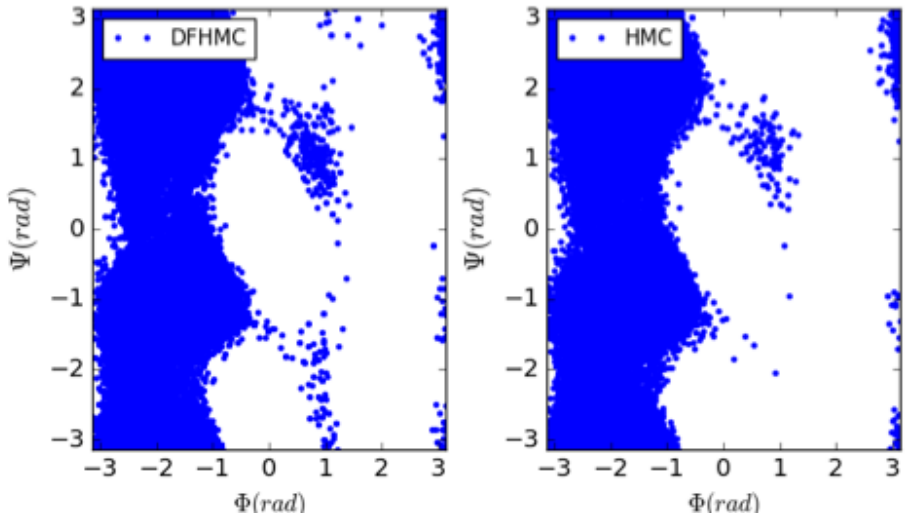
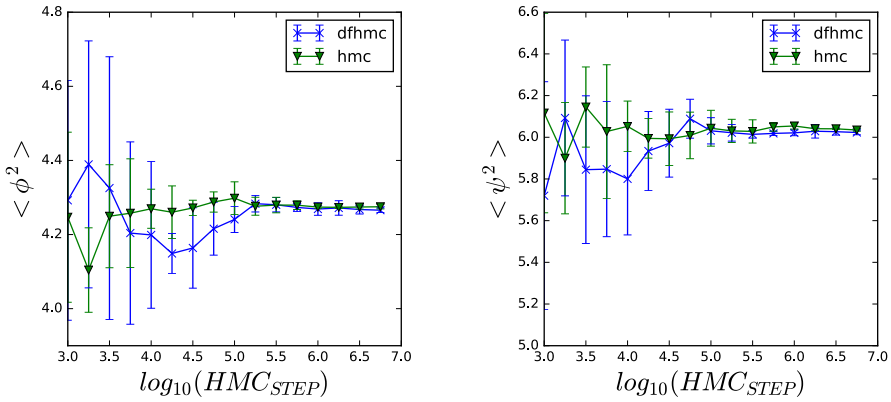


Figure 3.42 **ϕ - ψ plot of alanine dipeptide in implicit water - DFHMC vs HMC.**

lation is removed from the drawn velocities. Thus during the entire simulation, the alanine dipeptide molecule is prevented from rotating. Even though this interference with the simulation may disrupt the equilibrium property, our focus is to see the effect of the rotation on the convergence rate for the DFHMC method.



Convergence of the ϕ^2 and ψ^2 ensemble average for alanine dipeptide in implicit water. Compares the convergence of the ϕ^2, ψ^2 ensemble average between HMC and DFHMC methods for the alanine dipeptide in implicit water system. The graph is obtained by averaging over 4 independent simulation repeats and the error bars represent the standard deviation of the ensemble average. Simulation paramaters: see Table-3.4

Figure 3.43

	Production Simulations	
	HMC	DFHMC
MDStep	80	80
dt (fs)	2	2
temperature (K)	300	300
HMCStep	9×10^6	9×10^6
nstfilter	-	1000
filter (cm^{-1})	-	100
atom (index)	-	4,6,8,14,16
scl	-	2
nstout	1	1

Table 3.6 **Simulation parameters for production, alanine dipeptide implicit water.** COM mas rotation and translation is removed at every HMC step. 4^{th}, 6^{th}, 8^{th}, 14^{th}, 16^{th} atoms are part of the ϕ and ψ dihedral angles

3.5.3 Com rotation effect of DFHMC

In this section, the alanine dipeptide molecule is simulated with implicit water. However, at each HMC step, the centre of mass rotation is calculated and removed from the velocities thus preventing the molecule from rotating during the molecular dynamics step within each HMC step. The simulation parameters can be seen in Table-3.6. Note that the simulation parameters are kept the same with the simulations in the previous section where the centre of mass rotation was not removed from the alanine dipeptide molecule in implicit water. The only parameter difference from the previous section simulation is in the number of simulation steps, i.e., HMC steps. In this section, more HMC steps are used for the simulations to better assess the potential disruption of the equilibrium distribution by the removal of the centre of mass rotation.

If the rotation has the expected effect, then we should see that the DFHMC simulation where we removed the centre of mass rotation should enhance the sampling significantly more than the simulation where the centre of mass rotation is not removed. Note that by removing the centre of mass rotation after every draw of velocity, the simulation will not sample from the equilibrium distribution anymore. Since in this analysis the focus is on the sampling efficiency rather than the equilibrium distribution, the effect of the transformation of the velocities on the distribution is acceptable. However, removing the centre of mass rotation of the velocities at every HMC step cannot be proposed as a solution to overcome the rotation problem if one wants to conserve the equilibrium distribution.

Fig-3.44 shows the ensemble average of ϕ^2 and ψ^2 calculated at different HMC-Steps and averaged over 4 independent simulation for each method. The plot shows the results for the DFHMC (red cross) and HMC (yellow triangle) applied to the system

analysed in the previous section, i.e. alanine dipeptide in implicit water where the centre of mass rotation is not removed. Moreover, it also shows the result of the DFHMC (blue cross) and HMC (cyan triangle) applied to the system where the centre of mass rotation is removed. The first point to notice is that the DFHMC and HMC simulation where the centre of mass rotation is removed converges to a slightly different result as seen in the ensemble average ϕ. The convergence performance of all the simulations is very similar. However, the DFHMC where the centre of mass rotation is removed seems to converge slightly faster than the DFHMC where it is not removed. However, by just looking at this graph it is hard to say anything definite about the effect of the rotation.

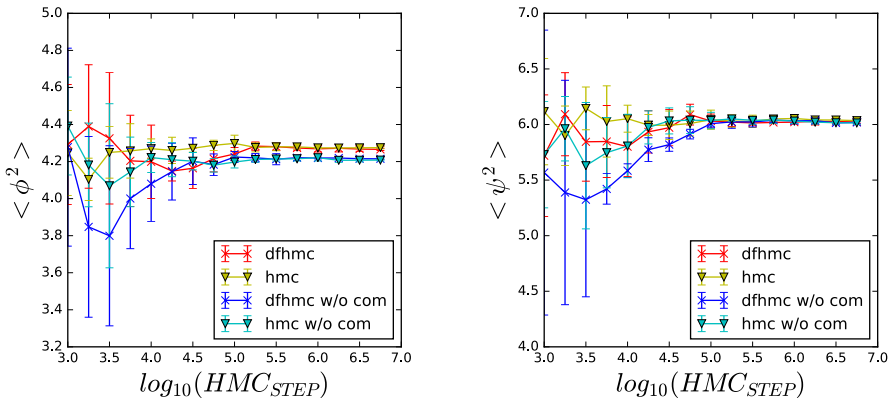


Figure 3.44

Convergence of the ϕ^2 and ψ^2 ensemble average for alanine dipeptide in implicit water, centre of mass rotation is removed every HCM step during the simulation. Compares the convergence of the ϕ^2, ψ^2 ensemble average between HMC and DFHMC methods for the alanine dipeptide in implicit water system. The graph is obtained by averaging over 4 independent simulation repeats and the error bars represent the standard deviation of the ensemble average. Simulation paramaters: see Table-3.6

To be able to see the potential effect of the rotation of the DFHMC simulation, a more strict way of assessing the convergence is designed. Since all simulations have sampling difficulties in the high energy minima along the ϕ dihedral the convergence measure will focus on that area. Thus a free energy difference between the high energy minima and the lower energy minima along the ϕ dihedral angle is calculated. More specifically the free energy region for the low energy minima (A) is defined to be between $-3 < \phi < -1$ and the high energy region (B) is defined to be between $0.5 < \phi < 1.5$. Then the free energy difference $F_A - F_B$ is calculated at different HMC steps to asses the convergence rate. The free energies F_A and F_B can be calculated from the PMF as follows,

$$F_A = -k_B T \ln \sum_{\phi \in [-3,-1]} \exp(-PMF(\phi)/k_B T)$$
$$F_B = -k_B T \ln \sum_{\phi \in [0.5,1.5]} \exp(-PMF(\phi)/k_B T) \tag{3.33}$$

Note the sum is taken over the regions specified as A and B.

Fig-3.45 shows the free energy difference, $F_A - F_B$ calculated at different HMC step, and is averaged over four independent simulations for each method. In this graph, the difference between the DFHMC with centre of mass rotation removed (blue cross) and the DFHMC where the centre of mass rotation is not removed (red cross) is more apparent. It can be seen that the removal of centre of mass rotation increases the convergence performance for the DFHMC method. The error bars are decaying faster and the convergence rate is quicker as well. We can also observe that the performance of HMC by the removal of the centre of mass rotation is not significantly affected. This is expected as the HMC method is adding energy uniformly to any direction and the potential energy of the system is independent of the centre of mass orientation, the rotation will not affect the sampling performance. Furthermore from the graph, it can be seen that the performance of the DFHMC where the centre of mass rotation is removed is slightly better than the HMC methods; however, the difference is minimal to make any conclusions.

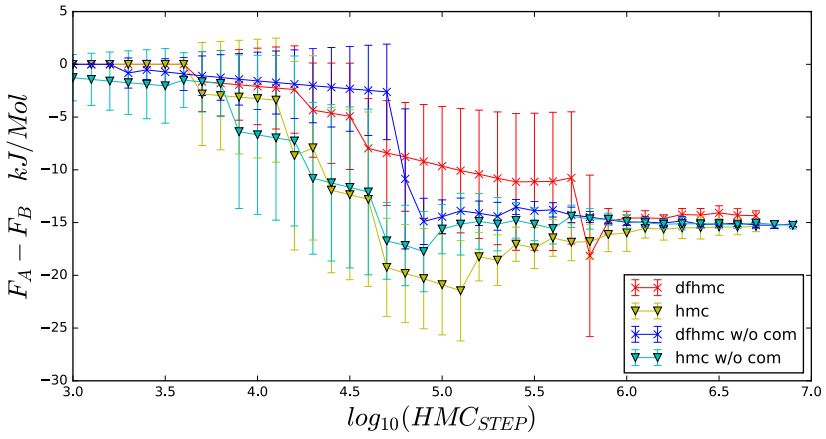


Figure 3.45 **Free energy difference between a high and low energy region in ϕ.** The graph shows the free energy difference where the regions are $-3 < \phi < -1$ and $0.5 < \phi < 1.5$. see Eq-3.33 for the calculation.

To summarise, In this section, we analysed the effect of centre of mass rotation on the DFHMC method. An alanine dipeptide system in implicit water is simulated where the centre of mass rotation is removed at every HMC step and the results are

compared to the simulations where the centre of mass rotation are not removed. It was seen that the removal of the centre of mass rotation during the simulation caused the HMC and DFHMC to converge to different ensemble average values. The difference is small, and along ψ no notable difference has been observed. Thus although small in this particular case, the removal of the centre of mass rotation disrupts the equilibrium distribution noticeably.

From observations on the convergence of the free energy difference, we conclude that the rotation of the molecule causes the DFHMC method to perform significantly worse. Thus to effectively apply the DFHMC on molecules, it is required that the molecule under simulation does not have a significant amount of rotation. As has been seen in the previous section where alanine dipeptide is analysed in an explicit water environment, the rotation of the molecule could be reduced by embedding the molecule in an environment such as water that restricts the motion of the molecule.

