University of Alberta

Bose-Einstein centroid molecular dynamics and applications

by

Paul Moffatt

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science.**

Department of Physics

Edmonton, Alberta
Fall 2004

Your file Votre référence
ISBN: 0-612-95817-5
Our file Notre référence
ISBN: 0-612-95817-5

Acknowledgements

I would like to express my gratitude for having had the opportunity to share a workspace with Dr. Nick Blinov. Not only did he provide clear and articulate answers to all my many questions on the spot (despite his constant insistence, "my English is not too good"), but I don't know if he even once didn't politely stand out of his chair at the sound of (chair swiveling) "uhh, Nick?"[1]

My thanks goes to both of my supervisors. Thank you to Dr. Frank Marsiglio, first for accepting me as his student and, more importantly, for keeping me in the world physics through his insightful group meetings and his lectures during his course. I am grateful that he gave me the opportunity to travel and learn from the larger physics community. I am also indebted to Dr. Pierre Nicholas Roy from the Chemistry Department. I thank him for accepting me into his group (even though I had never taken a chemistry course in my entire university career), for his patient and experienced help with all the aspects of my project, and for calling the ambulance for me the day before my thesis was due.

In addition to my supervisors and Dr. Nick Blinov, I received tremendous support and assistance from all the group members that I have had the chance to work with. In alphabetical order, I would like to thank Lucian Covaci, Javier Cuervo, Fatih Dogan, Bilkiss Issack, Wonkee Kim, Yongdong "Lucy" Liu, Mikyung Seo, XiaoGeng "James" Song, and Simona Verga.

Finally, I would like to thank Aaron Hryciw for his artistic contribution of Figure 4.1 (this is actually the second version. The first version of this picture may have been a bit too ridiculous for a thesis) and Vanessa Chong for her support during some of the stressful times over the past few years.

[1]This was taken from the Acknowledgments of Ben Harland's M.Sc thesis [1]. We both appear to have had the exact same experiences with Dr. Nick Blinov while sharing an office with him.

Contents

List of Figures

List of Tables

List of Abbreviations

BECMD	*Bose Einstein Centroid Molecular Dynamics*
BO	*Born-Oppenheimer*
CMD	*Centroid Molecular Dynamics*
DT-PIMC	*Discrete Time - Path Integral Monte Carlo*
FDCMD	*Fermi Dirac Centroid Molecular Dynamics*
F-PIMC	*Fourier - Path Integral Monte Carlo*
MC	*Monte Carlo*
MD	*Molecular Dynamics*
PIMC	*Path Integral Monte Carlo*
QDO	*Quasi-Density Operator*
QMC	*Quantum Monte Carlo*
UQDO	*Unnormalized Quasi-Density Operator*

Chapter 1

Introduction

"the underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble". A comment made by Dirac during the infancy of quantum mechanics.

Though it may be a bit harsh, in a sense, Dirac's comment is still true today. A large part of the theoretical physics community spends its efforts attempting to improve our current knowledge of the world around us not by finding new equations to solve, but by finding new ways to solve equations that have been with us for years.

Our goal here is to develop methods for the computer simulation of the quantum dynamics of atomic and molecular systems. The Centroid Molecular Dynamics (CMD) formalism allows us to express quantum dynamics in a classical-like way so that we can calculate quantum mechanical properties of the system in a similar fashion to a classical calculation. We will base our methods on current Quantum Monte Carlo (QMC) and centroid molecular dynamics methods and show by example that indeed we can calculate real time correlation functions while leaving the methods open for extension to more complicated systems.

While there are many techniques to obtain dynamical information from a quantum system, each technique has different drawbacks and advantages. We could perform exact calculations by directly solving the time dependent Schroedinger equation. Unfortunately the numerical cost of these exact calculations scales exponentially with the number of degrees of freedom in the system. There are imaginary time methods,

which are considered exact in imaginary time, however returning to real time requires advanced model fitting which is often a very difficult procedure. In contrast our CMD method hopes to take advantage of well developed algorithms such as the Monte Carlo method. Monte Carlo calculations scale linearly with the number of particles in the system. This linear scaling is a major improvement upon the exponential scaling that is required to solve the Schroedinger equation. The draw back to CMD is that it is an approximate method, part of this thesis will strive to justify CMD as a valid approximate method.

1.1 Quantum Dynamics

One of the major challenges in chemical physics is the calculation of the dynamical properties of a system. This problem is synonymous with solving the many body time dependent Schrodinger equation,

$$i\hbar\frac{\partial\psi(\mathbf{r},t)}{\partial t} = \hat{H}\psi(\mathbf{r},t), \tag{1.1}$$

where $\mathbf{r}$ is a vector representing the positions of all the particles in the system, $\hat{H}$ is the Hamiltonian of the system and $\psi(\mathbf{r},t)$ is the wavefunction. The quantity ubiquitous with the world of quantum mechanics $\hbar$ will from this point forward be eliminated by choosing units such that $\hbar = 1$, this is done merely for convenience.

1.2 Born-Oppenheimer Approximation

The ability to gain insight into physical systems analytically and computationally is in general is a complicated non-trivial problem. This is due to the fact that in an arbitrary system there are many degrees of freedom that need to be accounted for. For example, let us look at the Hamiltonian for a general system,

$$\hat{H} = -\sum_{i=1}^{N_{nuc}} \frac{\hbar^2}{2m_i}\nabla_i^2 - \sum_{j=1}^{N_{el}} \frac{\hbar^2}{2m_{el}}\nabla_j^2 + \sum_{k=1}^{N_{el}}\sum_{j<k}^{N_{el}} \frac{e^2}{4\pi\varepsilon_0|r_k - r_j|} + \sum_{l=1}^{N_{nuc}}\sum_{i<l}^{N_{nuc}} \frac{Z_l Z_i}{4\pi\varepsilon|R_l - R_i|} - \sum_{n_{el}}^{j=1}\sum_{i>j}^{N_{nuc}} \frac{eZ_i}{4\pi\varepsilon_0|r_j - R_i|}. \quad (1.2)$$

This basic Hamiltonian used in condensed matter physics and chemical physics in this form is quite cumbersome, and in many respects quite unnecessarily so. We have a system consisting of N_{el} electrons with mass m_{el} and N_{nuc} nuclear particles (ions) with mass m_{nuc}, and we have the charge of the electron e and the charge of the ion Z_{nuc}. The electronic and nuclear coordinates are r_j and R_i respectively. In this Hamiltonian we have accounted for in the first two terms the kinetic energy. The first term is associated with the kinetic energy of the ions, and the second term with the kinetic energy of the electrons. The third term is the energy associated with the interaction between electrons, the fourth term is the energy associated with the interaction between ions, and finally the fifth term is the energy associated with the interaction between ions and electrons.

The Born-Oppenheimer (BO) approximation [1] can help us reduce the complexity of this problem. The BO approximation allows us to separate the electronic and nuclear degrees of freedom. The justification for this approximation is that the nuclear motion is orders of magnitude smaller than the electronic motion. With this in mind, we can consider the nuclei to be moving around some averaged electronic charge density. Conversely, the electrons can be considered to be moving around statically positioned nuclei with classical charge distributions. There are of course many examples where the BO approximation is invalid, notably superconductivity is one of those systems where the thermal motion of the nuclei is commonly treated as a perturbation. In this work, we shall be adopting the BO approximation and we will be concerned with the nuclear motion of the atoms in some averaged potential.

[1] The BO approximation is explained in many introductory condensed matter text books, for example see [2]

1.3 Statistical Mechanics and Quantum Statistics

To understand the fundamental *how* and *why* of the basic properties of matter at a microscopic level is the subject known as statistical mechanics. Though we can gain a foundational understanding of the thermodynamics of matter without atomic detail, to be able to fully understand and appreciate the governing principles of thermodynamics we need to account for the statistics of a large ensemble of particles. That is, we need to know the similarities and differences of a system of one particle and that of a system with 10^{23} particles. In addition, only at the microscopic level can we attempt to introduce quantum mechanics, which brings in its own statistical description of reality.

An extremely useful, if not essential, construct of statistical mechanics is the partition function. The partition function is usually defined as the summation of Boltzmann factors over all the states, s, of a system,

$$Z = \sum_{s}^{allstates} e^{-E(s)\beta}, \tag{1.3}$$

where we introduce $\beta = 1/(k_b T)$ and where T is the temperature and k_B is Boltzmann's constant. This temperature dependent function Z tends to be underestimated in its importance to statistical mechanics, in fact we often use it merely as a normalization factor. However, the partition function allows us to calculate many useful temperature dependent properties of a system such as the internal energy U, heat capacity C_v, free energy F and even the pressure P.

$$\begin{aligned} U &= -\frac{1}{Z}\frac{dZ}{d\beta}, \\ C_v &= \frac{dU}{dT}, \\ F &= -k_B T \ln(Z), \\ P &= -\left(\frac{dF}{dV}\right)_{T,N} \end{aligned} \tag{1.4}$$

We will come back to the definition of the partition function to help illustrate important concepts in the formalism of Path Integrals, and in the formulation of

CMD. It is ideal to now introduce the definition of the partition function in quantum mechanics.

$$Z = Tr[e^{-\beta\hat{H}}] = \sum_{s}^{allstates} \langle s|e^{-\beta\hat{H}}|s\rangle \tag{1.5}$$

where $\hat{H}$ is the Hamiltonian operator for the system, and $|s\rangle$ is a complete set of states.

In addition to the partition function, another useful quantity in quantum statistical mechanics is the density operator,

$$\hat{\rho} = e^{-\beta\hat{H}}. \tag{1.6}$$

The density operator is in a sense the precursor to the partition function. The partition function can be written as the trace of the density matrix,

$$Z = Tr[\hat{\rho}], \tag{1.7}$$

which means that any of the properties from Equation (1.4) can be found from the density matrix. The position representation of the density matrix is defined as

$$\rho(x, x') \equiv \langle x|\hat{\rho}|x'\rangle, \tag{1.8}$$

where $|x\rangle$ is an eigenfunction of the position operator $\hat{x}$, and x and x' are two separate positions of the the particle.

In quantum mechanics we no longer have the convenience to label each particle and watch it evolve in time. We need to account for the indistinguishably of particles, and this introduces a quantum statistics of its own. In nature, particles come in one of two different classes, Bosons or Fermions, and they each have their own unique quantum statistics. Bosonic particles obey Bose-Einstein statistics and Fermionic particles obey Fermi-Dirac statistics. Though it should not be used as a definition, in fact it is merely a consequence of their statistics that a boson can be classified as a particle with integer spin (in units of $\hbar$), $S = 0, S = 1, S = 2$ etc., some examples are photons, helium-4 atoms and pions. Fermions on the other hand are particles that have half integer spin (in units of $\hbar$), $S = 1/2, S = 3/2$, etc., the most common

example of a fermion is the electrons, others include helium-3 atoms, protons and neutrons.

Aside from spin, the most significant difference between bosons and fermions is the symmetries of their wavefunctions upon particle exchange. A bosonic wavefunction is symmetric under particle exchange, that is, in a system consisting of at least two identical bosons, if we switch the labels of two particles, the resulting wavefunction will remain the same. The fermionic wavefunction is anti-symmetric upon particle exchange, that is, in a system consisting of at least two identical fermions, if we switch the labels of two particles, the resulting wavefunction will have the opposite sign. In summary we have for the bosonic/fermionic wavefunction, $\Psi^{B/F}$, being acted on by the particle exchange operator $\hat{P}$,

$$\begin{aligned} \hat{P}\Psi^{B} &= \Psi^{B} \\ \hat{P}\Psi^{F} &= -\Psi^{F}. \end{aligned} \tag{1.9}$$

This anti-symmetry of the fermionic wavefunction leads to the fundamental *Pauli Exclusion Principle*, which proclaims that two identical fermions cannot occupy the same state. There is no such restriction on bosons, in fact an infinite number of bosons can occupy the same state at any given time, this latter property is connected to the phenomenon of Bose-Einstein condensation.

In computational physics, the anti-symmetry properties of fermions causes a severe problem known as the *Fermion sign problem.* In computational simulations of quantum materials it is often required to calculate averages. When the value we are averaging is fluctuating back and forth between positive and negative numbers, the resulting subtraction often exceeds the precision of current computers. There are many proposed remedies to this, however the work following will be concerned with particles that are governed by Bose-Einstein statistics.

1.4 Correlation Functions

Of critical importance in the area of quantum dynamics, as well as in classical dynamics, is the notion of correlation functions and their connection to transport properties

and spectra. Classically we define a correlation function for some general system consisting of N particles as

$$C(t) = \langle B(0)A(t)\rangle = \int \dots \int dpdqB(p,q;0)A(p,q;t)f(p,q), \tag{1.10}$$

where we have phase space coordinates $p(t)$ and $q(t)$ which can be found from the systems equations of motion with the initial conditions that $p(0) = p$ and $q(0) = q$. The variables $B(t)$ and $A(t)$ have parametric time dependences on the dynamical phase space coordinates such that $B(t) = B(p(t), q(t))$ and $A(t) = A(p(t), q(t))$ and of course $B(0)$ and $A(0)$ are the values of $B(t)$ and $A(t)$ at $t = 0$ respectively. The phase space variables $p(t)$ and $q(t)$ as well as dp and dq should be thought of as N-dimensional objects for the N particles of the system, and we have also introduced a phase space distribution function $f(p,q)$. In the situation where $B(t) = A(t)$, we then refer to the correlation function as an *auto correlation function.*

In quantum mechanics we define the correlation function as a normalized trace over the operators $\hat{A}(t)$ at time zero and $\hat{B}(t)$ at some time t,

$$C(t) = \frac{1}{Z} Tr\left[e^{-\beta\hat{H}}\hat{B}e^{i\hat{H}t}\hat{A}e^{-i\hat{H}t}\right]. \tag{1.11}$$

Strictly speaking a correlation function tells us the correlation as a function of time between two variables (or operators if we use the quantum mechanical definition), and describes the average decay of some property of the system. However, in the 1950's Green and Kubo showed that correlation functions could be used for a wider range of topics in non-equilibrium statistical mechanics. They showed that we can use some correlation functions to describe transport properties and other macroscopic time dependent properties. It has even been stipulated [3] that correlation functions play as important a role in non equilibrium statistical mechanics as the partition function did in equilibrium statistical mechanics. This can be seen by Lars Onsager's *regression hypothesis:* The relaxation of macroscopic non-equilibrium disturbances is governed by the same laws as the regression of spontaneous microscopic fluctuations in an equilibrium system. Onsager received the 1968 Nobel Prize in Chemistry with the following citation:[2] "for the discovery of the reciprocal relations bearing his name,

[2]official text of the citation from the Nobel e-museum

which are fundamental for the thermodynamics of irreversible processes"

We can more elegantly express this hypothesis mathematically in the form of correlation functions as in [4]

$$\frac{\bar{A}(t) - \langle A \rangle}{\bar{A}(0) - \langle A \rangle} = \frac{C(t)}{C(0)}, \tag{1.12}$$

where we have defined $\bar{A}(t)$ as the phase space equilibrium average, and $\langle A \rangle$ as the time independent equilibrium average, and the correlation function we are concerned with is

$$C(t) = \langle \delta A(0) \delta A(t) \rangle, \tag{1.13}$$

where $\delta A(t)$ is the instantaneous fluctuation in $A(t)$ from its time independent equilibrium average,

$$\delta A(t) = A(t) - \langle A \rangle. \tag{1.14}$$

A commonly used example is the simple particle velocity auto correlation function

$$C(t) = \langle \mathbf{v}(0) \mathbf{v}(t) \rangle, \tag{1.15}$$

where $\mathbf{v}(t)$ is a vector representing the velocity of a single particle in a many body system. Regardless of how we calculate this autocorrelation function (which in general is not an easy task for a many body system), it can be shown that the time integral of this autocorrelation function is proportional to the self-diffusion coefficient. That is,

$$D = \frac{1}{3} \int_0^\infty \langle \mathbf{v}(0) \mathbf{v}(t) \rangle dt. \tag{1.16}$$

Of course, we are not limited to simply taking the direct time integral. In fact it is more common to see the Fourier transform of a correlation function. One example is the current autocorrelation function and its relation to the frequency dependence of the electrical conductivity,

$$\sigma(\omega) = \beta \int_0^\infty dt e^{-i\omega t} \langle J(0) J(t) \rangle. \tag{1.17}$$

We can even describe the absorption of radiation of a system via correlation functions. If we consider a system with a time dependent Hamiltonian in perturbation

theory such that

$$H = H_0 - \mathbf{M} \cdot \mathbf{E}(t), \tag{1.18}$$

where H_0 is time independent and solves $H_0\psi_j = E_j\psi_j$, $\mathbf{M}(t)$ is the total dipole moment of the system and $\mathbf{E}(t)$ is the time dependent electric field. We assume that the perturbative time dependence is *turned on* instantaneously at time $t = 0$. We can then use Fermi's Golden rule to express the probability per unit time that a transition from one state to another will occur. From this probability, we can find the spectral lineshape function $I(\omega)$, and re-write it in terms of the autocorrelation of the time dependence of the dipole moment[3],

$$I(\omega) = \frac{1}{2\pi}\int_0^\infty dt e^{-i\omega t}\langle \mathbf{M}(0) \cdot \mathbf{M}(t)\rangle \tag{1.19}$$

As a final example, we can use the density autocorrelation function to tell us about the dynamical structure factor, which can be easily found experimentally[4] for most systems

$$S(\mathbf{k}, \omega) = \int_0^\infty \frac{dt}{N}\langle \rho(\mathbf{k}, t)\rho(-\mathbf{k}, 0)\rangle e^{-i\omega t}, \tag{1.20}$$

where

$$\rho(\mathbf{k}, t) = \sum_j e^{i\mathbf{k}\cdot\mathbf{r}_j(t)} \tag{1.21}$$

is the density, and $\mathbf{k}$ is a wave vector.

Although we have only provided a short list of examples, we can see that correlation functions do indeed provide us with a tremendous amount of information. For a more thorough explanation of the general theory of correlation functions, please see [3, 4].

1.5 Overview of Thesis

As was stated, our goal is to develop practical methods to explore the quantum dynamics of atomic and molecular systems. Until now, the proper inclusion of quantum statistics in a real time dynamical simulation of many problems has remained a great

[3]For a complete derivation of this result, please see [3]

[4]Via neutron scattering experiments

challenge. We will also introduce new methods that will be essential for the extension to a general many body system.

In Chapter 2 we will begin by reviewing the concept of statistical mechanics in the language of Feynman path integrals. Then we will introduce the formalism of Bose-Einstein centroid molecular dynamics (and regular centroid molecular dynamics), which is based primarily on a concept due to Feynman [5]. The remainder of Chapter 2 will be devoted to the discussion of Monte Carlo and path integral Monte Carlo methods and their use in a centroid molecular dynamics simulation, including an original method of adding a specific constraint into a path integral Monte Carlo algorithm. Chapter 3 will show a proof of principle test that was done by the author to see how well the Bose-Einstein centroid molecular dynamics approximation holds for a model system [6]. Chapter 4 contains a series of practical algorithms developed by the author for performing CMD simulations for anisotropic systems. Though we will not be testing whether this works with quantum statistics, the formulation and algorithms developed have been left open for the inclusion of many particles and the quantum statistics.

Chapter 2

Theory and Methods

As expressed in the introduction, we wish to develop practical approaches to study the *real time* dynamics of atomic and molecular quantum systems. Unfortunately, with most real time quantum dynamics methods, we must inevitably make approximations. Our interest lies with a specific real time quantum dynamics method known as Bose-Einstein Centroid Molecular Dynamics (BECMD) [7], which is an extension of CMD [8, 9] to the case of systems that obey Bose-Einstein statistics. BECMD allows us to calculate real time correlation functions for various operators. The results for single particle correlation functions are returned in the form of a double Kubo transform of the quantum mechanical correlation functions. The most important aspect of the CMD and BECMD method is that we can create *classical-like* real time correlation functions, which can then be calculated with standard computational techniques.

The CMD and BECMD methods are approximations that we expect to break down after a long time period, as will be seen in the following sections. We expect CMD and BECMD to be exact at $t = 0$, where the length of time that the approximation is valid is dependent on the system chosen.

The following chapter will discuss in detail the necessary formalism and theory behind BECMD, and a general overview of the specialized computation techniques used in the simulations. We will begin with a brief review of Feynman path integrals in Section 2.1 which will be necessary for most of the computational work. The path integral representation will also allow for a more physical interpretation of the CMD theory. CMD was originally formulated in a path integral representation, whereas we

will present it in its recently developed operator formalism [8, 9, 7]. After this review of path integrals, Section 2.2 will explain the operator formalism of BECMD, which is a more analytically intuitive formalism than CMD's original path integral formalism.

2.1 Path Integral Formulation of Statistical Mechanics

The path integral concept was originated by Feynman [5, 10], and it was developed as a formulation of quantum mechanics completely consistent with modern quantum mechanics. In fact the path integral formulation of quantum mechanics is much more physically intuitive than the common wave formulation, but unfortunately it also happens to be considerably more mathematically arduous for even the simplest of problems. Here we shall look at a path integral formulation of Statistical mechanics, specifically how we can write the partition function in terms of a path integral.

The definition of the partition function is

$$Z = \sum_{j}^{allstates} e^{-\beta E_j} \tag{2.1}$$

where E_j is the energy eigenvalue of the Hamiltonian $\hat{H}$ acting on the state $|j\rangle$. With this in mind we can rewrite Equation (2.1) as

$$Z = \sum_{j}^{allstates} \langle j|e^{-\beta \hat{H}}|j\rangle = Tr[e^{-\beta \hat{H}}]. \tag{2.2}$$

The object inside the summation of Equation (2.2) is very similar in mathematical form to the definition of the propagator from the language of path integrals,[1]. The propagator is defined as

$$K(q', t; q, 0) = \langle q'|e^{-\imath t \hat{H}}|q\rangle. \tag{2.3}$$

We have implicitly introduced the position operator $\hat{q}$ which has a complete and orthonormal set of eigenstates $|q\rangle$, such that

[1]For a detailed account of how the propagator is derived please see [5]

$$\hat{q}|q\rangle = q|q\rangle,$$
$$\langle q'|q\rangle = \delta(q'-q),$$
$$\int dq|q\rangle\langle q| = 1. \tag{2.4}$$

We can then interpret the propagator as the probability amplitude that the particle starting at q at $t = 0$ will end up at q' after some time t.

The standard path integral expression for the propagator is, in reduced notation

$$K = \int Dq(t)e^{\imath S[q(t)]}, \tag{2.5}$$

where the integral should actually be interpreted as an infinite number of integrals over the infinite possible paths the particle could take from q to q'. We have introduced above the action functional of the system

$$S[q(t)] = \int_{t_a}^{t_b} \{\frac{\hat{p}^2}{2m} - V[x(t)]\}dt. \tag{2.6}$$

To make the connection with statistical mechanics, let us consider the situation where time is imaginary, $t = -\imath\beta$, this allows us to write the propagator as

$$K(q', -\imath\beta; q, 0) = \langle q'|e^{-\imath\hat{H}(-\imath\beta)}|q\rangle = \langle q'|e^{\beta\hat{H}}|q\rangle. \tag{2.7}$$

Now, let us insert an identity $\sum_j |j\rangle\langle j|$ to the left of the state $|q\rangle$, and, after some light algebra, we can re-write the propagator as

$$K(q', -\imath\beta; q, 0) = \sum_j e^{-\beta E_j}\langle j|q\rangle\langle q'|j\rangle. \tag{2.8}$$

If we set the initial and final positions of the particle equal, we create a closed path in imaginary time for the particle. Integrating Equation (2.8) over q leads to

$$\begin{aligned}\int dqK(q, -\imath\beta; q, 0) &= \int dq\sum_j e^{-\beta E_j}\langle j|q\rangle\langle q|j\rangle \\ &= \sum_j e^{-\beta E_j}\langle j|\int dq|q\rangle\langle q||j\rangle \\ &= \sum_j e^{-\beta E_j} = Z.\end{aligned} \tag{2.9}$$

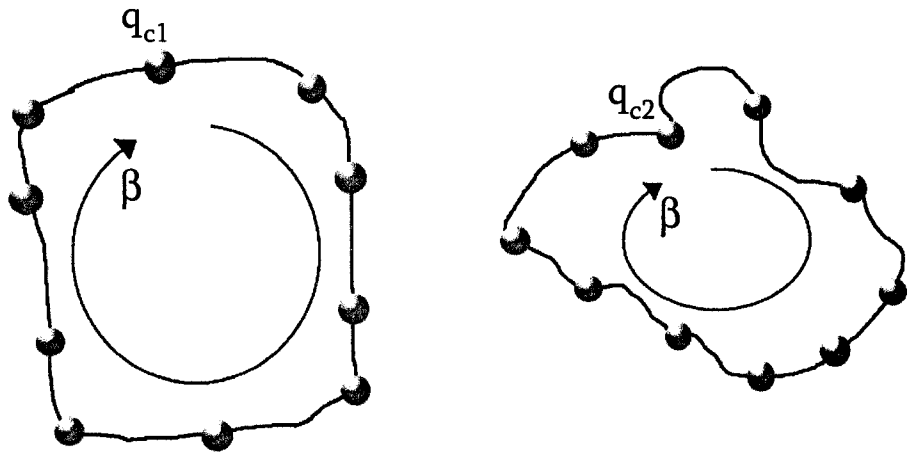

Figure 2.1: The closed Feynman paths of two indistinguishable particles that have not undergone particle exchange. Each individual particles path is discretized into $P = 10$ time slices.

