ITERATIVE MONTE CARLO FOR QUANTUM DYNAMICS

BY

VIKRAM JADHAO

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Physics
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Professor Martin Gruebele, Chair
Professor Nancy Makri, Director of Research
Assistant Professor Aleksei Aksimentiev
Professor John D Stack

Abstract

We present an exact path integral methodology for computing quantum dynamical information. This method combines the concepts of iterative propagation with the features of Monte Carlo sampling. The stepwise evaluation of the path integral circumvents the growth of statistical error with time and the use of importance sampling leads to a favorable scaling of required grid points with the number of particles. Three different Monte Carlo sampling procedures are presented. Time correlation functions for several multi-dimensional model systems are computed and accurate long time dynamics are obtained. In the end, the capabilities and limitations of the method are discussed.

To Baba and Aai

To Maren Somers

Acknowledgments

I would like to thank my advisor Professor Nancy Makri for giving me the opportunity to work with her and learn from her. I am honored to have witnessed her deep insight and clarity of thought. She has always been very patient, kind, and generous towards me even when it was not clear for quite some time where the project was going.

I want to thank Professor Todd Martinez for his suggestions and useful comments during my preliminary exam. I am grateful to Professor Martin Gruebele and Professor Aleksei Aksimentiev to have agreed to be on my final defense on such short notice. I want to thank Professor John Stack for being a member of my preliminary and final exams and always being very supportive and helpful when I talked to him.

I want to thank Professor Nigel Goldenfeld from whom I learned statistical mechanics in my first year and Professor Paul Goldbart whose lectures on quantum mechanics helped to improve my understanding of the subject. I am indebted to Professor Somnath Bharadwaj at IIT Kharagpur for igniting my interest in quantum mechanics. I owe a lot to Professor Pinaki Mazumdar and Professor G. Baskaran for my decision to do research in physics. I want to thank my teacher at D. M. School, Dr. Kameswar Rao, for creating my interest in mathematics, and K. K. Swain Sir for sharing his physics and mathematics with me during my high school years.

I want to thank Amit Pratap Singh Yadav for introducing me to the Monte Carlo method in my first year at Urbana, Shiladitya Chakraborty for giving me the book by Thijssen and not asking for it back, Jonathan Chen for valuable conversations and his books, Lisandro Hernández for helping me with my presentation for interviews and discussing various ideas,

and Cristian Baltaretu for useful discussions about IMC. I want to thank Professor Werner Krauth for his lectures on Monte Carlo which have helped me throughout my research.

I also want to express my gratitude towards all members of the Makri group, present and past. Especially, I want to thank Mohammad and Roberto for always being nice towards me and great fun to hang out with. I want to thank my friends from the Physics department — Soheil for a lot of fun playing board games and messing around, Stefanos for his encouragement and solid friendship, Satwik for laughing with me at the first adventures of Prem Chopra, Rahul Biswas for being at the café, Taner for memorable first year moments, Parag for all his help during my first year here, and Mitrabhanu for his help and teaching me how to cook.

I have met so many wonderful people during my stay here. I cherish the time I spent with Praveen, Ben, Patrick, Imran, Dan, Themis, Victor, Ghazi, Amélie, and Chelsea.

I want to thank Maren L. Somers for existing and Vasilica Crecea for calmness.

I thank Amit Mohanty for his unconditional care.

I thank Ghalib, who is 213 years old now, for his verses. As I near the end of my PhD I think I understand what he meant when he said *“Mushkilein mujh per padi itni ke aasaan ho gayeen”*.

I would like to thank my father for my first mathematics lesson and his endless physics questions. Finally, I thank my parents and my brother for their love.

This work was supported in part by the National Science Foundation under Award Nos. ITR 04-27082, CHE 05-18452, CRIF 05-41659, and CHE 08-09699. Calculations were performed on a Linux cluster acquired through Award No. CRIF 05-41659.

Table of Contents

List of Figures . . . . **viii**

Chapter 1 Introduction **1**
1.1 Quantum dynamics 2
1.2 Imaginary time path integral 5
1.3 The Monte Carlo method 8
1.4 The sign problem 10
1.5 Time correlation functions 13

Chapter 2 Motivation **17**
2.1 Monte Carlo-based approaches 18
2.2 Iterative propagation 20
2.3 Goal and outline 23

Chapter 3 Iterative Monte Carlo **25**
3.1 The basic idea 25
3.2 The method 27
3.2.1 Iterative structure 28
3.2.2 IMC algorithm 31
3.3 Implementation 33
3.3.1 Sampling strategies 34
3.3.2 Local summations 36
3.4 Recap and Preview 40
3.5 Potential-only sampling 41
3.6 First Results 43

Chapter 4 IMC with bead-adapted sampling **47**
4.1 Bead-adapted sampling strategy 47
4.1.1 Sampling function 48
4.1.2 Computing marginal distributions 49
4.2 $t = 0$ vs $t > 0$ 52
4.3 Results 54
4.4 Summary 56

Chapter 5 IMC with whole-necklace sampling **62**
5.1 Whole-necklace sampling strategy 63
5.2 Results 66
5.3 Summary 69

Chapter 6 Conclusion **74**

Appendix A **77**

References **78**

List of Figures

3.1 Discretization of the path integral for the propagator in the iterative procedure described in section 3.2.1 30
3.2 The circle described by the equation (3.29) 38
3.3 Complex time position autocorrelation function for a one dimensional harmonic oscillator with unit mass and frequency at $\beta = 1$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1000 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation. 45
3.4 Complex time position autocorrelation function for a one dimensional harmonic oscillator with unit mass and frequency at $\beta = 3$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1500 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation. 45
3.5 Complex time position autocorrelation function for a one dimensional anharmonic oscillator, $V(x) = \frac{1}{2}x^2 + \frac{1}{5}x^4$, with unit mass at $\beta = 0.5$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1500 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation. 46

4.1 Grid $P_{2k-1}(x'_k, x_k)$ generated using the bead-adjusted sampling procedure (blue dots), compared to the exact absolute value of the propagator $|R_{2k-1}(x'_k, x_k)|$ (red dots), for the k^{th} bead pair, in the case of a harmonic oscillator for $k = 1$. (a) $\hbar\omega\Delta t_c = -0.5i$ and (b) $\hbar\omega\Delta t_c = 3 - i0.5$. 58
4.2 Same as in Figure 4.1 but for $k = 2$. 58
4.3 Same as in Figure 4.1 but for $k = 5$. 59
4.4 Complex time position autocorrelation function for a harmonic oscillator with unit mass with fixed Δt_c. The solid line shows exact results. The blue squares show results obtained with the IMC method with 5000 grid points. The red hollow circles show PIMC results, with the number of samples adjusted to have the same number of operations as IMC. (a) $\hbar\omega\Delta\beta = 0.25, \omega\Delta t = 0$ (b) $\hbar\omega\Delta\beta = 0.25, \omega\Delta t = 0.5$. 59

4.5 Complex time position autocorrelation function for a one-dimensional harmonic oscillator with $\hbar\omega\beta = 1$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained from a conventional Metropolis Monte Carlo evaluation of the complex time path integral expression, with the number of samples adjusted to have the same number of operations as IMC. 60

4.6 Complex time position autocorrelation function for a 4-dimensional multi-frequency harmonic oscillator with unit mass at $\beta = 5$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained conventional PIMC procedure, with the number of samples adjusted to have the same number of operations as IMC. See the text for additional details. 60

4.7 Complex time position autocorrelation function for a 7-dimensional multi-frequency harmonic oscillator with unit mass at $\beta = 5$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained conventional PIMC procedure, with the number of samples adjusted to have the same number of operations as IMC. See the text for additional details. 61

4.8 Complex time position autocorrelation function of a d-dimensional harmonic oscillator with unit mass and frequency for $\beta = 5$. Solid Line: exact results as a function of the number d of dimensions. Solid blue squares: results obtained with the IMC method using $2N - 1 = 5$ (10 path integral beads). Hollow red circles with error bars: results obtained via conventional Metropolis Monte Carlo evaluation of the complex time path integral, with 10 path integral beads and the number of samples adjusted to have the same number of operations as IMC. (a) $t = 0$, IMC performed with a total of 10000 grid points. (b) $t = 4$, IMC with performed with $10000d$ grid points up for $d \leq 9$ and with an addition of 100,000 grid points per dimension for $d \geq 10$. 61

5.1 Illustration of the path integral necklace for the complex time correlation function, (5.1). The upper half segment corresponds to the complex time forward propagator, (3.9). The lower half necklace corresponds to the backward propagator. 63

5.2 Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 1$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling. 70

5.3 Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 3$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling. 70

5.4 Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 5$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling. 71

5.5 Single-bead distributions, (5.14), as obtained from integrating the two-bead distributions (obtained with whole-necklace sampling) shown in Figures 5.2, 5.3, and 5.4. Red, green, blue, cyan and gold correspond to results obtained from the two-bead distributions with $k = 1, 2, 3, 4, 5$ respectively. 71

5.6 Position correlation function with fixed complex time step, (5.15). Blue squares: whole-necklace IMC with 10000 grid points. Hollow circles: PIMC results with the same number of operations. 72

5.7 Zero-time position correlation function for a d-dimensional harmonic oscillator. Green triangles: IMC with bead-adapted sampling. Blue squares: IMC with whole-necklace sampling. 72

5.8 Position correlation function for a d-dimensional harmonic oscillator. Blue squares: IMC with whole-necklace sampling. Red circles: PIMC results with the same number of operations. 73

5.9 Position correlation function for a one-dimensional anharmonic oscillator. Blue squares: IMC with whole-necklace sampling. Red circles: PIMC results with the same number of operations. 73

Chapter 1

Introduction

Feynman makes a general argument in his paper "Simulating physics with computers" [1] on why quantum dynamics is a hard computational problem. It goes as follows. In quantum mechanics one is only allowed to compute probabilities. So to specify the initial state of a system we would need a probability distribution. For example, to specify the initial state of a single particle we need to supply probabilities at different configurations of this system, which in this case are different position values. Let's say we specify (or store in a computer) this information with n numbers. For M particles, the number of configurations that need to be specified with probabilities grows as n^M. Hence, n^M numbers are needed to be stored. It is easily seen that as n and M grow, storing the initial conditions will get out of hand pretty quickly. This exponential scaling is the central problem in computing quantum dynamics. It is a direct consequence of the non-locality inherent to the quantum theory. In contrast, for a system evolving under the laws of classical mechanics, $6M$ numbers are needed to be stored (the factor of 6 comes from the initial three position coordinates and three momenta). This linear scaling with number of particles has led to the development of methods based on numerical integration of classical equations of motion, also known as molecular dynamics, which have enjoyed enormous success in computing the dynamics of classical systems. In the same paper, Feynman suggested that a *quantum computer*, i.e a computer that itself operates under the laws of quantum mechanics, will offer a solution. While there has been recent progress in both the experimental [2] and algorithmic [3] aspects of quantum computing, large quantum computers are far in the future. This necessitates the development of quantum dynamical methods for classical computers and this dissertation presents one such method. In this chapter, we review the essentials needed to understand the method which will be presented in later chapters.

1.1 Quantum dynamics

The problem of quantum dynamics is to determine the properties of the wave function Ψ of a system at time t, given the initial wave function at $t = 0$ and the Hamiltonian $\hat{H}$ of the system. This problem is answered by solving the time-dependent Schrödinger equation (TDSE),

$$i\hbar\frac{\partial\Psi}{\partial t} = \hat{H}\Psi. \tag{1.1}$$

This equation can be solved analytically for very few Hamiltonians, e.g. the free particle, the linear potential, the harmonic oscillator. Computationally however, numerically exact solutions for arbitrary $\hat{H}$ can be obtained. For this to work, the (initial) wave function needs to be stored on a grid to start the numerical iteration. The cost (memory) of storing the wave function, as argued before, increases exponentially with the number of degrees of freedom. Therefore, exact results are only obtained for small systems via grid-based or basis-sets methods [4]. The exponential *storage* problem can be avoided in Feynman's path integral approach to quantum mechanics [5, 6]. In this formulation the dynamics is given by the integral

$$\Psi(x,t) = \int dx' K(x,x';t)\Psi(x',0), \tag{1.2}$$

where $K(x,x';t) \equiv \langle x|e^{-it\hat{H}/\hbar}|x'\rangle$ contains all information about the dynamical behavior of a system and is known as the propagator.[1] We will adopt a one-dimensional notation in much of the following, realizing that the generalization to many dimensions is easy. It can

[1] Also called the Green's function or the kernel

be shown that the propagator satisfies the TDSE

$$i\hbar\frac{\partial K}{\partial t} = \hat{H}K \tag{1.3}$$

with the initial condition $K(x, x'; 0) = \delta(x - x')$. Since the TDSE is not solvable analytically for a general Hamiltonian, $K(x, x'; t)$ is unavailable analytically except for the few special cases mentioned before. However, it can be proved [7] that the *short time* propagator of *any* system can be well approximated by simple analytic expressions, such as the standard Trotter formula [8]:

$$\begin{aligned} K(x, x'; \Delta t) \equiv \langle x|e^{-i\Delta t\hat{H}/\hbar}|x'\rangle =& \Big(\frac{m}{2\pi i\hbar\Delta t}\Big)^{1/2} \\ & \times \exp\left[\frac{i}{\hbar}\Big(\frac{m}{2\Delta t}\Big(x - x'\Big)^2 - \frac{\Delta t}{2}\Big(V(x) + V(x')\Big)\Big)\right], \end{aligned} \tag{1.4}$$

where Δt is a short time. The path integral provides the link between the propagator for any time t and the (analytically) known short time propagator. Here is how the link is established. The time evolution operator is factored into a product of N exponential operators, $e^{-it\hat{H}/\hbar} = e^{-i\Delta t\hat{H}/\hbar}e^{-i\Delta t\hat{H}/\hbar}\ldots e^{-i\Delta t\hat{H}/\hbar}$, where N (the number of time discretizations or slices) is chosen such that $\Delta t = t/N$ is a short time for the system of interest. It is clear that each of these exponential operators, called the short time evolution operators, propagates the system for time t/N. Complete sets of position eigenstates $\int dx_k|x_k\rangle\langle x_k| = 1$ are inserted between the short time evolution operators, and the following result is obtained:

$$\langle x_N|e^{-it\hat{H}/\hbar}|x_0\rangle = \int dx_1 \cdots \int dx_{N-1}\langle x_N|e^{-i\Delta t\hat{H}/\hbar}|x_{N-1}\rangle \cdots \langle x_2|e^{-i\Delta t\hat{H}/\hbar}|x_1\rangle\langle x_1|e^{-i\Delta t\hat{H}/\hbar}|x_0\rangle, \tag{1.5}$$

where $x_N \equiv x$ and $x_0 \equiv x'$. After using the Trotter formula for the short time propagators in (1.5) we arrive at the result

$$K(x_N, x_0; t) = \left(\frac{m}{2\pi i\hbar\Delta t}\right)^{\frac{N}{2}} \int dx_1 \cdots \int dx_{N-1} \exp\left[\frac{i}{\hbar}\left(\frac{m}{2\Delta t}\sum_{k=1}^{N}\left(x_k - x_{k-1}\right)^2 - \Delta t \sum_{k=0}^{N} \lambda_k V(x_k)\right)\right], \tag{1.6}$$

where $\lambda_k = \frac{1}{2}$ for $k = 0, N$ and $\lambda_k = 1$ otherwise. Each point x_k is associated with a particular time $t_k = k\Delta t/N$, and thus the sequence $\{x_k\} \equiv x_0, x_1, x_2, \ldots, x_{N-1}, x_N$ denotes a *path* in space-time connecting the fixed initial and final points, x_0 and x_N. The propagator is obtained by summing over all possible paths that connect x_0 and x_N, i.e. by performing a *path integral.* The paths contribute equally in magnitude since the absolute value of the integrand in (1.6) is 1 for any sequence x_k. The presence of i in the exponent makes the integrand oscillatory, implying paths in general can cancel each other's contributions (or interfere). Hence, computing the path integral amounts to summing the contributions (which involves cancellation) of an enormous number of equally-weighted paths. On a computer the evaluation of this integral (call it I) starts by initializing it to 0 ($I = 0$). A sequence $\{x_k\}$ is then picked and the integrand is computed and the result is added to I. The process is repeated for another sequence until all possible sequences are picked. The final result is then scaled by a factor which depends on the way the sequences were picked – for example, randomly or in an orderly fashion. Note that at no stage we are required to store the sequences (paths). Hence the path integral avoids the exponential storage problem associated with the Schrödinger representation. However, numerical summation over paths presents problems of its own. The path integral is a multi-dimensional integral with, in general, $(N-1)d$ dimensions, where d is the number of spatial dimensions. The dimensionality of this integral becomes very high for long propagation times as a larger N is needed to maintain the accuracy of short-time propagators. Numerical evaluation by quadrature (e.g. the trapezoidal rule) becomes exponentially more expensive as the dimensionality of the integral increases [4]. Monte Carlo methods [9], which are suited for high dimensional integrals, fail to compute (1.6) because the oscillatory nature of the integrand leads to the notorious sign problem [10, 11, 12](see Sec 1.4). One way to avoid the sign problem is to compute the integral in (1.5) one at a time from right to left [13]. This will take us back to the idea of storing the propagated quantity that, as we discussed earlier, gets exponentially more expensive with the system size d. The split propagator method [14] and numerical matrix multiplication (NMM) [13] scheme employ the conventional (uniform) grid to carry out this iterative propagation and work only for small systems of one, two or possibly three degrees of freedom. Therefore, real-time path integration remains a challenge despite significant efforts in the past few decades [11, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28] (see Chapter 2). Quite ironically, the path integral formalism – which was invented to extract quantum dynamical information, has become a powerful and practical computational approach to obtain time-independent (equilibrium) properties of a quantum system [29]. We explain this next.

1.2 Imaginary time path integral

If we let $t \to -i\hbar\tau$ in (1.6), that is go to an *imaginary time* τ (where τ is assumed to be a positive real number), we obtain

$$K^{Im}(x_N, x_0; \tau) = \left(\frac{mN}{2\pi\hbar^2\tau}\right)^{\frac{N}{2}} \int dx_1 \cdots \int dx_{N-1} \exp\Bigg[-\Bigg(\frac{mN}{2\hbar^2\tau}\sum_{k=1}^{N}\Big(x_k - x_{k-1}\Big)^2 + \frac{\tau}{N}\sum_{k=0}^{N}\lambda_k V(x_k)\Bigg)\Bigg]. \tag{1.7}$$

Equation (1.7) is the imaginary time path integral and $K^{Im}(x_N, x_0; \tau)$ is the imaginary time propagator. This path integral is easier to visualize than the real time counterpart (1.6). Here, we can see that there are paths that contribute very little to the integral (as compared to before when all of them contributed the same), for there are paths for which the exponent is very large and thus the integrand is negligibly small. Moreover, we don't need to worry about paths cancelling each other's contributions because the integrand is no longer oscillatory (there is no i in the exponent), making all contributions add together with some being large and others small [6]. This invites the application of Monte Carlo methods which are the subject of the next section. If we now identify $x_N \equiv x_0$ (*closing* the path) and take the integral over x_0 on both sides of (1.7), we get

$$\begin{aligned}\int dx_0 K^{Im}(x_0, x_0; \tau) = \Big(\frac{m}{2\pi\hbar^2\Delta\tau}\Big)^{N/2} \int dx_0 \int dx_1 \cdots \int dx_{N-1} \\ \exp\Big[-\Big(\frac{mN}{2\hbar^2\tau}\sum_{k=0}^{N}\Big(x_k - x_{k-1}\Big)^2 + \frac{\tau}{N}\sum_{k=0}^{N-1} V(x_k)\Big)\Big].\end{aligned} \tag{1.8}$$

The left hand side of (1.8) is the *trace* of the operator $e^{-\tau\hat{H}}$. Identifying τ with the inverse temperature $\beta = \frac{1}{k_B T}$ the operator becomes the Boltzmann operator and the trace gives the partition function of a quantum system described by the Hamiltonian H at a given temperature T [30]. After rearranging some terms in the exponent on the right hand side we get the following result for the partition function:

$$\begin{aligned}Z = \Big(\frac{mN}{2\pi\hbar^2\beta}\Big)^{N/2} \\ \times \int dx_0 \int dx_1 \cdots \int dx_{N-1} \exp\Big[-\beta\Big(\sum_{k=0}^{N-1}\frac{mN}{2\hbar^2\beta^2}(x_k - x_{k+1})^2 + \sum_{k=0}^{N-1}\frac{1}{N}V(x_k)\Big)\Big].\end{aligned} \tag{1.9}$$

This is an interesting result, because it gives the equilibrium statistical mechanics of a quantum system in terms of an integral that looks, apart from the presence of $\hbar$ and a multiplicative constant, like the configuration integral of N classical particles in a potential given by

$$U(x_1, x_2, \ldots, x_N) = \sum_{k=0}^{N-1}\Big[\frac{mN}{2\hbar^2\beta^2}(x_k - x_{k+1})^2 + \frac{1}{N}V(x_k)\Big]. \tag{1.10}$$

This means that a single quantum particle in a potential $V(x)$ is isomorphic to a chain of N (fictitious) classical particles that interact via harmonic springs of force constant $mN/\hbar^2\beta^2$ and each of them feels a potential equal to $V(x)/N$. The quantum partition function is thus equivalent to the classical partition function for this ensemble of pseudoparticles [7]. From the partition function thermodynamic quantities like free energy, internal energy, entropy, etc. can be computed. For calculating other physical observables, say the average position, we require the density matrix (operator) $\hat{\rho} = e^{-\beta\hat{H}}/Tr(e^{-\beta\hat{H}})$. Using the position representation the density matrix is expressed as

$$\rho(x_N, x_0) = \frac{K^{Im}(x_N, x_0; \beta)}{Z}, \tag{1.11}$$

where K^{Im} and Z are obtained from (1.7) and (1.9). Hence the density matrix is available as the ratio of path integrals. The thermodynamic average of a physical observable A is defined as

$$\langle A \rangle = \frac{Tr(\hat{\rho}\hat{A})}{Z}, \tag{1.12}$$

where $\hat{A}$ is the quantum mechanical operator corresponding to the observable A. By taking the trace in the position representation it is easily shown that this average can also be expressed as the ratio of two imaginary time path integrals. Thus the imaginary time *dynamics* as represented in the path integral formulation, gives the complete description of a quantum system at thermodynamic equilibrium [30]. Moreover, computing such thermal averages now amounts to an exercise in multi-dimensional integration, an exercise most welcome by the Monte Carlo method.