Following we will discuss some ideas on overcoming the centre of mass rotation problem within the DFHMC algorithm such that even if the system rotates the method would still sample efficiently. The goal is to give some ideas that we think could lead to an improvement of the DFHMC method and interesting research generally.

3.6 Rotation Problem - Possible Solutions

To tackle the rotation problem, we suggest two approaches to investigate. One is to eliminate the rotation effect by explicitly rotating the bias obtained by the digital filter. And the other is to consider an internal coordinates framework approach.

3.6.1 Rotating the bias vector: Non-Separable Hamiltonian Approach

Rotating the bias vector together with the molecule would make sure that the bias vector, obtained as a result of the filter application, always points to the same direction relative to the positions. Thus making sure that energy is added to the selected frequency of motions. However by rotating the bias vector the bias vector changes which changes the underlying velocity distribution and makes the process a non-markovian process. Hence the acceptance criteria would not ensure equilibrium.

To ensure the process is Markovian, the update made to the bias value needs to be reversible. If the bias update can be formulated as a function of the velocity and/or positions, then the update could be integrated using a generalized Verlet algorithm, and thus could be used within the HMC framework. This approach would not remove the molecule's centre of mass rotation, instead it would rotate the bias vector to align with the molecule's rotation.

Essentially the desired function might look like the following

$$\eta(b, q(t)) = rotate(b, Angle(q(t), q')) \tag{3.34}$$

where η is a new variable that depends on the bias, b and the rotation angle between q' and $q(t)$. q' defines a reference position obtained during the filter application. Thus q' and b are in phase and are updated every filter application. When b gets out of phase with the current position, then it will be rotated by the angle between the current position and the reference position, making η in phase with the current position. The bimodal distribution used in the DFHMC also needs to be updated by replacing the bias value b with η. With this new variable, the kinetic energy depends both on velocities and positions, thereby making the Hamiltonian non-separable.

$$H(q, v) = E(q, v) + U(q) \tag{3.35}$$

Thus as used in the Riemannian Manifold Hamiltonian Monte Carlo approach, a generalized Verlet algorithm may be used to integrate the Hamiltonian dynamics within each HMC step. We believe further investigation in this direction could provide improvements to DFHMC.

3.6.2 Internal Coordinates Molecular Dynamics

The rotation of the molecule in the HMC based methods arise because at each HMC step a new set of velocities are generated in cartesian coordinates. For the generated velocities to not induce a rotation the angular momentum at that time would need to be zero. The angular momentum L is conserved in an MD simulation and can be calculated as follows,

$$L = \boldsymbol{q}(t) \times \boldsymbol{p}(t) \tag{3.36}$$

where $\boldsymbol{q}$ and $\boldsymbol{p}$ are the position and momentum vector at time t. Thus for the angular momentum, L to be zero the cross product between the position and momentum needs to be zero, which can only happen when the momentum/velocity and position vector are parallel to each other. However, when we operate in cartesian coordinates and the velocities are drawn from a distribution it is highly unlikely that the positions and velocities will be parallel. If the velocities would be drawn into internal coordinates or if the molecular dynamics (ICMD) operates in the internal coordinate system then the induced rotation would not arise at each HMC step.

The common molecular dynamics packages all operate in cartesian coordinates. However there is a series of research that explores the idea to do molecular dynamics in the internal coordinates. A Lagrangian formulation of a molecular system in generalized coordinates that includes the internal coordinates was developed by Mazur et al. [58]. Different numerical methods that can solve the equations obtained from the Lagrangian formulation was studied and compared by Vladimir and Mazur [59]. Later Mazur developed a new analytical formulation based on a quasi hamiltonian formulation, that makes the numarical calculations easier [60]. Lee et al. developed a new ICMD method that replaces the mass matrix which is dependent on the internal coordinates, with the spectroscopic B-matrix to increase the computational efficiency [61]. A more efficient way of calculating the spectroscopic B-matrix is developed by Lee et al. [62]. A comparison of ICMD methods with conventional md is done by Lee et al. [63]. A method called GNEIMO (Generalized Newton–Euler Inverse Mass Operator) was developed to solve the ICMD equations efficiently [64], [65]. An application of GNEIMO was used for molecular structure refinement [66]. Further investigations into the ICMD methods might prove useful in finding ideas to improve the DFHMC method or even potentially apply the DFHMC method within the internal molecular dynamics framework. In addition, a method that provides a mechanism to draw velocities into internal coordinates is developed by Jain et al. [67], which is also used by GNEIMO. Thus similar to the bimodal distribution used in DFHMC, if a biased velocity distribution that can draw velocities in the internal coordinates can be developed, and then potentially be transformed into cartesian coordinates, then the rotation induced by HMC steps could be eliminated.

3.7 Riemannian Manifold Hamiltonian Monte Carlo - RMHMC

In this section, the RMHMC is investigated. As the methodology uses a non-separable Hamiltonian understanding of the algorithm might prove useful in extending the DFHMC to a non-separable Hamiltonian where the bias vector is dependent on the atomic co-ordinates, to make the DFHMC more robust of systems with centre of mass rotation.

In addition, the RMHMC method itself might be a promising method to enhance sampling or in overcoming energy barriers. As to our knowledge, in the literature the RMHMC method has not been applied to systems with energy barriers as of writing this thesis.

The section starts with the method applied to a Bayesian logistic regression problem and shows how the Bayesian inference problem relates to sampling from a canonical distribution. The RMHMC method is applied to the double well potential system which we used for DFHMC and HMC simulation in the previous sections.

3.7.1 Bayesian Logistic Regression

Given an input X and an output t the general idea is to find a function, f that best maps the inputs, X to the output t, i.e. $f(X) \approx t$. In logistic regression, the output variables are discrete and the function is a parameterised linear function. In this instance we consider the output to be binary values, i.e. $t \in 0, 1$ will be considered.

The Bayesian logistic regression is a logistic regression where the parameters of the function are estimated via Bayesian inference based on Bayes Rule.

$$P(w|X, t) = \frac{P(X, t|w)P(w)}{P(X, t)} \tag{3.37}$$

where w are the parameters of the function to be estimated, X and t represent the input and output data respectively.

$P(w|X, t)$ is called the posterior distribution, $P(X, t|w)$ is called the likelihood, $P(w)$ is called the prior. $P(X, t)$ does not depend on w, thus it is a normalisation constant, just like the partition function in the canonical ensemble. The product of the Prior and the likelihood is also called the joint likelihood. The goal of Bayesian inference is to sample the parameters w of the function/model from the posterior distribution. Similarly in molecular dynamics using HMC methods the goal is to sample the position vector of the molecule from the distribution of the positions which follows the canonical distribution. Thus the parameter w in Bayesian inference can be thought of as the positions in molecular systems. Further down the section, we will see that the negative log joint likelihood is related to the potential energy used in molecular sampling problem.

The likelihood gives the probability of the data X given the model parameter w. If each observation i is assumed to be independent and identically distributed the

likelihood can be written as the product of the likelihood of each data point i

$$P(X, t|w) = \prod_{i=1}^{N} P(x_i, t_i|w) \tag{3.38}$$

Since t is a binary variable, it can take values either 1 or 0. Thus if the probability of $t_i = 1$ at i^{th} data point given x_i is:

$$P(t_i = 1|x_i) = P(x_i) \tag{3.39}$$

then the probability of $t_i = 0$ at i^{th} data point given x_i is

$$P(t_i = 0|x_i) = 1 - P(x_i) \tag{3.40}$$

Therefore the likelihood at a given data point i

$$P(x_i, t_i|w) = P(x_i)^{t_i}(1 - P(x_i))^{1-t_i} \tag{3.41}$$

and the likelihood over all data points is:

$$P(X, t|w) = \prod_{i=1}^{N} P(x_i)^{t_i}(1 - P(x_i))^{1-t_i} \tag{3.42}$$

In logistic regression $P(x_i)$ is the logistic function, which maps the output of linear regression, i.e., $w^T x_i$ to an output that takes a value between 0 and 1.

$$\begin{aligned} P(x_i) &= \frac{1}{1 + e^{-w^T x_i}} \\ &= \frac{e^{w^T x_i}}{1 + e^{w^T x_i}} \end{aligned} \tag{3.43}$$

$$\begin{aligned} 1 - P(x_i) &= \frac{e^{-w^T x_i^T}}{1 + e^{-w^T x_i}} \\ &= \frac{1}{1 + e^{w^T x_i}} \end{aligned} \tag{3.44}$$

and the log of the ratio between $P(x_i)$ and $1 - P(x_i)$

$$\ln\left(\frac{P(x_i)}{1 - P(x_i)}\right) = w^T x_i \tag{3.45}$$

the log likelihood can be calculated

$$
\begin{aligned}
\ln(P(X, t|w)) &= \ln\left[\prod_{i=1}^{N} P(x_i)^{t_i}(1 - P(x_i))^{1-t_i}\right] \\
&= \sum_{i=1}^{N} t_i \ln(P(x_i)) + \sum_{i=1}^{N} (1 - t_i) \ln(1 - P(x_i)) \\
&= \sum_{i=1}^{N} t_i \ln(P(x_i)) + \ln(1 - P(x_i)) - t_i \ln(1 - P(x_i)) \\
&= \sum_{i=1}^{N} t_i \left[\ln(P(x_i)) - \ln(1 - P(x_i))\right] + \sum_{i=1}^{N} \ln(1 - P(x_i)) \\
&= \sum_{i=1}^{N} t_i \ln\left[\frac{P(x_i)}{1 - P(x_i)}\right] + \sum_{i=1}^{N} \ln(1 - P(x_i))
\end{aligned}
\tag{3.46}
$$

Substituting Eq-3.43, 3.44 into Eq-3.46 we obtain:

$$
\begin{aligned}
\ln(P(X, t|w)) &= \sum_{i=1}^{N} w^T x_i^T t_i + \sum_{i=1}^{N} \ln\left(\frac{1}{1 + e^{w^T x_i^T}}\right) \\
&= w^T X^T t - \sum_{i=1}^{N} \ln\left(1 + e^{w^T X_i^T}\right)
\end{aligned}
\tag{3.47}
$$

The prior in this particular example is given as a normal distribution.

$$
P(w) = \prod_{i=1}^{D} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(w_i - \mu)^2}{2\sigma}} \tag{3.48}
$$

where D is the number of parameters.

The log of the prior can be calculated as follows

$$
\begin{aligned}
\ln(P(w)) &= \sum_{i=1}^{D} \ln\left[\frac{e^{-\frac{(w_i - \mu)^2}{2\sigma}}}{\sqrt{2\pi\sigma}}\right] \\
&= \sum_{i=1}^{D} -\frac{1}{2}\ln(2\pi\sigma) - \frac{(w_i - \mu)^2}{2\sigma}
\end{aligned}
\tag{3.49}
$$

The logPrior

$$
\ln P(w) = -\frac{D}{2}\ln(2\pi\sigma) - \sum_{i=1}^{D} \frac{(w_i - \mu)^2}{2\sigma} \tag{3.50}
$$

when ignoring the constants, i.e., terms that are not a function of w, the log joint likelihood which is the sum of the log Prior and the log likelihood can be written as,

$$\begin{aligned}\ln(P(X,t|w)P(w)) &= \ln(P(X|w)) + \ln(P(w)) \\ &= w^T X^T t - \sum_{i=1}^{N} \ln\left(1 + e^{w^T x_i^T}\right) - \sum_{i=1}^{D} \frac{(w_i - \mu)^2}{2\sigma}\end{aligned} \tag{3.51}$$

The negative of the log joint likelihood can be thought of the potential energy divided by kT $U(w)/kT$ and w as the positions of the system. According to Bayes theorem the probability of w given the data X can be expressed as

$$P(w|X,t) \sim e^{-U(w)/kT} \tag{3.52}$$

where $U(w)/kT = -\ln(P(X,t|w)P(w))$ is the negative joint log likelihood.

To sample w from the distribution using the RMHMC method apart from the Force, i.e. $-dU/dw$ the second and third derivative of the potential energy is required.