In the language of path integrals, the partition function is a summation over all possible closed paths of a particle during the imaginary time β.

If one wishes to introduce quantum statistics into this formalism, it takes place quite naturally. We must account for the indistinguishably of particles. To do that, let us consider two particles. Without quantum statistics, we could imagine the particles having paths such as pictured in Figure 2.1, where each particle begins at some point $q_1$or q_2, and moves along the path, and finally, after an imaginary time β, returns to its original position q_1 or q_2. In Figure 2.2, we allow the particles to switch places. Within the time period β the particle starting at q_1 now ends up at q_2, and the particle that originated at q_2 ends at q_1. We refer to this as an exchange of particle labels, or as a permutation. This particle exchange happens in a *low temperature* (or high density) regime, where the time period β is large, and equivalently the path is long.

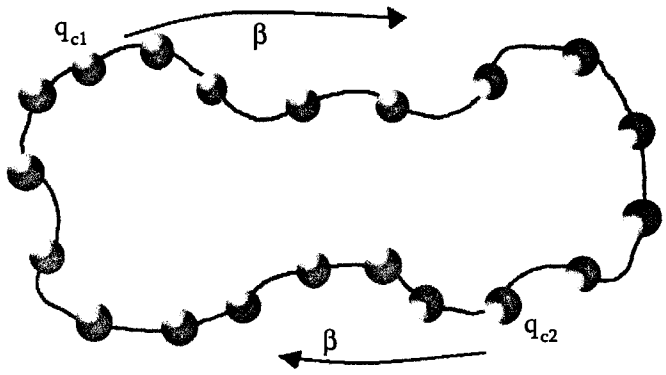

Figure 2.2: The closed Feynman path of two indistinguishable particles that have undergone particle exchange. The two particle path has been discretized identically as in Figure 2.1. After an imaginary time β the particles have exchanged places and created a closed path that is twice as long.

2.2 Feynman Path Centroid Formulation of Quantum Statistics

Starting with the Path Integral representation for a single particle[2], let us first begin with the definition of the centroid position which was suggested by Feynman [5] as one of the most classical-like objects,

$$q_c = \frac{1}{\hbar\beta}\int_0^{\hbar\beta} \mathrm{d}\tau q(\tau). \tag{2.10}$$

This is defined in terms of a closed Feynman Path Integral, with an imaginary time period in which the particle has traveled through and then returned to its original starting position. The centroid position corresponds to the centre of mass position of the Feynman Path.

Though the path integral representation of the centroid allows for a diagrammatic explanation of the centroid variable q_c, it is convenient to have an operator description for a centroid variable in general. An operator formalism has been developed [8, 9, 7] and will be explained here. It should be noted that Equation (2.10) is not necessary in the operator formalism. We have included Equation (2.10) because it gives a more

[2]The Path Integral representation for a particle will be explained in section 2.1.

physical meaning to the centroid position. From this point, the majority of the theory presented here in Section 2.2 was taken from [7], although it is not original work, it is necessary to understand the work presented in this thesis.

Within CMD, we have the ability to calculate the quantum mechanical Kubo transformed real time correlation function for two observables $\hat{A}$ and $\hat{B}$ given as,

$$\begin{aligned}\langle \hat{B}\hat{A}(t)\rangle^{Kubo} &\equiv \frac{1}{Z}\frac{1}{\beta}\int_0^\beta du Tr[e^{-(\beta-u)\hat{H}}\hat{B}e^{-u\hat{H}}e^{i\hat{H}t}\hat{A}e^{-i\hat{H}t}] \\ &= \int\int \frac{dp_c dq_c}{(2\pi)^{3N}}\frac{\rho_c(p_c,q_c)}{Z}B_c A_c(t) \end{aligned} \tag{2.11}$$

where $\hat{H}$ is the Hamiltonian of the system, q_c and p_c are 3N dimensional (for a system of N particles) vectors of the centroid position and centroid momentum and $\rho_c(p_c, q_c)$ is the phase space centroid density. The right hand side of Equation (2.11) is the CMD correlation function, which has the same mathematical form as the classical correlation function in Equation (1.10). The left hand side of Equation (2.11) shows how the CMD correlation function relates to the quantum mechanical correlation function.

For any stationary quantum mechanical observable $\hat{A}$, we can define a corresponding centroid variable $A_c(p_c, q_c)$ given by the definition

$$A_c(p_c, q_c) = Tr[\hat{\delta}_c(p_c, q_c)\hat{A}] \tag{2.12}$$

where $\hat{\delta}_c(p_c, q_c)$ is the quasi-density operator (QDO) given as

$$\hat{\delta}_c(p_c, q_c) = \hat{\varphi}(p_c, q_c)/\rho_c(p_c, q_c) \tag{2.13}$$

where

$$\hat{\varphi}(p_c, q_c) = \int\int \frac{d\zeta d\eta}{(2\pi)^{3N}} e^{i\zeta(\hat{q}-q_c)+i\eta(\hat{p}-p_c)-\beta\hat{H}} \tag{2.14}$$

is the unnormalized quasi-density operator (UQDO), and the centroid density is defined as

$$\rho_c(p_c, q_c) = Tr[\hat{\varphi}(p_c, q_c)] \tag{2.15}$$

with the convenient relation giving the partition function,

$$Z = \int\int \frac{dp_c dq_c}{(2\pi)^{3N}} \rho_c(p_c, q_c). \tag{2.16}$$

The parameters ζ and η in Equation (2.14), along with the position and momentum operators $\hat{q}$ and $\hat{p}$ in the above equations are $3N$ dimensional vectors like the phase space centroid position and momenta q_c and p_c.

Conveniently, the centroid phase space variables p_c and q_c corresponding to the quantum mechanical position and momentum operators can be written as follows:

$$q_c = Tr[\hat{\delta}_c(p_c, q_c)\hat{q}] \tag{2.17}$$

$$p_c = Tr[\hat{\delta}_c(p_c, q_c)\hat{p}]. \tag{2.18}$$

This is an important and convenient consequence of the definition of the QDO. In general a centroid variable corresponding to an operator is a function of the centroid phase space variables p_c and q_c. For example the centroid variable $(q^2)_c$ corresponding to the operator $\hat{q}^2$ is not (in general) equal to the square of the centroid variable q_c.

The above definitions have been for static variables only, but the inclusion of dynamical variables is straightforward. One can use the Heisenberg time evolution operators to include dynamics for centroid variables in the standard way,

$$\hat{\delta}_c(p_c, q_c; t) = e^{i\hat{H}t}\hat{\delta}_c(p_c, q_c)e^{-i\hat{H}t}. \tag{2.19}$$

The above equation is impractical, since it requires knowledge of the solution of Schroedinger's equation. In practical situations one has to resort to approximations. In the CMD approximation, the time dependence of the QDO is parametric with the time dependence of the centroid variables p_c and q_c

$$\hat{\delta}_c(p_c, q_c; t) \approx \hat{\delta}_c(p_c(t), q_c(t)). \tag{2.20}$$

This allows us to use the Heisenberg equations of motion

$$\dot{\hat{q}}(t) = \hat{p}(t)/m \tag{2.21}$$

$$\dot{\hat{p}}(t) = \hat{F}(t), \tag{2.22}$$

where the force operator, $\hat{F}(t)$, is minus the derivative of the potential in analogy to classical mechanics,

$$\hat{F} = -\partial\hat{V}/\partial q, \tag{2.23}$$

and the time evolution is given with the standard Heisenberg time evolution operators $\hat{F}(t) = e^{i\hat{H}t}\hat{F}e^{-i\hat{H}t}$. These create corresponding classical-like equations of motion that we can solve (either analytically or computationally depending on the system chosen)

$$\dot{q}_c(p_c, q_c; t) = p_c(p_c, q_c; t)/m \tag{2.24}$$

$$\dot{p}_c(p_c, q_c; t) = F_c(p_c, q_c; t). \tag{2.25}$$

Within the CMD approximation, we can use Equation (2.18) and Equation (2.20) to remove the centroid momentum dependence on the centroid force, resulting in the following classical like equations of motion for the centroid positions and momenta of the system

$$\dot{q}_c(t) = p_c(t)/m \tag{2.26}$$

$$\dot{p}_c(t) = F_c(q_c(t)). \tag{2.27}$$

With this formalism in place, we can now calculate centroid molecular dynamics correlation functions, which are equivalent to the Kubo transform of the exact quantum mechanical correlation function. However this has only been formulated for systems which obey Boltzmann statistics. The goal is to include quantum statistics for systems which obey Bose-Einstein statistics (as well as Fermi-Dirac statistics). This was first attempted [11] by using the following expression for the QDO, where we trace the UQDO over symmetric states (it is important to note, that this is, in general, not a valid definition for the QDO)

$$\hat{\delta}'_c(p_c, q_c) = \hat{S}\hat{\varphi}(p_c, q_c)/\rho'(p_c, q_c) \tag{2.28}$$

where the prime denotes that this is a modified operator. The $\hat{S}$ is a symmetrization operator dependent on the statistics of the system. We now can write a modified centroid density (denoted with a prime) defined with the inclusion of the symmetrization operator

$$\rho'(p_c, q_c) = Tr[\hat{S}\hat{\varphi}(p_c, q_c)]. \tag{2.29}$$

In addition, with this modified QDO we can still define a centroid variable for each quantum mechanical operator. For the case of Boltzmann statistics, $\hat{S}$ is the identity operator, and Equation (2.28) is identical to Equation (2.13) and gives us the correct statistics. However, when $\hat{S}$ is the symmetrization operator for Bose-Einstein (or Fermi-Dirac) statistics, this definition leads to centroid positions and momenta that are not equal to the corresponding centroid variables of the position and momentum operators,

$$q_c \neq Tr[\hat{\delta}'_c(p_c, q_c)\hat{q}] \tag{2.30}$$

$$p_c \neq Tr[\hat{\delta}'_c(p_c, q_c)\hat{p}] \tag{2.31}$$

from which we cannot derive the CMD equations of motion from the Heisenberg equations of motion [7]. Although the definition of the QDO in Equation (2.28) does not accurately account for quantum statistics in general, there are certain situations for which this definition of the QDO does. For example, if we use centre of mass and relative distance coordinates, which are symmetric with respect to particle exchange, Equation (2.28) does account for quantum statistical correctly.

To remedy this, an alternate formulation of the QDO has been proposed [7] and tested numerically for a model system [12]. If one defines the UQDO such that

$$\hat{\tilde{\varphi}}(p_c, q_c) = \int \frac{d\zeta d\eta}{(2\pi)^{3N}} \frac{1}{\beta}[\int_0^\beta du e^{-(\beta-u)\hat{H}'} \hat{S} e^{-u\hat{H}'}], \tag{2.32}$$

where again $\hat{S}$ is the symmetrization operator, the effective Hamiltonian $\hat{H}'$ is written for convenience to include the terms linear in position and momentum,

$$\hat{H}' = \hat{H} - i\zeta(\hat{q} - q_c)/\beta - i\eta(\hat{p} - p_c)\beta \tag{2.33}$$

where $\hat{H}$ is the ordinary Hamiltonian of the system. With this new UQDO, we can then define a centroid density which is identical to the one in Equation (2.29)

$$\tilde{\rho}_c(p_c, q_c) \equiv Tr[\hat{\tilde{\varphi}}(p_c, q_c)] = Tr[\hat{S}\hat{\varphi}(p_c, q_c)], \tag{2.34}$$

and then we can define a new QDO analogous to Equation (2.13)

$$\hat{\tilde{\delta}}_c(p_c, q_c) = \hat{\tilde{\varphi}}(p_c, q_c)/\tilde{\rho}_c(p_c, q_c) \tag{2.35}$$

Again we see that in the case of Boltzmann statistics, Equation (2.35) reduces to the corrected form of Equation (2.13). The tilde appearing in Equation (2.35) now signifies a "new" QDO which we shall refer to simply as QDO from this point on. Any reference to the operator in Equation (2.13) shall be referred to as the old QDO. In general, I will be using the tilde to refer to a quantity that correctly accounts for any quantum statistics, which are completely accounted for by the inclusion of $\hat{S}$ in Equation (2.32). In later sections, we may include a superscript on some quantities to distinguish which statistics have been chosen. However, at this point we do not make such a distinction. The tilde on the centroid density in Equation (2.35) again signifies that we are correctly accounting for quantum statistics, even though densities in Equations (2.34) and (2.29) are identical, we use the tilde in this instance to signify that there is a difference between Equation (2.34) and Equation (2.15).

With this QDO, we can again associate to each quantum observable a centroid variable,

$$\tilde{A}_c(p_c, q_c) = Tr[\hat{\tilde{\delta}}_c(p_c, q_c)\hat{A}]. \tag{2.36}$$

Most importantly, it is now possible to obtain centroid position and momentum variables that correspond to individual centroid coordinates as we did with Equation (2.17) and Equation (2.18):

$$\tilde{q}_c \equiv Tr[\hat{\tilde{\delta}}_c(p_c, q_c)\hat{q}] = q_c \tag{2.37}$$

$$\tilde{p}_c \equiv Tr[\hat{\tilde{\delta}}_c(p_c, q_c)\hat{q}] = q_c \tag{2.38}$$

.

To account for the time dependence of the QDO, we make an identical approximation for the QDO as we did for Boltzmann QDO in Equation (2.20), namely that the QDO's time dependence is parametric with the time dependence of the centroid variables themselves.

$$\tilde{\hat{\delta}}_c(p_c, q_c; t) \approx \tilde{\hat{\delta}}_c(p_c(t), q_c(t)). \tag{2.39}$$

With these definitions, we can now reformulate from the Heisenberg equations of motion, a new set of equations for the centroid dynamics again using the CMD approximation

$$\dot{q}_c(t) = p_c(t)/m, \tag{2.40}$$

$$\dot{p}_c(t) = \tilde{F}_c(p_c(t), q_c(t)) \tag{2.41}$$

, where the centroid force is defined using Equation (2.36)

$$\tilde{F}_c = Tr[\hat{\tilde{\delta}}_c(p_c(t), q_c(t))\hat{F}]. \tag{2.42}$$

It is an important difference that the centroid force, defined with the QDO of Equation (2.35) is now dependent on the centroid position and the centroid momentum, which was not the case with the old QDO. In addition, we can no longer write the centroid force as the derivative of the centroid density in analogy with other forms of MD,

$$\tilde{F}_c \neq -\frac{1}{\beta}\frac{\partial}{\partial q_c}\ln\tilde{\rho}_c. \tag{2.43}$$

As a consequence of the new definition of the QDO and the new centroid variables, we must modify the relationship between the true quantum correlation function and the BECMD correlation function. It was shown in [7] that since the new centroid variables in Equations (2.37) and (2.38) are represented by the Kubo transform of the corresponding operators we can then relate the BECMD correlation function to a second order Kubo transform of the quantum correlation function

$$\begin{aligned} \langle \hat{B}\hat{A}(t)\rangle^D &\equiv \frac{1}{\beta^2}\int_0^\beta d\mu \int_0^\beta d\nu \langle \hat{T}\hat{B}(-i\nu)\hat{A}(t-i\mu)\rangle \\ &= \int\int \frac{dp_c dq_c}{(2\pi)^{3N}} \frac{\rho_c^B(p_c,q_c)}{Z} \tilde{B}_c \tilde{A}_c(t), \end{aligned} \tag{2.44}$$

where we have introduce the superscript D, to denote a second order Kubo transform. $\hat{T}$ is the Dyson time ordering operator [13], which can be used for imaginary time as well and has the following effect:

$$\hat{T}\hat{B}(-i\nu)\hat{A}(t-i\mu) = \begin{cases} \hat{B}(-i\nu)\hat{A}(t-i\mu), \nu > \mu \\ \hat{A}(t-i\mu)\hat{B}(-i\nu), \nu < \mu\ . \end{cases} \tag{2.45}$$

2.3 Markov Chain Monte Carlo Methods and the Metropolis algorithm

In general a Monte Carlo (MC) algorithm is merely a computational method of calculating a multi-dimensional integral that cannot be (or is extremely difficult) to solve analytically or with standard numerical algorithms known as quadrature or trapezoidal methods. The method is based upon the statistics of large numbers and randomness, in fact it is suggested that its name, *Monte Carlo*, was taken from the city Monte Carlo, which is a Las Vegas style city that has numerous casinos, famous for their use of randomness and probability in gambling.[3]

Let us look at a trivial example[4] of the application of Monte Carlo. Suppose we wish to calculate

$$F = \int_a^b dx f(x). \tag{2.46}$$

[3]It is also said there is a game played by children in Monte Carlo, Monaco that exemplifies the nature of the method.

[4]This example is taken from a course on Quantum Monte Carlo Methods in Statistical Mechanics, taught by Dr. Massimo Bonninsengi 2004

where $f(x)$ is some non-negative (we make $f(x)$ non-negative for simplicity of this example) on the interval $[a, b]$. Mathematically we are merely calculating the area under the 'curve' of the function $f(x)$. We can compare this to some box that has a width $w = |b - a|$ and some known height h such that $h > f(x) \forall x \in [a, b]$. For a random number pair, (x_i, y_i) such that $a < x_i < b$ and $0 < y_i < h$ we record the number of times N_{hit} that the point (x_i, y_i) is less than $(x_i, f(x_i))$. Now, let us generate a large set of non biased random points (x_i, y_i), and we keep track of the number of random points we generate N_{trials} as well as the number of *hits* N_{hits}. Since the random points should be evenly spread over the surface of the box, the ratio N_{hit}/N_{trials} converges to the area under the curve $f(x)$ in the limit of the number of trials N_{trials} goes to infinity. This primitive method is often referred to as a *Hit or Miss* method.

A significant improvement on the *Hit or Miss* method is called *Importance Sampling.*[14] Suppose that instead of wanting to calculate a simple integral like in the example above, we want to calculate a multidimensional quantity such as

$$I = \frac{\int dx_1, \ldots dx_M f(x_1, \ldots dx_M) \rho(x_1 \ldots dx_M)}{\int dx_1 \ldots dx_M \rho(dx_1 \ldots dx_M)}, \tag{2.47}$$

which is not an unrealistic situation when we think about the quantity $\rho(x_1 \ldots x_M)$ as a density matrix, and its integral as the partition function. With the *Hit or Miss* method, we would generate a large set of random M-dimensional points and *blindly* start recording whether it *hits* or *misses*. In general this method is fine, but it may be extremely inefficient depending on what the functions $f(dx_1, \ldots x_M)$ and $\rho(x_1 \ldots x_M)$ are. For example, if $f(x_1, \ldots x_M)$ is some widely spread out Gaussian function and $\rho(x_1 \ldots x_M)$ is some sharply peaked Gaussian function centered at the same point as $f(x_1, \ldots x_M)$, we know that the product $f(x_1, \ldots x_M)\rho(x_1 \ldots x_M)$ will also be sharply peaked. By choosing randoms numbers *blindly*, we will most likely have a large number of *misses* compared to *hits*, and this will cause a very slow convergence to the correct answer. However, if we know what the function $\rho(x_1 \ldots x_M)$ looks like, we can use it to help *sample* our random numbers. Essentially, we will be choosing random numbers that are distributed so that we know they will have some

contribution to the integral. Since there is a well defined method for generating a random number according to a Gaussian distribution, importance sampling in the context of a Gaussian is sometimes called *Gaussian sampling.* It should be noted that importance sampling can be done for any function $\rho(x_1 \ldots x_M)$, as long as there is an unbiased method for generating random numbers according to that distribution.

In addition to importance sampling, we introduce the Metropolis algorithm [15], which is a Markov chain/random-walk acceptance-rejection algorithm. We can think of it as a random walk through *state-space* with states (configuration) $\{s_0, s_1, \ldots\}$ and a transition probability $P(s \to s')$ that describes the probability of a transition from state s to state s'. The difficulty is in choosing $P(s \to s')$ such that it satisfies detailed balance

$$\pi(s)P(s \to s') = \pi(s')P(s' \to s), \tag{2.48}$$

where $\pi(s)$ is a given probability density which we wish our set of random states to converge to. We can see from detailed balance that

$$\sum_s \pi(s)P(s \to s') = \pi(s'). \tag{2.49}$$

We can satisfy detailed balance by carefully choosing our transition probability as

$$P(s \to s') = T(s \to s')A(s \to s') \tag{2.50}$$

where $T(s \to s')$ is the sampling probability, and $A(s \to s')$ is the acceptance probability. We define the acceptance probability as

$$A(s \to s') = min\left(1, \frac{T(s' \to s)\pi(s')}{T(s \to s')\pi(s)}\right). \tag{2.51}$$

In practice, the sampling probability is usually the kinetic energy term of the action, since it is a Gaussian in momentum, it is easy to sample the distribution and the probability distribution $\pi(s)$ is usually the potential term of the action.

Further discussion of the implementation of general Monte Carlo methods, including a method or analysing statistical errors known as *blocking analysis* can be found in [14] or [16] as well as many other numerous resources. In blocking analysis, one

can calculate the error associated with the correlation of the data points by averaging (blocking) consecutive segments of data and then finding the variance of these blocks. As a function of the block size (number of data points averaged in a single block) we expect the ratio of the variance of the block averages over the expected uncorrelated variance to plateau at some value. At this limiting value there is little or no correlation between the blocks, and we can use the block averaged variance to calculate the *true* error estimation.

2.4 Path Integral Monte Carlo

A powerful method for simulating quantum mechanical liquids, solids and clusters is Path Integral Monte Carlo (PIMC). For our purposes it is a nearly essential method since inevitably the convenient representation of the operator formalism of CMD must be put aside and replaced with a path integral representation when we wish to develop a simulation approach. PIMC lends itself nicely to the CMD framework, and historically, CMD was originally formulated using path integral techniques.[17, 18, 19, 20, 21]

To begin with, we will look at the simplest version of a discretized time path integral Monte Carlo (DT-PIMC) method, and then in the following subsections we will describe in more detail more advanced methods, specifically, a multi level DT-PIMC method, and a Fourier Path Integral Monte Carlo (F-PIMC) method.

As with the path integral representation, we describe a particle by a closed path[5] in the imaginary time interval β. The Feynman path is discretized into P time slices and at each time slice, and we can record the position of the particle. We can visualize this as a polymer, where each *bead* on the polymer is the particle's position at that particular time slice. Each bead on the Feynman path is connected to it's neighbour by a harmonic potential (classically connected by a spring). In a many particle simulation, it is important to remember that the interaction between particles only

[5]Not all path integral based simulations require a closed path, methods such as and Path Integral Ground State (PIGS) are based upon Path Integrals yet do not have closed paths. PIMC however does require the paths to be closed

takes place at equivalent time slices.

