UNIVERSITY OF LEEDS

Development and Applications of New Basis Set Sampling and Basis Set Handling Procedures for the Coupled Coherent States Family of Methods

by

Christopher Charles Symonds

Submitted in accordance with the requirements for the degree of
Doctor of Philosophy

in the

Faculty Of Mathematics and Physical Sciences
School of Chemistry

December 2015

Declaration of Authorship

The candidate confirms that the work submitted is his own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

The work in Chapter 4 of this thesis has appeared in publication as follows:
Physical Review A, Vol 91, Page 023427, (2015), by C. Symonds, J. Wu, M. Ronto, C. Zagoya, C. Figuera de Morisson Faria and D. V. Shalashilin.
The candidate was responsible for writing the program which generated the results, the running of the simulations and generation of results, creating the graphs and the writing of the majority of the text. The contribution of the other authors was the idea to use wavefunction reprojection, the idea to apply the CCS method to the problem of high harmonic generation, the creation of the benchmark data and the idea to use adaptive grids for the basis set, along with some small sections of the introduction and conclusion in response to reviewers comments.

"I think I can safely say that nobody understands quantum mechanics."

Richard Feynman, in The Character of Physical Law (1965)

UNIVERSITY OF LEEDS

Abstract

Faculty Of Mathematics and Physical Sciences
School of Chemistry

Doctor of Philosophy

by Christopher Charles Symonds

The Coupled Coherent States family of methods have shown themselves capable of simulating the quantum dynamics of many different systems. The ability of these methods to accurately describe quantum behaviour is dependent on using a basis set which covers a sufficient area of phase space. If the area covered is too small, the basis set will be unable to adequately describe the dynamics of the system, however if the area is too large and the basis functions become too widely spaced, coupling will be lost between the coherent states and the simulation will fall into the semiclassical regime. In some situations the loss of this coupling becomes accelerated, through trajectories guiding the basis functions far from each other in phase space for example, limiting the ability of these methods to accurately describe quantum behaviour.

This thesis demonstrates two techniques for preserving a correct description of the wavefunction in phase space. Firstly a combination of initial sampling using swarms of basis function trains and basis function cloning during propagation of the wavefunction is shown to correct a disagreement seen between the two formulations of the Multi-Configurational Ehrenfest method when simulating the high dimensional spin boson model. This combination gives a good agreement with benchmark calculations found using the Multi-Configurational Time-Dependent Hartree method. The techniques used to correct this disagreement have been used previously for on-the-fly *ab initio* direct dynamics simulations, reported in references [1] and [2], and so this investigation provides validation for the results obtained in those publications. Secondly a system of adaptive reprojection of the wavefunction is shown to allow a large grid of coherent states to be reduced to only the area of interest, while keeping the basis set in that region. It is also demonstrated that this will still hold even if the equations of motion tend to move basis functions far away from this area. This adaptive reprojection technique is tested against the high harmonic generation of an electron bound to a pseudo-atomic potential in one dimension, yielding results which are in good agreement with benchmark calculations carried out using the Time-Dependent Schrödinger equation.

Acknowledgements

First and foremost, my eternal gratitude goes to my supervisor, Prof. Dmitry Shalashilin, not only for always being there with a helpful comment and invaluable guidance, but also for his immense faith in me and my work. His optimism in the face of setbacks and his great help in mathematical, computational and scientific matters has been a constant inspiration.

My colleagues in the Quantum and Classical Molecular Dynamics group have always been helpful, especially with the mathematical background to the work and with programming considerations. My thanks go especially to Dr. Miklos Ronto whose enthusiasm for the work is unparalleled and who provided a great deal of help with the simulations of high harmonic generation, and to Dr. Dmitry Makhov whose help understanding the cloning procedure had a direct impact on the work within this thesis.

Prof. Ben Whitaker and Dr. Marcelo de Miranda provided much useful feedback on internal reports and much helpful advice. Their comments have made me a better writer and have made the process of producing this work much easier.

Drs. Carla Figuera de Morisson Faria and Carlos Zagoya of University College London provided invaluable help in understanding the strong field phenomena discussed in this thesis and the intricacies of the strong field approximation, for which they have my thanks. I also acknowledge the help provided by Dr. Jie Wu, whose early work on modelling high harmonic generation with the CCS method was the basis for much of the work done in my own simulations of the phenomenon.

My gratitude goes to Dr. John Kattirtzi, who gave me a great deal of programming advice early on in the course of my PhD and whose CCS code served as an example of the modular design which was incorporated into my own program.

My thanks go to Dr. Julie Fisher who provided advice and support in her role as postgraduate tutor before her unfortunate passing, and also to Dr. Terry Kee who succeeded her and who also helped with making sure that I and my fellow postgraduates always had someone to talk to when advice was needed. My thanks also go to the postgraduate administration team in the School of Chemistry who were always happy to help me with any of the administrative aspects of postgraduate study (even when it wasn't strictly speaking their job).

My undying gratitude goes to Freya Wilson, Rachel Thompson and Matthew Everitt, who were always available for help with all the extra-curricular aspects of life as a postgraduate student and especially to Amanda McDonnell who in addition to this provided valuable help with proof-reading the early drafts of this thesis.

My father Jeremy will always have my gratitude for his support during the hardest times, and for never complaining when I started talking about my work, despite having no idea what I was talking about.

Finally, I would like to acknowledge the funding provided by the University of Leeds School of Chemistry DTG program and the EPSRC, without which none of the work contained within would have been possible.

Contents

List of Figures

List of Tables

Abbreviations

CCS	**C**oupled **C**oherent **S**tates
F-CCS	**F**ermionic **C**oupled **C**oherent **S**tates
MCE	**M**ulti-**C**onfigurational **E**hrenfest
MCEv1	**M**ulti-**C**onfigurational **E**hrenfest **v**ersion **1**
MCEv2	**M**ulti-**C**onfigurational **E**hrenfest **v**ersion **2**
AIMC	***A****b* ***I****nitio* **M**ultiple **C**loning
IVR	**I**nitial **V**alue **R**epresentation
HK	**H**erman-**K**luk
TDVP	**T**ime **D**ependent **V**ariational **P**rinciple
MS	**M**ultiple **S**pawning
FMS	**F**ull **M**ultiple **S**pawning
AIMS	***A****b* ***I****nitio* **M**ultiple **S**pawning
MCTDH	**M**ulti-**C**onfigurational **T**ime-**D**ependent **H**artree
G-MCTDH	**G**aussian **M**ulti-**C**onfigurational **T**ime-**D**ependent **H**artree
MCTDHB	**M**ulti-**C**onfigurational **T**ime-**D**ependent **H**artree for **B**osons
MCTDHF	**M**ulti-**C**onfigurational **T**ime-**D**ependent **H**artree-**F**ock
vMCG	**v**ariational **M**ulti-**C**onfigurational **G**aussian
ML-MCTDH	**M**ulti-**L**ayer **M**ulti-**C**onfigurational **T**ime-**D**ependent **H**artree
HHG	**H**igh **H**armonic **G**eneration
TDSE	**T**ime-**D**ependent **S**chrödinger **E**quation
SFA	**S**trong **F**ield **A**pproximation
ACF	**A**uto**C**orrelation **F**unction
ATI	**A**bove **T**hreshold **I**onisation

NSDI	**N**on-**S**equential **D**ouble **I**onisation
RK4	**R**unge-**K**utta $\mathbf{4}^{th}$ order
RKCK45	**R**unge-**K**utta-**C**ash-**K**arp $\mathbf{4}^{th}/\mathbf{5}^{th}$ order
OMP/OpenMP	**O**pen **M**ulti-**P**rocessing
MPI	**M**essage **P**assing **I**nterface

Notation

Throughout this thesis, the following conventions are used:

i, j, k	Basis function index, given as subscript, ie $\mathbf{z}_k$
r, s	Electronic state/potential energy surface index, given as superscript in parentheses, ie $d^{(r)}$
m, n	Dimensional/degree of freedom index, given as superscript in parentheses, ie $z^{(m)}$.
z	Refers to a single dimensional coherent state
$\mathbf{z}$	Refers to a multi-dimensional coherent state, where $\|\mathbf{z}\rangle = \prod_m \left\|z^{(m)}\right\rangle$
N_{xx}	Total number of a particular parameter, eg N_{bf} is total number of basis functions, N_{rpt} is total number of repetitions, etc.

Any other notation convention used is specified in the text. It should be noted that in calculations throughout, atomic units are used, with the convention that $\hbar = m_e = e = 1$, where $\hbar$ is the reduced Planck constant, m_e is electronic mass and e is electronic charge. For all systems, the coherent state width parameter γ, which is introduced in chapter 2 and used throughout, is set to $\gamma = 1$. To allow for generalisation however, all γ terms in the calculations, Hamiltonians, etc. remain.

This thesis is dedicated to my son Francis, who could make me smile even at the hardest times of the last few years, and who I love more than anything in the world, and also to the bravest and best person I have ever known, my mother Jan, who I miss.

Chapter 1

Introduction

The modelling of quantum interactions has long presented a problem for both computational chemists and computational physicists. Unlike calculations using classical mechanics, quantum mechanical calculations make use of large matrices which can make computation extremely difficult. When solving the time-dependent Schrödinger equation for small systems, typically the wavefunction will be expressed as a set of static, often orthogonal, basis states with amplitudes which evolve in time. The time dependence of these amplitudes is generally found through calculating the eigenvectors and eigenvalues of the Hamiltonian, however for larger systems with many degrees of freedom this process becomes extremely computationally expensive, scaling exponentially with the number of degrees of freedom.

In an effort to circumvent this "curse of dimensionality", many alternative methods have been formulated. Many of these methods use time-dependent basis sets and are often semiclassical in nature. A particularly successful group of simulation methods arise from the work of Heller, who proposed the use of "Frozen Gaussian" wave packets to construct the basis set [3–5]. These "Frozen Gaussians" contained both position and momentum information and so were, by definition, coherent states. Many different methods grew out of Heller's approach[6, 7], notably the Herman-Kluk method[8–10], however due to the semiclassical nature of these methods researchers computing quantum interactions still had to sacrifice a degree of accuracy when propagating the wavefunction in classically forbidden regions. More recently, various methods have been developed which have a stronger quantum mechanical basis and so yield more accurate results for classically forbidden transitions. These methods range from the more classical, such as the *ab initio* Multiple Spawning (AIMS) technique developed by Martinez and Ben-Nun [11–15], to methods which are more quantumly exact such as the Multi-Configurational Time-Dependent Hartree (MCTDH) developed by Meyer and co-workers [16–18].

The Coupled Coherent States (CCS) method and its variations form a family of numerical methods which draws on the work of Heller and his successors, and yet is capable of carrying out fully quantum calculations in classically forbidden regions. The CCS method describes the wavefunction on a basis of coherent states which, as the name suggests, are coupled through the time propagation equations [19]. These coherent states share many properties with Heller's "Frozen Gaussians", and in fact are identical when the Gaussian wave packets are applied to a harmonic potential, as was often the case with these earlier methods [9, 20, 21]. The use of coherent states as a basis allows the wavefunction to be propagated in phase space rather than coordinate space, which is particularly useful as the classical energy of a point in phase space is always known, allowing easy scaling of the Hamiltonian. The phase space propagation also allows for the calculation of quantum amplitudes which, unlike for the semiclassical methods, are formally exact [22]. This results in simulations which are considerably more accurate than the semiclassical alternatives, and in many cases comparable in accuracy to the MCTDH method which is used as a benchmark for many other simulation methods, CCS included [23–25]. Due to the structure of the basis set and the construction of the initial wavefunction however, the CCS methods typically require fewer computational resources than MCTDH. Variations of the CCS method exist, such as the Multi-Configurational Ehrenfest (MCE) method [23, 24], which is generalised to treat multiple electronic states and the *ab initio* Multi-Configurational Ehrenfest method [26, 27] which has shown itself capable of propagating along a potential energy surface which is calculated "on-the-fly", requiring no precalculated potential energy surface to be used.

While the CCS methods have had some success, there are still some applications that are very difficult to simulate. A property common to many trajectory-guided basis set methods is that the basis functions will spread out in phase space as the basis set is propagated in time. This spreading allows the wavefunction to be described over larger areas of phase space, however in some systems the spreading which occurs may not be sufficient to cover a large enough area to properly describe the dynamics of the wavefunction. In other cases however as the basis functions spread they may become sufficiently separated to lose coupling, thus becoming unable to correctly model classically forbidden regions and falling into the semiclassical regime. For some model Hamiltonians, this process becomes accelerated under certain conditions, and so they become very difficult to simulate. Other systems may need longer time propagation, and thus be difficult to simulate properly due to the spreading out of the basis set. In previous publications the basis set has been propagated continuously with no modifications made to the basis set during propagation. There is no requirement for this to be the case however, and various basis set handling procedures are available which can limit the loss of coupling between the basis functions.

In this thesis, methods to ensure that the wavefunction remains properly described by the basis set in phase space are investigated using basis set sampling methods and modifications to the propagation of the wavefunction. Chapter 2 will discuss in more detail the background to the CCS and MCE methods, compare them to other current numerical methods and discuss where they stand in the landscape of methods used for computational chemistry. Chapter 3 will investigate the application of the MCE method to the spin boson model, a two level system-bath model, and reveal a discrepancy between the results obtained with the two formulations of the MCE equations. It will be shown that this discrepancy can be overcome through implementation of modifications to the propagation of the wavefunction used in the *ab initio* Multiple Cloning method of Makhov *et al.* [1, 2], thus verifying the results obtained by this method. Chapter 4 will investigate the applicability of the CCS method to high harmonic generation, an extremely non-linear strong field phenomenon, and the implementation of modifications to the propagation of the wavefunction through reprojection of the wavefunction onto adaptive grid basis sets. Chapter 5 will give a summary and conclusion for the work presented in the previous chapters. The appendices give details of the functioning of the program that was written for the purpose of carrying out the CCS and MCE calculations (Appendix A), derivations of the governing equations for the methods used (Appendix B), and the use of the CCS/MCE program (Appendix C).

Chapter 2

Background and Theory

2.1 Introduction

To solve the Schrödinger equation for a Hamiltonian with multiple degrees of freedom, one risks the "exponential curse", also known as the "curse of dimensionality" where the computational cost of modelling the interaction grows exponentially with the number of degrees of freedom such that when modelling on a grid, the number of grid points needed, N, is given by

$$N = l^M \tag{2.1.1}$$

where l is the number of grid points per degree of freedom for M degrees of freedom. Many different methods have been developed of modelling the quantum mechanical behaviour of molecules semiclassically or in a semiclassically inspired manner in the hope of circumventing this effect. Many of these methods can be traced back to the work of Heller [4] and later work by Herman, Kluk and co-workers [8, 28, 29], whose methods were based on the propagation of "Frozen Gaussian" wave packets, and to the earlier work of Miller and George [30], who first described a method of continuing classical mechanics in classically forbidden regions through the use of an Initial Value Representation (IVR). In the last few decades, a group of simulation methods based on the propagation of coherent basis states along classical trajectories has had not inconsiderable successes in modelling fully quantum interactions over short time scales, while avoiding the exponential scaling which renders direct calculation prohibitively computationally expensive through use of Monte Carlo sampling which, instead of scaling exponentially, can in principal scale quadratically. The method of Coupled Coherent States [31] and related techniques belong to this group. In this section, the historical basis of the CCS method will be considered and a review of the work carried out using this method will be presented, along with work carried out using the more general Multi-Configurational Ehrenfest

(MCE) method which succeeded it [32]. The place of the CCS family of methods will also be considered in the overarching context of computational chemistry through a comparison with alternative approaches to the simulation of quantum effects.

2.2 Initial Value Representation methods and "Frozen Gaussian" Dynamics

2.2.1 The Miller Initial Value Representation

The work of Miller and co-workers in the early 1970s focused mainly on describing atomic collisions in terms of classical S matrix theory [33, 34]. The initial value representation was first introduced as a more general expression for the S matrix in terms of classical quantities and a way of more accurately treating highly non-classical transitions [34]. This was expanded later to utilise different representations since one representation is equivalent to another provided one transforms between the two properly [30]. It later became apparent however that Miller's formalism, and the similar technique of Marcus [35], required "*a difficult root search for all classical trajectories, starting with specified classical action for reactants and ending with a specified classical action for products, proving to be prohibitive computationally for even the simplest chemical reactions*" [10]. Through later expansions on Miller's work however, the Initial Value Representation (IVR) became one of the more favoured methods of modelling classically forbidden interactions [31]. The appeal of these IVR techniques was that they allowed the researcher to extract quantum information from trajectories directly determined by their initial conditions and as such "*allowed phenomena to be understood in intuitive physical terms familiar from ordinary quasi-classical and classical molecular dynamics treatments* [9]".

In the basic IVR method, one often starts with the time evolution operator in co-ordinate representation

$$K_t(\mathbf{x}', \mathbf{x}) = \left\langle \mathbf{x}' \right| \hat{K}_t \left| \mathbf{x} \right\rangle = \left\langle \mathbf{x}' \right| \mathrm{e}^{\frac{-i\hat{H}t}{\hbar}} \left| \mathbf{x} \right\rangle \tag{2.2.1}$$

where $\hat{H}$ is the Hamiltonian. Knowledge of $K_t(\mathbf{x}', \mathbf{x})$ allows one to describe the time evolution of the wavefunction in the usual way through an integral over all $\mathbf{x}$. One can then apply the van Vleck semiclassical approximation [9, 21, 36], such that

$$K_t(\mathbf{x}', \mathbf{x}) = (2\pi i\hbar)^{f/2} \sum \left| \det \frac{\partial \mathbf{x}'}{\partial \mathbf{p}} \right|^{-\frac{1}{2}} \mathrm{e}^{iS_t(\mathbf{x}', \mathbf{x})/\hbar - i\pi\nu/2}, \tag{2.2.2}$$

where f is the number of degrees of freedom of the system and the sum is taken over all trajectories which originate at point $\mathbf{x}$ at time 0 and end at point $\mathbf{x}'$ at time t. ν is

an integer known as the Morse phase, which changes discontinuously at the roots of the Jacobian $\det\frac{\partial \mathbf{x}'}{\partial \mathbf{p}}$, at which the van Vleck expression will diverge. $S_t(\mathbf{x}', \mathbf{x})$ is the classical action, given by

$$S_t(\mathbf{x}', \mathbf{x}) = \int_0^t \mathrm{d}\tau[\mathbf{p}_\tau \dot{\mathbf{q}}_\tau - H(\mathbf{p}_\tau, \mathbf{q}_\tau)], \tag{2.2.3}$$

where $(\mathbf{p}_t, \mathbf{q}_t)$ are the position and momentum at time t, $\mathbf{q}_t = \mathbf{x}'$, and $\mathbf{q}_0 = \mathbf{x}$, and H is the classical Hamiltonian. The formulation of equation (2.2.2) gives the boundary value problem mentioned earlier where a root search will result in multiple roots of the system. Simplification of (2.2.2) can be carried out by substituting the van Vleck propagator in the identity

$$K_t(\mathbf{x}', \mathbf{x}) = \int \mathrm{d}\mathbf{q} \int \mathrm{d}\mathbf{q}_t \delta(\mathbf{x}' - \mathbf{q}_t) K_t(\mathbf{q}_t, \mathbf{q}) \delta(\mathbf{x} - \mathbf{q}), \tag{2.2.4}$$

and subsequently it is possible to change the integration parameters such that

$$K_t(\mathbf{x}', \mathbf{x}) = \int \mathrm{d}\mathbf{p} \int \mathrm{d}\mathbf{q} \delta(\mathbf{x}' - \mathbf{q}_t) \left|\det \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}}\right|^{\frac{1}{2}} \frac{\mathrm{e}^{iS_t(\mathbf{x}', \mathbf{x})/\hbar - i\pi\nu/2}}{(2\pi i\hbar)^{f/2}} \delta(\mathbf{x} - \mathbf{q}). \tag{2.2.5}$$

As the Jacobian is now in the numerator rather than the denominator (as in equation (2.2.2)) the integrand no longer diverges at roots, but becomes zero instead. While the integrand no longer diverges due to the Jacobian however, the roots still cause a degradation in accuracy [9]. If the above expression is applied to the correlation function about $\hat{K}_t$, the delta functions are replaced by the wavefunctions at time 0 and time t. In light of this, it is possible to convert this expression into Dirac notation to obtain an initial value representation for the time evolution propagator itself in terms of coordinate states, such that

$$\hat{K}_t = \int \mathrm{d}\mathbf{p} \int \mathrm{d}\mathbf{q} \left[\left|\det \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}}\right| (2\pi i\hbar)^{-f}\right]^{\frac{1}{2}} \mathrm{e}^{iS_t(\mathbf{q}, \mathbf{p})/\hbar - i\pi\nu/2} \left|\mathbf{q}_t\right\rangle \left\langle \mathbf{q}\right|. \tag{2.2.6}$$

The change in the terms of the action is allowable as $\mathbf{q}_t = \mathbf{q}_t(\mathbf{q}, \mathbf{p})$, and so $S_t(\mathbf{q}_t(\mathbf{q}, \mathbf{p}), \mathbf{q}) = S_t(\mathbf{q}, \mathbf{p})$.

The above expression gives a method of computationally propagating a wave function in time, provided the roots can be found (which on systems with many degrees of freedom can be computationally expensive). This type of calculation became the basis of much of the later work in semiclassical molecular dynamics and in attempts to accurately model quantum mechanical behaviour, as will be shown later.

2.2.2 The "Frozen Gaussian" Approximation to Quantum Molecular Dynamics

The difficulty of using the basic IVR formulation led Heller [3–5] to propose a system based on "Frozen Gaussian" wave packets. In this system, a swarm of Gaussian wave packets with fixed width parameters (hence 'frozen') were propagated along classical trajectories, which had some success in describing quantum mechanical behaviour in the modelling of vibrational spectra [4]. Further improvements were formulated by Metiu and co-workers [6, 7] who suggested that the wavefunction could be expanded using a basis set of trajectory guided Gaussian functions.

A drawback of Heller's method was that while reasonable results could be obtained for short time dynamics, it "*lacked a rigorous semiclassical basis, in the sense of passing over to the [van Vleck]* $\hbar \to 0$ *limit of the quantum propagator*" [10]. The major improvement on Heller's "Frozen Gaussian" approach was provided by Herman and Kluk [8, 28, 29]. As Heller's derivation had been largely heuristic, Herman and Kluk set out to rigorously derive the "Frozen Gaussian" approximation using the van Vleck propagator as a starting point[9]. This derivation revealed a new pre-factor which had previously been missing from Heller's approach, and which was to be applied to each member of the swarm as well as giving Gaussian smoothing to the van Vleck pre-factor and thus removing the problem of discontinuities around the caustics. Both Kay [9] and Miller [21] have shown how the HK method, as it became known, could be given in terms of the IVR representation, reconciling the "Frozen Gaussian" approximation and the older Initial Value Representation method. The qualities of the "Frozen Gaussian" states place them well in line with the definition of coherent states. Indeed, if applied to a harmonic oscillator potential with a frequency ω, as they most often have been [4, 8, 9, 20, 21, 28], these "Frozen Gaussians" are themselves coherent states. The standard coherent state in coordinate representation is defined as a state which describes non-spreading wave packets for the harmonic oscillator [37]. First formulated by Schrödinger [38] as a minimum uncertainty Gaussian wave packet, this was expanded upon by many, most importantly by von Neumann [39] and later by Glauber [40, 41]. If equation (2.2.6) is given in terms of coherent states, it can be shown that

$$\hat{K}_t = (2\pi\hbar)^{-f} \int \mathrm{d}\mathbf{q} \int \mathrm{d}\mathbf{p} C_i(\mathbf{q},\mathbf{p}) \mathrm{e}^{iS_t(\mathbf{q},\mathbf{p})/\hbar} \left|\mathbf{p}_t\mathbf{q}_t\right\rangle \left\langle\mathbf{p}\mathbf{q}\right|, \tag{2.2.7}$$

where the S_t integral is unchanged from equation (2.2.3) and the HK pre-factor C_i is given by

$$C_i(\mathbf{q},\mathbf{p}) = \left|\frac{1}{2}\left(\frac{\partial\mathbf{q}_t}{\partial\mathbf{q}} + \frac{\partial\mathbf{p}_t}{\partial\mathbf{p}} + \frac{\hbar\gamma}{i}\frac{\partial\mathbf{q}_t}{\partial\mathbf{p}} + \frac{i}{\hbar\gamma}\frac{\partial\mathbf{p}_t}{\partial\mathbf{q}}\right)\right|^{\frac{1}{2}}. \tag{2.2.8}$$

The coordinate space wavefunction for the coherent states is given by

$$\langle \mathbf{x}|\ \mathbf{pq}\rangle = \left(\frac{\gamma}{\pi}\right)^{\frac{f}{4}} \mathrm{e}^{(\gamma/2)|\mathbf{x}-\mathbf{q}|^2} \mathrm{e}^{i\mathbf{p}(\mathbf{x}-\mathbf{q})/\hbar} \tag{2.2.9}$$

with a similar expression for $\langle \mathbf{x}|\ \mathbf{p}_t\mathbf{q}_t\rangle$. As $\gamma \to \infty$ these coherent states become position states, and as $\gamma \to 0$ they become momentum states. It can be seen therefore that in the limit $\gamma \to \infty$ equation (2.2.7) reduces to equation (2.2.6), and similar for the momentum representation of equation (2.2.6). As such, this HK equation is a form of IVR which follows on from the Miller method.

Further modifications to the basic HK method were made by including Monte Carlo sampling, which improved scaling with dimensionality and resulted in many applications of the theory in the following years [42–50]. Even recently, methods which are very much based on developments of the HK method have been used to model quantum interactions [51–59], under names such as Semiclassical Initial Value Representation or the Heller-Herman-Kluk-Kay method. The HK method, and its successors, are still semiclassical models however, which do not model the exact quantum waveform. In addition to this, the HK pre-exponential factor can increase uncontrollably in certain applications, meaning that convergence is only possible with the use of very large numbers of trajectories, a situation which was still the case even fifteen years after the original formulation of the method [22]. In light of this, there was motivation to create a method which could accurately model the quantum mechanics of a system and as such many methods were developed, among these the CCS method.

2.3 Formulation and Refinement of the Coupled Coherent States Method

The Coupled Coherent States method was first developed by Shalashilin and Child in 2000 [60]. In this paper, Shalashilin and Child set forth to propagate the quantum wave function exactly on a coherent state basis drawing from work done by Shalashilin and Jackson [61] and earlier by Metiu and co-workers [6, 7], rather than using the HK propagator which, as they point out, may be regarded as a semiclassical propagator in coherent states Husimi representation [62], since it is a Gaussian smoothing of the van Vleck semiclassical propagator in coordinate representation [60]. The approximation of the integral form of the Schrödinger equation to a finite sum, as is done with this method, leads to the propagation of the wavefunction in a continuous basis of coherent states in phase space, rather than coordinate space as was the case previously. This phase space propagation is particularly useful as the classical energy of a point in phase

space is always known, allowing easy scaling of the Hamiltonian. It also allows for the calculation of quantum amplitudes which, unlike for the HK or Heller approaches, are in principle exact [22]. Another important difference with respect to the HK and other semiclassical methods is that the trajectories along which the wavefunction is propagated are not guided by a classical potential, but instead by a quantum average over the guiding coherent state [63]. Due to the formally exact nature of the CCS model, it was found that a number of semiclassical propagation techniques such as the Heller "Frozen Gaussian" approximation, HK method and "Thawed Gaussians" could be derived as approximate analytical solutions of the CCS integro-differential Schrödinger equation [64]. This integro-differential form of the Schrödinger equation causes the Hamiltonian to appear in the form of a kernel, rather than a matrix [31].

The first formulation of the CCS method exploited special properties of harmonic oscillator coherent states to form a non-orthogonal basis, and was given as a one-dimensional case [60], however in later formulations multi-dimensionality was included [19, 65]. In the following derivation, this later multi-dimensional version is shown.

2.3.1 Governing Equations for the Coupled Coherent States approach

The CCS equations can be found by using a coherent state representation of the wavefunction, known as the Husimi representation, as an initial starting point,

$$\langle q, p|\ \Psi\rangle = \int dx\, \langle p, q|\ x\rangle\, \langle x|\ \Psi\rangle\,, \tag{2.3.1}$$

where the overlap between x and the (p, q) state is the same as for the HK method, shown in equation (2.2.9). The $|q, p\rangle$ notation is changed to the $|z\rangle$ notation, which means that equation (2.2.9) becomes

$$\langle x|\ z\rangle = \left(\frac{\gamma}{\pi}\right)^{\frac{f}{4}} \exp\left[-\frac{\gamma}{2}(x-q)^2 + \frac{i}{\hbar}p(x-q) + \frac{ipq}{2\hbar}\right], \tag{2.3.2}$$

which you will note differs by an insignificant phase factor of $\exp(ipq/2\hbar)$. The $|z\rangle$ state is an eigenstate of the annihilation operator, such that

$$\begin{aligned} \hat{a}\,|z\rangle &= z\,|z\rangle \\ \langle z|\,\hat{a}^\dagger &= \langle z|\,z^*, \end{aligned} \tag{2.3.3}$$

where the creation and annihilation operators can be given in terms of position and momentum operators as

$$\begin{aligned}\hat{a} &= \left(\frac{\gamma}{2}\right)^{\frac{1}{2}} \hat{q} + \frac{i}{\hbar}\left(\frac{1}{2\gamma}\right)^{\frac{1}{2}} \hat{p} \\ \hat{a}^{\dagger} &= \left(\frac{\gamma}{2}\right)^{\frac{1}{2}} \hat{q} - \frac{i}{\hbar}\left(\frac{1}{2\gamma}\right)^{\frac{1}{2}} \hat{p}.\end{aligned} \tag{2.3.4}$$

This lets us define the eigenvalues z as

$$\begin{aligned}z^{(m)} &= \left(\frac{\gamma^{(m)}}{2}\right)^{\frac{1}{2}} q^{(m)} + \frac{i}{\hbar}\left(\frac{1}{2\gamma^{(m)}}\right)^{\frac{1}{2}} p^{(m)} \\ z^{*(m)} &= \left(\frac{\gamma^{(m)}}{2}\right)^{\frac{1}{2}} q^{(m)} - \frac{i}{\hbar}\left(\frac{1}{2\gamma^{(m)}}\right)^{\frac{1}{2}} p^{(m)}.\end{aligned} \tag{2.3.5}$$

The above equations are the one-dimensional versions, representing a point in phase space, and can be easily transformed to the multi-dimensional versions by way of

$$|\mathbf{z}(t)\rangle = \prod_{m=1}^{M} \left|z^{(m)}\right\rangle \tag{2.3.6}$$

for a system of M dimensions. Using this notation it is convenient to use the ordered form of the Hamiltonian, such that

$$\hat{H}(\hat{p},\hat{q}) = \hat{H}(\hat{a},\hat{a}^{\dagger}) = \hat{H}_{\mathrm{ord}}(\hat{a}^{\dagger},\hat{a}). \tag{2.3.7}$$

As a consequence of this, a matrix element of the Hamiltonian on a coherent states basis reduces to

$$\left\langle \mathbf{z}' \right| \hat{H} \left| \mathbf{z} \right\rangle = \left\langle \mathbf{z}' \right| \left. \mathbf{z} \right\rangle H_{\mathrm{ord}}(\mathbf{z}^{*\prime}, \mathbf{z}), \tag{2.3.8}$$

where $H_{\mathrm{ord}}(z^{*\prime(m)}, z^{(m)})$ is the classical analogue of $\hat{H}_{\mathrm{ord}}(\hat{a}^{\dagger},\hat{a})$. Equation (2.3.5) allows z and z^* to be used as the variables of propagation in place of q and p. Through this, the time evolution of $\mathbf{z}$ and $\mathbf{z}^*$ can be found from Hamilton's equations, such that

$$\frac{d\mathbf{z}}{dt} = -\frac{i}{\hbar}\frac{\partial H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z})}{\partial \mathbf{z}^*} \quad , \quad \frac{d\mathbf{z}^*}{dt} = \frac{i}{\hbar}\frac{\partial H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z})}{\partial \mathbf{z}}. \tag{2.3.9}$$

It should be noted that the coherent states form an over-complete basis since any coherent state can be expressed as a superposition of all others. This can be seen by the expression

$$|\mathbf{z}\rangle = \int \frac{\mathrm{d}^2\mathbf{z}'}{\pi^M} \left|\mathbf{z}'\right\rangle \left\langle \mathbf{z}' \right| \left. \mathbf{z} \right\rangle \quad , \quad \frac{\mathrm{d}^2\mathbf{z}}{\pi^M} = \prod_{m=1}^{M} \frac{\mathrm{d}q^{(m)}\mathrm{d}p^{(m)}}{2\pi\hbar}, \tag{2.3.10}$$

where the integration is taken over the entire phase space. The overlap in the above expression can be found by

$$\left\langle \mathbf{z}' \middle| \mathbf{z} \right\rangle = \exp\left[\mathbf{z}^*\mathbf{z} - \frac{\mathbf{z}^{*\prime}\mathbf{z}'}{2} - \frac{\mathbf{z}^*\mathbf{z}}{2}\right] = \prod_{m=1}^{M} \exp\left[z^{*\prime(m)} z^{(m)} - \frac{z^{*\prime(m)} z^{\prime(m)}}{2} - \frac{z^{*(m)} z^{(m)}}{2}\right]. \tag{2.3.11}$$

From equation (2.3.10), one can obtain an expression for the identity operator in the coherent states basis,

$$\hat{I} = \int \frac{\mathrm{d}^2\mathbf{z}'}{\pi^M} \left|\mathbf{z}'\right\rangle \left\langle \mathbf{z}' \right|, \tag{2.3.12}$$

and as such we obtain a form of the Schrödinger equation,

$$\frac{d\Psi(\mathbf{z},t)}{dt} = -\frac{i}{\hbar} \int \left\langle \mathbf{z} \middle| \hat{H} \middle| \mathbf{z}' \right\rangle \Psi(\mathbf{z}',t) \frac{\mathrm{d}^2\mathbf{z}'}{\pi^M}, \tag{2.3.13}$$

which is altered by the inclusion of terms describing nonadiabatic coupling, such that

$$\frac{d\Psi(\mathbf{z},t)}{dt} = \int \left[\left\langle \dot{\mathbf{z}} \middle| \mathbf{z}' \right\rangle - \frac{i}{\hbar} \left\langle \mathbf{z} \middle| \hat{H} \middle| \mathbf{z}' \right\rangle\right] \Psi(\mathbf{z}',t) \frac{\mathrm{d}^2\mathbf{z}'}{\pi^M}. \tag{2.3.14}$$

If a wavefunction representation is used which includes a pre-exponential factor and the classical action, S, i.e.

$$\left\langle \mathbf{z} \middle| \Psi(t) \right\rangle = \Psi(\mathbf{z}(t),t) = C\left(\mathbf{z}(t)\right) \exp\left(\frac{i}{\hbar} S_{\mathbf{z}}\right), \tag{2.3.15}$$

then equation (2.3.14) becomes an exact integro-differential form of the Schrödinger equation in a basis set of moving coherent states,

$$\frac{dC(\mathbf{z}(t))}{dt} = -\frac{i}{\hbar} \int \left\langle \mathbf{z} \middle| \mathbf{z}' \right\rangle \delta^2 H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z}') \exp\left(\frac{i}{\hbar}(S'-S)\right) C(\mathbf{z}(t)) \frac{\mathrm{d}^2\mathbf{z}'}{\pi^M}, \tag{2.3.16}$$

where the classical action is given by

$$S = \int \left[\frac{i\hbar}{2}\left(\mathbf{z}^* \frac{d\mathbf{z}}{dt} - \mathbf{z}\frac{d\mathbf{z}^*}{dt}\right) - H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z})\right] \mathrm{d}t \tag{2.3.17}$$

and

$$\delta^2 H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z}') = H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z}') - H_{\mathrm{ord}}(\mathbf{z}^*,\mathbf{z}) - \frac{\partial H_{\mathrm{ord}}}{\partial \mathbf{z}}(\mathbf{z}'-\mathbf{z}). \tag{2.3.18}$$

The coupling between the basis states in equation (2.3.16) is local and weak, which is an advantage in terms of computational expense. It is also important to recognise that the pre-exponential factor $C(\mathbf{z}(t))$ used is smooth both in space and time; the majority of oscillations encountered during simulation are due to the classical action. This is good as it allows for Monte Carlo sampling of the integral, which would not be possible for an oscillatory wavefunction. By switching the integral in equation (2.3.16) for a finite

sum over N randomly selected basis functions, and using this discretised version of the coherent basis Schrödinger equation in conjunction with similarly discretised versions of the Hamilton's equations (2.3.9) the trajectories can be mapped and the wavefunction propagated computationally from the initial state

$$C_n(\mathbf{z}_n(t=0)) = \langle \mathbf{z}_n(0)| \Psi \rangle \quad \text{for} \quad (n = 1, N). \tag{2.3.19}$$

Using Liouville's theorem, conditions are met allowing the use of importance sampling in this system, meaning that the integration over the initial phase space can be biased towards the peak of the initial wavefunction, and the classical mechanics of the coherent states keeps the basis set localised around the wave function [19]. This can speed up convergence of the simulation.