In the following the first, second and third derivative of the potential energy for the Bayesian logistic regression is derived.

if $\mu = 0$ then the potential energy becomes

$$-U(w)/kT = w^T X^T t - \sum_{i=1}^{N} \ln\left(1 + e^{w^T x_i^T}\right) - \frac{w^T w}{2\sigma} \tag{3.53}$$

Then the derivative of the potential energy with respect to parameters w is:

$$-\frac{1}{kT}\frac{dU}{dw} = X^T t - \sum_{i=1}^{N} \frac{x_i e^{w^T x_i^T}}{1 + e^{w^T x_i^T}} - \frac{\sigma}{w} \tag{3.54}$$

the second derivative of the potential energy with respect to parameter w is:

$$\begin{aligned}-\frac{1}{kT}\frac{d^2U}{dw^2} &= -\sum_{i=1}^{N} \frac{d}{dw}\left(\frac{x_i^T}{1 + e^{-w^T x_i^T}}\right) - \frac{1}{\sigma} \\ &= -\sum_{i=1}^{N} -x_i^T \left(1 + e^{-w^T x_i^T}\right)^{-2} (-x_i) e^{-w^T x_i^T} - \frac{1}{\sigma} \\ &= -\sum_{i=1}^{N} x_i^T \frac{1}{1 + e^{-w^T x_i^T}} \frac{e^{-w^T x_i^T}}{1 + e^{-w^T x_i^T}} x_i - \frac{1}{\sigma}\end{aligned} \tag{3.55}$$

Eq-3.55 is used as the mass matrix, $G = \frac{d^2U}{dw^2}$ as explained in the RMHMC theory section.

Eq-3.42, 3.43 can be re-substituted into Eq-3.55 to make it more compact and easier to calculate the third derivative of the potential energy.

$$\frac{1}{kT}\frac{d^2U}{dw^2} = \frac{1}{kT}G(w) = \sum_{i=1}^{N} x_i^T P(x_i)^T (1 - P(x_i)) x_i + \frac{1}{\sigma} \tag{3.56}$$

$$\begin{aligned}
\frac{dG}{dw} &= \frac{d}{dw}\sum_{i=1}^{N} x_i^T p(x_i)^T (1 - P(x_i)) x_i \\
&= \sum_{i=1}^{N} x_i^T \left[\frac{dP(x_i)^T}{dw}(1 - P(x_i)) + P(x_i)^T \frac{d}{dw}(1 - P(x_i)) \right] x_i \\
&= \sum_{i=1}^{N} x_i^T \left[\frac{dP(x_i)^T}{dw}(1 - P(x_i)) - P(x_i)^T \frac{d}{dw} P(x_i) \right] x_i \\
&= \sum_{i=1}^{N} x_i^T \left[\frac{dP(x_i)^T}{dw} - P(x_i)^T \frac{dP(x_i)^T}{dw} - P(x_i)^T \frac{d}{dw} P(x_i) \right] x_i \\
&= \sum_{i=1}^{N} x_i^T \left[\frac{dP(x_i)^T}{dw} - 2P(x_i)^T \frac{dP(x_i)}{dw} \right] x_i
\end{aligned} \tag{3.57}$$

The derivative of $P(x_i)$ with respect to w is as follows,

$$\begin{aligned}
\frac{dP(x_i)}{dw} &= -(1 + e^{-w^T x_i^T})^{-2}(-x_i)e^{-w^T x_i^T} \\
&= x_i \frac{e^{-w^T x_i^T}}{(1 + e^{-w^T x_i^T})^2}
\end{aligned} \tag{3.58}$$

substituting Eq-3.58 into Eq-3.57

$$\begin{aligned}
\frac{dG}{dw} &= \sum_{i=1}^{N} x_i^T \left[\frac{e^{-w^T x_i^T}}{1 + e^{-w^T x_i^T}} \frac{1}{1 + e^{-w^T x_i^T}} x_i - 2\frac{1}{1 + e^{-w^T x_i^T}} \frac{e^{-w^T x_i^T}}{1 + e^{-w^T x_i^T}} \frac{1}{1 + e^{-w^T x_i^T}} x_i \right] x_i \\
&= \sum_{i=1}^{N} x_i^T \left[P(x_i)(1 - P(x_)) x_i - 2P(x_i)P(x_i)(1 - P(x_i)) x_i \right] x_i \\
&= \sum_{i=1}^{N} x_i^T \left[P(x_i)(1 - P(x_i))(1 - P(x_i)) x_i \right] x_i
\end{aligned} \tag{3.59}$$

3.7.2 Double Well Potential

$$U(x, y) = 0.5K_x x^2 + K_y y^2 (y - b)^2 \tag{3.60}$$

To apply the RMHMC to this system the force, F and G mass matrix and the derivative of the G matrix is required.

$$\begin{aligned}
\{F_x, F_y\} &= \left\{ -\frac{dU}{dx}, -\frac{dU}{dy} \right\} \\
&= \left\{ -K_x x, -2K_y y(y - b)^2 - 2K_y y^2 (y - b) \right\}
\end{aligned} \tag{3.61}$$

In a system with two degrees of freedom, the second derivate of the potential energy is a 2×2 matrix

$$G = \begin{pmatrix} \frac{d^2U}{dx^2} & \frac{d^2U}{dxdy} \\ \frac{d^2U}{dydx} & \frac{d^2U}{dy^2} \end{pmatrix} = \begin{pmatrix} -\frac{dF_x}{dx} & -\frac{dF_y}{dx} \\ -\frac{dF_x}{dy} & -\frac{dF_y}{dy} \end{pmatrix} \tag{3.62}$$

From Eq=3.61 we can see $\frac{dF_x}{dy} = \frac{dF_y}{dx} = 0$ as the force along x is not dependent on y and the force along y is not dependent on x.

The derivative of F_x with respect to x is a constant, i.e. it does not change with the positions or the velocities of the system.

$$-\frac{dF_x}{dx} = K_x \tag{3.63}$$

where as the derivative of F_y with respect to y canges with the position along y coordinate.

$$\begin{aligned} -\frac{dF_y}{dy} &= 2K_y(y-b)^2 + 4K_y y(y-b) + 4K_y y(y-b) + 2K_y y^2 \\ &= 2K_y(y-b)^2 + 8K_y y(y-b) + 2K_y y^2 \\ &= 2K_y[y^2 + 4y(y-b) + (y-b)^2] \\ &= 2K_y[(2y-b)^2 + 2y(y-b)] \end{aligned} \tag{3.64}$$

Thus the mass matrix, G used in RMHMC for the double well potential is a diagonal matrix, where the G_{22} element changes depending on the y position. Effectively the mass of the system is adjusted based on the potential energy's curvature.

To make the G matrix positive definite a transformation called softAbs is suggested [68].

When G is a diagonal matrix, The softAbs transformation applied to G is defined as follows:

$$\hat{G} = softAbs(G) = G\coth(\alpha G) \tag{3.65}$$

To understand the properties of this transformation, the exponential form can be analysed

$$\hat{G} = G\frac{e^{\alpha G} + e^{-\alpha G}}{e^{\alpha G} - e^{-\alpha G}} \tag{3.66}$$

from the exponential form we can easily see that $softAbs(G) = softAbs(-G)$. Thus any negative values in the G matrix with this transformation becomes positive. It effectively takes the absolute value. The transformation is also differentiable which is a requirement to calculate the derivative of G

When $\alpha \to \infty$ then $\hat{G} \to G$, Thus α is choosen as big as possible to reduce the distortion of G by the transformation. $\alpha = 10^6$ is used.

In Fig-3.46-c, The effect of the softAbs transformation can be seen for the double well potential. The transformation makes G positive where $G < 0$ and is unchanged for $G > 0$. However this transformation results in having a discontinuity in the derivative of the transformed G, Figure-3.46-d

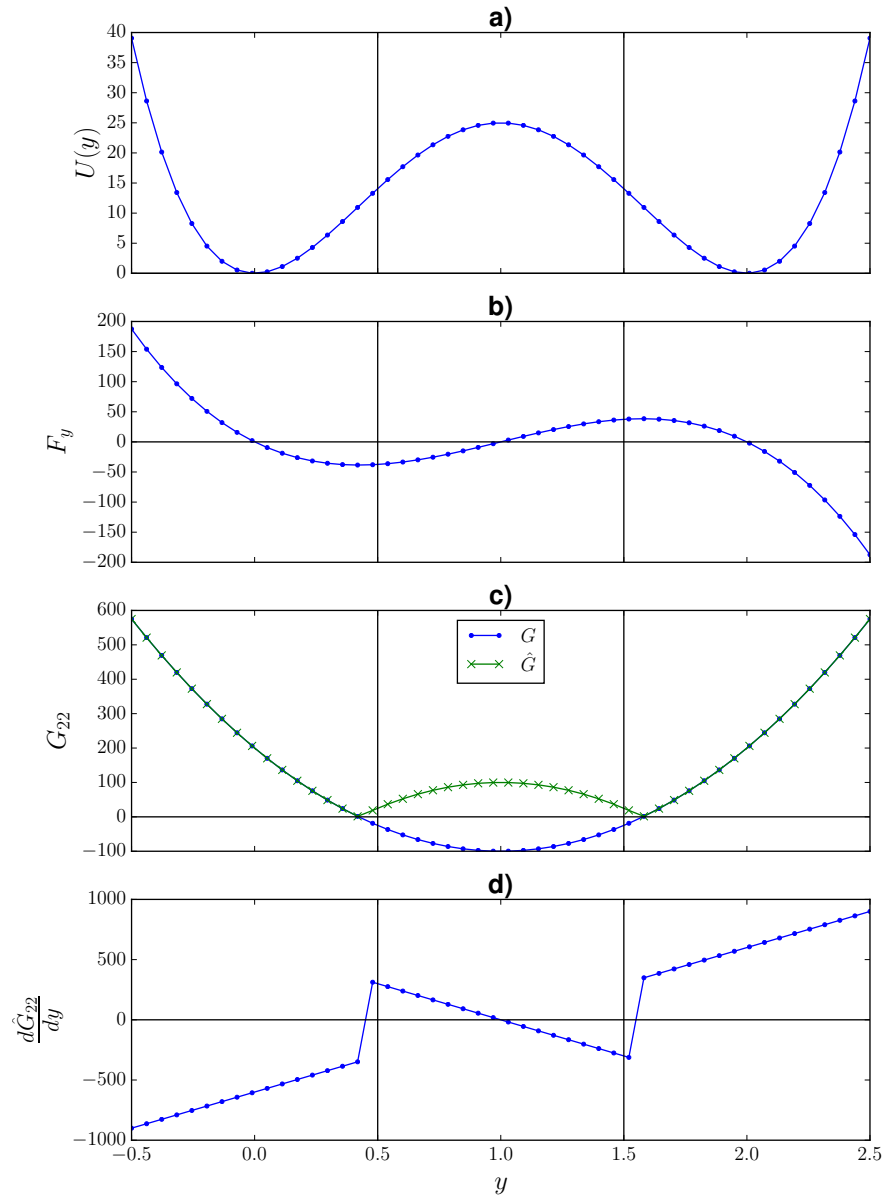
The discontinuity in the derivative of the transformed G makes the generalised Verlet integrator not conserving the Hamiltonian. Thus the softAbs transformation is not really suitable for systems with energy barriers.

Fig-3.47 shows an NVE simulation with 50 MDSteps of the doublewell potential system with a generalized Verlet integrator where the G matrix is transformed using the softAbs transformation. The effect of the discontinuity in the derivative of the G matrix is reflected in the momentum, position trajectory and in the total energy of the system. It can be seen that at 15 MDSteps corresponding to $y = 1.5$ the disruption in the total energy is observed. Note that from Fig-3.46 $y = 1.5$ is also the place where one of the discontinuity in the derivative of G occurs. Thus to apply RMHMC to molecular systems where the energy surfaces have many energy barriers, mass metrics that have continues derivatives and are positive definite in such surfaces would need to be constructed and are open to further investigation.

In this specific case of the double well potential, a positive definite and with continues derivative can be constructed by adding a constant to the mass matrix such that G_{22} in Fig-3.46 is shifted upwords above the zero line. Running a generalised Verlet integrator using the shifted mass matrix, results in a trajectory that conserves the total energy Fig-3.48.