1.3 The Monte Carlo method

To illustrate the ideas in this section we shall consider the integral[2]

$$I = \int_0^1 dx_1 \int_0^1 dx_2 \ldots \int_0^1 dx_d \, f(x_1, x_2, \ldots, x_d). \tag{1.13}$$

One can compute this integral by using a quadrature rule and summing f over, say, a series of equally spaced lattice points in the d-dimensional hypercube. The error associated with such a quadrature formula, like the trapezoidal rule, goes as $O(N^{-2/d})$, where N is the number of (lattice) points used [32]. But again, I can be thought of as an average of $f(x)$ in the hypercube of volume 1, and in this light one can evaluate it by summing the function over random points distributed homogeneously in the d-dimensional hypercube. (1.13) then becomes the sum

$$I = \frac{1}{N} \sum_{i=1}^{N} f(X_i), \tag{1.14}$$

where $X_i \equiv (x_1, x_2, \ldots, x_k, \ldots, x_d)$ is a point selected at random in the d-dimensional hypercube and N is the total number of such points. (1.14) is the Monte Carlo estimate of the integral (1.13). Metropolis and Ulam say in their original paper [31] the estimate "will never be confined within given limits with certainty, but only—if the number of trials is great—with great probability." There will always be a statistical error associated with a Monte Carlo result. Using the central limit theorem the error in the estimate can be shown [32] to be

$$\sigma_I = N^{-1/2} \sigma_f, \tag{1.15}$$

where σ_f is the square root of the variance of the integrand; that is, a measure of the deviation of f from its average value.

Equation (1.15) reveals two important aspects about Monte Carlo (MC). First, the error

[2] Metropolis and Ulam present a similar integral to illustrate the method in perhaps the first paper on Monte Carlo – [31].

in the estimate of the integral decreases as $N^{-1/2}$ no matter what the dimensionality of the integral is! Compare this scaling with the error expression for the trapezoidal rule, $O(N^{-2/d})$. Clearly, as the dimensionality d of the integral increases, the number of points needed to maintain the same precision (error) in a trapezoidal rule will have to be exponentiated with d. On the other hand, MC will converge with about the same number of points as needed to converge the result in one dimension. Second, the error in an MC estimate is less if σ_f is smaller; that is, if f is (or can be manipulated to be) as smooth as possible. For example, take f to be a constant function (the smoothest possible function). Then, even one point gives a great estimate for the average. Next, say the function f is sharply peaked around a particular point and small elsewhere (in other words, f is not smoothly varying). Clearly, a homogeneous distribution of random points will give a very poor estimate to the integral of this function as a majority of points will lie outside the region that contributes most to the integral. Ideally, we would like to select/sample more points in the region where f is large or *important*. This biasing of random points is acheived via importance sampling [32] by introducing a weight function (also called probability density or sampling function) $w(x_1, x_2, \ldots, x_d)$ that is positive definite at all points and is normalized to unity.[3] We write the integral of (1.13) as,

$$I = \int_0^1 dx_1 \int_0^1 dx_2 \ldots \int_0^1 w(x_1, x_2, \ldots, x_d) \frac{f(x_1, x_2, \ldots, x_d)}{w(x_1, x_2, \ldots, x_d)} \tag{1.16}$$

and instead of summing over a homogeneous distribution of random points (in other words, points that are selected with a uniform weight), we distribute points according to the density w, and then sum f/w over the selected distribution. The MC equivalent sum for the integral becomes

$$I = \sum_{X_i \text{ distributed according to } w} \frac{f(X_i)}{w(X_i)}. \tag{1.17}$$

[3] w may not be normalized to unity, but its normalization should be known. When we compute averages as ratios of two integrals, even the latter requirement is relaxed.

One can now see that if w is chosen to have more or less the shape of f, in the sense that the ratio f/w is close to 1, then $\sigma_{f/w}$ is small. This dramatically reduces the error in the MC estimate.

The next question is how to generate points distributed with the probability density w. One can achieve this in an analytical manner by a change of variables (such that $dy = w(x)dx$) and subsequent inversion or by numerical techniques such as the von Neuman rejection method [32]. But these methods work only for a simplified probability density and are not very efficient (if at all applicable) in large dimensions. A general method that easily adapts to high dimensions and works for complicated probability densities is the method by Metropolis *et al* [9]. This algorithm generates a sequence of points $X_0, X_1, \ldots$ via a random walk proceeding through the d-dimensional configuration space according to a *rule*. As the walk becomes longer and longer, the points connected by the random walker approximate more closely the desired distribution. Here is the rule. Suppose the walker is at a point X_n in the sequence. To generate X_{n+1}, it proposes a trial point X_{trial}. This trial point can be chosen in any convenient manner, for example uniformly at random within a small hypercube about X_n. The trial point is then "accepted" or "rejected" according to the ratio $r = w(X_{trial})/w(X_n)$. If $r \geq 1$, then the point is accepted, while if $r < 1$, the point is accepted with probability r. Otherwise, the point is rejected. If accepted we put $X_{n+1} = X_{trial}$, and if rejected $X_{n+1} = X_n$. Successive application of the rule generates $X_{n+2}, X_{n+3} \ldots$, and so on. For details about why this rule works and other subtleties that come in the implementation of the Metropolis algorithm the reader is referred to the textbooks [33][34]. Now let us see in what physical problems the Monte Carlo technique is useful and where it fails.

1.4 The sign problem

In section 1.2 we stated that the expected value of a physical observable in a system that is in equilibrium is given by (1.12). For example, in the case where the observable is potential energy V, we have

$$\langle V \rangle = \frac{Tr[e^{-\beta \hat{H}} \hat{V}]}{Tr[e^{-\beta \hat{H}}]} = \frac{\int dx_0 \langle x_0 | e^{-\beta \hat{H}} | x_0 \rangle V(x_0)}{\int dx_0 \langle x_0 | e^{-\beta \hat{H}} | x_0 \rangle}. \tag{1.18}$$

Using (1.7) and (1.9), we can write this average as a ratio of two path integrals,

$$\langle V \rangle = \frac{\int dx_0 \int dx_1 \cdots \int dx_{N-1} V(x_0) \exp\Big[-\beta \sum_{k=0}^{N-1} \big[\frac{mN}{2\hbar^2 \beta^2} (x_k - x_{k+1})^2 + \frac{1}{N} V(x_k) \big] \Big]}{\int dx_0 \int dx_1 \cdots \int dx_{N-1} \exp\Big[-\beta \sum_{k=0}^{N-1} \big[\frac{mN}{2\hbar^2 \beta^2} (x_k - x_{k+1})^2 + \frac{1}{N} V(x_k) \big] \Big]}. \tag{1.19}$$

Note that we have cancelled the prefactors and $x_0 \equiv x_N$. Also, so far we have been using one-dimensional notation for convenience, but the x_k's are d-dimensional in general and we will remind ourselves of that at this point. Let us examine these integrals. The integrals are high-dimensional, even for one spatial dimension the integral is N-dimensional. For a d-dimensional system the integrand is Nd-dimensional, which even for a small system (say $d = 10$) and large enough time discretization for the Trotter approximation to be valid (say $N = 15$), easily becomes an integral in many (150) dimensions. Because of the presence of the exponential with a real (negative) exponent and high dimensionality, the integrand is peaked in a small region (so has a high variance), and most of the points (or paths) end up with a very small contribution to the integral. Most importantly, the contributions from different points (or paths) in the high-dimensional space all *add*; some being large and some being small. This allows for efficient application of importance sampling, resulting in the reduction of the variance of the sharply peaked integrand and hence lowering the MC error. Moreover, the weight function (which in this case can be taken to be the exponential term) entering the Metropolis algorithm need not be normalized as the computation of ratios results in cancelling of any multiplicative constants. Hence, the path integral representation of the thermal average (or the Boltzmann operator) lends itself naturally to the use of Metropolis Monte Carlo. If the sampling function w appearing in (1.16) is taken to be $w = \exp\Big[-\beta \sum_{k=0}^{N-1} \big[\frac{mN}{2\hbar^2 \beta^2} (x_k - x_{k+1})^2 + \frac{1}{N} V(x_k) \big] \Big]$ then after distributing the points using the Metropolis algorithm, the thermal average becomes

$$\langle V \rangle = \sum_{x_0} V(x_0), \tag{1.20}$$

where the points $x_0, x_1, \ldots, x_{N-1}$ are distributed according to w. This method, known as the path integral Monte Carlo (PIMC), has been established as a powerful tool for evaluating thermal averages in systems with hundreds of quantum degrees of freedom that obey Boltzmann or Bose-Einstein statistics [35][36].

Unfortunately, these advantages do not pertain to quantum dynamics,[4] which is obtained via the path integral representation of the time evolution operator (1.6). The integrand (call it f) entering in (1.6) has the imaginary i in the exponent. Because of this, f is oscillatory and so has a high variance. Since the weight function has to be positive, a natural choice (and the best choice) is the absolute value of f, which is 1. This means that the high variance of f, can not be reduced by a weight function w since f/w has the same variance as f. Most importantly, the contributions to the integral from different regions now need to not just be added (as in the thermal equilibrium case) but also *subtracted*. And there is no way of translating this subtraction, or the delicate cancelling, into w (the positive definite probability density) that enters the Monte Carlo method. As time t increases, f becomes more oscillatory and its variance becomes huge, which leads to a large error (*noise*) in the MC estimate. To make matters worse the result for the integral of f (the *signal*), because of the cancellation, is a very small number. And this causes the *signal/noise* ratio to rapidly approach zero such that the computer time needed to achieve a given accuracy grows exponentially with the dimensionality (Nd) of the integral. This problem is known as the **sign problem**[5] and it has severely hindered the applications of the Monte Carlo method in computing quantum dynamical properties for long times and/or large many-body systems [12, 37, 38]. The problem is not limited to quantum dynamics. In the simulation of fermions

[4] Nor to Fermi-Dirac statistics, we will soon see why.

[5] To be precise the dynamical sign problem

by MC (even in the study of equilibrium properties), due to the antisymmetry of the wavefunction, one has to place a minus sign in the integrand for odd permutations and subtract the contribution of negative permutations from that of the positive permutations, which leads to the (fermion) sign problem [37][38]. In the field-theoretic polymer simulations MC runs into a sign problem when constraints are taken into account via delta functions which when expressed as exponentials involve an i in the exponent [39][40][41]. Many interesting sytems in high-energy physics also encounter this problem where it goes by another name, the 'complex-action problem' [42]. In this thesis we focus on the quantum dynamical sign problem. We will face this problem in the context of computing time correlation functions, which is the subject of the next section. In the next chapter we will review the progress that has been made towards circumventing or alleviating this problem, and will discuss the applicability and limitations of existing methods. We conclude this section with a quote that sums up the power and the weakness of Monte Carlo –"Monte Carlo can add but it can *not subtract*".[6]

1.5 Time correlation functions

Time correlation functions can be shown, by the theory of linear response, to be related to phenomenological coefficients describing time-dependent phenomena [43]. For example, the self-diffusion coefficient can be expressed as an integral over the velocity time correlation function. A general quantum time correlation function is given by

$$\mathbb{C}_{AB}(t) = \langle \hat{A}(0)\hat{B}(t)\rangle = \frac{1}{Z}Tr\big(e^{-\beta\hat{H}}\hat{A}e^{it\hat{H}/\hbar}\hat{B}e^{-it\hat{H}/\hbar}\big), \tag{1.21}$$

where $\hat{A}$ and $\hat{B}$ are quantum mechanical operators corresponding to physical observables. $\mathbb{C}_{AB}(t)$ is also known as the real time correlation function as the correlations are measured between points on the real time axis. We will assume for simplicity that the operators are

[6] I do not know whose quote this is, I heard it from Professor David Ceperley.

local in position, i.e, $\hat{A} = A(\hat{x})$ and $\hat{B} = B(\hat{x})$. We showed in previous sections that the Boltzmann operator, $e^{-\beta\hat{H}}$, and the time evolution operators, $e^{-it\hat{H}/\hbar}$ (forward) and $e^{it\hat{H}/\hbar}$ (backward), can be expressed as path integrals. So, just like equilibrium averages, thermally averaged time correlation functions can also be expressed as ratios of path integrals. Working in position representation, the above expression becomes

$$\mathbb{C}_{AB}(t) = \frac{1}{Z}\int\int\int dx dx' dx'' \langle x|e^{-\beta\hat{H}}|x'\rangle\langle x'|e^{it\hat{H}/\hbar}|x''\rangle\langle x''|e^{-it\hat{H}/\hbar}|x\rangle A(x')B(x''). \tag{1.22}$$

Expressing the three propagators as path integrals results in summing imaginary time paths connecting x and x', forward (real-time) paths connecting x' and x'' and backward paths connecting x'' and x. We know that the real time path summations will lead to cancellation problems which severely limit a direct application of MC based schemes to compute $\mathbb{C}_{AB}(t)$. A form that appears much more convenient from a computational point of view is the symmetrized time correlation function [44],

$$C_{AB}(t) = \langle A(0)B(t - i\hbar\beta/2)\rangle = \frac{1}{Z}Tr\big(\hat{A}e^{i(t+i\hbar\beta/2)\hat{H}/\hbar}\hat{B}e^{i(t-i\hbar\beta/2)\hat{H}/\hbar}\big) \tag{1.23}$$

which is obtained by shifting the domain of the real time correlation function by $-i\hbar\beta/2$ such that the correlations are now measured between points in the complex time plane. Therefore, $C_{AB}(t)$ is also called the complex time correlation function. Introducing the complex time $t_c = t - i\hbar\beta/2$ and working in the position representation the symmetrized time correlation function becomes

$$C_{AB}(t) = \frac{1}{Z}\int\int dx dx' \langle x|e^{it_c^*\hat{H}/\hbar}|x'\rangle\langle x'|e^{-it_c\hat{H}/\hbar}|x\rangle A(x)B(x'). \tag{1.24}$$

We now have two complex time propagators that, when expressed as path integrals, result in summing forward complex time paths connecting x and x' and backward complex time paths connecting x' and x. Just like before with the imaginary time propagator, the (forward) complex time propagator can be expressed as a path integral by letting $t \to t_c$ in (1.6), and one gets

$$G(x_N, x_0; t_c) = \Big(\frac{mN}{2\pi\hbar^2 t_c}\Big)^{\frac{N}{2}} \int dx_1 \cdots \int dx_{N-1} \exp\Bigg[- \Bigg(\frac{mN}{2\hbar^2 t_c} \sum_{k=1}^{N} \Big(x_k - x_{k-1} \Big)^2 + \frac{t_c}{N} \sum_{k=0}^{N} \lambda_k V(x_k) \Bigg) \Bigg]. \tag{1.25}$$

The main advantage of using the complex time correlation function is that the two complex time propagators involved in the integrand are not pure phase, but have an inbuilt "Boltzmann weight" that enables the application of importance sampling technique. Of course, the sign problem is still present and will make the numerical evaluation of $C_{AB}(t)$ very demanding – especially if real (physical) time t is larger compared with the imaginary (thermal) time $\hbar\beta/2$. Nevertheless, Monte Carlo methods should be more stable (especially, when combined with tricks like contour distortion [18] or staging [45] – see chapter 2) in the direct evaluation of $C_{AB}(t)$ as compared to a direct attack on $\mathbb{C}_{AB}(t)$.

It has been shown that the symmetrized time correlation function is related to the real time form through a Fourier transform identity [46]. Thus both the forms contain the same physical information and an accurate estimate of one of them can be used to calculate the other if needed. Also, certain phenomenological coefficients like the thermal rate constant of a reactive process are shown to be directly related to the symmetrized form [47]. It is also useful to know that the symmetrized time correlation function is always purely real while the real time form is in general a complex function of time. Throughout this dissertation we focus on the symmetrized time correlation functions and an exact method will be developed to compute them. But before we introduce the method, let us review existing approaches that were developed in the context of evaluating the real (or complex) time path integral and the time correlation functions. We will discuss their applicability and limitations, and in the process will hopefully reveal the motivations and goals of the method presented in this work.

Chapter 2

Motivation

In this chapter we review computational approaches to the problem of quantum dynamics, showing their applicability and shortcomings, and in the process reveal the motivations behind our work. At the end, we will state the goal of this dissertation and outline the path towards attaining it.

The focus of this brief review is primarily on two major approaches — Monte Carlo-based methods and iterative techniques, which have been developed to evaluate real (or complex) time path integrals. Other alternate approaches to quantum dynamics such as wavepacket methods [48], basis sets techniques [49][50], and quasiclassical methods like centroid molecular dynamics [51] will not be reviewed. These methods are capable of extracting exact dynamics but are in general limited in their applications to certain systems and/or system sizes. We will also not review methods that arise directly from the semiclassical theory, like the semiclassical initial value representation [52] or the forward-backward semiclassical dynamics for time correlation functions [53][54], where the goal is to add in quantum effects to a classical MD simulation and much of the effort is spent in dealing with the 'semiclassical sign problem'. While these methods introduce a great improvement in dealing with large systems [55], they are not reliable if quantum effects are not moderate or if long time dynamics is desired. We have another reason for not carrying out a more complete survey — the method we develop in subsequent chapters derives its ideas from Monte Carlo path integration and iterative methodologies, and so a closer look at the past employment of these techniques will hopefully be of assistance in understanding the goals and features of our method. A good review of other approaches listed before can be found elsewhere [56].

2.1 Monte Carlo-based approaches

We learned in section 1.2 that in imaginary time PIMC faces no cancellation problems and thermal averages can thus be computed. Say, we compute the correlation function (either the standard or the symmetrized form) in imaginary time via PIMC. Thirumalai and Berne showed that one can obtain dynamical information via an analytic continuation of this imaginary time average to real time [15]. The sign problem is thus avoided as no direct computation of the real time path integral is needed and only an accurate inversion of imaginary time data is required. However, due to the rather extreme instability of the numerical inversion (of the Laplace transform) process, even small statistical errors in imaginary time data can lead to large deviations in analytically continued real time quantities [57]. This problem becomes particularly severe with high temperature and long times. The maximum entropy method [25] has improved the applicability of this idea but its range is still limited to short times and systems where quantum coherences dissipate rapidly [57]. Thirumalai and Berne also suggested a direct MC attack on the complex time correlation function introduced in the last chapter [44]. As we discussed in section 1.5, MC would only be reliable for short times, $t < \hbar\beta/2$, and this was confirmed in the work of Behrman and Wolynes [16]. For long times ($t > \hbar\beta/2$) the complex time correlation function was computed by Chang and Miller [18], as well as by Doll *et al* [58], via a contour distortion (CD) technique which converted the kinetic energy part of the integrand into a real (no i) Gaussian factor (which was used as the weight function). This simple coordinate rotation was successfully applied for problems involving potential barriers but was invalid for potential wells as the integrand became unbounded for these potentials [18].

The first direct approaches to purely real time path integration using Monte Carlo were explored by Filinov [59], Doll *et al* [21], and Makri and Miller [19]. The central idea was to introduce a weight function (filter) in the integrand, via certain approximations, that could be used as a Monte Carlo sampling function. The effect of the weight function was to suppress paths whose actions vary rapidly but favor those whose actions are close to being stationary. Hence, a Monte Carlo procedure would sample the regions of "stationary phase" of the integrand, thereby implicitly finding the stationary phase paths and nearby paths, and avoid any explicit work to do so. Makri and Miller had the most viable approach of all as they could tune a parameter to make the method work, at least in principle, from the stationary phase limit up to the exact (unfiltered) integral. However, the full quantum limit could not be reached because the Monte Carlo statistics became prohibitively poor as the dynamics got dominated by many interfering trajectories [19]. Mak and Chandler fused the ideas of stationary phase Monte Carlo (SPMC) and contour distortion in order to sample paths near the true *complex* stationary phase of the integrand [20]. Using importance sampling they could obtain correlation functions for a spin boson problem for times longer than the thermal time, but for systems that possess enormous number of important stationary paths, their method was unable to circumvent the sign problem. Inspired by the observation that grouping paths into sets (either analytically or numerically) before they are sampled always reduces the sign problem, Mak proposed a stochastic resummation method [23] that included an analytic procedure for summing the paths before conventional stochastic sampling began. This idea has evolved into an approach called multilevel block algorithm with cumulant action [60]. While this method, and other methods like a recursive summation over classes of paths [24], have improved the estimates of path integral calculations, they have not managed to give converged results for a general system over long times. More recently, Makri developed a method based on correlating the errors associated with the integrals of positive and negative parts of the integrand by knowing some information about the integrand exactly. This information guided noise reduction (IGNoR) [61] is demonstrated to improve on raw Monte Carlo results and the idea of using it in conjuction with other methods [62], including the one discussed in this thesis, appears promising.