2.3.2 Applications and Modifications to the Coupled Coherent States method

The Coupled Coherent States method has shown itself to be applicable to many different models. The first application [60] was the simulation of a Morse oscillator in one dimension. Through this simulation it was shown that the CCS method allowed for propagation without the need for the inverting of a near singular overlap matrix, unlike other simulation methods which use a non-orthogonal basis [60]. There was also an advantage over the HK method in that there was no possibility of exponentially growing pre-factors. In this first application a fixed rectangular grid was found to give the highest accuracy, with comparable accuracy being possible from a moving rectangular grid centred on the mean guiding classical path at the expense of the speed of time propagation. At this point Monte Carlo sampling was not used, but the later possibility of including it in a multi-dimensional version of the method was considered. The terms of the derivation were given as resulting from a harmonic oscillator, and this link to a harmonic system carried over to all subsequent formulations of the CCS equations as the properties of the harmonic oscillator are exploited to achieve the phase space qualities of the method. A second application of the method proved that the CCS could accurately simulate a fully quantum interaction, and so was a fully quantum technique, by simulation of one dimensional tunnelling and the handling of zero point energy [22]. These types of simulation are extremely computationally expensive using most semi-classical methods, but this was not the case for the CCS simulation. Monte Carlo and quasi-random Sobol grids were used for the simulation, with stable results being found for the Monte Carlo grid using 2000 trajectories, and for the quasi-random Sobol grid using 500 trajectories for the one dimensional case, showing tunnelling related beating in the autocorrelation function as is shown in "exact" split-operator propagation. The HK

autocorrelation function however had no beating structure. A two-dimensional case was also examined using 10^4 trajectories, but unfortunately convergence could only be seen on timescales less than the beating period. The number of trajectories needed however was an order of magnitude lower than for the HK propagator.

The use of random and quasi-random grids introduced an important feature of the CCS method, in that it does not necessarily require a regular grid on which to propagate the wavefunction. This is in contrast to most simulation methods capable of modelling quantum interactions. In place of a regular grid, the CCS method is capable of describing the wavefunction using a compressed random swarm in phase space. The coherent state grid points travel along averaged trajectories in order to propagate the wavefunction, as is illustrated in figure 2.1. This reduces the computational expense considerably,

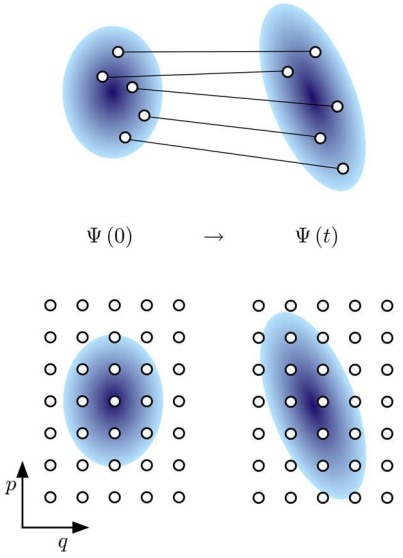


FIGURE 2.1: Comparison between the propagation of a wavefunction represented on a dynamic random grid and that on a regular grid. As can be immediately seen, the computational cost for propagation of the wavefunction on a regular grid will be higher as every grid point must be simulated for all time steps. Since the grid points (coherent states) are trajectory guided for the CCS method, they are capable of fully describing the wavefunction after time t. Original from ref [66]

however requires multiple repeat simulations to obtain a well converged result. In the CCS methods, the initial wavefunction is taken to be a Gaussian wave packet. As this can be shown as a coherent state the initial wavefunction can then be given as $|\Psi(0)\rangle = |\mathbf{z}_0\rangle$, using the coherent state z notation. To construct the initial basis set $|\mathbf{z}_k(t)\rangle$ as a random swarm, it is necessary to calculate a set of basis functions biased towards the centre of the initial wavefunction [67] and sampled from a compressed initial distribution such that

$$F(\mathbf{z}_k) \propto \exp\left[-\alpha_c |\mathbf{z}_k(t) - \mathbf{z}_0|^2\right], \tag{2.3.20}$$

where α_c is an empirical compression parameter related to the width of the initial Gaussian swarm which can be changed depending upon the numbers of basis functions or degrees of freedom of the system.

The next publication on the CCS method [19] introduced a multi-dimensional generalisation of the CCS equations, as detailed in the previous section. This multi-dimensional version of the CCS allowed the scaling properties of the Monte Carlo sampling to be tested as applied to the coherent states, and it was found that the only quantity which detracts from the exact accuracy of the integro-differential Schrödinger equation is the truncation of the integral to a finite sum, provided that the Hamiltonian is re-ordered exactly. The multi-dimensional CCS method was applied to 2D, 6D, 10D, and 14D Henon-Heiles potentials, and an approximately linear scaling was seen for these models, as opposed to quadratic scaling under the HK approach. During work on this simulation it was noticed that the use of importance sampling meant that eventually the coherent basis states separated sufficiently in phase space so as to stop "talking" to each other, and thus uncoupled, imposing a long time limit on the model. With the viability of the multi-dimensional CCS equations confirmed, the method was applied to a real system - the Fermi-resonance and intra-molecular energy transfer in the CHD_3 molecule [65]. In this simulation, it was shown that the CCS method was able to "perform fully quantum simulations of energy flow between the modes in a fashion similar to classical trajectory molecular dynamics methods [65]". Various strengths of the CCS method were highlighted in this work, such as confirming the good scaling with dimensionality that had been mentioned in the previous publication, showing that the method was capable of treating multi-dimensional quantum dynamics, and proving that the method did not rely on the sparsity of the interaction, unlike some other methods. It could also be seen from the simulation that the choice of initial conditions allowed the simulation to be focused on a narrow part of the spectrum, in this case the Fermi-resonance. A further result of this simulation, highlighting one of the main limitations of the CCS method, was that the noise intrinsic in any Monte Carlo simulation causes the autocorrelation function to decay. It was found to be possible however to reduce the noise to a sufficient level to extract eigenvalues from the autocorrelation function. The success of this simulation indicated that the CCS method was able to carry out fully quantum molecular dynamics simulations of realistic systems with many degrees of freedom.

Following this work, the CCS method was applied to the calculation of the infra-red absorption spectra of a 3D model of water trimer torsional dynamics[68], the modelling of the 26D absorption spectrum of pyrazine[69], and multi-dimensional tunnelling in 2, 10 and 20 dimensions[63]. The simulation of the water trimer infra-red absorption spectrum required a reformulation of the equations of propagation in order to change the pre-factor and allow calculation of the Boltzmann operator in imaginary time. The

modelling of the 26D absorption spectrum of pyrazine required propagation over two potential energy surfaces (PESs) rather than one, as was the case for all previous simulations reported, which was achieved by way of a so-called 'mapping' Hamiltonian based on earlier semiclassical simulations by Meyer and Miller [70], changing the 24 vibrational modes of pyrazine into a 26D problem by incorporating two electronic modes relating to the PESs. While this simulation cannot be called entirely successful, it did reproduce some important features of the spectrum. The simulations of multi-dimensional tunnelling confirmed the fully quantum nature of the multi-dimensional CCS approach, showing very good agreement with the results of other simulation methods for symmetric 2D and 10D tunnelling, but showing some discrepancies attributable to the sampling of the initial wave packet in the simulation of asymmetric 20D tunnelling.

One fairly interesting application of the CCS method was to the electron dynamics of the two lowest states of the H_2 molecule [71]. This was achieved through modifications to the propagation equations. It was demonstrated that not only was the CCS method able to model electronic dynamics successfully, but there were also advantages to the using the CCS method for this type of simulation over the Multiple Spawning method of Martínez and Ben-Nun [11, 12, 72, 73], which is also based upon trajectory guided coherent states. The use of the coherent states and a non-orthogonal basis allowed for quantum averaging, which eliminated the problem of Coulombic singularities, replacing them with an error function. This simulation also showed that the CCS technique is well suited for the simulation of collisions, unlike methods based on standard grids or basis sets, since the most important part of the wavefunction is always followed by the trajectory guided grids. The importance of including motion in all 3 dimensions for the electron was shown also when compared with a 1D simulation, and information about the trajectories of re-collision events was shown to be extractable from the simulation. A comparison to this technique was given in a companion paper [74] which gave a version of the diffusion Monte Carlo method in terms of coherent state quantum mechanics. Building upon the work in ref. [71], a method of Fermionic Coupled Coherent States (F-CCS) was developed [75], which served to describe Fermionic molecular dynamics in terms of the CCS framework. As a test of principle, the strong field dynamics of helium atoms were analysed with the simulations of double ionisation events yielding results that were in good agreement with experimental data.

Due to the multiple sampling methods used in the course of the development of the CCS method, a test using three sampling methods was carried out [67]. These three methods were (i) swarm sampling, where coherent states are sampled from a normally distributed random swarm, (ii) pancake sampling where the coherent states are sampled from a swarm which is normally distributed on the real (q) axis but is flat (or infinitely

compressed) on the imaginary (p) axis, and finally (iii) train sampling[1] where a "train" of coherent states are created by propagating a central coherent state backwards and forwards in time and sampling along this propagation, thus creating a basis set where every member follows the same path in phase space. The three sampling methods were tested using a Henon-Heiles potential, and compared against the results of a simulation carried out using the MCTDH method. A compression factor, α_c, was used along with changing the basis set size to improve convergence, a method which gave very promising results. In the same paper, some changes to the CCS equations were reported, and as these equations became the basis for the later Multi-configurational Ehrenfest method, and are used as the basis of simulation in Chapter 4, a brief overview of the changes will be given here; a more rigorous derivation is given in Appendix B.2. The ansatz for the wavefunction is set as

$$|\Psi\rangle = \sum_k A_k \, |\mathbf{z}_k\rangle \,, \tag{2.3.21}$$

with the identity matrix taking the form

$$\hat{I} = \sum_{j,k} |\mathbf{z}_j\rangle \, (\mathbf{\Omega}^{-1})_{jk} \, \langle \mathbf{z}_k| \,, \tag{2.3.22}$$

where $\mathbf{\Omega}^{-1}_{jk}$ is the inverse of the overlap matrix $\mathbf{\Omega}_{jk} = \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle$ with elements $(\mathbf{\Omega}^{-1})_{jk}$. The time evolution of the coherent states is still given by equation (2.3.9), however as the pre-factor has changed, so too will the equations governing it. For ease of propagation, the coefficient A_i in equation (2.3.21) is split into a fast oscillating exponent and a smooth pre-factor $A_k = D_k \mathrm{e}^{iS_k}$ (with the action still controlled by equation (2.3.17)), which results in the time evolution equation

$$\sum_i \langle \mathbf{z}_j | \, \mathbf{z}_i \rangle \, \frac{dD_i}{dt} = -\frac{i}{\hbar} \sum_k \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, \delta^2 H'^*_{\mathrm{ord}}(\mathbf{z}_j, \mathbf{z}_k) D_k \exp\left(i(S_k - S_i)\right) , \tag{2.3.23}$$

where $\delta^2 H'^*_{\mathrm{ord}}(\mathbf{z}_j, \mathbf{z}_k)$ differs slightly from equation (2.3.18) in that now,

$$\delta^2 H'^*_{\mathrm{ord}}(\mathbf{z}_j, \mathbf{z}_k) = H_{\mathrm{ord}}(\mathbf{z}^*_j, \mathbf{z}_k) - H_{\mathrm{ord}}(\mathbf{z}^*_k, \mathbf{z}_k) - i(\mathbf{z}^*_j - \mathbf{z}^*_k)\mathbf{z}^*_k . \tag{2.3.24}$$

A comparison between (2.3.23) and the discretised version of equation (2.3.16) shows that C_k and D_k are related by way of the inverse overlap matrix, and indeed using equation (2.3.22), bearing in mind the initial conditions for C_k from equation(2.3.19), it

[1] Train sampling had previously been used in conjunction with the Ab-Initio Multiple Spawning technique by Ben-Nun and Martínez[76], who referred to them as time displaced basis sets.

can be seen that

$$\begin{aligned}|\Psi(0)\rangle &= \sum_{j,k} |\mathbf{z}_j(0)\rangle \, (\Omega^{-1})_{jk} \, \langle \mathbf{z}_k(0)| \, \Psi(0)\rangle \\ &= \sum_{j} |\mathbf{z}_j(0)\rangle \, (\Omega^{-1})_{jk} C_k(0) \qquad\qquad \therefore D_k(0) = \sum_j (\Omega^{-1})_{jk} C_k(0) e^{-iS_k(0)}. \\ &= D_k(0) \mathrm{e}^{iS_k(0)} \, |\mathbf{z}_j(0)\rangle\end{aligned} \tag{2.3.25}$$

This is the form of the equations which were later developed into the basis of the Multi-Configurational Ehrenfest method (MCE), which will be covered in the next section.

2.4 The Multi-Configurational Ehrenfest method

The Multi-Configurational Ehrenfest method can be seen as a generalised form of the CCS method applied to the interaction of more than one PES for non-adiabatic dynamics. It is a relatively new technique in comparison to the others considered in this chapter, first presented in 2009 [23], but has given very encouraging results in all the applications reported. The method is based on the Ehrenfest approximation using a self-consistent mean-field formalism developed originally by Billing [77]. The Ehrenfest-Billing approach splits the Hamiltonian into a quantum system, a classical bath and an interaction potential such that

$$\hat{H}(\hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{p}, \mathbf{q}) = \hat{H}^{sstm}_{qntm}(\hat{\mathbf{P}}, \hat{\mathbf{Q}}) + \hat{V}_{int}(\hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{p}, \mathbf{q}) + H^{bth}_{cl}(\mathbf{p}, \mathbf{q}), \tag{2.4.1}$$

where $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$ are the quantum operators of the system and $\mathbf{p}$ and $\mathbf{q}$ are the classical position and momentum vectors for the bath. The trajectories of the classical bath are found through Hamilton's equations for $\mathbf{p}$ and $\mathbf{q}$ applied to the Ehrenfest Hamiltonian,

$$H^{\mathrm{Ehr}} = \left\langle \Psi^{sstm}(t) \right| \hat{H}(\hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{p}, \mathbf{q}) \left| \Psi^{sstm}(t) \right\rangle, \tag{2.4.2}$$

where

$$\left|\Psi^{sstm}(t)\right\rangle = a^{(1)}(t) \left|\varphi_1^{sstm}\right\rangle + a^{(2)}(t) \left|\varphi_2^{sstm}\right\rangle + ..., \tag{2.4.3}$$

and the time evolution of the coefficients, giving the time evolution of the quantum system, is found from

$$\frac{da^{(l)}}{dt} = -\frac{i}{\hbar} \sum \left\langle \varphi_l \right| \hat{H}(\hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{p}, \mathbf{q}) \left| \phi_j \right\rangle a^{(j)}. \tag{2.4.4}$$

The Ehrenfest-Billings approach has had many successful applications in the past, however by definition as an approximation it has limits on its accuracy. Drawing on the

success of the CCS method, an exact method was developed using the Ehrenfest dynamics equations (2.4.1-2.4.4) as a starting point. Unlike regular Ehrenfest dynamics, the MCE technique treats all degrees of freedom, including those of the classical bath, at a fully quantum level making MCE a fully quantum technique. The use of coherent states to model the bath was already known to be efficient from previous tests using the CCS method where a set of trajectory guided coherent states was able to accurately model tunnelling in a double well potential coupled with a harmonic bath [63]. It had also been demonstrated by Martinazzo *et al.*[78], whose work indicated the ability of single coherent states to describe a bath of thousands of degrees of freedom when the system is described by a regular basis.

2.4.1 Governing Equations for the Multi-Configurational Ehrenfest method

Two formulations of the MCE equations have been presented to date, the first in the 2009 paper which introduced the method, [23] and then the second in a 2010 paper [24] which made important modifications to the propagation system. As both formulations are used in Chapter 3, the equations of both will be presented here. These equations are given using atomic units, i.e. $\hbar = 1$. The equations are formulated for the interaction of two PESs however they can be easily generalised for more, and when applied to a single PES they collapse down to the CCS equations given earlier. A brief outline of the ways in which these equations are derived is given here, and a more robust derivation is given in Appendix B.

2.4.1.1 The Governing Equations for the First Formulation of the Multi-configurational Ehrenfest method (MCEv1)

The starting point for the first formulation of the MCE method, which shall be referred to as MCEv1, is the ansatz of the wavefunction. This ansatz is given by

$$\begin{aligned} |\Psi(t)\rangle &= \sum_{k=1,N} |\psi_k(t)\rangle \\ &= \sum_{k=1,N} \left(a_k^{(1)}(t)\,|1\rangle + a_k^{(2)}\,|2\rangle\right)|\mathbf{z}_k(t)\rangle\,, \end{aligned} \tag{2.4.5}$$

where the orthonormal number states make up the quantum system states and the bath states are supplied by the set of $|\mathbf{z}_k(t)\rangle$. The individual configurations are coupled through the amplitudes $a_k^{(r)}$, and so these configurations are not normalised. The wavefunction therefore is described by a linear combination of several configurations $|\psi_k(t)\rangle$.

Propagation of the wavefunction is carried out by way of the simultaneous propagation of $|\mathbf{z}_k(t)\rangle$, $a_k^{(1)}(t)$, and $a_k^{(2)}(t)$. The time dependence for $|\mathbf{z}_k(t)\rangle$ is found through applying a variational principle [79] to the single configuration Lagrangian of the system, and the time dependence for the $a_k^{(1,2)}(t)$ amplitudes is found through the multi-configurational Lagrangian. The single configurational Lagrangian in given by

$$L = \left\langle \phi_k(t) \middle| i\frac{\hat{\partial}}{\partial t} - \hat{H} \middle| \phi_k(t) \right\rangle, \tag{2.4.6}$$

where unlike for the earlier CCS two PES simulation, the Hamiltonian is not a 'mapping' Hamiltonian but instead is

$$\hat{H} = \begin{vmatrix} H_{\text{ord}}^{(11)} & H_{\text{ord}}^{(12)} \\ H_{\text{ord}}^{(21)} & H_{\text{ord}}^{(22)} \end{vmatrix}, \tag{2.4.7}$$

which is similar to the Hamiltonian used by Worth and coworkers [80, 81] in previous simulations of the vibration of pyrazine in that the off diagonal elements provide the coupling between the two PESs. Expansion of the Lagrangian introduces the Ehrenfest Hamiltonian which for a system with 2 electronic states takes the form

$$\begin{aligned} H^{\text{Ehr}} &= \frac{\left\langle \phi \middle| \hat{H} \middle| \phi \right\rangle}{\sum_r \left(a^{(r)*} a^{(r)} \right)} \\ &= \frac{\left\langle \mathbf{z} \middle| \hat{H}^{(11)} \middle| \mathbf{z} \right\rangle a^{(1)*} a^{(1)} + \left\langle \mathbf{z} \middle| \hat{H}^{(22)} \middle| \mathbf{z} \right\rangle a^{(2)*} a^{(2)}}{a^{(1)*} a^{(1)} + a^{(2)*} a^{(2)}} \\ &+ \frac{\left\langle \mathbf{z} \middle| \hat{H}^{(12)} \middle| \mathbf{z} \right\rangle a^{(1)*} a^{(2)} + \left\langle \mathbf{z} \middle| \hat{H}^{(21)} \middle| \mathbf{z} \right\rangle a^{(2)*} a^{(1)}}{a^{(1)*} a^{(1)} + a^{(2)*} a^{(2)}}. \end{aligned} \tag{2.4.8}$$

It should be noted that the Ehrenfest Hamiltonian is similar in spirit to the ordered Hamiltonian used in the CCS method, in that for a single electronic state $\left\langle \phi \middle| \hat{H} \middle| \phi \right\rangle \equiv \left\langle z \middle| \hat{H} \middle| z \right\rangle$. Applying the Lagrange equations of motion for z^* to the single configuration Lagrangian gives

$$i\dot{\mathbf{z}} = \frac{\partial H^{\text{Ehr}}}{\partial \mathbf{z}^*}, \tag{2.4.9}$$

which it should be noted is exactly the Hamilton's equations for $|z\rangle$ notation with an Ehrenfest Hamiltonian. When the Lagrange equations of motion for the $a_k^{(1)}$ amplitudes are applied to the multi-configurational Lagrangian we get

$$\begin{aligned} \sum_k i a_k^{(1)} \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle - \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, H^{(11)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(1)} - \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, H^{(12)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(2)} + \\ i \left[\left(\mathbf{z}_j^* - \mathbf{z}_k^* \right) \dot{\mathbf{z}}_k + \frac{\dot{\mathbf{z}}_k \mathbf{z}_k^*}{2} - \frac{\mathbf{z}_k \dot{\mathbf{z}}_k^*}{2} \right] \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, a_k^{(1)} = 0. \end{aligned} \tag{2.4.10}$$

Introducing the expression

$$a^{(r)} = d^{(r)} \mathrm{e}^{iS^{(r)}} \tag{2.4.11}$$

allows for the reformulation of equation (2.4.10) in terms of a smooth pre-exponential factor, such that in general

$$\begin{aligned}\sum_k i\dot{d}_k^{(r)} e^{iS_k^{(r)}} \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle = &\sum_k \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, \delta^2 H'^{(rr)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) d_k^{(r)} e^{iS_k^{(r)}} \\ &+ \sum_k \sum_{s \neq r} \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \, H^{(rs)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) d_k^{(s)} e^{iS_k^{(s)}},\end{aligned} \tag{2.4.12}$$

where the classical action is given by

$$S^{(l)} = \int \left[i \frac{\dot{\mathbf{z}}\mathbf{z}^* - \mathbf{z}\dot{\mathbf{z}}^*}{2} - \left\langle \mathbf{z} \right| H^{(rr)} \left| \mathbf{z} \right\rangle \right] \mathrm{d}t, \tag{2.4.13}$$

and where

$$\delta^2 H'^{(rr)} = H^{(rr)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) - H^{(rr)} \left(\mathbf{z}_k^*, \mathbf{z}_k \right) - i \left(\mathbf{z}_j^* - \mathbf{z}_k^* \right) \dot{\mathbf{z}}_k. \tag{2.4.14}$$

Through equation (2.4.12) both the configurations and the potential energy surfaces are coupled. As such, for the MCEv1 method equations (2.4.12), (2.4.9) and (2.4.13) make up the propagation scheme for the wavefunction.

2.4.1.2 The Governing Equations for the Second Formulation of the Multi-configurational Ehrenfest method (MCEv2)

The second formulation of the Multi-Configurational Ehrenfest method, referred to here as MCEv2, was the result of modifications to the ansatz (Eq. 2.4.5). These modifications were needed as in the original formulation of the method the coefficients of the system basis states were coupled, not just within the same configuration (intraconfigurational), but also across configurations (interconfigurational), making the bath coherent states effectively coupled also. To remedy this the coupling of the configurations is separated from the coupling of the potential energy surfaces by the inclusion of an extra amplitude. This changes the ansatz to

$$\begin{aligned}|\Psi(t)\rangle &= \sum_{k=1,N} D_k(t) \, |\phi_k(t)\rangle \\ &= \sum_{k=1,N} D_k(t) (a_k^{(1)}(t) \, |1\rangle + a_k^{(2)} \, |2\rangle) \, |\mathbf{z}_k(t)\rangle \, .\end{aligned} \tag{2.4.15}$$

Now, propagation of the wavefunction is carried out by way of the simultaneous propagation of $|\mathbf{z}_k(t)\rangle$, $a_k^{(1)}(t)$, $a_k^{(2)}(t)$ and $D_k(t)$. As the amplitudes $a_k^{(r)}$ act only within a single configuration, the time dependence for $|\mathbf{z}_k(t)\rangle$, $a_k^{(1)}(t)$ and $a_k^{(2)}(t)$ are found through applying a variational principle to the single configuration Lagrangian of the system, and the time dependence for $D_k(t)$ is given through application to the Schrödinger equation

as was done for C_k in the CCS equations. Coupling between the configurations is therefore only present in the propagation of D_k. As a result of the separation of the couplings the individual configurations ($|\phi_k\rangle$ in eq. 2.4.15) are normalised, i.e. $\sum_r \left|a_k^{(r)}\right| = 1$. The time evolution of the coherent basis is unchanged from the MCEv1 method and still given by equation (2.4.9), however the equations for the amplitudes are changed. Applying the Lagrange equations of motion for $a^{(r)}$ to the single configuration Lagrangian gives

$$\dot{a}^{(r)} = i\left[i\frac{\dot{\mathbf{z}}\mathbf{z}^* - \mathbf{z}\dot{\mathbf{z}}^*}{2} - \left\langle \mathbf{z}\right| H^{(rr)} \left|\mathbf{z}\right\rangle\right] a^{(r)} - i\sum_{s\neq r}\left\langle \mathbf{z}\right| H^{(rs)} \left|\mathbf{z}\right\rangle a^{(s)}. \tag{2.4.16}$$

If we again introduce equation (2.4.11), equation (2.4.16) becomes

$$\dot{d}^{(r)} = -i\sum_{s\neq r}\left\langle \mathbf{z}\right| H^{(rs)} \left|\mathbf{z}\right\rangle d^{(s)} \mathrm{e}^{i(S^{(s)}-S^{(r)})}, \tag{2.4.17}$$

where the classical action is still given by equation (2.4.13). Derivation of the time evolution of the pre-exponential factor D_k begins with the Schrödinger equation

$$\frac{d\left|\Psi\right\rangle}{dt} = -i\hat{H}\left|\Psi\right\rangle \tag{2.4.18}$$

and substituting in the first part of equation (2.4.15) such that

$$\sum_i \frac{dD_i(t)}{dt}\left|\varphi_i(t)\right\rangle = -i\sum_k D_k(t)\hat{H}\left|\varphi_k(t)\right\rangle - \sum_k D_k(t)\left|\frac{d\varphi_(t)}{dt}\right\rangle, \tag{2.4.19}$$

and hence

$$\sum_i \frac{dD_i(t)}{dt}\left\langle\varphi_j(t)\right|\left.\varphi_i(t)\right\rangle = -i\sum_k D_k(t)\left[\left\langle\varphi_j(t)\right|\hat{H}\left|\varphi_k(t)\right\rangle - i\left\langle\varphi_j(t)\right|\left.\frac{d\varphi_k(t)}{dt}\right\rangle\right]. \tag{2.4.20}$$

Using the results from the equations for the other amplitudes this can be simplified to

$$\begin{aligned}\sum_i \frac{dD_i(t)}{dt}\left\langle\varphi_j(t)\right|\left.\varphi_i(t)\right\rangle = -i\sum_k \Big[&\left\langle\varphi_j(t)\right|\hat{H}\left|\varphi_k(t)\right\rangle - \left\langle\mathbf{z}_j(t)\right|\left.\mathbf{z}_k(t)\right\rangle H_{jk} \\ &-i\left\langle\varphi_j(t)\right|\left.\varphi_k(t)\right\rangle\left(\mathbf{z}_j^*(t),\mathbf{z}_k^*(t)\right)\dot{\mathbf{z}}_k(t)\Big] D_k(t) \\ &= -i\sum_k \Delta^2\langle H\rangle_{jk} D_k(t),\end{aligned} \tag{2.4.21}$$

where

$$\Delta^2\langle H\rangle_{jk} = \left\langle\varphi_j(t)\right|\hat{H}\left|\varphi_k(t)\right\rangle - \left\langle\mathbf{z}_j(t)\right|\left.\mathbf{z}_k(t)\right\rangle H_{jk} - i\left\langle\varphi_j(t)\right|\left.\varphi_k(t)\right\rangle\left(\mathbf{z}_j^*(t),\mathbf{z}_k^*(t)\right)\dot{\mathbf{z}}_k(t) \tag{2.4.22}$$

and

$$\begin{aligned} H_{jk} =& a_j^{(1)*} \left\langle \mathbf{z}_k(t) \right| \hat{H}^{(11)} \left| \mathbf{z}_k(t) \right\rangle a_k^{(1)} + a_j^{(1)*} \left\langle \mathbf{z}_k(t) \right| \hat{H}^{(12)} \left| \mathbf{z}_k(t) \right\rangle a_k^{(2)} \\ &+ a_j^{(2)*} \left\langle \mathbf{z}_k(t) \right| \hat{H}^{(21)} \left| \mathbf{z}_k(t) \right\rangle a_k^{(1)} + a_j^{(2)*} \left\langle \mathbf{z}_k(t) \right| \hat{H}^{(22)} \left| \mathbf{z}_k(t) \right\rangle a_k^{(2)}. \end{aligned} \tag{2.4.23}$$

As such for the MCEv2 method equations (2.4.17), (2.4.9), (2.4.13) and (2.4.21) make up the propagation scheme for the wavefunction.

Due to the separation of the intraconfigurational and interconfigurational couplings, the propagation equations for the single configurations (eqs. (2.4.17), (2.4.9) and (2.4.13)) can be calculated independently of each other, with the expensive cross-configurational equation (2.4.21) calculated separately. This means that the MCEv2 method is suitable for on-the-fly *ab initio* calculations such as would be necessary for the simulation of the dynamics of small molecules. The inclusion of this property in the MCE method was the main motivation for the reformulation of the equations.

2.4.2 Applications of the Multi-Configurational Ehrenfest approach

The first application of the MCE model in the introductory article was to simulate the spin boson model [23]. This is a typical system-bath model with applications to photonic crystals, quantum dots, decoherence and quantum information, among others. Some extremely good results were available for comparison from the work of Wang and Thoss [82, 83] who simulated a system with thousands of degrees of freedom without approximations using MCTDH, the only method at that time with abilities sufficient to model systems on this scale. The simulations with the MCEv1 formulation showed that a basis set of only 50-200 Ehrenfest configurations was sufficient to accurately reproduce the MCTDH calculations using up to 2000 degrees of freedom. This result was encouraging to say the least, as the accuracy of the MCTDH method had previously been outside the reach of almost all alternatives.

The follow-up paper in 2010 used the MCEv2 formulation to simulate the vibrational modes of pyrazine [24] with a much greater success than achieved with the CCS method [69]. As mentioned earlier, this paper included modifications to the MCE equations with an eventual aim of looking at *ab initio* dynamics. For the full 24D model, good results were obtained with only 250 trajectories per state when compared against results from MCTDH and G-MCTDH simulations. This number was reduced through an optimisation process reported the following year [32], which brought it down to 34 trajectory-guided configurations. Also reported in the same paper was a set of simulations to model the sticking of hydrogen to a surface, which can be modelled as a Morse oscillator coupled to a harmonic bath of phonons, however due to an approximation in

the form of the density operator, the MCE results for the sticking probability were 15% below MCTDH results for the same system. In this paper it was also suggested that the MCE approach could be applied to *ab initio* environments in light of the independence of the trajectories from each other, and their relatively slow divergence. This was applied by Saita in 2012 [26], who used the MOLPRO electronic structure program to calculate the potential energy surfaces on the fly, and applied the technique to the simulation of the excited state dynamics of ethylene. The results compared very favourably to simulations using the AIMS technique despite the difference in trajectory guiding equations. The adaptation to an *ab initio* method makes the technique easily applicable to large molecules. This was shown in a 2013 paper by the same author [27] where the ultrafast photodynamics of pyrrole were calculated, showing a good agreement with earlier calculations using surfaces from the CASSCF method corrected by the MRCI method. In addition to this, the results showed a hitherto unknown photodisassociation mechanism whereby the wave packet follows the first excited electronic state after the initial dissociation of the H atom, before relaxing to the ground state very soon after by way of another intersection reached by the vibrational motion of the pyrrole radical ring.

A further application of the MCE method was demonstrated by applying the MCEv1 method and its success with the spin boson model to the problem of the Choi fidelity of a unitary controlled-Z gate by Ye, Shalashilin and Serafini [84], which is a difficult system to model outside of the perturbative regime. As well as the spin boson Hamiltonian, the rotating-wave Hamiltonian was also used which is an extremely useful Hamiltonian in the fields of quantum information and quantum optics. In using this rotating-wave Hamiltonian, it was found that when mediated by a discretised Ohmic bath, the counter-rotating terms improve the fidelity. The analysis did however raise some questions regarding the effectiveness of the method in the strong coupling regime as convergence required limiting of the coupling strengths.

2.5 Comparison with Other Simulation Techniques

2.5.1 The MCTDH Family of Simulation Methods

When considering the benefits of the CCS and MCE methods it is standard practice to speak in terms of computational expense and of accuracy, with accuracy being the more important quality. In terms of accuracy, due to the fully quantum nature of the CCS method, there are definite advantages to using CCS over the semiclassical methods discussed in section 2.2. There exist however other alternatives, and chief among

these is the Multi-Configurational Time Dependent Hartree (MCTDH) method. The MCTDH method was introduced in 1990 by H.-D. Meyer and co-workers [16] as a way of introducing correlation between the dimensions for a multi-dimensional wavefunction to the earlier Time Dependent Hartree method, and has since become established as one of the most favoured simulation methods. In the MCTDH method the variational principle is applied to an entire multi-configurational wavefunction, constructed as a sum of orthonormal single particle functions with time-dependent amplitudes. Through application of the Dirac-Frenkel variational principle, one can solve the Time Dependent Schrödinger equation giving a fully variational set of equations of motion for *all* parameters coupled through mean field operators [17]. As all possible configurations are built the method is fully quantum however as a result the method is subject to exponential scaling, albeit with a small base compared with the standard methods. The small base of exponentiation makes the method practicable despite the scaling problems, and so allows systems with many degrees of freedom to be treated with extremely high levels of numerical accuracy [18], provided sufficient computation resources are available. The accuracy possible with MCTDH and the later modifications of this method is such that it has become a benchmark for most other methods, CCS included.