To summarise, It is highlighted that the Bayesian inference problem from a mathematical point is equivalent to the sampling of molecular configurations. The model parameters and the joint log likelihood is related to the positions and the potential energy of the molecular system respectively. Therefore, looking at both problems as close variants, extending the literature review to encompass the Bayesian inference and the configurational sampling field would enrich both fields and may spark innovative methods to develop.

The RMHMC method without any further modifications does not seem to be suitable for systems with energy barriers. Even though system dependent modifications could be done, more research would be required to find a generic metric for the mass matrix such that the method is suitable for molecular systems. However, the underlying framework is useful in terms of laying the foundations for the further development of the DFHMC method explained in the previous section.



Double Well System.. a) Potential energy along y **b)** Force along y **c)** G_{22} in y (blue dot) and softAbs transformed $\hat{G_{22}}$ in (green cross) **d)** The derivative of $\hat{G}_{22}$

Figure 3.46

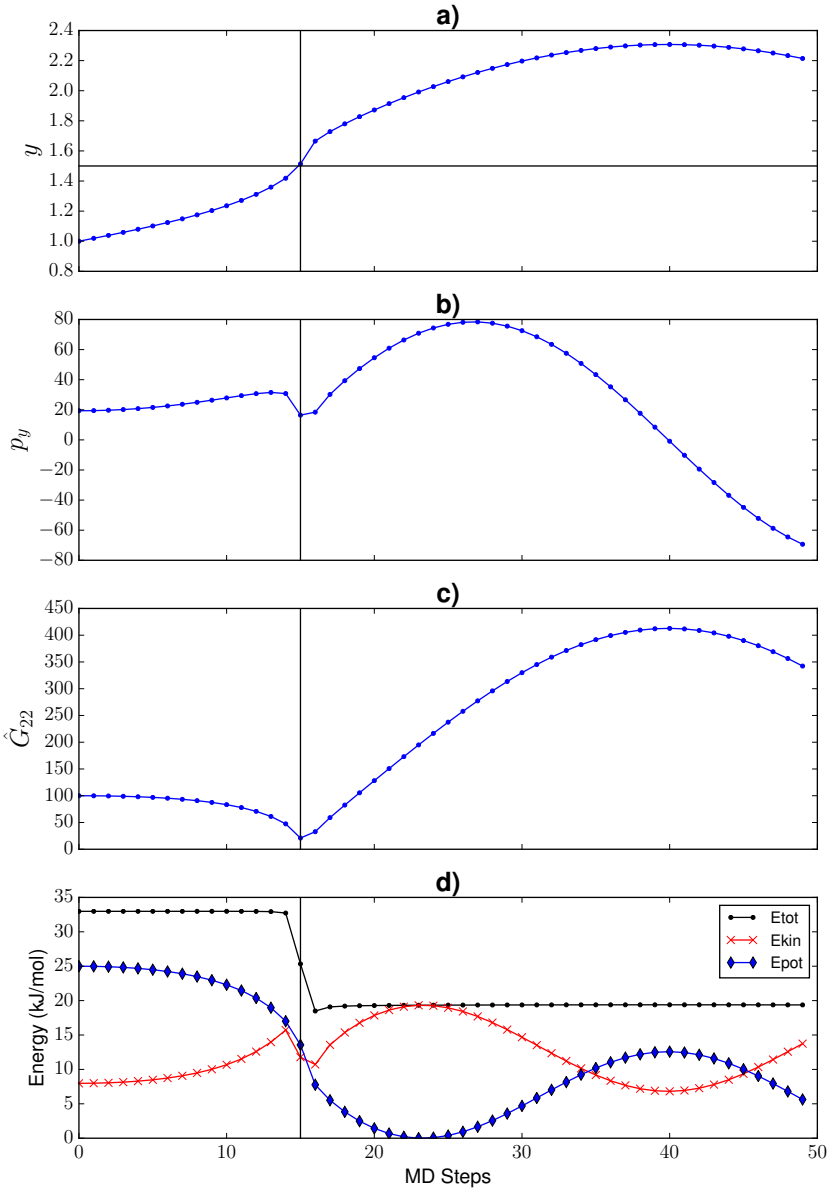
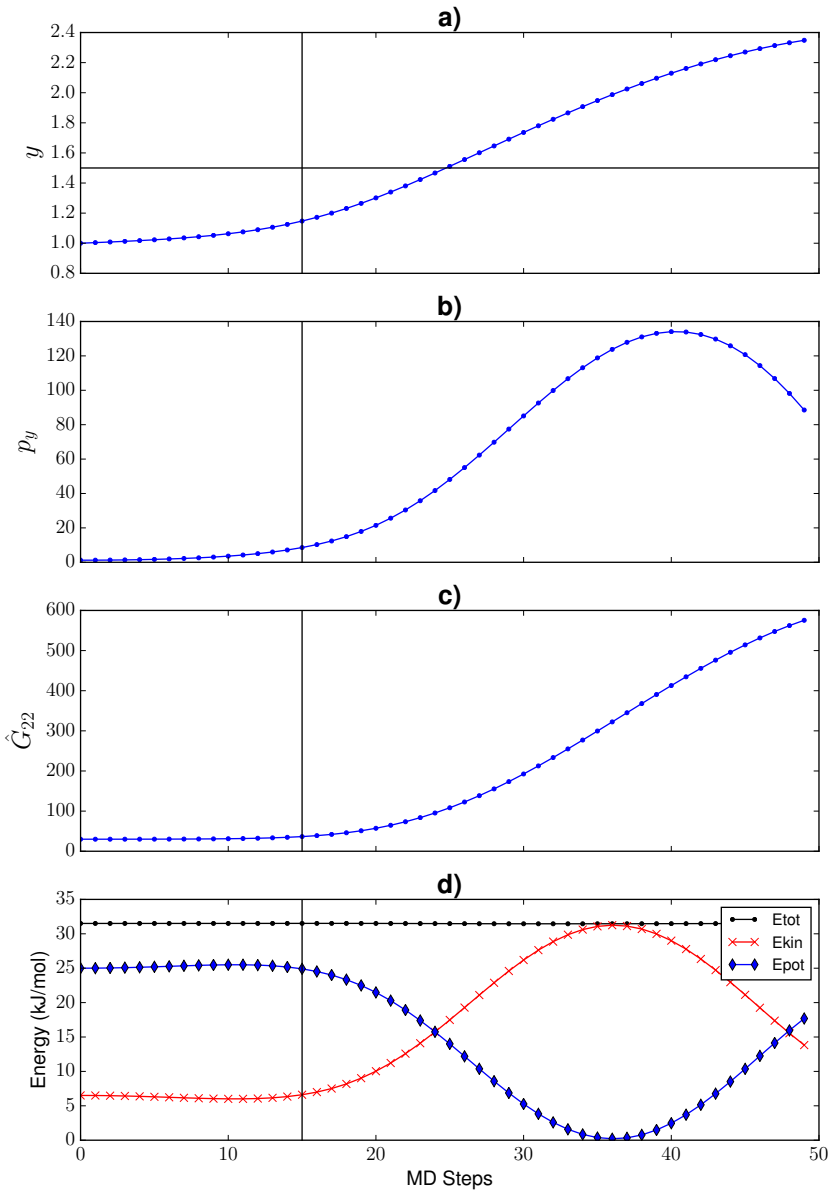


Figure 3.47 **Double Well System.. a)** Potential energy along y **b)** Force along y **c)** G_{22} in y (blue dot) and softAbs transformed $\hat{G_{22}}$ in (green cross) **d)** The derivative of $\hat{G}_{22}$



Double Well System. **a)** Potential energy along y **b)** Force along y **c)** G_{22} along y (blue dot) and softAbs transformed $\hat{G_{22}}$ in green cross **d)** The derivative of $\hat{G}_{22}$

Figure 3.48

3.8 Abl Kinase

Preliminary simulations of DFHMC is conducted on Abl Kinase molecule. The configuration of the DFG motif in Abl kinase is critical in determining the binding of a drug. The DFG has two states DFG-in and DFG-out and the conformation change between states act as a switch to activate and inactivate the binding of a drug. Fig-3.49 shows the Abl kinase molecule **b** and **c** is a magnified version of the molecule focusing around the DFG residues that shows the DFG-in and DFG-out configuration respectively. As can be seen in the DFG-in configuration, the sidechain of Phe residue (purple) is pointing inward, and the side chain of Asp residue (blue) is pointing outward of the graph. Whereas in the DFG-out configuration its reversed, i.e. the sidechain of Phe residue (purple) is pointing outward and the side chain of Asp residue (blue) is pointing inward.

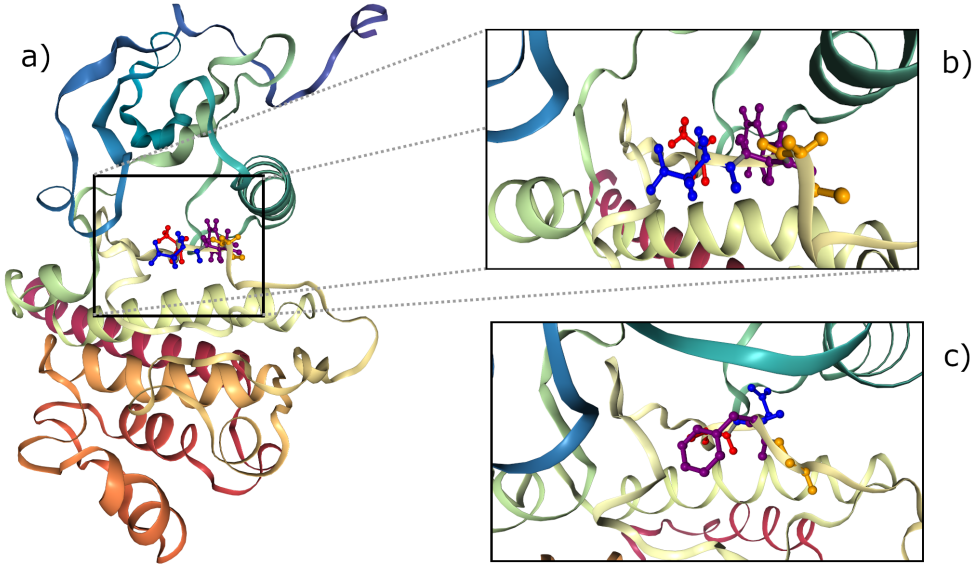


Figure 3.49 **Abl kinase DFG-in and DFG-out configurations.** Secondary structure representation of Abl kinase, the ADFG (Ala-Asp-Phe-Gly) residues are shown in ball and stick representation and are coloured as red, blue, purple and orange respectively. **a)** Shows the full Abl kinase protein in DFG-in configuration based on PDB entry 2F4J [69]. **b)** Focuses **a** around the DFG residues. **c)** Also focuses on the DFG residues but the configuration represent a DFG-out configuration based on PDB entry 1OPK [70]

The starting configuration is prepared based on 2Fj4 [69] PDB entry which has a DFG-in configuration. During the PDB preparation stage, any non-protein molecules are removed. The 396^{th} residue, Pro based on the enumeration given in 2FJ4 PDB entry is mutated back to its wild-type sequence, i.e. to a His residue. Hydrogen atoms are added to residues based on a $pH = 7$ environment. After that, the system is minimized in energy based on L-BFGS optimization algorithm [71].

The Abl system is modelled with a Generalized Born implicit water model. In this

	Production Simulations	
	HMC	DFHMC
MDStep	80	20
dt (fs)	2	0.5
temperature (K)	300	300
HMCStep	1.49×10^6	6×10^6
nstfilter	-	100
filter (cm^{-1})	-	100
scl	-	1
nstout	1	1

Simulation parameters for production, Abl kinase in implicit water. For DFHMC simulation, the digital filter is applied only on $C\alpha(A380)$, $C\beta(A380)$, $C\alpha(D381)$, $C\gamma(D381)$, $C\alpha(F382)$, $C\gamma(F382)$ atoms, enumerations of residues is based on 2FJ4 pdb entry

Table 3.7

setup, the physical presence of the water molecules are eliminated, but the effect of the water on the Abl kinase's equilibrium distribution is approximately conserved. Since no explicit water is present in this setup, the equilibration protocol used with systems in explicit water is not needed. Thus production simulation is run for HMC and DFHMC simulation directly. The simulation parameters used are defined in Table-3.7.