Summarizing, Monte Carlo-based methods provide accurate quantum dynamical information only for short times in general systems or longer times in case of special potentials.

All these methods (with the exception of a direct attack on the complex time correlation function suggested by Thirumalai and Berne) require some kind of approximation or assumptions to be reliable – for example, a valid coordinate transformation (CD) or introduction of artificial weight via an approximation (SPMC) or prior knowledge of some exact information (IGNoR). Thus, the integrand that enters the path integral is substantially modified (or specialized) in the process of making the method feasible. To date, an accurate computation of the time corrleation function over long times for any general quantum system has not been possible with a Monte Carlo procedure. The reason for so much effort over the past few decades on developing Monte Carlo-based approaches for real time dynamics is the fact that the path integral is a multi-dimensional integral, so some kind of importance sampling has to be utilized in its computation. We concur with this reasoning and this dissertation will present a Monte Carlo method for computing complex time correlation functions which does not require any modification of the integrand, involves no approximations beyond the standard short time approximation to the propagator (which can be made arbitrarily accurate by increasing the number of time slices), and is applicable to a general potential or interactions between particles. Most importantly, the method will be shown to accurately compute the correlation function for long times. Since we are dealing with the complex time path integral as it is, the sign problem associated with stochastic summations is still present in its entirety, but (as we shall soon see) much progress can be made in extending the Monte Carlo technique to long time if there is some iterative support.

2.2 Iterative propagation

We learned from the previous section and the last chapter that the Monte Carlo method is unable to get quantum dynamical information of an aribitrary many particle system for long times. We noted before that this is due to the sign problem that all stochastic methods face when they are applied to sample integrands of oscillatory nature. This forces us to look at deterministic methods for the global evaluation of the path integral. The system-specific discrete variable representations [63] and their time dependent extensions offer maximal efficiency for this purpose [64]. But their practical ability, just like conventional quadrature schemes, critically depends of the dimensionality of the integral which involves both the spatial degrees of freedom (d) and the (extra) dimensions coming from time slicing ($N-1$). Even for one degree of freedom the number of dimensions can be huge, if we are interested in many periods of real time, so to evaluate the path integral accurately over long times we need to depart from global integration methods.

The structure of the path integral [see Eq. (1.5)], where only nearest-neighbor factors are coupled, implies that the path integral expression may be evaluated iteratively, performing the integrals one at a time from right to left. This approach appears promising, because it replaces the evaluation of a single many-dimensional integral by multiple fewer-dimensional integrals. Clearly, if M points are employed in this iterative calculation, the result obtained after N iterations is equivalent to having summed the propagator (amplitudes) along M^N paths. Thus, by converting path integration into iterative matrix multiplication, one can circumvent the sign problem and obtain accurate results. Thirumalai and Berne implemented this idea and developed the numerical matrix multiplication (NMM) scheme [13] which gave accurate long time results for the complex time correlation function. Feit and Fleck provided a more general and efficient method, the pseudospectral-split propagator method [14], by which pure real time propagators for long times could be computed. However, the major drawback of both these techniques was their limited applicability to systems with one, two, or possibly three degrees of freedom. This is because the iterative nature of these methods demanded the storage of grid points and propagator matrices, leading to the employment of conventional (uniformly spaced) grids, the size of which (as we discussed in Sec. 1.1) scales exponentially with the number of degrees of freedom.[1] Thus, exact long time computations

[1] The pseudospectral-split propagator method, unlike NMM, avoided the storage of the propagator matrix and hence is the more efficient of the two.

of quantum time correlation functions for a general system were possible via iterative path integral methods but only if the system had very few degrees of freedom. It is important here to point out an iterative influence functional technique [65] developed by Makri to obtain dynamics governed by certain Hamiltonians with many degrees of freedom. For certain special cases, like the harmonic environment, it can be shown [66] that the storage of the propagator for all degrees of freedom can be avoided by pre-integrating out all (bath) but few degrees of freedom (system). The price to be paid is that the resulting path integral expression will be non-local in time, but for a dissipative bath these non-local interactions will be finite in their extent and this could be exploited, as shown by Makri, to obtain exact time evolution of the low-dimensional system via an iterative algorithm. Makri and co-workers used a combination of several tricks, like the usage of physically motivated reference Hamiltonians and the employment of system-specific discrete variable representation, to make the method economical enough to study a host of interesting problems like the electron transfer in solution or systems interacting with coherent laser radiation [67]. Yet, situations which did not allow a decomposition of the Hamiltonian into a system coupled to a harmonic bath, and many-dimensional 'systems' in general, were out of reach with this technique.

Hence, we conclude that although computing the path integral iteratively circumvents the sign problem and enables the knowledge of long time quantum dynamical information, the storage requirements imposed in this approach get prohibitively expensive with system size, just as in the direct Schrödinger wave function propagation on a computer. Efforts that tackle many-dimensional quantum systems, like the influence functional approach, impose significant restrictions on the choice of the Hamiltonian. As a consequence, long time information is only obtained for general systems with very few degrees of freedom. Can iterative propagation be extended to higher integral dimensions of the path integral with assistance from Monte Carlo walks? We will answer this question in the next chapter; for now, let us state the goal and present the outline of this dissertation.

2.3 Goal and outline

Each of the two existing approaches for computing quantum dynamics under Feynman's formulation — Monte Carlo-based computation of the path integral and iterative evaluation of the path integral, possess important advantages and major drawbacks. When the system exhibits a great amount of quantum coherence or long time dynamics is desired, Monte Carlo fails, but it is the only feasible method for carrying out integrations in several dimensions. When the system has more than a few degrees of freedom, iterative propagation fails, but it provides a reliable access to get arbitrarily long time quantum dynamical information. In view of this situation, calculations of time-dependent properties in general polyatomic quantum systems have been possible at short times, in simplified models or via approximate treatments. We also learn two important facts from the discussion so far. First, the path integral is a many-dimensional integral and so Monte Carlo concepts must be employed. And second, the path integral has a built-in iterative structure, the computation along which avoids the sign problem, and so such a structure must be exploited. Keeping these facts in mind, we state the goal of this dissertation and present the outline for the rest of the document.

The goal of this thesis is to develop a stable, numerically exact, fully quantum mechanical methodology that is capable of calculating complex time correlation functions over long times for a general many-body quantum system. In the light of previous two sections, a more specific goal is to show to the reader that the two seemingly different approaches to quantum dynamics via Feynman's path integral formulation — Monte Carlo path integration and iterative propagation, can be combined into one unified framework which shows promise for long-time propagation of many-body systems.

Here is the outline of what is to follow. Chapter 3 will present the method — the idea, the iterative structure, the algorithm; and a simple sampling strategy to illustrate the features of the method will be described. Chapter 4 describes a novel sampling scheme that leads to improved results. The *optimal* sampling strategy for the method is described in chapter 5. Finally, in chapter 6 we present some concluding remarks.

Chapter 3

Iterative Monte Carlo

In this chapter we develop a fully quantum mechanical methodology to calculate complex time correlation functions that does not suffer as severely from the drawbacks of the methods reviewed in the last chapter, and combines their advantages, thus showing promise for long time propagation of many-body systems.

3.1 The basic idea

Consider, for clarity, the discretized path integral representation of the complex time propagator:

$$\begin{aligned}\langle x_N|e^{-it_c\hat{H}/\hbar}|x_0\rangle = \int dx_1 \cdots \int dx_{N-1} \langle x_N|e^{-i\Delta t_c\hat{H}/\hbar}|x_{N-1}\rangle \cdots \\ \times \langle x_2|e^{-i\Delta t_c\hat{H}/\hbar}|x_1\rangle\langle x_1|e^{-i\Delta t_c\hat{H}/\hbar}|x_0\rangle.\end{aligned} \tag{3.1}$$

This can be obtained by letting $t \to t_c$ in (1.5). In this, t_c is a complex time, N is the number of time slices, $\Delta t_c = t_c/N$, and we have used one-dimensional notation for simplicity. Equation (3.1) is a multi-dimensional integral with the number of integration variables equal to $(N-1) \times d$, where $(N-1)$ is the number of path integral discretizations (or "beads") and d is the number of degrees of freedom represented by x. As discussed before, Monte Carlo methods offer the only viable approach for evaluating integrals of such large dimension. However, for $t \neq 0$ the short time propagators are oscillatory (particularly so as the ratio $t/\hbar\beta$ approaches or exceeds unity) and the sign problem in the $(N-1)d$-dimensional integration space is extremely severe. In the conventional PIMC method, the number of samples required to control the statistical error grows exponentially with the number of integration variables, quickly reaching astronomical values. As a result, PIMC calculations are feasible in practice only for small systems and very few time steps.

As we argued earlier in section 2.2, the calculation of a single $(N-1)d$-dimensional integral can be replaced by $(N-1)$ d-dimensional integrals if an iterative evaluation of the path integral is carried out, performing the integrals one at a time from right to left. The resulting schemes highlighted before require storage that scales exponentially with the number d of degrees of freedom, but they avoid the sign problem. Although grid and basis set reduction techniques can often improve the storage problem, the prognosis for application to condensed phase systems remains poor.

We want to exploit the advantages of both these approaches and avoid their drawbacks. Here is the idea. We use Monte Carlo to sample important paths, but rather than attempting to evaluate the desired average from the Monte Carlo random walk, we simply store the sampled path coordinates, generating grids. We use these grids (which consist of judiciously selected points) to evaluate the complex time path integral iteratively. The stepwise evaluation of the path integral circumvents the exponential growth of statistical error with time (or time slices N) and the use of importance sampling in the multi-dimensional grid selection and path summations leads to favorable scaling with the number of degrees of freedom d. Thus, iterative Monte Carlo (IMC) circumvents the exponential scaling of the PIMC error as the real time t is increased, leading to stable results over long propagation time. Moreover, the required grid size for IMC generally grows slowly with number of dimensions d because of the use of importance sampling in the grid selection, compared to the scaling associated with other iterative methods that use quadrature-generated grids.

Here is the idea condensed into one line — *use Monte Carlo to form a grid of points, using the path coordinates visited by the random walker, and propagte the quantity of interest iteratively on this grid.* Now we describe the method in detail.

3.2 The method

We start by recalling the complex time correlation function defined in chapter 1,

$$C_{AB}(t) = \frac{1}{Z} Tr(\hat{A} e^{it_c^* \hat{H}/\hbar} \hat{B} e^{-it_c \hat{H}/\hbar}), \tag{3.2}$$

where $t_c = t - i\hbar\beta/2$ is a complex time that arises by combining the real time with one half of the Boltzmann constant. Inserting the resolution of identity, (3.2) is expressed as a double integral,

$$C_{AB}(t) = \frac{1}{Z} \int \int dx dx' \langle x | \hat{A} e^{it_c^* \hat{H}/\hbar} | x' \rangle \langle x' | \hat{B} e^{-it_c \hat{H}/\hbar} | x \rangle. \tag{3.3}$$

We will use one dimensional notation for clarity, it is understood that x in general represents d degrees of freedom. Assuming that the operators $\hat{A}$ and $\hat{B}$ are local in position, i.e, $\hat{A} = A(\hat{x})$ and $\hat{B} = B(\hat{x})$ (or that they can be expressed in terms of coordinate derivatives, as in the case of the momentum operators), the correlation function takes the following form:

$$C_{AB}(t) = \frac{1}{Z} \int \int dx dx' \langle x | e^{it_c^* \hat{H}/\hbar} | x' \rangle \langle x' | e^{-it_c \hat{H}/\hbar} | x \rangle A(x) B(x'). \tag{3.4}$$

Operators that are not local in position space can be treated by inserting two additional integration variables in (3.4). Notice that the partition function can be also expressed as a double integral by setting $\hat{A} = \hat{B} = \hat{1}$,

$$\begin{aligned} Z &= Tr(e^{-\beta\hat{H}}) = Tr(e^{it_c^* \hat{H}/\hbar} e^{-it_c \hat{H}/\hbar}) = Tr(\hat{1} e^{it_c^* \hat{H}/\hbar} \hat{1} e^{-it_c \hat{H}/\hbar}) \\ &= \int \int dx dx' \langle x | e^{it_c^* \hat{H}/\hbar} | x' \rangle \langle x' | e^{-it_c \hat{H}/\hbar} | x \rangle. \end{aligned} \tag{3.5}$$

Our focus is the forward complex time propagator

$$G(x', x) = \langle x' | e^{-it_c \hat{H}/\hbar} | x \rangle. \tag{3.6}$$

$\langle x|e^{it_c^*\hat{H}/\hbar}|x'\rangle$ is the backward complex time propagator. It is easily seen that the forward and backward complex time propagators are complex conjugates of each other, and therefore,

$$\langle x|e^{it_c^*\hat{H}/\hbar}|x'\rangle = \langle x'|e^{-it_c\hat{H}/\hbar}|x\rangle^* = G(x',x)^*. \tag{3.7}$$

From (3.6) and (3.7), and expressing Z in terms of $G(x',x)$, the expression of correlation function in (3.4) becomes

$$C_{AB}(t) = \frac{\int\int dxdx'|G(x',x)|^2 A(x)B(x')}{\int\int dxdx'|G(x',x)|^2}. \tag{3.8}$$

Thus the evaluation of the forward complex time propagator, $G(x',x)$, determines the correlation function completely via (3.8). From now onwards, unless otherwise stated, when we say “the complex time propagator” or just “the propagator” we mean the forward complex time propagator. Below we focus on its evaluation starting by expressing it as a discretized path integral.

3.2.1 Iterative structure

We begin by splitting the complex time t_c into $2N-1$ slices $\Delta t_c = t_c/(2N-1)$. N is chosen large enough for a convenient approximation of the propagator to be sufficiently accurate over the complex time step Δt_c. Expressing the complex time evolution operator in terms of a product of $2N-1$ short time factors, the propagator $G(x',x) \equiv G(x'_N, x_N) \equiv R_{2N-1}(x'_N, x_N)$ is written in the form of a discretized path integral:

$$\begin{aligned} R_{2N-1}(x'_N, x_N) = \int\cdots\int & dx_1 dx'_1 \cdots dx_{N-1} dx'_{N-1} \langle x'_N|e^{-i\Delta t_c\hat{H}/\hbar}|x'_{N-1}\rangle \cdots \langle x'_2|e^{-i\Delta t_c\hat{H}/\hbar}|x'_1\rangle \\ & \times \langle x'_1|e^{-i\Delta t_c\hat{H}/\hbar}|x_1\rangle \langle x_1|e^{-i\Delta t_c\hat{H}/\hbar}|x_2\rangle \cdots \langle x_{N-1}|e^{-i\Delta t_c\hat{H}/\hbar}|x_N\rangle. \end{aligned} \tag{3.9}$$

A large enough choice of N guarentees that the short time propagator is a known function (either analytically or numerically). Let us call this function s:

$$s(x_k, x_{k-1}; \Delta t_c) = \langle x_k | e^{-iH\Delta t_c/\hbar} | x_{k-1} \rangle. \tag{3.10}$$

It is useful to define propagators for $2k-1$ time steps; as follows:

$$R_{2k-1} \equiv R_{2k-1}(x'_k, x_k; (2k-1)\Delta t_c) = \langle x'_k | e^{-iH((2k-1)\Delta t_c)/\hbar} | x_k \rangle. \tag{3.11}$$

We start the iterative propagation with $R_1(x'_1, x_1; \Delta t_c)$, which is the same as the short-time propagator $s(x'_1, x_1; \Delta t_c)$ and hence is available. The rest of the R_k's can be derived in an iterative fashion as follows. Note that we will suppress the parametric dependence of R_k and s on Δt_c for brevity.

We obtain R_3 from R_1 by using two short-time propagators, $s(x'_2, x'_1)$ and $s(x_1, x_2)$, as

$$R_3(x'_2, x_2) = \int dx'_1 \int dx_1 s(x'_2, x'_1) R_1(x'_1, x_1) s(x_1, x_2). \tag{3.12}$$

Next, combining the short-time propagators with R_3, we get R_5,

$$R_5(x'_3, x_3) = \int dx'_2 \int dx_2 s(x'_3, x'_2) R_3(x'_2, x_2) s(x_2, x_3). \tag{3.13}$$

Repeated use of this process yields R_7, R_9, etc., and eventually R_{2N-1} as illustrated in the

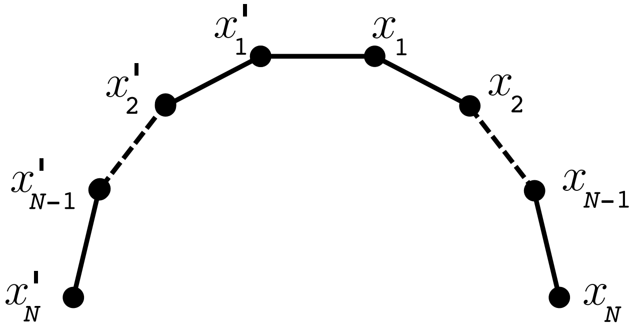

Figure 3.1: Discretization of the path integral for the propagator in the iterative procedure described in section 3.2.1

next block of equations:

$$\begin{aligned}
R_7(x'_4, x_4) &= \int dx'_3 \int dx_3 s(x'_4, x'_3) R_5(x'_3, x_3) s(x_3, x_4) \\
R_9(x'_5, x_5) &= \int dx'_4 \int dx_4 s(x'_5, x'_4) R_7(x'_4, x_4) s(x_4, x_5) \\
&\vdots \\
R_{2N-1}(x'_N, x_N) &= \int dx'_{N-1} \int dx_{N-1} s(x'_N, x'_{N-1}) R_{2N-3}(x'_{N-1}, x_{N-1}) s(x_{N-1}, x_N). \qquad (3.14)
\end{aligned}$$

Thus an iterative structure or loop is seen to emerge,

$$R_{2k+1}(x'_{k+1}, x_{k+1}) = \int dx'_k \int dx_k s(x'_{k+1}, x'_k) R_{2k-1}(x'_k, x_k) s(x_k, x_{k+1}), \qquad k = 1, 2, \ldots, N-1 \qquad (3.15)$$

the computation of which yields the full propagator $G(x', x) \equiv R_{2N-1}(x', x)$ after $N-1$ steps. The series of iterations required to obtain the complex time propagator are shown schematically in the diagram of Figure 1. Conventional iterative path integral schemes discretize the function $R_{2k-1}(x'_k, x_k)$ on a grid and evaluate the required integrals by quadrature, i.e., by some form of matrix-vector multiplication. We plan to perform these operations by the Metropolis Monte Carlo procedure. This procedure is described below.

3.2.2 IMC algorithm

The novel element of IMC is the use of Monte Carlo to evaluate (3.15). Here is the general algorithm that accomplishes this; depending on the particular sampling strategy the details might be slightly different.

1. We perform a random walk in the space of $x'_1, x_1, \ldots, x'_k, x_k, \ldots$ variables to sample paths. Dimensionality of the space in which the walk is performed and the choice of the weight function $\rho(\ldots, x'_k, x_k, \ldots)$ depend on the sampling strategy employed. Moves are accepted or rejected according to the standard Metropolis criteria.

2. The (accepted) coordinates visited by these paths are stored in the form of a two-dimensional grid for each bead pair (x'_k, x_k). A rejected move is dealt with by incrementing the multiplicity of an already stored grid point pair.

3. The probability of selecting the coordinates (x'_k, x_k), in other words, the marginal distribution of the points on the grid corresponding to the k^{th} bead pair, is computed by evaluating the integral

$$\int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1} \int dx'_{k+1} \int dx_{k+1} \cdots \rho(\ldots, x'_k, x_k, \ldots). \qquad (3.16)$$

   The result is stored in the array $P_{2k-1}(x'_k, x_k)$. Notice how the integral is over all the variables except x'_k, x_k, which is precisely the definition of marginal probability. The marginal distributions P will in general be determined as un-normalized functions. Since the unavailable normalization (which is a pure number) will cancel out in the final step, we will keep the discussion simple and omit writing these factors.

4. The iterative process is initialized by setting up an array $R_1(x'_1, x_1) = s(x'_1, x_1; \Delta t_c)$ on the stored (x'_1, x_1) grid.

5. The propagator for the 2nd bead pair, or the three step propagator R_3, is obtained

from R_1 by carrying out the double integral in (3.12). Multiplying and dividing the integrand by $P_1(x_1', x_1)$ and exploiting the fact that the distribution of the stored grid points (x_1', x_1) is given by this function, the Monte Carlo estimate of the double integral becomes

$$R_3(x_2', x_2) = \theta_1 \sum_{x_1', x_1} s(x_2', x_1') \frac{R_1(x_1', x_1)}{P_1(x_1', x_1)} s(x_1, x_2), \tag{3.17}$$

where θ_1 is a normalization constant, which is unavailable, but will cancel out in the final step. Thus, we compute the un-normalized function $\tilde{R}_3$,

$$\tilde{R}_3(x_2', x_2) = \sum_{x_1', x_1} s(x_2', x_1') \frac{R_1(x_1', x_1)}{P_1(x_1', x_1)} s(x_1, x_2), \tag{3.18}$$

where the array $\tilde{R}_3$ set up on the x_2', x_2 grid stores the three step propagator. For each point (x_2', x_2) we sum the summand in (3.18) over the x_1', x_1 points and store the result in the array $\tilde{R}_3$. Since the short complex time propagators s decay exponentially as the difference of the end point coordinates increases, the sum in this equation needs to include only those coordinate pairs (x_1', x_1) for which the absolute value of $s(x_2', x_1')s(x_1, x_2)$ exceeds a certain threshold.[1] This is the second stage where importance sampling ideas reduce the computational effort in IMC, the first stage being the Monte Carlo sampling carried out in step 1.