The MCTDH method has been applied in many situations including calculating the spectra of various processes [85–87], modelling relaxation of molecules with many degrees of freedom [80, 81], calculating reaction rates [88], tunnelling [89], and simulating the behaviour of the spin boson model [82, 83]. Modifications to the standard MCTDH equations to take advantage of the presence of many identical particles has resulted in the formulation of the Multi-Configurational Time Dependent Hartree method for Bosons (MCTDHB) [90, 91] and the Multi-Configurational Time Dependent Hartree-Fock method (MCTDHF)[92–94] which is optimised for use with fermionic systems. This has allowed the MCTDH methods to be extended to applications such as the splitting of a Bose Einstein Condensate [95, 96], laser-matter interactions [97–100], simulation of quantum carpets [101] and the excited state dynamics of molecules [102]. Further modifications to the MCTDH method have allowed hybrids of the MCTDHF and MCTDHB methods capable of simulating various Bose/Fermi mixtures[103] and hybrids of the MCTDH and Optimal Control Theory able to simulate unitary quantum gates [104].

One of the most important modifications to the MCTDH method is the Multi-Layer Multi-Configurational Time Dependent Hartree or ML-MCTDH method [83]. This method takes the single particle functions of MCTDH and expresses them using a time-dependent multi-configurational expansion resulting in an extra layer of time dependent coefficients to be propagated. This allows the method to treat systems of up to a few hundred or thousand degrees of freedom, which is much larger than was previously

possible with MCTDH, although due to the larger overhead this method is less efficient for systems with fewer degrees of freedom [25]. The ML-MCTDH method has had many successful applications, notable among those being the accurate modelling of pyrazine and a 1458D Henon-Heiles potential [25], modelling the Boltzmann operator in a time-correlation function [105], proton transfer in condensed phase [106], photo-induced electron transfer [107], and the spin boson model at zero temperature with very high dimensionality[108].

A different set of modifications, inspired by the earlier semiclassical simulation methods, made use of Gaussian wave packets in the MCTDH framework. In this method, dubbed G-MCTDH, some or all of the single particle functions of standard MCTDH are replaced by parametrised Gaussian functions which are then applied to the variational principle. The method was first proposed in 1999 [109] yet the first application was not published until 2003 [110] which showed that the use of Gaussians simplified the equations of motion due to Ehrenfest's theorem (stating that the centre of a Gaussian will move classically), and also showed that using this method the potential no longer had to be known globally, but could be known locally instead meaning that a large regular grid was no longer necessary [111]. The use of Gaussians also had the effect of removing the orthogonality of the basis functions, however it was found that only a few Gaussians were needed to fully describe the wave function. The G-MCTDH method was later applied to such systems as a system-bath model with up to 60 harmonic bath oscillators [112], and to the absorption spectrum of pyrazine [113]. A special case of the G-MCTDH method, not limited to system-bath dynamics, is the vMCG method. The vMCG method is equivalent to the G-MCTDH method under the conditions that only a set of Gaussians is used as a basis, with no need for the coupling mean field operators present in standard MCTDH. The vMCG method was introduced in 2004 [114] and has received a modest amount of attention since, showing itself able to perform direct dynamics calculations using on-the-fly electronic structure calculations [115]. The equations of the vMCG method need not be presented in the style of MCTDH, as was shown first by Shalashilin and Burghardt in 2008 [116] and then more comprehensively five years later by Ronto and Shalashilin [117], who reformulated the working equations of vMCG in the style of the CCS equations. It should be noted that like CCS the wavefunction is expressed in terms of Gaussians yet unlike CCS this is a fully variational method where all treatments are quantum in nature.

2.5.2 The Multiple Spawning Family of Simulation Methods

Another family of methods which also has links to the CCS method is the Multiple Spawning (MS) group which comprises Full Multiple Spawning (FMS) and the later *ab*

initio Multiple Spawning (AIMS). FMS was first introduced in 1996 [11] as a method of modelling multi-electronic state dynamics but in an almost classical framework. The general idea behind MS is that classical mechanics can be used to generate a basis set within which the nuclear Time Dependent Schrödinger Equation can be solved [11–13]. "Frozen Gaussians" are used for the basis functions which travel along classical trajectories. The defining feature of MS is that a wave packet, upon reaching an area of strong nonadiabatic coupling between two potential energy surfaces will not undergo surface hopping as would occur in a semiclassical method. Instead, a "child" wave packet will be "spawned" on the other PES and both "parent" and "child" will propagate together for a time. This means that effectively for a short time while the "parent" and "child" have a non-zero overlap the wavefunction can be said to be in a superposition of both states. In situations where multiple traversals occur over a short time period such as the quenching of Na^*+H_2 [73, 118] this type of system can be extremely useful as it allows quantum mechanical behaviour to be approximated in an almost classical manner. While there are similarities to the surface hopping method of Tully [119], it has been shown that branching ratios are better converged under FMS [120, 121] as the wavefunction normalisation is automatically conserved [72]. Although FMS was initially designed for electron dynamics, due to the quasi-quantum nature of the simulation it has also shown itself adept at simulating tunnelling [122], photo-dissociation [123] and quantum non-adiabatics [124].

A modification of FMS, hinted at originally in the first MS paper [11] is *Ab Initio* Multiple Spawning (AIMS). The first application was in 1998 [14] where *cis-trans* photoisomerisation of ethylene was studied using an *ab initio* potential. It soon became apparent that AIMS was a very powerful tool for non-adiabatic transitions [15] in situations where the potential energy surfaces and their non-adiabatic couplings are not known beforehand including simulations of non-saturated hydrocarbons, charge transfer in excited small hydrocarbons such as 1,3-Butadiene [125] and chromophores of photoactive proteins [126] to name a few, and the acceptance of this method has been such that it has been included in the widely used MOLPRO software [127]. Although the design of the system has high flexibility while keeping a low computational cost [128], the fact remains that this method, rooted as far as it is in classical mechanics, has problems with high accuracy when more than a very small number of degrees of freedom are included (although for a couple of degrees of freedom the FMS method has the ability to return results which border on numerically exact [15]). One of the main impediments to accuracy for the FMS and AIMS systems is the parent-child coupling which begins to decay as the trajectories move away from an intersection [32]. While attempts have been made to optimise the initial conditions so as to maximise this coupling [129], it is still an issue which limits the applications available to FMS and AIMS.

2.5.3 Comparisons Between the CCS, MCTDH and MS Methods

It has been well documented that a properly constructed problem acted on by the MCTDH or ML-MCTDH methods can be taken to be numerically exact [18, 105]. These methods, and the G-MCTDH and vMCG methods, have extremely high accuracy, being fully variational methods, and so have often been used as benchmarks against which other methods can be checked [23–25]. It is however the case that the exponential scaling needed for the accurate simulation of a system under MCTDH combined with the need for a certain analytical form of the PES makes the MCTDH method and the related methods highly computationally expensive [32]. The attraction of the CCS method, and the multi surface generalisation MCE especially, is that comparable accuracy is possible [23, 24] while keeping the computational cost much lower [66]. It should be noted however that in a recent comparison between single surface CCS and the CCS style reformulation of vMCG running on even footing, the expense and accuracy were comparable for some applications [117].

At the other end of the scale, the strong classical flavour of AIMS and FMS allows simulations to be run under this method with lower computational expense than the CCS or MCTDH methods. It was never the intention of this method to be numerically exact however, and as such in terms of accuracy both of the more expensive methods are superior [66, 71]. In addition to the accuracy of the system, the CCS method, or in this case the MCE method as there is more than a single PES, has the advantage over the MS method that the length of time during which quantum behaviour can be simulated is greater. This is due to the coupling between the "parent" and "child" Gaussians in MS. As mentioned earlier the coupling between "parent" and "child" is only valid for a short amount of time. Once the coupling is lost the system becomes equivalent to the "Frozen Gaussian" approximation and thus can only give semiclassical results. This is true of the MCE method also, except that the basis functions stay close together and well coupled for much longer thus preserving quantum behaviour longer. This behaviour is illustrated in figure 2.2.

In light of the above comparisons, it can be said that the CCS and MCE methods represent a comfortable middle ground between low expense yet low accuracy methods and high accuracy yet high expense methods, however there is promise that this position could be improved. Indeed it is the aim of the continuing development of the MCE and CCS methods that a comparable level of accuracy could be achieved in simulation as is available to the MCTDH methods while preserving a low computational expense, an aim which is made all the more possible when considering that for some particular simulations using the CCS and MCE methods, results have been obtained which were indistinguishable from the MCTDH results [23, 67].

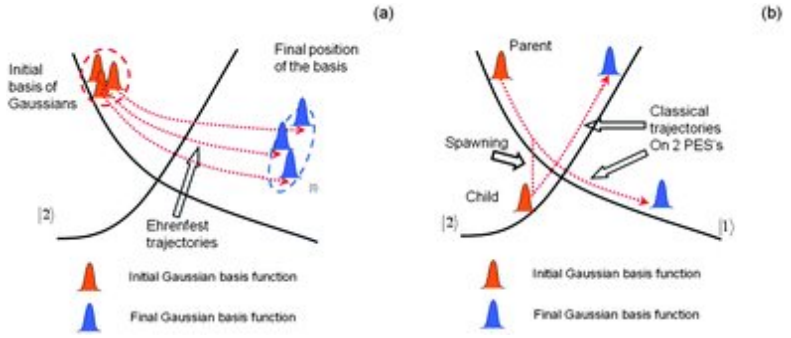


FIGURE 2.2: Comparison between MCE trajectories (a) and MS trajectories (b) in an area near the intersection between two PESs. The grouping of the MCE trajectories stays compact for a comparatively long time, thus preserving coupling and quantum behaviour. The coupling between a MS "parent" and "child" however is quickly lost as the Gaussian bases rapidly separate. Initial and final positions of the bases are shown as red and blue respectively. Original from ref [32]

2.6 Concluding Remarks

This section has described the Coupled Coherent States group of methods, showing the successes and limitations of these methods, as well as the background behind them. The generalisation of the CCS method to the MCE method marked a clear and important step up, and the results published show a system with the benefits of a fully quantum approach but with a much lower computational expense. There is a common feature of all trajectory guided methods, in that once the basis functions become sufficiently separated they lose coupling and become equivalent to the Heller "Frozen Gaussian", misguiding the basis by classical dynamics. As such the use of correct sampling methods become important so as to keep the coupling and position the basis in the right place. The range of possible applications of the CCS family of methods has to date been very promising, with the recent applications to problems in quantum information, fermionic molecular dynamics, and *ab initio* molecular dynamics showing very encouraging results. More recent enhancements of the CCS group of methods are in development or planned, including the development of a two-layer MCE method which will be used on systems where there are a small number of important degrees of freedom in a larger system such as assymetric tunnelling where the tunnelling mode can be treated in a more quantum manner, or 24D pyrazine where there are 4 important modes which can be propagated differently to the rest. A second very important enhancement is the use of basis set cloning which has been developed by D. Makhov and coworkers [1, 2] which draws on the AIMS technique to increase the size of the basis set when needed through cloning particular basis functions. This technique will be discussed in much greater detail in the following sections.

Chapter 3

Adaptive Basis Sets for Better Convergence with the Multi-Configurational Ehrenfest method (MCEv2)

3.1 Introduction

The Multi-Configurational Ehrenfest method has shown itself to be a useful tool for the simulation of quantum mechanics. The first formulation of the MCE method was tested on the spin boson model with very good results, producing results which were almost indistinguishable from earlier numerically exact calculations [23]. While this formulation showed promise, it was subject to some drawbacks. Firstly, as the couplings between the trajectories and between the electronic states were combined into a single set of variables, it was a property of the trajectories that they would "push" on each other. There was also the problem that the individual basis functions were unable to be run separately from each other, something which can be very useful when dealing with electronic structure calculations. To remedy this, a second formulation of the MCE equations was developed [24], which promised greater stability in the simulations as well as an ability to run the single configuration equations independently, allowing use with *ab initio* "on-the-fly" direct dynamics calculations. This second formulation has shown itself to be capable of simulating various systems, especially the dynamics of small organic molecules [24, 26, 27]. Improvements to the MCEv2 equations have been made also, using the cloning of basis functions in a manner reminiscent of the AIMS method to account for problems in regions where the electronic states are decoupled.

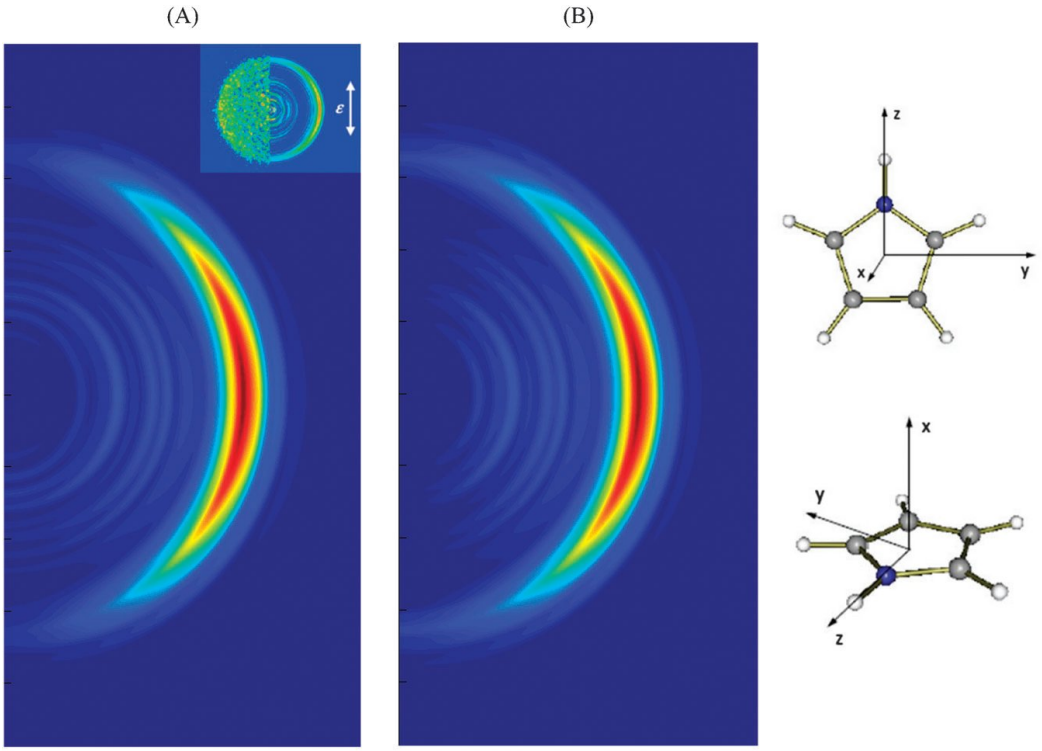


FIGURE 3.1: Simulated velocity map image for the photodisociation of pyrrole with respect to laser pulse polarisation, assuming a transition dipole moment pointed in x(A) and y(B) directions, calculated using the AIMC method, taken from ref [2]. Definition of the axes is supplied also, and a comparison to experimental data is given inset for the transition dipole moment pointed in the x direction. This shows that the AIMC method is capable of reproducing the main features shown in the experimental data.

This method, dubbed *ab initio* Multiple Cloning (AIMC) was developed by Makhov *et al.* [1, 2] and has shown that the MCEv2 equations are capable of accurately reproducing experimental results, such as is shown in figure 3.1. Despite this, there has never been a direct comparison of the two formulations of the MCE method on a system with high dimensionality.

In this chapter this comparison is made using the spin boson model, a high dimensional two state system, and a surprising discrepancy is observed. It is seen through an investigation and discussion of the discrepancies between the two formulations that they are due to a misguiding of the coherent states resulting in the basis set not covering a large enough area of phase space to properly describe the quantum mechanics of the system. The existence of this discrepancy becomes important when one considers that the second formulation of the MCE method is used in "on-the-fly" *ab initio* calculations. This discrepancy shows the need for modifications to the second formulation of the MCE method. It will be shown that the modifications first implemented in the MCE method by Makhov *et al.* [1, 2] are capable of resolving this discrepancy, thus confirming these results.

3.2 The Spin Boson Model

The subject of open-system quantum mechanics has been of strong interest and importance since the early days of quantum theory. Particularly in condensed matter and chemical physics, the description of dissipative systems and their quantum effects is a central task. It has long been known that a large collection of harmonic oscillators can simulate dissipation when coupled to a discrete set of quantum states [130]. The spin boson model [131] is such a system, a paradigmatic physical model which at its most basic consists of a two state (spin 1/2) system linearly coupled to a bosonic bath, and is the most simple model to describe the effect of an environment on constructive and destructive quantum interference, also allowing the investigation of decoherence and dampening on the quantum system [132].

In the spin boson model the two state system (with diabatic donor and acceptor states $|1\rangle$ and $|2\rangle$) and harmonic bath use the Hamiltonian

$$\hat{H} = H_B + \sigma_z H_C + \sigma_z \epsilon + \sigma_x \Delta, \tag{3.2.1}$$

where H_B and H_C are the bath and coupling Hamiltonians respectively, σ_x and σ_z are the Pauli matrices

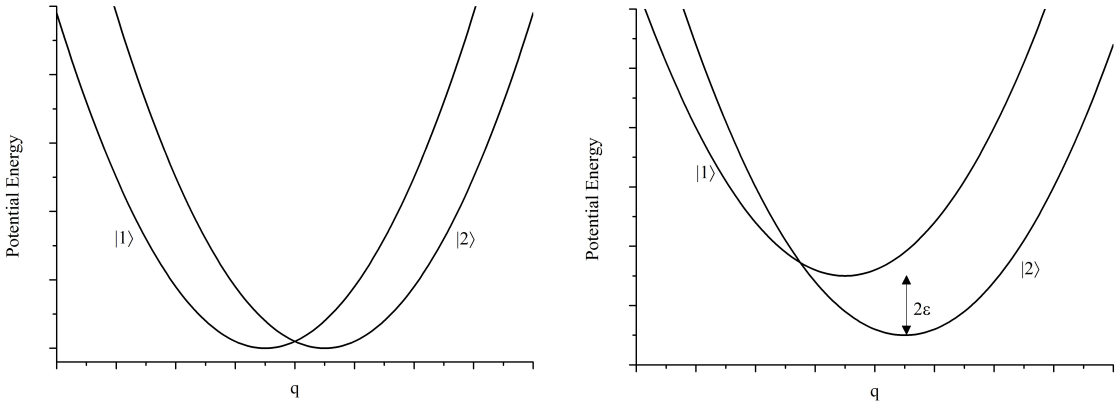
$$\begin{aligned} \sigma_z &= \left(|1\rangle\langle 1| - |2\rangle\langle 2|\right) \\ \sigma_x &= \left(|1\rangle\langle 2| + |2\rangle\langle 1|\right), \end{aligned} \tag{3.2.2}$$

and where the bias detuning parameter ϵ and the tunnelling amplitude between states Δ can both be taken to be constant. It can be reasonably assumed that the latter of these parameters is approximately independent of the vibrational degrees of freedom [133]. The partial Hamiltonians H_B and H_C can be expressed in terms of mass-weighted co-ordinates [82, 83] as

$$\begin{aligned} H_B &= \sum_m \frac{1}{2}\left(p^{(m)2} + \omega^{(m)2} q^{(m)2}\right) \\ H_C &= \sum_m C^{(m)} q^{(m)} \end{aligned} \tag{3.2.3}$$

or, as is more appropriate for our purposes, in terms of the creation and annihilation operators [23, 84] as

$$\begin{aligned} H_B &= \sum_m \omega^{(m)}\left(\hat{a}^\dagger \hat{a} + \frac{1}{2}\right) \\ H_C &= \sum_m \frac{C^{(m)}}{\sqrt{2\omega^{(m)}}}\left(\hat{a}^\dagger + \hat{a}\right). \end{aligned} \tag{3.2.4}$$



(A) Symmetric Wells (B) Asymmetric Wells

FIGURE 3.2: Illustrations of the two state system in the spin boson model for the symmetric and asymmetric cases, showing how the bias detuning parameter, ϵ, affects the relative depths of the wells. The separation of the wells has been greatly exaggerated for the purposes of clarity.

In the above equations, the strength of the coupling between the bath and the two state system is given by the parameter $C^{(m)}$. Information about the harmonic bath is encapsulated in the spectral density

$$J(\omega) = \frac{\pi}{2} \sum_m \frac{C^{(m)2}}{2\omega^{(m)2}} \delta(\omega - \omega^{(m)}), \tag{3.2.5}$$

which can be chosen to model various physical systems such as a solvent, phonons of a solid or other condensed phase environments [108]. In the general case, $J(\omega)$ is often taken to have a power-law distribution, i.e. $J \propto \omega^s$, where $0 < s < 1$ is referred to as the sub-Ohmic case and $s > 1$ is referred to as the super-Ohmic case. A widely used special case is that of the Ohmic case which has a characteristic low-frequency behaviour $J(\omega) \propto \omega$ and is peaked by at a cutoff frequency ω_c, which defines the time-scale distribution of the bath dynamics [134], such that

$$J_O(\omega) = \frac{\pi}{2} \alpha_k \omega e^{(-\omega/\omega_c)}, \tag{3.2.6}$$

where α_k is the Kondo parameter[1]. In this case, the quantum system is damped equally at all frequencies, which is the case for many physical systems [132].

The simplicity of the spin boson model makes it an ideal candidate for modelling various systems. The ability to model decoherence in a two state system makes it well suited for the investigation of decoherence which is a primary issue for the realisation of quantum

[1]This parameter appears due to the applicability of the spin boson model to the Kondo problem when using path-integral methods. The mechanics of the Kondo problem are beyond the scope of this work and will not be explored here, however the interested reader can find full discussion of the relation of the spin boson model to the Kondo problem in refs [130] and [131] where it is discussed at length.

computation [84, 135, 136]. Further applications in theoretical and condensed matter physics include investigating the light-atom interaction of an ensemble of two-state atoms interacting with a single quantized mode of the electromagnetic field [135], the coupling of a quantum dot to a boson and a femion bath [137], macroscopic quantum coherence [138] and many more. The spin boson model has also found many uses in chemical physics such as modelling of hydrogen tunnelling [139], the vibrational relaxation of a molecular impurity in a crystalline solid [140] and particularly electron transfer processes in the condensed phase [141], which cannot be described in purely classical terms as they often exhibit significant quantum effects arising from the tunnelling motion of electrons and nuclei [133]. Often application to different systems requires the careful choosing of an appropriate spectral density to describe the harmonic bath, for example to investigate outersphere electron transfer processes in polar solvents, Thoss *et al.* use the so-called Debye spectral density [142], an Ohmic spectral density with a Lorentzian cutoff, or a bimodal spectral density [133], which comprises both a Gaussian part to describe ultrafast inertial decay and a Debye part describing slower diffusion.

The physical applications of the spin boson model are not its only strength. It is argued by Leggett [131] that as the spin boson model is "probably the simplest example of a quantum-mechanical problem that has simultaneously the features of being in the extreme nonclassical limit and being extremely sensitive to dissipation, it is therefore a crucial test-bed for quantum-mechanical many-body theory and would be important as a model problem even if it had no experimental realizations". To that end it has been used to test and develop many different simulation methods, both approximate and numerically exact [82, 136]. Many of these approximate methods are based in perturbation theory, such as the non-interacting blip approximation [131] or Redfield theory [143–147], or based on mixed quantum-classical methods such as the classical Ehrenfest model [148–151], the semiclassical self-consistent field approach [134] or the surface hopping approach [30, 152–154].

The vast majority of numerically exact approaches applied to the spin boson model rely on path integral calculations [155–159] based on the Feynmann-Vernon influence functional [160, 161]. When using path integral methods, the dissipative bath can be integrated out [50, 135] which leads to an effective action reminiscent of a spin chain with long range correlations after the fashion of an Ising chain [162]. Algorithmic improvements to the path integral approach have been developed, such as decomposing the path integral into a series of shorter time operations leading to an iterative algorithm [140] or applying importance sampling to the set of possible paths [82]. A numerically exact alternative to path integral methods is using a basis set approach. The use of basis set methods for system-bath problems like the spin boson model has in the past been thought to be impossible due to the problem of exponential scaling mentioned in chapter

2 from which such methods generally suffer. The feasibility of basis set approaches for larger systems, such as the spin boson model, has been demonstrated through simulations using the MCTDH [82] and ML-MCTDH [108] methods, and also by the MCE method [23] all of which have shown that numerically exact simulations of the spin boson model using a basis set approach can be obtained with computational expense comparable to the path integral approach.

3.3 The Spin Boson Model in the Multi-configurational Ehrenfest method

As mentioned in sections 2.4.2 and 3.1, the first formulation of the MCE method was tested against the spin boson model with great success [23]. For this application the spectral density was taken to be the Ohmic case. In this case, the continuous bath spectral density given in equation (3.2.6) is discretized to the form of equation (3.2.5) by way of the relation [163]

$$C^{(m)2} = \frac{2}{\pi}\omega_j \frac{J_O(\omega^{(m)})}{\rho(\omega^{(m)})}, \tag{3.3.1}$$

where $\rho(\omega)$ is a density of frequency satisfying

$$\int_0^{\omega^{(m)}} d\omega \rho(\omega) = m, \;\; m = 1, ..., M. \tag{3.3.2}$$

It is determined in [163] and elsewhere that the precise functional form of $\rho(\omega)$ does not affect the final answer provided a large enough total number of bath modes M is used, however it can affect the total number of bath modes needed to correctly represent the continuum. In this case $\rho(\omega)$ is taken to be

$$\rho(\omega) = a\frac{J_O(\omega)}{\omega}, \tag{3.3.3}$$

where

$$a = \frac{2}{\pi}\frac{M}{\alpha_k \omega_c}\frac{1}{1 - e^{-\omega_{max}/\omega_c}}, \tag{3.3.4}$$

where ω_{max} is the largest frequency of the bath modes considered, taken to be $\omega_{max} = 5\omega_c$. Using this discretization, the equation for the coupling coefficient between the system and the bath C_m can be found as

$$C^{(m)} = \omega^{(m)}\sqrt{\frac{\alpha_k \omega_c}{M}\left(1 - e^{-\omega_{max}/\omega_c}\right)}, \tag{3.3.5}$$

and the frequencies of the bath can be found as

$$\omega^{(m)} = -\omega_c \ln\left[1 - \frac{m}{M}\left(1 - e^{-\omega_{max}/\omega_c}\right)\right]. \tag{3.3.6}$$

As already discussed, the MCE method builds a basis set around a single coherent state $|\mathbf{z}_0\rangle$. For the spin boson model the values for this initial coherent state are sampled from a thermal Boltzmann distribution by way of the 1D density operator

$$\hat{\rho}(z_0^{(m)}) = \sigma^{(m)} e^{-\sigma^{(m)}\left|z_0^{(m)}\right|}. \tag{3.3.7}$$

In practise this means the M values for the $z_0^{(m)}$ coordinates are sampled from a normally distributed random swarm centred at $(q, p) = (0, 0)$ with width $\sigma^{(m)}$, where

$$\sigma^{(m)} = e^{\beta\omega^{(m)}} - 1. \tag{3.3.8}$$

The variation of the parameters ω_c, α_k, ϵ and the thermal parameter $\beta = 1/(k_B T)$ allows a range of different systems to be modelled each requiring different numbers of degrees of freedom and basis set sizes. The basis set for MCE is often constructed as a normally distributed random swarm similar to the initial $|\mathbf{z}_0\rangle$ state around which it is centred, using the distribution

$$F(\mathbf{z}_k) \propto e^{-\alpha_c|\mathbf{z}_k - \mathbf{z}_0|^2}, \tag{3.3.9}$$

where α_c is the empirically determined compression parameter, an inverse of the width of the Gaussian swarm, which can be tuned so as to best describe the wavefunction with a finite number of basis functions. Due to the random swarm used to generate $|\mathbf{z}_0\rangle$, a number of repeat propagations N_{rpt} is needed to obtain fully converged results for all methods of sampling the initial basis set. As with all Monte Carlo based systems however, convergence with the number of repetitions is the easiest to achieve and to check (for details of this, see Appendix C.4).

As described in section 2.4, the Hamiltonian used by the MCE method must be in the form

$$\hat{H} = H(\mathbf{z}^*, \mathbf{z}). \tag{3.3.10}$$

This can be easily achieved through the relation given in equation (2.3.3) and thereby, as with equation (2.3.8), the bath and coupling Hamiltonians for the spin boson model become

$$\left\langle \mathbf{z}_j \middle| \hat{H}_B \middle| \mathbf{z}_k \right\rangle = \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \sum_m \omega^{(m)} \left(z_j^{*(m)} z_k^{(m)} + \frac{1}{2} \right) \tag{3.3.11}$$

and

$$\left\langle \mathbf{z}_j \middle| \hat{H}_C \middle| \mathbf{z}_k \right\rangle = \langle \mathbf{z}_j |\ \mathbf{z}_k \rangle \sum_m \frac{C^{(m)}}{\sqrt{2\omega^{(m)}}} \left(z_j^{*(m)} + z_k^{(m)} \right). \tag{3.3.12}$$

3.4 Comparisons of the two formulations of the Multi-configurational Ehrenfest method

The two formulations of the MCE method differ in the ansatz used to describe the wavefunction, with the MCEv1 method using

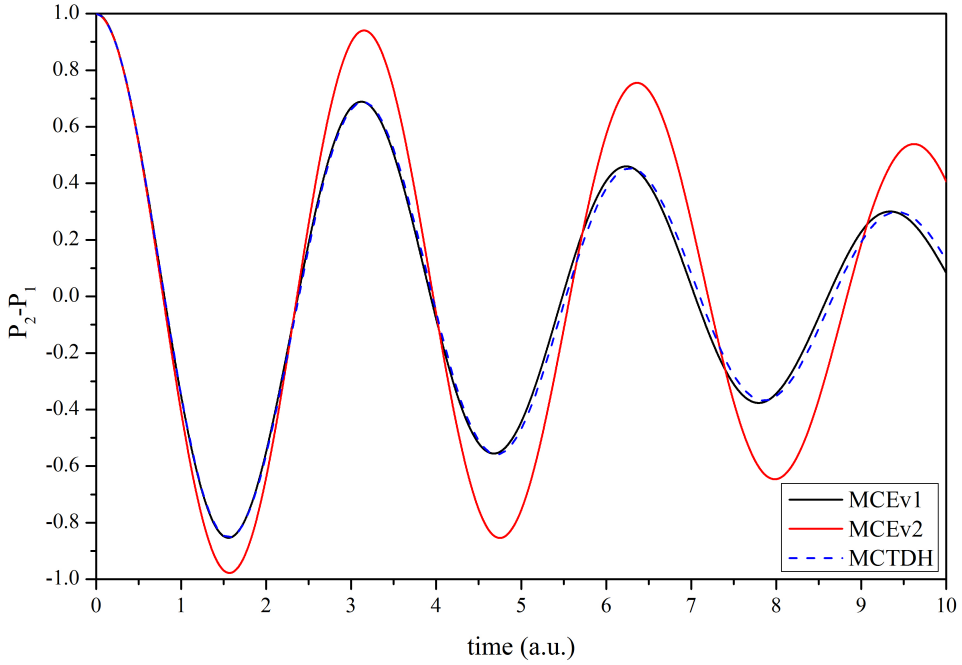
$$|\Psi(t)\rangle = \sum_k \left(\sum_r d_k^{(r)} e^{iS_k^{(r)}} |r\rangle \right) |\mathbf{z}_k\rangle \tag{3.4.1}$$

and the MCEv2 method using

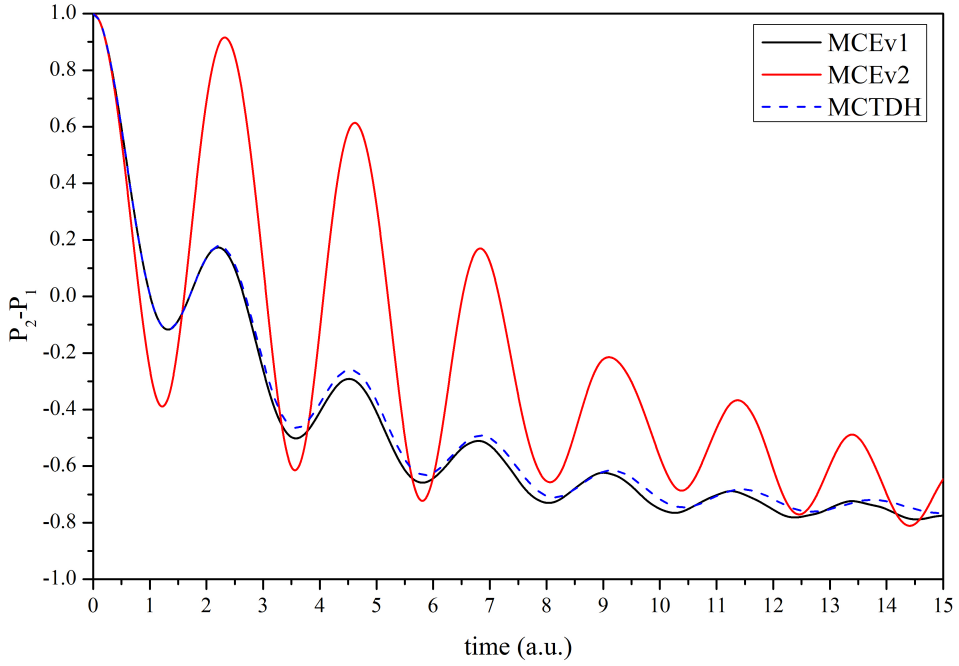
$$|\Psi(t)\rangle = \sum_k D_k \left(\sum_r d_k^{(r)} e^{iS_k^{(r)}} |r\rangle \right) |\mathbf{z}_k\rangle , \tag{3.4.2}$$

where as elsewhere in this work, the index r refers to electronic states or potential energy surfaces, and the index k refers to the set of basis functions. As a result of the differences in the ansatz, the methods of coupling between the basis functions and between the electronic states are different, with the MCEv1 method coupling both the electronic states and the basis functions through the time propagation equations for the $d_k^{(r)}$ amplitudes, and the MCEv2 method coupling only the electronic states through the time propagation equations for the $d_k^{(r)}$ amplitudes while the basis functions are coupled through the equations for the D_k amplitudes. While the MCEv2 method has been successful at simulating small organic molecules [24, 26, 27, 32], there has never been a test on a model system with high dimensionality comparing the two methods on an even footing.

To this end, a program was written which was capable of comparing these two methods on an even footing through use of a modular design which treated the wavefunction in exactly the same way for both methods, differing only in the subroutines called for the calculation of the time evolution of the wavefunction parameters. Through this modular design, the program, written in Fortran 95, was also capable of propagating the wavefunction using different Hamiltonians and incorporating different modifications into the basis set sampling and propagation processes (details of the notable algorithms and programming details can be found in Appendix A, and information on the use of the program can be found in Appendix C).