In a study by Meng et al. $C\beta(A380) - C\alpha(A380) - C\alpha(F382) - C\gamma(F382)$ and $C\beta(A380) - C\alpha(A380) - C\alpha(D381) - C\gamma(D381)$ dihedral angles were used as collective variables in an umbrella sampling simulation to observe the DFG-in and DFG-out conformation changes [72]. Thus in this experiment, the digital filter is only applied to a selected set of atoms ($C\alpha(A380)$, $C\beta(A380)$, $C\alpha(D381)$, $C\gamma(D381)$, $C\alpha(F382)$, $C\gamma(F382)$) which make up these dihedral angles. All the other atoms behave like a in normal HMC simulation.

Fig-3.50 shows the sampling of $C\beta(A380) - C\alpha(A380) - C\alpha(F382) - C\gamma(F382)$ and $C\beta(A380) - C\alpha(A380) - C\alpha(D381) - C\gamma(D381)$ dihedral angles for an HMC and DFHMC simulation of $1.49x10^6$ HMC steps. The DFG-in configuration is around $(1, -2)$ and the DFG-out configuration is around $(-2, 0)$. As can be seen, neither DFHMC nor HMC were able to sample the DFG-out configuration within the given simulation time. Both simulations only sampled the vicinity of the DFG-in configuration. It can also be observed that the DFHMC sampling is more directed along a specific direction, whereas the HMC's sampling is closer to a uniform sampling along every direction. The direction the DFHMC is sampling is the direction where the energy change in the energy landscape along those dihedral angles is the lowest, see [72] for the fully sampled energy landscape. And if the sampling would continue along that direction, the simulation would eventually sample the DFG-out configuration.

Fig-3.51 shows two DFHMC simulation at two different HMC steps **a** shows the

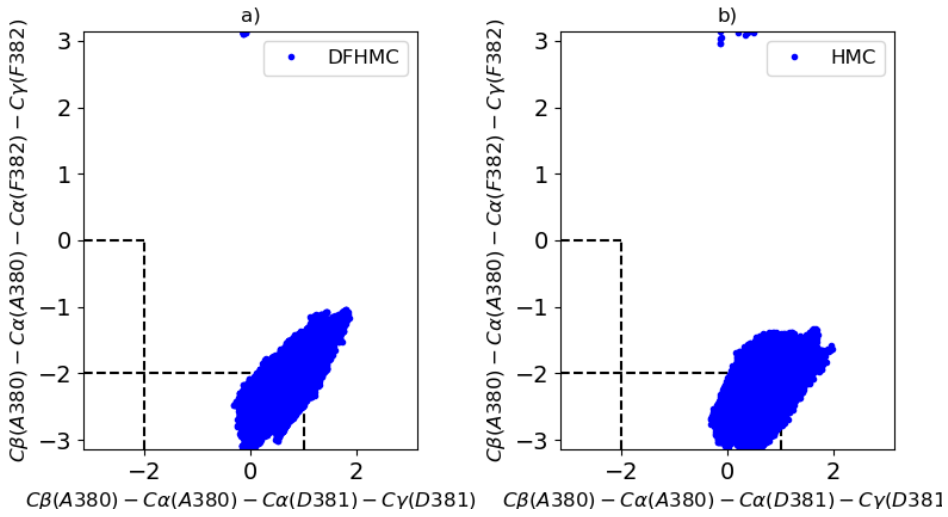
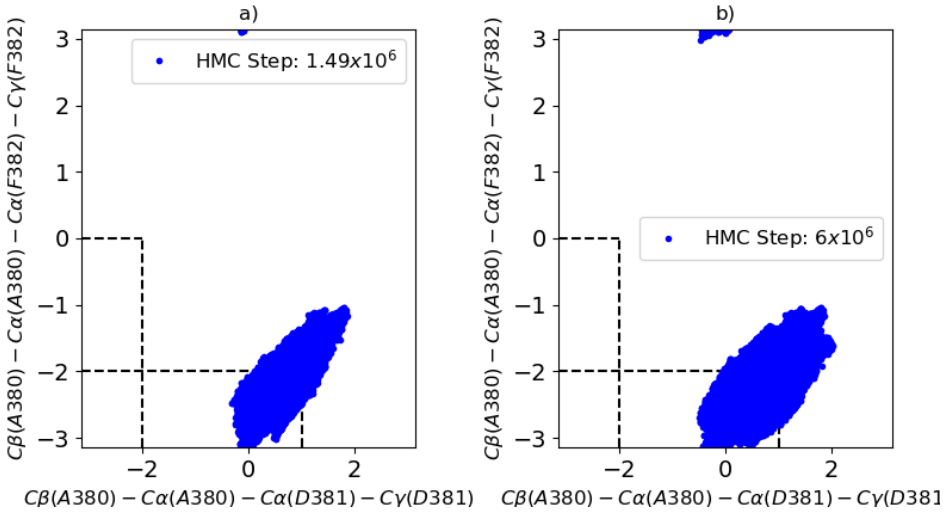


Figure 3.50 **Abl kinase in implicit water.** Shows the scatter plot of $C\beta(A380) - C\alpha(A380) - C\alpha(F382) - C\gamma(F382)$, $C\beta(A380) - C\alpha(A380) - C\alpha(D381) - C\gamma(D381)$ dihedral angles of DFHMC and HMC simulations in implicit water.HMC steps = $1.49x10^6$

DFHMC simulation up to $1.49x10^6$ HMC steps which is the same as Fig-3.50-**a** and **b** shows the DFHMC simulation at $6x10^6$ HMC step. It can be seen that the increased HMC step does not increase the sampling significantly along the direction that DFHMC enhances, but more sampling is observed in the orthogonal direction. Thus rather than increasing the HMC steps of the simulations, the scaling of the designed filter could be increased by adjusting the *scl* parameter to add more energy along the filtered direction.

To summarise, in this experiment, the DFG-in and DFG-out configuration change was not observed via DFHMC or HMC simulations. However, the sampling of the DFHMC simulation in the vicinity of the DFG-in configuration is directed along a line compared to the HMC simulation indicating that the DFHMC method is biasing the simulation along the low frequency direction. Further optimising the parameters of the DFHMC method might lead to an observation of the DFG-in, DFG-out configuration change.



Abl kinase in implicit water. Shows the scatter plot of $C\beta(A380) - C\alpha(A380) - C\alpha(F382) - C\gamma(F382)$, $C\beta(A380) - C\alpha(A380) - C\alpha(D381) - C\gamma(D381)$ dihedral angles of DFHMC simulation in implicit water. **Left**: Shows the results upto $1.49x10^6$ HMC steps. **Right**: Shows the results upto $6x10^6$ HMC steps.

Figure 3.51

Code Integration

Chapter 4

In order to apply the Hybrid Monte Carlo based methods to molecular systems, in an easily usable and computationally efficient way, we decided to develop our algorithms around a core molecular dynamics simulation package. By using a well developed, optimised and tested molecular dynamics package as our MD integrator, we will be avoiding reinventing the wheel and can focus on the essential parts of the HMC algorithms. More generally our ultimate goal is to write the algorithms in such a way that they can be used with various highly used molecular dynamics packages. Thus, we hope that the HMC based algorithms especially the DFHMC method can be used by researchers with a learning curve independent from their choice of molecular simulation package.

Before the DFHMC method is integrated into an optimised molecular dynamics package, an HMC based code with its own molecular dynamics integrator is build for two dimensional systems to allow for testing the methodology. The code can be found in the below linked repository

Code repository: https://bitbucket.org/CSPer/newtondynamics/

As can be seen from the flowchart of the DFHMC algorithm Fig-4.1, the molecular dynamics module is required within the digital filter and in the HMC module.

To achieve the aforementioned objectives, we would need to develop the HMC based algorithms (Includes DFHMC) as a module independent from the Molecular Dynamics integrator module, and find a way to pass information between those two modules as depicted in Figure 4.2.

The information required by the HMC module is:

Variables

- Velocities
- Total Energy

The HMC module needs to be able to restart the MD simulation with given velocities and positions. Thus, the required information to be passed to the MD module is

Variable

- Velocities
- Positions

Functions

- Restart MD or re-initialize MD simulation
- Remove centre of mass translation and rotation from velocities

4.1 MD-Python-Fortran integration

Code repository: https://bitbucket.org/CSPer/md-python-fortran/

This integration scheme was developed previously in our lab to enable the integration of user developed algorithms with several MD packages. Currently, only the wrapper for NAMD MD package [73] exist, more wrappers for other MD packages can be added. In this scheme, the information flow between the MD package and the user developed algorithms (HMC based methods) is controlled via a wrapper written in python. However, the python wrapper can only execute the modules and can not get the output value directly and pass it to the other module. The information flow between the modules is established by reading and writing to a file on the hard drive. Thus, the python wrapper only serves as a platform where functionalities from other two modules can be imported into the python interface. The desired algorithm can be designed by using the functions of the HMC and MD modules inside the python interface. A schematic of the information flow of this system can be seen in Figure 4.3. The reading and writing to the hard drive makes this integration very slow. Thus testing the various HMC algorithms where many simulations for relatively long steps are performed makes this type of integration unsuitable. Thus, an integration design that bypasses the hard drive will be efficient and preferable. The integration in the next section resolves the bottleneck of communication over the hard drive and gives a much faster implementation.

4.2 MD-Plumed integration

Code repository: https://bitbucket.org/CSPer/md-plumed/

In this scheme, the Plumed module [9] is able to pass and receive information directly from the MD Package without needing to interact with the hard drive. Inside the Plumed module, some submodules have functions specifically written for the MD package of interest, which are injected via a patch inside the MD package. These injected functions create a pipe between Plumed and the MD package which enables to share some information. Thus, since the Plumed module can communicate effectively

with the MD module, the aim is to add the HMC based module inside the Plumed module such that the HMC based module can communicate with Plumed and indirectly with the MD module as depicted in Figure 4.4.

Unfortunately, the information passed between Plumed and the MD module does not include the velocities. Thus to make use of Plumed to integrate the HMC modules the patch of Plumed needs to be modified such that velocities from MD are passed to Plumed. Since the Plumed patch is written separately for each MD package program, the plumed patch needs to be adapted to each MD package to enable HMC module functionality.
Since in the previous section (MD-Python-Fortran integration), NAMD was used as the MD package. At first, the plumed patch for NAMD was attempted to be modified. However, the velocities in the parallelized partly documented NAMD program are not stored in a variable as straightforward as the positions, thus making it difficult to modify the plumed patch to get the velocities from NAMD. Therefore the plumed patch for the Gromacs MD package [10] is modified instead where the access to the MD velocities are straightforward. Even though the implementation is computationally more efficient, the changes required makes the design and any new method development highly dependent on different MD engines. The integration is not modularised very well; thus a method developer that wants to extend or add a new method would require to understand nearly all the code base. Thus the integration is not very user friendly for method extensions. The next section introduces an integration that alleviates the extensibility problem.