6. Subsequent iterations are performed in a similar fashion by noting that the propagator for the k^{th} bead pair (or for $2k + 1$ time steps) is obtained from the propagator for $(k-1)^{\text{th}}$ bead pair (or for $2k - 1$ time steps) according to the integrals in the iterative loop of (3.15). The following Monte Carlo sums, corresponding to the double integrals

[1] We will expand on this point later in section 3.3.2

entering in the iterative loop, are carried out,

$$\tilde{R}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{x'_k, x_k} s(x'_{k+1}, x'_k) \frac{\tilde{R}_{2k-1}(x'_k, x_k)}{P_{2k-1}(x'_k, x_k)} s(x_k, x_{k+1}), \qquad k = 1, 2, \ldots, N-1. \tag{3.19}$$

Thus the arrays $\tilde{R}_5, \tilde{R}_7, \ldots$ are generated and stored.

7. It is easy to see that the function obtained after $N-1$ iterations, namely the propagator $\tilde{R}_{2N-1}(x'_N, x_N)$, is proportional to the complex time propagator G. We have,

$$G(x', x) \equiv G(x'_N, x_N) = \theta_1 \theta_2 \ldots \theta_{N-1} \tilde{R}_{2N-1}(x'_N, x_N). \tag{3.20}$$

8. Substituting the above expression for G in the correlation function (3.8) and cancelling out the normalization factors, we get

$$C_{AB}(t) = \frac{\sum_{x'_k, x_k} \frac{|\tilde{R}_{2N-1}(x'_N, x_N)|^2}{P_{2N-1}(x'_N, x_N)} A(x_N) B(x'_N)}{\sum_{x'_k, x_k} \frac{|\tilde{R}_{2N-1}(x'_N, x_N)|^2}{P_{2N-1}(x'_N, x_N)}}. \tag{3.21}$$

Equation (3.21) is the IMC estimate of the complex time correlation function.

3.3 Implementation

Evaluation of integrals via IMC algorithm begins, just like in other MC algorithms, with the choice of sampling function ρ. Performing the Metropolis random walk in some space with the sampling function generates a grid in that space. The distribution of points on this grid is exactly given by ρ, and this distribution is known as the joint probability distribution. In the MC-based methods we reviewed in Chapter 2, the knowledge of joint distribution (or equivalently ρ) is enough to evaluate the MC estimate. However, in IMC, the iterative computation of the path integral forces us to depart from joint probability distribution and acquire the knowledge of the marginal probabilities. For a general ρ knowing the marginal distribution exactly[2] from the grid generated via a joint distribution, is not an easy problem. We will explain why this is so in the first sub-section and also present the reader with a trailer of the sampling strategies that form the bulk of the rest of this thesis.

3.3.1 Sampling strategies

No harm will be done if we rewrite the full path integral expression for the complex time propagator, (3.9), in terms of the functions s:

$$\begin{aligned} R_{2N-1}(x'_N, x_N) = \int \cdots \int dx_1 dx'_1 \cdots dx_{N-1} dx'_{N-1} s(x'_N, x'_{N-1}) \cdots s(x'_2, x'_1) \\ \times\, s(x'_1, x_1) s(x_1, x_2) \cdots s(x_{N-1}, x_N). \end{aligned} \tag{3.22}$$

As is well known, the optimal choice of the sampling function ρ for a MC computation of an integral is the absolute value of the entire integrand that enters the integral. In the case of the above expression the optimal choice is the absolute value of the product of all short time propagators present in the integrand:

$$\rho(\ldots x'_k, x_k, \ldots) = |s(x'_1, x_1)| \prod_{k=1}^{N-1} |s(x'_k, x'_{k+1})||s(x_k, x_{k+1})|. \tag{3.23}$$

If a Metropolis sampling is done with ρ, the marginal distribution for the grid x'_k, x_k is given by (3.16) as

$$\begin{aligned} P_{2k-1}(x'_k, x_k) = \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1} \int dx'_{k+1} \int dx_{k+1} \cdots \; |s(x'_1, x_1)| \\ \times \prod_{k=1}^{N-1} |s(x'_k, x'_{k+1})||s(x_k, x_{k+1})|. \end{aligned} \tag{3.24}$$

To proceed with the IMC prescription we need to know $P(x'_k, x_k)$.[3] The s functions in general, via kinetic energy (recall harmonic springs), couple the grid point (x'_k, x_k) to its

[2] The correct thing to say will be knowing the marginals up to a normalization constant

[3] Note from now on, $P \equiv P(x'_k, x_k) \equiv P_{2k-1}(x'_k, x_k)$

nearest neighbors – (x'_{k-1}, x_{k-1}) and (x'_{k+1}, x_{k+1}). Therefore, an analytical evaluation of this integral for arbitrary potential[4] is not possible. A possible solution could be to compute this integral numerically, say via MC, using a sampling function whose shape is similar to the integrand in (3.24) and whose normalization (which would be a function of (x'_k, x_k) and not just a number) is available. This restricts the kind of systems that one can simulate and moreover carrying out these sums for every x'_k, x_k on each grid will be extremely inefficient. The ideal scenario would be to use the grids already generated via ρ to estimate P. For example, if the (marginal) distribution of the beads $x'_1, x_1, \ldots, x'_{k-1}, x_{k-1}, x'_{k+1}, x_{k+1}, \ldots$ (that is all beads except x'_k, x_k) is given by Ξ then we could write P as

$$P(x'_k, x_k) = \sum_{\ldots, x'_{k-1}, x_{k-1}, x'_{k+1}, x_{k+1}, \ldots} \frac{|s(x'_1, x_1)| \prod_{k=1}^{N-1} |s(x'_k, x'_{k+1})||s(x_k, x_{k+1})|}{\Xi}. \tag{3.25}$$

But Ξ is unavailable, the computation of which demands the knowledge of P, which is what we set out to compute in the first place! Note, we can not make a feasible self-consistent scenario of computing P and Ξ, because Ξ would require a storage of all the variables (except two) which would get prohibitively expensive (as N increases) and clearly we lack a sufficiently accurate choice of a starting function (for arbitrary k) to ignite the self-consistent loop.

Thus, we depart from global (single walk) ways of using MC to compute integrals and introduce a new sampling strategy that involves multiple walks with bead-adapted sampling functions, which enables the knowledge of the marginal distributions from the grids generated to compute the propagator. This strategy is described in the next chapter. We then move on to design the optimal sampling scheme for IMC, using the conventional global sampling function, by exploiting the iterative structure of the *entire* path integral necklace[5] and computing marginals with a self-consistent solution of equations that does not present any storage issues and possesses a sufficiently accurate initializing function to start the pro-

[4] Under the Trotter approximation of s, the potential does not couple x_k (for any k) to other beads

[5] Not just the half necklace shown in Figure 3.1

cedure. This strategy is described in chapter 5. Later in the current chapter we will present the simplest scheme, which compromises on the optimal choice of the sampling function, but leads to an easy determination of the marginal distributions. We tabulate below three sampling strategies we have developed for IMC; the contents of the table will become clear as the reader visits the relevant sections.

SAMPLING STRATEGY	WEIGHT FUNCTION	MARGINALS COMPUTED
Potential-only	Employs only potential terms	Analytically
Bead-adapted	Adapted for each bead pair	Iteratively
Whole-necklace	Conventional PIMC weight	Recursively

3.3.2 Local summations

We have discussed what we call the first stage of importance sampling or MC ideas that are incorporated into IMC. As mentioned in step 5 of the IMC algorithm, the use of importance sampling ideas is not just limited to the grid selection. After the marginals are known, via any of the sampling schemes tabulated above, we perform the local summations of (3.19). These local sums, after $N-1$ iterations, lead to an exponentially large number of paths getting included globally in the computation of the path integral. In this section we will look at how IMC carries out these summations.

For each grid point x'_{k+1}, x_{k+1}, strictly speaking, the local sum is over all x'_k, x_k grid points. However, the short complex time propagators that enter this sum decay exponentially as the the distance between the end point coordinates (for example, distance between x_{k+1} and x_k) increases. This suggests the possibility of getting away by summing over a fraction of x'_k, x_k grid points, thus saving significant computer time. Since we use the Trotter form for short time propagators in all of our sampling strategies, let us rigorously analyze the above argument for that case. It is important to mention here that the sampling strategies presented in the next two chapters are applicable to *any* approximation for the short complex time propagator, and are not limited to the Trotter formula. Similar analysis, like the one we are about to begin, can also be done for those approximations.

From (1.4), which gives the Trotter form for the short real time propagator, we obtain the Trotter formula for the short complex time propagator by letting $t \to t_c$:

$$\begin{aligned} s(x_{k+1}, x_k; \Delta t_c) = \Big(\frac{m}{2\pi i\hbar\Delta t_c}\Big)^{1/2} &\exp\Big(\frac{im}{2\hbar\Delta t_c}\Big(x_{k+1}-x_k\Big)^2\Big) \\ &\times \exp\Big(-\frac{i\Delta t_c}{2\hbar}\Big(V(x_{k+1})+V(x_k)\Big)\Big). \end{aligned} \tag{3.26}$$

Recall $\Delta t_c = t_c/N = (t - i\hbar\beta/2)/N$. Similar expressions are obtained for $s(x'_{k+1}, x'_k)$ and $s(x'_1, x_1)$ by changing the variables of the function s. Our focus here is the absolute value of the first exponential term in (3.26). Taking the product of two such terms coming from $s(x'_{k+1}, x'_k)$ and $s(x_{k+1}, x_k)$ we get

$$g_{\mathrm{w}} = e^{-\frac{m\beta N}{4|t_c|^2}\{(x'_{k+1}-x'_k)^2+(x_{k+1}-x_k)^2\}}. \tag{3.27}$$

As can be seen, the above function is a Gaussian and can be interpreted as a probability (weight) with which (x'_k, x_k) will contribute to the sums in the iterative loop. Writing in terms of g_{w} the k^{th} iteration of (3.19) becomes

$$\tilde{R}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{x'_k, x_k} g_{\mathrm{w}} \ldots \frac{\tilde{R}_{2k-1}(x'_k, x_k)}{P_{2k-1}(x'_k, x_k)}, \tag{3.28}$$

where ... contains other terms from the short time propagator that were not included in g_{w}. From (3.27) we see that if (x'_k, x_k) exceeds a certain large distance (determined by a user-fixed threshold $\mathbb{T}$ that plays the role of a convergence parameter) from (x'_{k+1}, x_{k+1}), which is fixed in the sum of (3.28), then g_{w} is exponentially small. More precisely if the points (x'_k, x_k) that are being summed fall outside the circle (see figure 3.2) described by the

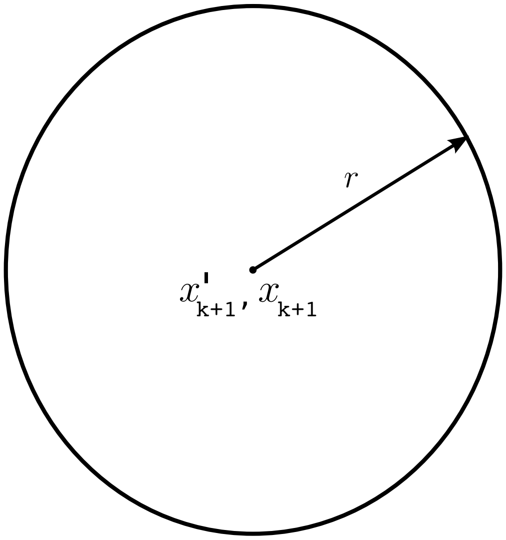

Figure 3.2: The circle described by the equation (3.29)

equation

$$(x'_k - x'_{k+1})^2 + (x_k - x_{k+1})^2 = r^2, \tag{3.29}$$

with x'_{k+1}, x_{k+1} as the center and the radius r derived to be

$$r = \sqrt{\frac{4\mathbb{T}}{mN\beta}\Big(t^2 + \frac{\hbar^2\beta^2}{4}\Big)}\ , \tag{3.30}$$

then we can throw them away. The points that fall inside the circle, we name them *connections*, are included with weight g_{w}. Thus, a dramatic enhancement in the speed of the algorithm results, without any compromise on the desired accuracy of the estimate. To sum up, while the first stage uses a random walk directed by a sampling scheme to select the best possible collection of grid points, the second stage works by inspection. It uses an analytic weight function, inbuilt in the path integral representation of the complex time correlation function, to select a subset of the collection that is statistically significant for employment in the local sum. Thus, the working expression that enters the code is the following iterative cycle

$$\tilde{R}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{\substack{x'_k, x_k \text{ such that} \\ (x'_k - x'_{k+1})^2 + (x_k - x_{k+1})^2 < r^2}} g_{\mathrm{w}} \dots \frac{\tilde{R}_{2k-1}(x'_k, x_k)}{P_{2k-1}(x'_k, x_k)}, \quad k = 1, 2, \dots N-1. \tag{3.31}$$

No matter what sampling strategy is used, after it reveals the marginals, the above set of sums are then performed to obtain the correlation function in the end. Note that this analysis easily generalizes to more dimensions d, where the condition now places the points on the k^{th} grid within a $2d$-dimensional hypersphere centered at the point in question on the $(k+1)^{\mathrm{th}}$ grid. The only effect of adding more degrees of freedom will be an effective decrease in the radius of the hypersphere, which leads to the fall in the number of connections, necessitating increasing the number of grid points to ensure adequate representation of the integrand. The fall in the number of connections affects the stability of the first two sampling schemes as d increases. However, the problem is overcome, for all practical purposes, in the optimal sampling scheme described in chapter 5.

One last important point before we take a small break in the next section. Above, in (3.30) we wrote the expression for the radius of the circle that determines the statistically significant set of points or connections that are included in the local summations. Clearly, the bigger the circle – in other words, the larger the area of the circle – more are the connections to be found. The area of the circle is proportional to the radius squared which means, more grid points will be summed as the radius increases. From (3.30) it is seen that increasing the real time t leads to a larger radius r, which means for longer times more connections will be utilized, proliferating the number of global paths employed to obtain the IMC estimate of the path integral. This is consistent with Feynman's sum over all paths picture, where quantum dynamics at longer times demands a rise in the number of paths. Similarly, lowering of temperature, meaning a rise in β, results in larger r. Lowering the temperature strengthens the quantum effects in the system and this is correctly transferred into the IMC procedure because more connections are summed owing to a greater r. Similar arguments can be made with other physical attributes of the system and the interpretation is found to be consistent with the nature of quantum dynamics. Apart from being physically sound, this in-built self-adjusting feature of IMC is clearly computationally desirable. In extracting short time dynamics for example, only the most relevant paths are summed, minimizing any wasteful operations. Then again, IMC enlarges the necessary connection base, if long time information is to be delivered. For example, in some cases, when $t/\beta > 1$, the entire (x'_k, x_k) grid becomes statistically significant. This characteristic is absent in PIMC where the only way to improve the accuracy is to increase the total number of MC samples, most of which end up being statistically irrelevant forcing an exponential effort to converge. In contrast IMC is designed to select, from a given number of MC samples, paths that will never be selected by PIMC unless the latter uses an exponentially large number of samples. This point is proven more rigorously in section 4.2 of the next chapter.

3.4 Recap and Preview

A lot has been said, derived and argued in the last few sections. Let us at this point take a short break to recap the characteristics of IMC and present a preview of its potential and limitations. This will hopefully help the reader to know what to expect from the method as we will swiftly move, from next section onwards, to IMC sampling strategies and results.

- In IMC, the iterative evaluation of the path integral expression avoids the direct integration of an oscillatory function in $2(2N-1)d$-dimensional space, thus circumventing the exponential growth of statistical error with time that characterizes the real (or complex) time path integral. If an average of M grid points is selected for each $2d$-dimensional grid, the result obtained through (3.21) is equivalent to a summation over $M^{2(2N-1)}$ integrand points.

- At the same time, the use of importance sampling to generate the grid *and* connections implies the storage requirements and the number of operations of IMC will be dramatically smaller than in standard basis set or grid-based methods. For $t = 0$ the grid size required is the typical number of paths necessary to converge the Monte Carlo evaluation of the path integral; thus this grid scales slowly with the number of degrees of freedom d.

- For $0 < t \leq \hbar\beta/2$, the propagator is somewhat oscillatory. If the number of particles is not very large, such a propagator can still be represented adequately on a grid of realistic size. However, the conventional multistep Monte Carlo path integral (PIMC) will typically be out of reach in that case, as phase cancellation becomes dominant at an exponential rate when the number of time steps is increased. On the other hand, the iterative evaluation of the path integral on a Monte Carlo grid will still allow propagation for many time steps. Of course, as the real time is increased further for a fixed temperature, and/or the number of particles becomes large, the propagator will become more oscillatory, and more grid points will be required. If the real time becomes very long as compared to the "thermal time" $t \sim \hbar\beta/2$ and the number of degrees are high enough, such that even the single-step integrals performed in the IMC method become too costly, the results will be harder to get.

- In summary, to the extent that the single-step complex time propagator for a given system is amenable to Monte Carlo sampling, the IMC methodology will allow evaluation of the complex time path integral for many time steps.

3.5 Potential-only sampling

In section 3.3.1 we dismissed an analytical evaluation of the marginal distributions on the grounds that the ideal choice of the sampling function, (3.23), has terms that are coupled to one another, making it hard to analytically do the integral, (3.24), that outputs marginals.

A closer look at the Trotter approximation, (3.26), tells us that this coupling is *only* present in the first exponential, the kinetic energy part of the short time propagator. We already took advantage of this feature during the second stage of IMC when connections are selected. The absolute value of the second exponential in (3.26), or the potential part of the short time propagator, is $e^{-\frac{\beta}{4N}\left(V(x_{k+1})+V(x_k)\right)}$. And it does not couple the beads x_{k+1} and x_k, which suggests that if the grid is selected with a sampling function that is a product of such terms, we can get the marginal distributions analytically. Specifically, one can show after doing some algebra that the 'optimal' sampling function, if one is forced to include *only* potential terms of the short time propagator, is

$$\rho = \prod_{k=1}^{N} e^{-\frac{\beta}{2N}V(x'_k)} e^{-\frac{\beta}{2N}V(x_k)}. \tag{3.32}$$

The marginals, computed merely by inspection using (3.16) with the above expression for ρ as the integrand, are found to be

$$\begin{aligned} P_{2k-1}(x'_k, x_k) &= \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1} \int dx'_{k+1} \int dx_{k+1} \cdots \rho(\ldots, x'_k, x_k, \ldots) \\ &= \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1} \int dx'_{k+1} \int dx_{k+1} \cdots \prod_{k=1}^{N} e^{-\frac{\beta}{2N}V(x'_k)} e^{-\frac{\beta}{2N}V(x_k)} \\ &= e^{-\frac{\beta}{2N}V(x'_k)} e^{-\frac{\beta}{2N}V(x_k)} \times \text{constant}. \end{aligned} \tag{3.33}$$

Since the constant will cancel eventually in the last step of the IMC algorithm, we won't worry about it. Note that if ρ were to have terms that coupled the beads x'_k, x_x to their neighbors, then the constant will be replaced by a function of x'_k, x_k, which in general will be unknown. Thus, quite easily, marginal distributions of all the grids are available within the potential-only sampling. Of course, this is possible at a rather huge cost. Firstly, notice that P is independent of t! Implying no matter at what time the dynamics is simulated, a similar grid is generated, which is clearly undesirable as the integrand that enters the sums in the iterative cycle changes considerably with time t and this should be reflected in the grid selection. Second, the grid is always a high temperature distribution — β/N corresponds to higher temerature than β. This means a lowering of the temperature (increasing β) will not be reflected (because of increasing N to keep a small value for β/N), implying the distribution will be classical like at all temperatures. Finally, we have already mentioned the stability issues that will come up as the connections fall with increasing d. The first two points will be addressed by the sampling scheme presented in the next chapter. And all the three issues mentioned here will be taken care of by the optimal whole-necklace sampling.

The choice of potential-only sampling is adequate for the purpose of illustrating the features of IMC, mostly because the second stage of incorportating importance sampling ideas is still intact, and the iterative propagation will make up for the poor selection of grid points. In the next section we will show the first IMC results obtained from potential-only sampling. We will focus on one-dimensional problems, applications to multi-dimensional systems will be taken up in the next two chapters via much improved sampling strategies.