A PIMC simulation works basically the same way as a MC calculation with the exception that we now have this strange (but simple) interaction between time slices. The simplest sampling method in PIMC is referred to as *single slice sampling*, wherein we move only one single time slice (bead) per move. We define the acceptance probability identically as we do in Equation (2.51). The sampling distribution that we use is defined as

$$T(r_i) = (2\pi\lambda\tau)^{-3/2} e^{-\frac{(r_i - r_m)^2}{2\lambda\tau}}, \tag{2.52}$$

which is a Gaussian centered around

$$r_m = \frac{r_{i-1} + r_{i+1}}{2}. \tag{2.53}$$

This sampling distribution has also been commonly referred to as *Gaussian sampling* and *free particle sampling*. The multiplicative constant in Equation (2.52) is extraneous information, since we only use the sampling distribution in a ratio in Equation (2.51) we never have to worry about it.

2.4.1 Multi Level Metropolis

The multi-level Metropolis method is a staging algorithm designed to improve the convergence of a MC calculation, and has been used extensively in Path Integral Monte Carlo simulations [22]. In a single slice Monte Carlo procedure, a bead[6] of the path is moved on average a distance proportional to $\sqrt{\lambda\tau}$. However, in the limit that P, the number of beads, goes to infinity, we see that $\tau = \frac{\beta}{P} \longrightarrow 0$, and our displacement which is proportional to $\sqrt{\tau}$ also goes to zero. From a computational viewpoint, this means that our particle would never move and the estimator we are trying to calculate would never reach a converged value. For a further discussion on the necessity of multi-slice moves, please see [22].

The multi-slice move increases the efficiency of the calculation by sampling a larger section of the path initially during the first level, and then moves smaller subsections of

[6]The term *bead* is referring to the position of the particle at a some specified time slice.

that portion of the path during the following levels. During the first level we attempt to make a coarse move that covers a full section of the path with a large effective τ_{eff}. Each subsequent higher level has a smaller effective τ_{eff} until $\tau_{eff} = \tau$, at which point all the beads in that portion of the path have been individually moved. The increase in efficiency comes from the fact that the lower level moves (with larger τ_{eff}) are attempted first. These lower level moves have lower probability of being accepted than do the higher level moves. If one of the lower level moves is not accepted, it is not worthwhile to continue, and we save computational time by not checking whether the higher level moves are accepted. We simply reject the move and continue onto the next MC step.

To actually perform the multi level algorithm we first choose a section of the path that has $2^L + 1$ slices, where L is the number of levels (which will eventually be a parameter of the simulation), starting at the bead r_i. Keeping the endpoints fixed, we sample the position of the bead $r_{i+2^{L-1}}$ with an associated $\tau_{eff} = 2^{L-1}\tau$. The bead $r_{i+2^{L-1}}$ is the midpoint between beads r_i and r_{i+2^L}, as shown in Figure 2.3.

For the second level of the move, assuming the first level was accepted by the Metropolis algorithm, the segment of the path is bisected at the midpoint, using bead $r_{i+2^{L-1}}$ as a fixed endpoint of the new segments of the path. The midpoints, beads $r_{i+2^{L-2}}$ and $r_{i+2^{L-2}+2^{L-1}}$ are then sampled with an effective $\tau_{eff} = 2^{L-2}\tau$. These moves are then tested with the Metropolis algorithm and, if accepted, the next level bisects each segment again, doubling the number of segments and halving the value of τ_{eff}.

At each level l the Gaussian distribution (free particle sampling probability) T_l used is

$$T_l = (2^l \tau \lambda \pi)^{(-3/2)} e^{-\frac{(r-r_m)^2}{2^l \tau \lambda}} \tag{2.54}$$

, where $2^l\tau$ is the effective τ_{eff} at level l and we have centered this Gaussian at $r_m = \frac{1}{2}(r_a + r_b)$, where r_a and r_b are the two endpoints of the segment.

It is important to stress that a multi level move is *only* accepted if *all* the levels have been accepted individually. This is why we simply reject the whole step if one

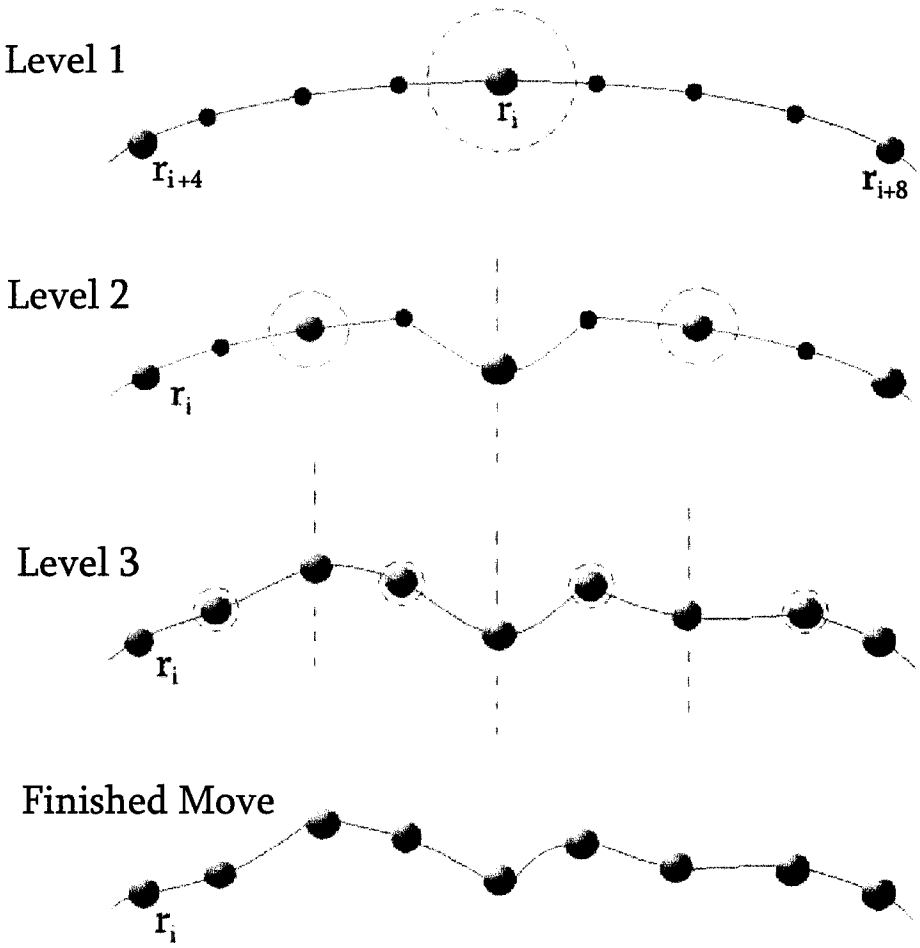

Figure 2.3: For a $L = 3$ level move, we choose a segment of the Feynman path starting at r_i. At level one, time slices r_i and r_{i+8} are treated as endpoints, and we move time slice r_{i+4} with $\tau_{eff} = 2^2\tau$, the dashed circle represents the size of the move we are allowed to make. In level two,*if the move in the first level was accepted* we bisect the path segment at r_{i+4} and we attempt to move time slices r_{i+2} and r_{i+6} using $\tau_{eff} = 2^1\tau$. At level three, *if both the moves in level 2 were accepted*, we bisect the path segment again at r_{i+2} and r_{i+6} and we attempt to move time slices r_{i+1}, r_{i+3}, r_{i+5} and r_{i+7} using $\tau_{eff} = \tau$. If *all* the moves have been accepted, only then do we keep the final configuration.

single level is rejected. We don't want to waste computational time by attempting a move to an unfavourable configuration of the path.

2.4.2 Fourier Path Integral Monte Carlo

Though DT-PIMC has been the most successful and widely used method of PIMC, for our purposes the Fourier path integral Monte Carlo (F-PIMC) [23] algorithm will later introduce a convenient method of constraining the centroid position of a Feynman path, which will be a necessary part of our algorithm to do CMD. In the Fourier path integral approach the displacements of the Feynman path integral are represented by a Fourier series of the path. In recent literature,[23, 24, 25, 26, 27, 28, 29, 30, 31] the Feynman path was represented by the Fourier sine series

$$q(u) = q + \frac{u}{\beta}(q' - q) + \sum_{k=1}^{\infty} a_k sin(k\pi u/\beta), \tag{2.55}$$

where the path begins at q and after an imaginary time β, ends at q'. The Fourier components a_k are $3N$ dimensional vectors for each Euclidean component of the N-particles. However, when the end point of the path is identical to the starting point $q' = q$, as in the case of a closed Feynman path, it is possible to use a full Fourier expansion to represent the path [25], such that

$$q(u) = \sum_{k=-\infty}^{\infty} c_k e^{i2\pi ku/\beta}, \tag{2.56}$$

where c_k is the full Fourier series coefficient, and is a $3N$ dimensional vector. It is important to realize that $q(u)$ is a purely real object, and as such $c_{-k} = c_k^*$ allows for simplification in the application of this method. We would like to use this representation of the Feynman path to make the moves in a Metropolis algorithm. This requires us to rewrite some of the path integral results in terms of a Fourier expanded path.

We can write down the path integral representation of the quantum mechanical partition function as

$$Z = J \int dq \int Dq(u) e^{-S[q(u)]}, \tag{2.57}$$

where J is a constant whose value is not important since we will always be concerned with the ratio of two such objects. In the argument of the exponential is $S[q(u)]$, the imaginary time action functional of the system written in a non discretized form,

$$S[q(u)] = \int_0^\beta du\{\frac{m\dot{q}^2(u)}{2} + V[q(u)]\}. \tag{2.58}$$

Substituting Equation (2.56) into Equation (2.58), the integral for the kinetic term of the action can be solved analytically:

$$S[q(u)] = \beta \sum_{k=1}^{\infty} m(\frac{\pi n}{\beta})^2 |c_k|^2 + \int_0^\beta du V[q(u)]. \tag{2.59}$$

This can then be used in Equation (2.57), or any path integral expression.

As a computational method, one must of course truncate the infinite summation in Equation (2.59). In DT-PIMC methods, we have a ubiquitous parameter P which is the number of beads or time slices of the Feynman path. Similarly, in Fourier path integral techniques, we choose some finite number M of Fourier modes. If one intends to remain in Fourier space to calculate estimators of observables, it is advised [23, 25, 24] to use a partial averaging technique, which approximately accounts for the upper Fourier modes which have been neglected by correcting the action. In the work following, we do not use any partial averaging, and do not try to account for the neglected higher modes. Instead, we only adjust the path in Fourier space, and then transform back to real space to perform the Metropolis acceptance rejection algorithm and calculate averages.

The Fourier expanded kinetic term in the action, when placed back into the path integral expression, still remains in a Gaussian form. Now however, we have a sum of quadratic terms instead of a single quadratic term. In contrast to DT-PIMC methods, instead of moving a single time slice, or even a segment of the path, we move the entire path, ie: we move *all* of the Fourier components.

The move is made uniformly in a cube about the old coordinates such that the new Fourier component is given by

$$c'_k = c_k + \Delta\sigma_k(u_k - 0.5), \tag{2.60}$$

where Δ is a displacement parameter (magnitude of the displacement) that can be adjusted to encourage convergence. Δ must also be changed with the number of Fourier components M to keep the acceptance rate at a decent[7] value. This can be accomplished by scaling Δ linearaly with the number of Fourier components M. u_k is a uniformly distributed random number in (0,1), and $\sigma_k = \sqrt{\frac{\beta}{4m\pi^2k^2}}$. We can see that for each Fourier component the move is weighted by $\frac{1}{k}$. This can cause problems with the zeroth, $k = 0$ mode. This can be circumvented in two ways, first, we can shift the indices of the modes so that $k \to k + 1$ which should not have a dramatic affect on moves in the large M limit, or we can omit the $k = 0$ mode from the move, since the we have the adventitious fact that the $k = 0$ mode is

$$c_0 = \frac{1}{M}\sum_{i=0}^{M} q_i, \tag{2.61}$$

which is simply the centre of mass, or centroid position of the path, so we can move the centre of mass position in real space, and move the shape of the path in Fourier space.

2.5 Constraints in Quantum Monte Carlo

In a standard DT-PIMC or F-PIMC calculation, we can define a quantity q_c, which is a Cartesian vector given by

$$q_c = \frac{1}{P}\sum_{i=0}^{P} q_i \tag{2.62}$$

where P is the number of time slices in the PIMC calculation (Not the number of modes used in the F-PIMC method), and q_i is the position of the particle at the

[7]By this we mean that the acceptance rate should be about 30% to 50%.

$i-th$ time slice. In the $P \to \infty$ limit, for a closed Feynman path, this is exactly the centroid position.

With this definition, we would now like to use a MC calculation to calculate properties of the system. Specifically we want to calculate the centroid force F_c for later use in a MD simulation, when the particle is confined to having a centroid position pinned at q_c. That is, we want to constrain the freedom of the 'centre of mass' or centroid position of the Feynman Path. This implies that during a PIMC calculation, any move we make is only allowed to alter the shape of the path as long as q_c remains fixed.

First of all, in a general PIMC algorithm, we have what is referred to as a 'global' move of the path. This is where the position of the particle at every time slice is displaced by some random number, such that for all time slices

$$q_i \to q_i + \Delta r_{random}, \tag{2.63}$$

where r_{random} is a random number in the interval $(-1, 1)$ and is the same for all time slices. The parameter Δ governs the magnitude of the displacement and is usually on the order of the thermal wavelength λ_T. It is used as an adjustable parameter which only affects the acceptance rate and the rate of convergence of the calculation. Obviously a global move changes the value of q_c by exactly Δr_{random}, in fact a global move does nothing but change the position of the centroid, leaving the shape of the path the same. For a constrained system, we have no choice but to eliminate moves of this type in the algorithm.

Path shape altering moves such as single slice or multi slice moves primarily change the shape of the path. Single slice moves change the shape of the path one slice at a time, although it also changes q_c. In a multi slice move, we attempt to alter the shape of the path by *wiggling* a segment of the path. Unfortunately this also moves the centroid position.

2.5.1 Constraint in Discrete Time Path Integral Monte Carlo

The first, naive attempt to add a constraint into a DT-PIMC calculation would be to simply re-locate the path of the particle after each Metropolis move. This can simply be done by calculating the centroid position of the new path q_c', and subtracting from each time slice the difference $q_c - q_c'$ of the two centroids. This, however, is now changing the position of all the time slices without a sampling distribution, this breaks our Markov chain random walk.

Instead, let us look back at the path integral representation of the density matrix and add in the constraint for the centroid position in the form of a delta function

$$\rho(q_c;\beta) = \int D(u) e^{-S[q(u)]} \delta(q_c - \sum_{i=0}^{\infty} \frac{q_i}{P}), \tag{2.64}$$

where the infinite sum inside the delta function is the the centroid position of the path after the move. We must keep in mind that the infinite sum will be truncated down to $\sum_{i=0}^{P} \frac{q_i}{P}$, where P is the number of time slices in the MC simulation.

Now, let us approximate the delta function in Equation 2.64 by a Gaussian function with a finite, but small, width $\frac{\alpha}{\beta}$. Since the action of the system is found as the exponent in the density matrix, this then allows us to write a modified discretized action as

$$S(q,q';\tau) = 3N/2 \log(4\pi\lambda\tau) + \frac{(q-q')^2}{4\lambda\tau} + \tau V(q,q') + \frac{\alpha}{\beta}(q_c - \sum_{i=0}^{P} \frac{q_i}{P})^2, \tag{2.65}$$

where α is a user defined positive number that needs to be large enough for the Gaussian function to appear *delta function like.* In a computational simulation, α will be used as a parameter, and even though we have the freedom to choose it arbitrarily, there are some limitations we must consider. First, we must not choose α too large, to illustrate this, consider a configuration of the path that is already in a position where $q_c = \sum_{i=0}^{P} \frac{q_i}{P}$. If α were incredibly large, the centroid constraint term would dominate over the action and any displacement from this configuration would always be rejected during the metropolis procedure. This would result in a simulation that

would be frozen at that configuration and any estimator we are trying to calculate would very slowly converge to the correct value.

The first two terms in Equation (2.65) correspond to the kinetic energy. The first of those terms is not position dependent; in fact it is a constant. When we perform a PIMC calculation, the contribution from this constant term will always be canceled since we will be looking at a ratio of equations similar to Equation (2.64). The second term in the kinetic energy has the same form as the constraint, the square of the difference of a position. We know that the kinetic energy has a significant contribution to the action of the path, therefore we know that for the constraint to hold we need to have an equal or greater weighting on the constraint compared to the kinetic energy term. This leads to a minimum requirement that $\alpha > \frac{P}{4\lambda}$.

This method of constraining the centroid position is completely new, and has now been tested. The results will be summarized in Section 4.2.

2.5.2 Constraints in Fourier - Path Integral Monte Carlo

Constraining the centroid position in F-PIMC is significantly simpler than in DT-PIMC. Let us revisit Equation(2.61), which describes the zeroth mode of the Fourier series expansion of the Feynman path

$$c_0 = \frac{1}{M}\sum_{i=0}^{M} q_i. \tag{2.66}$$

It is extremely important to realize that this $k = 0$ mode is *identically the centroid position.* To constrain its value, we simply do not change its value when making a move.

In no way are we breaking the Markov chain, or are we making an approximation. We have simply found a type of MC move that does not alter the centroid position. As such, we can use this as an exact method of constraining the centroid position.

Since the constrained F-PIMC method is not based on a effective action like the constrained DT-PIMC method, nor does it alter the Markov chain, we can consider the constrained F-PIMC method exact (in a statistical MC sense). The time scale of the constrained F-PIMC method is slightly faster than the constrained DT-PIMC

method and, in addition, there is room for significant improvement by using a method of partial averaging [25, 23]. The main limitation for this method is that it does not include quantum exchange effects in a many body system. To date, there has been very little progress in the inclusion of quantum statistics in a F-PIMC simulation, though there has been some work done in [30]. It is however possible to include exchange in DT-PIMC [22].

Chapter 3

Proof of Principle in One Dimension

With the formalism of Section 2.2 we can now attempt to assess the accuracy of the BECMD approximation for *single-particle* time correlation functions. This is an important step since we are in a regime where we expect quantum exchange effects to take place, at this point we can no longer distinguish between the particles under normal circumstances. This proof of principle calculation allows us to judge what type of potentials will work well within the CMD approximation since we will have an exact solution to compare to. Some of the results presented here have appeared in the article entitled *On the calculation of single particle time correlation functions from Bose-Einstein Centroid Molecular Dynamics*, [6].

3.1 Physical System

It is possible to write an analytical solution in the BECMD formalism for the case of two bosons interacting in a 1 dimensional harmonic potential trap [12]. However, in this case, the CMD approximation is exact and we do not learn any information about the validity of the approximation. To accurately test this method, we need to choose a potential that is anharmonic. Our choice is a 1 - dimensional potential of the form

$$V(x_1, x_2) = \frac{1}{2}m\omega^2(x_1^2 + x_2^2) + c(x_1^3 + x_2^3) + g(x_1^4 + x_2^4), \tag{3.1}$$

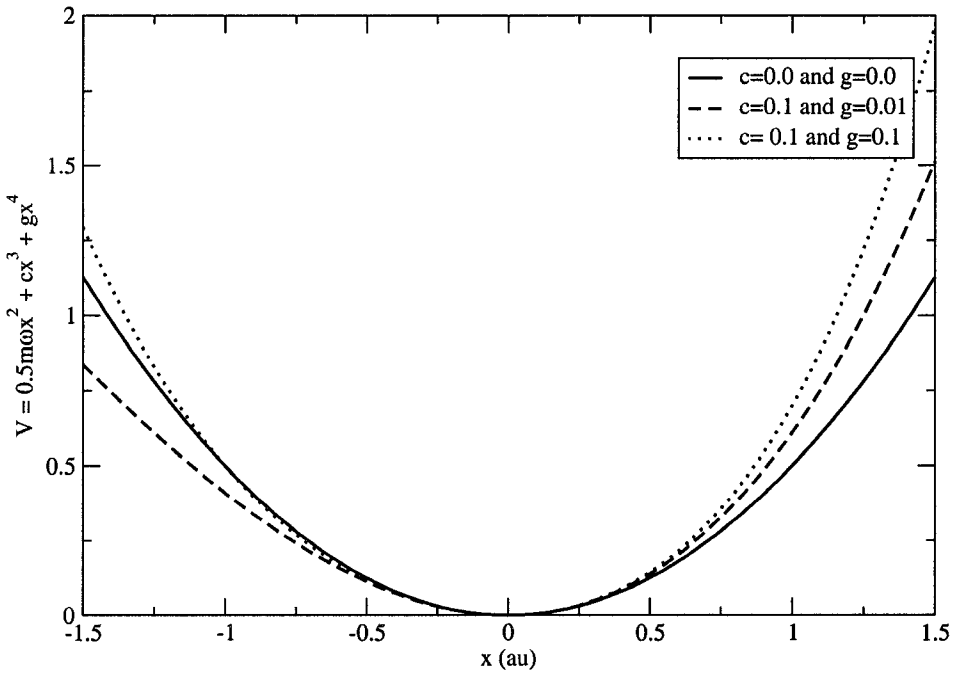

Figure 3.1: The potential for our system as a function of the position x. The solid line is the potential with $c = 0$ and $g = 0$ which is the harmonic potential. The dashed line is the potential with $c = 0.1$ and $g = 0.01$ and the dotted line is the potential with $c = 0.1$ and $g = 0.1$.

where ω is the frequency of the trap, m is the mass of the particle, and the parameters c and g are chosen arbitrarily with the condition that $g > 0$ for stability. This model has also been used to test the validity of the CMD method without exchange [9] and has also been used to test CMD with exchange using centre of mass and relative distance coordinates [11]. By varying the parameters we can test a wide range of potentials while in most cases staying close to a harmonic system. Figure 3.1 shows the potential as a function of position for a few choices of parameters. This potential with the specific choice of parameters $c = 0.1$ and $g = 0.01$ was used to test the BECMD method in [6].

Since the system is small, consisting of only two bosons, and the potential has a simple analytic form, we are able to calculate exactly the centroid density and the centroid force prior to the calculation, as opposed to more advanced algorithms that would calculate the force *on the fly*.[1]

3.2 Analysis of the Centroid Density and Centroid Force

Within the BECMD (and FDCMD) formalism, we can write the general centroid density, with the inclusion of exchange as

$$\rho(x;\beta) = \frac{1}{N!}\sum_{\wp}(\pm 1)^{\wp}\prod_{i=1}^{N}\rho^{D}(x_1, x_2;\tau)\ldots\rho^{D}(x_P, \wp x_{P+1};\tau), \tag{3.2}$$

where $\wp$ indicates the permutation of two particles (since any permutation of N particles can be found by a successive number of two particle permutations we need not consider anything higher than two particle permutations), P is again the discretization of the path, and we have introduced the *distinguishable* density $\rho^{D}(x;\beta)$, which is the density if we were to have an ensemble of particles that obeyed Boltzmann statistics (i.e. a system of distinguishable particles).

This method of accounting for permutations in the density is identical for the centroid density as well. Explicitly, since we will be looking at the case of two bosons, we can write the full centroid density as

$$\rho_c^{B/F}(x_{c1}, x_{c2}, p_{c1}, p_{c2}) = \frac{1}{2}[\rho_c^{D}(x_{c1}, x_{c2}, p_{c1}, p_{c2}) \pm \rho_c^{L}(x_{c1}, x_{c2}, p_{c1}, p_{c2})], \tag{3.3}$$

where we now have the *linked* density labeled by the superscript L, which corresponds to the density with an exchange of particle labels, i.e. linked paths as in Figure 2.2.

It may be more informative to see how Equation (3.3) is found using the QDO, as opposed to an extension from Equation (3.2). By using Equations (2.32), (2.33), and (2.34) and noting that we have re-adopted the notation that q_c, p_c, ζ and η are

[1] By *on the fly* we mean that we calculate the necessary quantity at that configuration only when we need it, as opposed to pre-calculating and reading from a table.

all vectors, i.e. $q_c = (x_{c1}, x_{c2})$ and $\zeta = (\zeta_1, \zeta_2)$ (unless specifically labeled with a subscript), we can write the two particle centroid density as

$$\rho_c^{B/F}(x_c, , p_c) = Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2}\frac{1}{\beta}\int_0^\beta du e^{-(\beta-u)(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)} \times \hat{S}^{B/F} e^{-u(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)}\right], \tag{3.4}$$

where $\hat{S}^{B/F}$ is the symmetrization operator. For a two particle system we can write it as

$$\hat{S}^{B/F} = \frac{1}{2}(\hat{I} \pm \hat{P}), \tag{3.5}$$

where $\hat{I}$ is the identity permutation, which makes no modification to anything it operates on, and $\hat{P}$ is the permutation operator,[2] which changes particle labels.