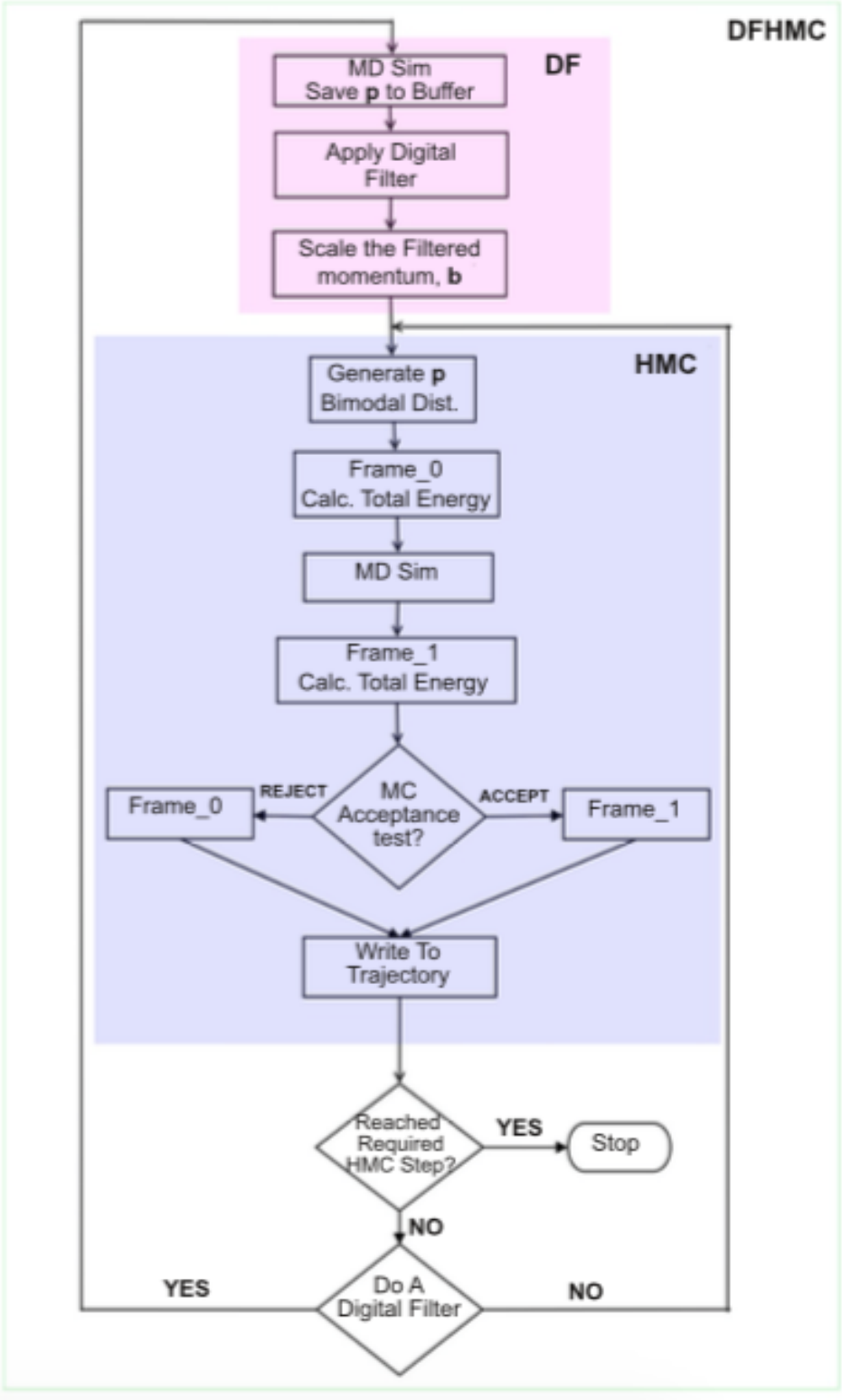
4.3 HMC-OpenMM

Code repository-1: https://bitbucket.org/CSPer/openmm-dfhmc/
Code repository-2: https://bitbucket.org/CSPer/openmm-source/

In this scheme, the OpenMM package [11] is used to integrate the DFHMC or HMC based module. OpenMM is a well modularised code and has different abstraction levels for different types of users. Thus the code is not just designed for the user that only run a simulation, but it is also user friendly for method developers. OpenMM has a python module called the custom integrator. It is based on a mathematical expression parser called Lepton. This feature of OpenMM allowed us to quickly develop relatively fast HMC based algorithms. The custom integrator feature allows the developer to write integrators by giving access to state variables such as the position, velocity, force, potential energy and mass of the system. The developer can also define and use custom variables that can interact with the state variables. These options give enough flexibility for the developer to design new integrators or sampling algorithms without requiring to dive into the OpenMM's more complex lower abstraction levels. The code

written via the custom integrator level is automatically optimized and parallelised to work within different platforms such as CPU and GPU. The developed HMC based methods that use the custom integrator are collected in code repository-1.

Although most of the HMC based integration were able to be done through the custom integrator framework, the requirement of removing the centre of mass translation and rotation could not be done via that framework. Thus the OpenMM source code is modified and a new function called "AddremoveCom()" is added such that it can be used within the custom integrator. The current implementation runs on a CPU and is not parallelised. However, the implementation is set up such that it can be used in the OpenMM's GPU platform setting, where all the native OpenMM code will be optimized to run on the GPU. For future work, the removal of the centre of mass rotation could be developed for each OpenMM's supported platform,i.e., CPU, CUDA, OpenCL. The methods developed via the custom integrator module can be developed in a lower abstraction level in C++ to increase computational efficiency.



DFHMC Flowchart. The graph shows the flowchart for the DFHMC method. The submodules of the DFHMC method are labeled as DF (digital filter) and HMC (Hybrid Monta Carlo)

Figure 4.1

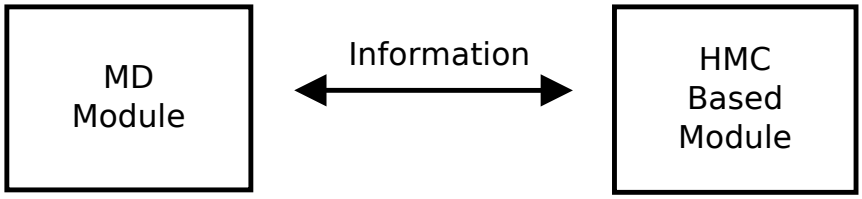


Figure 4.2 **Information Flow.** Information flow between MD and HMC based modules

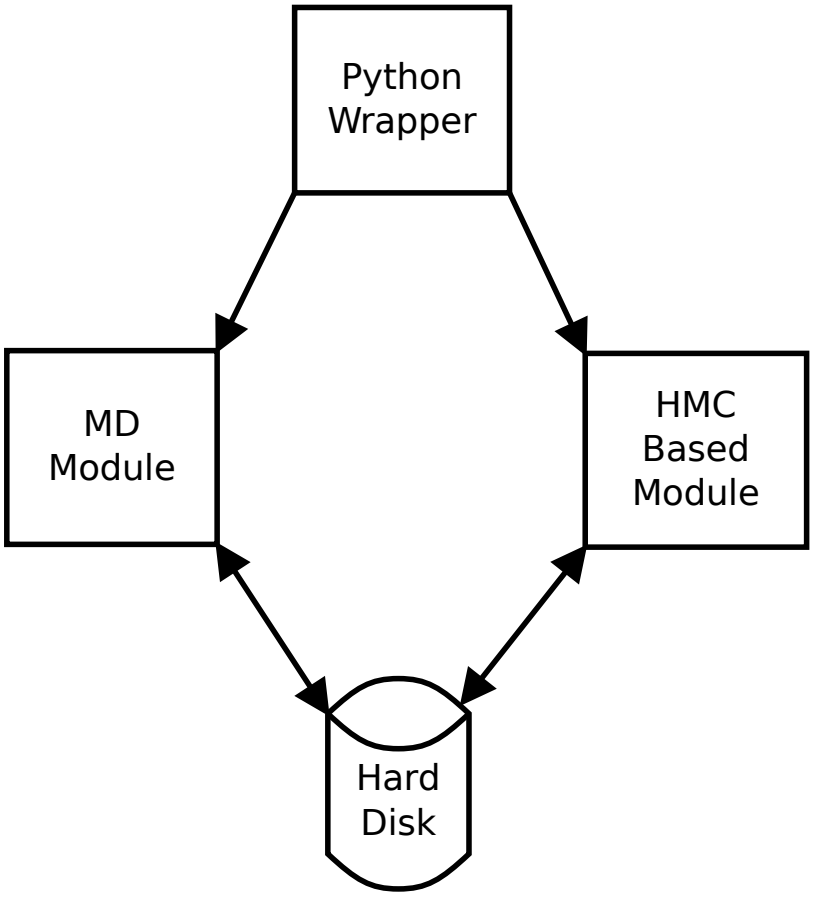


Figure 4.3 **Python-MD-HMC Module: Information Flow.** Information flow between MD and HMC based module via hard drive controlled through a Python wrapper.

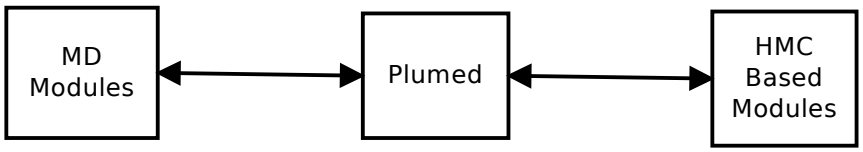


Figure 4.4 **Plumed Module: Information Flow.** Information flow between MD and HMC based module via the Plumed module.

Chapter 5

Conclusions

In this thesis, we proposed a multiphysics method called digitally filtered hybrid Monte Carlo (DFHMC) that combines Digital Filter (DF), Molecular Dynamics (MD) and Monte Carlo (MC) methods to alleviate the sampling problem in molecules with highly rugged energy landscapes such as proteins. Knowing the magnitudes of the vibrations present in the molecule, a DF can be used to filter out specific frequency ranges in the molecule selectively. When the digital filter is designed to filter out high frequencies a.k.a low-pass filter, an enhanced sampling can be achieved by biasing the velocities of the slow degrees of freedom as observed in the results of the double well and curved potential systems simulations. The usage of DF in conjunction with MD as in the DFMD [46], RDFMD [1] methods, disturbs the equilibrium of the MD simulation. To preserve the equilibrium property, the DF is used in conjunction with the HMC method. The conjunction of DF with HMC is done by introducing a bias parameter into the velocity distribution used in HMC. As used in the MEHMC method [2], a bimodal distribution together with the relevant acceptance criteria are used to introduce the bias parameter. In the DFHMC method, the bias parameter is updated using a predesigned digital filter. In chapter 2, it is shown that energy can be added to the low frequency motion via bimodal velocity distribution where the bias values are obtained with a digital filter application. By testing on a system with double well potential, water, a dihedral angle system and alanine dipeptide, we observed that the energy can be added not just along a cartesian coordinate but also along an angle and dihedral angle vibration. In alanine dipeptide, which has many more degrees of freedom compared to the toy systems studied in this thesis, we observed that the energy added to the lower frequencies by the bimodal distribution quickly dissipates (1-2 ps time scale) to higher vibrational modes.

In 2D double well potential system simulated at $k_BT = 1$ where the energy barrier height is 1 units, it was observed that the simulations with a bias value different than zero converge more slowly locally. However, according to the global convergence measure the simulations with a bias in the slow dimension, y seems to result in smaller percentage error for the average related to the y coordinate, $< y^2 >$ than the simulations without the bias in that coordinate, but seems to perform worse for the $< x^2 >$ average. For a conclusive decision on the performance effect of the bias values, more simulation experiments were constructed at lower k_BT values where the

HMC without any bias needs relatively long HMC steps to converge globally.

From the 2D double well potential system simulation results, which are simulated at low $k_BT = 0.1$ values compared to the energy barrier height, i.e. 1, we noted that the simulations that use the bimodal velocity distribution with the bias obtained via a digital filter converge much more quickly to the ensemble average along the slow direction (y)compared to the standard Hybrid Monte Carlo method. We observed that while the ensemble average error along the slow direction for the HMC method is still close to 100% for the DFHMC, the error goes down to 1%. On the other hand, the convergence on the ensemble average along the fast direction (x) is slower with the DFHMC method compared to the HMC method. Suggesting that the DFHMC method is better in exploring the surface globally; however, it is not as good in exploring the surface locally. The enhancement in the global convergence of the DFHMC method is especially prominent when the energy barrier is high enough such that the standard HMC method has difficulties to overcome it.

The double well potential in the 3D setup is simulated via the HMC and DFHMC methods. Consistent with our observations of the 2D double well system, the DFHMC method converges faster to the ensemble average compared to the HMC method. Both HMC and DFHMC methods conserve the equilibrium distribution. Thus the performance is consistent with the observed result for the 2D Double Well potential analysed via the non-optimised python code in the DoubleWell- 2D section, which indicates that the integration of the DFHMC method into the OpenMM molecular dynamics package works as expected. The DFHMC and HMC application can be scaled to more complex and bigger systems in a computationally efficient way via the OpenMM implementation. The observed enhancement of the DFHMC method on the double well potential shows us that the DFHMC is capable of adding energy along a slow degree of freedom that is aligned along a cartesian coordinate.