3.6 First Results

Once the marginal distributions are known, they are to be plugged into the iterative loop given by (3.31) and after $N-1$ iterations the complex time correlation function is outputed. We present in this section calculations of the complex time position autocorrelation function, which is obtained by setting $\hat{A} = \hat{B} = \hat{x}$. Figure 3.3 shows this correlation function for a one dimensional harmonic oscillator of unit mass and frequency. The results of the IMC methodology with $2N-1 = 7$ (i.e, 14 path integral beads[6]) are compared to numerically exact results, and also to the results of a conventional Metropolis Monte Carlo (PIMC) evaluation of the complex time path integral, where the number of Monte Carlo samples was adjusted to use the same number of operations employed in IMC. The IMC calculation was

[6] The number of beads is twice the number of slices

done with just 1000 grid points. As it is seen very clearly in the figure, the statistical error of the direct Monte Carlo calculation grows exponentially with time and the results become meaningless beyond $t \sim \hbar\beta$, in striking contrast to the results obtained with IMC with the same numerical effort, which track the exact results at all times.

Figure (3.4) shows the same, but for a lower temperature, $\beta = 3$. Again, PIMC starts to fail around $t = 3$ while IMC is consistently tracking the exact results. The error in IMC can be further reduced with a small increase in the number of points. The fact that with just ∼1000 points the correlation function can be reliably computed for long times clearly indicates that in IMC an exponentially large number of paths are included over all, without a similar increase in computational effort. If the number of grid points were to scale exponentially with time in PIMC, IMC-like results would be obtained, but clearly we are looking at an astronomical number of computations in that case. Next in Figure (3.5) we compute the same correlation function for a strongly anharmonic oscillator described by the potential $V(x) = \frac{1}{2}x^2 + \frac{1}{5}x^4$. This calculation is done at $\beta = 0.5$. Again, PIMC gets meaningless beyond $t \approx \beta$, while IMC with the same numerical effort maintains accuracy even at times for which $t/\hbar\beta \approx 15$. This suggests the efficiency of IMC is not affected by the kind of potential the particle feels. Thus, it is seen that IMC avoids the exponential growth of statistical error, so characteristic of MC, as the real time is increased.

In the next chapter we will look at many-particle quantum systems after we describe an improved sampling strategy.

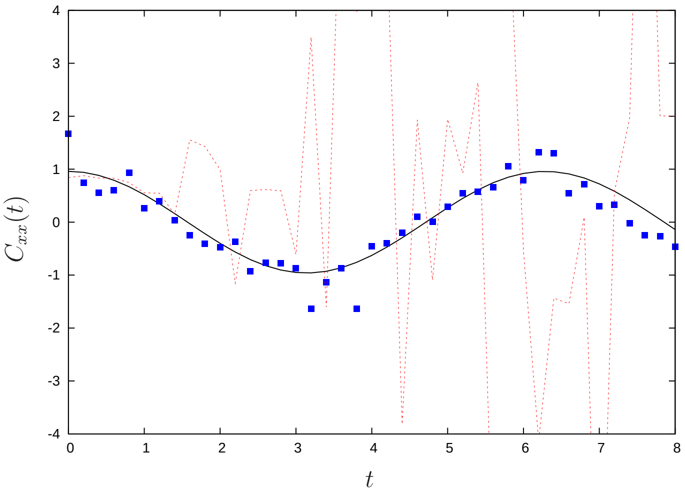

Figure 3.3: Complex time position autocorrelation function for a one dimensional harmonic oscillator with unit mass and frequency at $\beta = 1$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1000 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation.

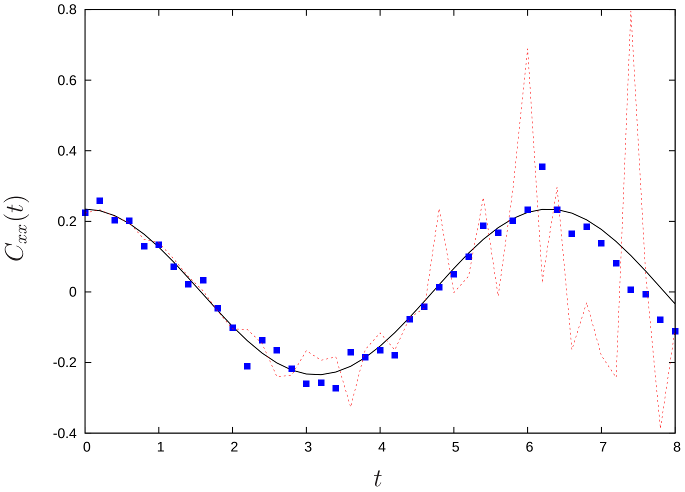

Figure 3.4: Complex time position autocorrelation function for a one dimensional harmonic oscillator with unit mass and frequency at $\beta = 3$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1500 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation.

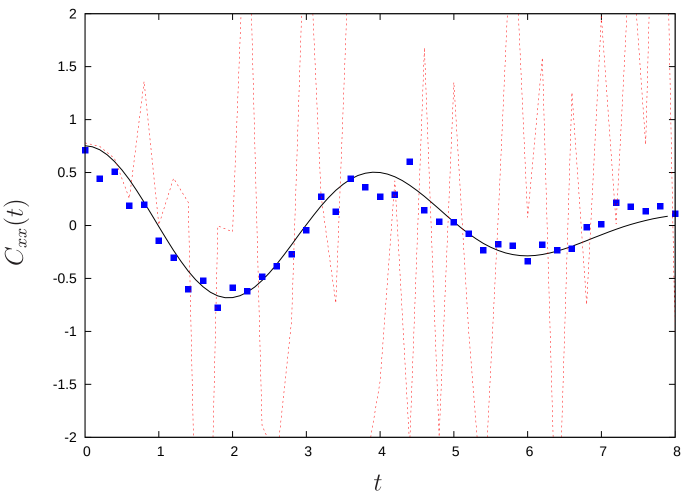

Figure 3.5: Complex time position autocorrelation function for a one dimensional anharmonic oscillator, $V(x) = \frac{1}{2}x^2 + \frac{1}{5}x^4$, with unit mass at $\beta = 0.5$. Solid black line: Exact results. Blue squares: Results obtained with IMC using 14 beads and 1500 points. Red dashed line: Results obtained through PIMC for 14 beads, with the number of samples adjusted to have the same number of operations as in the IMC calculation.

Chapter 4

IMC with bead-adapted sampling

In the previous chapter we used the potential part of the short complex time propagator as the sampling function, generating a grid of points distributed similar to the classical Boltzmann factor at the high temperature corresponding to the imaginary part of the time step. This choice was adequate for the purpose of illustrating the features of IMC. It is well known, however, that the Monte Carlo procedure is most efficient when the sampling function is as close as possible to the (absolute value of) the entire integrand. In this chapter we present an optimized IMC procedure where the sampling function used in the evaluation of (3.15) satisfies this requirement. Further, the scheme we describe here is quite general and does not rely on the use of the Trotter approximation to construct the short time propagator. Thus, the present IMC method can be used in conjunction with improved propagators [68] that allow larger time steps, such as the pair-product form in the case of neat fluids [35].

4.1 Bead-adapted sampling strategy

The main part of the integrand in (3.15) is, $R_{2k-1}(x'_k, x_k)$, the propagator for $2k-1$ complex time steps. For $k = 1$, this is the high-temperature/short time propagator, which is sharply peaked about $|x_1 - x'_1|$ (because of the kinetic energy term) and extended along the $x_1 = x'_1$ axis (because of the small value of the exponent in the potential part). As k increases, the propagator $R_{2k-1}(x'_k, x_k)$ broadens along the direction $|x'_k - x_k|$ but becomes less extended along the x'_k and x_k coordinates. Thus, it is clear that the shape of the integrand varies drastically with k. This behavior should be reflected in the range of the grid on which

$R_{2k-1}(x'_k, x_k)$ is stored, and therefore should be incorporated in the sampling function.

4.1.1 Sampling function

As we argued above, we would like to generate points distributed roughly as the envelope of the complex time propagator to be used at each iteration. This can be achieved through a Monte Carlo random walk that uses $|R_{2k-1}(x'_k, x_k)|$ as the sampling function. However, this idea appears impractical, because the propagator is not available analytically, except within a short time approximation that should be inadequate beyond $k = 1$.

In order to address this issue, our sampling function is adapted to each path integral bead. To generate the grid for the k^{th} bead pair, we perform a Monte Carlo random walk in the space of $x_1, x'_1, \ldots, x_k, x'_k$, accepting or rejecting moves according to the weight function

$$\rho_k(x'_1, x_1 \ldots x'_k, x_k) = \left|\langle x'_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\right| \prod_{l=1}^{k-1} \left|\langle x'_l|e^{-i\Delta t_c \hat{H}/\hbar}|x'_{l+1}\rangle\right|\left|\langle x_l|e^{-i\Delta t_c \hat{H}/\hbar}|x_{l+1}\rangle\right|. \quad (4.1)$$

We have reverted back to the bra-ket notation for the short time propagator as it will be useful later; recall $s(x_{k+1}, x_k; \Delta t_c) = \left|\langle x_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x_k\rangle\right|$. The coordinates of accepted moves for the bead pair (x'_k, x_k) are stored, forming the two-dimensional grid for these beads. The other bead pairs $\{(x'_1, x_1), \ldots, (x'_{k-1}, x_{k-1})\}$ act as auxillary variables and are discarded. Just as before, in the IMC algorithm prescribed in 3.2.2, a rejected move is dealt with by incrementing the multiplicity of an already stored grid point pair.

The probability of selecting the coordinates (x'_k, x_k) for the k^{th} bead pair is

$$\begin{aligned} P_{2k-1}(x'_k, x_k) &= \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1}\ \rho_k(x'_1, x_1, \ldots, x'_k, x_k) \\ &= \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1}\ \left|\langle x'_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\right| \\ &\qquad \times \prod_{l=1}^{k-1} \left|\langle x'_l|e^{-i\Delta t_c \hat{H}/\hbar}|x'_{l+1}\rangle\right|\left|\langle x_l|e^{-i\Delta t_c \hat{H}/\hbar}|x_{l+1}\rangle\right|. \end{aligned} \quad (4.2)$$

At this point we will define the short real time step (Δt) and short imaginary time step ($\Delta\beta$) as $\Delta t_c = t_c/(2N-1) \equiv \Delta t - i\hbar\Delta\beta$. In the special case of purely imaginary time ($t = 0$), the sampling function in (4.1) is simply the Boltzmann weight of the path integral segment with endpoints (x'_k, x_k), which contains $2k-1$ steps, such that

$$P_{2k-1}(x'_k, x_k) = \int dx'_1 \int dx'_1 \cdots \int dx'_{k-1} \int dx_{k-1}\ \langle x'_1|e^{-\Delta\beta\hat{H}}|x_1\rangle \\ \times \prod_{l=1}^{k-1} \langle x'_l|e^{-\Delta\beta\hat{H}}|x'_{l+1}\rangle\langle x_l|e^{-\Delta\beta\hat{H}}|x_{l+1}\rangle. \tag{4.3}$$

Removing the resolution of identity multiple times, this becomes

$$P_{2k-1}(x'_k, x_k) = \langle x'_k|e^{-(2k-1)\Delta\beta\hat{H}}|x_k\rangle = R_{2k-1}(x'_k, x_k). \tag{4.4}$$

Thus, in the case of zero real time, the sampled points for the k^{th} bead pair have precisely the desired distribution. While this is not the case for $t > 0$, the distribution in (4.2) is still similar to (and only somewhat broader than) $|R_{2k-1}(x'_k, x_k)|$, thus ideally suited to our purpose. The close similarity of the distribution $P_{2k-1}(x'_k, x_k)$ (resulting from the bead-adapted sampling procedure) to the ideal distribution $|R_{2k-1}(x'_k, x_k)|$ of the k^{th} grid point pair is illustrated in Figures 4.1, 4.2, and 4.3.[1]

4.1.2 Computing marginal distributions

We start by the definition of the marginal distribution, (4.2), within the bead-adapted sampling. For $k = 1$, i.e. the first grid or bead pair, the grid is sampled with $\rho_1(x'_1, x_1)$. Since there are no other variables in the function ρ_1, this itself is the marginal distribution. Confirming this with the expression in (4.2), for $k = 1$ there are no integrals to evaluate, we

[1] The figures on the left refer to the parameters (a), the figures on the right to (b). We will adopt this convention throughout.

have

$$P_1(x_1', x_1) = \rho_1(x_1', x_1) = \left|\langle x_1'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\right|. \tag{4.5}$$

For the second bead pair, $k = 2$, we start with the definition and find

$$\begin{aligned} P_3(x_2', x_2) &= \int dx_1' \int dx_1 \left|\langle x_2'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1'\rangle\right|\left|\langle x_1'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\right|\left|\langle x_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_2\rangle\right| \\ &= \int dx_1' \int dx_1 \left|\langle x_2'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1'\rangle\right| \, P_1(x_1', x_1) \, \left|\langle x_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_2\rangle\right|. \end{aligned} \tag{4.6}$$

Can we do this double integral? It turns out we can, the strategy was designed for precisely this purpose. This is the double integral over x_1', x_1 coordinates and we just showed in (4.5) that their distribution is given by $P_1(x_1', x_1) = \rho_1$. Therefore, we convert the integral into a Monte Carlo sum to get

$$\begin{aligned} P_3(x_2', x_2) &= \theta_1 \sum_{x_1', x_1} \frac{\left|\langle x_2'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1'\rangle\right| \, P_1(x_1', x_1) \, \left|\langle x_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_2\rangle\right|}{P_1(x_1', x_1)} \\ &= \theta_1 \sum_{x_1', x_1} \left|\langle x_2'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1'\rangle\right|\left|\langle x_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_2\rangle\right|, \end{aligned} \tag{4.7}$$

where θ_1 is some normalization constant that will eventually cancel out. Thus the marginal distribution of the second grid is known. Working from the definition again, the distribution of x_3', x_3 grid points is given by

$$\begin{aligned} P_5(x_3', x_3) = \int dx_1' \int dx_1 \int dx_2' \int dx_2 &\left|\langle x_3'|e^{-i\Delta t_c \hat{H}/\hbar}|x_2'\rangle\right|\left|\langle x_2'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1'\rangle\right|\Big| \\ &\times \langle x_1'|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\Big|\left|\langle x_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_2\rangle\right|\left|\langle x_2|e^{-i\Delta t_c \hat{H}/\hbar}|x_3\rangle\right|. \end{aligned} \tag{4.8}$$

In the above expression, just by inspection, if we carry out the integral over x_1', x_1 first, we get a familiar function — $P_3(x_2', x_2)$. Thus, we can write (4.8) as

$$P_5(x_3', x_3) = \int dx_2' \int dx_2 \left|\langle x_3'|e^{-i\Delta t_c \hat{H}/\hbar}|x_2'\rangle\right| \, P_3(x_2', x_2) \, \left|\langle x_2|e^{-i\Delta t_c \hat{H}/\hbar}|x_3\rangle\right|. \tag{4.9}$$

Clearly we can do this double integral by Monte Carlo too, as we are now aware of the distribution of x'_2, x_2 grid points. Thus, an iterative structure very reminiscent of the structure derived in section 3.2.1 emerges. The marginal distribution of the grid points on the $(k+1)^{\text{th}}$ grid is expressed in terms of the marginal distribution of grid points on the k^{th} as:

$$P_{2k+1}(x'_{k+1}, x_{k+1}) = \int dx'_k \int dx_k \big|\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\big| \; P_{2k-1}(x'_k, x_k) \; \big|\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle\big|. \tag{4.10}$$

By replacing P with R and removing the absolute values on the short time propagators in the above expression, the reader can convince herself that the iterative structure (3.15) of last chapter is recovered. This means that, just like $R_{2k-1}(x'_k, x_k)$, the marginal distributions $P_{2k-1}(x'_k, x_k)$ are also subject to iterative propagation and can be computed with the same prescription with which the propagators were computed. Specifically, we get the following MC estimate for the marginal distribution of the $(k+1)^{\text{th}}$ grid,

$$\begin{aligned} P_{2k+1}(x'_{k+1}, x_{k+1}) &= \theta_k \sum_{x'_k, x_k} \big|\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\big| \frac{P_{2k-1}(x'_k, x_k)}{P_{2k-1}(x'_k, x_k)} \big|\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle\big| \\ &= \theta_k \sum_{x'_k, x_k} \big|\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\big| \big|\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle\big|. \end{aligned} \tag{4.11}$$

Again, in practice we compute the un-normalized $\tilde{P}_{2k+1}$,

$$\tilde{P}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{x'_k, x_k} \big|\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\big| \big|\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle\big|. \tag{4.12}$$

In terms of the these un-normalized marginals, the IMC estimate from (3.19) for the propagated R function is obtained as

$$\tilde{R}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{x'_k, x_k} \langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle \frac{\tilde{R}_{2k-1}(x'_k, x_k)}{\tilde{P}_{2k-1}(x'_k, x_k)} \langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle. \tag{4.13}$$

Equations (4.12) and (4.13) represent the summations that enter the iterative loop. As the number of iterations increase we get all the marginals and all the propagators up to the final propagator $\tilde{R}_{2N-1}(x'_N, x_N)$ and final marginal $\tilde{P}_{2N-1}(x'_N, x_N)$, which are then used to compute the complex time correlation function. To ignite the iterative cycle the intializing propagator function $R_1(x'_1, x_1)$ is known via the short time approximation $s \equiv \langle x'_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle$ and the intializing marginal distribution is known, as emphasized earlier, by *design*.

4.2 $t = 0$ vs $t > 0$

Consider the case of zero time, i.e., $t_c = -i\hbar\beta/2$. In this case the iterative structures, (4.10) and (3.15), corresponding to the marginals and propagators respectively are not similar, they are *exactly* the same! This is because for zero real time the absolute values of the short time propagator is the same as the short time propagator, there is no phase. Thus $\tilde{R}_{2k-1}(x'_k, x_k) = \tilde{P}_{2k-1}(x'_k, x_k)$ and (4.13) becomes

$$\tilde{R}_{2k+1}(x'_{k+1}, x_{k+1}) = \tilde{P}_{2k+1}(x'_{k+1}, x_{k+1}) = \sum_{x'_k, x_k} \left|\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\right|\left|\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle\right|. \tag{4.14}$$

In this special case, $\tilde{R}_{2k+1}$ does not depend on $\tilde{R}_{2k-1}$; thus, the accuracy of this imaginary time ($t = 0$) propagator at each iteration does not depend on the precision with which the previous step was evaluated, but only on the number of grid points and their placement. By the same argument, only made iteratively, in the calculation of the correlation function the sums evaluated in earlier iterations cancel out and thus do not affect the final result. Thus, one concludes that the IMC estimate of the zero-time result is in this case *identical* to that of PIMC performed via a standard Metropolis random walk with ρ_N as the sampling function. The additional summations in the IMC steps are redundant in the case of pure imaginary time.

The situation could not be more different in real-time calculations. When $t \neq 0$, the sum in (4.13) contains phase factors, and $\tilde{R}_{2k-1}(x'_k, x_k) \neq \tilde{P}_{2k-1}(x'_k, x_k)$. Defining

$$f_k(x'_{k+1}, x_{k+1}, x'_k, x_k) = \frac{\langle x'_{k+1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_k\rangle\langle x_k|e^{-i\Delta t_c \hat{H}/\hbar}|x_{k+1}\rangle}{\tilde{P}_{2k+1}(x'_k, x_k)}, \tag{4.15}$$

iteration of (4.13) leads to the result

$$\tilde{R}_{2N-1}(x'_N, x_N) = \sum_{x'_{N-1}, x_{N-1}} \cdots \sum_{x'_1, x_1} f_{N-1}(x'_N, x_N, x'_{N-1}, x_{N-1}) \ldots f_1(x'_2, x_2, x'_1, x_1) R_1(x'_1, x_1). \tag{4.16}$$

If M points are used for the k^{th} grid pair, (4.16) shows that the $N-1$ iterative steps performed to obtain $\tilde{R}_N$ amount to a total of M^{N-1} evaluations of the integrand. Thus, the number of terms included in the IMC estimate of the propagator grows exponentially with the number of time slices (although the computational effort through the iterative process scales linearly). This exponential increase in the number of integrand evaluations counteracts the exponential growth of statistical error due to phase cancellation. Thus, the precision of the IMC result will tend to remain constant as the number of iterations is increased, in sharp contrast to the conventional Monte Carlo estimate, which is characterized by exponential proliferation of statistical error. We should mention here that the argument just made is valid for a fixed complex time step. If the Δt_c is changed, say as a result of increasing t then we will require more IMC grid points to maintain the same statistical error. Also if the oscillatory character of the short time propagator is increased as a result of added degrees of freedom, the number of points M must grow to account for the convergence of single step integrals.

4.3 Results

As promised in the last chapter, we present results for the position correlation function of a system of d uncoupled harmonic oscillators described by the Hamiltonian

$$H = \sum_{i=1}^{d} \frac{1}{2}(p_i^2 + \omega_i^2 x_i^2). \tag{4.17}$$

Exact results were generated analytically. The IMC results were compared against those generated by the conventional path integral Monte Carlo method with the number of samples set equal to the total number of operations performed in the IMC calculation. The following parameters were used:

(i) $d = 1,\ \omega_1 \equiv \omega = 1$ (Figures 4.4(a), 4.4(b) and 4.5),

(ii) $d = 4,\ \omega_i = 0.81, 0.92, 1.00, 1.16$ (Figure 4.6),

(iii) $d = 7,\ \omega_i = 0.81, 0.92, 1.00, 1.07, 1.16, 1.22, 1.32$ (Figure 4.7), and

(iv) $\omega_i = 1$ with $d = 1, \ldots, 13$ (Figures 4.8(a) and 4.8(b)).