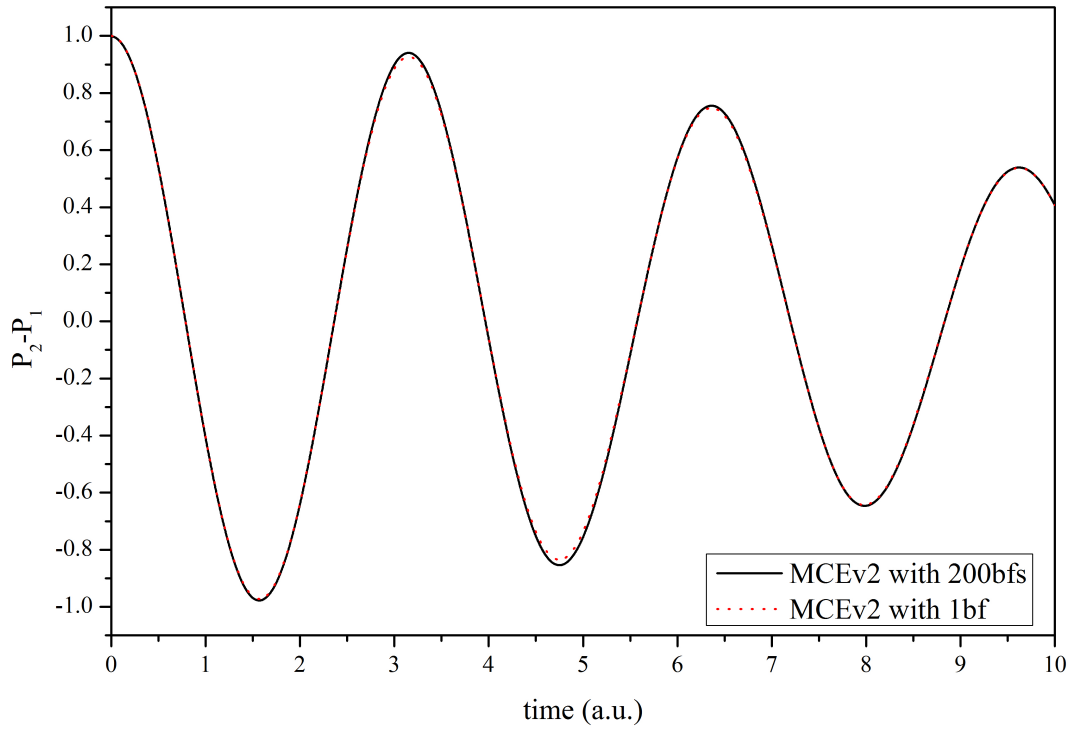


(A) Comparison of the MCEv1 and MCEv2 population difference for the spin boson model with symmetric wells, using the parameters $\omega_c = 2.5$, $\alpha_k = 0.09$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 0$ with $M = 50$ degrees of freedom and $N = 50$ basis functions and $N_{rpt} = 256$ repetitions. Results are also compared to the numerically exact MCTDH result from ref [83]

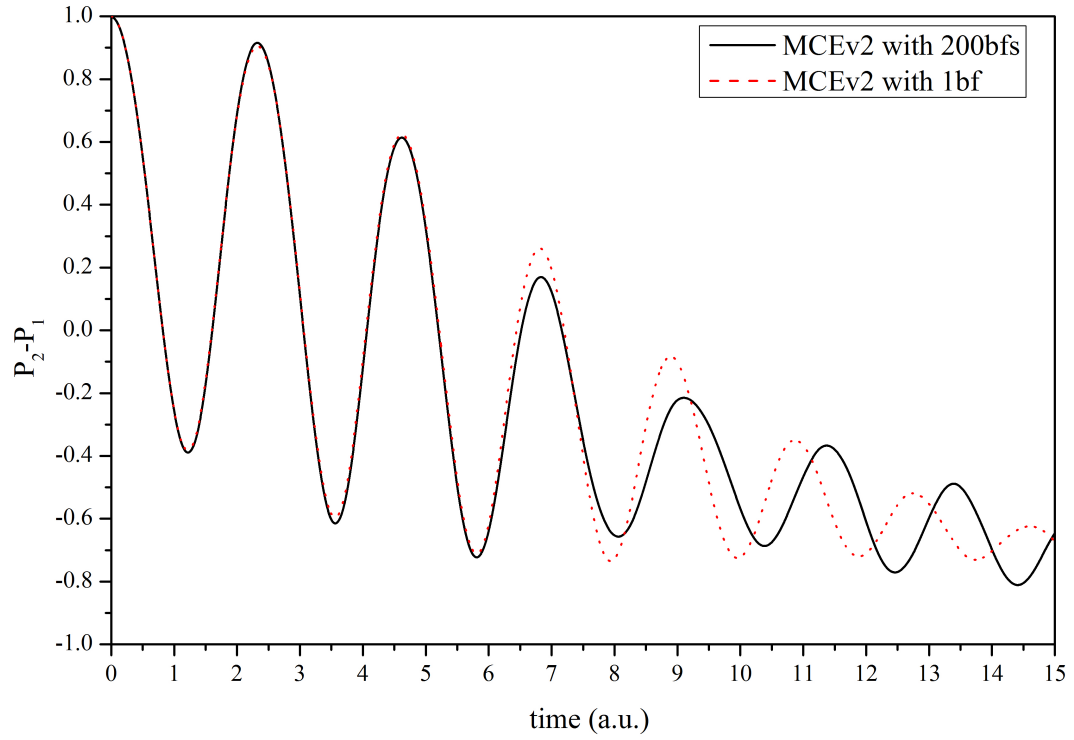


(B) Comparison of the MCEv1 and MCEv2 population difference for the spin boson model with asymmetric wells, using the parameters $\omega_c = 7.5$, $\alpha_k = 0.10$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom and $N = 200$ basis functions and $N_{rpt} = 256$ repetitions. Results are also compared to the numerically exact MCTDH result from ref [82]

FIGURE 3.3: Comparisons of the two formulations of the MCE method for symmetric and asymmetric wells



(A) Comparison of the single basis function and MCEv2 population difference for the spin boson model with symmetric wells



(B) Comparison of the single basis function and MCEv2 population difference for the spin boson model with asymmetric wells

FIGURE 3.4: Comparisons of the MCEv2 formulation with single basis function simulations

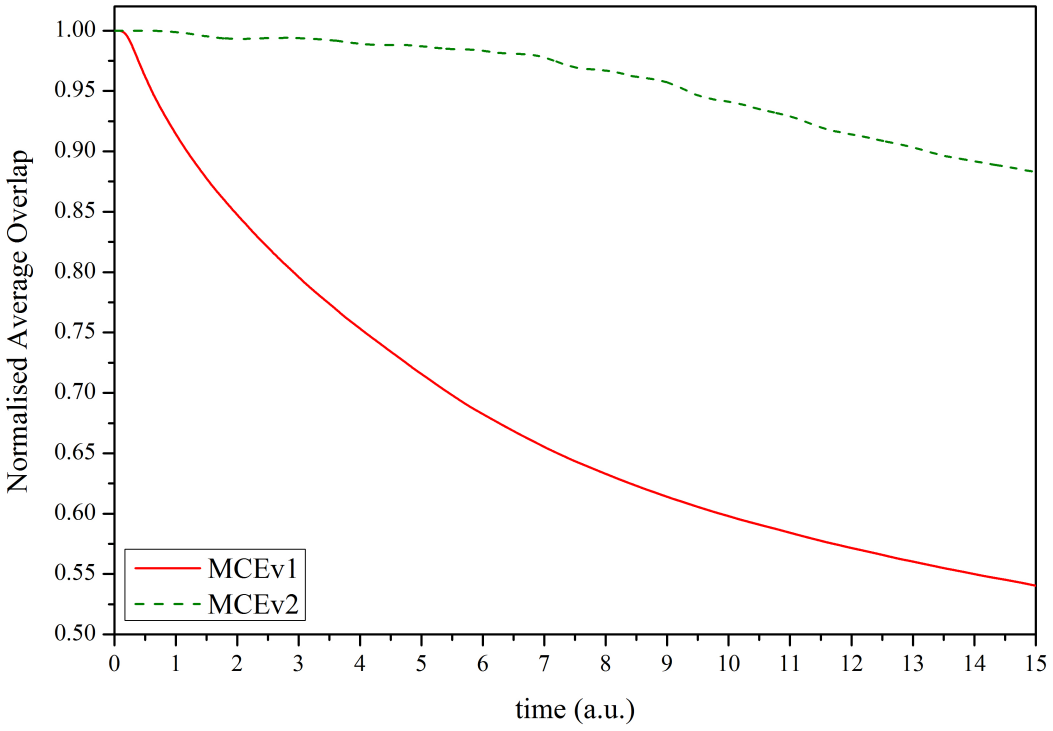


FIGURE 3.5: Comparison of the normalised average overlap between the coherent states for both formulations of the MCE equations for the asymmetric case of the spin boson model, using $N_{bf} = 200$ basis functions and the parameters $\omega_c = 7.5$, $\alpha_k = 0.10$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom, averaged over $N_{rpt} = 100$ repetitions.

A comparison was made using the spin boson model looking initially at a pair of symmetric wells (figure 3.3a) and at a pair of asymmetric wells (figure 3.3b). It can be immediately seen that there is a great disparity between the two results, with the oscillations for the MCEv2 result overemphasised in both cases. When discovered, this result was very surprising, as numerically the wavefunctions should be identical and it was thought that the strength of the coupling between the basis functions should not have been affected by the MCEv2 equations. It is a cause of further surprise that when contrasted against the result obtained when using a single basis function for the calculations, rather than an ensemble of coupled basis functions, it can be seen that for the symmetric case the MCEv2 population difference is identical (fig 3.4a) and for the asymmetric case the MCEv2 population difference matches for the first few oscillations (fig 3.4b). This would seem to indicate either that the basis set has fallen into the semi-classical regime, behaving as either an ensemble of independent non-interacting basis functions due to a loss of coupling or that the basis is behaving as a set of basis functions guided by trajectories which are too similar to each other, thus behaving almost as a single larger basis function. A loss of coupling between the basis functions would, in most cases, indicate a loss of overlap between the coherent states, however if the guiding

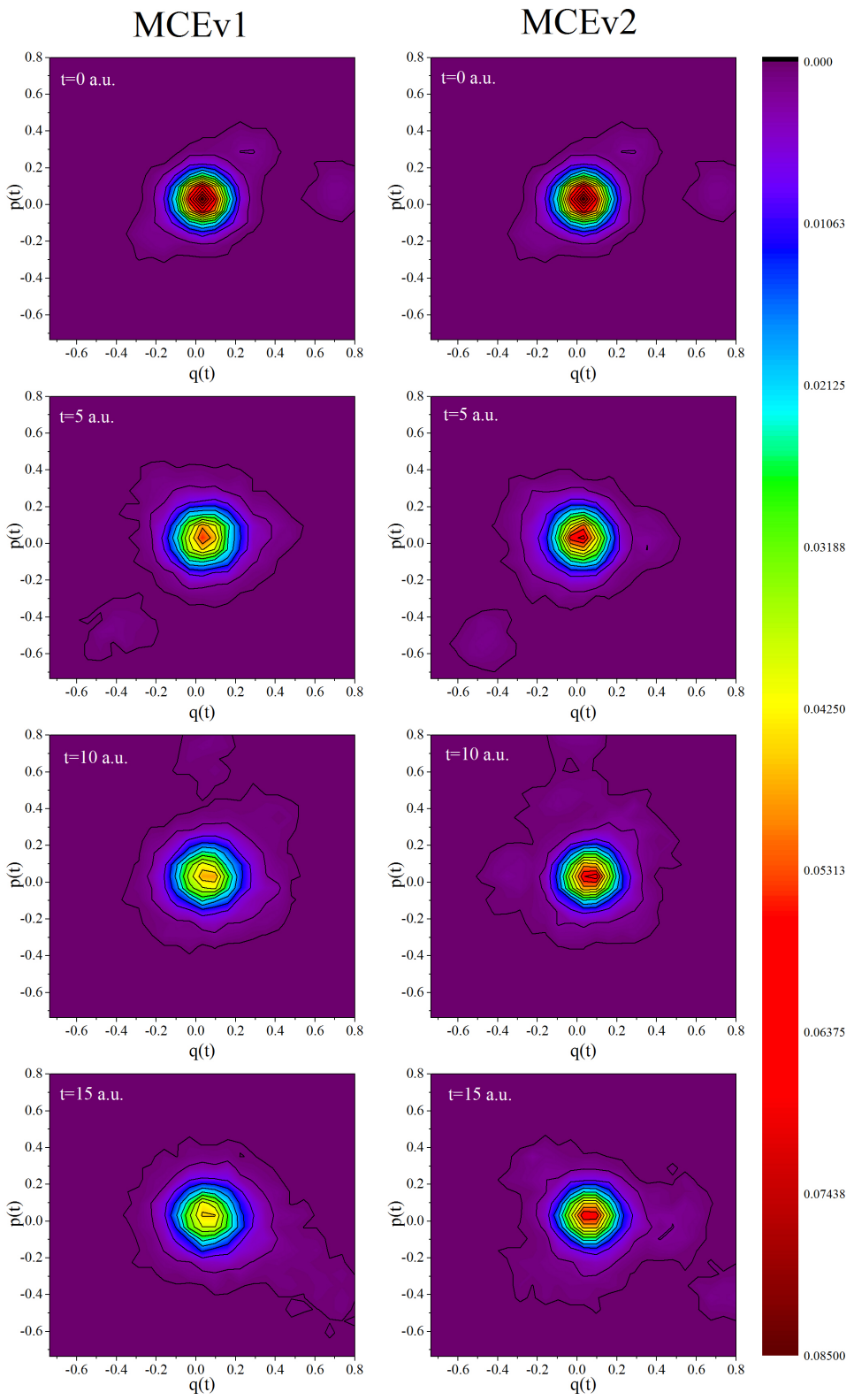


FIGURE 3.6: Comparison of the density plots for identical swarms of coherent states acted on by the MCEv1 and MCEv2 equations for the asymmetric case of the spin boson model, using $N_{bf} = 200$ basis functions and the parameters $\omega_c = 7.5$, $\alpha_k = 0.10$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom, shown at $t = 5$ *a.u.* intervals. To create plots, data for a set of 2D histograms was divided by the total number of points present and each pair of (q, p) coordinates for all basis functions and degrees of freedom is considered for a histogram over 10,000 points.

trajectories are too similar this would most likely result in a higher overlap. As such, a comparison of the normalised absolute average overlap, i.e. an average over the absolute values of all elements of the overlap matrix $\mathbf{\Omega}_{jk}$, was prepared. As is seen in figure 3.5 the overlap for the MCEv2 simulation is higher than that of the MCEv1 simulation, decaying much slower and not as smoothly. This means that the basis set does not spread to cover as much of an area of phase space when using the MCEv2 equations. This is confirmed in figure 3.6, which shows the evolution of the coherent states in the wavefunction at four different time intervals for both formulations of the MCE method, starting with the same initial basis set swarm. It can be seen from these plots that while both methods have the same starting point, the wavefunction spreads out more for the MCEv1 simulation, as is indicated by the density of contour lines and the higher peak at the centre of the wavefunction for the MCEv2 simulation plots. This effect can be understood by considering that the time propagation equations for the coherent states uses the Ehrenfest Hamiltonian H^{Ehr}, which has a dependency on the amplitudes $d_k^{(r)}$. For the MCEv1 equations, the interconfigurational coupling is contained within these amplitudes and so the coherent states will effectively "push" on each other, spreading the basis functions out to cover a larger area in phase space. As this is not the case for the MCEv2 equations, the coherent states become less spread out and so the basis set cannot adequately describe a sufficient area of phase space to fully account for the quantum mechanics of the system. Obviously this behaviour is not ideal, and so steps should be taken to allow the basis functions to spread more and cover a larger area of phase space.

3.5 Basis set Refinements and Improvements in the MCEv2 method

There have been various methods used for improving the MCE method, one of which involves using coherent state "trains" for initial sampling [67] which applies a "smoothing" to the propagation of the wavefunction. Another method involves basis set cloning [1] which grows the basis set when intersections are encountered. In the following sections, both of these options are explored and a discussion of the ways in which the MCEv2 method can be improved through their inclusion will be presented.

3.5.1 Use of Basis Function Trains to Improve MCEv2

Coherent state trains were first proposed in the context of coupled coherent states in 2008 [67] as a way of inserting some "regularity" into a random swarm. The argument is

that a random swarm, while improving scalability, necessitates a sacrifice in convergence. At the other end of the scale, a regular grid allows extremely fast convergence but scales exponentially, resulting in high numerical expense for all but the smallest of systems. The ideal compromise lies somewhere between these two extremes, with not total regularity but not a true random swarm either.

In a coherent state train, the basis functions form a line in phase space along the path of propagation for a single basis function. This allows the basis set to cover a larger area in phase space than is covered by a compressed random swarm. Due to the structure of the trains, the process of constructing the initial basis set is somewhat different to that described earlier. As with the construction of a random swarm the initial wave packet is calculated first, which for the spin boson model is given by equation (3.3.9), however a random swarm is not constructed around this initial wave packet. Instead the single configuration amplitudes are set be $(1 + 0i)$ for the initial electronic state in $|\mathbf{z}_0\rangle$, and a parameter Δt_{trn} is set which is defined to be the timespace between two adjacent basis function "carriages". The wave packet is then propagated backwards in time by a number of time steps equal to half the desired length of the train multiplied by the number of time steps required to make up the "train spacing" parameter, Δt_{trn} before being propagated forward again by twice this number of time steps. After every $t = \Delta t_{trn}$ the configuration is saved such that a set of coherent states with single configuration amplitudes are obtained, all of which follow the same path in phase space, hence the term "trains". The cross configuration amplitude D_k is then calculated over all the single configuration basis function "carriages" and so the wavefunction is spread out over the length of the train.

The train spacing parameter determines the degree to which the initial basis functions overlap. Too large a spacing would result in a loss of coupling between the basis functions, as this coupling is dependent upon the overlap matrix. As such, care must be taken in choosing the correct spacing between the basis functions and it should be remembered that the optimum spacing parameter will be dependent upon the system being simulated. In some systems the basis functions may move faster through phase space than in others, resulting in a more rapid reduction of the coupling. It should also be noted that the number of basis functions in the ensemble is of great importance, as the combination of a small Δt_{trn} value and a small number of basis functions will result in a train which does not have sufficient size in phase space to properly describe the wavefunction.

3.5.2 Use of Basis Function Cloning to Improve MCEv2

Multiple cloning is a recent inclusion into the MCE method, having been introduced in a coupled coherent states context in two papers published by Makhov *et al.* [1, 2]. It takes inspiration from the Multiple Spawning method of Martínez and Ben-Nun [11–15, 122, 126] in which a Gaussian wave packet, guided by classical mechanics, is split upon reaching an area of strong nonadiabatic coupling. This allows the wavefunction to be simultaneously projected onto two potential energy surfaces, thus describing population transfer properly while still using classical mechanics to propagate the wave packet. This is very different from the standard MCE method, in which a wavefunction, projected upon a set of Gaussian wave packets, propagates not along the potential energy surface classically but on a quantum average of the potential using the time propagation equations given earlier which are derived from the variational principle and the time dependent Schrödinger equation.

The Multiple Cloning MCE method uses the same quantum average over the potential, however it is a problem inherent to all methods based on Ehrenfest dynamics that in a region of nonadiabatic coupling where the population of an Ehrenfest trajectory is split almost equally across multiple potential energy surfaces with different forces, this average is not a faithful representation of the system, propagating the wavefunction subject to a force which is an average of different forces on each electronic state. To remedy this, that basis function is cloned, with one instance projected onto the first potential energy surface, and the other projected onto the second potential energy surface (assuming a two state system). If, before a cloning event, a single basis function for a two state

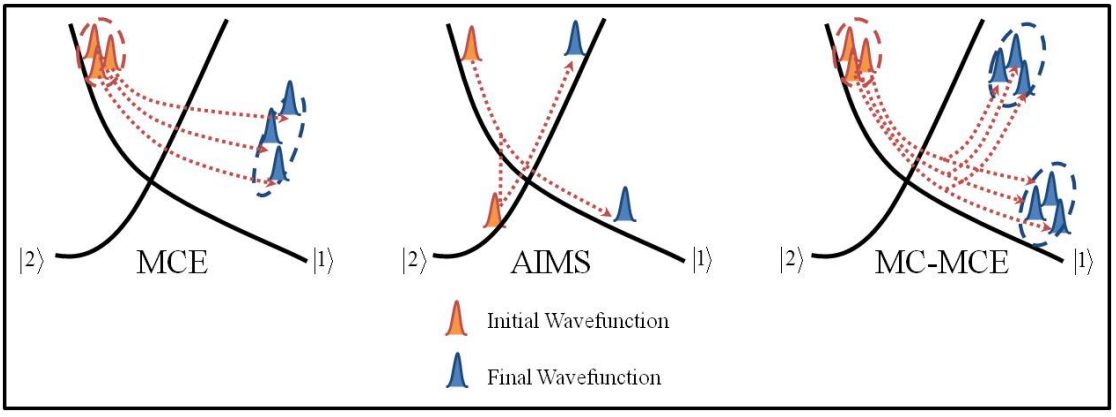


FIGURE 3.7: Illustration of the differences between the way in which the MCE, AIMS and Multiple Cloning MCE systems behave in the region of an intersection. Adapted from a version which appears in [27].

system is given by

$$|\psi_k(t)\rangle = D_k \left(d_k^{(1)} e^{iS_k^{(1)}} |1\rangle + d_k^{(2)} e^{iS_k^{(2)}} |2\rangle \right) |\mathbf{z}_k\rangle \,, \tag{3.5.1}$$

then after cloning two basis functions will exist, given by

$$\begin{aligned}
\left|\psi_k'(t)\right\rangle &= \left(D_k \left|d_k^{(1)}\right|\right)\left(\frac{d_k^{(1)}}{\left|d_k^{(1)}\right|}e^{iS_k^{(1)}}\left|1\right\rangle + 0e^{iS_k^{(2)}}\left|2\right\rangle\right)\left|\mathbf{z}_k\right\rangle, \\
\left|\psi_k''(t)\right\rangle &= \left(D_k\sqrt{1-\left|d_k^{(1)}\right|}\right)\left(0e^{iS_k^{(1)}}\left|1\right\rangle + \frac{d_k^{(2)}}{\sqrt{1-\left|d_k^{(1)}\right|}}e^{iS_k^{(2)}}\left|2\right\rangle\right)\left|\mathbf{z}_k\right\rangle.
\end{aligned} \tag{3.5.2}$$

The determination of when to clone a basis set is dependent upon the force between the potential energy surfaces, as given by

$$\mathbf{F}_{1,k}^{br} = -\mathbf{F}_{2,k}^{br} = \left|a_k^{(1)}a_k^{(2)}\right|^2 \nabla(V_1 - V_2). \tag{3.5.3}$$

In the spin boson model the differential of the potential is a constant, and as such the maximum of the breaking force can be determined to be where the single configurational amplitudes are equal for both potential energy surfaces. As such, an appropriate condition for cloning would be when $\left|a_k^{(1)}a_k^{(2)}\right|^2 < 0.249$. A further necessary condition would be the limiting of cloning events on the same configuration within an appropriate number of timesteps, allowing the basis function to move away from the intersection of the two potential energy surfaces, thus preventing multiple cloning events being applied to a single basis function due to the same intersection.

As the basis set increases in size this means that the wavefunction can be better described in phase space. Furthermore, as the cloned basis functions propagate, they will spread out further than the original basis function as a result of the difference in the electronic state amplitudes. This should counteract the effect of the higher overlap seen in figure 3.5 for simulations using the MCEv2 equations. This is in addition to the fact that due to the cloning process the wavefunction no longer becomes ill-defined in the region immediately after passing through an intersection. This makes it a useful addition to the MCE method and potentially very useful in solving the problems encountered with the spin boson model.

3.6 Results

The following sections detail the process of determining the best combination of methods and parameters to accurately describe the spin boson model using the MCEv2 method. As for all variations on the method there is an aspect of Monte Carlo sampling. To ensure convergence unless otherwise stated all results are an average over 256 repeat simulations. This is deemed sufficient as it is shown in ref [23] that for most cases 100

repeats is sufficient for convergence. For details on how convergence is assured through comparison of the cumulative average results at the time of data collection refer to Appendix A.

3.6.1 Tests for the Effectiveness of Basis Function Trains

As mentioned in section 3.5.1, when projecting the wavefunction onto a train-type basis set, the quality of said basis set is dependent on the basis set size N_{bf} (that is, the "length" of the train) and on the space between adjacent basis functions, which is given by the parameter Δt_{trn}. In order to assess the improvement to the MCEv2

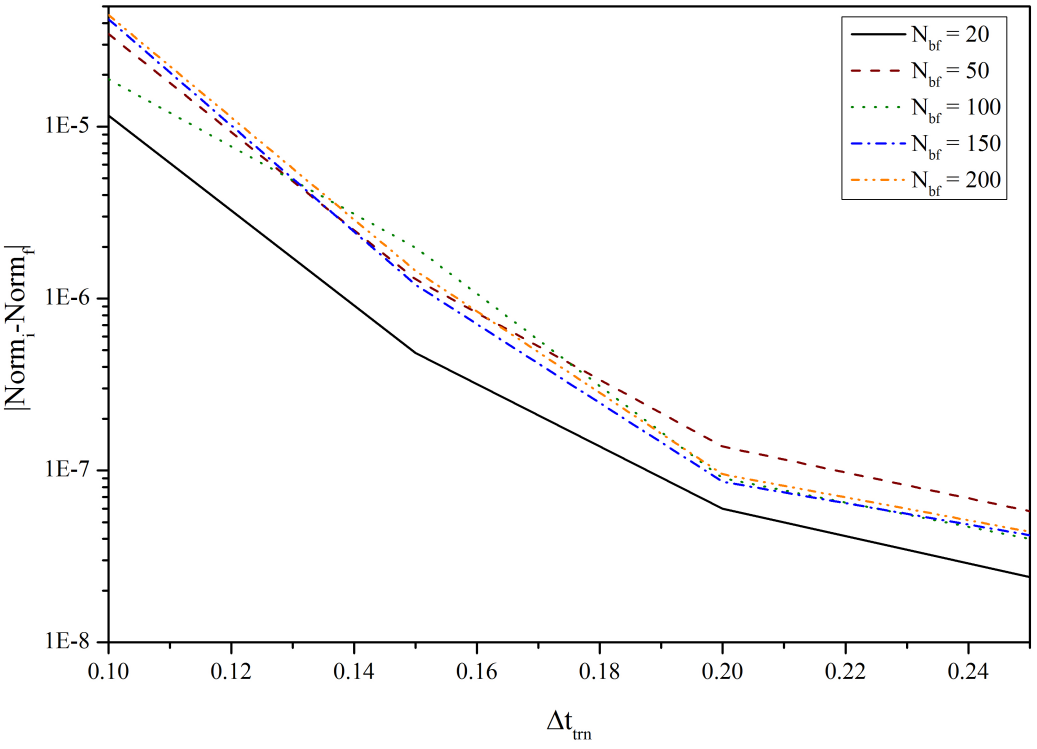


FIGURE 3.8: Plot showing the way in which the range of the fluctuation in the norm changes for different values of Δt_{trn} and train length (N_{bf}) for the symmetric case of the spin boson model using a single train as the structure of the basis set, taken as an average over $N_{rpt} = 256$ repetitions.

method provided by the use of a train-type basis set the correct combination of these parameters must be found. To this end, various combinations of these parameters were tested against the symmetric and asymmetric cases of the spin boson model, using the same system parameters as in section 3.4. As a first test, the behaviour of the norm of the wavefunction was investigated over the duration of propagation.

In simulations using the basis set approach, it is often the case that an improperly constructed basis set will not conserve the norm of the wavefunction properly. Even

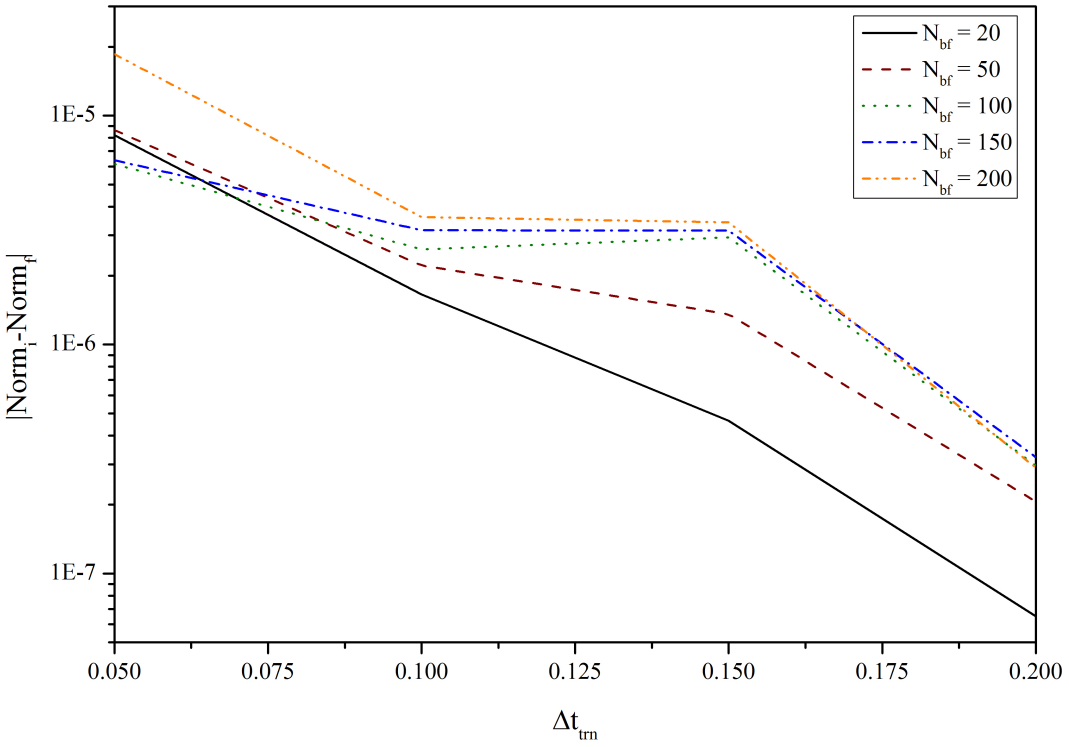


FIGURE 3.9: Plot showing the way in which the range of the fluctuation in the norm changes for different values of Δt_{trn} and train length (N_{bf}) for the asymmetric case of the spin boson model using a single train as the structure of the basis set, taken as an average over $N_{rpt} = 256$ repetitions.

in the best cases, the norm is not completely conserved as small numerical errors often creep in during propagation. As such the deviation of the norm over the length of propagation is a good measure of the quality of the simulation. Figures 3.8 and 3.9 give plots showing this deviation, indicating the difference between the initial and final values of the norm on a logarithmic scale, taken from an average over 256 repeat simulations. From previous work it has been found that a deviation on the order of 1×10^{-6} is adequate for faithfully reproducing results generated using the MCTDH approach in MCEv1 simulations of the spin boson model [23].

It can be seen in figure 3.8 that to maintain the norm within these limits a basis set train with a separation of $\Delta t_{trn} \geq 0.17$ *a.u.* is sufficient for all train lengths. It should be noted that for all train lengths, a spacing of $\Delta t_{trn} = 0.1$ *a.u.* results in an increase in the norm, which is often an indicator of instability in the propagation of the wavefunction, and so spacings of this size should be ruled out for all train lengths. Also, it was found that when $\Delta t_{trn} \geq 0.3$ *a.u.*, the initial norm of the wavefunction is too low being in the region of $\langle \Psi |\ \Psi \rangle = 0.97$ regardless of the length of the train.

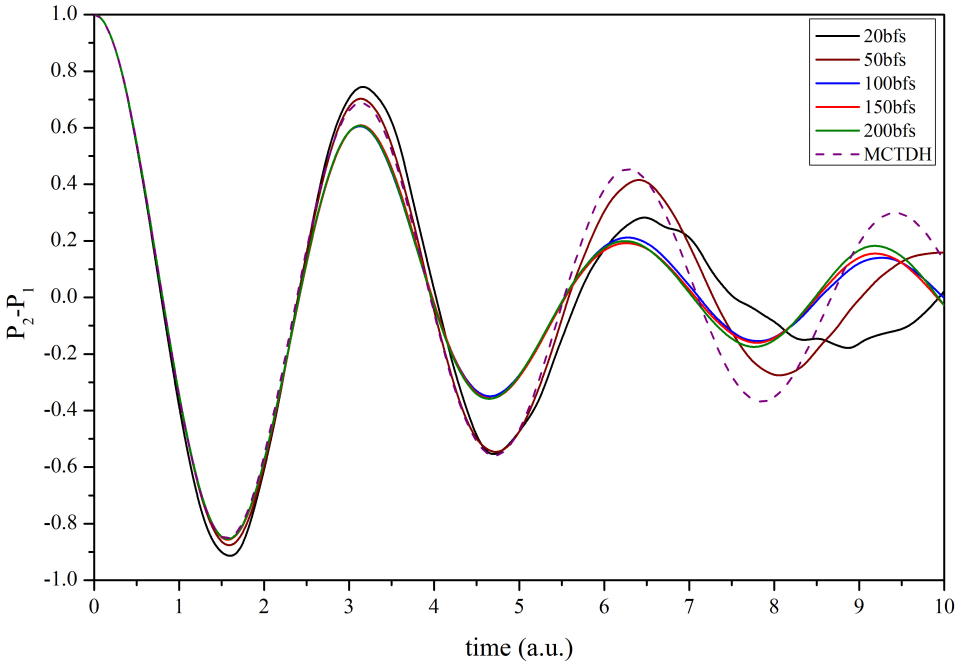
For the asymmetric case (figure 3.9) it appears that the longer trains exhibit similar

behaviour past a spacing of $\Delta t_{trn} = 0.1$ *a.u.*, and for all train lengths a spacing of $\Delta t_{trn} \geq 0.175$ *a.u.* is sufficient to ensure that the norm does not decay by more than the limit of 1×10^{-6}. The plateau between $\Delta t_{trn} = 0.1$ *a.u.* and $\Delta t_{trn} = 0.15$ *a.u.* in the region of 4×10^{-6} for trains of a length of $N_{bf} \geq 100$ allows for some leeway in this figure. The fact that the larger basis sets behave in similar ways with regards to the norm is expected considering that to properly describe this case the MCEv1 method requires a swarm of 200 basis functions. It should be noted however that for the asymmetric case when $\Delta t_{trn} > 0.20$ *a.u.*, the initial norm of the wavefunction is again too low being in the region of $\langle \Psi | \, \Psi \rangle = 0.97$. As such, trains with these values for the separation between the basis functions can be deemed as unsuitable for properly describing the wavefunction.

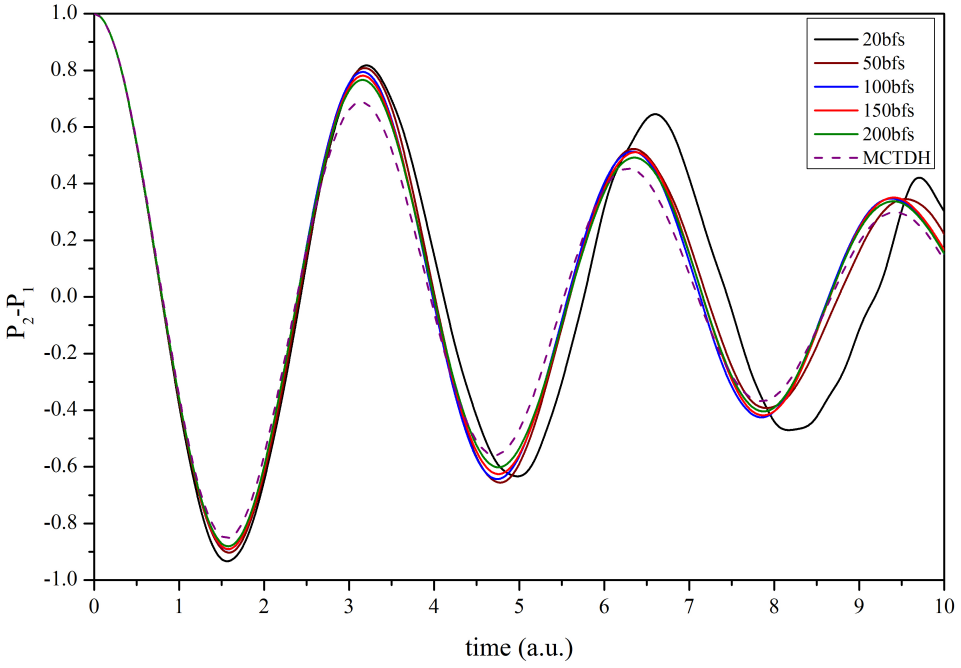
From these preliminary tests, an indication of acceptable ranges of the train spacing can be determined, however while the behaviour of the norm can be a good indication of the stability of the simulation, it cannot predict the quality of the final result. To that end, a comparison of the agreement of the population difference with that from MCTDH benchmark simulations is required. As before, the MCTDH results for the symmetric case are from ref [83] and those for the asymmetric case are from ref [82]. Such a comparison is given in figures 3.10 and 3.11 for the symmetric case and in figures 3.12 and 3.13 for the asymmetric case.