It has been shown that on a system where the energy barrier is along the cartesian coordinate and a linear combination of the cartesian coordinates the DFHMC method is able to enhance the sampling much better than the HMC method. Furthermore, the convergence performance of DFHMC and HMC did not change significantly by the rotation of the cartesian coordinates, given the energy surface has the same curvatures. It is observed that the direction of velocities generated by DFHMC aligns along the energy barrier with higher probability, whereas in HMC there is no significant preference of the direction of the velocities.

The DFHMC and HMC methods are simulated on a system with a curved transition path. The DFHMC method has shown enhanced global sampling compared to the HMC method. The convergence rate for DFHMC is much higher than the HMC method. Both methods generated the equilibrium distribution reasonably well. However, the DFHMC method showed a slight difference from the benchmark PMF around the transition state, indicating the local sampling of DFHMC suffers from the

performance increase in global sampling.

Summarising the results observed from the double well potential system and the system with curved transition path; it is observed qualitatively and quantitatively that the implementation of a multivariate and a single variate bimodal velocity distribution to the HMC algorithm with the modified acceptance criteria generates samples from the correct target distribution for various bias values. In all systems, we observed that the samples generated by the DFHMC method follow the canonical ensemble reasonably well. It has been shown that the DFHMC method enhances the sampling significantly compared to the HMC method on systems where the energy barrier is along the cartesian coordinates, a linear combination of the cartesian coordinates and on a nonlinear curved path. The enhanced sampling observed on the curved system shows that the DFHMC method is able to adapt to the directional change of the energy barrier and add energy along the curved transition path. By analysing the trajectories obtained from HMC and DFHMC, we observed that the rate of energy barriers crossing is much higher in DFHMC and the simulation time spend on local minima sampling is much lower compared to the HMC method. Thus for DFHMC, there is a tradeoff between local and global sampling, a higher rate of global sampling reduces the amount of local sampling.

The alanine dipeptide system is equilibrated in an explicit water environment. Independently repeated simulation with different starting conditions are run for HMC and DFHMC methods. Although the difference is small, we observe that the DFHMC is sampling better than the HMC method, especially along the ϕ dihedral angle. We also observed that the Langevin implementation of the NVT simulation performed very well.

The HMC and DFHMC simulations of the alanine dipeptide molecule in implicit water are able to generate the equilibrium NVT distribution. However, compared to the alanine dipeptide explicit system, the enhancement of DFHMC versus HMC observed in the implicit water system is significantly lower. We note that the overall convergence performance for HMC is better compared to the DFHMC method in terms of the speed of generating stable results as observed by the decreasing error bars. Thus we can conclude that the DFHMC method performs worse in the implicit water system compared to the explicit water system. To explain a possible reason for the observed decline, we hypothesised that the observed difference in enhancement between the implicit and explicit water system is due to the difference of the centre of mass rotation of the alanine dipeptide molecule. The molecule in explicit water is less prone to the centre of mass rotation due to the friction between the water molecules, and thus the added energy to the centre of mass rotation can be dissipated much faster and easier. At each HMC step, a new velocity is drawn, and some portion will contribute to the centre of mass rotation of the system. Since the whole system in the implicit water case is the alanine dipeptide molecule, rotational motion of the centre of

mass is conserved within each HMC step due to the conservation of angular momentum. The rotation of the molecule will make the obtained filtered velocities to be out of phase with the coordinates, and thus the direction of the slow degree of freedom cannot be identified with the velocities. As a result, the extra energy we put in the system could be added in any direction, making the procedure in principle the same as the HMC simulation. Thus the benefit of the DFHMC to add energy into the direction of the energy barrier will be affected negatively if not lost by the rotation. However, the amount of energy that is added is generally higher in DFHMC due to the non zero velocity bias and the scaling parameter. Thus the DFHMC would add more energy to the wrong direction and hence would be expected to perform worse compared to HMC, which we also observed in the alanine dipeptide implicit simulation results.

We analysed the effect of centre of mass rotation on the DFHMC method. An alanine dipeptide system in implicit water is simulated where the centre of mass rotation is removed at every HMC step, and the results are compared to the simulations where the centre of mass rotation are not removed. It was seen that the removal of the centre of mass rotation during the simulation caused the HMC and DFHMC to converge to different ensemble average values. The observed difference is small, and along ψ no notable difference has been observed. Thus although the observed difference was small, the removal of the centre of mass rotation disrupts the equilibrium distribution noticeably.

From the observations on the convergence of the free energy difference of the alanine dipeptide system, we conclude that the rotation of the molecule causes the DFHMC method to perform significantly worse. Thus to effectively apply the DFHMC on molecules, it is required that the molecule under simulation does not have a significant amount of rotation. As has been seen from simulations where alanine dipeptide is analysed in an explicit water environment, the rotation of the molecule could be reduced by embedding the molecule in an environment such as water that restricts the motion of the molecule.

To summarise the alanine dipeptide experiment results; The DFHMC method has been applied to the alanine dipeptide system with explicit and implicit water. It has been observed that the sampling performance of the DFHMC is better than the HMC method in explicit water; however, in implicit water, the DFHMC performs worse than the HMC. The performance of DFHMC in implicit water, improved and became better than HMC when the centre of mass rotation was removed at each HMC step during the simulation. The rotation of the molecule causes the velocities to be dephased with the positions, thereby adding energy to the wrong direction and reducing the sampling efficiency of the DFHMC method.

Two suggestions to tackle the rotation problem for future work are discussed. We believe a non-separable Hamiltonian approach to rotating the bias values and an internal molecular dynamics framework could lead to an improvement in the DFHMC

method. The Riemannian Manifold Hamiltonian (RMHMC) Monte Carlo method has been introduced to lay the foundations to the non-separable Hamiltonian approach. The application of the RMHMC method together with the softabs transformation that ensures the mass matrix is positive definite leads to trajectories that do not conserve energy if applied to a system with an energy barrier. At the point where the surface changes from a convex to concave shape the derivative of the mass matrix at that point becomes discontinuous, causing the dynamics violating the conservation of energy. The RMHMC method without any further modifications does not seem to be suitable for systems with energy barriers. Even though system dependent modifications could be done, more research would be required to find a generic metric for the mass matrix such that the method is suitable for molecular systems. However, the underlying framework could be useful in terms of laying the foundations for the further development of the DFHMC method.

It is highlighted that the Bayesian inference problem from a mathematical point is equivalent to the sampling of molecular configurations. The model parameters and the joint log likelihood is related to the positions and the potential energy of the molecular system, respectively. Therefore, looking at both problems as close variants, extending the literature review to encompass the Bayesian inference and the configurational sampling field would enrich both fields and may spark innovative methods to develop.

Preliminary simulations of a bigger molecular system, Abl kinase has been conducted. The simulation has been started from the DFG-in configuration. The DFG-in and DFG-out configuration change were not observed via DFHMC or HMC simulations. However, the sampling of the DFHMC simulation in the vicinity of the DFG-in configuration is directed along a path with slow increase of energy compared to the HMC simulation which does not show a preferred direction of sampling. This indicates that the DFHMC method is biasing the simulation along the low frequency direction. Further optimising the parameters of the DFHMC method might lead to an observation of the DFG-in, DFG-out configuration change.

The HMC type methods need to be implemented and integrated to optimised and well tested MD packages to apply the DFHMC method on biomolecules in an efficient way. The integration of the HMC module to the NAMD package via a Python wrapper showed to be very slow due to the information exchange between the HMC and MD module via hard-disk. Thus, a faster implementation is achieved by using the Plumed program to exchange information between the modules, which stores the information to be exchanged in RAM and not in hard-disk. However, on default, plumed cannot access the velocities of the system. Thus changes in plumed patches had to be done to be able to extract the velocities of the system during the simulation. It was found that the access of velocities in NAMD is not straightforward; thus, the plumed patch was modified to work with Gromacs, which has a more straightforward way to access the simulation velocities. Although the Plumed integration of DFHMC together with

the custom patch on Gromacs worked, many extra tweaks had to be done on the Gromacs code base which is specific to DFHMC and does not fit to the general code structure of Gromacs. Thus a more general implementation which is also fast was achieved by using the OpenMM simulation program. Although in OpenMM, most of the HMC based integration was able to be done through the custom integrator framework, the requirement of removing the centre of mass translation and rotation could not be done via that framework. Thus the OpenMM source code is modified and a new function called "AddremoveCom()" is added such that it can be used within the custom integrator. The current implementation runs on a CPU and is not parallelised. However, the implementation is set up such that it can be used in the OpenMM's GPU platform setting, where all the native OpenMM code will be optimised to run on the GPU.

Future works can be structured into three parts. For the implementation part, the computational performance of the current implementation could be improved further by implementing the algorithms into lower abstraction levels of OpenMM that uses C++ instead of Python. Also, the additional implemented feature, i.e., removing the centre of mass translation and rotation from velocities could be implemented to run on the GPU to achieve higher computational performance. For the methodology part, the rotation problem also described as the dephasing of the bias velocities with the molecules' positions due to centre of mass rotation induced by resetting the velocities at each HMC step is seen as the next important milestone to tackle. Two general suggestions to give starting points for the attempt to solve the rotation problem are described. We believe a non-separable Hamiltonian approach to rotating the bias values and an internal molecular dynamics framework could lead to an improvement in the DFHMC method. As a separate feature, an extension of the method could involve introducing an adaptive filter design that allows the designed filter to be changed during the simulation. For the application part, Further optimisation of DFHMC parameters towards the Abl kinase simulations might lead to an observation of the DFG-in, DFG-out configuration change. Also, a detailed analysis of the DFHMC parameters that optimise the sampling performance could prove to be useful.

Bibliography

[1] S. C. Phillips, M. T. Swain, A. P. Wiley, J. W. Essex, and C. M. Edge. *Reversible Digitally Filtered Molecular Dynamics*. The Journal of Physical Chemistry B, **107**(9):2098–2110, 2003.

[2] I. Andricioaei, A. R. Dinner, and M. Karplus. *Self-guided enhanced sampling methods for thermodynamic averages*. The Journal of Chemical Physics, **118**(3):1074–1084, 2003.

[3] M. Girolami and B. Calderhead. *Riemann manifold Langevin and Hamiltonian Monte Carlo methods*. Journal of the Royal Statistical Society: Series B (Statistical Methodology), **73**(2):123–214, 2011.

[4] A. Laio and F. L. Gervasio. *Metadynamics: a method to simulate rare events and reconstruct the free energy in biophysics, chemistry and material science*. Reports on Progress in Physics, **71**(12):126601, 2008.

[5] S. Izrailev, S. Stepaniants, B. Isralewitz, D. Kosztin, H. Lu, F. Molnar, W. Wriggers, and K. Schulten. Steered molecular dynamics. In Computational Molecular Dynamics: Challenges, Methods, Ideas, pages 39–65. Springer-Verlag, 1998.

[6] K. C. Chou and L. Carlacci. *Simulated annealing approach to the study of protein structures*. Protein Engineering, **4**(6):661–667, 1991.

[7] Y. Sugita and Y. Okamoto. *Replica-exchange molecular dynamics method for protein folding*. Chemical Physics Letters, **314**(1):141–151, 1999.

[8] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth. *Hybrid Monte Carlo*. Physics Letters B, **195**(2):216 – 222, 1987.

[9] G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, and G. Bussi. *PLUMED 2: New feathers for an old bird*. Computer Physics Communications, **185**(2):604 – 613, 2014.

[10] M. Abraham, D. van der Spoel, E. Lindahl, B. Hess, and the GROMACS development team. *GROMACS User Manual version 5.0*. 2014.