All calculations presented below were performed with the Trotter discretization of the short complex time propagator (3.26).

Figures 4.4(a) and 4.4(b) illustrates the performance of IMC at each iteration with a (fixed) complex time step $\Delta t_c = \Delta t - i\hbar\Delta\beta$. Since a graph of the propagated function $\tilde{R}_{2k-1}$ would not be very informative, we use this function in the expression for the correlation function, (3.21), reporting the value of the symmetrized correlation function at the complex time $(2k-1)\Delta t_c$, i.e., at the inverse temperature $2(2k-1)\Delta\beta$ and time $(2k-1)\Delta t$,

$$C_{xx}((2k-1)\Delta t_c) = \frac{\int dx_k' \int dx_k |R_k(x_k', x_k)|^2 x_k' x_k}{\int dx_k' \int dx_k |R_k(x_k', x_k)|^2}. \tag{4.18}$$

Note that the temperature decreases with each iteration. Results are shown in Figures 4.4(a) and 4.4(b) for $\hbar\omega\Delta\beta = 0.25$. When $\omega\Delta t = 0$ there is no sign problem, and according to the discussion following (4.14) both IMC and PIMC should yield comparable results. Figure 4.4(a) confirms this behavior. However, for the value $\omega\Delta t = 0.5$ the real and imaginary time parameters are comparable, leading to severe phase cancellation in the multi-dimensional path integral. Indeed, as seen in Figure 4.4(b), the statistical error of the conventional PIMC calculation grows exponentially with the number of path integral beads, while the IMC results with 5000 grid points faithfully track the exact values with statistical deviations of nearly constant magnitude.

Figure 4.5 shows the symmetrized correlation function as a function of real time for a one dimensional harmonic oscillator at a fixed temperature corresponding to $\hbar\omega\beta = 1$. The number of beads varies between 6 and 54 ($2N - 1 = 3$–27) in order to keep the real-time step Δt within the range 0.3–0.5. Since the temperature is fixed, the imaginary time step decreases as N and t are increased. For the longest times shown in the figure, $\Delta t/\Delta\beta = 30$. Thus, the oscillatory character of the integrand increases with time. For this reason we set the grid size according to the relation $2500 + 800N$, increasing it linearly with the number of beads. PIMC results are also shown for comparison. For each run, the number of PIMC samples was adjusted to yield the same number of operations employed in the IMC calculation. As can be seen in Figure 4.5, the IMC results are stable and accurate even when t exceeds the thermal time $t \sim \hbar\beta/2$ by an order of magnitude. By comparison, the PIMC results become meaningless after the first quarter of the oscillation period.

Figures 4.6 and 4.7 present similar results for 4- and 7-dimensional systems respectively with $\beta = 5$. For $d = 4$ (Figure 4.6), the IMC calculation was performed with 14–30 beads ($2N - 1 = 7$–15). The grid size was increased slightly faster than linearly with the number of beads, following the relation $-40000 + 3500N^{4/3}$. In the case of $d = 7$ (Figure 4.7) we used 14 beads ($2N - 1 = 7$) for all calculations. The number of points was increased linearly with time, following the relation $60000 + 20000t$. Again, the statistical error of the conventional PIMC procedure becomes very large ($\geq 100\%$) as soon as $t > \hbar\beta/2$ and continues to grow exponentially as the real time is increased. In striking contrast to that behavior, the IMC procedure led to accurate results with small statistical uncertainty.

Figure 4.8(a) investigates the performance of IMC for calculations with several particles. Shown in this figure is the value of the correlation function at fixed $t = 0$ as a function of the number d of degrees of freedom for $\beta = 5$. Since the real time is zero, these results correspond to an equilibrium calculation where there is no sign problem. Once again, the PIMC method is extremely efficient in this case; in line with the discussion of the previous section, the IMC and PIMC results are comparable in accuracy. Figure 4.8(b) shows analogous results for $t = 4$ (which exceeds significantly the characteristic thermal time). This calculation employed $10000d$ grid point pairs for $d \leq 9$ and $10000d + 100000(d - 9)$ for $d \geq 10$. The increase of the number of grid points for 10 or more degrees of freedom was necessary to maintain small statistical error in IMC. There are two reasons for this: (i) As discussed at the end of the previous section, the integrand is oscillatory, thus there is phase cancellation in IMC (although the resulting sign problem is far less severe compared to PIMC); (ii) When the number of degrees of freedom is large, distances among grid points become large on average, and the number of connections drops below the acceptable threshold, unless the number of grid points is increased. With the given parameters the error bars associated with the PIMC results exceed 100% for $d > 4$, while the IMC results remain close to the exact values.

4.4 Summary

The improved IMC scheme described in this chapter employs bead-adapted sampling with a weight function that closely resembles the absolute value of the entire integrand corresponding to each step in the iterative process. This way the selected grid points are guaranteed to span the most important regions of each integration variable, leading to high efficiency. The calculations reported in the previous section revealed a nearly constant statistical error with increasing number of path integral beads using grids of modest size. We found that the number M of grid points must be increased somewhat faster than linearly with the number d of degrees of freedom to guarantee stability. The present scheme includes kinetic energy terms and this gradually decreases grid spread in each iteration. As compactness of the IMC grid leads to avoidance of unnecessary oscillatory components, this procedure performs well at later iterations. However, grid sampling during the first few time steps corresponds to a high-temperature distribution, and thus is much wider than necessary, decreasing efficiency. We describe in the next chapter an IMC scheme that uses the *optimal* grid, composed exclusively of coordinates visited by the entire path integral necklace.

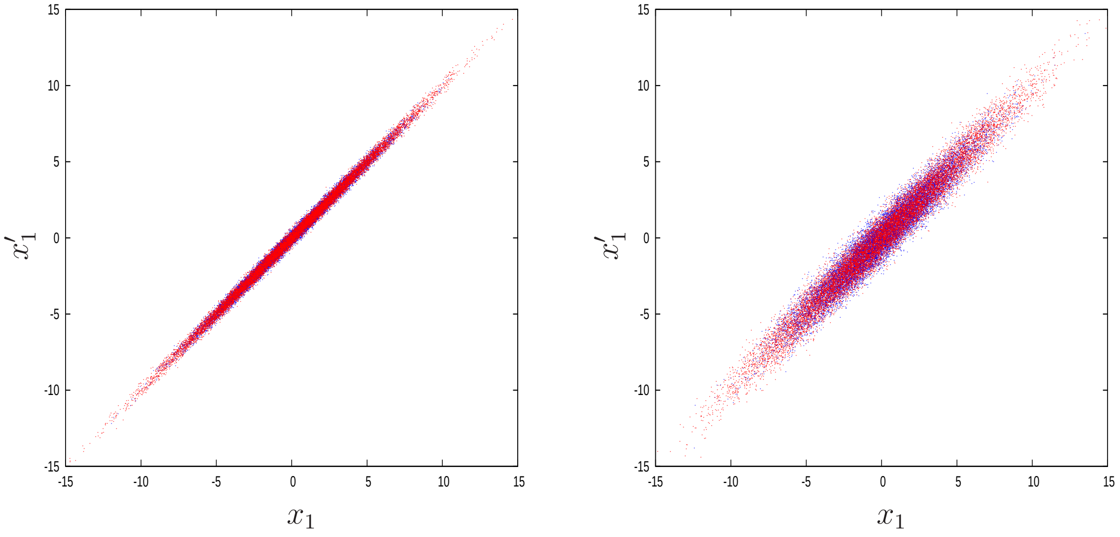

Figure 4.1: Grid $P_{2k-1}(x_k', x_k)$ generated using the bead-adjusted sampling procedure (blue dots), compared to the exact absolute value of the propagator $|R_{2k-1}(x_k', x_k)|$ (red dots), for the k^{th} bead pair, in the case of a harmonic oscillator for $k = 1$. (a) $\hbar\omega\Delta t_c = -0.5i$ and (b) $\hbar\omega\Delta t_c = 3 - i0.5$.

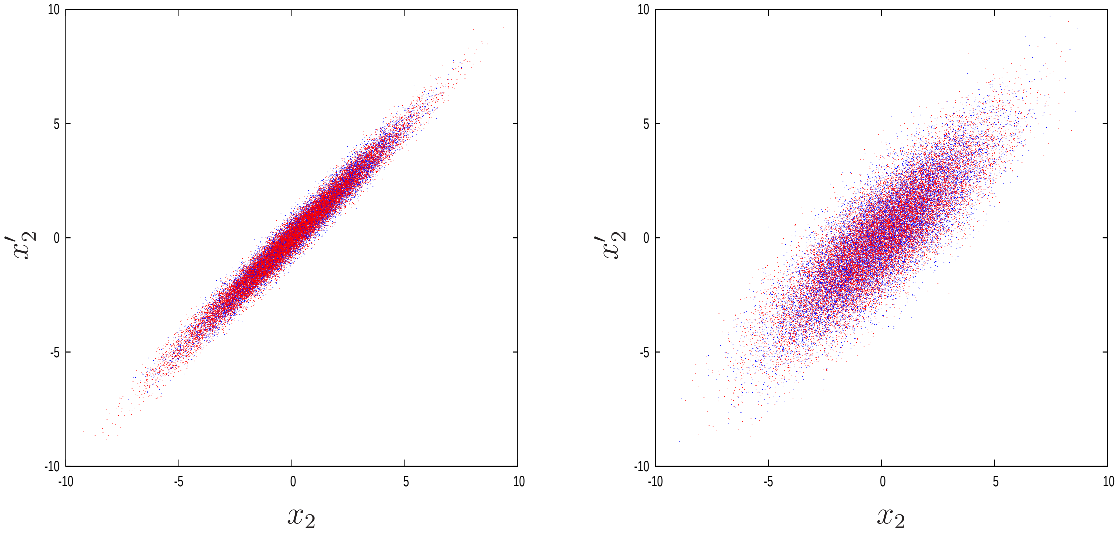

Figure 4.2: Same as in Figure 4.1 but for $k = 2$.

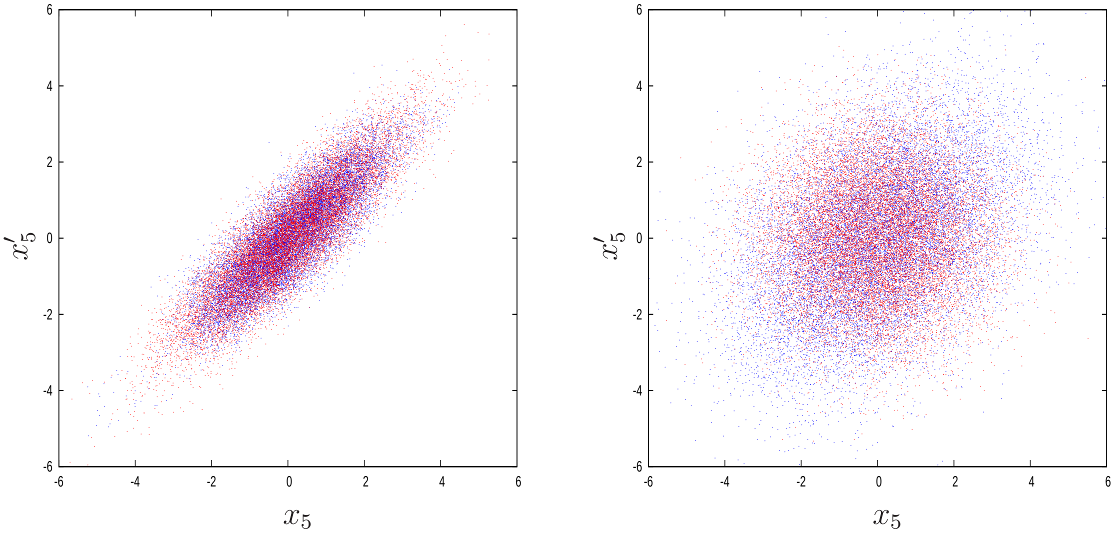

Figure 4.3: Same as in Figure 4.1 but for $k = 5$.

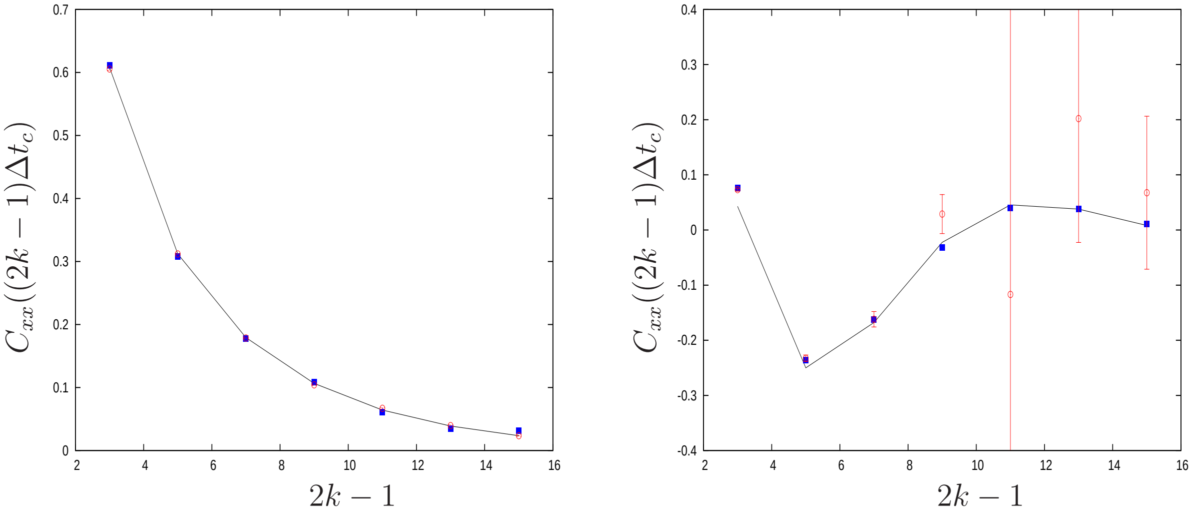

Figure 4.4: Complex time position autocorrelation function for a harmonic oscillator with unit mass with fixed Δt_c. The solid line shows exact results. The blue squares show results obtained with the IMC method with 5000 grid points. The red hollow circles show PIMC results, with the number of samples adjusted to have the same number of operations as IMC. (a) $\hbar\omega\Delta\beta = 0.25, \omega\Delta t = 0$ (b) $\hbar\omega\Delta\beta = 0.25, \omega\Delta t = 0.5$.

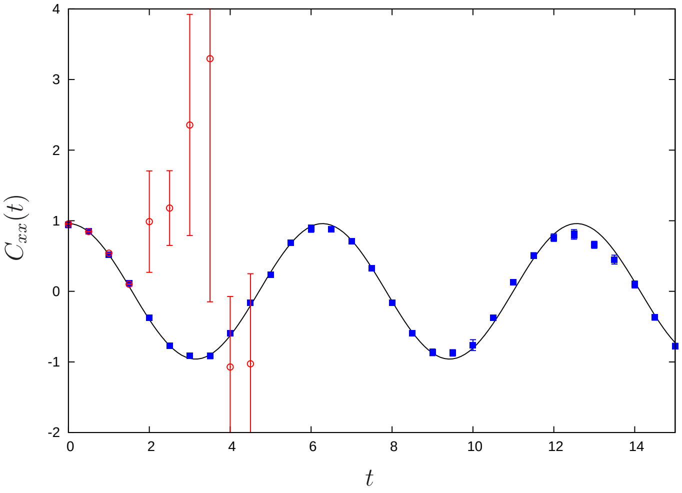

Figure 4.5: Complex time position autocorrelation function for a one-dimensional harmonic oscillator with $\hbar\omega\beta = 1$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained from a conventional Metropolis Monte Carlo evaluation of the complex time path integral expression, with the number of samples adjusted to have the same number of operations as IMC.

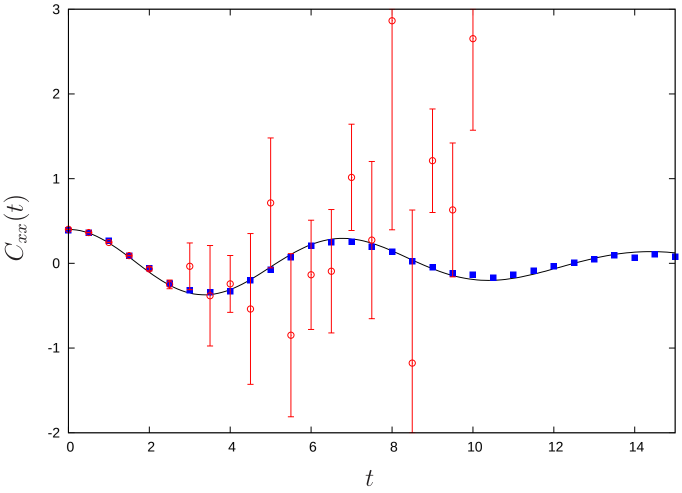

Figure 4.6: Complex time position autocorrelation function for a 4-dimensional multi-frequency harmonic oscillator with unit mass at $\beta = 5$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained conventional PIMC procedure, with the number of samples adjusted to have the same number of operations as IMC. See the text for additional details.

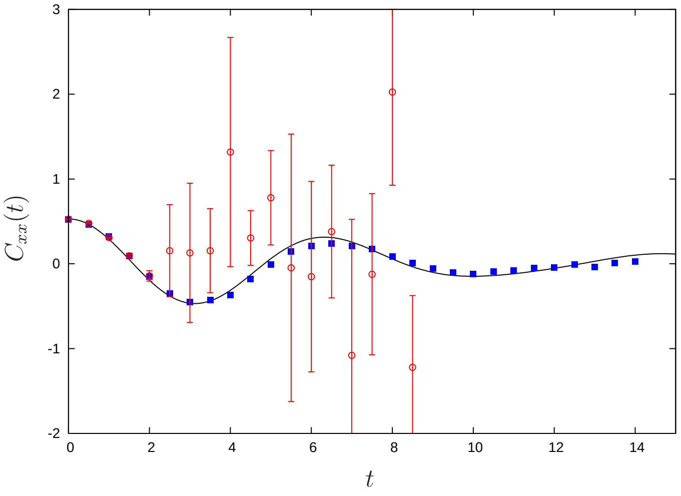

Figure 4.7: Complex time position autocorrelation function for a 7-dimensional multi-frequency harmonic oscillator with unit mass at $\beta = 5$. Solid Line: exact results. Blue squares: results obtained with the IMC method. Red circles: results obtained conventional PIMC procedure, with the number of samples adjusted to have the same number of operations as IMC. See the text for additional details.

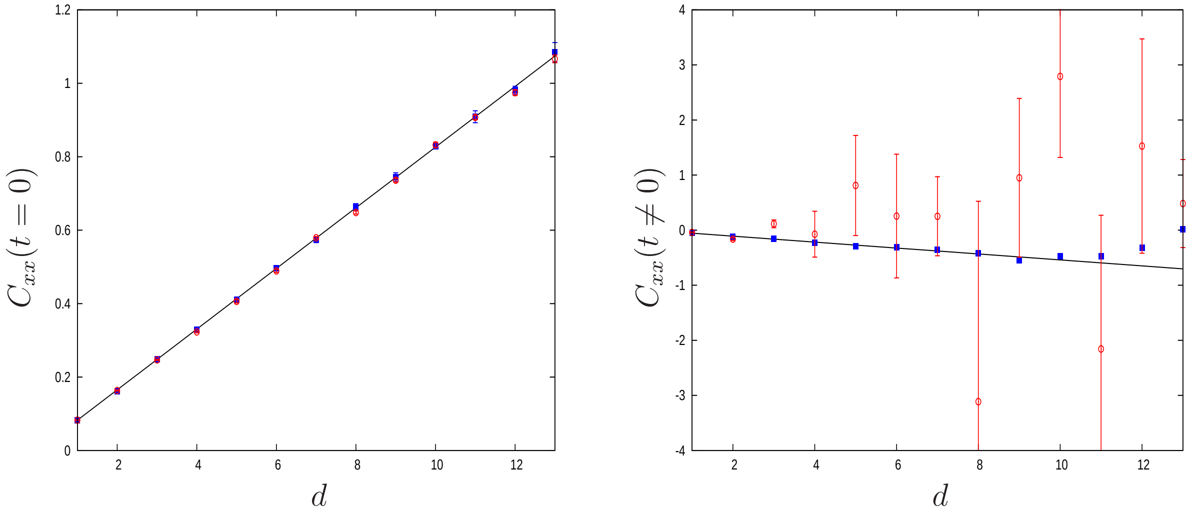

Figure 4.8: Complex time position autocorrelation function of a d-dimensional harmonic oscillator with unit mass and frequency for $\beta = 5$. Solid Line: exact results as a function of the number d of dimensions. Solid blue squares: results obtained with the IMC method using $2N - 1 = 5$ (10 path integral beads). Hollow red circles with error bars: results obtained via conventional Metropolis Monte Carlo evaluation of the complex time path integral, with 10 path integral beads and the number of samples adjusted to have the same number of operations as IMC. (a) $t = 0$, IMC performed with a total of 10000 grid points. (b) $t = 4$, IMC with performed with $10000d$ grid points up for $d \leq 9$ and with an addition of 100,000 grid points per dimension for $d \geq 10$.