By inserting Equation (3.5) into Equation (3.4) we can split the centroid density into two terms:

$$\begin{aligned}\rho_c^{B/F}(q_c, p_c) = & \frac{1}{2}Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2}\frac{1}{\beta}\int_0^\beta du e^{-(\beta-u)(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)} \times \hat{I} e^{-u(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)}\right] \\ & \pm\frac{1}{2}Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2}\frac{1}{\beta}\int_0^\beta du e^{-(\beta-u)(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)} \times \hat{P} e^{-u(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)}\right].\end{aligned} \tag{3.6}$$

Looking at the first term in Equation (3.6), applying the identity operator (which does nothing), and utilizing the cyclic invariance of the trace allows us to re-write the first term, which we will label ρ_1, as

$$\rho_1 = \frac{1}{2}Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2}\frac{1}{\beta}\int_0^\beta du e^{-\beta(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)}\right]. \tag{3.7}$$

The integral over u in Equation (3.7) is now trivial, and leads to

$$\rho_1 = \frac{1}{2}Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2} e^{-\beta(\hat{H}-i\zeta(\hat{q}-q_c)/\beta-i\eta(\hat{p}-p_c)\beta)}\right]. \tag{3.8}$$

[2]The notation here begins to get a little cluttered. Previously we used P to denote discretization in a Feynman path, and we used $\wp$ to denote permutations. Here $\hat{P}$ has the same meaning as $\wp$ except it is in operator form

The final four integrals (over ζ_1, ζ_2, η_1 and η_2), which are mathematically equivalent to Fourier integrals, are performed numerically using the FFTW numerical library [32], and ρ_1 is identically $\rho_c^D(q_c, p_c)$.

The second term from Equation (3.6) cannot be analytically simplified as much as the first term. We define the entire second term as $\rho_c^L(x_c, p_c)$, which then allows us to rewrite Equation (3.6) exactly as Equation (3.3).

As was previously mentioned, for this proof of principle we calculate the centroid force and the centroid density prior to performing the dynamics. A brief explanation of the calculation of the centroid density was given above. The calculation of the centroid force is nearly identical. We first start off with the definition of the centroid force in terms of the QDO.

$$F_c^{B/F}(p_c, x_c) = Tr\left[\hat{\delta}_c^{B/F}(p_c, x_c)\hat{F}\right], \tag{3.9}$$

where again, x_c and p_c are treated as vectors for the particles. With Equation (2.35) and Equation (2.32) we can rewrite Equation (3.9) in similar two part expression as we did with the centroid density:

$$\begin{aligned} F_c^{B/F}(x_c, p_c) \;=\; & \frac{1}{\rho_c^{B/F}(x_c, p_c)} \frac{1}{2} Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2} \frac{1}{\beta} \int_0^\beta du e^{-(\beta-u)(\hat{H}-i\zeta(\hat{q}-q_c)/\beta - i\eta(\hat{p}-p_c)\beta)}\right. \\ & \left. \times \hat{I} e^{-u(\hat{H}-i\zeta(\hat{q}-q_c)/\beta - i\eta(\hat{p}-p_c)\beta)} \hat{F}\right] \\ & \pm \frac{1}{\rho_c^{B/F}(x_c, p_c)} \frac{1}{2} Tr\left[\int\int \frac{d\zeta d\eta}{(2\pi)^2} \frac{1}{\beta} \int_0^\beta du e^{-(\beta-u)(\hat{H}-i\zeta(\hat{q}-q_c)/\beta - i\eta(\hat{p}-p_c)\beta)}\right. \\ & \left. \times \hat{P} e^{-u(\hat{H}-i\zeta(\hat{q}-q_c)/\beta - i\eta(\hat{p}-p_c)\beta)} \hat{F}\right]. \end{aligned} \tag{3.10}$$

As with the centroid density, we can label the first term as distinguishable and the second term as linked (with exchange). The Kubo integral in both of these terms (as with the linked term from the centroid density) will be calculated numerically.

The trace operation and the four Fourier-like integrals are both linear operations, which allows us to bring the trace operation inside to the integrand of the Fourier-like integrals. This is merely a convenience that makes both the trace operation and the Fourier-like integrals significantly easier. Once we calculate the trace of the integrands

we have a centroid density and centroid force in "ζ-η space", which is a reciprocal space to the centroid phase space. Numerically we then used a fast Fourier transform to create the true centroid density and centroid force.

Since in our examples the centroid density and centroid force are four dimensional objects in x_{c1}, x_{c2}, p_{c1} and p_{c2}, we cannot view them in their full form. Instead we look at them as a function of one variables, where the other variables have been integrated over. This allows us some insight into the shape of the centroid density and force.

In Figure 3.2, we see the plot of the integrated one dimensional centroid density as a function of x_{c1}. Note that the dependence on x_{c1} and x_{c2} are identical. In Figure 3.3 is the integrated centroid density as a function of p_{c1}, where the dependence on p_{c1} and p_{c2} are identical as well. As a function of centroid momentum, we see that they are completely symmetric about $p_{ci} = 0$. As a function of centroid position, however, we see that the centroid density is not centered about $x_c = 0$.

It was stated that the centroid force may in general also depend on the centroid momentum. Let us look at the centroid force in the same way we looked at the centroid density. In Figures 3.4 and 3.5 we can see the centroid position and momentum dependencies of the integrated centroid force weighted by the centroid density. It was shown [6] that there is a small momentum dependence of the centroid force (relative to the position dependence), i.e. the momentum dependence is roughly two orders of magnitude smaller than the position dependence. In addition, we have assumed that the centroid force is a completely real quantity. We can see that the imaginary components of the centroid force weighted by the centroid density are several orders of magnitude smaller than the real components of the centroid force This is true for both the centroid position and momentum dependence.

It is also interesting to look at the centroid force if using the *old* QDO as defined in Equation (2.13). In Figure 3.6 we can see that the general shape of the centroid force is retained, but there are some differences with the magnitude, especially when we look at the integrated centroid force as a function of centroid position of the particle that the potential was not differentiated with respect to, i.e. the centroid force on particle 1, as a function of the centroid position of particle 2.

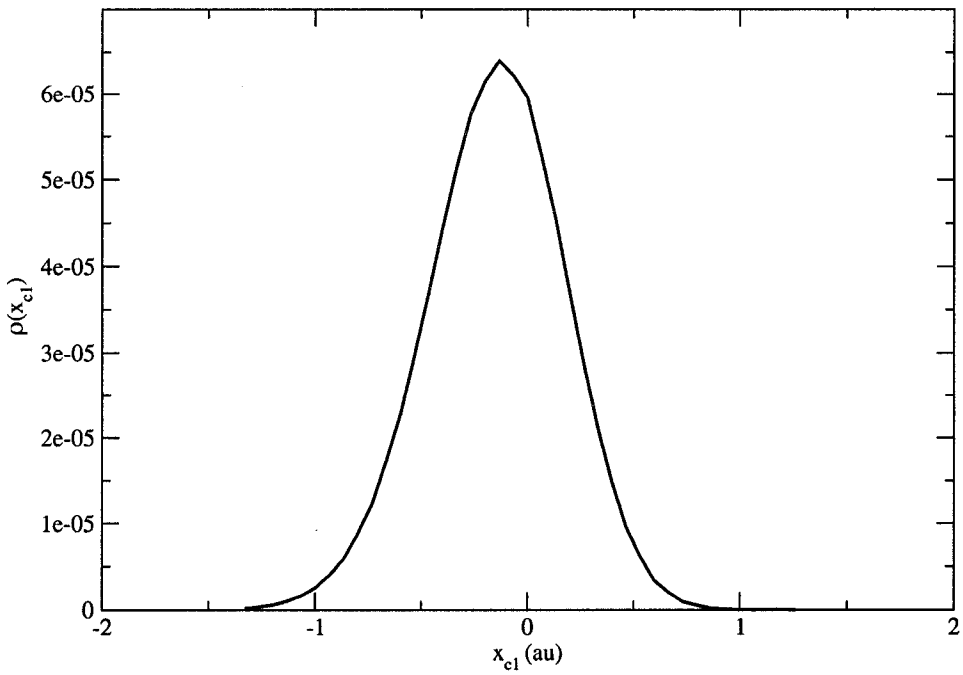

Figure 3.2: The centroid density as a function of the centroid position of particle 1. The centroid density has been integrated over the remaining three centroid phase space variables.

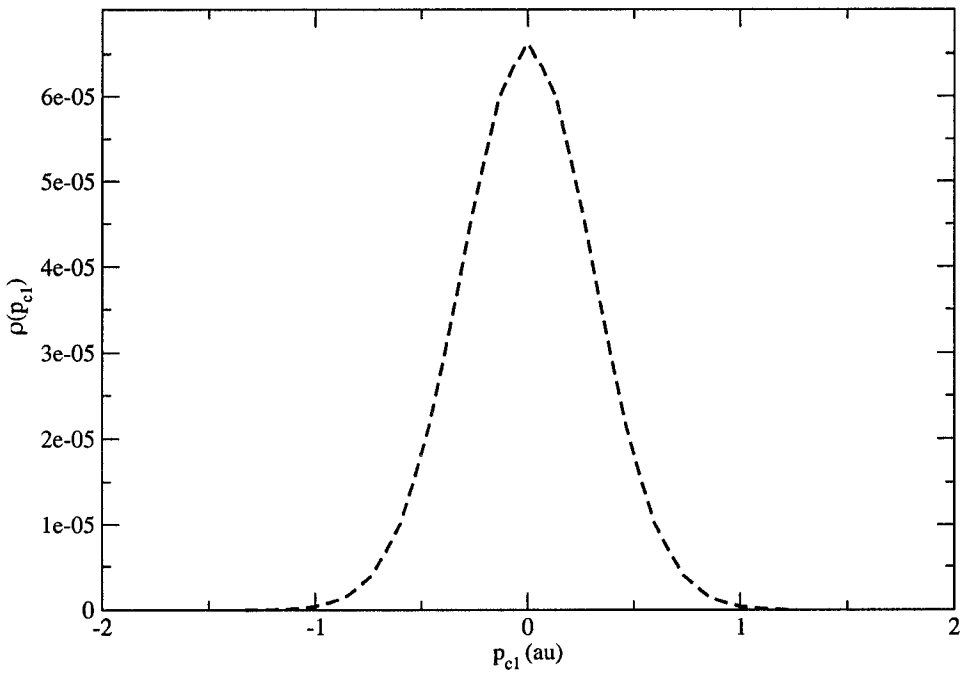

Figure 3.3: The centroid density as a function of the centroid momentum of particle 1. The centroid density has been integrated over the remaining three centroid phase space variables.

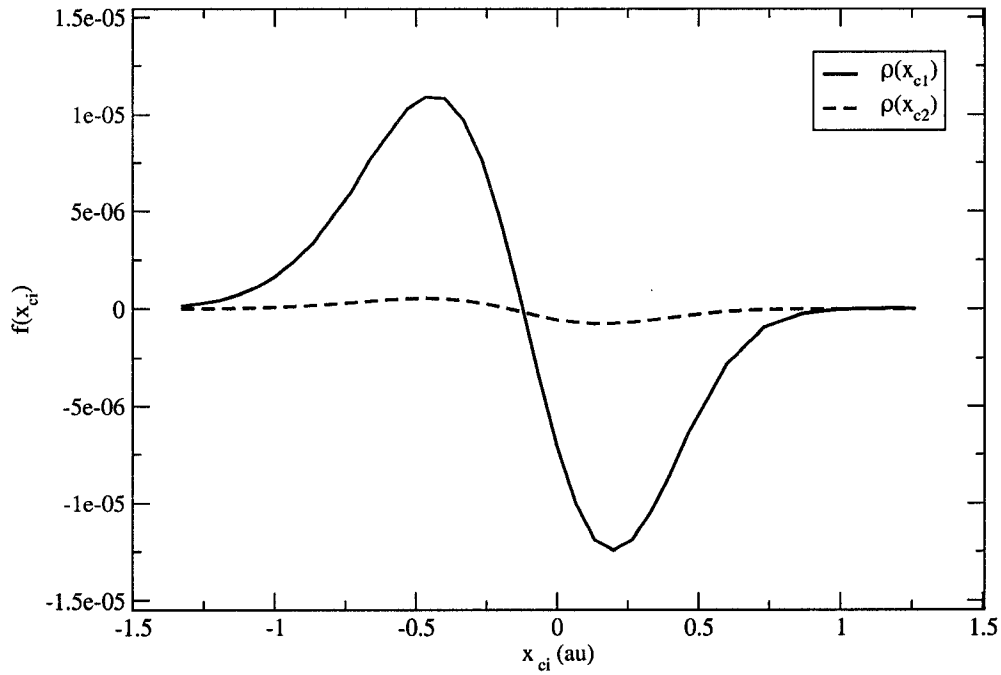

Figure 3.4: The centroid force has been weighted by the centroid density and integrated over the centroid momenta and one of the centroid position variables. The solid line is for the dependence on the centroid position of particle one and the dashed line for particle two.

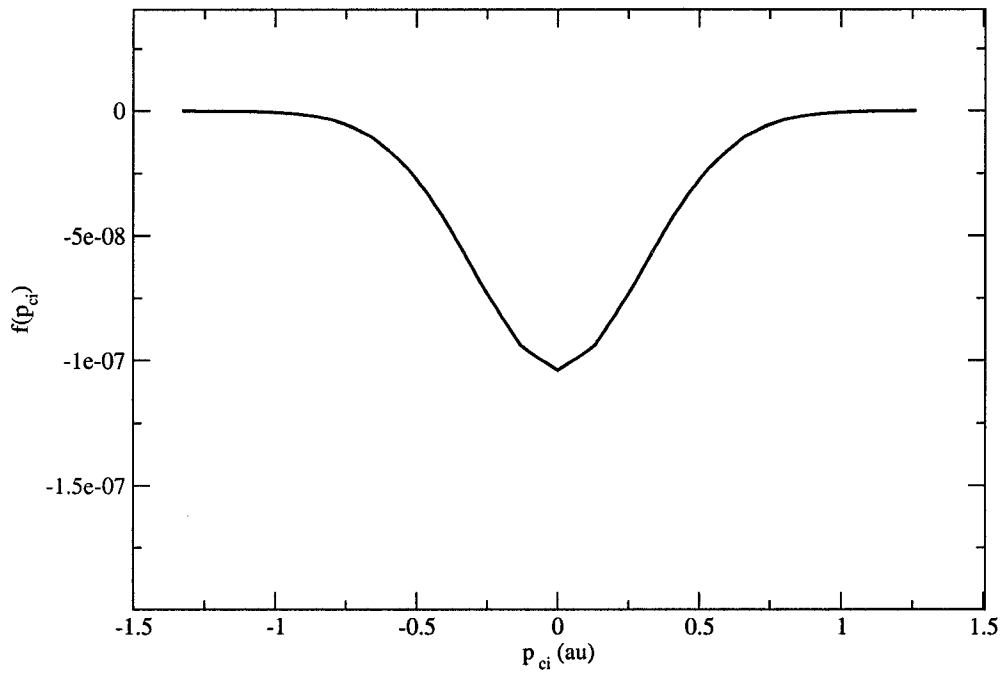

Figure 3.5: The centroid force has been weighted by the centroid density and integrated over the centroid positions and one of the centroid momentum of particle two. The dependence on momentum is identical for both particle one and particle two.

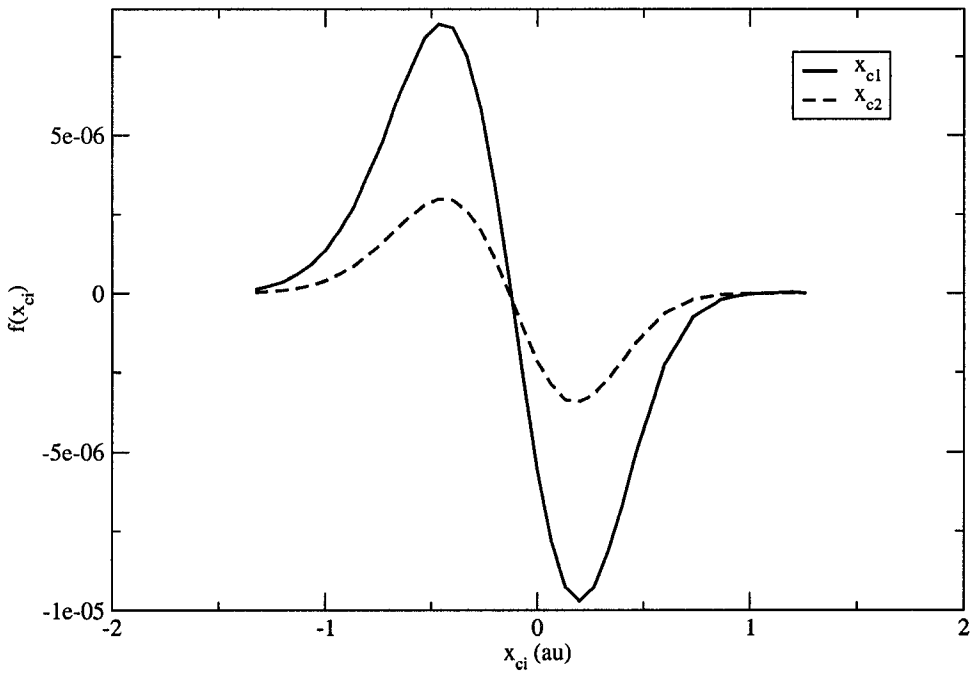

Figure 3.6: The centroid force as defined using the old QDO from Equation (2.13) has been weighted by the centroid density and integrated over the centroid momenta and one of the centroid position variables. The solid line is for the dependence on the centroid position of particle one and the dashed line for particle two.

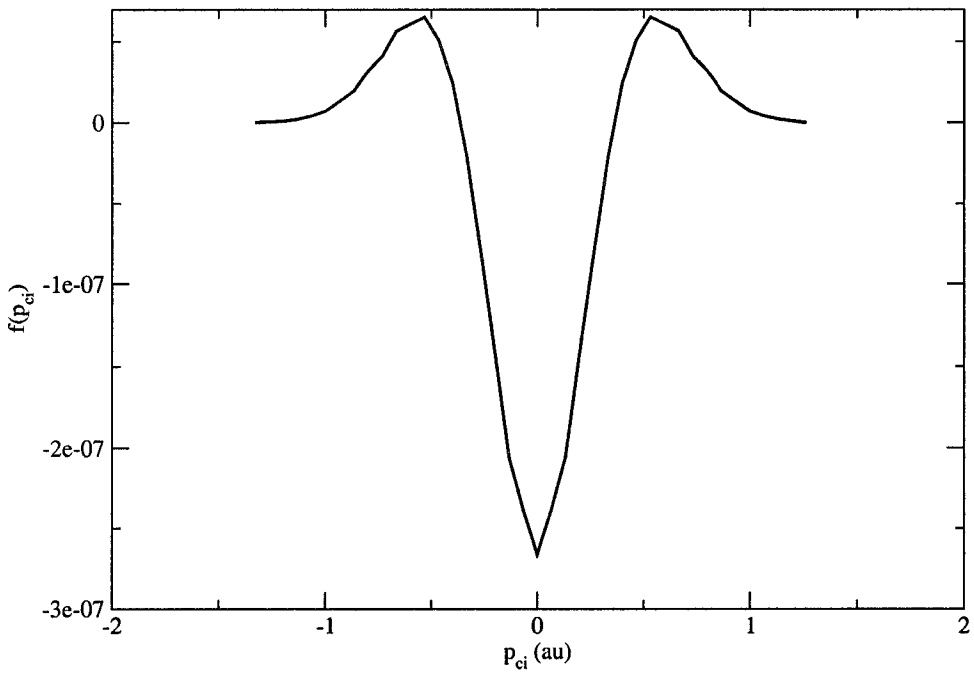

Figure 3.7: The centroid force as defined using the old QDO from Equation (2.13) has been weighted by the centroid density and integrated over the centroid positions and one of the centroid momentum of particle two. The dependence of on momentum is identical for both particle one and particle two.

The centroid momentum dependence of the centroid force, if using the *old* QDO, is drastically different than with the BECMD QDO. Though the magnitude of the centroid force is approximately the same for both definitions of the QDO, the shape is significantly different. As we can see in Figure 3.7, the integrated centroid force changes sign when the *old* QDO is used.

Another peculiar feature of the *old* QDO is that the imaginary component of the integrated weighted centroid force on particle 1 is non-zero as a function of the centroid position of particle 1 or particle 2. In all other cases, the imaginary component was effectively zero. There was a very small finite value for all the calculations. However

it was many orders of magnitude smaller and had no distinguishable shape. We therefore discarded it as numerical noise from the simulation. It is symmetrically non-zero with each particle; that is the imaginary component of the centroid force on particle 1 as a function of particle 1 is equal to minus the imaginary component of the centroid force on particle 1 as a function of particle 2,

$$\Im(f_{c1}(x_1)) = -\Im(f_{c1}(x_2)). \tag{3.11}$$

We can see this in Figure 3.8, and we note that it is symmetric in such a fashion that we will have cancellation of the imaginary momentum dependence that allows us to obtain the correct dynamics in certain circumstances.

3.3 Single Particle Correlation Functions

It was shown numerically in [6] that with the BECMD formalism we can correctly account for the quantum exchange effects of two bosonic particles confined in the potential described by Equation (3.1) using the parameters $c = 0.1$ and $g = 0.01$. Single particle real time double Kubo transformed position autocorrelation functions were calculated and compared to the double Kubo transform of exact quantum mechanical result obtained from a variational calculation. The autocorrelation function in question is given by

$$\langle \hat{x}\hat{x}(t)\rangle^D = \int\int \frac{dp_c dq_c}{(2\pi)^{3N}} \frac{\rho_c^B(p_c, q_c)}{Z} x_c x_c(t). \tag{3.12}$$

Here we will extend the results of that study by varying the cubic and quartic coefficients of the potential.

It should be noted that there was success using a QDO defined by Equation (2.13). However, it was only capable of correctly calculating correlation functions using certain coordinates. Specifically for the two particle system, centre of mass and relative distance coordinates were used and the results were as accurate as with the QDO defined by Equation (2.35). However, for single particle correlation functions it begins to deviate from the exact value shortly after the $t = 0$ value.

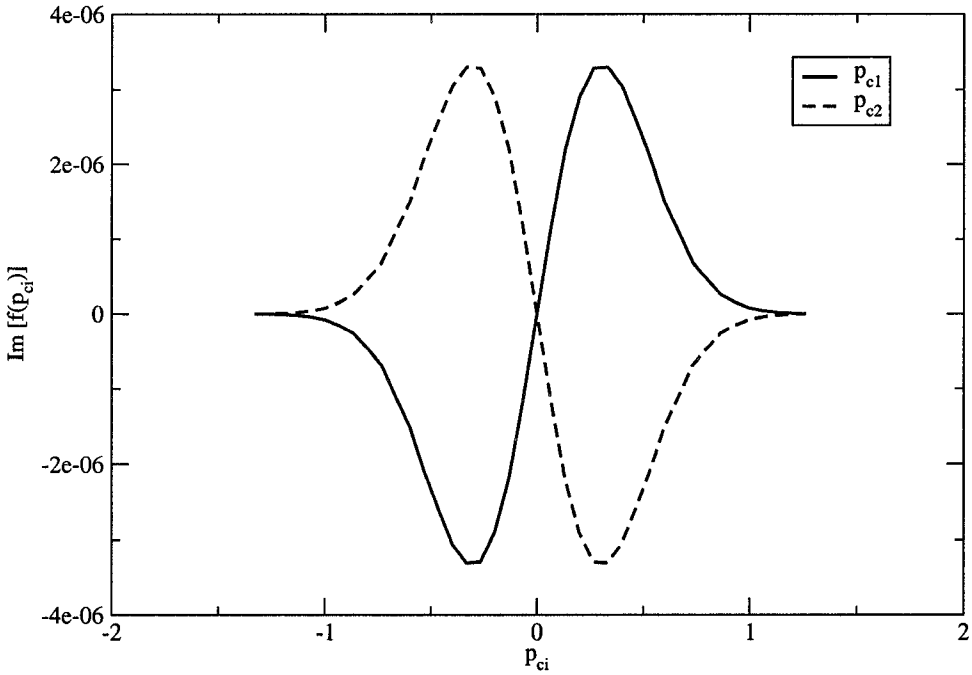

Figure 3.8: The imaginary component of the centroid force defined using the old QDO from Equation (2.13) has been integrated over the centroid positions and one of the centroid momentum. The solid line is for the dependence on the centroid momentum of particle one and the dashed line for particle two.