[11] P. Eastman, J. Swails, J. D. Chodera, R. T. McGibbon, Y. Zhao, K. A. Beauchamp, L.-P. Wang, A. C. Simmonett, M. P. Harrigan, C. D. Stern, R. P. Wiewiora, B. R. Brooks, and V. S. Pande. *OpenMM 7: Rapid development of high performance algorithms for molecular dynamics*. PLOS Computational Biology, **13**(7):1–17, 2017.

[12] W. Greiner, L. Neise, and H. Stöcker. *Thermodynamics and Statistical Mechanics*. Springer, 1997.

[13] M. Tuckerman. *Statistical mechanics: theory and molecular simulation*. Oxford university press, 2010.

[14] D. Frenkel and B. Smit. Understanding molecular simulation: From algorithms to applications, 2002.

[15] B. J. Alder and T. E. Wainwright. *Studies in Molecular Dynamics. I. General Method*. The Journal of Chemical Physics, **31**(2):459–466, 1959.

[16] H. C. Andersen. *Molecular dynamics simulations at constant pressure and/or temperature*. The Journal of Chemical Physics, **72**(4):2384–2393, 1980.

[17] P. H. Hünenberger. *Thermostat Algorithms for Molecular Dynamics Simulations*. In Advanced Computer Simulation, volume 173 of *Advances in Polymer Science*, pages 105–149. Springer Berlin Heidelberg, 2005.

[18] D. S. Lemons and A. Gythiel. *Paul Langevin's 1908 paper "On the Theory of Brownian Motion" ["Sur la théorie du mouvement brownien", C. R. Acad. Sci. (Paris) 146, 530-533 (1908)]*. American Journal of Physics, **65**(11):1079–1081, 1997.

[19] T. Schneider and E. Stoll. *Molecular-dynamics study of a three-dimensional one-component model for distortive phase transitions*. Phys. Rev. B, **17**:1302–1322, 1978.

[20] T. Schneider and E. Stoll. *Molecular-dynamics study of second sound*. Phys. Rev. B, **18**:6468–6482, 1978.

[21] L. Woodcock. *Isothermal molecular dynamics calculations for liquid salts*. Chemical Physics Letters, **10**(3):257 – 261, 1971.

[22] W. G. Hoover, A. J. C. Ladd, and B. Moran. *High-Strain-Rate Plastic Flow Studied via Nonequilibrium Molecular Dynamics*. Phys. Rev. Lett., **48**:1818–1820, 1982.

[23] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. *Molecular dynamics with coupling to an external bath*. The Journal of Chemical Physics, **81**(8):3684–3690, 1984.

[24] S. Nosé. *A molecular dynamics method for simulations in the canonical ensemble*. Molecular Physics, **52**(2):255–268, 1984.

[25] S. Nosé. *A unified formulation of the constant temperature molecular dynamics methods*. The Journal of Chemical Physics, **81**(1):511–519, 1984.

[26] W. G. Hoover. *Canonical dynamics: Equilibrium phase-space distributions*. Phys. Rev. A, **31**:1695–1697, 1985.

[27] H. Yoshida. *Symplectic Integrators for Hamiltonian Systems: Basic Theory*. Symposium - International Astronomical Union, **152**:407–411, 1992.

[28] L. Verlet. *Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules*. Phys. Rev., **159**:98–103, 1967.

[29] B. Leimkuhler and S. Reich. *Simulating Hamiltonian Dynamics*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.

[30] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. *Equation of State Calculations by Fast Computing Machines*. The Journal of Chemical Physics, **21**(6):1087–1092, 1953.

[31] W. K. Hastings. *Monte Carlo Sampling Methods Using Markov Chains and Their Applications*. Biometrika, **57**(1):97–109, 1970.

[32] A. Brass, B. J. Pendleton, Y. Chen, and B. Robson. *Hybrid Monte Carlo simulations theory and initial comparison with molecular dynamics*. Biopolymers, **33**(8):1307–1315, 1993.

[33] J. B. J. Fourier. *The Analytical Theory of Heat*. Cambridge Library Collection - Mathematics. Cambridge University Press, 2009.

[34] R. X. Gao and R. Yan. *From Fourier Transform to Wavelet Transform: A Historical Perspective*, pages 17–32. Springer US, Boston, MA, 2011.

[35] A. Grossmann and J. Morlet. *Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape*. SIAM Journal on Mathematical Analysis, **15**(4):723–736, 1984.

[36] A. Haar. *Zur Theorie der orthogonalen Funktionensysteme*. Mathematische Annalen, **69**(3):331–371, 1910.

[37] I. Daubechies. *Orthonormal bases of compactly supported wavelets*. Communications on Pure and Applied Mathematics, **41**(7):909–996, 1988.

[38] MATLAB. *Release 2013b*. The MathWorks Inc., Natick, Massachusetts, 2013.

[39] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C++: the art of scientific computing, Second Edition*. 2002.

[40] T. Huber and W. F. van Gunsteren. *SWARM-MD: Searching Conformational Space by Cooperative Molecular Dynamics*. The Journal of Physical Chemistry A, **102**(29):5937–5943, 1998.

[41] W. Wang, S. Cao, L. Zhu, and X. Huang. *Constructing Markov State Models to elucidate the functional conformational changes of complex biomolecules*. WIREs Computational Molecular Science, **8**(1):e1343, 2018.

[42] B. E. Husic and V. S. Pande. *Markov State Models: From an Art to a Science*. Journal of the American Chemical Society, **140**(7):2386–2396, 2018.

[43] G. Torrie and J. Valleau. *Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling*. Journal of Computational Physics, **23**(2):187 – 199, 1977.

[44] J. Kästner. *Umbrella sampling*. Wiley Interdisciplinary Reviews: Computational Molecular Science, **1**(6):932–942, 2011.

[45] O. Valsson, P. Tiwary, and M. Parrinello. *Enhancing Important Fluctuations: Rare Events and Metadynamics from a Conceptual Viewpoint*. Annual Review of Physical Chemistry, **67**(1):159–184, 2016.

[46] S. C. Phillips, J. W. Essex, and C. M. Edge. *Digitally filtered molecular dynamics: The frequency specific control of molecular dynamics simulations*. The Journal of Chemical Physics, **112**(6):2586–2597, 2000.

[47] S. Trebst, D. A. Huse, and M. Troyer. *Optimizing the ensemble for equilibration in broad-histogram Monte Carlo simulations*. Phys. Rev. E, **70**:046701, 2004.

[48] S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman. *The weighted histogram analysis method for free-energy calculations on biomolecules. I. The method*. Journal of Computational Chemistry, **13**(8):1011–1021, 1992.

[49] A. M. Ferrenberg and R. H. Swendsen. *Optimized Monte Carlo data analysis*. Phys. Rev. Lett., **63**:1195–1198, 1989.

[50] J. Kästner and W. Thiel. *Bridging the gap between thermodynamic integration and umbrella sampling provides a novel analysis method: "Umbrella integration"*. The Journal of Chemical Physics, **123**(14):144104, 2005.

[51] A. Laio and M. Parrinello. *Escaping free-energy minima*. Proceedings of the National Academy of Sciences, **99**(20):12562–12566, 2002.

[52] S. Piana and A. Laio. *A Bias-Exchange Approach to Protein Folding*. The Journal of Physical Chemistry B, **111**(17):4553–4559, 2007.

[53] A. Barducci, G. Bussi, and M. Parrinello. *Well-Tempered Metadynamics: A Smoothly Converging and Tunable Free-Energy Method*. Phys. Rev. Lett., **100**:020603, 2008.

[54] P. Liu, B. Kim, R. A. Friesner, and B. J. Berne. *Replica exchange with solute tempering: A method for sampling biological systems in explicit water*. Proceedings of the National Academy of Sciences, **102**(39):13749–13754, 2005.

[55] L. Wang, R. A. Friesner, and B. J. Berne. *Replica Exchange with Solute Scaling: A More Efficient Version of Replica Exchange with Solute Tempering (REST2)*. The Journal of Physical Chemistry B, **115**(30):9431–9438, 2011.

[56] B. Mehlig, D. W. Heermann, and B. M. Forrest. *Hybrid Monte Carlo method for condensed-matter systems*. Phys. Rev. B, **45**:679–685, 1992.

[57] W. L. Jorgensen, J. D. Madura, and C. J. Swenson. *Optimized intermolecular potential functions for liquid hydrocarbons*. Journal of the American Chemical Society, **106**(22):6638–6646, 1984.

[58] A. Mazur, V. Dorofeev, and R. Abagyan. *Derivation and testing of explicit equations of motion for polymers described by internal coordinates*. Journal of Computational Physics, **92**(2):261 – 272, 1991.

[59] V. E. Dorofeyev and A. K. Mazur. *Optimization of Numerical Algorithms for Internal Coordinate Molecular Dynamics*. Journal of Computational Physics, **107**(2):359 – 366, 1993.

[60] A. K. Mazur. *Quasi-Hamiltonian equations of motion for internal coordinate molecular dynamics of polymers*. Journal of Computational Chemistry, **18**(11):1354–1364, 1997.

[61] S.-H. Lee, K. Palmo, and S. Krimm. *A new formalism for molecular dynamics in internal coordinates*. Chemical Physics, **265**(1):63 – 85, 2001.

[62] S.-H. Lee, K. Palmo, and S. Krimm. *A new method to calculate B-matrix elements for external rotations with application to the B-matrix internal coordinate molecular dynamics formalism*. Chemical Physics Letters, **342**(5):643 – 651, 2001.

[63] S.-H. Lee, K. Palmo, and S. Krimm. *A comparative study of molecular dynamics in Cartesian and in internal coordinates: Dynamical instability in the latter caused by nonlinearity of the equations of motion*. Journal of Computational Chemistry, **28**(6):1107–1118, 2007.

[64] J. R. Wagner, G. S. Balaraman, M. J. M. Niesen, A. B. Larsen, A. Jain, and N. Vaidehi. *Advanced techniques for constrained internal coordinate molecular dynamics*. Journal of Computational Chemistry, **34**(11):904–914, 2013.

[65] A. Jain, N. Vaidehi, and G. Rodriguez. *A fast recursive algorithm for molecular dynamics simulation*. Journal of Computational Physics, **106**(2):258 – 268, 1993.

[66] I.-H. Park, V. Gangupomu, J. Wagner, A. Jain, and N. Vaidehi. *Structure Refinement of Protein Low Resolution Models Using the GNEIMO Constrained Dynamics Method*. The Journal of Physical Chemistry B, **116**(8):2365–2375, 2012.

[67] A. Jain, I. H. Park, and N. Vaidehi. *Equipartition Principle for Internal Coordinate Molecular Dynamics*. Journal of Chemical Theory and Computation, **8**(8):2581–2587, 2012.

[68] M. J. Betancourt. *A General Metric for Riemannian Manifold Hamiltonian Monte Carlo*. arXiv e-prints, page arXiv:1212.4693, 2012.

[69] M. A. Young, N. P. Shah, L. H. Chao, M. Seeliger, Z. V. Milanov, W. H. Biggs, D. K. Treiber, H. K. Patel, P. P. Zarrinkar, D. J. Lockhart, C. L. Sawyers, and J. Kuriyan. *Structure of the Kinase Domain of an Imatinib-Resistant Abl Mutant in Complex with the Aurora Kinase Inhibitor VX-680*. Cancer Research, **66**(2):1007–1014, 2006.

[70] B. Nagar, O. Hantschel, M. A. Young, K. Scheffzek, D. Veach, W. Bornmann, B. Clarkson, G. Superti-Furga, and J. Kuriyan. *Structural Basis for the Autoinhibition of c-Abl Tyrosine Kinase*. Cell, **112**(6):859 – 871, 2003.

[71] D. C. Liu and J. Nocedal. *On the limited memory BFGS method for large scale optimization*. Mathematical programming, **45**(1-3):503–528, 1989.

[72] Y. Meng, Y.-l. Lin, and B. Roux. *Computational Study of the "DFG-Flip" Conformational Transition in c-Abl and c-Src Tyrosine Kinases*. The Journal of Physical Chemistry B, **119**(4):1443–1456, 2015.

[73] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kalé, and K. Schulten. *Scalable molecular dynamics with NAMD*. Journal of Computational Chemistry, **26**(16):1781–1802, 2005.