When considering the results for the symmetric case, it is apparent that for the smaller Δt_{trn} values the smallest basis set ($N_{bf} = 20$) gives unstable results. This is as expected when the behaviour of the norm is considered. It should also be noted that for $\Delta t_{trn} = 0.10$, the different values of N_{bf} give very different results, with the larger basis sets experiencing greater dampening of the oscillations than is seen in the MCTDH benchmark while the smaller basis sets show less dampening. For $\Delta t_{trn} = 0.15$ much better and stable results are seen with the $N_{bf} \geq 100$ results agreeing both with each other and to a good extent with the MCTDH benchmark. Unfortunately, for these results there is too large a deviation seen in the norm and as such these are unsuitable. Better results are seen for $\Delta t_{trn} = 0.20$ and $\Delta t_{trn} = 0.25$, and indeed for $\Delta t_{trn} = 0.25$ the results for all but the smallest basis set size are almost completely converged with no difference visible for any of the population differences where $N_{bf} \geq 100$, and only a very slight difference visible for $N_{bf} = 50$. From this and considering the deviations in the norm the best separation between the basis functions is $\Delta t_{trn} = 0.25$ for a basis set of $N_{bf} = 50$ or more.

For the asymmetric case when the separation between adjacent basis functions is too small the instability is more pronounced compared to the symmetric case, noticeable to some degree for all sizes of the basis set. Increasing the separation to $\Delta t_{trn} = 0.10$ improves the result slightly for $N_{bf} \geq 150$ however smaller basis sets still show

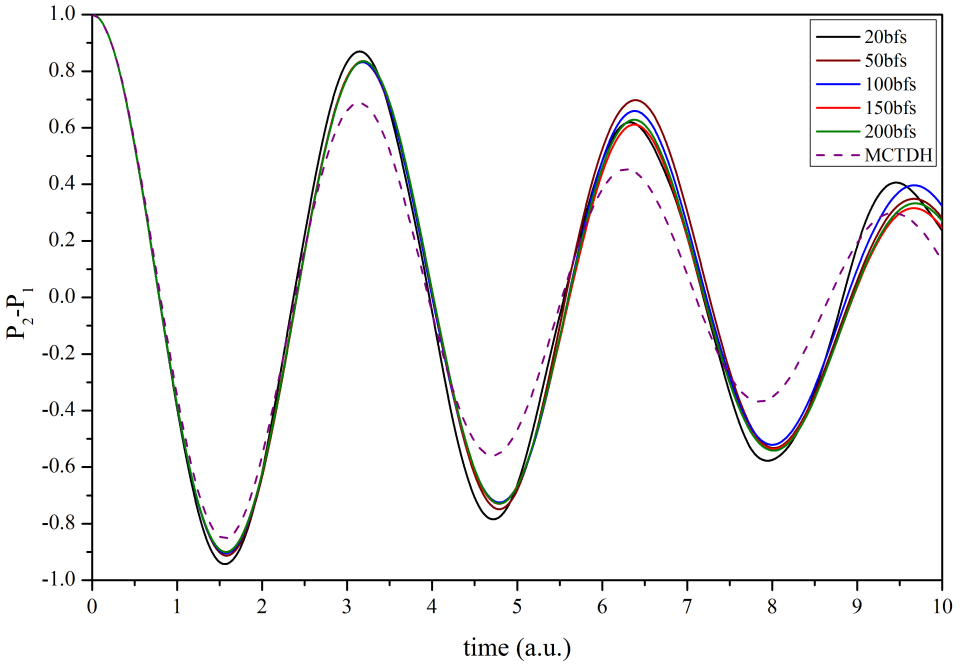


(A) $\Delta t_{trn} = 0.10$

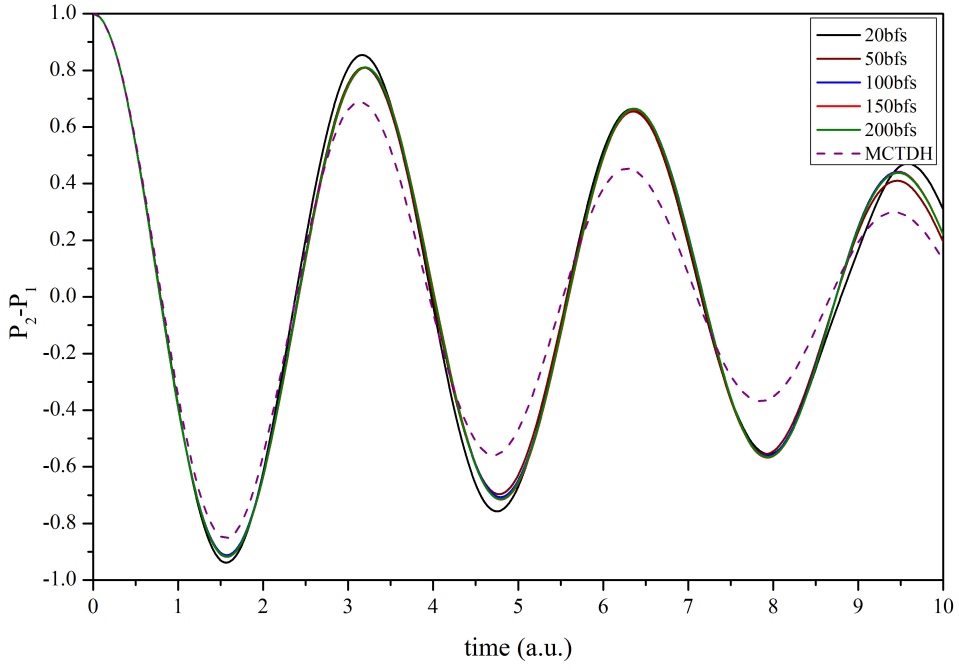


(B) $\Delta t_{trn} = 0.15$

FIGURE 3.10: Comparisons of population differences from MCEv2 simulations of the symmetric case of the spin boson model with a single train as the basis set, using different lengths of basis train for particular values of the train spacing parameter Δt_{trn}. As before the spin boson parameters for this case are $\omega_c = 2.5$, $\alpha_k = 0.09$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 0$ with $M = 50$ degrees of freedom and $N = 50$ basis functions and $N_{rpt} = 256$ repetitions. The MCTDH population difference is included for comparison [83]. Further values of Δt_{trn} are compared in 3.11.

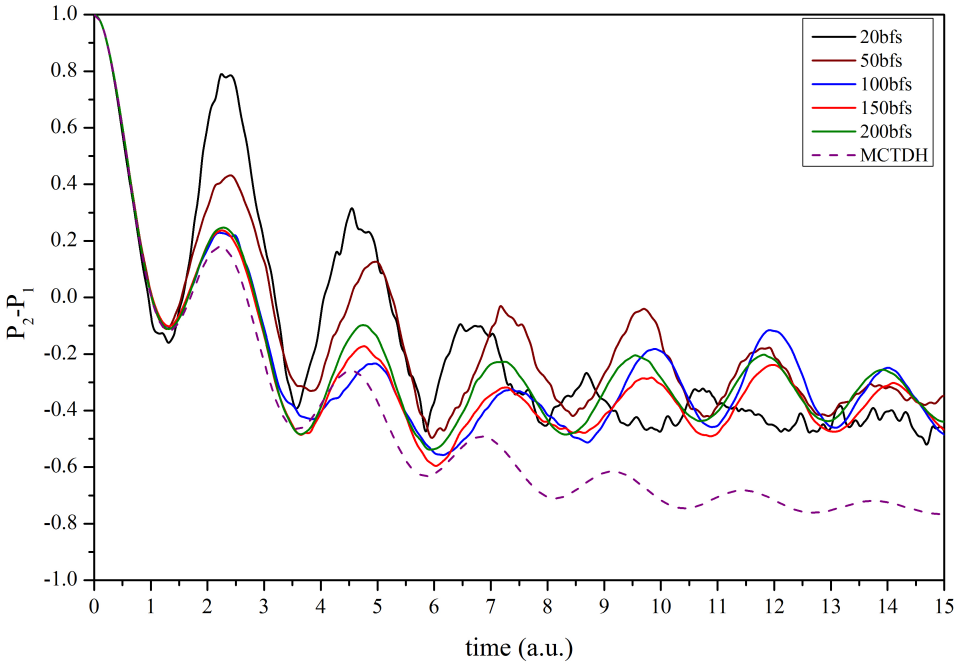


(A) $\Delta t_{trn} = 0.20$

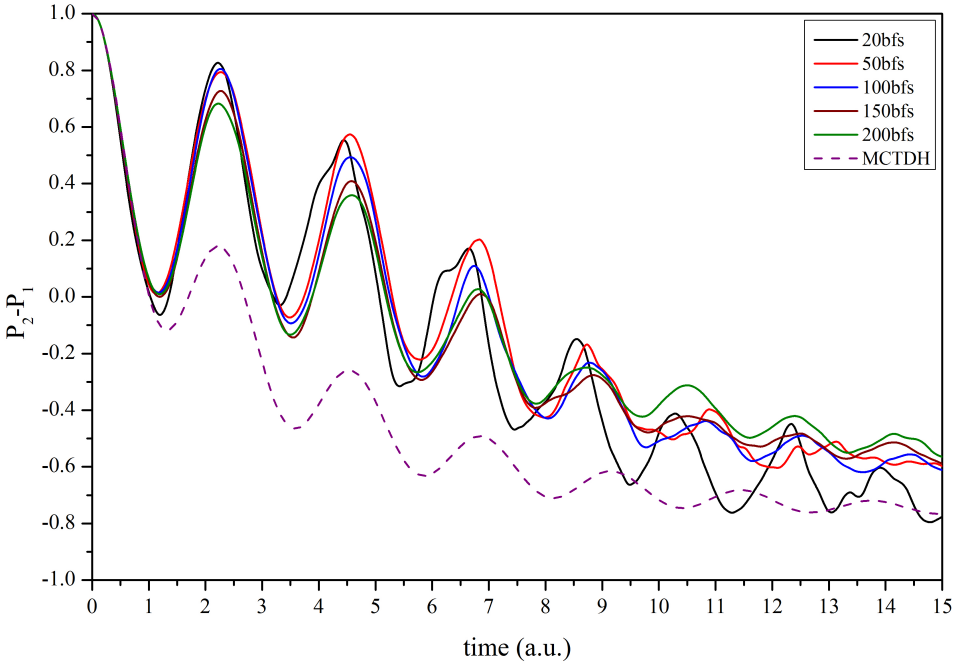


(B) $\Delta t_{trn} = 0.25$

FIGURE 3.11: Comparisons of population differences from MCEv2 simulations of the symmetric case of the spin boson model with a single train as the basis set, using different lengths of basis train for particular values of the train spacing parameter Δt_{trn}. As before the spin boson parameters for this case are $\omega_c = 2.5$, $\alpha_k = 0.09$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 0$ with $M = 50$ degrees of freedom and $N_{rpt} = 256$ repetitions. The MCTDH population difference is included for comparison [83]. Further values of Δt_{trn} are compared in 3.10.

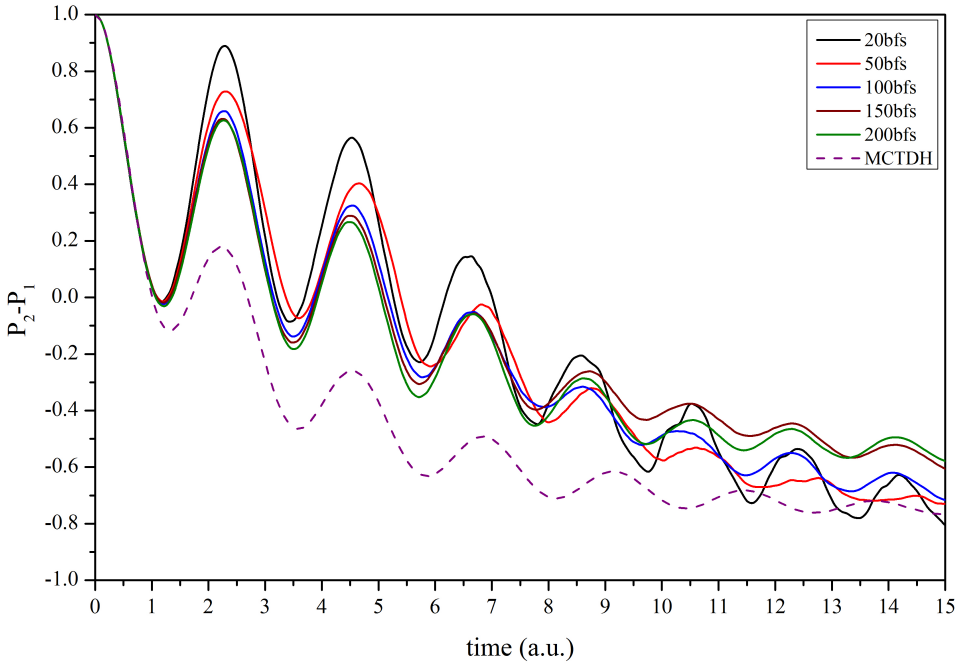


(A) $\Delta t_{trn} = 0.05$

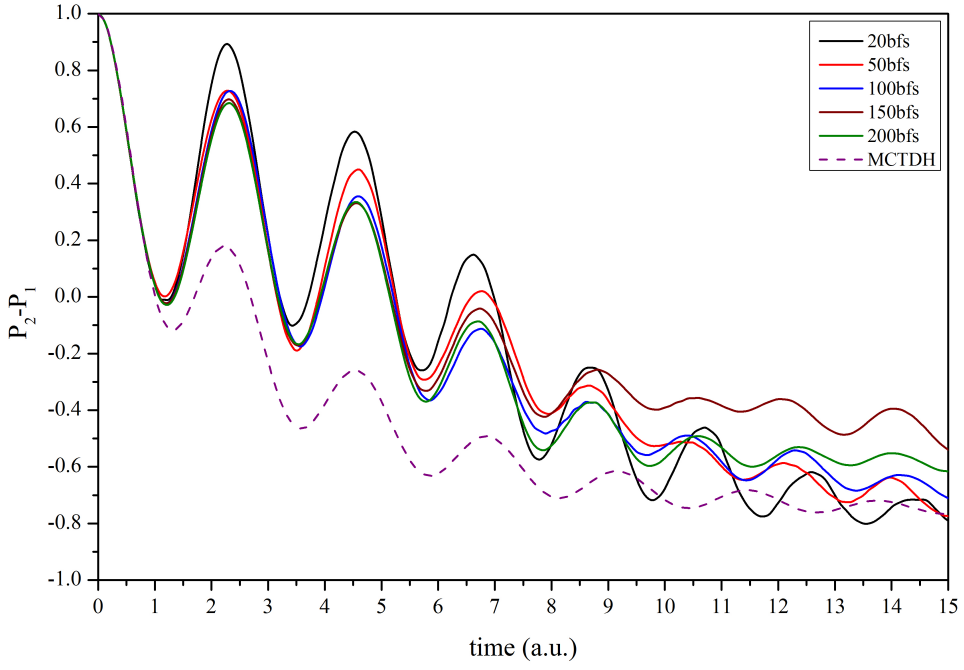


(B) $\Delta t_{trn} = 0.10$

FIGURE 3.12: Comparisons of population differences from MCEv2 simulations of the asymmetric case of the spin boson model with a single train as the basis set using different lengths of basis train for particular values of the train spacing parameter Δt_{trn}. As before the spin boson parameters for this case are $\omega_c = 7.5$, $\alpha_k = 0.1$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom and $N_{rpt} = 256$ repetitions. The MCTDH population difference is included for comparison [82]. Further values of Δt_{trn} are compared in 3.13.

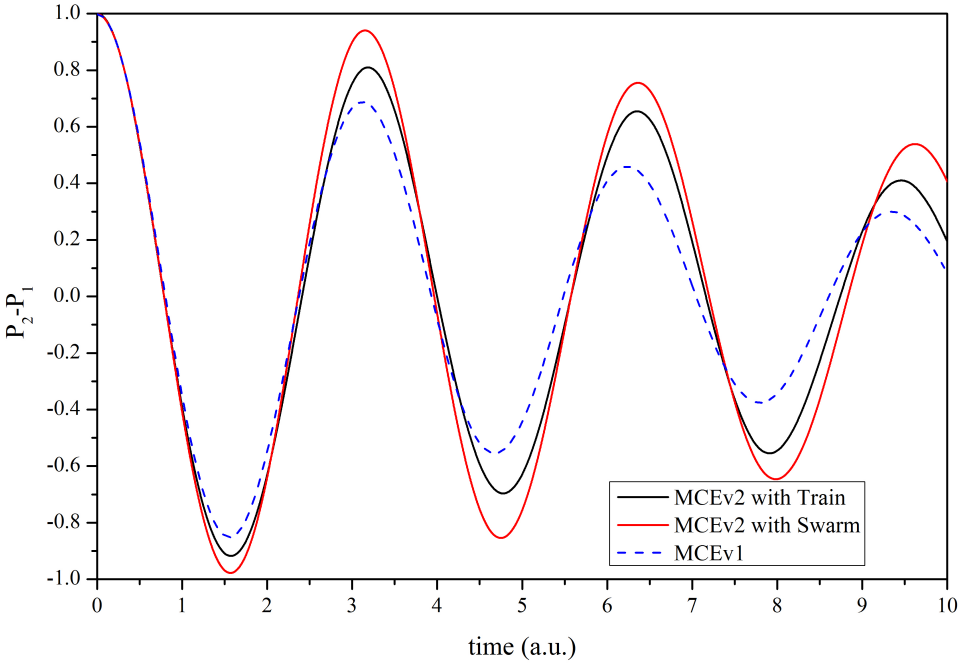


(A) $\Delta t_{trn} = 0.15$

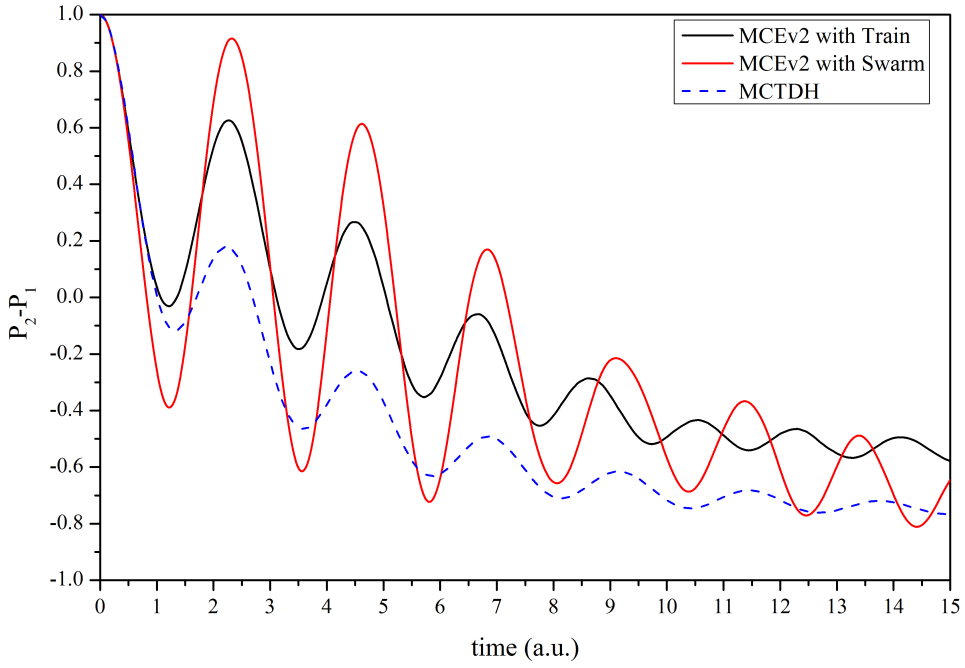


(B) $\Delta t_{trn} = 0.20$

FIGURE 3.13: Comparisons of population differences from MCEv2 simulations of the asymmetric case of the spin boson model with a single train as the basis set, using different lengths of basis train for particular values of the train spacing parameter Δt_{trn}. As before the spin boson parameters for this case are $\omega_c = 7.5$, $\alpha_k = 0.1$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom and $N_{rpt} = 256$ repetitions. The MCTDH population difference is included for comparison [82]. Further values of Δt_{trn} are compared in 3.12.



(A) Symmetric case, using $N_{bf} = 50$ $\Delta t_{trn} = 0.25$ *a.u.*



(B) Asymmetric case, using $N_{bf} = 200$ $\Delta t_{trn} = 0.15$ *a.u.*

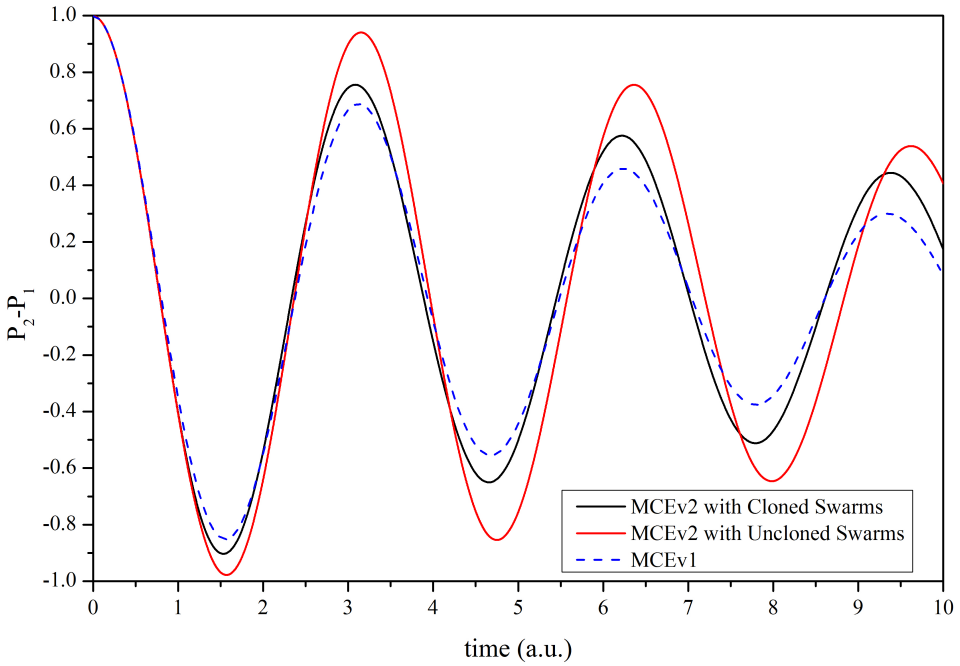
FIGURE 3.14: Comparisons of the single-train-type MCEv2 simulations with the most acceptable parameter sets and $N_{rpt} = 256$ repetitions with the swarm-type MCEv2 simulations and those from the MCTDH method [82, 83]

instability and misshapen oscillations, especially as the wavefunction decays into the second electronic state. The results improve somewhat for $\Delta t_{trn} = 0.15$, with instability only seen in the $N_{bf} = 20$ and $N_{bf} = 50$ results and well defined oscillations seen for all other results. The results for the larger two basis sets appear to begin converging also for this separation. Increasing the separation past this point causes convergence to be lost slightly, as seen in the plot for $\Delta t_{trn} = 0.20$. The reason for this is most likely that increasing this separation by even 0.02 *a.u.* causes the initial norm to be too low, indicating that past $\Delta t_{trn} = 0.15$ the basis set is not dense enough to properly describe the motion of the wavefunction. As such it would appear that the best separation value would be $\Delta t_{trn} = 0.15$, and as the smaller of the two largest basis sets seems more sensitive to the changes in the separation, using a train length of $N_{bf} = 200$ would be more prudent.

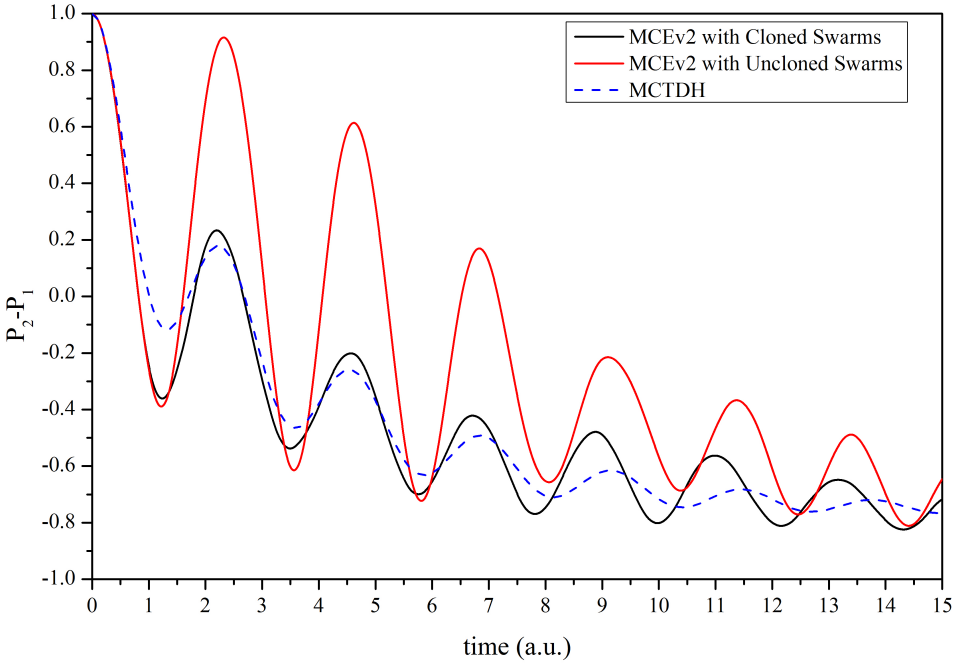
As the most acceptable parameter sets have been selected for both the symmetric and asymmetric cases it is now possible to assess the improvement that using a train-type basis set would make to the simulation of the spin boson model. A direct comparison of the MCEv2 simulations using both swarm-type and train-type basis sets against the MCTDH benchmarks is given for the symmetric case in figure 3.14a and for the asymmetric case in figure 3.14b. In the symmetric case the oscillations are noticeably damped by the inclusion of trains, however not so much as to agree with the MCTDH benchmark. Similarly, for the asymmetric case the oscillations are damped to a greater extent, however the wavefunction does not decay on to the second electronic state as quickly as for the MCTDH benchmark and the oscillations are still much larger than desired. From this it can be deduced that a train-type basis set cannot, on its own, fully improve propagation of the basis set for simulations with the MCEv2 method.

3.6.2 Tests for the Effectiveness of Basis Set Cloning

As the cloning procedure increases the basis set size greatly, this can cause problems with the system requirements of the simulation. While a simulation using a swarm of $N_{bf} = 50$ basis functions will take usually less than an hour to complete normally, when cloning is included this can increase the basis set by a factor of up to $2^{N_{cln}}$ for N_{cln} cloning events, increasing the runtime to a matter of days, and furthermore the memory requirements increase from the range of tens on MB to a few GB. As such limits must be put on the amount of cloning allowed. Obviously a larger N_{cln} is better as too few cloning events will mean that the simulation cannot benefit properly from the procedure. In the symmetric case of the spin boson model we set $N_{cln} = 4$, as in this case exponential growth of the basis set is encountered. For the asymmetric case the situation is slightly different, as for this system the wavefunction as a whole is decaying onto the second



(A) Symmetric case



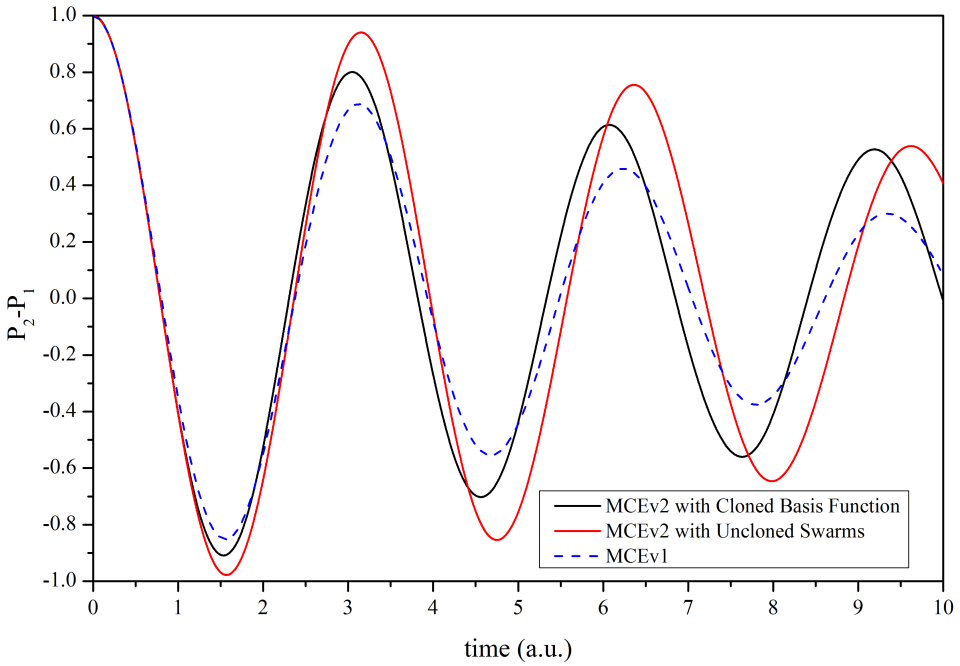
(B) Asymmetric case

FIGURE 3.15: Comparisons of the population differences from cloned MCEv2 simulations with those from the uncloned MCEv2 simulations, both using swarm-type basis set and $N_{rpt} = 100$ repetitions. These are also compared against those from MCTDH simulations [82, 83].

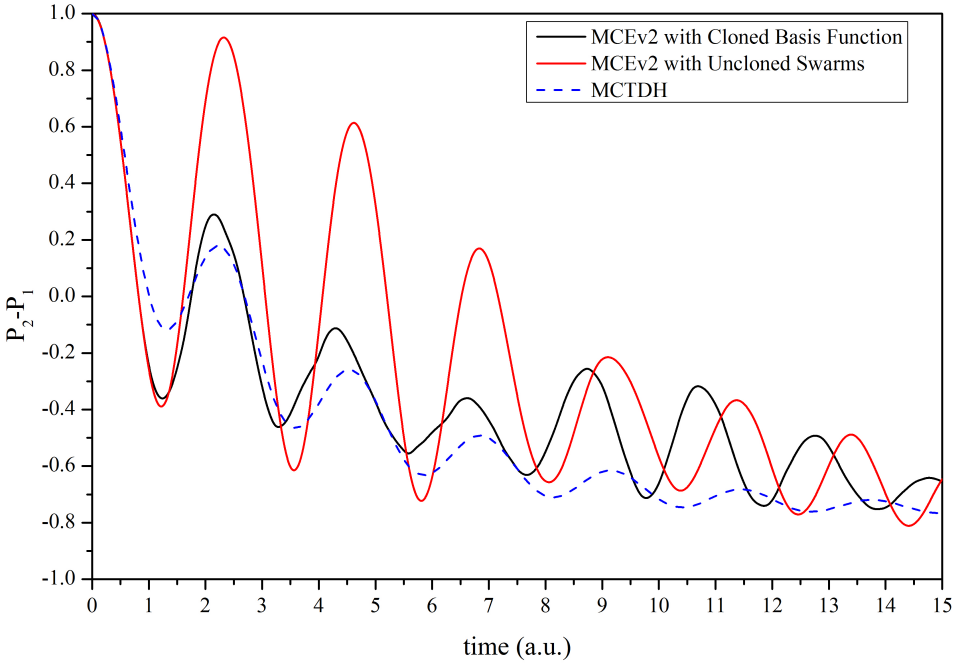
electronic state and so once the cloned basis functions are placed wholly onto the two states it will only be the function placed on the first electronic state which will experience cloning again. As such the basis set only grows by a factor of $N_{cln} + 1$ which allows much more cloning to occur before the size of the basis set becomes unmanageable. As such for this case we set $N_{cln} = 8$.

Figures 3.15a and 3.15b show the degree to which cloning can improve the MCEv2 method for the spin boson model for the symmetric and asymmetric cases respectively. As with the application of basis set trains, cloning dampens the overlarge oscillations in the population difference from the MCEv2 method, bringing the result closer the MCTDH benchmark simulations. Also like the application of trains, this dampening is not sufficient to bring the two results into complete agreement. A similar situation is seen for the asymmetric case, where the oscillations are much smaller than for the standard MCEv2 method but still too large. It should be noted however that despite the larger oscillations, with cloning the wavefunction seems to decay onto the second electronic state at the same rate as for the MCTDH benchmark. This is in contrast to the result when using trains, where the wavefunction appears to decay much slower, ending with a higher population difference than expected. It should also be noted that when the cloning procedure is applied there is no instability seen in the oscillations, unlike for basis set trains.

The cloning procedure is a powerful tool and can even improve the results starting with only a single basis function (i.e. a single coherent state at $|\mathbf{z}_0\rangle$ located using equation (3.3.9) rather than a swarm centred around this point found using equation (3.3.8)), as is shown in figures 3.16a and 3.16b. While not as good as for a cloned swarm of basis functions (as one would expect) there is a definite improvement in these cases. It is apparent however that once cloning stops the increased spread of the basis functions can be lost quickly when the basis set is so small, and at that time the population difference will tend back towards the uncloned MCEv2 result. It is also apparent that in the results for the asymmetric case for the small basis the oscillations are poorly formed, however this is not present for the larger cloned swarm so should not be an issue for future simulations. It is extremely promising that cloning of such a small basis set can have such a marked result; nevertheless, these results, however promising, indicate that on its own cloning is not sufficient to correct the discrepancy between the MCEv2 results and the MCTDH benchmark.



(A) Symmetric case



(B) Asymmetric case

FIGURE 3.16: Comparisons of the population differences from MCEv2 simulations using a single cloned basis function at $|\mathbf{z}\rangle = |\mathbf{z}_0\rangle$, with those from the uncloned MCEv2 simulations, both with $N_{rpt} = 100$ repetitions. These are also compared against those from MCTDH simulations [82, 83].

3.6.3 Tests for the Effectiveness of the Combination of Basis Function Cloning with Basis Function Trains

While neither of the improvements considered are sufficient to correct the MCEv2 method alone, it is entirely possible that a combination of the two can be used successfully. In light of this it was decided that this should be tested. Figures 3.17 and 3.18 show the population differences for a cloned single train with initial basis set $N_{bf} = 50$ in size. Due to the increase in basis set size as a result of cloning, it is entirely probable that the longer train needed for the asymmetric case will be unnecessary. The results for the cloning of a single basis set given in the previous section would seem to give weight to this. The amount of cloning experienced by the basis set is an important consideration. In the previous section the assumption is made that a greater number of cloning events is better, and so for the asymmetric case we set $N_{cln} = 8$. While this assumption is logical it requires testing, especially when one considers that in the asymmetric case it is only the basis functions placed in the $|1\rangle$ electronic state which undergo cloning, and the amplitudes associated with the basis functions in this state become ever smaller with each successive cloning event. As such, there is a possibility that a lesser number of cloning events are needed and so figure 3.18 also includes the population differences from basis sets which undergo only 4 cloning events. Due to the extra computational expense caused by the exponential scaling of the basis set when the symmetric case undergoes cloning, we keep $N_{cln} = 4$ for figure 3.17. The dynamics of the wavefunction cause each basis function to be cloned around every $0.75\ a.u. - 0.8\ a.u.$ and so the majority of cloning for $N_{cln} = 4$ would take place in the first $4.5\ a.u.$ of the simulation. Therefore any deviation caused by insufficient cloning can be expected to be seen after this time.