If we look at Equation (3.12), we can see that when we consider the centroid variables to be classical-like phase space variables. The expression is identical in form to the classical dynamics definition of the correlation function in Equation (1.10). We can then use classical MD techniques using the classical-like equations of motion defined in Equations (2.40) and (2.41) to calculate individual MD trajectories, which are then integrated over the initial conditions using a Markov chain MC calculation.

In Figure 3.9 we can see the centroid position autocorrelation function as a function of time. This calculation is taken from [6], and it shows that the BECMD result agrees very well with the exact calculation. Also it shows what the result would be if we were to use the QDO defined in Equation (2.13). We can see that this old definition yields a correlation function that begins to differ from the exact result significantly sooner than the BECMD result. As we expect, the $t = 0$ equilibrium result is exact, since at this point we are not making the CMD approximation. For a more indepth discussion of this calculation for these specific parameters the reader is referred to [6].

We would also like to see the correlation function for varying values of the parameters c and g. In Figures 3.10 - 3.16 we see that the CMD approximation becomes less effective as the anharmonicity is increased. In Figure 3.10 we have the same choice of parameters as in Figure 3.9. However, we have changed to a temperature corresponding to $\beta = 7$ and we see similar results to those in Figure 3.9.

In Figure 3.11 we have increased the anharmonicity in the quartic term of the potential and are using the parameters $c = 0.1$ and $g = 0.05$ at a temperature of $\beta = 7$. Although we still have good agreement between the BECMD result and the exact quantum mechanical result, as expected the increase in the parameter g has caused the BECMD result to degrade sooner. We can see this even further in Figure 3.12, where we have increase g to $g = 0.1$. At this point the BECMD result is only accurate for roughly one oscillation.

In Figure 3.13 we have set $c = 0$ and we are looking at the effect of adding only a quartic term to the potential with $g = 0.01$. We have now regained the symmetry of the oscillation about zero. This choice of parameters causes the potential to become *sharper* and we again see that the BECMD approximation is only valid for short time

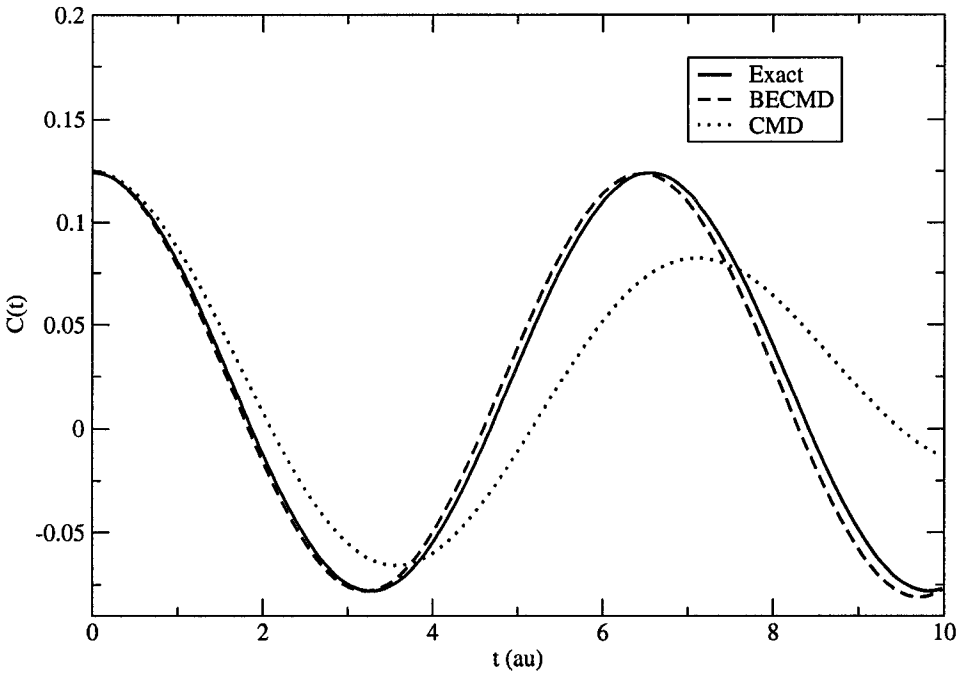

Figure 3.9: The double Kubo transformed correlation function for an individual particle. Results are for $\beta = 10$. The solid line corresponds to the exact result and the dashed line corresponds to the BECMD approximation. The dotted line is the result obtained when using the QDO defined in Equation (2.13). The calculation was carried out for 400000 MC steps and with a MD time step of 0.025 (au). The parameters used were $c = 0.1$ and $g = 0.01$.

periods. By making c non-zero and choosing the parameters $c = 0.01$ and $g = 0.01$, we can see we in Figure 3.14 the BECMD approximation holds extremely well. In Figure 3.15 we have again increased the cubic parameter and have $c = 0.05$ and $g = 0.01$, and we still see very good agreement between BECMD and the quantum result. Finally by increasing c again, choosing $c = 0.11$ and $g = 0.01$, we can see in Figure 3.16 that the BECMD result begins to differ more from the exact result. As we increase the anharmonicity of the system (increasing c), the pre-calculation of the centroid force and centroid density becomes increasingly difficult.

In addition, there is the limiting case when we choose $c = 0$ and $g = 0$ and we return to a purely harmonic potential. We do not bother to calculate this result since it can be written analytically. In this specific system the BECMD approximation is exact, and this has been shown and derived in [12].

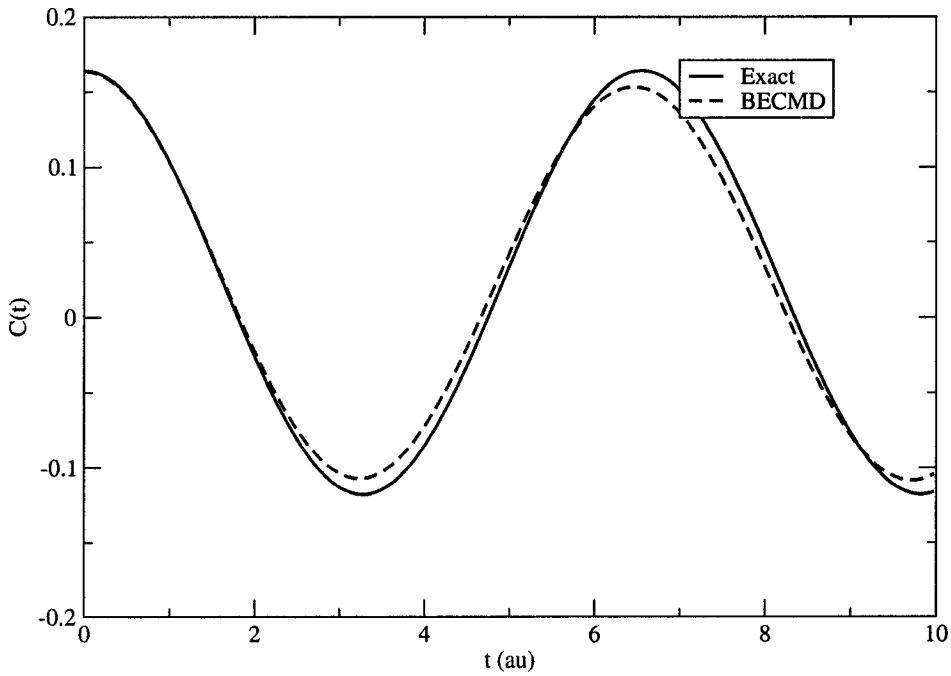

Figure 3.10: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.1$ and $g = 0.01$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

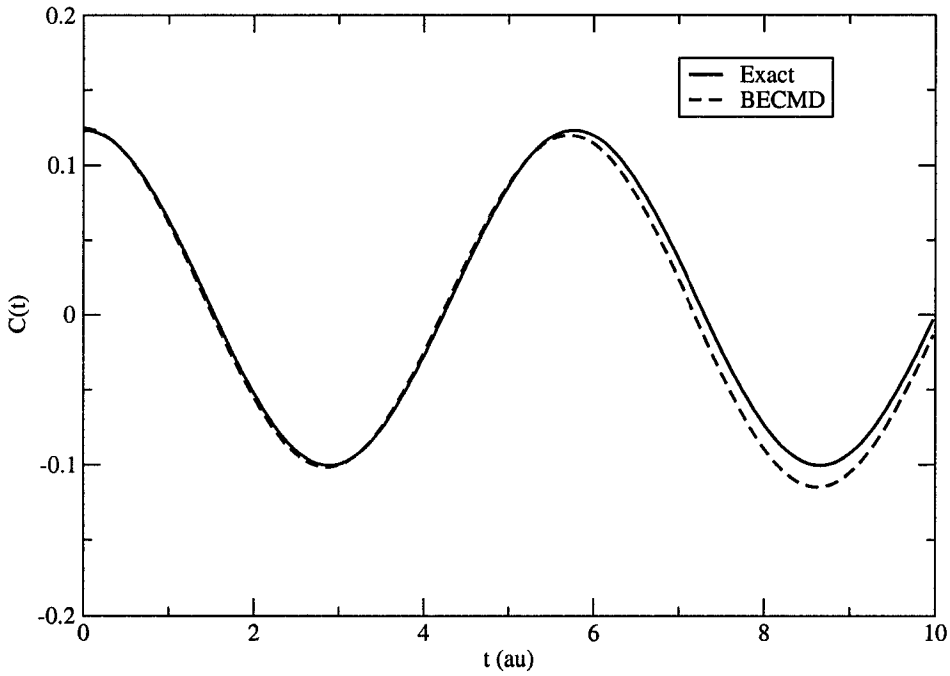

Figure 3.11: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.1$ and $g = 0.05$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

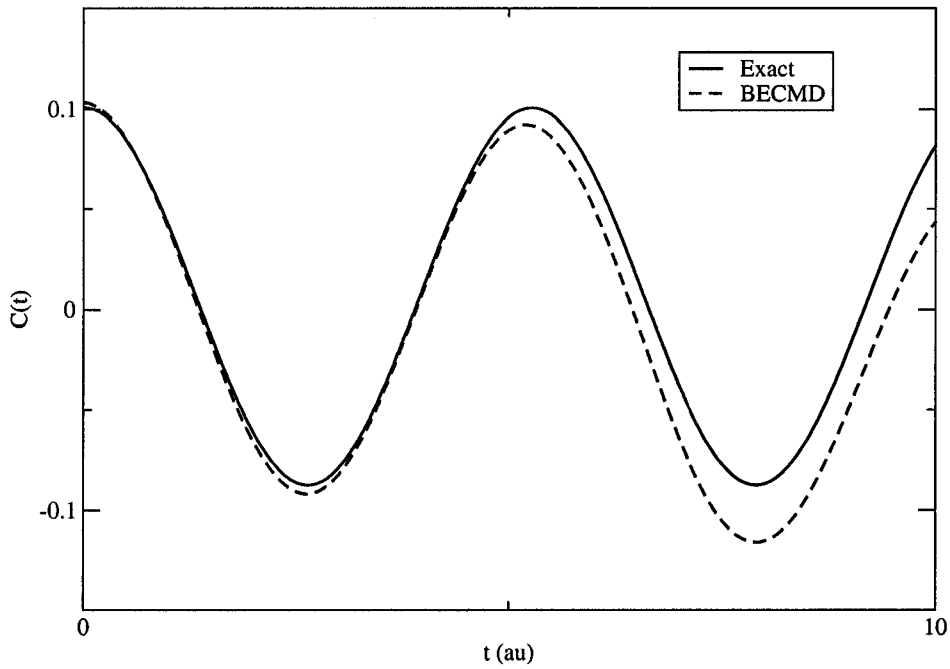

Figure 3.12: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.1$ and $g = 0.1$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

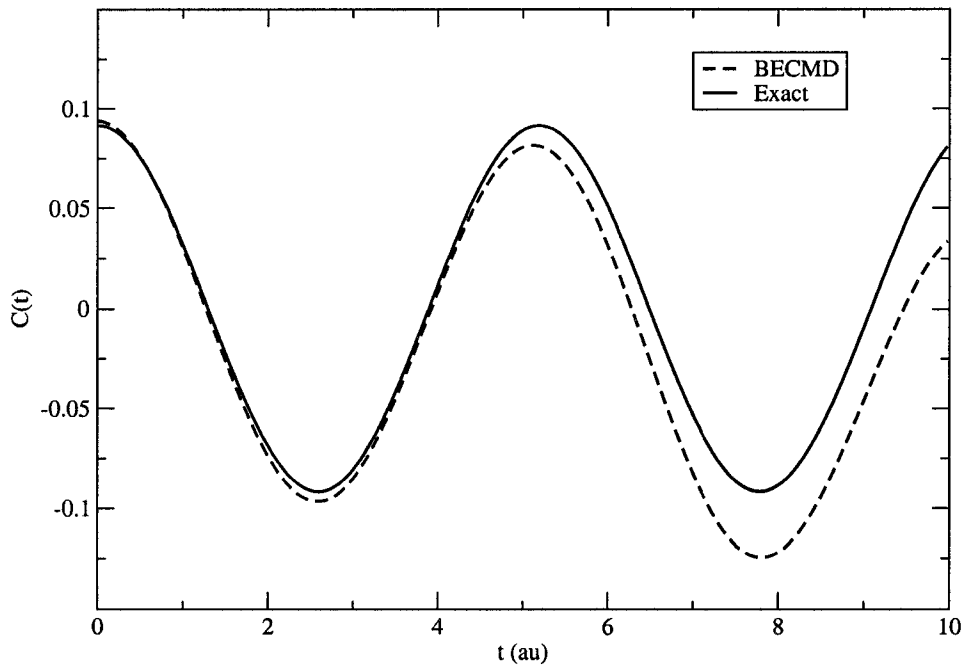

Figure 3.13: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0$ and $g = 0.01$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

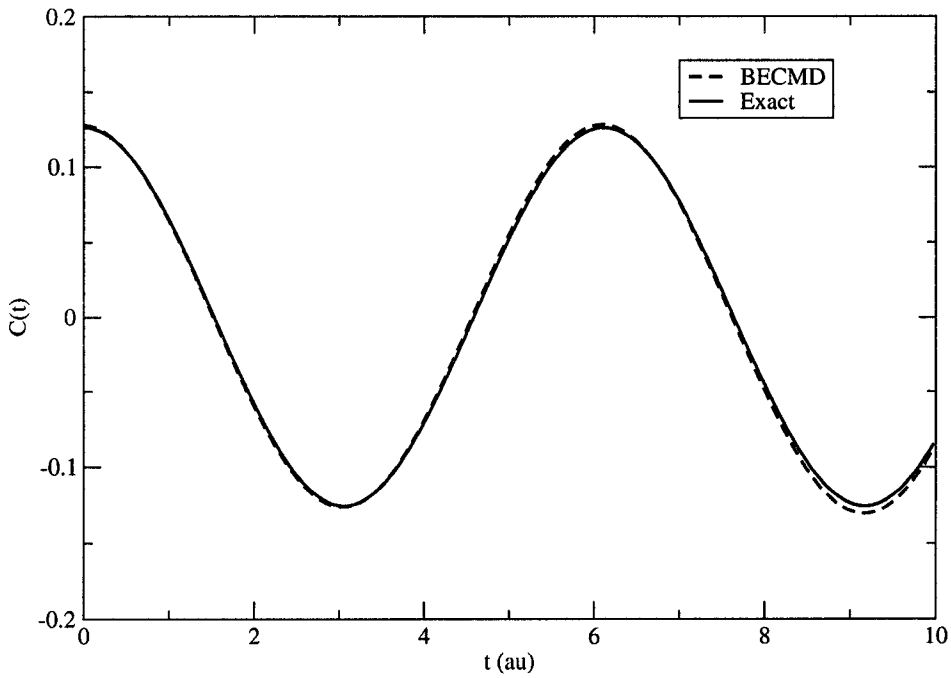

Figure 3.14: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.01$ and $g = 0.01$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

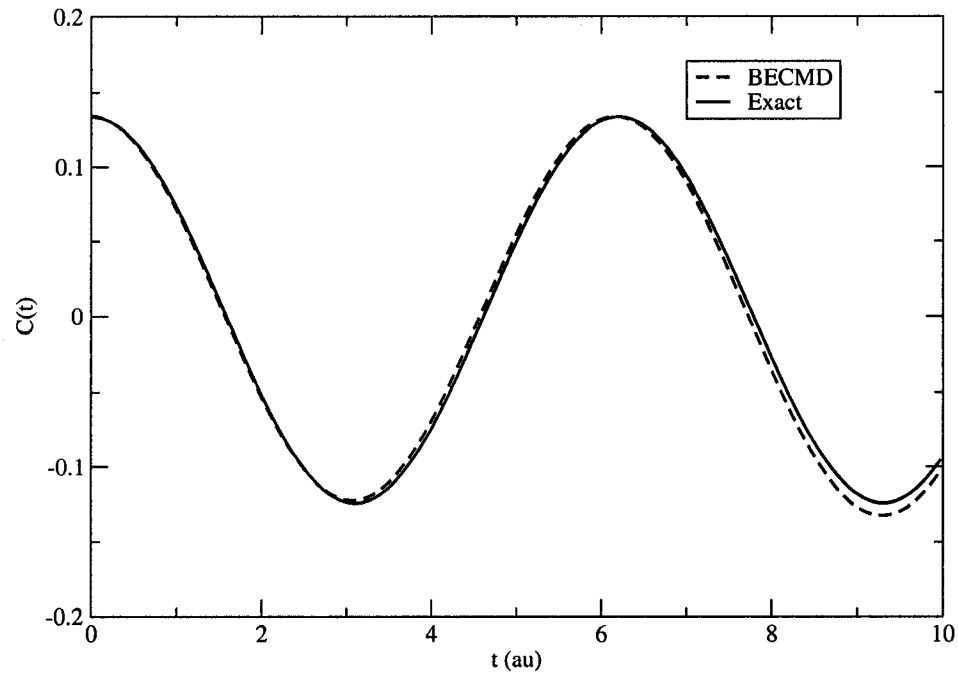

Figure 3.15: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.05$ and $g = 0.01$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

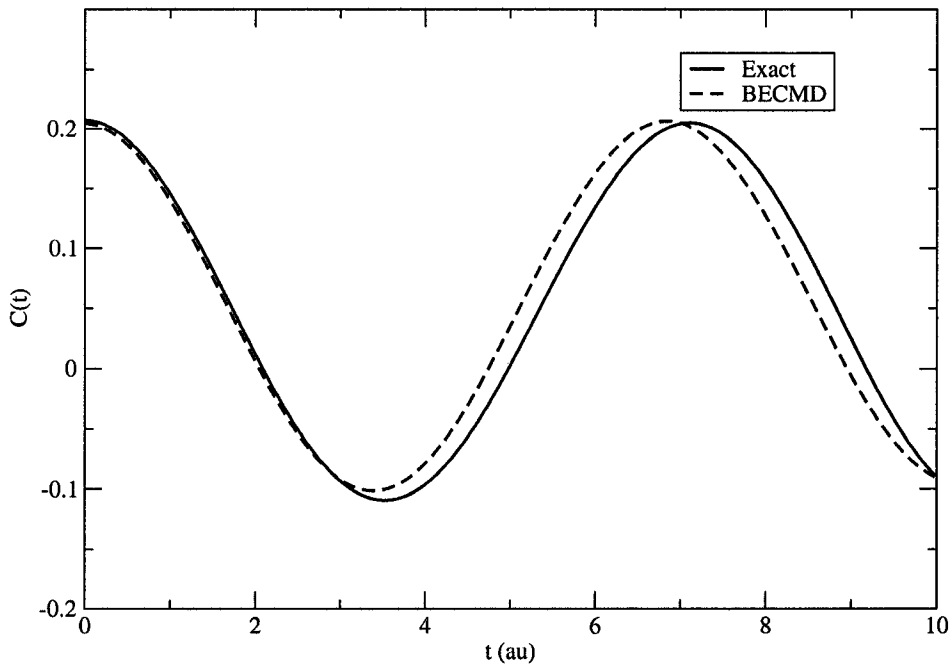

Figure 3.16: The double Kubo transformed correlation function for an individual particle for $\beta = 7$, $c = 0.13$ and $g = 0.01$. The solid line is the exact quantum mechanical result, and the dashed line is the BECMD approximation.

Chapter 4

Centroid Dynamics with Anisotropy

To be able to perform a BECMD simulation there are many steps that need be looked at in detail. We have already discussed the theory in Chapter 2 and now it is time to see how the pieces fit together, while keeping in mind that we would like to eventually extend the simulation to a many body system. The algorithm we intend to use can be broken down and described nicely in pictorial form and is shown in Figure 4.1.

We wish to calculate the correlation function similar to Equation (2.44). We attempt this by using a PIMC simulation to sample the initial conditions from the centroid density and this first initialization calculation is represented by the horizontal *squiggle-arrow.* For several thousand of those initial conditions we then perform a classical-like MD algorithm for each initial condition, which are represented in Figure 4.1 by the vertical *squiggle-arrows.* In addition to this, at each MD step the classical-like force is calculated from a PIMC calculation that has its centroid position constrained, i.e. we need to know the force on that particle when it is at that point only.

4.1 Anisotropic Potential

To test CMD in an anisotropic potential, we choose the system of a single helium atom in the presence of an N_2O molecule. For our simulation, the N_2O molecule

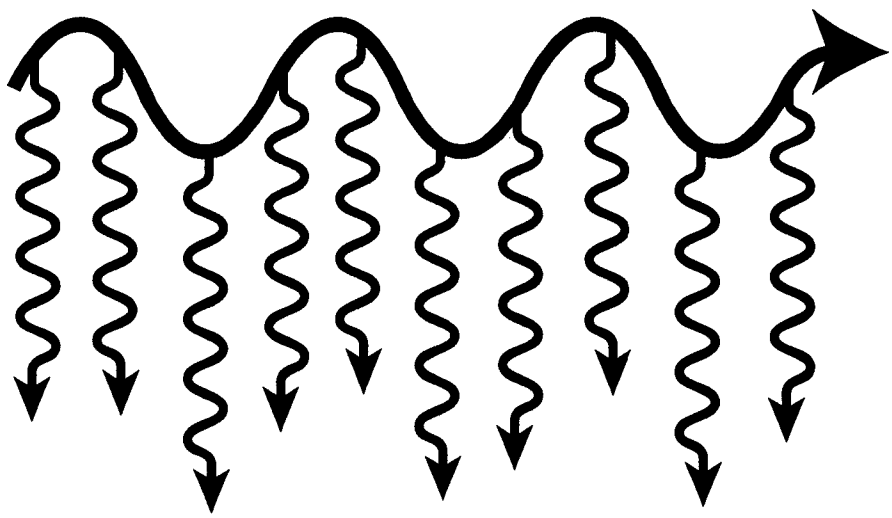

Figure 4.1: A simple schematic of the algorithm we intend to use. The horizontal *squiggle* arrow represents the sampling of the initial conditions which is done by a long MC calculation. The vertical *squiggle* arrows represent a MD trajectory, for which we use a constrained PIMC simulation to calculate the force. After these trajectories have been run, we average them and we have our correlation function. In principle we need thousands of these MD trajectories to accurately calculate a correlation function.

will be rigid, and pinned.[1] Molecules such as N_2O and their interactions with helium clusters, which range from the size of a few atoms to thousands of atoms, have been studied in great detail both experimentally and numerically [33, 34, 35].

We define a Cartesian coordinate system for the N_2O potential, such that the $z - axis$ is along the length of the molecule with the origin centered at the centre of mass of the molecule. The potential features azimuthal symmetry around the $z-axis$, and though we could represent the potential in cylindrical coordinates, we would gain no advantage from a computational viewpoint. We do, on some occasions, express positions in spherical coordinates. This is done only to help interpret results.