Chapter 5

IMC with whole-necklace sampling

Clearly, the choice of grid points is extremely important to the success of a Monte Carlo-based method. In earlier chapters we discussed two possible schemes for selecting grid pairs. Specifically, in chapter 4 we used a potential sampling procedure. This resulted in similar grids for all iterations, but the spread of the distributions was rather large because the sampling function was a high temperature distribution. In chapter 5 we improved the grid selection process by using a bead-adapted sampling procedure. The latter leads to grid pair distributions of varying shape; the first bead pair is associated with a high temperature distribution and thus a large span, while the addition of complex time steps in the sampling function with increasing k effectively lowers the temperature, leading to more compact grids. The wide span of the grid for the initial iterations is not optimal for convergence. Recall the expression for the complex time correlation function, (3.4),

$$C_{AB}(t) = \frac{1}{Z} \int \int dx dx' \langle x| e^{it_c^* \hat{H}/\hbar} |x'\rangle \langle x'| e^{-it_c \hat{H}/\hbar} |x\rangle A(x) B(x'). \tag{5.1}$$

If we express both the forward and backward complex time propagators as path integrals, and if number of time slices in each direction is $2N-1$, then the above expression for the correlation function uses $2(2N-1)$ variables or beads. Since the correlation function is a trace, these beads form a closed or whole necklace. Our goal is to compute the correlation function and ideally, we would like to have grid pairs distributed as the absolute value of the integrand for the entire path integral necklace (see Figure 5.1). Such sampling generates equivalent and economical grid distributions at all path integral steps (beads). The apparent

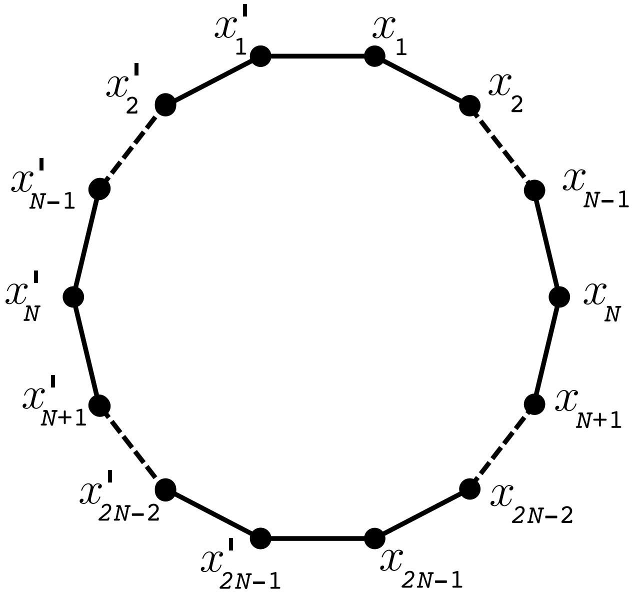

Figure 5.1: Illustration of the path integral necklace for the complex time correlation function, (5.1). The upper half segment corresponds to the complex time forward propagator, (3.9). The lower half necklace corresponds to the backward propagator.

difficulty in the IMC use of such grids is that one must know the marginal distribution of individual bead pairs. Thus, we develop here a procedure for evaluating the required marginal distributions. We find that the whole-necklace grid selection procedure leads to superior performance, significantly enhancing stability.

5.1 Whole-necklace sampling strategy

The conventional PIMC procedure uses the absolute value of the integrand as the sampling function. In the case of the path integral representation of the complex time correlation function, this is given by the product of absolute values of the short time propagators,

$$\rho(x'_1,\ldots,x'_{2N-1},x_1,x_{2N-1}) = \big|\langle x'_1|e^{-i\Delta t_c \hat{H}/\hbar}|x_1\rangle\big|\big|\langle x_{2N-1}|e^{-i\Delta t_c \hat{H}/\hbar}|x'_{2N-1}\rangle\big|\times$$
$$\prod_{l=1}^{N-1}\big|\langle x'_l|e^{-i\Delta t_c \hat{H}/\hbar}|x'_{l+1}\rangle\big|\big|\langle x_l|e^{-i\Delta t_c \hat{H}/\hbar}|x_{l+1}\rangle\big|\big|\langle x'_{N+l}|e^{i\Delta t_c^* \hat{H}/\hbar}|x'_{N+l-1}\rangle\big|\big|\langle x_{N+l}|e^{i\Delta t_c^* \hat{H}/\hbar}|x_{N+l-1}\rangle\big|. \tag{5.2}$$

Equation (5.2) is the joint probability distribution of all the variables (beads) that enter the sampling function. Further, we use the absolute value of the short time propagator,

$$Q_1(x',x) = \left|\langle x'|e^{-i\Delta t_c \hat{H}/\hbar}|x\rangle\right| \tag{5.3}$$

and its iteration,

$$Q_{2k+1}(x',x) = \int dx'' \int dx''' Q_1(x',x''')Q_{2k-1}(x''',x'')Q_1(x'',x), \tag{5.4}$$

to define functions Q_{2k-1} analogous to the propagator iteration, (3.15). (Note that the absolute value of the complex time propagator is the same for a complex time step and its complex conjugate, thus the value of Q_{2k-1} does not depend on whether the time steps it spans are on the forward or backward part of the necklace.) These functions are symmetric, i.e., $Q_{2k-1}(x',x) = Q_{2k-1}(x,x')$. Given the joint probability distribution (5.2), the desired grid pair distribution $P_{2k-1}(x'_k, x_k)$ is the marginal distribution

$$\begin{aligned} P_{2k-1}(x'_k,x_k) = \int dx'_1 \int dx_1 \cdots \int dx'_{k-1} \int dx_{k-1} \int dx'_{k+1} \int dx_{k+1} \cdots \\ \int dx'_{2N-1} \int dx_{2N-1}\, \rho(x'_1,\ldots,x'_{2N-1},x_1,\ldots,x_{2N-1}). \end{aligned} \tag{5.5}$$

From this and (5.4) it follows that

$$P_{2k-1}(x'_k,x_k) = Q_{2k-1}(x'_k,x_k)Q_{4N-2k-1}(x'_k,x_k). \tag{5.6}$$

Below we describe a recursive scheme for calculating Q_{2k-1}.

Application of (5.4) gives

$$Q_3(x'_2,x_2) = \int dx'_1 \int dx_1 Q_1(x'_2,x'_1)Q_1(x'_1,x_1)Q_1(x_1,x_2). \tag{5.7}$$

Since the distribution of the grid point pairs (x'_1,x_1) is $P_1(x'_1,x_1) = Q_1(x'_1,x_1)Q_{4N-3}(x'_1,x_1)$

(from (5.6)), the Monte Carlo estimate to (5.7) is

$$\begin{aligned}
Q_3(x_2', x_2) &= \sum_{x_1', x_1} \frac{\int dx_1' \int dx_1 P_1(x_1', x_1)}{P_1(x_1', x_1)} Q_1(x_2', x_1') Q_1(x_1', x_1) Q_1(x_1, x_2) \\
&= \lambda \sum_{x_1', x_1} \frac{Q_1(x_2', x_1') Q_1(x_1, x_2)}{Q_{4N-3}(x_1', x_1)}.
\end{aligned} \tag{5.8}$$

Here $\lambda = \int dx_k' \int dx_k P_{2k-1}(x_k', x_k)$ is a normalization constant which is going to be the same for any bead pair and will eventually cancel out just like before. Similarly, we find

$$\begin{aligned}
Q_5(x_3', x_3) &= \sum_{x_2', x_2} \frac{\int dx_2' \int dx_2 P_3(x_2', x_2)}{P_3(x_2', x_2)} Q_1(x_2', x_2') Q_3(x_2', x_2) Q_1(x_2, x_3) \\
&= \lambda \sum_{x_2', x_2} \frac{Q_1(x_3', x_2') Q_1(x_2, x_3)}{Q_{4N-5}(x_2', x_2)}.
\end{aligned} \tag{5.9}$$

More generally,

$$\begin{aligned}
Q_{2k-1}(x_k', x_k) &= \lambda \sum_{x_{k-1}', x_{k-1}} \frac{Q_1(x_k', x_{k-1}') Q_{2k-3}(x_{k-1}', x_{k-1}) Q_1(x_{k-1}, x_k)}{P_{2k-3}(x_{k-1}', x_{k-1})} \\
&= \lambda \sum_{x_{k-1}', x_{k-1}} \frac{Q_1(x_k', x_{k-1}') Q_1(x_k, x_{k-1})}{Q_{4N-2k+1}(x_{k-1}', x_{k-1})}
\end{aligned} \tag{5.10}$$

for $k = 2, \ldots, N$. Since the single-step propagator is assumed available, (5.10) is a (proportionality) relation between Q_n and Q_{4N-n} for odd values of n.

For closure, we must obtain additional relations. Starting again with Q_1, we use the two-step propagator on both sides to obtain the relation

$$Q_5(x_3', x_3) = \int dx_1' \int dx_1 Q_2(x_3', x_1') Q_1(x_1', x_1) Q_2(x_1, x_3). \tag{5.11}$$

whose Monte Carlo estimate is

$$\begin{aligned} Q_5(x'_3, x_3) &= \sum_{x'_1, x_1} \frac{\int dx'_1 \int dx_1 P_1(x'_1, x_1)}{P_1(x'_1, x_1)} Q_2(x'_3, x'_1) Q_1(x'_1, x_1) Q_2(x_1, x_3) \\ &= \lambda \sum_{x'_1, x_1} \frac{Q_2(x'_3, x'_1) Q_2(x_1, x_3)}{Q_{4N-3}(x'_1, x_1)}. \end{aligned} \tag{5.12}$$

This procedure leads to the new set of relations

$$\begin{aligned} Q_{2k+1}(x'_{k+1}, x_{k+1}) &= \lambda \sum_{x'_{k-1}, x_{k-1}} \frac{Q_2(x'_{k+1}, x'_{k-1}) Q_{2k-3}(x'_{k-1}, x_{k-1}) Q_2(x_{k-1}, x_{k+1})}{P_{2k-3}(x'_{k-1}, x_{k-1})} \\ &= \lambda \sum_{x'_{k-1}, x_{k-1}} \frac{Q_2(x'_{k+1}, x'_{k-1}) Q_2(x_{k-1}, x_{k+1})}{Q_{4N-2k+1}(x'_{k-1}, x_{k-1})}. \end{aligned} \tag{5.13}$$

These relations connect Q_{4N-n} and Q_{n+2} for odd n.

It is clear that these relations connect all Q_n for $n = 3, 5, \ldots, 4N-3$. Starting with a zeroth order approximation for Q_3, we use its relation to Q_{4N-3} to correct Q_3 self-consistently, obtaining accurate distributions Q_3 and Q_{4N-3}. Next, we use the distribution Q_{4N-3} to obtain Q_5, from which we determine Q_{4N-5}, etc.

Finally, 5.6 implies that the single-bead distributions

$$S_k(x_k) = \int dx'_k P_{2k-1}(x'_k, x_k) \tag{5.14}$$

are independent of the bead index k, thus all beads share the same distributions. In fact, $S_k(x) \equiv S(x)$ is the PIMC distribution that characterizes each bead of the necklace (with the same complex time argument).

5.2 Results

In this section we present numerical results that illustrate the IMC methodology with whole-necklace sampling. In all calculations we use the Trotter factorization of the short time propagators just like before. As it should be clear from the discussion in the previous section, the main advantage of the present IMC is its use of optimal grids, generated via whole-necklace PIMC sampling. Figures 5.2, 5.3 and 5.4 show the distributions $P_{2k-1}(x'_k, x_k)$ for $k = 1, 3$ and 5 generated by the procedure described in section 5.1 for a one-dimensional harmonic oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. The distributions are also compared to those obtained using the potential sampling and bead-adapted sampling procedures. As expected, the original potential sampling scheme results in broad circular distributions that have similar shapes for all bead pairs. Our recent bead-adapted sampling gives rise to distributions that are significantly more compact, starting out elongated along the main diagonal and becoming more symmetric as the inclusion of additional beads in the sampling function effectively lowers the temperature. As seen in the figures, the present whole-necklace sampling scheme leads to the most compact grid distributions in all cases, which have shapes that are similar to those obtained from bead-adapted sampling but are less spread out. Shown in Figure 5.5 are single-bead distributions, (5.14), obtained from the calculated pair distributions (some of which are displayed in Figures 5.2, 5.3, and 5.4) by integrating (summing) with respect to one of the coordinates and binning the results. As argued in the previous section, the single-bead probabilities are independent of bead index in whole-necklace sampling. Figure 5.5 verifies this behavior, showing that the single-bead distributions obtained from the distinct pair distributions are indeed indistinguishable.

Figures 5.6–5.9 illustrate the performance of the present IMC scheme, which employs whole-necklace sampling. Error bars were estimated from the variance of several separate IMC calculations. Figure 5.6 illustrates the performance of the method at each iteration with a (fixed) complex time step $\Delta t_c = \Delta t - i\hbar\Delta\beta$. Since a graph of the propagated function $\tilde{R}_{2k-1}$ would not be very informative, we present in Figure 5.6 the value of the symmetrized correlation function at the complex time $(2k-1)\Delta t_c$, i.e., at the inverse temperature $2(2k -$

$1)\Delta\beta$ and time $(2k-1)\Delta t$,

$$C_{xx}((2k-1)\Delta t_c) = \frac{\int dx_k' \int dx_k |R_k(x_k', x_k)|^2 x_k' x_k}{\int dx_k' \int dx_k |R_k(x_k', x_k)|^2}. \tag{5.15}$$

Note that the temperature decreases with each iteration in this calculation. Results are shown in Figure 5.6 for a one-dimensional harmonic oscillator with $\hbar\omega\Delta\beta = 0.05$ and $\omega\Delta t = 0.1$. The IMC calculation was performed with 10000 grid points, and results are presented with up to 15 iterations (29 path integral slices in the propagator). Also shown are results obtained by the PIMC method with the number of samples adjusted to have the same number of operations as the IMC calculation. As expected, the statistical error of the PIMC calculation grows exponentially with the number of path integral beads, while the IMC results with 10000 grid points faithfully track the exact values with statistical deviations of nearly constant magnitude. More specifically, the statistical error does not increase noticeably with the number of iterations when the ratio $\Delta t/\Delta\beta$ is held fixed.

Next, we compare the IMC methodology with the whole-necklace sampling procedure presented in the present paper to our recent scheme which employed a bead-adapted sampling procedure. Figure 5.7 present such IMC calculations on a d-dimensional harmonic oscillator of unit mass and frequency for $\beta = 1$, $t = 0$, and $N = 3$ (10 beads) as a function of the number d of spatial degrees of freedom. The total number of grid points was set equal to 10000 for $d \leq 15$ and 20000 for $d > 15$. The two schemes perform almost equally well up to $d = 10$. However, the bead-adapted IMC results begin to degrade as the number of spatial dimensions is increased, requiring a larger grid to maintain comparable precision. In contrast to this behavior, the whole-necklace IMC procedure yields accurate results up to 50 degrees of freedom. This comparison demonstrates the superiority of the present sampling technique.

Figure 5.8 shows the position autocorrelation function of a d-dimensional harmonic oscillator with unit mass and frequency for $\beta = 7$, $t = 3.5$, and $N = 4$ (14 beads) as a function of the number d of spatial degrees of freedom. The total number of grid points employed in IMC was increased linearly with d as the function $5000d$.

Finally, we show in Figure 5.9 the position autocorrelation function of an anharmonic oscillator with potential function $V(x) = \frac{1}{2}x^2 + \frac{1}{5}x^4$ at a fixed temperature corresponding to $\beta = 1$. The number of beads ranged between $6(N = 2)$ and $78(N = 20)$. The number of grid points in the IMC calculations was increased linealy with the number of beads according to the relation $1500 + 2000N$.

5.3 Summary

We have shown that the IMC methodology can be formulated in terms of grids obtained via ordinary PIMC sampling. Because the scheme uses the absolute value of the entire complex time propagator, the result grid distributions are spatially confined as the Boltzmann density itself at the given temperature. Thus, unnecessary phase cancellation from broader regions of space is avoided, leading to better statistics. The recursive procedure presented in section 5.1 leads to determination of all bead-pair probability distributions, converging very rapidly. Our numerical results presented in the above section show that converged results are obtained using modest-sized grids of $\sim 10^4$ points on systems of 10–20 degrees of freedom. In these calculations, the grid size was increased linearly with system dimension. However, because the IMC integrand contains an oscillatory factor, phase cancellation will eventually (as the number of particles is increased) become extensive, necessitating the use of many more grid points to achieve convergence.

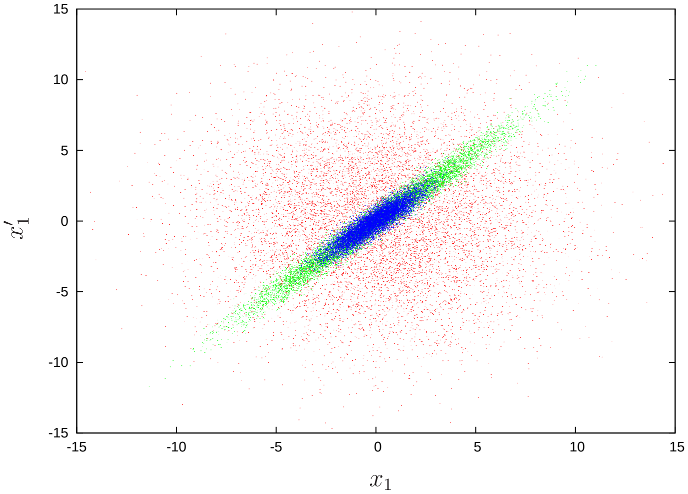

Figure 5.2: Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 1$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling.

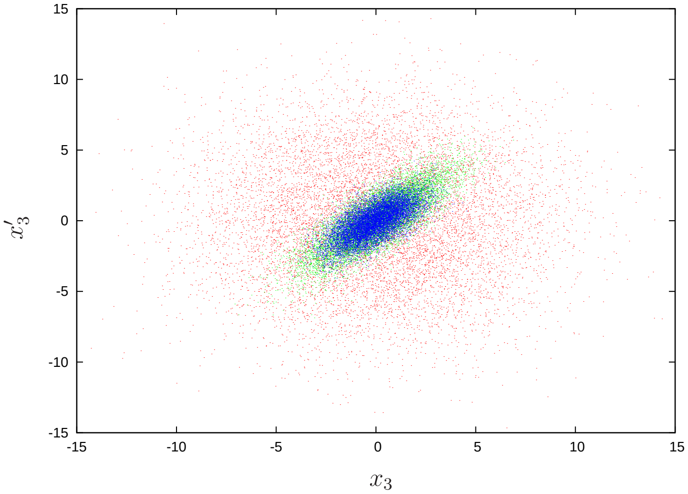

Figure 5.3: Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 3$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling.

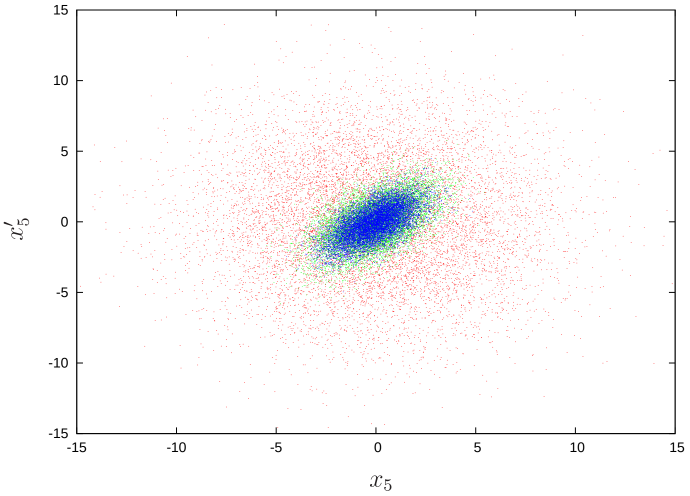

Figure 5.4: Bead pair distributions $P_{2k-1}(x'_k, x_k)$ for $k = 5$ generated by the whole-necklace sampling procedure for a one dimensional oscillator with $\hbar\omega\beta = 1$, $\omega t = 1$, and $2N - 1 = 9$. Red: potential-only sampling. Green: bead-adapted sampling. Blue: whole-necklace PIMC sampling.

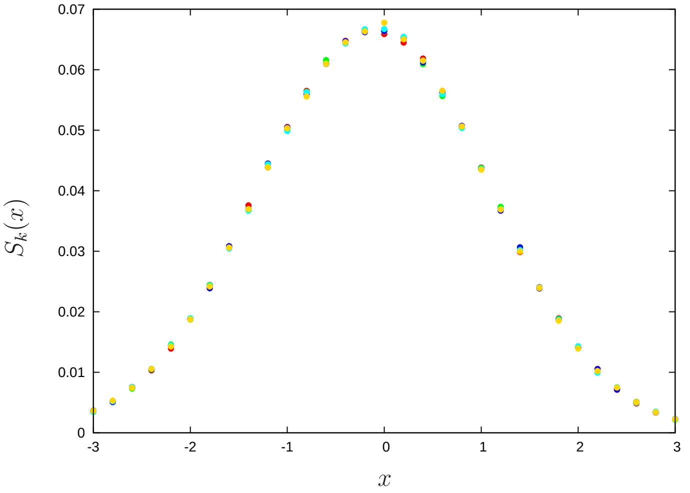

Figure 5.5: Single-bead distributions, (5.14), as obtained from integrating the two-bead distributions (obtained with whole-necklace sampling) shown in Figures 5.2, 5.3, and 5.4. Red, green, blue, cyan and gold correspond to results obtained from the two-bead distributions with $k = 1, 2, 3, 4, 5$ respectively.