In figure 3.17 it can be seen that the agreement between the population differences from the MCEv2 simulations and the MCTDH benchmark calculations [83] are in much better agreement than was see previously. The agreement is, in fact, almost to the same level as is given by the MCEv1 method with only a slight difference in the period of the last half oscillation. The degree to which the population differences agree at this stage is unsurprising as the symmetric case is considered to be one of the easier cases of the spin boson model to simulate.

Figure 3.18 is a different matter. While the agreement for both the given population differences is again much closer to the MCTDH result [82] than using trains or cloning alone, the level of agreement is not yet at the level of the MCEv1 simulations. The expected discrepancy between the results as a consequence of insufficient cloning starting around $t = 4.5\ a.u.$ is seen, with the oscillations becoming less dampened once cloning

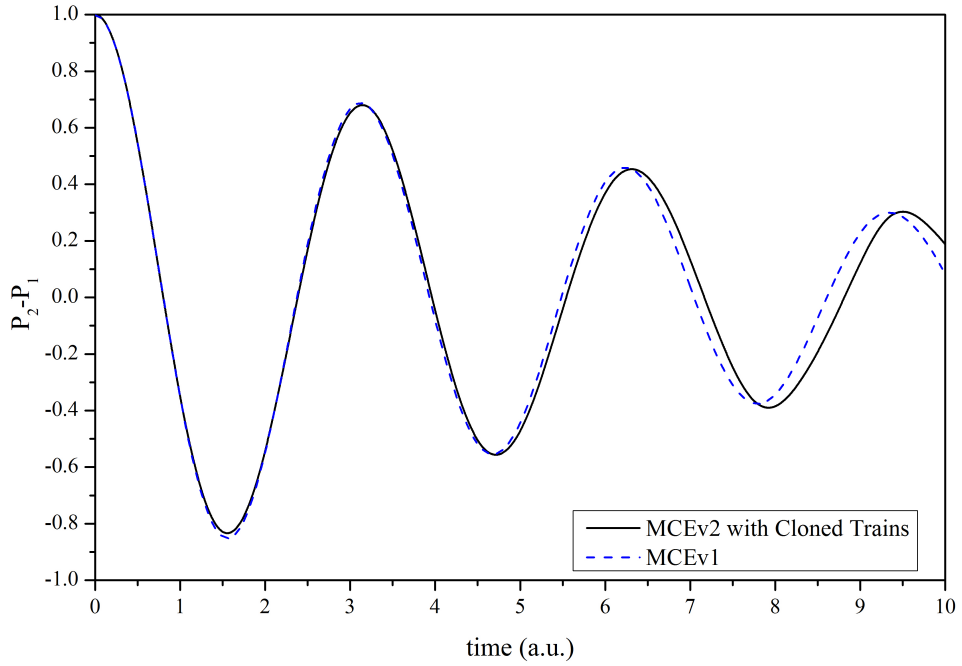


FIGURE 3.17: Comparison of the population difference for the symmetric case of the spin boson model from the single-train-type basis set MCEv2 method using $N_{rpt} = 100$ repetitions with basis function cloning against that from the MCTDH benchmark calculations [83], with initial basis set of $N_{bf} = 50$ basis functions

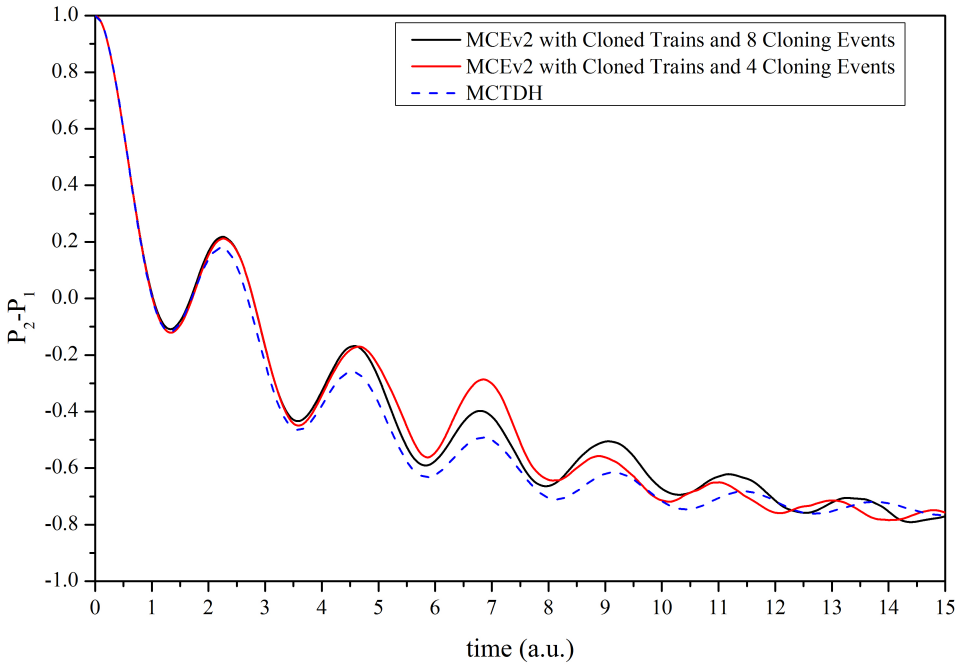


FIGURE 3.18: Comparison of the population difference for the asymmetric case of the spin boson model from the single-train-type basis set MCEv2 method with basis function cloning using either 4 or 8 cloning events and $N_{rpt} = 100$ repetitions against that from the MCTDH benchmark calculations [82], with initial basis set of $N_{bf} = 50$ basis functions

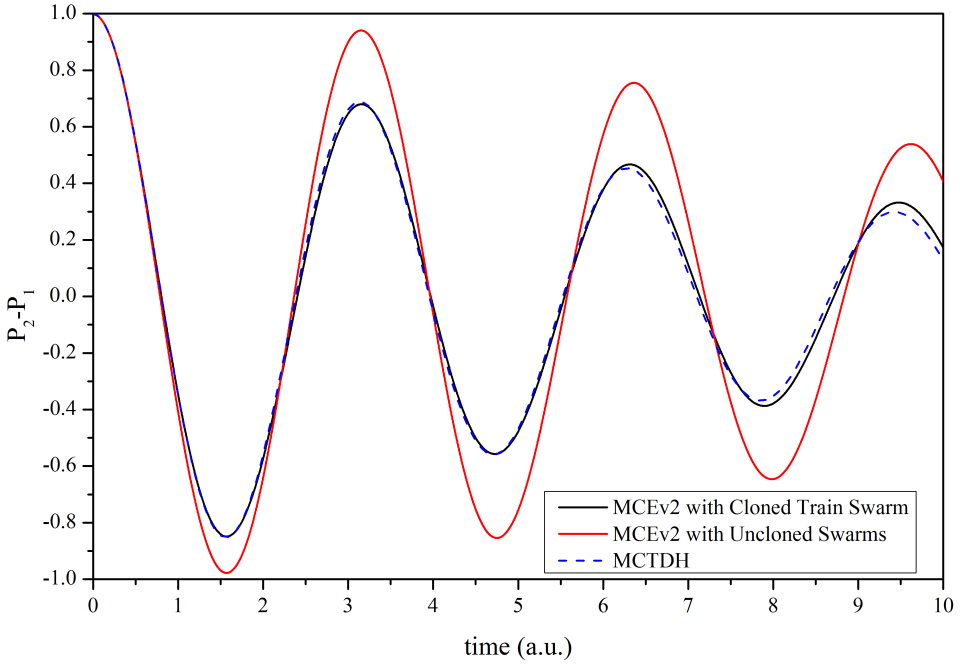
has ceased before decaying rapidly onto the second electronic state towards the end of the simulation.

It can be determined therefore that a larger number of cloning events gives the best results; however, these results still do not agree fully with the MCTDH benchmark. The disagreement could be a consequence of the fact that when only a single train is used, even when cloning is present, every single basis function in the basis set will follow the same trajectory. This may result in the motion of the wavefunction being improperly described. A possible solution would be to construct the basis set as a hybrid of trains and swarms.

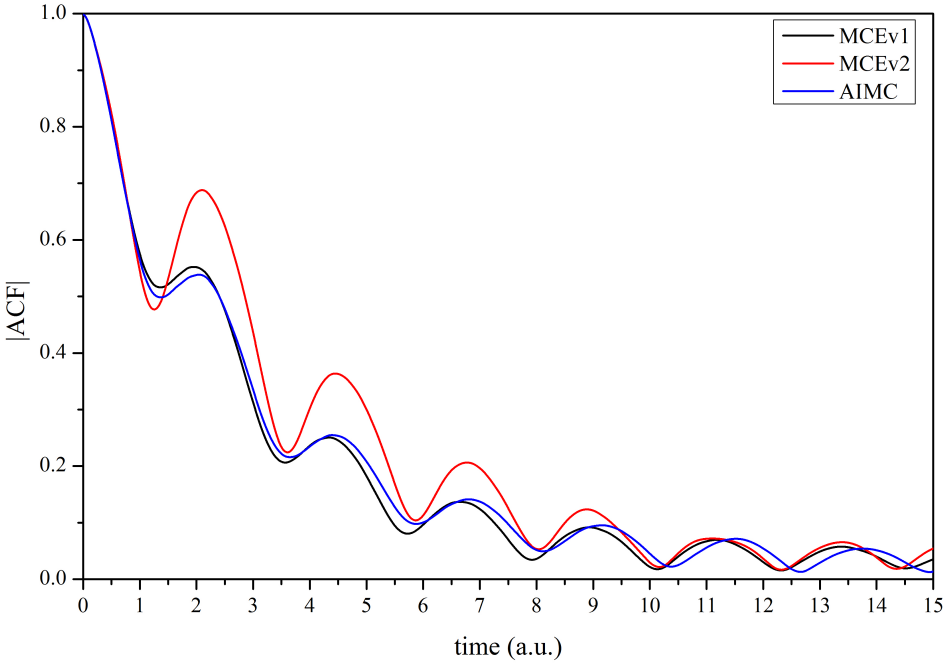
3.6.4 Basis Function Cloning with Swarms of Basis Function Trains

A basis set can be constructed which has the benefit of the regularity and smoothing afforded by trains, but also covers a larger area in phase space by starting with a small swarm of basis functions biased to $|\mathbf{z}_0\rangle$ rather than a single central basis function with position $|\mathbf{z}_0\rangle$. If this swarm is then propagated backwards and forward in the manner described in section 3.5.1, the basis set will take the form of a swarm of basis function trains. This swarm of trains can then undergo cloning as was done with single trains in the previous section. The construction of a swarm/train type basis set could possibly give better convergence and, it is hoped, better agreement with the MCTDH simulations. An important consideration with this method however is the necessity of larger basis sets. A swarm of 20 trains, each only 10 basis functions in length, would have an initial size of $N_{bf} = 200$, growing to $N_{bf} = 3200$ for the symmetric case with 4 cloning events and to $N_{bf} = 1800$ for the asymmetric case with 8 cloning events. This represents a significant cost in terms of memory and computing expense. The size of basis set discussed for the symmetric case would put the memory and time requirements for the simulation beyond what is feasible for a system based on OpenMP parallelisation (see Appendix A.4), requiring weeks of simulation time and well over 32 GB of memory. As such, for this case the basis set was constructed as a swarm of 10 trains, each 10 basis functions in length for an initial basis set of $N_{bf} = 100$ and a final basis set of $N_{bf} = 1600$. This test uses $N_{rpt} = 100$ repetitions, which is sufficient to attain convergence.

Figure 3.19a shows an excellent level of agreement with the MCTDH benchmark, slightly improved from the cloned single train result in figure 3.17. As with the cloned single train, the agreement is not as good at the very end of the oscillations however in light of the difference due to a lower number of cloning events seen in figure 3.18 this could be simply due to the fact that only four cloning events occur during propagation. Definitely, the agreement is complete for the first 2 oscillations. The improvement seen in figure



(A) Symmetric case of the spin boson model with the parameters $\omega_c = 2.5$, $\alpha_k = 0.09$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 0$ with $M = 50$ degrees of freedom and $N_{rpt} = 100$ repetitions. The basis set is constructed as 10 trains, 10 basis functions in length with $\Delta_{trn} = 0.25$ *a.u.*



(B) Asymmetric case of the spin boson model with the parameters $\omega_c = 7.5$, $\alpha_k = 0.1$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom and $N_{rpt} = 100$ repetitions. The basis set is constructed as 20 trains, 10 basis functions in length with $\Delta_{trn} = 0.15$ *a.u.*

FIGURE 3.19: Comparisons of the population differences for cloned MCEv2 simulations using a swarm/train type basis set against those from uncloned swarm-type MCEv2 simulations and those from the MCTDH benchmark calculations [82, 83]

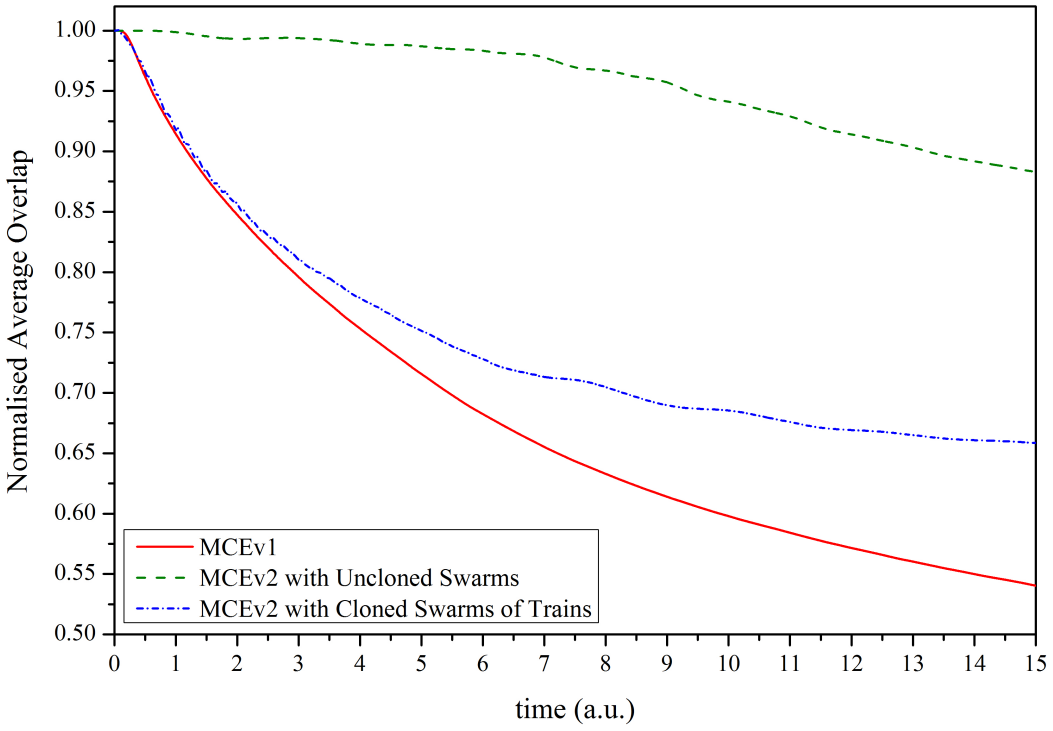


FIGURE 3.20: Comparison of the normalised average overlap between the coherent states for both formulations of the MCE equations with uncloned swarm type basis sets and the cloned swarm-of-trains type basis sets with the MCEv2 equations for the asymmetric case of the spin boson model, using $N_{bf} = 200$ basis functions and the parameters $\omega_c = 7.5$, $\alpha_k = 0.10$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom, averaged over $N_{rpt} = 100$ repetitions.

3.19b is much better than is seen for a single cloned train with either four or eight cloning events. For the majority of propagation, the agreement is almost complete, with a slight overestimation of the oscillations towards the end of the simulation. This result is extremely encouraging, especially when one considers that the final cloning event occurs around $t = 6.75$ *a.u.* and that it is shortly after this that the discrepancies begin. Figure 3.20 confirms the earlier assertion that with cloning the basis functions are able to spread out more to cover a greater area in phase space, and it can be seen that the decay of the overlap with cloning is more in line with the way in which the basis functions spread when propagated using the MCEv1 equations. Again, however, the decay of the overlap levels out somewhat once cloning has ceased. It can therefore be reasonably expected that if more cloning events were allowed, the agreement with the benchmark calculations would persist for longer. This behaviour in the overlap is confirmed in figure 3.21, which shows the evolution of a swarm of trains undergoing cloning over the course of propagation. When compared against the MCEv2 plots in figure 3.6, it can be seen that the density of contour lines and the colour of the centre of the wavefunction decreases much more when the modifications to the MCE method

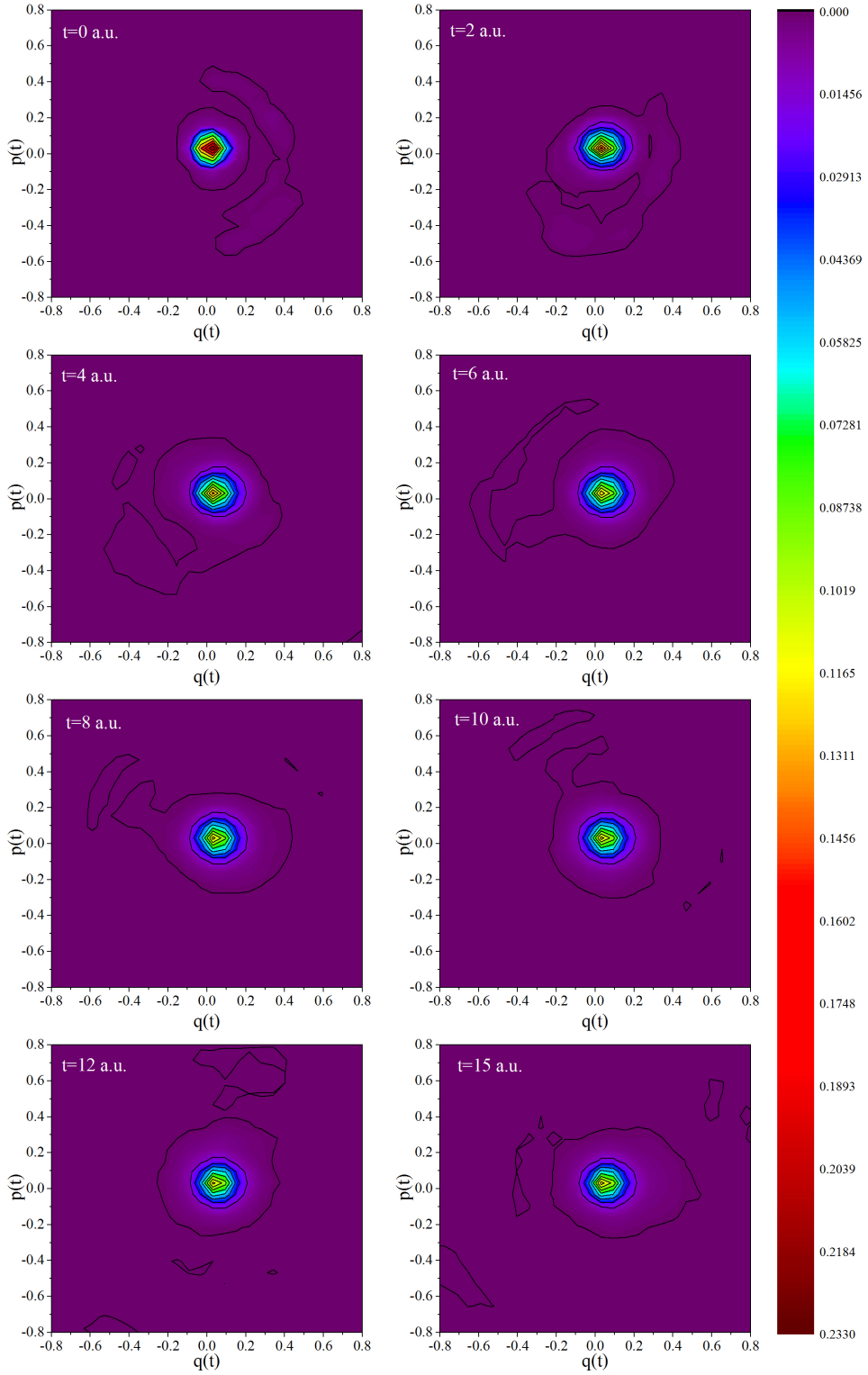


FIGURE 3.21: Density plots for the coherent states of an AIMC wavefunction for the asymmetric case of the spin boson model, using $N_{bf} = 200$ basis functions and the parameters $\omega_c = 7.5$, $\alpha_k = 0.10$, $\beta = 5.0$, $\Delta = 1.0$ and $\epsilon = 1.0$ with $M = 50$ degrees of freedom, shown at $t = 2$ *a.u.* intervals. To create plots, data for a set of 2D histograms was divided by the total number of points present and each pair of (q, p) coordinates for all basis functions and degrees of freedom is considered.

are included, more in line with the behaviour of the coherent states when acted upon by the MCEv1 equations. It should also be mentioned however that this decrease in the proportion of the wavefunction at the centre stops once cloning has ceased, confirming the conclusions inferred from figure 3.20. One interesting feature shown by figure 3.21 is that when compared with both the MCEv1 and MCEv2 density plots, the spread of the wavefunction in the central area for the simulation with modifications is in fact lower, with the area immediately around central point of the wavefunction encompassing around 23% of the total wavefunction coordinates at $t = 0$, while in the MCEv1 and MCEv2 simulations this number is closer to 8.5%. This is thought to be due to the smaller swarm used as central points to construct the trains which by necessity is more compressed than a simple swarm-type basis set and so it is possible that a larger initial swarm would change this. Despite the larger initial concentration of the wavefunction in the centre, it can be seen that more coherent states are present in the area of phase space away from this central region than is seen in either the MCEv1 or MCEv2 density plots in figure 3.6. The level of improvement in the results afforded by the modifications to the propagation of the basis set is proven through figure 3.19, and, as such, this represents the main result of this chapter.

3.7 Conclusion

In summary, this chapter has investigated the disagreements between the two formulations of the Multi-Configurational Ehrenfest method when simulating a two-level model system, and the procedures necessary to correct this disagreement. It has been seen that for a system which cannot be accurately simulated with the unmodified MCEv2 method, the combination of using swarms of basis function trains when sampling the basis set and basis set cloning during propagation with the MCEv2 method can generate results with an excellent level of agreement to benchmark calculations. It has also been shown that this combination is necessary, as using just one of these modifications is not sufficient to correct the disagreement entirely. While the computational cost of the MCEv2 method with these modifications is high compared to that of MCTDH for the spin boson model, it has been shown that comparative levels of accuracy are achievable

This result is of great significance as it confirms the validity of the methods applied by Makhov *et al.* [1, 2] in their simulations of ultrafast processes in small organic molecules. In such simulations, due to the structure of the MCEv2 equations, the single configurational equations can be run separately to the calculations for the cross-configurational amplitudes. It is the single configurational equations which contain computationally expensive electronic structure calculations and so separating these from the rest of the

simulation can be beneficial. If information is saved at each time step, this can speed up calculations significantly, as each individual configuration propagated can be made into a basis function train at no extra cost, meaning that the swarm of trains basis set is effortlessly constructed. Cloning also only occurs on this level, meaning that the cross-configurational calculations can be run independently of the modifications discussed in this chapter. In confirming the validity of the modified MCEv2 method, the original motivation for creating the second formulation of these equations is finally and fully realised.

Further simulations of the spin boson model are ongoing, as the investigation of only two parameter sets is not sufficient to say definitively that this method will correct any disagreements between the two methods in all situations. A possible area where this may not work would be for a simulation of localisation at $\beta = \infty$ (i.e. zero temperature) such is presented in ref [23] Fig 6, as in this case the wavefunction does not approach an intersection meaning that cloning would not occur.

Chapter 4

Using Adaptive Basis Sets for Simulation of High Harmonic Generation with the Coupled Coherent States method*

4.1 Introduction

In the previous chapter, modification of the basis set during propagation was investigated through the use of basis function cloning. This technique was designed to account for the equations of motion for the coherent states not allowing the basis set to spread out in phase space to enough of an extent to properly describe the dynamics of the wavefunction. In this chapter, a different problem is investigated, where the coherent states, guided by Hamiltons equations, become too widely spaced during propagation of the wavefunction. This results in a decrease in the overlap between the basis functions, and thus the coupling between the basis functions is lost.

By periodic reprojection of the wavefuction onto a new basis, the coherent states can be prevented from becoming too widely spaced. Provided the amplitude terms are properly recalculated and the sampling of the new basis set is acceptable, the wavefunction itself would remain properly described after such a process. As a test for this, a system in which the coherent states would be guided far away from the initial position and from each other is considered. This system is an electron in a strong laser field undergoing high-order harmonic generation.

*The work discussed in this chapter is the basis of a paper in Physical Review A[164].

High-order harmonic generation (HHG) is a phenomenon in which electrons bound to a nucleus respond in a highly nonlinear fashion to a strong input near-infrared field, generating high frequency radiation up to the extreme ultraviolet and soft X-ray regimes[165]. This phenomenon has many applications in spectroscopy due to its ability to create attosecond pulses of coherent light. Various theoretical methods have been developed to study this and other nonlinear optical effects with varying degrees of success. All methods have drawbacks though, from exponential scaling to improper behaviour under certain conditions. In this chapter, high harmonic generation from a single electron bound to a pseudo-atomic potential will be considered, and it will be shown that with modifications to the basis set during propagation the CCS method can carry out this simulation with a high level of accuracy.

4.2 High Harmonic Generation

Experimental results involving intense laser fields in the 1980s and early 1990s showed very unusual results. It was found that when a sufficiently intense laser pulse was applied to an atomic medium, the frequency of the driving radiation would be up-converted by up to two orders of magnitude[166]. The reason these results were seen to be so unusual is that to generate these harmonics the atom would have to absorb many photons, however the absorption of this many photons would be above the number needed for ionisation, i.e. I_p/ω where I_p is the ionisation energy of the atom. As a free electron should not absorb any more photons, the highest harmonics seen should not be much higher than this, however it was seen that, for example, where hydrogen in an intense 800nm field should release no harmonics higher than $N = 11$, harmonics upwards of $N = 100$ were being seen. Furthermore, when considering the harmonic spectrum, the harmonic peaks did not fall off as some power law of the intensity but instead exhibited a singular structure, in that the harmonics rapidly fell off for the first few harmonics, followed by a plateau structure where a series of odd harmonics of comparable strengths were seen up to a high harmonic, at which point the harmonics exhibited a sharp cutoff and fell off rapidly. An example of such a high harmonic spectrum is given in figure 4.1. For a single atom or molecule, the harmonic spectrum can be found by taking the Fourier transform of the dipole acceleration

$$S(\omega) \propto \left| \int_{-\infty}^{\infty} dt \left\langle \Psi(t) \right| \ddot{d}_z \left| \Psi(t) \right\rangle e^{i\omega t} \right|^2 , \tag{4.2.1}$$

for a pump linearly polarized along z [168]. It has been shown by Protopapas *et al.* that the HHG spectrum requires a few cycles to build up[169], as can be seen in figure 4.2. This shows that it is only after a few cycles that the harmonic peaks appear,

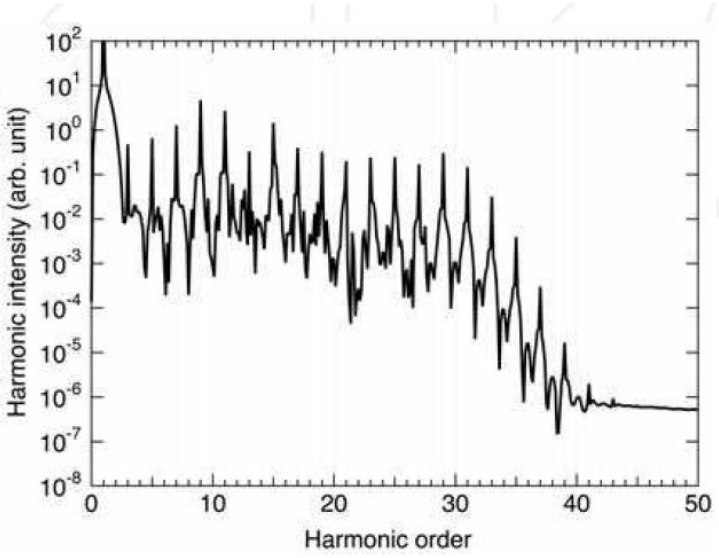


FIGURE 4.1: Example of a high harmonic spectrum showing the plateau of odd harmonics and the cutoff. From [167]

although the general shape of the spectrum is visible right from the first half cycle of the laser, i.e. the first recollision. The singular shape of the harmonic spectrum, originally seen experimentally, was backed up by numerical calculations using the time dependent Schrödinger equation [170] which placed the cutoff in the region of

$$E_c \approx 3.2U_p + I_p, \tag{4.2.2}$$

where U_p is the ponderomotive energy, the cycle-average energy of electron oscillations in the laser field equal to $U_p = \mathcal{E}_0^2/4\omega_0^2$ for a laser with amplitude $\mathcal{E}_0$ and frequency ω_0. While such simulations confirmed the high harmonic generation, an explanation of the mechanism behind it was still missing. This conundrum was answered with the semiclassical three-step model [171–173].

4.2.1 Explaining High Harmonic Generation with the Three-Step Model

As the name suggests the three step model explains the behaviour of the atom/light interaction by splitting it into three parts, all of which take place over a single optical period. Firstly, the combination of the intense field with the atomic potential increases the tunnelling probability of the valence electron. Secondly, once the electron is freed from the atomic potential it is accelerated by the laser field. Finally, it is driven back towards the parent ion where it recombines, releasing a high energy photon[1]. The three step model acts under the following assumptions:

[1]It should be noted that the three step model, as well as explaining the mechanism behind high harmonic generation also provided explanations for other non-linear optical effects, namely above-threshold

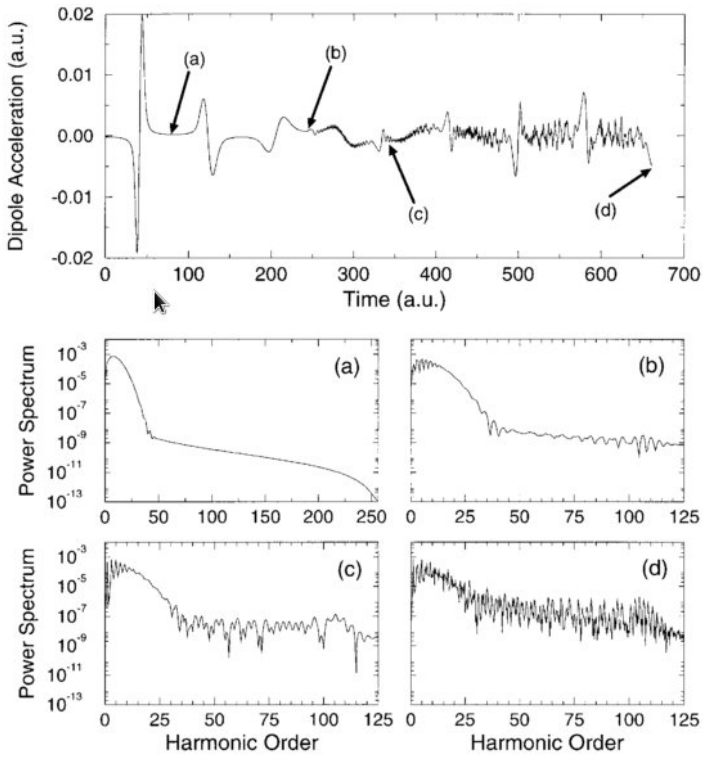


FIGURE 4.2: Comparisons of the HHG spectra taken at different times along a simulation, showing how the spectrum pattern builds up over multiple laser cycles, with harmonic spectra calculated after (a) one half laser cycle, (b) 1.5 laser cycles, (c) 2 laser cycles and (d) 4 laser cycles. From [169]

(a) The electron is born in the continuum at any time within the quarter of the laser cycle;

(b) The electron is born near the ionic core with zero velocity;

(c) The electron returns to the ionic core and its instantaneous energy of return is converted into the harmonic photon;

(d) The pull of the ionic core on the electron can be neglected.

This model can be used to explain the key features of the harmonic spectrum (the plateau, the cutoff and the existence of only odd harmonics). The cutoff can be explained as originating from the maximum kinetic energy an electron can acquire from a monochromatic laser. Numerical calculations by Corkum [171] for a helium atom in a $\lambda = 800nm,\ I = 5 \times 10^{14} W cm^{-2}$ laser field shed some light on this. Through classical mechanics it was shown that half of the electrons that are field ionised by linearly

ionisation (ATI) and non-sequential double ionisation (NSDI). The study of these phenomena comprises a large field itself and is beyond the scope of this work.

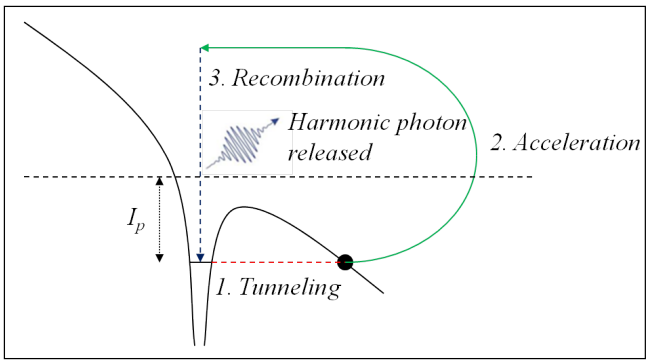


FIGURE 4.3: Illustration of the three-step model

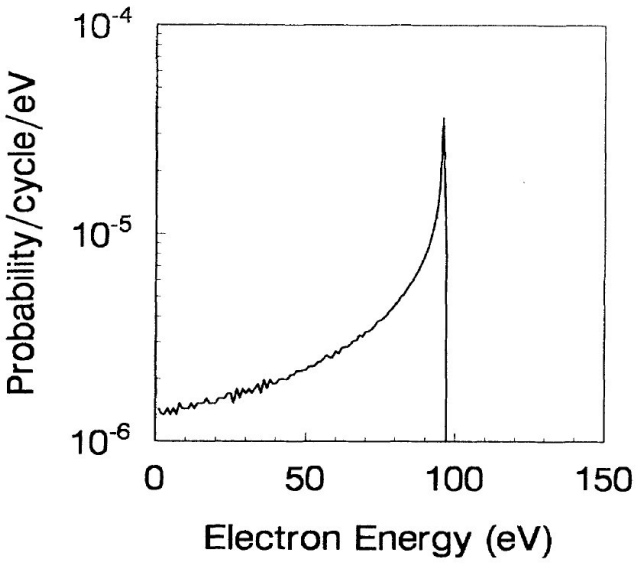


FIGURE 4.4: The energy distribution of electrons at the first encounter with the parent ion in the case of helium with $\lambda = 800nm$, $I = 5 \times 10^{14} Wcm^{-2}$, from [171]

polarised light pass the ionic core once during the first laser period following ionisation, and in figure 4.4 the probability per unit energy per laser cycle of an electron with a particular kinetic energy passing the core in this time is given. This shows clearly that there is a maximum value of the kinetic energy which was found to be at $E_k = 3.17U_p$, meaning that the maximum total energy that can be released upon recombination is

$$E_c = 3.17U_p + I_p. \tag{4.2.3}$$

This gives a physical basis for the cutoff seen in high harmonic spectra and agrees with the approximate value found by Krause *et al.*[170].