There is a global minimum of the potential in a torus around the N_2O molecule, and in addition there is a local minimum at the oxygen end of the molecule and a shallow local minimum at the nitrogen end. At a temperature of $1K$ we do not expect the helium atom to localize in the local minimum at the end of the molecule. At this temperature, we expect it to remain in the torus shaped global minimum around the molecule.

We choose this potential with the foresight that we will eventually want to include more than one helium atom, which will require, in addition to the BECMD formalism, a method of correctly account for quantum exchange. In DT-PIMC this is possible, but has not yet been done with a constraint on the centroid position. In F-PIMC there has been progress regarding the inclusion of exchange effects [30], although it has not been used to the extent that DT-PIMC has.

4.2 Centroid Density and Centroid Force

The two most important objects in the simulations we will be working with, as in Chapter 3, are the centroid density and the centroid force. As we know, from a theoretical viewpoint, the centroid density will lead to the partition function, which gives us a plethora of information. However the centroid density is also used extensively in our simulation, and it will be used as the sampling distribution of our initial

[1] We mean that the molecule will be fixed in place and has no degrees of freedom

conditions for the MD calculations. It can be shown that the centroid momentum dependence of the centroid density is Gaussian (for three dimensions we will have three Gaussian functions) and can be factored out yielding

$$\rho_c(p_c, q_c) = e^{-\frac{\beta}{2m}p_c^2}\rho_c(q_c), \tag{4.1}$$

where $q_c = (x_c, y_c, z_c)$ is a standard Cartesian vector defining the centroid position of the helium atom, and p_c is defined similarly. This allows us to generate the initial centroid momentum conditions by sampling Gaussian distributed numbers. The result of this, for a large number of initial conditions, is shown in Figure 4.2.

The position dependence of the centroid density does not have a convenient analytic expression like the centroid momentum distribution. To obtain this distribution, we run a long PIMC calculation and record the positions q_c. It is difficult to interpret the x_c and y_c dependence of the centroid density since there is azimuthal symmetry and it is therefore highly degenerate in the $x - y$ plane. However, the x_c and y_c dependences of the centroid force are shown in Figures 4.3 and 4.4. We can see from these plots that there is a peak in the distributions near x_c (or y_c) $\approx \pm 3$Å. These peaks correspond to the global minimum of the N_2O centroid potential.

The centroid density dependence on z_c (the z-axis is along the length of the N_2O molecule) in Figure 4.5 has its peak also corresponding to the global minimum of the centroid potential, which is to the right of zero. At higher temperatures, we could expect to see a peak in all three centroid position density profiles near the local minimum. However, at a temperature of $1K$ we do not see this feature.

The calculation for the sampling of the initial conditions was for one million MC steps with $P = 50$ discretizations of the Feynman path. The initial conditions were sampled at every 100th MC step and the total computational time for this initialization step of the procedure was approximately 13 hours.

At the end of the day, we wish to do an MD calculation, which means we need to integrate a set of equations of motion. In the BECMD formalism, we have a set of classical-like equations of motion given by Equations (2.38) and (2.37). Regardless of what numerical integration technique you choose to use, whether it be the Euler

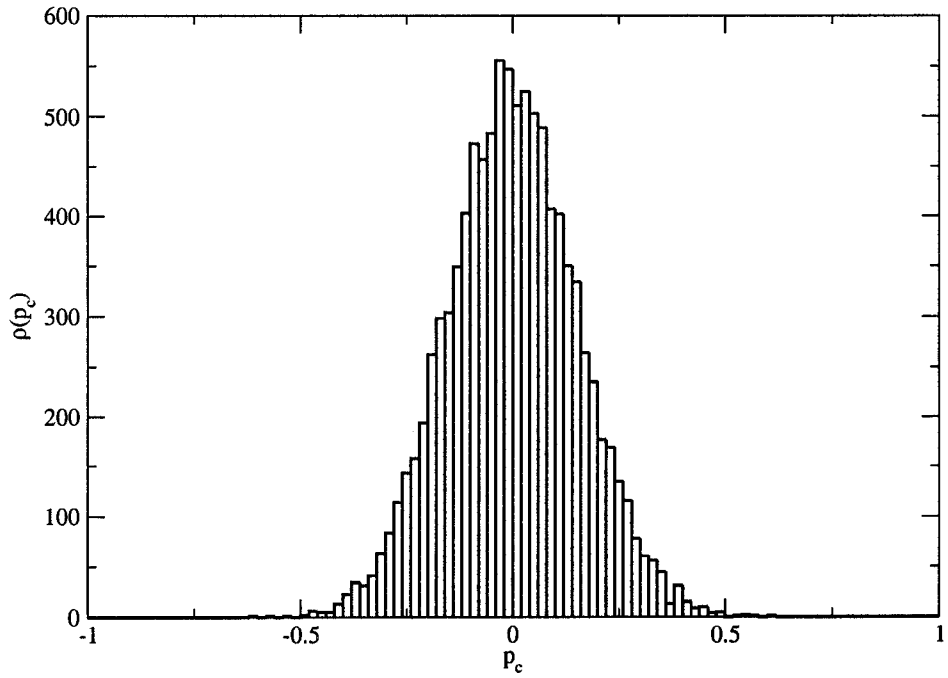

Figure 4.2: Histogram representation of the Gaussian distribution of centroid momenta. The distribution was sampled using standard Gaussian distributed random numbers with the correct mean and Gaussian width.

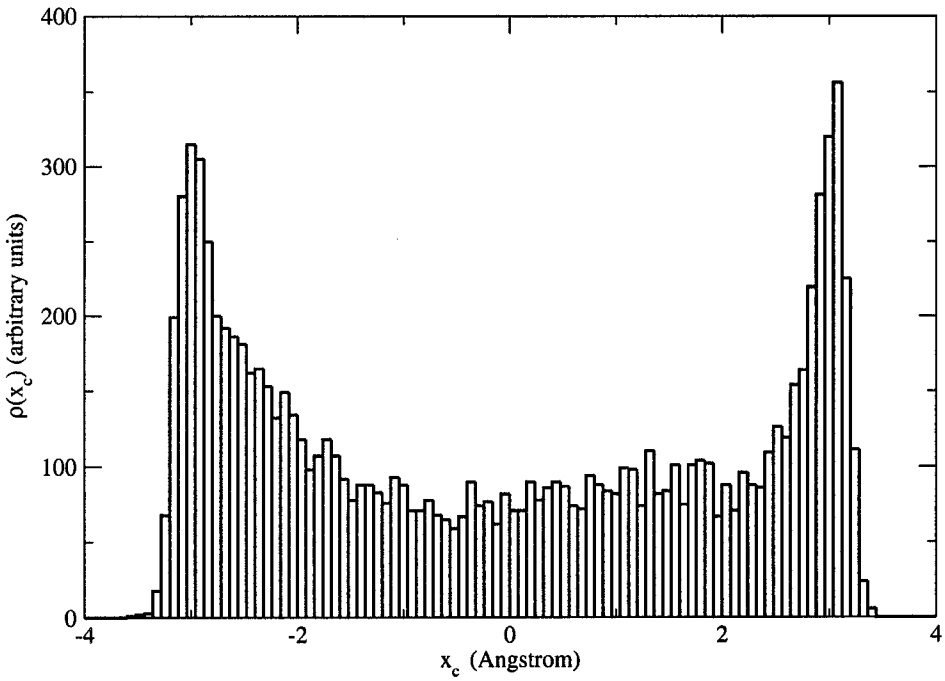

Figure 4.3: Histogram representation of the centroid position density for the x_c coordinate. The two peaks to the left and the right of the zero are the locations of the global minimum in this $y - z$ plane.

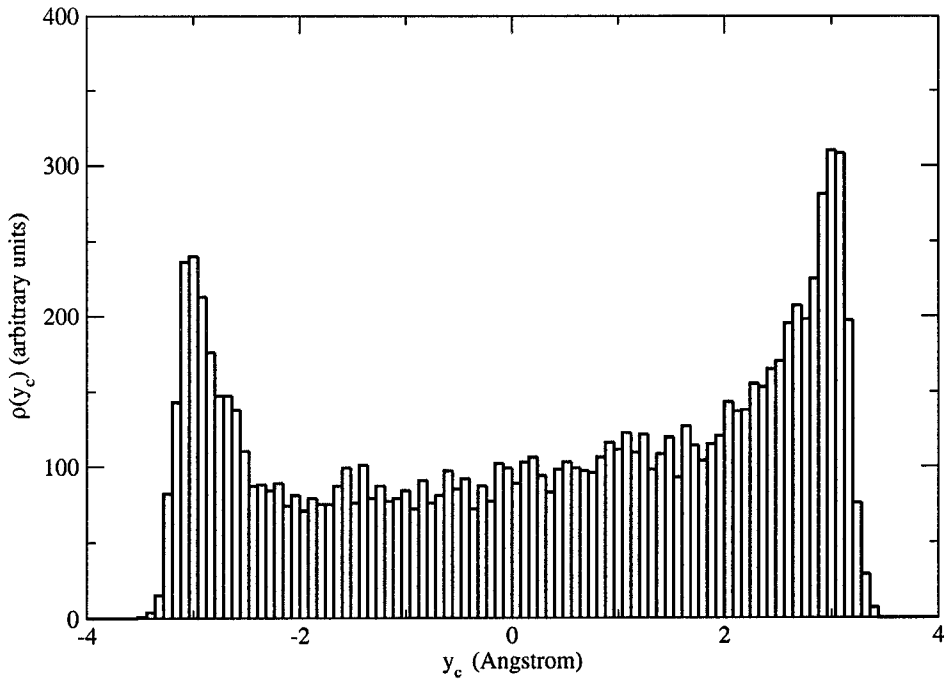

Figure 4.4: Histogram representation of the centroid position density for the y_c coordinate. The two peaks to the left and the right of the zero are the locations of the global minimum in this $x - z$ plane.

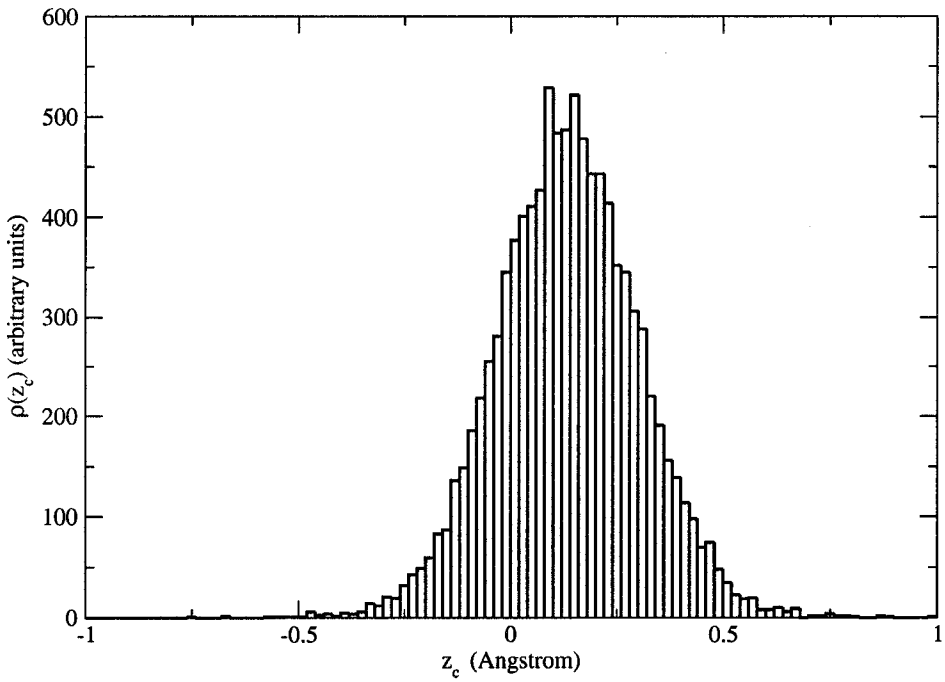

Figure 4.5: Histogram representation of the centroid position density for the z_c coordinate. The peak of this density is located to the right of zero and corresponds to the global minimum.

algorithm, a velocity Verlet algorithm [14], or a Runge-Kutta algorithm [36], the integral[2] step is the calculation of the centroid force.

As was stated earlier, one of the main differences between this calculation and the proof of principle calculation of the unphysical anharmonic oscillator is that we do not pre-calculate the centroid force prior to the CMD calculation. Instead we calculate it *on the fly* as we move through the CMD simulation. This means that every time the integrator algorithm calls for the centroid force, we must calculate the centroid force for that configuration at that point in time.

The calculation of the centroid force is done via a PIMC method. For our purposes, we wish to constrain the centroid position and average the force at that point. This means we need to use the constraint methods proposed in Chapter 2. Since it is an approximation, we would first like to test the constrained DT-PIMC method. In DT-PIMC we used a Gaussian function to approximate a delta function which constrains the centroid position. We can alter the *strength* of this approximate delta function by varying the parameter α from Equation (2.65)

In Figure 4.6 we have a calculation of the average radial position, r_{avg}, for a He atom in the presence of an N_2O rigid rotor that is pinned[3] in place using different values for α. We are setting the constraint so that the particle's centroid position is placed at (in Cartesian coordinates) $q = (3.704, 3.704, 1.058)$Å, which corresponds to a radial magnitude of 5.34443Å. The initial starting configuration of the path is a random distribution around the position of the constraint. The initial centroid position value is not equal to the constraint value. As we see with a value[4] of $\alpha = 5 \times 10^{48}$ ($\alpha \approx \infty$ for computational purposes) after a few MC steps, the r_{avg} value is identical to the constrained value. This would be the ideal situation, but unfortunately this is too large a value of α. If we look at Figure 4.7 we can see how the constraint value affects the average squared radial position. In the limiting case of $\alpha = 5 \times 10^{48}$, we can see that the r^2_{avg} has been constrained to approximately the value of $(r_{avg})^2$. This

[2]A pun, since we will be integrating to calculate the force.

[3]The term pinned means that we do not allow the N_2O molecule to rotate.

[4]The value $\alpha = 5 \times 10^{48}$ has no significance, it was chose arbitrarily to test the upper limit when the Gaussian appears like a delta function.

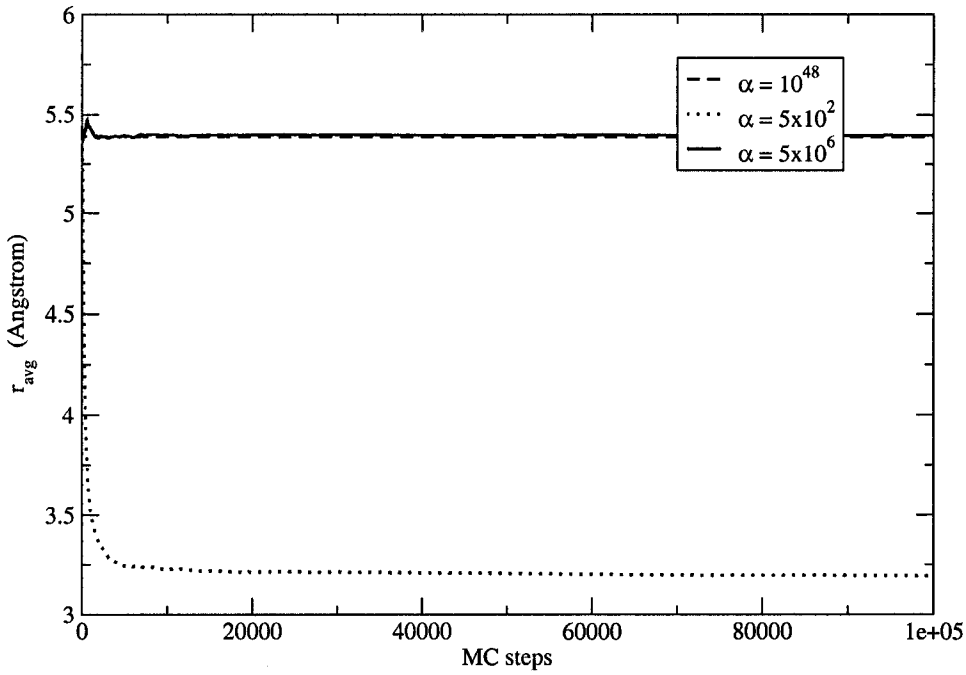

Figure 4.6: Convergence of the short time average constrained radial position, r_{avg}, for one Helium atom in proximity to an N_2O molecule as a function of the MC steps. The simulation was run with a discretization of 20 time slices for the Feynman path, at a temperature of $1K$. The calculation was carried out for various different values of α, which is a measure of the strength of constraint. Here we show how the optimal value of α compares to the extreme values.

means that we are not only constraining the centroid position, but we are also forcing all the time slices to the centroid position.

In addition, in Table 4.1 we can see the short time averaged values of the centroid positions, as well as how the constraint value α changes the acceptance rate, which we aim to keep in the range of 0.3 to 0.5. However, the acceptance rate does not change the simulation result, it only changes the length of time it takes to reach a converged value. It should be kept in mind that the choice of the value for the parameter α is dependent on the temperature and number of discretizations of the path and should

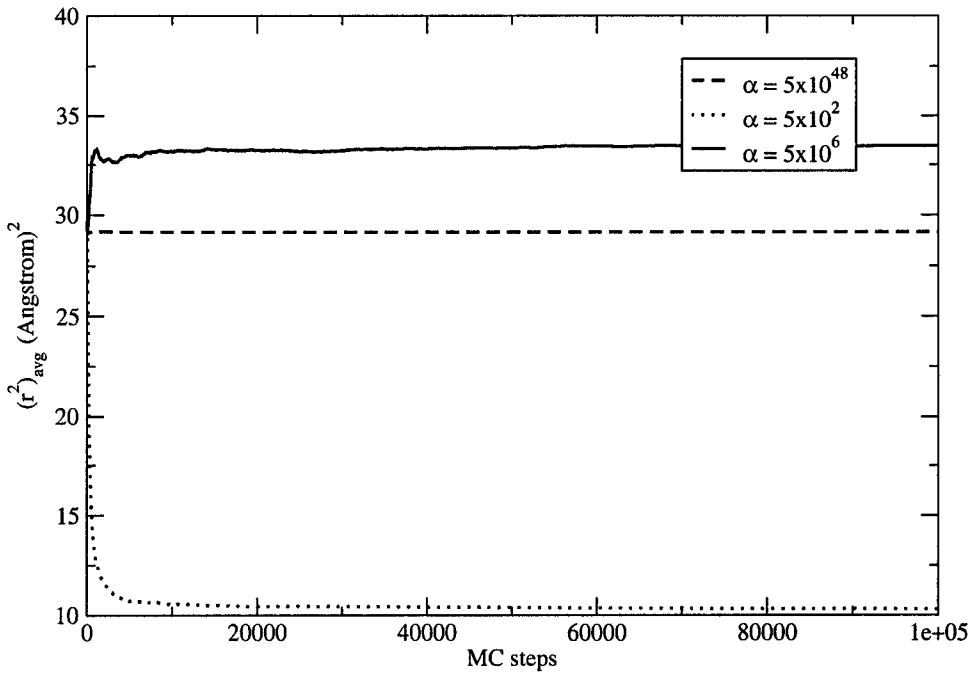

Figure 4.7: Convergence of the short time average squared constrained radial position, $(r^2)_{avg}$, for one Helium atom in proximity to an N_2O molecule as a function of the MC steps. The simulation was run with a discretization of 20 time slices for the Feynman path, at a temperature of $1K$. The calculation was carried out for various different values of α, which is a measure of the strength of constraint. Here we show how the optimal value of α compares to the extreme values.

Table 4.1: Effects of the constraint value α can be seen on the average radial position and average squared radial position. As well, the effect of the constraint value on the acceptance rate of the calculation is shown. The final entry in this table corresponds to a F-PIMC simulation where the constraint is considered exact.

α	Acceptance Rate	r_{avg} Å	$(r^2)_{avg}$ Å^2	MC method
0	0.40707	3.19642	10.30547	DT-PIMC
5×10^2	0.40598	3.19453	10.29209	DT-PIMC
5×10^3	0.41006	3.19869	10.32009	DT-PIMC
5×10^4	0.41717	3.21088	10.40541	DT-PIMC
5×10^5	0.58360	3.80748	15.61761	DT-PIMC
5×10^6	0.59972	5.39756	33.45641	DT-PIMC
5×10^7	0.30114	5.46852	34.39759	DT-PIMC
5×10^8	0.03725	5.44280	31.16018	DT-PIMC
5×10^{48}	0.00013	5.38840	29.16049	DT-PIMC
N/A	0.70060	5.48519	34.58493	F-PIMC

be chosen with the criterion $\alpha > \frac{P}{4\lambda}$.

Overall, the inclusion of a constraint in DT-PIMC is possible with this method and, as a future prospective, the constrained DT-PIMC method will allow for the inclusion of quantum exchange effects in a many body system. However, the addition of the constraint is an approximation, and it severely slows the DT-PIMC calculation. The algorithm used in this project could of course be improved, but the time scale improvements would not be large enough to make a significant difference.

To calculate the centroid force we must use a constrained PIMC method. As was discussed in Sections 2.4.2 and 2.5.2, F-PIMC lends itself naturally to the language of a constrained centroid position, while for DT-PIMC methods we must make an additional approximation by means of a sharply peaked Gaussian function added to the path integral representation of the centroid density. The constrained centroid force given by the F-PIMC method can be considered exact, and we can use it to test how well the constrained DT-PIMC method agrees.[5] Due to the numerical cost of the

[5]As stated earlier, DT-PIMC will be most useful for a simulation with multiple particles since particle permutation using DT-PIMC methods is a well studied algorithm.

constrained DT-PIMC method, it will be more beneficial for us to use the F-PIMC method for our calculations.

Before we move on with the simulation, we need to know that the centroid force we calculate is accurate.

The analytical operator expression for the centroid force (in one dimension, the three dimensional case is a trivial extension) is

$$F_c(p_c, q_c) = Tr\left[\hat{\tilde{\delta}}_c(p_c, q_c)\frac{\partial}{\partial \hat{q}}\hat{V}\right]. \tag{4.2}$$

However, in the simulation we use the path integral representation for convenience, so that here the centroid force can be written as

$$F_c(p_c, q_c) = -\frac{1}{P}\sum_{i=0}^{P}\frac{\partial}{\partial q_i}V(q_i), \tag{4.3}$$

where q_i corresponds to the position of the particle at the $i-th$ time slice.

Since the potential, and as such the centroid force, are dependent only on the position, we know that the convergence will be *quick* compared to the convergence of the total energy, which is dependent on the momentum. In fact, the convergence of the force has been found to have the same timescale as the convergence of the potential, which is not surprising, since the estimator used to find the force is simply a two-point finite difference method:

$$F(q_i) = \frac{V(q_i + h) - V(q_i - h)}{2h}, \tag{4.4}$$

where h is a parameter of the simulation. To find an acceptable value of h, we can calculate the force for a range of values of h and choose a value at which the calculated value of the force doesn't change significantly. We do have numerical restrictions, so that we cannot choose a value of h that is too small or the subtraction in Equation (4.4) will be beyond the precision of our computers. In Figure 4.8 we can see the value of the centroid force as a function of h. We expect the centroid force to be correct in the limit of h going to zero. It may be difficult to see in Figure 4.8, however, that the value of the centroid force plateaus for small h. Therefore, we choose a value of $h = 0.01$ Bohr,[6] in the plateau region, for all our simulations.