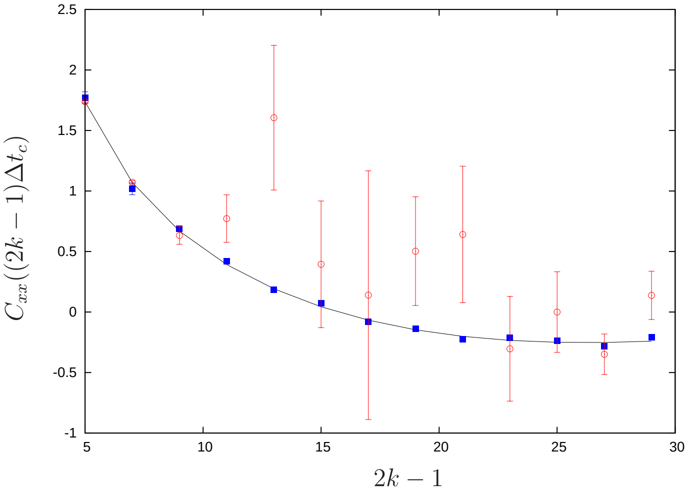

Figure 5.6: Position correlation function with fixed complex time step, (5.15). Blue squares: whole-necklace IMC with 10000 grid points. Hollow circles: PIMC results with the same number of operations.

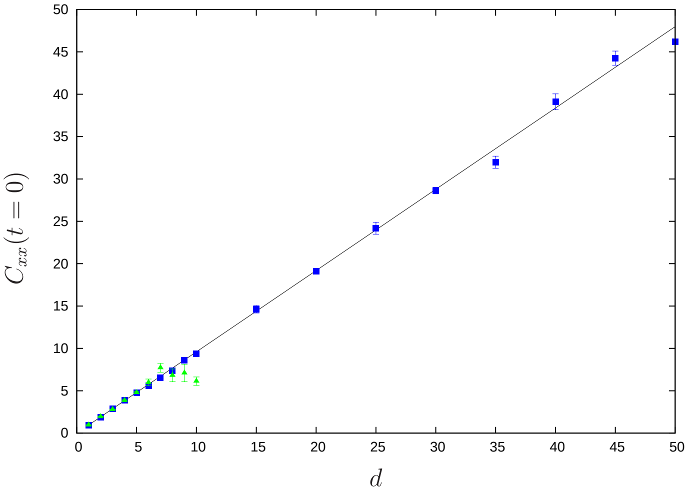

Figure 5.7: Zero-time position correlation function for a d-dimensional harmonic oscillator. Green triangles: IMC with bead-adapted sampling. Blue squares: IMC with whole-necklace sampling.

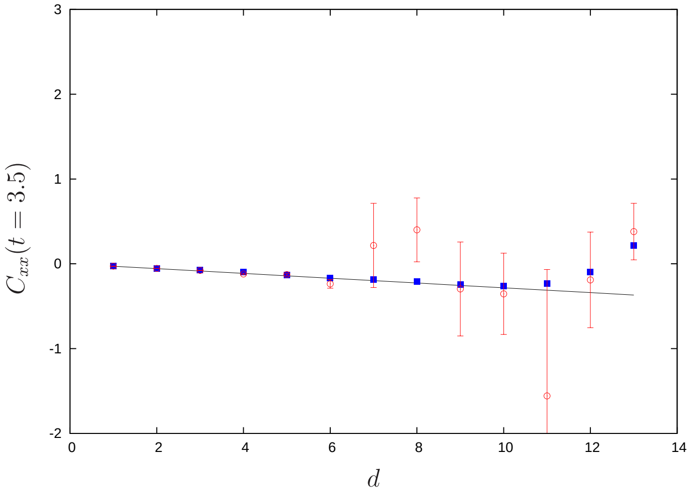

Figure 5.8: Position correlation function for a d-dimensional harmonic oscillator. Blue squares: IMC with whole-necklace sampling. Red circles: PIMC results with the same number of operations.

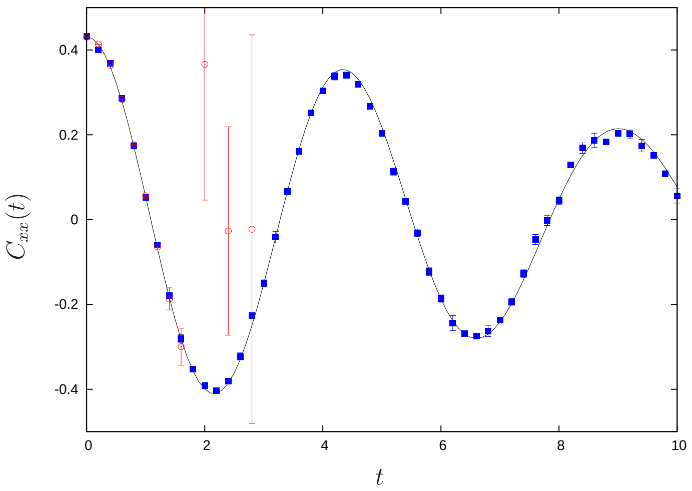

Figure 5.9: Position correlation function for a one-dimensional anharmonic oscillator. Blue squares: IMC with whole-necklace sampling. Red circles: PIMC results with the same number of operations.

Chapter 6

Conclusion

The IMC path integral methodology avoids the exponential proliferation of the sign problem with increasing number of time steps by evaluating the path integral expression iteratively. At the same time, it takes advantage of Monte Carlo sampling, both in the selection of grid points and the integral evaluation. These features make IMC a promising approach for performing fully quantum mechanical calculations on systems with several degrees of freedom. In the conventional PIMC method, a random walk is performed in the space of all path integral variables, and the average value of the quantity of interest is computed from the sampled path configurations, which have a probability distribution given by the given weight function. If M paths are sampled, the computed value is the average of M integrand evaluations. The presence of the Boltzmann factor in finite-temperature expression implies that the integrand is localized over a very small fraction of path space. Because the Metropolis sampling procedure focuses on statistically important paths, the PIMC algorithm yields results with acceptable statistical error using modest values of M as long as the integrand is a smooth function, as in the case of equilibrium averages (i.e., $t = 0$). Under such conditions, the PIMC statistical error generally grows linearly with integral dimension, thus results of adequate precision can be obtained even with many particles and many path integral variables (beads). On the other hand, the integrand is highly oscillatory in real time calculations ($t \neq 0$), leading to exponential dependence of the Monte Carlo error on integral dimension. In such cases the PIMC method requires astronomical numbers of terms to converge if there are more than a few degrees of freedom and/or path integral beads.

The IMC scheme uses a Monte Carlo-based grid of M points for each integration variable, which is constructed from the coordinates of paths visited in a PIMC calculation. Rather than averaging the desired function values over paths, IMC evaluates the multi-dimensional integral iteratively, summing over only a pair of variables in each step. After N such steps, the IMC value of the complex time propagator is equivalent to the sum of M^N integrand values, and when the propagator is squared to give the correlation function, the number of combined terms is M^{2N}. This exponential proliferation of terms with the number of beads counters the exponential growth of statistical error due to phase cancellation. If the number M of grid points required to perform a single step Δt_c in the iterative process is within reach for the problem at hand, the IMC calculation will yield results of comparable precision to complex times $\sim N\Delta t_c$, i.e., for low temperatures and over long real time. Thus, the numerical effort in IMC scales *linearly* with propagation time and inverse temperature.

We note that the above remarks pertain to the precision attained as the number of path integral beads increases with a fixed complex time step. If, on the other hand, Δt_c is changed as a result of incrementing the real time t while keeping the temperature fixed, the oscillatory character of the short complex time propagator will increase. In that case it may be necessary to increase the number of IMC grid points in order to maintain a nearly constant statistical error.

Similarly, the statistical error of IMC is expected to grow as the number of particles is increased. For d degrees of freedom, the number of grid points required in IMC must be sufficiently large to account properly for phase cancellation in the $2d$-dimensional integral performed in each step. Compared to the $2(2N-1)d$-dimensional integral performed in PIMC, the gain in efficiency again grows exponentially with the number of path integral beads.

Applications of IMC we envision include simulation of barrier crossing processes through the evaluation of reactive flux correlation functions [69], which is an ongoing project, and calculation of force and velocity correlation functions in atomic clusters. As the number of degrees of freedom increases, we expect that the required number of grid points will eventually become unrealistically large. Establishing the practical system size limitations of IMC will also be the subject of future work. IMC has already been extended to compute the more demanding real time correlation function by Baltaretu and Makri [70]. Possible extensions of this idea to the problem of fermion statistics are discussed in the work of Chen [71], though the sign problem seems too severe in that case. Finally, we note that the present scheme can take advantage of advanced and highly efficient PIMC algorithms, as well as path integral molecular dynamics (PIMD) techniques [72]. These algorithms have proven extremely useful for simulating equilibrium properties of quantum many-body systems. The iterative Monte Carlo methodology described in this dissertation extends these capabilities to time-dependent properties.

Appendix A

The source codes for the three versions of IMC – Potential-only (IMC1), Bead-Adapted (IMC2), and Whole-necklace (IMC3), are included in the file source.zip and are uploaded via the Electronic Thesis Deposit. The codes are written in FORTRAN 90.

References

[1] R. P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6-7):467–488, 1982.

[2] B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik, and A. G. White. Towards quantum chemistry on a quantum computer. *Nature Chemistry*, 2(2):106–111, Feb 2010.

[3] Ivan Kassal, Stephen P. Jordan, Peter J. Love, Masoud Mohseni, and Alán Aspuru-Guzik. Polynomial-time quantum algorithm for the simulation of chemical dynamics. *Proceedings of the National Academy of Sciences*, 105(48):18681–18686, 2008.

[4] C. Leforestier, R. H. Bisseling, C. Cerjan, M. D. Feit, R. Friesner, A. Guldberg, A. Hammerich, G. Jolicard, W. Karrlein, H. D. Meyer, N. Lipkin, O. Roncero, and R. Kosloff. A comparison of different propagation schemes for the time dependent schrdinger equation. *Journal of Computational Physics*, 94(1):59 – 80, 1991.

[5] R. P. Feynman. Space-time approach to non-relativistic quantum mechanics. *Rev. Mod. Phys.*, 20(2):367–387, Apr 1948.

[6] R. P. Feynman and A. R. Hibbs. *Quantum Mechanics and Path Integrals*. McGraw-Hill, New York, 1965.

[7] Nancy Makri. Feynman path integration in quantum dynamics. *Computer Physics Communications*, 63(1-3):389 – 414, 1991.

[8] M. F. Trotter. *Proc. Am. Math. Soc.*, 10:545, 1959.

[9] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[10] D. L. Freeman J. D. Doll and T. L. Beck. *Adv. Chem. Phys.*, 78:61, 1990.

[11] N. Makri. *Comput. Phys. Commun.*, 63:389, 1991.

[12] C. H. Mak and R. Egger. *Adv. Chem. Phys.*, 93:39, 1996.

[13] Devarajan Thirumalai, Eric J. Bruskin, and Bruce J. Berne. An iterative scheme for the evaluation of discretized path integrals. *The Journal of Chemical Physics*, 79(10):5063–5069, 1983.

[14] M. D. Feit, J. A. J. Fleck, and S. A. Steiger. *J. Comput. Phys.*, 47:412, 1982.

[15] Devarajan Thirumalai and Bruce J. Berne. On the calculation of time correlation functions in quantum systems: Path integral techniques. *The Journal of Chemical Physics*, 79(10):5029–5033, 1983.

[16] E. C. Behrman, Gary A. Jongeward, and Peter G. Wolynes. A Monte Carlo approach for the real time dynamics of tunneling systems in condensed phases. *The Journal of Chemical Physics*, 79(12):6277–6281, 1983.

[17] J. D. Doll. Monte carlo fourier path integral methods in chemical dynamics. *The Journal of Chemical Physics*, 81(8):3536–3541, 1984.

[18] Johnny Chang and William H. Miller. Monte carlo path integration in real time via complex coordinates. *The Journal of Chemical Physics*, 87(3):1648–1652, 1987.

[19] Nancy Makri and William H. Miller. Monte carlo integration with oscillatory integrands: implications for feynman path integration in real time. *Chemical Physics Letters*, 139(1):10 – 14, 1987.

[20] C. H. Mak and David Chandler. Solving the sign problem in quantum monte carlo dynamics. *Phys. Rev. A*, 41(10):5709–5712, May 1990.

[21] J. D. Doll, Thomas L. Beck, and David L. Freeman. Quantum monte carlo dynamics: The stationary phase monte carlo path integral calculation of finite temperature time correlation functions. *The Journal of Chemical Physics*, 89(9):5753–5763, 1988.

[22] Omar A. Sharafeddin, Donald J. Kouri, Naresh Nayar, and David K. Hoffman. Quadrature-based, coarse-grained treatment of the coordinate representation free particle real-time evolution operator. *The Journal of Chemical Physics*, 95(5):3224–3231, 1991.

[23] C. H. Mak. Stochastic method for real-time path integrations. *Phys. Rev. Lett.*, 68(7):899–902, Feb 1992.

[24] Reinhold Egger and C. H. Mak. Dynamical effects in the calculation of quantum rates for electron transfer reactions. *The Journal of Chemical Physics*, 99(4):2541–2549, 1993.

[25] E. Gallicchio and B. J. Berne. The absorption spectrum of the solvated electron in fluid helium by maximum entropy inversion of imaginary time correlation functions from path integral monte carlo simulations. *The Journal of Chemical Physics*, 101(11):9909–9918, 1994.

[26] Maria Topaler and Nancy Makri. Multidimensional path integral calculations with quasiadiabatic propagators: Quantum dynamics of vibrational relaxation in linear hydrocarbon chains. *The Journal of Chemical Physics*, 97(12):9001–9015, 1992.

[27] Maria Topaler and Nancy Makri. Quasi-adiabatic propagator path integral methods. exact quantum rate constants for condensed phase reactions. *Chemical Physics Letters*, 210(1-3):285 – 293, 1993.

[28] Dmitrii E. Makarov and Nancy Makri. Tunneling dynamics in dissipative curve-crossing problems. *Phys. Rev. A*, 48(5):3626–3635, Nov 1993.

[29] K. Binder and D. W. Heermann. *Monte Carlo simulation in statistical physics*. Springer-Verlag, 1988.

[30] R. P. Feynman. *Statistical Mechanics*. Addison-Wesley, Reading, MA, 1972.

[31] Nicholas Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44:335–341, 1949.

[32] J. M. Thijssen. *Computational Physics*. Cambridge University Press, 1999.

[33] S. E. Koonin. *Computational Physics*. Westview Press, 1990.

[34] Werner Krauth. *Statistical Mechanics: Algorithms and Computations*. Oxford University Press, 2006.

[35] D. M. Ceperley. Path integrals in the theory of condensed helium. *Rev. Mod. Phys.*, 67(2):279–355, Apr 1995.

[36] D. M. Ceperley and L. Mitas. *Adv. Chem. Phys.*, 93:1, 1996.

[37] M. Suzuki. *Quantum Monte Carlo Methods in Condensed Matter Physics*. World Scientific, 1993.

[38] D. M. Ceperley. Microscopic simulations in physics. *Rev. Mod. Phys.*, 71(2):S438–S443, Mar 1999.

[39] André G. Moreira, Stephan A. Baeurle, and Glenn H. Fredrickson. Global stationary phase and the sign problem. *Phys. Rev. Lett.*, 91(15):150201, Oct 2003.

[40] Erin M. Lennon, Kirill Katsov, and Glenn H. Fredrickson. Free energy evaluation in field-theoretic polymer simulations. *Phys. Rev. Lett.*, 101(13):138302, Sep 2008.

[41] F. W. Weigel. *Introduction to Path-Integral Methods in Physics and Polymer Science*. World Scientific Publishing Company, 2006.

[42] K. N. Anagnostopoulos and J. Nishimura. New approach to the complex-action problem and its application to a nonperturbative study of superstring theory. *Phys. Rev. D*, 66(10):106008, Nov 2002.

[43] D. A. Mcquarrie. *Statistical Mechanics.* University Science Books, 2000.

[44] D. Thirumalai and B. J. Berne. Time correlation functions in quantum systems. *The Journal of Chemical Physics*, 81(5):2512–2513, 1984.

[45] Michiel Sprik, Michael L. Klein, and David Chandler. Staging: A sampling technique for the monte carlo evaluation of path integrals. *Phys. Rev. B*, 31(7):4234–4244, Apr 1985.

[46] B. J. Berne and C. D. Harp. *Adv. Chem. Phys.*, 17:63, 1970.

[47] William H. Miller, Steven D. Schwartz, and John W. Tromp. Quantum mechanical rate constants for bimolecular reactions. *The Journal of Chemical Physics*, 79(10):4889–4898, 1983.

[48] Daniel Huber and Eric J. Heller. Generalized gaussian wave packet dynamics. *The Journal of Chemical Physics*, 87(9):5302–5311, 1987.

[49] Nancy Makri and William H. Miller. Basis set methods for describing the quantum mechanics of a "system" interacting with a harmonic bath. *The Journal of Chemical Physics*, 86(3):1451–1457, 1987.

[50] André Nauts and Robert E. Wyatt. New approach to many-state quantum dynamics: The recursive-residue-generation method. *Phys. Rev. Lett.*, 51(25):2238–2241, Dec 1983.

[51] Jianshu Cao and Gregory A. Voth. A new perspective on quantum time correlation functions. *The Journal of Chemical Physics*, 99(12):10070–10073, 1993.

[52] William H. Miller. Semiclassical Methods in Chemical Physics. *Science*, 233(4760):171–177, 1986.

[53] Nancy Makri and Keiran Thompson. Semiclassical influence functionals for quantum systems in anharmonic environments. *Chemical Physics Letters*, 291(1-2):101 – 109, 1998.

[54] Keiran Thompson and Nancy Makri. Rigorous forward-backward semiclassical formulation of many-body dynamics. *Phys. Rev. E*, 59(5):R4729–R4732, May 1999.

[55] Akira Nakayama and Nancy Makri. Simulation of dynamical properties of normal and superfluid helium. *Proceedings of the National Academy of Sciences of the United States of America*, 102(12):4230–4234, 2005.

[56] William H. Miller. Quantum dynamics of complex molecular systems. *Proceedings of the National Academy of Sciences of the United States of America*, 102(19):6660–6664, 2005.

[57] Goran Krilov, Eunji Sim, and B. J. Berne. Quantum time correlation functions from complex time monte carlo simulations: A maximum entropy approach. *The Journal of Chemical Physics*, 114(3):1075–1088, 2001.

[58] J. D. Doll, R. D. Coalson, and D. L. Freeman. Toward a monte carlo theory of quantum dynamics. *The Journal of Chemical Physics*, 87(3):1641–1647, 1987.

[59] V. S. Filinov. *Nucl. Phys. B*, 271:717, 1986.

[60] C. H. Mak. The sign problem in real-time path integral simulations: Using the cumulant action to implement multilevel blocking. *The Journal of Chemical Physics*, 131(4):044125, 2009.

[61] Nancy Makri. Information guided noise reduction for monte carlo integration of oscillatory functions. *Chemical Physics Letters*, 400(4-6):446 – 452, 2004.

[62] E. Bukhman and N. Makri. Forward-backward semiclassical dynamics with information-guided noise reduction for a molecule in solution. *J. Phys. Chem.*, A111:11320–11327, 2007.

[63] David O. Harris, Gail G. Engerholm, and William D. Gwinn. Calculation of matrix elements for one-dimensional quantum-mechanical problems and the application to anharmonic oscillators. *The Journal of Chemical Physics*, 43(5):1515–1517, 1965.

[64] Maria Topaler and Nancy Makri. System-specific discrete variable representations for path integral calculations with quasi-adiabatic propagators. *Chemical Physics Letters*, 210(4-6):448 – 457, 1993.

[65] Nancy Makri. Quantum dissipative dynamics: A numerically exact methodology. *The Journal of Physical Chemistry A*, 102(24):4414–4427, May 1998. doi: 10.1021/jp980359y.

[66] R. P. Feynman and F. L. Vernon. *J. Ann. Phys.*, 24:118–173, 1963.

[67] Dmitrii E. Makarov and Nancy Makri. Stochastic resonance and nonlinear response in double-quantum-well structures. *Phys. Rev. B*, 52(4):R2257–R2260, Jul 1995.

[68] Hans De Raedt and Bart De Raedt. Applications of the generalized trotter formula. *Phys. Rev. A*, 28(6):3575–3580, Dec 1983.

[69] William H. Miller, Steven D. Schwartz, and John W. Tromp. Quantum mechanical rate constants for bimolecular reactions. *The Journal of Chemical Physics*, 79(10):4889–4898, 1983.

[70] Cristian Baltaretu and Nancy Makri. Iterative Monte Carlo Formulation for Real-Time Correlation Functions. In preparation.

[71] Jonathan E. Chen. *Theoretical methods for the study of quantum dynamics*. PhD thesis, University of Illinois at Urbana-Champaign, 2010.

[72] M. Parrinello and A. Rahman. Study of an f center in molten kcl. *The Journal of Chemical Physics*, 80(2):860–867, 1984.