It has been shown, through the use of the Floquet theorem, that due to parity transformations in the presence of a symmetric potential, only the odd Fourier components of a transform of the dipole will result. This is the origin of the odd harmonics rule [174]. It

should be noted however, that when the target is a molecule rather than a single atom, the breakdown of the Born-Oppenheimer approximation will result in violation of the odd harmonics rule [175].

The formation of the plateau has its roots in interference between the trajectories in the HHG process. Through a comparison of high harmonic spectra found numerically from full quantum mechanical, semiclassical Herman-Kluk, and classical Monte-Carlo calculations, it was seen that the plateau was only visible for the former two methods, indicating an interference effect [176]. By identifying different classes of trajectories determined by initial momenta, it was found that this was the interference of free trajectories with trajectories whose return to the core has been delayed, deemed "stranded" (delayed above the barrier of the effective potential $V_{eff}(x) = V(x) - x\mathcal{E}_0$) and "trapped" (subject to chaotic behaviour). This interference has also been associated with high frequency oscillations in the acceleration form of the dipole moment, found using TDSE computations [169] and the HK propagator [177, 178] for cases where an initial wave packet starts far from the core.

4.2.2 Previous Theoretical Work on High Harmonic Generation

For theoreticians, HHG poses an interesting problem in that by its nature it is difficult to calculate. Many theoretical approaches have been established to compute HHG spectra. The numerical solution of the time-dependent Schrödinger equation contains no physical approximations, including both the field and the potential, and is straight forward for one-dimensional, one-electron systems [179, 180], but as the numerical effort increases exponentially with the number of degrees of freedom, it is not applicable to a system in which large amplitude motion requires a very large basis set even in one dimension such as is the case for HHG. As such the TDSE has at most been employed in realistic models of helium [181]. This is a serious obstacle towards modelling correlated multi-electron dynamics in the attosecond regime, which is important as these dynamics play a major role in many strong-field phenomena such as, for example, electron migration and attosecond hole creation (for a review see, e.g., [182] and references therein). Furthermore, as it lacks the clarity of an orbit-based picture it is difficult to define a physical interpretation using this method unless this information is extracted indirectly. Typical examples include the use of windowed Fourier transforms, both in the time-frequency and in the intensity/reciprocal-intensity domains in order to extract such information [183, 184].

One may however employ semi-analytical approaches such as the strong field approximation (SFA). The SFA is numerically inexpensive and underlies many of the current

analytical approaches to HHG and also other strong-field phenomena, such as above-threshold ionisation (ATI) or non-sequential double ionisation (NSDI), providing a transparent physical picture for quantum interference in these phenomena [185–187]. The SFA has the advantage that the propagator can be found analytically, giving the quantum mechanical description of the behaviour of a free electron in the laser field [166], and also allows an intuitive physical interpretation in terms of trajectories. When using the SFA however, the laser field is neglected when the electrons are bound to atoms or molecules, the Coulomb potential is neglected when the electrons are in the continuum, and the internal atomic structure is over simplified[167]. It has further problems in that it is not gauge invariant[166], and the continuum states, when approximated by field-dressed plane waves, are no longer orthogonal to the bound states. This approximation poses serious difficulties when the interplay between the Coulomb potential and the external field becomes important, and can result in loss of translational invariance [188] and ambiguities in patterns related to structural interference [189]. Recently, Coulomb-corrected analytical approaches have been developed and successfully applied to strong field phenomena [190–193]. These approaches however require the external field to be dominant. In the specific case of HHG, the SFA has had problems in that there is an overall discrepancy between the intensity in the spectra, which is much larger in the TDSE, and that the SFA does not reproduce below-threshold and low-plateau harmonics in a satisfactory way, as these harmonics depend very strongly on the core structure. Futhermore, even in the cutoff region, a quantitative agreement between the SFA and the TDSE is only obtained if the SFA is modified to allow for a momentum and position spread of the returning electron, and continuum-to-continuum transitions [194]. It has also been reported in refs [183] and [184] that time-resolved spectra indicate that there is an over-enhancement of the long SFA orbit, in comparison with the TDSE. This has been attributed to the lack of spatial extension in the core in the SFA case[2].

A second approach introduced recently is time dependent R-matrix (TDRM) theory. This method has its roots in the R-Matrix introduced in the late 1940s [196, 197] to describe the boundary of the internal region for complex nuclear reactions, which was later expanded in the 1970s as an *ab initio* procedure for calculating accurate electron-atom and electron-ion collision cross sections[198, 199]. Further expansions of R-matrix theory by Burke and Burke introduced a time-dependent variant capable of solving a one-dimensional model problem [200] which was then extended to a three-dimensional method by Lysaght *et al.*twelve years later [201], capable of describing the rearrangement dynamics of complex multielectron atoms in intense ultrashort light pulses. The recent incarnation of R-matrix theory works by splitting the configuration space into an inner region in which electron exchange and correlation effects are important, and an external

[2]For a discussion of this spatial extension the reader is referred to reference [195].

region where only the scattered electron exists and is subject to long range potentials, with both regions acted upon by an input laser field. TDRM has shown itself to be a powerful tool for examining the dynamics of many-electron atoms in strong fields, and in the past few years it has been applied to the problem of harmonic generation for noble gasses, investigating the interference between competing pathways [202] and the influence of multiple ionisation thresholds [203] on generation of harmonic photons for the Ar^+ ion in an ultraviolet field, as well as the effect of different initial magnetic orientations of the Ar^+ ion on the harmonic spectrum [204]. These studies were only able to generate spectra with the first few harmonics since they were carried out at an intensity of 4×10^{14} W/cm^2, and it was found that higher intensities increased the ionisation probability to the extent that it was impossible to accurately determine the harmonic spectrum [205]. Later simulations of the Ne^+ ion were able to use intensities of up to 1×10^{15} W/cm^2, allowing generation of a full high harmonic spectrum [205], and simulation of HHG was also carried out for Kr and Xe atoms in a strong near-IR field [206]. Further work based on TDRM has been carried out by the group of O. Smirnova using a modification of TDRM referred to as analytical R-matrix (ARM) theory [207] which they have shown to be capable of simulating strong field photoionisation of the CO_2 [208] and NO_2 [209] molecules, with an aim for accurately simulating HHG spectra for molecules in the near future.

High harmonic generation is not just a phenomenon of interest to theoreticians, as it has useful applications, most notably in spectroscopy. If the recombination occurs to the exact same state that the electron left from then the phase of the emitted high harmonic photons is the same from one atom to another. As such, coherent radiation can be generated in the medium. This makes it ideal for two major applications. Firstly, high harmonic emission can generate attosecond pulses of light for use in time-resolved pump-probe experiments [210–212]. Secondly, it can generate coherent light sources in the extreme ultraviolet [213–215] and soft X-ray [216–218] frequency ranges which can be used in attosecond molecular imaging [219, 220] with a combination of sub-Angstrom spatial and attosecond temporal resolution [166].

4.3 High Harmonic Generation with Coupled Coherent States

The above problems serve as motivation for the development of a method able to model strong field phenomena which is orbit-based but makes no simplifications on the target

or binding potential. The Coupled Coherent States (CCS) approach[3] is such a method, allowing simulation of multi-dimensional many-body quantum dynamics without the constraints of exponential scaling present in the TDSE, while allowing the treatment of the binding potential and of the laser field on an even footing unlike the SFA. In the past variations of the CCS method have shown themselves capable in a strong field environment, accurately describing the ionization yield from the ATI of helium [75] as well as the low energy structure and angular distribution of photoelectrons from NSDI in intense long-wavelength laser fields [221].

The Hamiltonian for an electron acted upon by a strong laser field is given in atomic units by

$$\hat{H} = \frac{\hat{p}^2}{2} + V_a + V_{\mathcal{E}}, \tag{4.3.1}$$

where V_a is the binding potential and V_ε is the interaction potential with the laser field in the length gauge. For the sake of simplicity and as a proof of principle we consider only the motion of the electron along the principle axis of the laser field as a one-dimensional system. As the more commonly used soft core potential or one-dimensional Coulomb potential cannot be found analytically in the coherent states z-notation, the binding potential used is a 1D short range Gaussian potential $V_G(x) = -\exp\left[-\lambda x^2\right]$ with a width of $\lambda = 0.5$.

The kinetic part of the Hamiltonian in equation (4.3.1) is easy to find through substitution of equation (2.3.5). From combination of $\mathbf{z}^*$ and $\mathbf{z}$, it is possible to get

$$\begin{aligned} \mathbf{z}^* - \mathbf{z} &= -i\sqrt{\frac{2}{\gamma}}\mathbf{p} \\ \therefore \mathbf{p} &= i\sqrt{\frac{\gamma}{2}}\left(\mathbf{z}^* - \mathbf{z}\right). \end{aligned} \tag{4.3.2}$$

From this, it is possible to calculate the kinetic energy term of the Hamiltonian in z-notation, such that

$$\begin{aligned} \left\langle \mathbf{z}_j \middle| \frac{\mathbf{p}^2}{2} \middle| \mathbf{z}_k \right\rangle &= -\frac{\gamma}{4}\left\langle \mathbf{z}_j \middle| \mathbf{z}^{*2} - \mathbf{z}^*\mathbf{z} - \mathbf{z}\mathbf{z}^* + \mathbf{z}^2 \middle| \mathbf{z}_k \right\rangle \\ &= -\frac{\gamma}{4}\left\langle \mathbf{z}_j \middle| \mathbf{z}^{*2} - 2\mathbf{z}^*\mathbf{z} - M + \mathbf{z}^2 \middle| \mathbf{z}_k \right\rangle \\ &= -\frac{\gamma}{4}\left(\mathbf{z}_j^{*2} - 2\mathbf{z}_j^*\mathbf{z}_k - M + \mathbf{z}_k^2\right)\left\langle \mathbf{z}_j \middle| \mathbf{z}_k \right\rangle, \end{aligned} \tag{4.3.3}$$

where M appears due to the commutator

$$\left[z_k, z_j^*\right] = \delta_{jk} \tag{4.3.4}$$

[3]The equations for the Coupled Coherent States method are given in sections 2.3.1 and 2.3.2, and the ansatz used is the one given in the latter of these sections, ie $|\Psi(t)\rangle = \sum_k D_k e^{iS_k} |z_k\rangle$.

being applied for every degree of freedom. In practise, this would be applied as

$$\left\langle \mathbf{z}_j \right| \frac{\mathbf{p}^2}{2} \left| \mathbf{z}_k \right\rangle = -\frac{\gamma}{4} \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \sum_{m=1}^{M} \left(z_j^{*(m)2} - 2 z_j^{*(m)} z_k^{(m)} - 1 + z_k^{(m)2} \right). \tag{4.3.5}$$

The conversion of the pseudo-atomic potential is less straight forward to calculate, requiring integration using

$$\left\langle \mathbf{z}_j \right| \hat{O}(\mathbf{x}) \left| \mathbf{z}_k \right\rangle = \int_{-\infty}^{\infty} \left\langle \mathbf{z}_j \right| \left. \mathbf{x} \right\rangle \left\langle \mathbf{x} \right| \hat{O} \left| \mathbf{z}_k \right\rangle d^M\mathbf{x} = \int_{-\infty}^{\infty} \left\langle \mathbf{z}_j \right| \left. \mathbf{x} \right\rangle \left\langle \mathbf{x} \right| \left. \mathbf{z}_k \right\rangle \hat{O}(\mathbf{x}) d^M\mathbf{x}, \tag{4.3.6}$$

and

$$\left\langle \mathbf{z}_j \right| \left. \mathbf{x} \right\rangle \left\langle \mathbf{x} \right| \left. \mathbf{z}_k \right\rangle = \left(\frac{\gamma}{\pi} \right)^{\frac{M}{2}} \exp\left[-\gamma \left(\mathbf{x} - \boldsymbol{\rho}_{jk} \right)^2 \right] \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle, \tag{4.3.7}$$

where

$$\boldsymbol{\rho}_{jk} = \frac{\left(\mathbf{z}_j^* + \mathbf{z}_k \right)}{\sqrt{2\gamma}} = \sum_{m=1}^{M} \frac{\left(z_j^{(m)*} + z_k^{(m)} \right)}{\sqrt{2\gamma}}. \tag{4.3.8}$$

Equation (4.3.7) is derived in Appendix B.1.1, and in the above equations, the vector $\mathbf{x}$ denotes a spatial vector with elements along all axes, and should not be confused with the spatial x axis which will be used later. As such, for the inverted Gaussian potential in an arbitrary number of spatial dimensions, $V_G = -e^{-\lambda \mathbf{x}^2}$,

$$\begin{aligned} \left\langle \mathbf{z}_j \right| - e^{-\lambda \mathbf{x}^2} \left| \mathbf{z}_k \right\rangle &= -\int_{-\infty}^{\infty} e^{-\lambda \mathbf{x}^2} \left\langle \mathbf{z}_j \right| \left. \mathbf{x} \right\rangle \left\langle \mathbf{x} \right| \left. \mathbf{z}_k \right\rangle d^M\mathbf{x} \\ &= -\left(\frac{\gamma}{\pi} \right)^{\frac{M}{2}} \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \int_{-\infty}^{\infty} \exp\left[-\lambda \mathbf{x}^2 - \gamma \left(\mathbf{x} - \boldsymbol{\rho}_{jk} \right)^2 \right] d^M\mathbf{x} \\ &= -\left(\frac{\gamma}{\pi} \right)^{\frac{M}{2}} \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \int_{-\infty}^{\infty} \exp\left[-\left(\lambda + \gamma \right) \mathbf{x}^2 + 2\gamma \mathbf{x} \boldsymbol{\rho}_{jk} - \gamma \boldsymbol{\rho}_{jk}^2 \right] d^M\mathbf{x}. \end{aligned} \tag{4.3.9}$$

Using the well known Gaussian integral identity that

$$\int_{-\infty}^{\infty} \exp\left[-ax^2 + bx + c \right] dx = \sqrt{\frac{\pi}{a}} \exp\left[\frac{b^2}{4a} + c \right], \tag{4.3.10}$$

this, when integrated over all degrees of freedom, becomes

$$\begin{aligned} \left\langle \mathbf{z}_j \right| - e^{-\lambda \mathbf{x}^2} \left| \mathbf{z}_k \right\rangle &= -\left(\frac{\gamma}{\pi} \right)^{\frac{M}{2}} \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \left(\sqrt{\frac{\pi}{\lambda + \gamma}} \exp\left[\frac{\gamma^2 \boldsymbol{\rho}_{jk}^2}{\lambda + \gamma} - \gamma \boldsymbol{\rho}_{jk}^2 \right] \right)^M \\ &= -\left(\frac{\gamma}{\lambda + \gamma} \right)^{\frac{M}{2}} \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \exp\left[-\frac{\gamma \lambda}{\lambda + \gamma} \boldsymbol{\rho}_{jk}^2 \right], \end{aligned} \tag{4.3.11}$$

and if $\eta = \gamma\lambda/(\lambda+\gamma)$, this becomes

$$\left\langle \mathbf{z}_j \right| - e^{-\lambda \mathbf{x}^2} \left| \mathbf{z}_k \right\rangle = - \left(\frac{\eta}{\lambda}\right)^{\frac{M}{2}} \exp\left[-\eta \boldsymbol{\rho}_{jk}^2\right] \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle, \tag{4.3.12}$$

which in practise would be applied as

$$\left\langle \mathbf{z}_j \right| - e^{-\lambda \mathbf{x}^2} \left| \mathbf{z}_k \right\rangle = (-1)^M \prod_{m=1}^{M} \left(\frac{\eta}{\lambda}\right)^{\frac{1}{2}} \exp\left[-\eta \rho_{jk}^{2(m)}\right] \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle. \tag{4.3.13}$$

The laser field can be converted in a similar way to the atomic potential. The laser field interaction potential for a linearly polarised field will have the form

$$V_{\mathcal{E}}(x,t) = x\mathcal{E}(t), \tag{4.3.14}$$

and because the field itself only acts in one direction it need only be integrated along that degree of freedom. As such, the equation for the laser field in z-notation can be found by

$$\left\langle z_j \right| x\mathcal{E}(t) \left| z_k \right\rangle = - \left(\frac{\gamma}{\pi}\right)^{\frac{1}{2}} \mathcal{E}(t) \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \int_{-\infty}^{\infty} x \exp\left[-\gamma \left(x - \rho_{jk}\right)^2\right] dx \tag{4.3.15}$$

in the direction of the laser field. This is true for simulations in one or in multiple dimensions, although if a circularly polarised field is used this does not apply. By using the well known identity that

$$\int_{-\infty}^{\infty} x \exp\left[-ax^2 + bx + c\right] dx = \sqrt{\frac{\pi}{a}} \frac{b}{2a} \exp\left[\frac{b^2}{4a} + c\right], \tag{4.3.16}$$

it can be found that

$$\begin{aligned} \left\langle z_j \right| x\mathcal{E}(t) \left| z_k \right\rangle &= - \left(\frac{\gamma}{\pi}\right)^{\frac{1}{2}} \mathcal{E}(t) \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle \left[\sqrt{\frac{\pi}{\gamma}} \frac{2\gamma\rho_{jk}}{2\gamma} \exp\left(\frac{4\gamma^2\rho_{jk}^2}{4\gamma} - \gamma\rho_{jk}^2\right)\right] \\ &= \rho_{jk}^{(x)} \mathcal{E}(t) \left\langle \mathbf{z}_j \right| \left. \mathbf{z}_k \right\rangle, \end{aligned} \tag{4.3.17}$$

where $\rho_{jk}^{(x)}$ is the component of $\boldsymbol{\rho}_{jk}$ that lies along the x-axis, which here is taken to be the direction of the laser field.

Using these terms, an ordered Hamiltonian for this system in one dimension can be constructed in the coherent state formalism using the z-notation such that

$$\begin{aligned} H_{ord}\left(z_j^*, z_k\right) = &- \frac{\gamma}{4}\left(z_j^{*2} + z_k^2 - 2z_j^* z_k - 1\right) \\ &- \sqrt{\frac{\eta}{\lambda}} \exp\left[-\eta\rho_{jk}^2\right] + \rho_{jk}\mathcal{E}(t). \end{aligned} \tag{4.3.18}$$

This reordering results in changes to the effective potential, giving a shallower potential than the form in coordinate representation such as that used in [222]. The HHG spectrum of this system is calculated from the Fourier transform of the time-dependent dipole moment $d(t)$ which can be given as the expectation value of the dipole operator $\hat{d}$

$$d(t) = \left\langle \Psi(t) \middle| \hat{d} \middle| \Psi(t) \right\rangle . \tag{4.3.19}$$

This is easily expanded out in terms of the elements of the dipole momentum matrix d_{jk} such that

$$\begin{aligned} d(t) &= \sum_{j,k} D_j^* D_k \, \exp\left[i(S_k - S_j)\right] \left\langle \mathbf{z}_j \middle| \hat{d} \middle| \mathbf{z}_k \right\rangle \\ &= \sum_{j,k} D_j^* D_k \, \exp\left[i(S_k - S_j)\right] \left\langle \mathbf{z}_j \middle| \, \mathbf{z}_k \right\rangle d_{jk}(t). \end{aligned} \tag{4.3.20}$$

The dipole moment can be expressed in length, velocity or acceleration forms [223, 224]. In the length form, the dipole moment is very large in regions away from the core and hence the dynamics in these regions is over-emphasised. In the velocity form, no spatial region is favoured over another. In the acceleration form however, the region close to the core is probed, which is the region in which HHG occurs according to the three-step model. Using the Heisenberg equations this acceleration form can be computed, and for the Gaussian potential this gives

$$\left\langle \mathbf{z}_j \middle| \ddot{d} \middle| \mathbf{z}_k \right\rangle = \left\langle \mathbf{z}_j \middle| -\boldsymbol{\nabla} \vec{V}(x) \middle| \mathbf{z}_k \right\rangle = \left\langle \mathbf{z}_j \middle| -2\lambda \mathbf{x} e^{-\lambda \mathbf{x}^2} \middle| \mathbf{z}_k \right\rangle . \tag{4.3.21}$$

By using equation 4.3.7 this becomes

$$\left\langle z_j \middle| \ddot{d} \middle| z_k \right\rangle_z = -2\lambda \left(\frac{\gamma}{\pi}\right)^{\frac{1}{2}} \left\langle z_j \middle| \, z_k \right\rangle \int_{-\infty}^{\infty} x \exp\left[-\left(\lambda + \gamma\right) x^2 + 2\gamma x \rho_{jk} - \gamma \rho_{jk}^2\right] dx \tag{4.3.22}$$

in the direction of the laser field and

$$\left\langle \mathbf{z}_j \middle| \ddot{d}(t) \middle| \mathbf{z}_k \right\rangle = -\left(\frac{\gamma}{\pi}\right)^{\frac{M-1}{2}} \left\langle \mathbf{z}_j \middle| \, \mathbf{z}_k \right\rangle \int_{-\infty}^{\infty} \exp\left[-\left(\lambda + \gamma\right) \mathbf{x}^2 + 2\gamma \mathbf{x} \boldsymbol{\rho}_{jk} - \gamma \boldsymbol{\rho}_{jk}^2\right] d^{(M-1)}\mathbf{x} \tag{4.3.23}$$

in all other directions. It should be noted that here as earlier, the vector $\mathbf{x}$ denotes a spatial vector with elements along all axes, and should not be confused with x used in equation 4.3.22, which denotes the x-axis along which the laser field travels. By applying equation (4.3.16) in the direction of the laser field and equation (4.3.10) in all

other directions, these then combine to give

$$\begin{aligned}\left\langle \mathbf{z}_j \middle| \ddot{d} \middle| \mathbf{z}_k \right\rangle = -2\lambda \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \langle \mathbf{z}_j | \, \mathbf{z}_k \rangle \left[\sqrt{\frac{\pi}{\lambda+\gamma}} \frac{\gamma}{\lambda+\gamma} \rho_{jk}^{(x)} \exp\left(-\frac{\gamma\lambda}{\lambda+\gamma}\rho_{jk}^2\right)\right] \\ \times \left[\left(\frac{\pi}{\lambda+\gamma}\right)^{\frac{M-1}{2}} \exp\left(-\frac{\gamma\lambda}{\lambda+\gamma}\sum_{m=2}^{M}\rho_{jk}^{(m)2}\right)\right] \\ = -2\lambda \left(\frac{\gamma}{\lambda+\gamma}\right)^{\frac{M+2}{2}} \rho_{jk}^{(x)} \exp\left(-\frac{\gamma\lambda}{\lambda+\gamma}\boldsymbol{\rho}_{jk}^2\right).\end{aligned} \tag{4.3.24}$$

By introducing the term η as with the previous equations, this becomes

$$d_{jk} = -2\sqrt{\frac{\eta^{M+2}}{\lambda^M}} \rho_{jk}^{(x)} \exp\left(-\eta\boldsymbol{\rho}_{jk}^2\right). \tag{4.3.25}$$

It should be noted that the CCS method is not without its limitations. In phenomena for which coherence is important, there exists a major challenge to overcome, namely that as the wavefunction propagates in phase space, the interference may not be accurately represented by a small CCS basis. This happens due to the fact that at longer times trajectories can misguide the basis, making propagation increasingly inaccurate. This is not as much an issue when modelling electron momentum distributions from NSDI for example, as often quantum interference is not paramount, insomuch as classical models are very successful in this case. For HHG however, quantum interference is essential. This is due to the fact that HHG is a coherent phenomenon, which needs at least a few cycles to build up. Hence, the phase of the wave function must be obtained and the coherent sum over the CCS trajectories must be performed. Furthermore, while being close to the core limits the number of trajectories that must be taken, as discussed in this work, this too brings difficulties as it is in this region that the interplay between the laser field and the binding potential is most relevant. This interplay leads to an irregular behaviour in phase space [176, 225, 226], which would not be a problem for trajectories far away from the core.

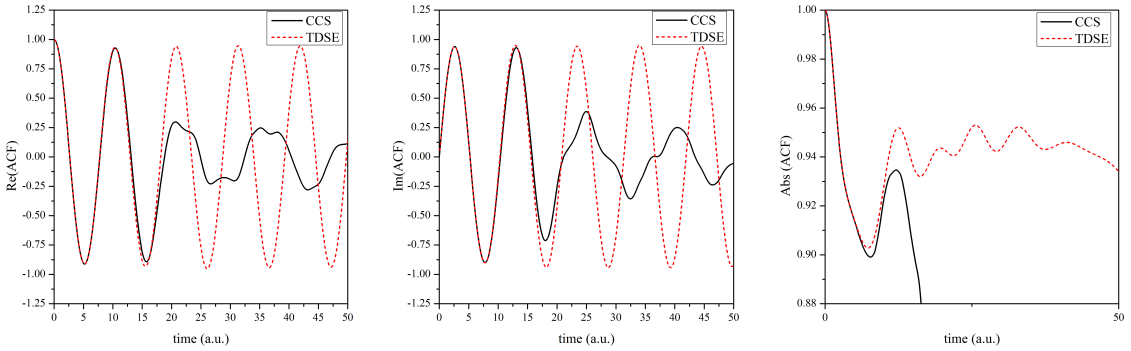
4.4 Preliminary Tests and Necessary Modifications

To carry out a preliminary set of tests to ensure that the CCS method can be applied to the generation of HHG spectra, the program discussed in Chap. 3 and Appendix A was expanded to allow propagation using the CCS equations and the Hamiltonian given in eq. 4.3.18. Further expansions were made to the way in which the initial basis set is constructed. It has been usual practice in previous applications of the CCS method to construct the basis set for a CCS wavefunction from a compressed random

swarm which follows a Gaussian distribution [67]. In the case considered here this initial distribution is unsuitable for modelling the wavefunction as a compressed basis is too localised to sufficiently describe the wavefunction fully at long times. To remedy this, the wavefunction was described as a basis set corresponding to a grid in phase space with a regular spacing between grid points. This allows the initial wavefunction to be described over a much larger area in phase space. As we consider a case where the electron starts in the ionic core, the grid was symmetrically ordered around the initial basis function z_0 which was located in phase space at $(q, p) = (0, 0)$. For these preliminary tests we consider a simple linearly polarised cosine field with the form $E(t) = x\mathcal{E}_0 \cos(\omega_0 t)$ with intensity $\mathcal{E}_0 = 0.1$ and frequency $\omega_0 = 0.05$. Calculating using classical mechanics, this gives a maximum electron excursion length of $\mathcal{E}_0/\omega_0^2 = 40$ *a.u.* and a maximum momentum of $(2E_{k_{max}})^{1/2} = \left(3.17\frac{\mathcal{E}_0^2}{2\omega_0^2}\right)^{1/2} = 2.5$ *a.u.* It was found empirically that an grid of 80×20 points was sufficient to model the initial wavefunction with an adequately high norm, with an additional point located at $\mathbf{z}_0$ which serves to provide further stability. The size of the grid on the q-axis must of course account for the maximum displacement experienced by the wavefunction so as to avoid significant loss of amplitude from the trajectories near the edge of the grid which would greatly affect the norm. In addition to the high harmonic spectrum, we consider the autocorrelation function (ACF), given by

$$ACF = \sum_k \langle z_0 |\, z_k \rangle\, D_k e^{iS_k}. \tag{4.4.1}$$

Unlike some other observables, the ACF is sensitive to changes over the entire wavefunction. It can be seen from figure 4.5 that even with a large grid the wavefunction decays after a short time and the autocorrelation function no longer matches the benchmark, which was calculated using the time dependent Schrödinger equation.



(A) Real part of the ACF (B) Imaginary part of the ACF (C) Absolute of the ACF

FIGURE 4.5: Real and imaginary parts of the autocorrelation function, together with the absolute value of the autocorrelation function, compared against those from TDSE calculations, for unmodified propagation

This has a detrimental effect upon the harmonic spectrum, as when there is a very small overlap between the wavefunction and the initial basis function located at the core, recombination effects cannot take place. As can be seen in figure 4.6 the characteristic features of the high harmonic spectrum are absent, with little evidence of the initial decrease, a steady loss of amplitude where the plateau would be expected and no sharp cutoff.

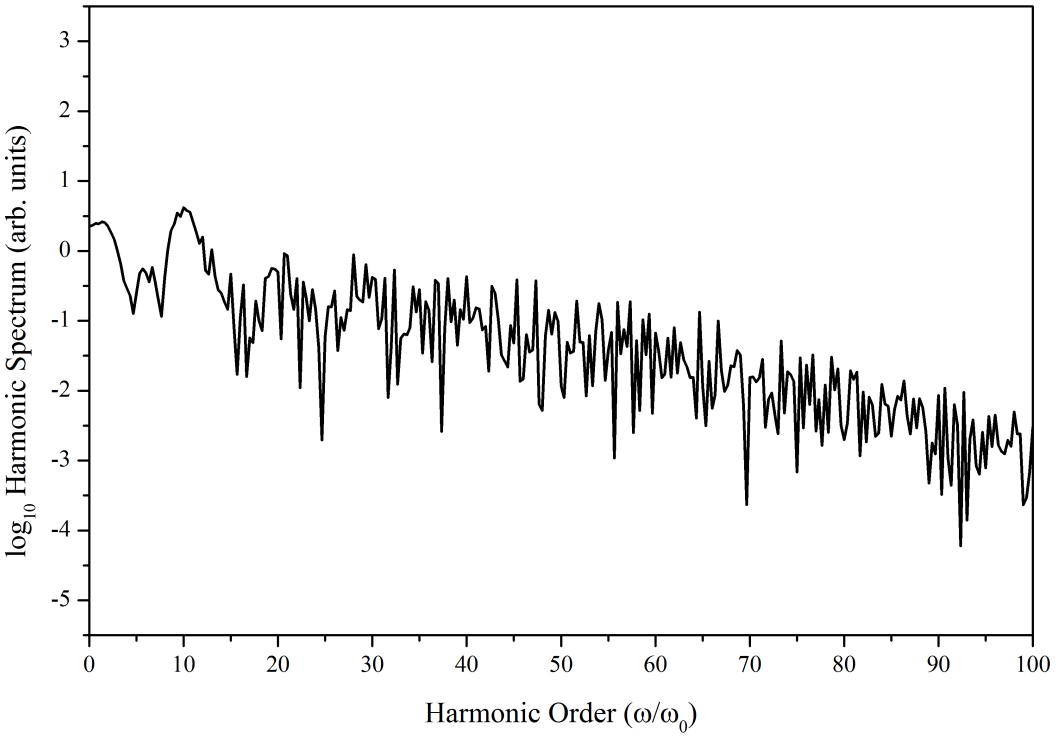


FIGURE 4.6: High harmonic spectrum with unmodified propagation

This behaviour can be understood by considering the trajectories along which the coherent states move. Figure 4.7 shows two of these trajectories from near the centre of the initial grid. It can be seen that only one trajectory stays in the vicinity of the core for any length of time; indeed of all the trajectories it is only this one trajectory and the one originating at $|z_0\rangle$ that does. Eventually even this trajectory travels far from the core. For trajectories with higher initial momentum this is more pronounced, and after a short time the wavefunction as a whole is guided far away from the core.

A second problem encountered is that a regular grid requires more trajectories to describe the wavefunction than a swarm of basis functions, and the larger the grid the higher the computational requirement, meaning that past a certain grid size the computational requirements are too large for the system to be modelled effectively.

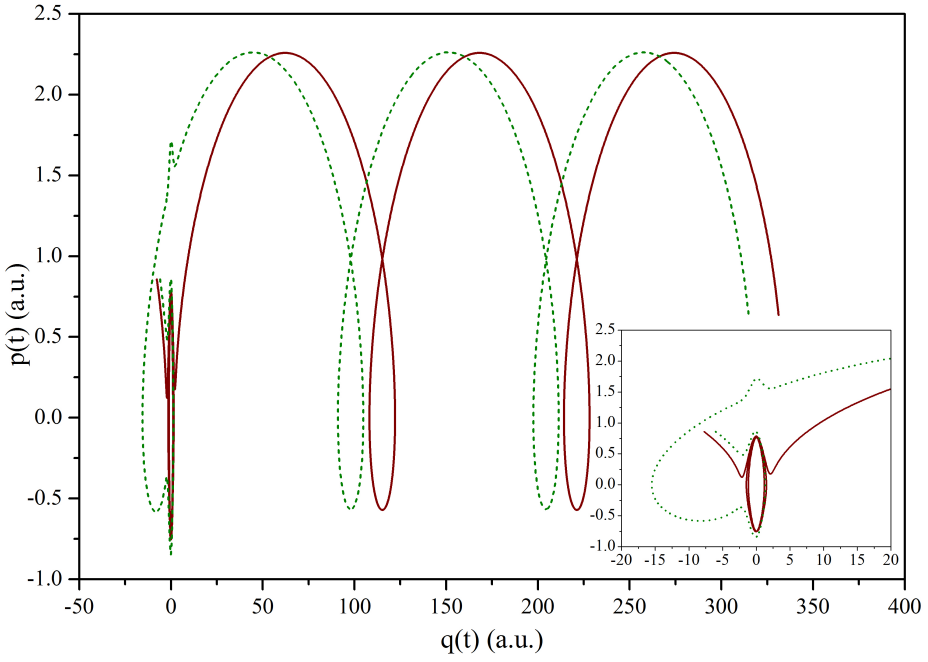


FIGURE 4.7: The path of the trajectories initially located close to the centre of the wavefunction as they propagate through phase space when not reprojected. The smaller frame shows the path the trajectories take in the region close to the origin. As a result of this after a short time, approximately 20 a.u., the overlap between the wavefunction trajectories and the initial grid points becomes too small for the CCS dynamics to be accurately calculated and the coherent states lose coupling.

A solution to these problems can be found by using an adaptive reprojection technique. This technique serves to keep the maximum size of the grid low, needing to only describe a large enough area in phase space to account for the movement of the wavefunction over a small time τ. A further reduction in overall computational cost of over 50% can be achieved by truncating those trajectories which do not give a large contribution to the wavefunction. Similar techniques have been used in the past to counter time restrictions in simulations using the Herman-Kluk method [61, 227], the hybrid Heller "Frozen Gaussian" method [228, 229], or in the phase space approach [230] with success. This technique consists of three stages:

(a) The overlap between the wavefunction $|\Psi(t)\rangle$ made up of N basis functions $|\psi_k(t)\rangle$ and the initial regular grid made up of N_0 grid points z_j^{grid}, separated by distances $dq = \Delta\sqrt{\frac{2}{\gamma}}$ and $dp = \Delta\sqrt{2\gamma}$ (where Δ is the grid spacing parameter), is calculated to find the quantity

$$C'_j = \sum_k^N \left\langle z_j^{grid} \middle| z_k(t) \right\rangle D_k(t) \mathrm{e}^{iS_k(t)}, \tag{4.4.2}$$

then values of C'_j that do not satisfy the condition

$$\left|C'_j\right| \geq \zeta \tag{4.4.3}$$

are discarded, leaving N' of the initial N_0 values. In equation (4.4.3), ζ is the basis threshold parameter. As equation (4.4.2) is calculated at each reprojection using the entire initial grid, grid points that were previously discarded at a prior reprojection using the condition given in equation (4.4.3) need not be discarded for all following reprojections. This ensures that the area of importance in phase space remains properly described as the wavefunction propagates.