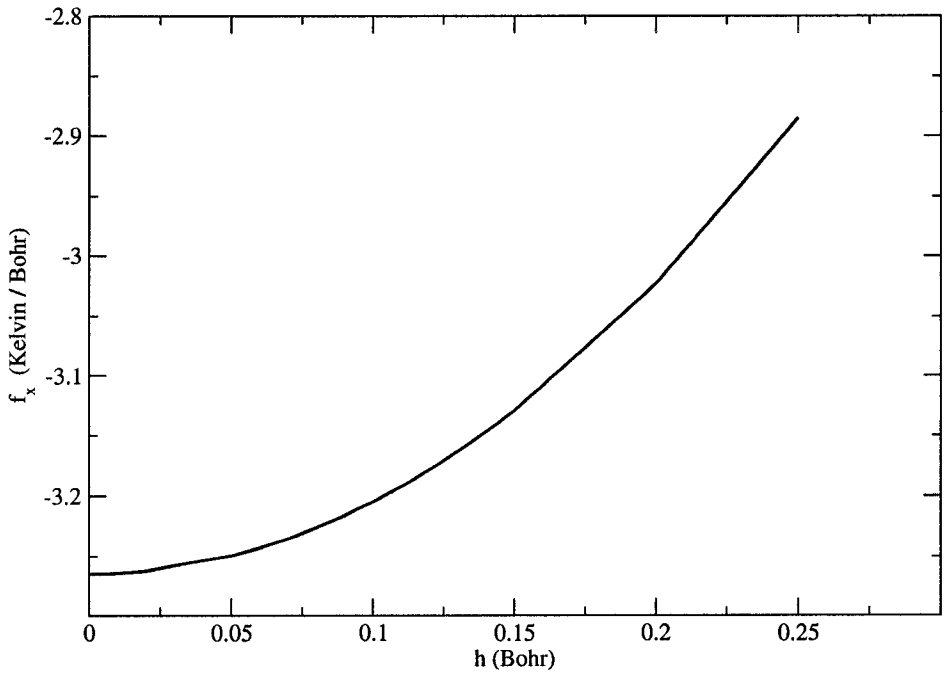

Figure 4.8: The average value of the centroid force on the helium atom along the x direction as a function of the finite difference size used in the force estimator from Equation (4.4).

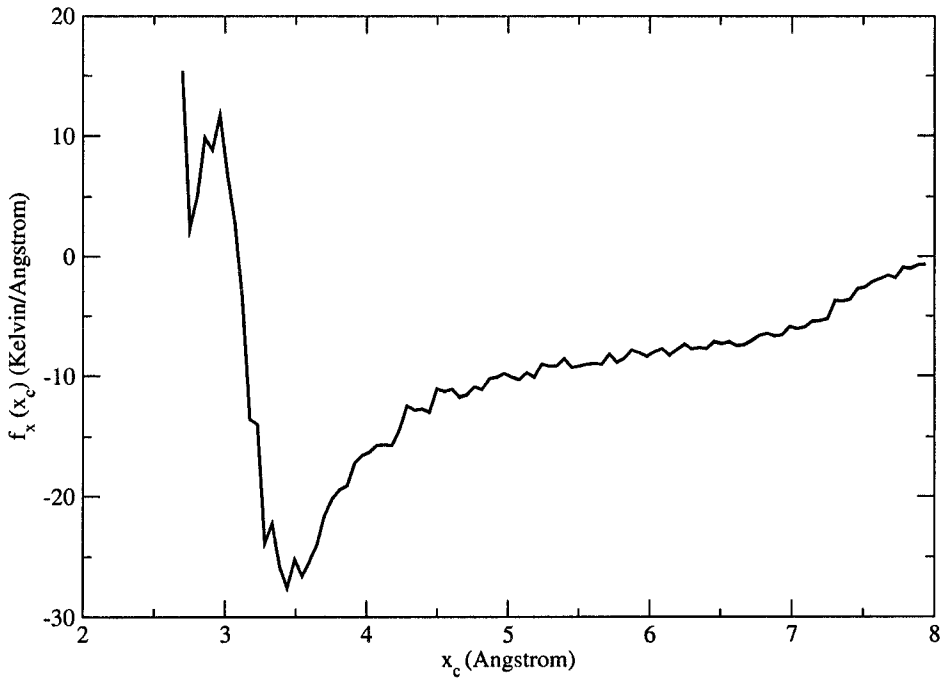

Figure 4.9: The centroid force on the helium atom along the x direction as a function of centroid position x_c. The y_c and z_c components were kept fixed at zero.

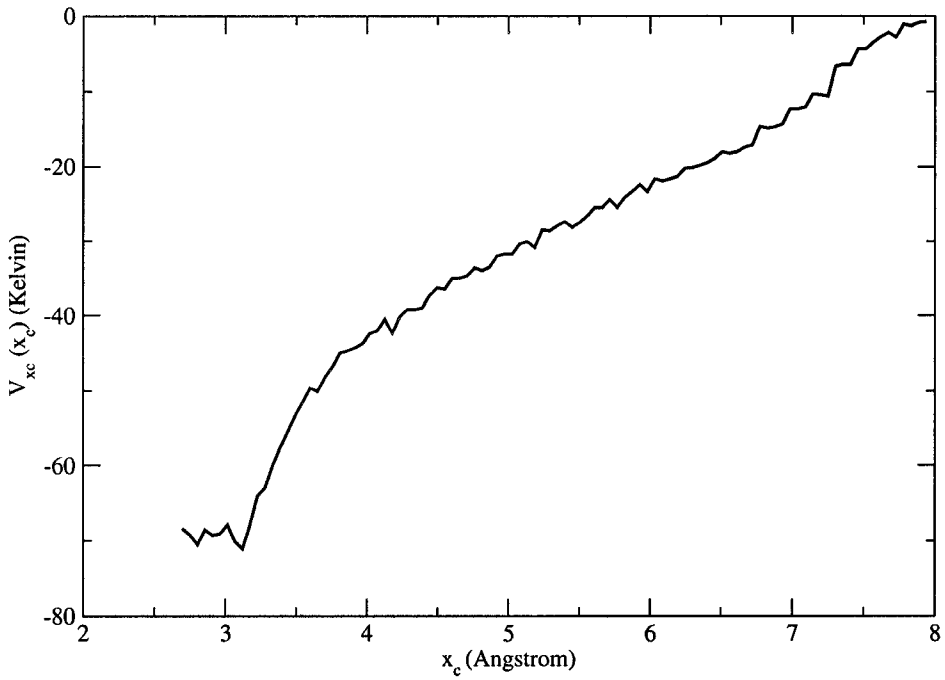

Figure 4.10: The centroid potential as a function of centroid position x_c. The y_c and z_c components were kept fixed at zero.

In Figure 4.9 we can see a profile of the centroid force, as a function of the centroid position. In this calculation, the Feynman path was discretized into 20 time-slices, which means that there is inherently (as always in a PIMC calculation) a systematic error in the calculation. The y_c and z_c values were held constant at $y_c = 0$ and $z_c = 0$ during this calculation, which allows us to see the centroid force as a function of x_c, which was incremented by a step size of $\Delta x_c = 0.1$. This cross-sectional view of the centroid force has the expected shape when we look at the centroid potential in Figure 4.10. The noisiness of the centroid force and the centroid potential is an artifact of both the step size Δx_c and the MC convergence of the estimators for the centroid force and centroid potential. In all PIMC calculations we have two inherent statistical errors, the first being the MC *time* which is the number of MC steps in the simulation. Ideally we would like to have an infinite number of MC steps to statistically reach a completely converged and exactly correct answer. The statistical error in an MC calculation is approximately $1/\sqrt{N}$, where N is the number of MC steps. Unfortunately, the number of MC steps is proportional to the computational time for the calculation. The second inherent error comes from the discretization of the Feynman path, which translates in our simulation to the number of Fourier components we keep. Analytically the path needs to be discretized into an infinite number of time slices, which corresponds to an infinite Fourier expansion. The number of time slices of path also affects the computational time, and how it affects the computational time is algorithm dependent. For our simulation the computational time is proportional to the number of time slices. We would then like to minimize the computational time while maximizing accuracy.

In our simulation, the most important estimator from the PIMC calculation is the centroid force. We would like to know the minimum number of discretizations of the Feynman path that are required to reach an average that is close to the infinite discretization limit. In Figure 4.11 we see the centroid force as a function of the discretization for a converged F-PIMC calculation. As the number of discretizations

[6] I apologize for switching units here, but the algorithm was easier to write in atomic units, while the results are better represented in Kelvin and Angstrom units.

Table 4.2: The acceptance rate, which we would like to maintain within the range from 0.3 to 0.5, for different values of M and the associated step size Δ.

Acceptance Rate	M	Δ
0.4787	10	1.5
0.46858	20	3.0
0.62968	30	3.0
0.46694	30	4.5
0.5247	50	7.0
0.41228	100	15.0

increases, the average of the centroid force along the x direction begins to converge to a certain value. For our simulations, we chose $P = 50$, since at this value we are very close to the converged value of the force.

To help increase the convergence efficiency, we have often talked about keeping track of the acceptance rate, which is the ratio of the number of individual MC moves that were accepted to the number of MC moves in total. We strive to keep the acceptance rate within the range from 0.3 to 0.5. The acceptance can be adjusted for an individual calculation by modifying the length of the move. In the algorithm this corresponds to adjusting Δ in Equation (2.63) for DT-PIMC or in Equation (2.60) for F-PIMC. In DT-PIMC Δ can be chosen to be approximately the thermal wavelength and the acceptance rate will generally fall in a desirable range. For F-PIMC we simply must test what values of Δ give a decent acceptance rate. For both DT-PIMC and F-PIMC, a very small Δ will mean that the configuration will not have changed very much, and most likely the move will be accepted, thus giving an acceptance rate that is too high. For a very large Δ, the new configuration will be significantly different, and will most likely be an unfavourable configuration, which means most of the moves will be rejected, leading to a small acceptance rate. Table 4.2 shows for a F-PIMC calculation of a single helium atom in the presence of a pinned N_20 molecule the desirable acceptance rates for differing values of Δ and M.

The overall efficiency of the calculation of the centroid force can be optimized by consideration of the algorithm. There are two major *bottle necks* in our F-PIMC

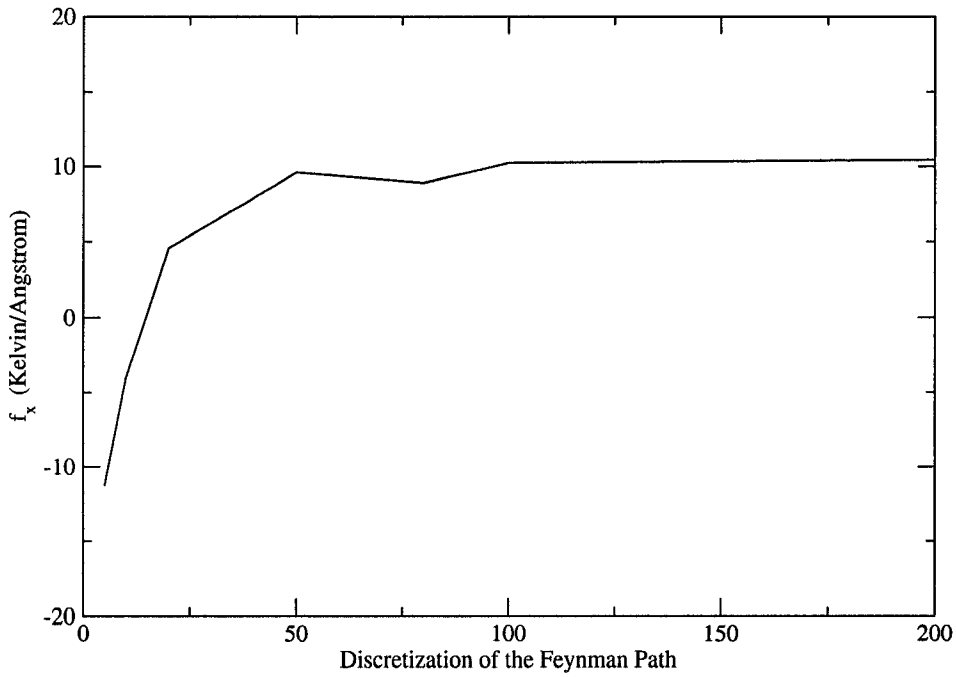

Figure 4.11: The F-PIMC converged centroid force along the x direction for a single helium atom in the presence of a pinned rigid N_2O molecule at $1K$ as a function of M, the number of components of the Fourier expansion of the Feynman path that were used in the simulation. For these simulations M is always equal to th P, the number of discretizations, or the number of time slices of the Feynman Path. This calculation was run for 400000 MC steps, and the acceptance ratio was kept between 0.3 and 0.5.

algorithm. The first is the actual Metropolis MC rejection algorithm, for which we need to calculate the potential for the entire path twice for each MC step and we must perform two numerical Fourier transforms. Unfortunately, there is not much we can do for this. The second *bottle neck* is the averaging of quantities. Naively, we would update the average of some quantity at each step. However, as described in [14], there is correlation in the averages. So, instead of calculating the average of a quantity at each MC step, we can calculate the average after every 10th MC step or every 100th step. More specifically, we want to only calculate the average once it is uncorrelated with the last calculation of the average.

Finally, we have one more trick to help make the calculation faster and more efficient. For an individual trajectory we will now spend most of the time calculating the centroid force using F-PIMC, although the CMD/BECMD formalism does not require the notion of a path integral obviously we see that the calculation does. In addition to the initial centroid conditions, we must also supply initial conditions for the individual beads of the Feynman path. In other words, we need an initial *shape* for the path each time we use the F-PIMC algorithm. If we start out in an unfavourable *shape*, then the beginning of the calculation will be spent *reshaping* the path, whereas if we started in a favourable *shape* then there would only be minor adjustments to the *shape* of the path.

What we would like to do is always start in a favourable configuration. To do this, at the first MD step of each MD trajectory we use a long MC calculation which hopefully brings the path into a nice *shape*. For the rest of the consecutive MD steps, when we move the centroid position according to the previously calculated force, we keep the *shape* of the path from the last MC step of the previous MD step. This way, we can assume that the *shape* of the path is close to a favourable configuration, and we will only have to make minor adjustments. This then allows us to cut down the MC time of the calculation of the centroid force for all the MD steps after the first step. So for each MD trajectory, the first MD step is a long MC time *shaping* step, and the rest are *short run* average steps.

We also implement a similar technique when we begin the trajectory. During

the initial sampling of the centroid density we also stored the *shape* of the path corresponding to the initial condition. When we begin the MD trajectory, we use not only the centroid initial condition, but we start with this configuration of the path instead of starting from some random *shape*.

4.3 Centroid Dynamics

As a first test of this anisotropic potential, we would like to calculate the centroid position autocorrelation function for the z coordinate. We choose this correlation function as opposed to the x or the y autocorrelation function due to the symmetry of the system. The period of oscillation in the x and y directions are unfortunately longer than the time that the MD trajectories can be considered correct, at which point we will not know if the CMD approximation is failing or just the numerical integrator.

For each trajectory calculated, we used the constrained F-PIMC simulation to calculate the force at each step. The DT-PIMC would work as well. However, it is currently a more expensive calculation than the F-PIMC method and is only an approximation, whereas the F-PIMC can be considered exact.

The dynamics of each trajectory was calculated using a Verlet integrator [14]. This numerical integrator was chosen for the fact that it minimizes the number force calculations and returns the most accurate result. Specifically, the Verlet algorithm requires only one calculation of the force per time step and returns a position whose error is of the fourth order in the time step. Unfortunately the Verlet algorithm does not calculate the velocity as well as it calculates the position, since it relies on a finite difference method to calculate the velocity which is only accurate to the second order in the time step.

There is a modified Verlet algorithm, *Velocity Verlet*, which gives more accurate velocities. The compromise is that *Velocity Verlet* requires an additional calculation of the force for each time step. Since we are looking for the centroid position autocorrelation function, we are not overly concerned with the velocity and will choose,

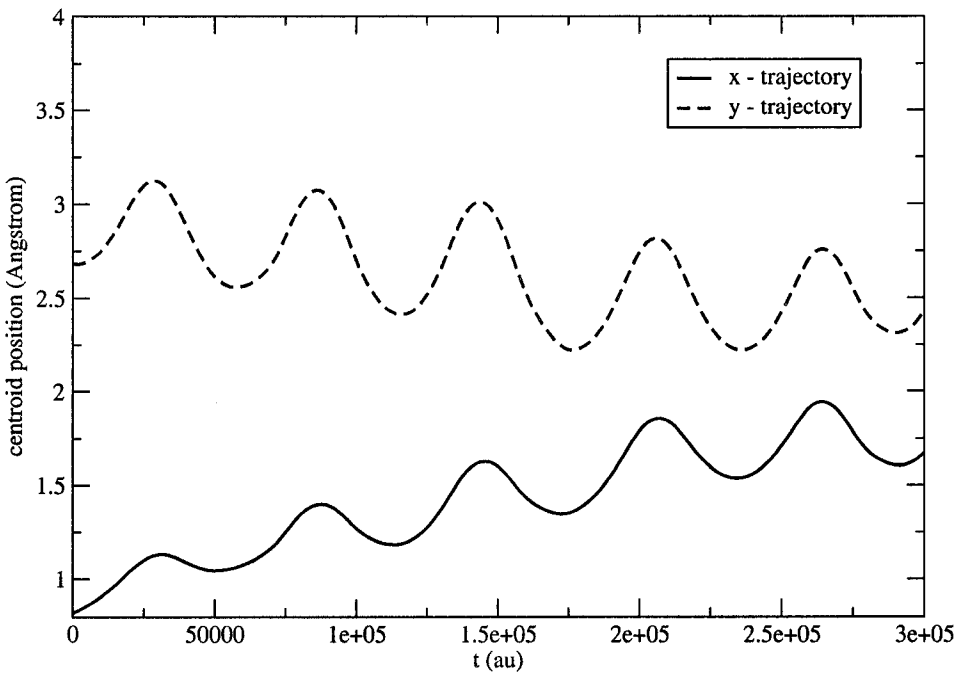

Figure 4.12: The x_c and y_c components of a typical trajectory. The solid line corresponds to the x_c component, and the dashed line to the y_c component.

in favour of time constraints, the regular Verlet algorithm. We chose the Verlet algorithm as opposed to the Euler algorithm purely because the Euler integrator generally gives poor results, and we chose it over a Runge-Kutta method due to the fact that Runge-Kutta methods require multiple calculations of the force.

We can see some of the typical dynamics by looking at an individual trajectory of a helium atom. In Figure 4.12 we have the trajectory of a single helium atom in the x and y directions. We expect on average the x and y components of the trajectory to have the same dynamics. Again this is due to the cylindrical symmetry of the system. This will cause the autocorrelation functions in the x and y directions to look almost identical.

In Figure 4.13 we can see the trajectory in the z-direction. For this trajectory,

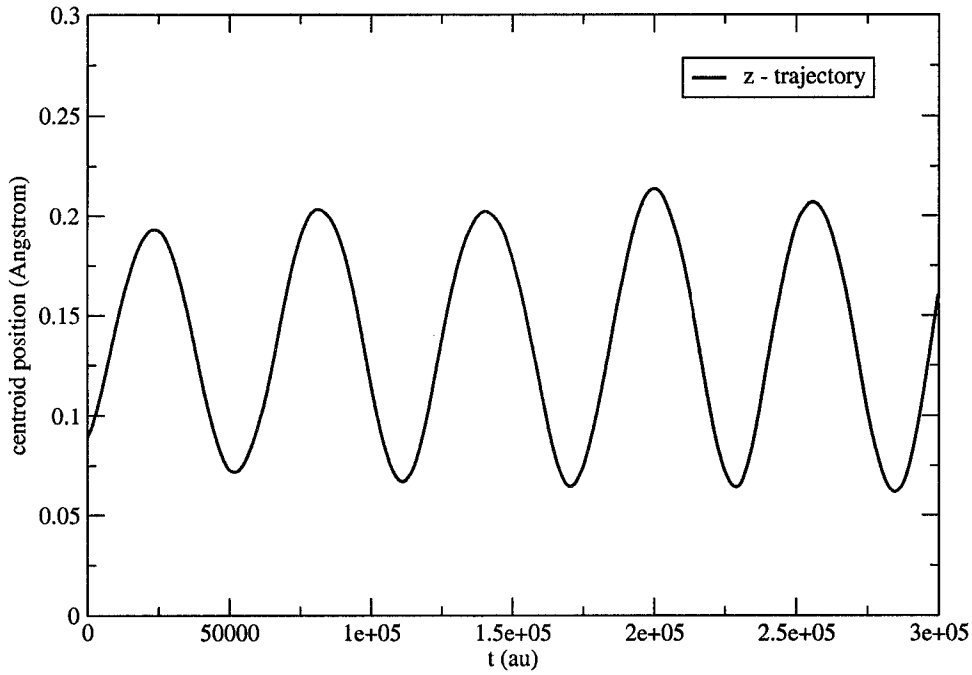

Figure 4.13: The z_c component of a typical trajectory.

the helium atom oscillates nicely about the global minimum centered on the positive side of the centre of mass of the molecule. We expect at low temperatures that the potential will have a harmonic shape. This is shown nicely by the oscillating nature of this trajectory.

We can also see the phase space trajectory for the centroid position and centroid momentum in the z-direction in Figure 4.14. We see that the trajectory is almost a closed loop after one period, which means that there is some energy loss, which is probably given to the motion in the $x - y$ plane. We must keep in mind that the centroid velocities we calculated are not as accurate as the centroid positions, due to the fact that we used the regular Verlet integration algorithm

Within the CMD framework, we are trying to calculate the single Kubo transform

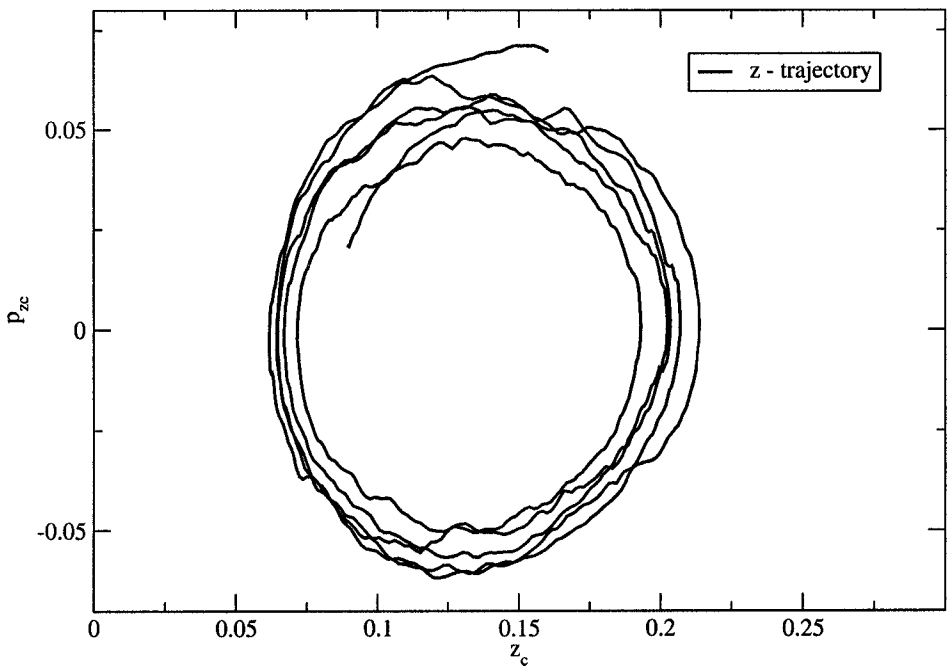

Figure 4.14: The phase space trajectory for the z_c component of a helium atom in the presence of an N_2O molecule.

.

of the quantum mechanical autocorrelation function for the z_c component, which is given by:

$$\langle \hat{z}(0)\hat{z}(t)\rangle^{Kubo} = \int\int \frac{dp_c dq_c}{(2\pi)^3} \frac{\rho_c(p_c, q_c)}{Z} z_c z_c(t). \tag{4.5}$$

We calculate this integral by averaging the trajectories started from initial conditions distributed by $\rho_c(p_c, q_c)$, which was shown in Figures 4.2 - 4.5. For this calculation, we averaged over 2880 trajectories at which point the correlation function was unmodified by the addition of more trajectories in the average. It is possible to form the correlation function with as little as 300 trajectories and maintain the form of the z_c autocorrelation function. Unfortunately, the x_c and y_c autocorrelation functions do not agree with what is expected. We sampled our initial conditions at every 100th MC step. In Figure 4.15 we can see in the rectangular box, the initial conditions that were used in the averaging. These initial conditions were chosen because they are produced near the end of the MC calculation, since the beginning of the calculation is considered an *initialization* stage. In other words, we choose an area where we can consider the MC calculation to have reached a converged value.

The z_c autocorrelation function we obtain can be seen in Figure 4.16, as well as in Figure 4.17, where we can see the x_c and y_c autocorrelation functions. We can see that in magnitude the z_c autocorrelation's amplitude is on a completely different scale than the x_c or y_c autocorrelation functions. As well, the period of oscillation of the z_c autocorrelation is on a different scale as well.

As stated, the x_c and y_c autocorrelations functions, due to the cylindrical symmetry, are nearly identical. Unfortunately, we cannot see the full time evolution of these correlation functions. It would be possible to increase the number of MD steps, but we cannot assume that those results would be the true dynamics of the system. Due to the limitations of the Verlet integrator (and all other integrators), we are not able to extend the length of the calculation without increasing the computational time. We show the results in Figures 4.16 and 4.17, with the understanding that they are preliminary results, showing that the method developed can be used to calculate correlation functions. These results may not be accurate, as they are most likely not properly converged. The accuracy of the CMD approximation in an anisotropic

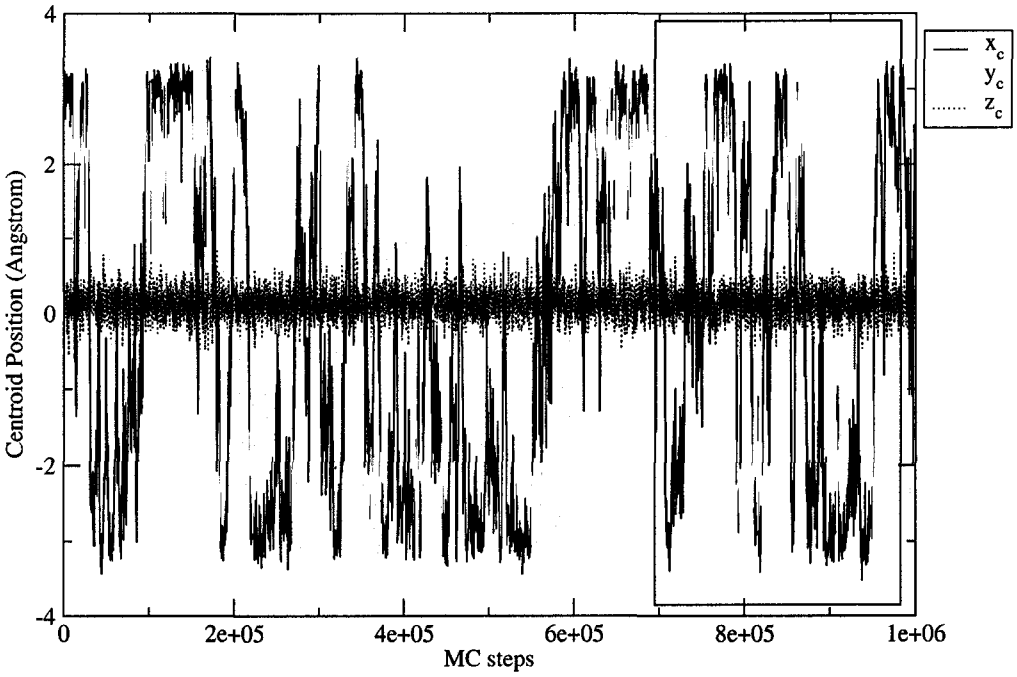

Figure 4.15: The configurational initial conditions from a MC calculation. The solid black line corresponds to the the x_c initial conditions, the solid grey line corresponds to the y_c initial conditions, and the dotted black line corresponds to the z_c initial conditions.

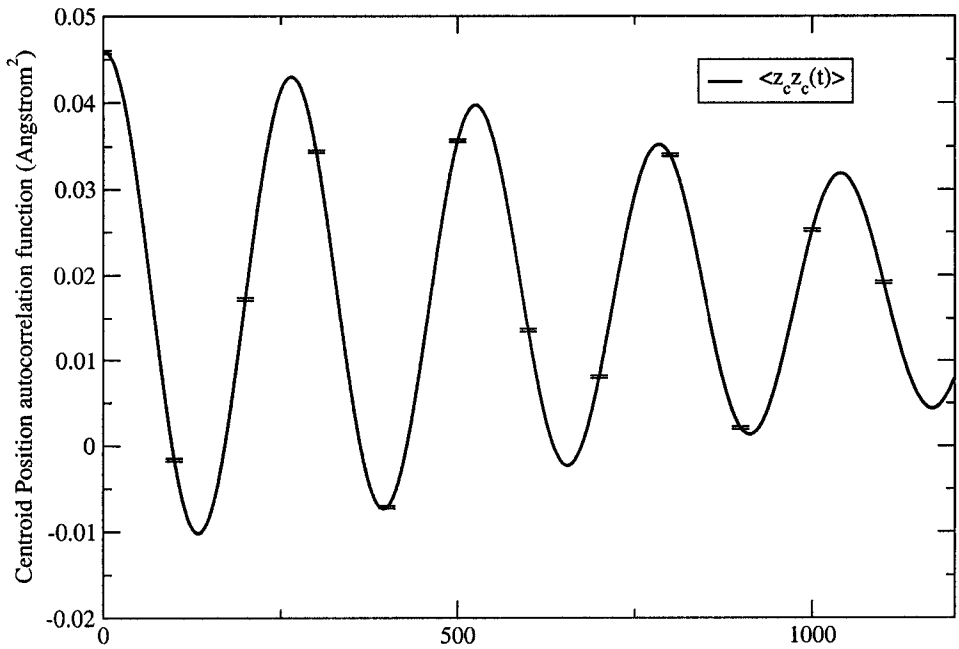

Figure 4.16: The z_c autocorrelation function for a single helium atom in the presence of an N_2O molecule. The error bars correspond to the Monte Carlo statistical error of the initial conditions. The calculation of the errors in the dynamics is a significantly more complicated problem and have not been included.

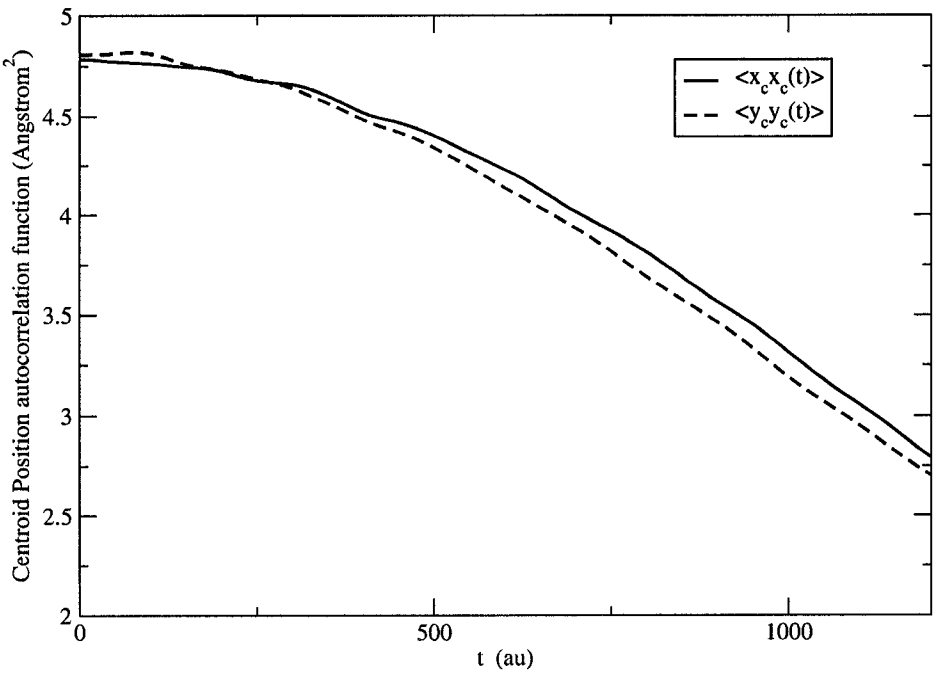

Figure 4.17: The x_c and y_c autocorrelation functions for a single helium atom in the presence of an N_2O molecule. The solid line corresponds to the x_c autocorrelation function, and the dashed line corresponds to the y_c autocorrelation function.

system is still an open question, although we now have a well formed method to test it.

We can, from the Fourier transform of the z_c autocorrelation function, find the dominant energy level spacing which we can estimate at about $-21cm^{-1}$. This value is reassuring, since we are in a low temperature regime of $T = 1K$, and we know from exact calculations that the ground state energy is approximately $-26cm^{-1}$, which is the same order of magnitude of (and lower than) our calculated energy level spacing. We would like to stress that the purpose of the studies presented in this chapter is to set the stage for large scale CMD simulations of systems with anisotropy. Several algorithms and methodologies were developed and we have shown that these types of simulations are possible. More time will have to be spent on "convergence studies" of the correlation functions, however, and these will be the subject of future work in the Roy research group.

Chapter 5

Conclusions

Obtaining dynamical information from a quantum mechanical system is to date still a problem that is not entirely solved. There are numerous methods in use to gain dynamical information but all of them in some way have a drawback. Some methods are exact, such as directly solving the time dependent Schroedinger equation. These direct methods unfortunately are very time consuming. They become exponentially more difficult as the number of particles is increased. To avoid this exponential scaling we chose an approximate method. Although BECMD is an approximation, it allows us to write the quantum mechanical system in a classical-like form. In this classical-like form we can use classical-like methods such as Monte Carlo algorithms which scale linearly with the number of particles in the system. This linear scaling in an immense improvement upon the exponential scaling of the direct method. This thesis hoped to show that BECMD is a valid approximation to the true quantum mechanical dynamics.

We can make two principal conclusions about the work presented in this document. First, from Chapter 3 we can say that using the BECMD formalism, we can correctly account for quantum exchange effects. We presented a proof of principle demonstration of BECMD's ability to calculate single particle correlation functions, a feat which has never been successfully accomplished with similar approximate methods. Secondly, in Chapter 4, the CMD method has been laid out for further studies of anisotropic systems. This thesis was concerned with developing new tools to allow us to achieve these goals and we can categorize this work as *applications and*

advancements of CMD methods.

The ability to perform a CMD simulation on an anisotropic system, using an *on the fly* force calculation is a major stepping stone in the world of CMD. This is due to the fact that we require a constrained PIMC calculation to perform the simulation. An excerpt from a recent review [37] on CMD reported *"The main limitation of CMD is a consequence of being a constrained dynamic approximation, ie., the dynamic states accessible along the time evolution are severely limited. These states corresponds to a fixed centroid path integral and we do not see any feasible way to avoid the limitation of using fixed centroid path integrals in the constrained dynamics"*. With this in mind, we now have a functional method to do just that.

We have tested two methods of performing a constrained PIMC calculation. We have used the natural formalism of F-PIMC with a full Fourier series expansion, where the zero-th mode of the expansion is identically the centroid position. The constraint on the centroid position is accomplished by simply not adjusting the zero-th mode in the MC Metropolis step. We have also developed an approximate method using a constraint with DT-PIMC. We formally introduce the constraint via the addition of a delta function in the centroid density. We express this delta function as the integral of an exponential and approximate it by a Gaussian, $\delta(q_c - \frac{1}{P}\sum_{i=1}^{P} q_i) = e^{-\frac{\alpha}{\beta}(q_c - \frac{1}{P}\sum_{i=1}^{P} q_i)^2}$. We stress that the importance of the DT-PIMC approximate method will be most useful since there is already a well documented method of including exchange in DT-PIMC calculations [22], whereas there is a less documented method in F-PIMC [30].

The inclusion of quantum statistics [6, 11, 7, 38, 39, 40] will play the pivotal role in the successful application of CMD methods to systems such as small helium clusters and nano-droplets as well as the study of cool trapped noble gases that will allow us to study the phenomenon of Bose-Einstein condensation. However, CMD has been applied to many systems that do not require quantum effects, such as the application to non superfluid ^4He [41], para-H_2 [42, 43, 44, 45, 46] and general anharmonic systems [47].

Since the work presented in this thesis consists of new methodologies and algo-

rithms, all computer software codes had to be designed and programmed[1] by the author. We present in Appendix A.1 a schematic of the general code used for the anisotropic system. Simulations run with this code took several days and even several weeks to complete. Having a parallel computer cluster was tantamount to finishing any calculation. On a single processor the same calculation would have taken approximately a year to complete.

Future Directions

As has been stated many times, we have now set down the framework for future research to be conducted. The first aspects that need to be addressed are the issues of convergence and accuracy. MC calculations are by their nature statistical calculations and are subject to statistical errors. These, of course, can be eliminated simply by running longer calculations. There are also systematic errors, due to the finite discretization of the Feynman path, which can be eliminated by choosing a large enough discretization. There may also be errors associated with the integration of the equations of motion from the MD portion of the algorithm. All of these convergence issues must be addressed before we can claim to have an accurate result.

After we can accurately assess the quality of the CMD approximation in a system such as the one used in Chapter 4, we can then move on to adding more than one particle into the system. With the BECMD formalism, the current methods to add exchange in a PIMC calculation, and the new method developed within this project to add a constraint on the centroid position, we should be able to account for the quantum statistics of the system. At this stage, we would no longer need to limit ourselves to systems of rotors. The algorithm and code developed in this project are general enough to take any potential and thus will work in general for almost any system. Other such systems we could study are trapped noble gases, or other trapped atoms like Rubidium that at low temperature undergo Bose-Einstein condensation.

Currently we are limited by the fact that for systems such as the N_2O potential, we have no formal way to include rotational dynamics within the CMD framework.

[1] Approximately 10000 lines of code were written for this project

To go beyond pinning the molecule in place like we did in this project, we would of course need to develop a method to include rotational degrees of freedom. With the ability to perform simulations with rotational degrees of freedom would allow us to accurately model a doped helium cluster. There have been many recent experiments and simulations with these types of systems [33, 34, 35]. BECMD should in principle allow us to calculate and compare properties of these systems with the results from experiment and assess the importance of exchange effects on the dynamics.

Finally, as one more prospect, since in principal CMD can be extended to any quantum statistics, we can essentially treat fermionic systems using FDCMD. Unfortunately, this may be a lofty goal, since FDCMD does not eliminate the fermion sign problem, which plagues all fermionic calculations.

Bibliography

[1] B. HARLAND, *Semiclassical dynamics of constrained molecular systems in Cartesian coordinates* (M.Sc. Thesis, University of Alberta, 2002).

[2] M. P. Marder, *Condensed Matter Physics* (Wiley, 2000).

[3] D. A. McQuarrie, *Statistical Mechanics* (Harper and Row, 1976).

[4] D. Chandler, *Introduction to Modern Statistical Mechanics* (Oxford University Press, 1987).

[5] R. P. Feynman and A. R. Hibbs, *Quantum Mechanics and Path Integrals* (McGraw-Hill, 1965).

[6] P. Moffatt, N. Blinov, and P.-N. Roy, J. Chem. Phys **120**, 4614 (2004).

[7] N. Blinov and P.-N. Roy, J. Chem. Phys **115**, 7822 (2001).

[8] S. Jang and G. A. Voth, J. Chem. Phys **111**, 2357 (1999).

[9] S. Jang and G. A. Voth, J. Chem. Phys **111**, 2371 (1999).

[10] R. P. Feynman, *Statistical Mechanics* (W. A. Benjamin, 1972).

[11] P.-N. Roy and G. Voth, J. Chem. Phys. **110**, 3647 (1998).

[12] N. Blinov and P.-N. Roy, J. Chem. Phys **116**, 4808 (2002).

[13] R. Kubo, M. Toda, and N. Hashitsume, *Statistical Physics II* (Springer, 1992).

[14] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, 1987).

[15] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[16] K. Binder and D. Heermann, *Monte Carlo Simulation in Statistical Physics* (Springer, 2002).

[17] J. Cao and G. A. Voth, J. Chem. Phys. **100**, 5093 (1994).

[18] J. Cao and G. A. Voth, J. Chem. Phys. **100**, 5106 (1994).

[19] J. Cao and G. A. Voth, J. Chem. Phys. **101**, 6157 (1994).

[20] J. Cao and G. A. Voth, J. Chem. Phys. **101**, 6168 (1994).

[21] J. Cao and G. A. Voth, J. Chem. Phys. **101**, 6184 (1994).

[22] D. Ceperley, Rev. Mod. Phys **67**, 279 (1995).

[23] J. D. Doll, D. L. Freeman, and T. L. Beck, Adv. Chem. Phys **LXXVIII**, 61 (1990).

[24] J. Doll, J. Chem. Phys. **81**, 3536 (1984).

[25] J. Lobaugh and G. A. Voth, J. Chem. Phys. **97**, 4205 (1992).

[26] C. Chakravarty, Int. Rev. Phys. Chem. **16**, 421 (1997).

[27] C. Chakravarty, Mol. Phys. **84**, 845 (1994).

[28] C. Chakravarty, J. Chem. Phys. **116**, 8938 (2002).

[29] C. Chakravarty, M. Gordillo, and D. Ceperley, J. Chem. Phys **109**, 2123 (1998).

[30] C. Chakravarty, J. Chem. Phys. **99**, 8038 (1993).

[31] S. L. Mielke and D. G. Truhlar, J. Chem. Phys. **114**, 621 (2002).

[32] M. Frigo and S. G. Johnson, FFTW: An adaptive software architecture for the FFT, in *Proc. 1998 IEEE Intl. Conf. Acoustics Speech and Signal Processing*, volume 3, pages 1381–1384 (IEEE, 1998).

[33] X. Song, Y. Xu, P.-N. Roy, and W. Jäger, J. Chem. Phys. (2004), Submitted.

[34] Y. Xu, W. Jäger, and A. R. W. McKellar, Phys. Rev. Lett. **91**, 163401 (2003).

[35] K. Nauta and R. E. Miller, J. Chem. Phys. **115**, 10254 (2001).

[36] W. H. Press, W. T. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical Recipes in C* (Cambridge, 1998).

[37] R. Ramirez and T. Lopez-Ciudad, Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms **10**, 325 (2002).

[38] K. Kinugawa, H. Nagao, and K. Ohta, Chem. Phys. Lett. **307**, 187 (1999).

[39] K. Kinugawa, H. Nagao, and K. Ohta, J. Chem. Phys. **114**, 1454 (2001).

[40] K. Kinugawa, H. Nagao, and K. Ohta, Prog. Theor. Phys. Supp , 531 (2000).

[41] S. Miura, S. Okazaki, and K. Kinugawa, J. Chem. Phys. **110**, 4523 (1999).

[42] M. Pavese and G. A. Voth, Chem. Phys. Lett. **249**, 231 (1996).

[43] F. J. Bermejo, K. Kinugawa, C. Cabrillo, and S. M. B. et al, Phys. Rev. Lett. **84**, 5359 (2000).

[44] K. Kinugawa, P. B. Moore, and M. L. Klein, J. Chem. Phys. **109**, 610 (1998).

[45] K. Kinugawa, P. B. Moore, and M. L. Klein, J. Chem. Phys. **106**, 1154 (1997).

[46] K. Kinugawa, Chem. Phys. Lett. **292**, 454 (1998).

[47] G. Krilov and B. J. Berne, J. Chem. Phys. **111**, 9140 (1999).

Appendix A

Schematic of the code used for the anisotropic system

To help illustrate this project, a schematic of the C++ code written to perform the simulations is given in Figure A.1. The figure describes the essential parts of the calculation for a single trajectory while calculating the centroid force *on the fly*. The first step before the MD loop, *Initialize centroid position and centroid momentum*, is in itself a complex PIMC calculation. From this step we obtain a set of initial conditions that we can use to start the individual trajectories.

If we look at the cartoon representation of our algorithm in Figure 4.1, this first step corresponds to the horizontal *squiggle arrow*, and for each vertical *squiggle arrow* we must run the following MD loop. It is at this point when we can see the extreme parallel nature of the algorithm. Once the first initialization step is completed we can run each trajectory on a different processor.

The path refining move inside the MC loop is the only difference between the F-PIMC and DT-PIMC calculations. For F-PIMC we need to Fourier transform the path, and make the re-shaping move by altering the Fourier components where as in DT-PIMC we use the multi-level Metropolis algorithm. Since we are trying to perform a constrained PIMC calculation, we must first eliminate the global move, which moves the entire path of the particle without altering the shape of the path. If we are using F-PIMC we must then also make sure we are not adjusting the zero-th Fourier mode since it corresponds to the *centre of mass* of the path, which is identically the centroid

position. If we are using DT-PIMC, we must alter the action by adding the Gaussian approximation to a delta function. The Metropolis algorithm checks to see if the new configuration of the path will be accepted or rejected. After the Metropolis algorithm, regardless of whether we accepted or rejected the new configuration, we calculate the value of our estimator and update the average.

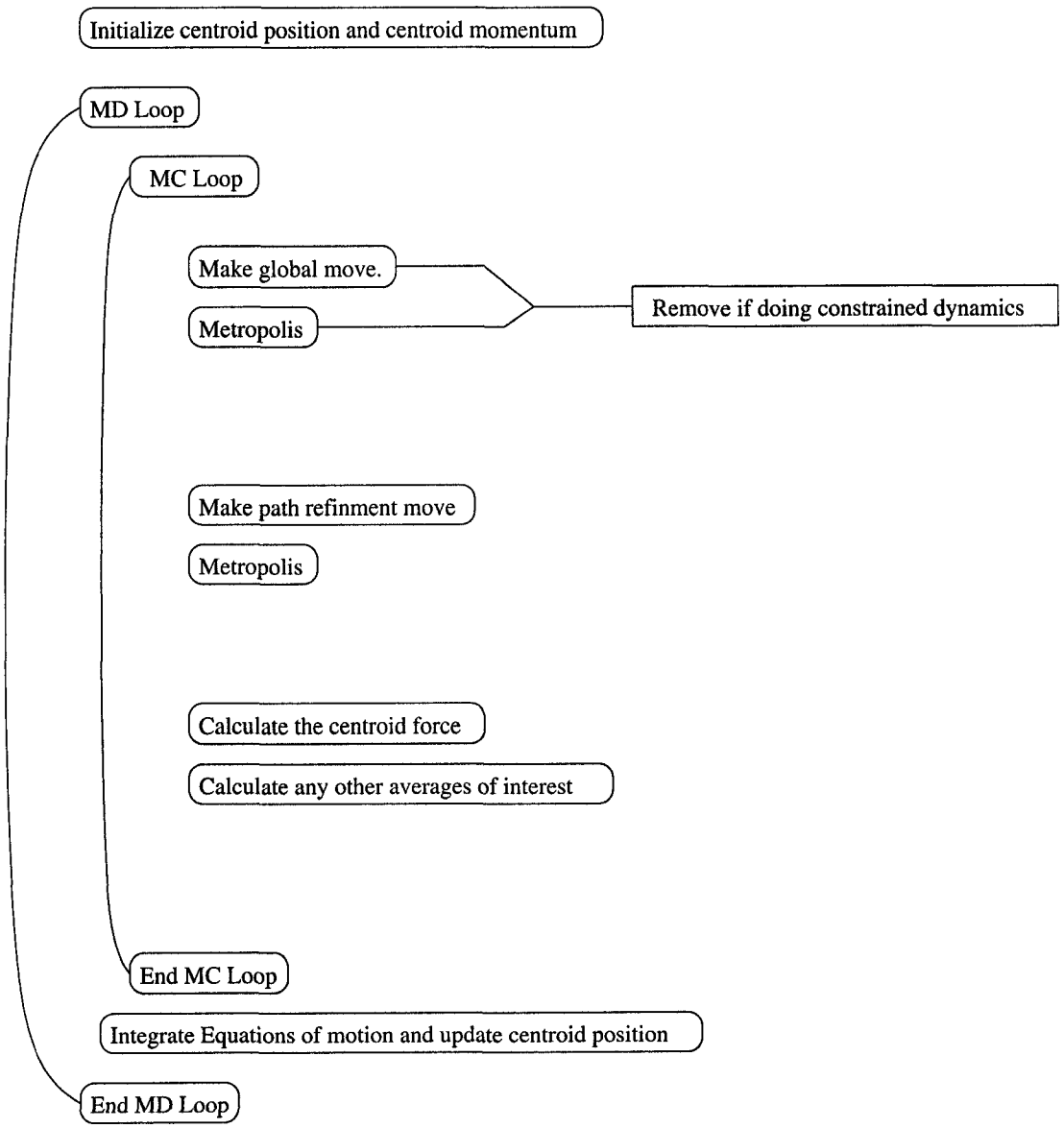

Figure A.1: Schematic of developed algorithm.