(b) The wavefunction is reprojected upon the initial regular grid, not including those grid points which correspond to values of C'_j which have been discarded. This is done by using the form of the identity operator in equation (2.3.22) such that

$$\begin{aligned} |\Psi(t)\rangle &= \sum_{i,j}^{N'} \sum_{k}^{N} \left|z_i^{grid'}\right\rangle \mathbf{\Omega}_{ij}^{-1} \left\langle z_j^{grid'}\right| z_k(t)\Big\rangle D_k(t) \mathrm{e}^{iS_k(t)} \\ &= \sum_{i}^{N'} D'_i \left|z_i^{grid'}\right\rangle . \end{aligned} \tag{4.4.4}$$

As such the set of trajectories describing the wavefunction after reprojection is $\left|z_i^{grid'}\right\rangle$ and the set of amplitudes D'_i can be calculated from the set of linear equations

$$C'_j = \mathbf{\Omega}_{ij}^{grid'} D'_i \tag{4.4.5}$$

with the action set back to $S'_i = 0 \; \forall \; i$.

(c) The wavefunction is propagated for an amount of time τ using the equations (2.3.9), (2.3.17) and (2.3.23). The selected grid points move and exchange amplitudes, then reprojection is started again for the wavefunction $|\Psi(t+\tau)\rangle$.

It can be immediately seen that both the accuracy and computational cost of the method will increase as $\zeta \to 0$.

4.5 Results

4.5.1 High Harmonic Generation with Reprojection

To rigorously test the effectiveness of the reprojection process, a Hamiltonian should be used which is as simple as possible. As such, for these tests the linearly polarised

cosine field used in the preliminary tests from the previous section is retained. Since for larger values of the basis threshold parameter the wavefunction is likely to be projected on fewer trajectories, these values will require denser grids to adequately describe the wavefunction. The density of the grid is determined by the spacing parameter Δ and, by choosing the value of Δ correctly, the initial norm of the wavefunction is set very to close to unity. To evaluate the quality of results, the properties of a wavefunction calculated using the CCS method are compared against those calculated using the TDSE solver, considering various values for the basis threshold parameter.

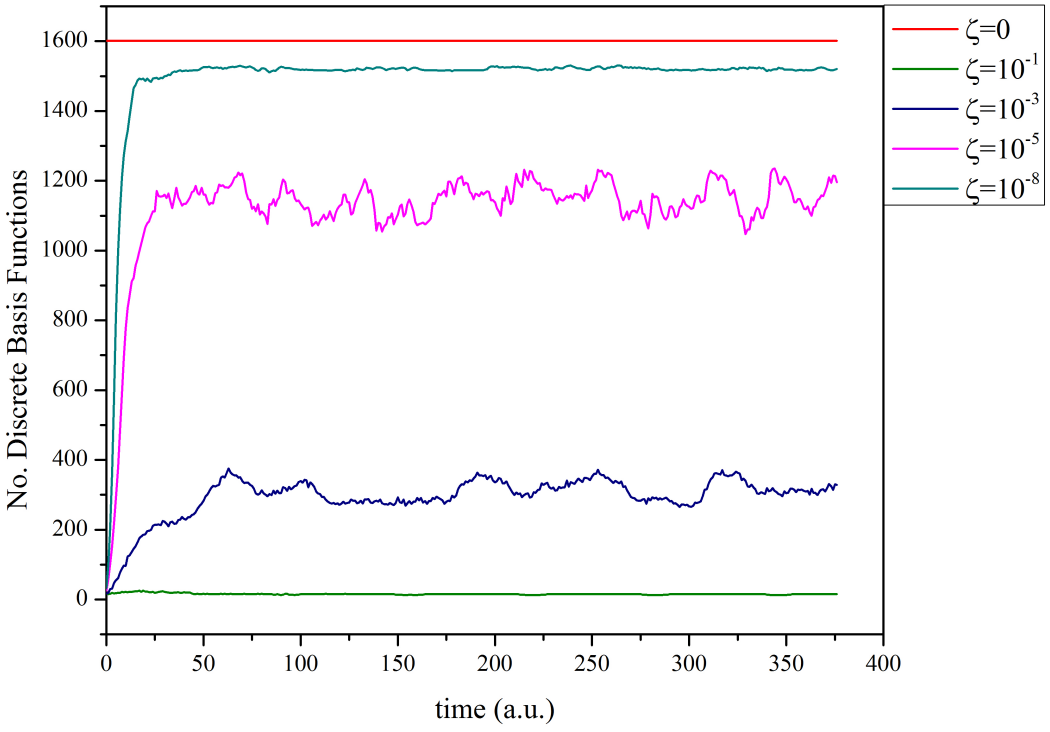


FIGURE 4.8: Changing number of basis functions for different values of the basis threshold parameter investigated in a logarithmic fashion. Selected values values from a range of $\zeta = 10^{-1}$ to $\zeta = 10^{-10}$ are given, namely $\zeta = 10^{-1}$ (large dashed), $\zeta = 10^{-3}$ (small dashed), $\zeta = 10^{-5}$ (dotted) and $\zeta = 10^{-8}$ (dot-dashed), as well as the case where $\zeta = 0$ (solid) and so the number of basis vectors stays constant.

For $\zeta \neq 0$, as the system is propagated in time the wavefunction spreads out and the number of basis vectors above the cutoff increases until it reaches a plateau value around which it stays for the duration of propagation with small fluctuations where trajectories are discarded or reintroduced as the wavefunction moves in phase space. This can be seen in figure 4.8. The value at which the number of basis vectors plateaus determines how computationally expensive the simulation is, and so it is desirable to use the largest possible value of the basis threshold parameter which allows for accurate simulation of the system, referred to hereafter as the maximum effective cutoff, ζ_{MEC}. The ζ_{MEC}

value can be determined based on the agreement of the autocorrelation function (ACF) with that calculated using the TDSE solver.

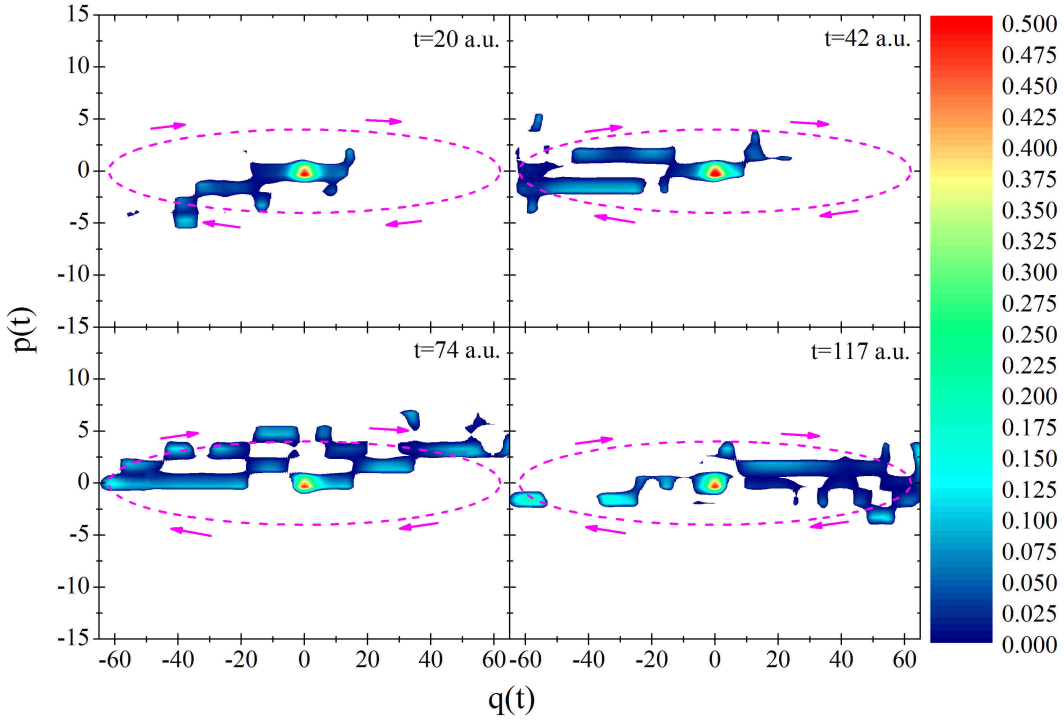
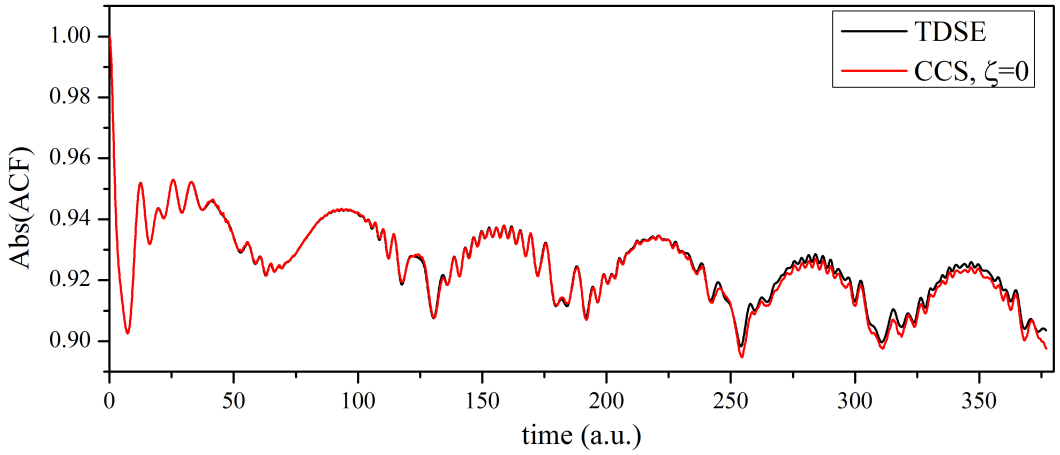


FIGURE 4.9: Motion of the wavefunction over time on a grid of 80×20 coherent states with a separation of $\Delta = 1.59$ *a.u.* and with $\zeta = 10^{-3}$. Each pane illustrates the entire grid, with whitespace denoting areas where the coherent states at those grid points have been removed during reprojection. The panes show the motion of the important parts of the wavefunction after reprojection at $t = 20$ *a.u.*, 42 *a.u.*, 74 *a.u.*, and 117 *a.u.*, and clearly illustrates how the external field has an effect on the wavefunction, drawing a portion of it into an elliptical motion around the core of the wavefunction. The elliptical motion of this portion of the wavefunction is illustrated by the dashed line.

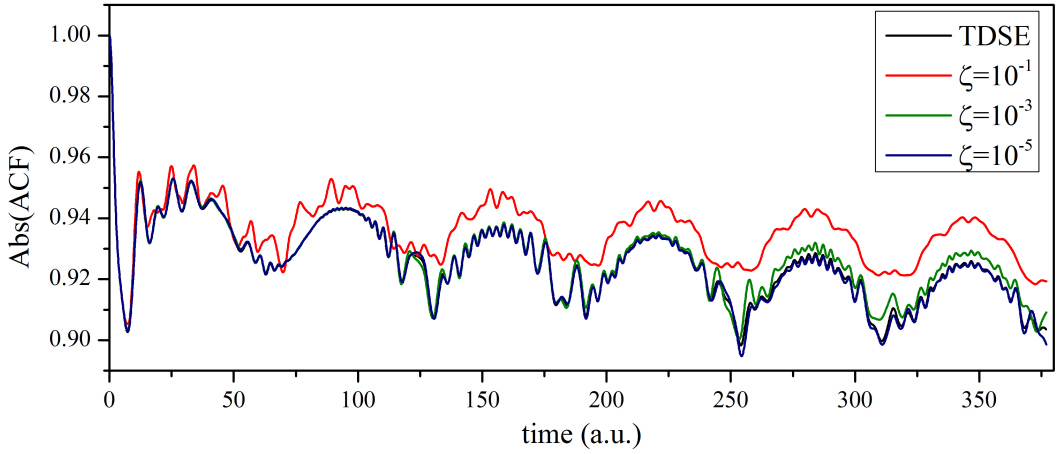
As the CCS method represents quantum dynamics in phase space, the motion of the wavefunction can easily be visualised. Figure 4.9 illustrates the dynamics of the wavefunction in a periodic field, with the colour scale showing the amplitude D_k at a particular point in phase space and white space indicating where trajectories have been removed during the reprojection process for four values of t. For an initial wave packet starting at the origin, the wavefunction starts very localised before spreading starting from the high and low momenta points. The wavefunction then spreads more in the region where the momenta of the grid points are close to zero and a substantial portion of the wavefunction begins motion in an elliptical orbit around the origin as it interacts with the external field. This motion illustrates how the wavefunction loses some amplitude at the edges of the grid when the external field is at maxima or minima, and as such shows that during propagation a part of the wavefunction irreversibly leaves the dynamically important part of the phase space. It can be seen that the path travelled

by the orbiting portion of the wavefunction differs from that of the classical motion because without reprojection the basis coherent states guided by Hamilton's equations (Eq. 2.3.9) move quickly away from the most important region, even when starting very close to the core as is shown in figure 4.7. If allowed to follow this path, the overlap between the basis functions would very quickly tend to zero and so the most important area of phase space could not be described adequately.

Furthermore, much of the wavefunction moves off in opposite directions, resulting in a loss of coupling between the basis functions and the ACF decaying in both the real and imaginary parts as is shown in figure 4.5. With reprojection however, the coupling



(A) Comparison between the absolute values of the autocorrelation functions produced by the TDSE solver and the CCS method with $\zeta = 0$.



(B) Comparison between the absolute values of the autocorrelation functions produced by the TDSE solver and the CCS method for the first few values of the basis threshold parameter.

FIGURE 4.10: Comparisons between absolute values of the autocorrelation functions produced by the TDSE solver and the CCS method for various values of the basis threshold parameter. Reprojection is carried out at every $\tau = 1a.u.$ and the intensity of the external cosine field is set at $\mathcal{E}_0 = 0.1$ with a frequency of $\omega_0 = 0.05$ for both comparisons. In all cases the wave packet initially starts at (q,p) = (0,0).

ζ	Δ	N_{final}	% speed-up
0	1.75	1601	-
1×10^{-10}	1.75	1547	11.27
1×10^{-8}	1.75	1521	16.04
1×10^{-6}	1.75	1427	34.67
1×10^{-5}	1.74	1196	61.56
1×10^{-4}	1.71	715	89.23
1×10^{-3}	1.59	328	98.44
1×10^{-2}	1.28	39	99.88
1×10^{-1}	1.00	15	99.99

TABLE 4.1: Table of the parameters for the different wavefunctions. The grid parameter Δ shows how the density of the grid increases as the basis threshold parameter increases, and the link between the computational expense and the number of basis functions can be clearly seen.

is maintained and although a small amount of information is lost at the edges of the grid the wavefunction still remains valid, as is shown in figure 4.10a, which shows the absolute value of the ACF for both the TDSE solver and the CCS method with $\zeta = 0$. As the two autocorrelation functions are in almost complete agreement this confirms the efficacy of the reprojection procedure with the entire initial basis set used throughout propagation and no basis functions discarded. For other values of the basis threshold parameter the agreement between the plots varies.

For the higher values of ζ there are noticeable discrepancies (figure 4.10b), however for values of $\zeta \leq 10^{-3}$, the plots converge towards the TDSE solver calculation result. Indeed, the ACF for $\zeta = 10^{-5}$ is virtually indistinguishable from the TDSE result, and it can be seen from this and from table 4.1 that using $\zeta_{MEC} = 10^{-5}$ is sufficient for accurate calculation of the ACF of the system while also reducing the computational expense of the simulation by over 60%. Even a very cheap calculation with the threshold parameter $\zeta = 10^{-3}$, which reduces computational cost by 98.4%, produces a result which is in good agreement with the benchmark for most of the propagation time.

As mentioned earlier, the HHG spectrum is found by taking the Fourier transform of the dipole moment $d(t)$, which is calculated here in acceleration form so as to better probe the core. The plot of the dipole acceleration is given in figure 4.11, comparing the TDSE and CCS dipole accelerations, with the CCS plot calculated with $\zeta = \zeta_{MEC}$. The dipole acceleration approximately follows the laser field, which is shown inset in figure 4.11 for comparison, and exhibits a series of high-frequency oscillations. As mentioned earlier,

these oscillations have been associated with the interference between electron trajectories which, together with spatial localisation, are responsible for the HHG plateau. The oscillations have also been studied in a different context, namely the adiabatic approximation [231, 232] and Bohmian trajectories [195, 233].

The extent to which the plots agree is high, being almost identical until the very end of the third oscillation of the cosine field. This agreement persists in the HHG spectrum. Figures 4.12 and 4.13 show the HHG spectrum obtained from the TDSE solver compared against that from the CCS method for various values of the basis threshold parameter. Figure 4.12(a) shows the HHG spectrum for $\zeta = 10^{-1}$, and as can be seen the comparison with the TDSE solver generated spectrum is not very good, showing differences in the overall intensity in the region between around $25\omega_0$ and the cutoff, disagreements with the positions of many of the peaks in the spectrum and fluctuations after the cutoff point which could confuse the position of the cutoff. Figure 4.12(b) is an improvement on this, using $\zeta = 10^{-3}$. This gives a much better agreement for the overall intensities over the spectrum, and much better agreement in the peak positions. It does, however, show fluctuations after the cutoff which are not present in the TDSE solver spectrum. Figure 4.13(a) gives the spectrum for $\zeta = \zeta_{MEC}$, and it can be seen that there are few

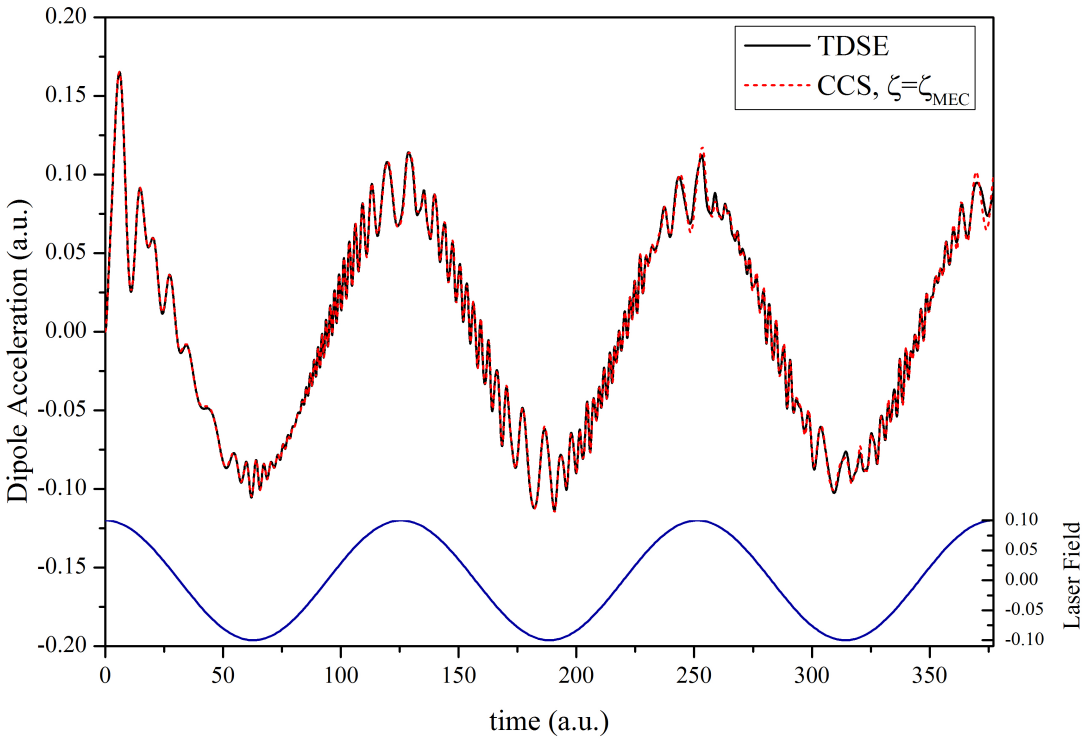


FIGURE 4.11: Comparison between dipole acceleration graphs for the $\zeta = \zeta_{MEC}$ CCS calculated wavefunction and the TDSE calculated plot. Inset is also the cosine laser field (right axis).

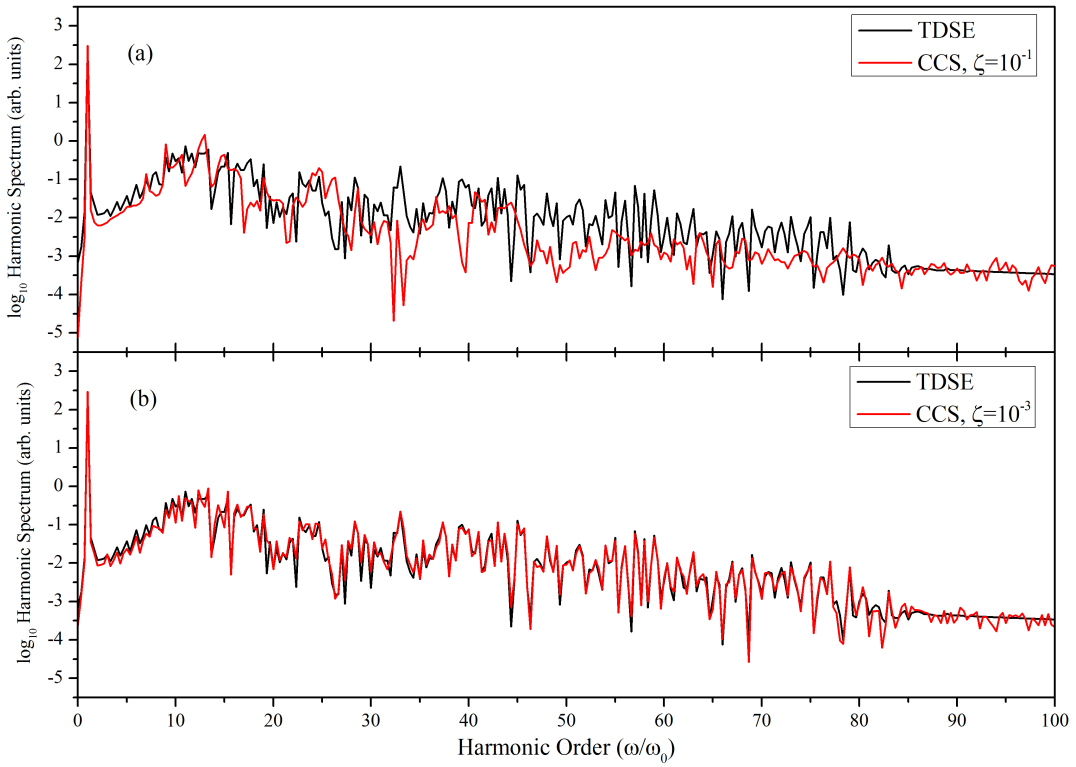


FIGURE 4.12: Comparisons of HHG spectra generated by the TDSE solver and those generated by the CCS method, with $\zeta = 10^{-1}$ in panel (a), $\zeta = 10^{-3}$ in panel (b)

differences between this spectrum and that in figure 4.13(b) where $\zeta = 0$, indeed despite the smaller basis set the HHG spectrum when $\zeta = \zeta_{MEC}$ is virtually unchanged from the $\zeta = 0$ spectrum. In both panels of figure 4.13 the spectra generated by the CCS calculation show excellent agreement with that of the TDSE solver, exhibiting the cutoff and plateau in the correct regions and agreeing almost completely for the features of the spectrum, although some low amplitude fluctuations after the cutoff are still present due to numerical effects.

4.5.2 Use of Alternative Laser Fields

The success of the CCS method with reprojection at accurately simulating HHG on a level comparable to a TDSE solver is an important result. Two questions arise from these results however: firstly, can this method be applied to a Hamiltonian with a less simple laser interaction potential, and secondly, will this reprojection procedure still produce valid results at longer times. To answer these questions we consider an electron bound to the same inverted Gaussian potential, but acted on by an N-cycle sine-squared

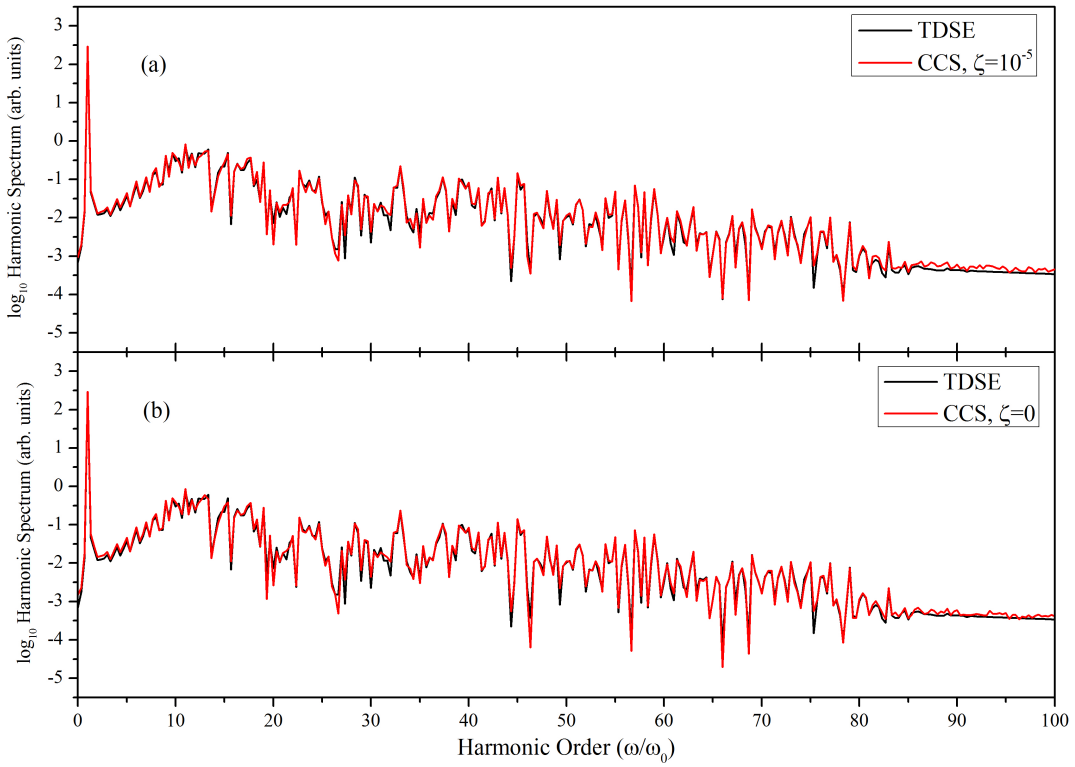


FIGURE 4.13: Comparisons of HHG spectra generated by the TDSE solver and those generated by the CCS method, with $\zeta = 10^{-5}$ in panel (a), $\zeta = 0$ in panel (b)

pulse, with the equation

$$\mathcal{E}(t) = \mathcal{E}_0 \sin^2\left(\frac{\omega_0 t}{2N}\right) \sin\left(\omega_0 t\right). \tag{4.5.1}$$

The amplitude and frequency of the laser field were set to be $\mathcal{E}_0 = 0.1$ *a.u.* and $\omega_0 = 0.057$ *a.u.* respectively, as is used by Song *et. al.* in ref [234]. These parameters allow the same size and density of grid to be used in the calculations as the excursion length for the electron becomes $\mathcal{E}_0/\omega_0^2 = 30.8$ *a.u.* and the maximum momentum becomes $(2E_{k_{max}})^{1/2} = \left(3.17\frac{\mathcal{E}_0^2}{2\omega_0^2}\right)^{1/2} = 2.2$ *a.u.* As the motion of the wavefunction is not appreciably faster the value of $\zeta = \zeta_{MEC} = 10^{-5}$ is also still used for the reprojection process. Three pulse lengths are considered, with $N = 2$, $N = 3$, and $N = 5$. The inclusion of the 5-cycle sine-squared pulse requires a total propagation time of $t = 551$ *a.u.*, a significant increase to the previous propagation time.

Figure 4.14 shows the dipole acceleration and harmonic spectrum generated when a 2-cycle sine-squared laser pulse is used. As is seen from the dipole acceleration, the higher frequency oscillations which are responsible for the main features of the harmonic spectrum appear in more significant amounts after the first cycle of the laser field. The smaller number of high frequency oscillations resulting from the short pulse length is

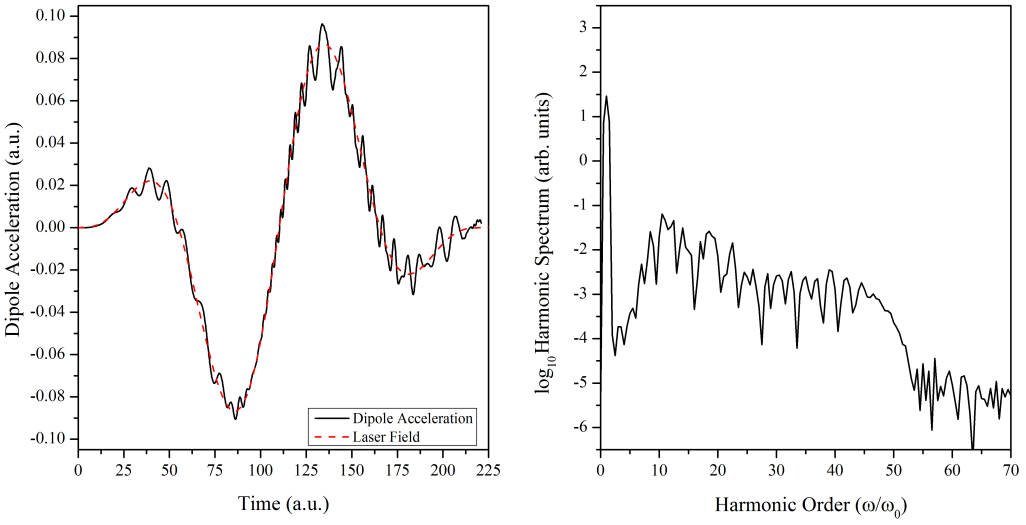


FIGURE 4.14: Dipole acceleration (left) and harmonic spectrum (right) for a 2-cycle sine-squared laser pulse with frequency $\omega_0 = 0.057$ *a.u.* and amplitude $\mathcal{E}_0 = 0.1$ *a.u.* Also shown on the graph for the dipole acceleration is the shape of the laser pulse.

echoed in the harmonic spectrum, which shows fewer peaks than is seen for the 3-cycle cosine field considered in the previous section. The main features of the high harmonic spectrum are all present, with the cutoff in the region of $I_p + 3.17U_p$ very easily recognisable and with a stable plateau preceding this cutoff, although it should be noted that while it may appear at first glance that the odd-harmonics rule is obeyed in this spectrum, closer examination shows that this is not the case. Since this simulation only considers a system with a single electron present however, this is to be expected.

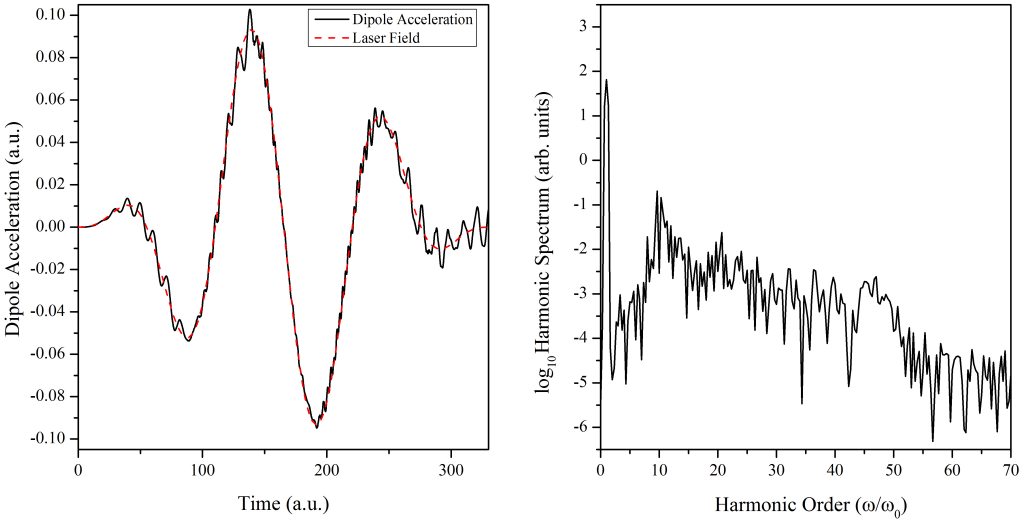


FIGURE 4.15: Dipole acceleration (left) and harmonic spectrum (right) for a 3-cycle sine-squared laser pulse with frequency $\omega_0 = 0.057$ *a.u.* and amplitude $\mathcal{E}_0 = 0.1$ *a.u.* Also shown on the graph for the dipole acceleration is the shape of the laser pulse.

The dipole acceleration and harmonic spectrum generated when a 3-cycle sine-squared laser pulse is used is shown in figure 4.15. As with the 2-cycle pulse, the amplitude of the high frequency oscillations in the dipole acceleration gets larger towards the end of the pulse. The harmonic spectrum shows more peaks than is seen for the 2-cycle pulse also, which is to be expected since the longer pulse length allows more time for the harmonic spectrum to build up. The main features of the harmonic spectrum are present, however due to the greater number of frequencies represented in the spectrum, the cutoff is not a smooth dropoff as it is for the 2-cycle pulse.

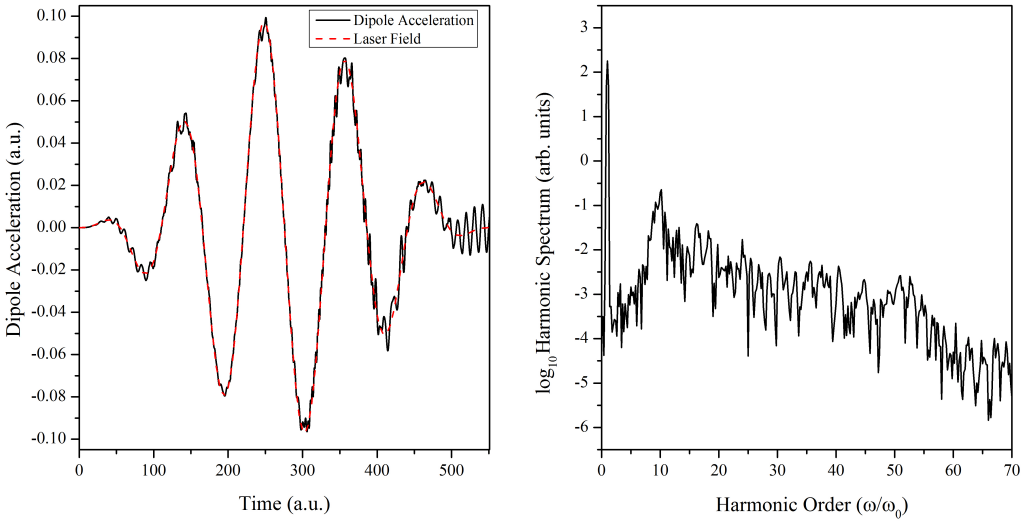


FIGURE 4.16: Dipole acceleration (left) and harmonic spectrum (right) for a 5-cycle sine-squared laser pulse with frequency $\omega_0 = 0.057$ *a.u.* and amplitude $\mathcal{E}_0 = 0.1$ *a.u.* Also shown on the graph for the dipole acceleration is the shape of the laser pulse.

The dipole acceleration and harmonic spectrum for the third pulse structure, a 5-cycle sine-squared pulse, are shown in figure 4.16. As with the dipole acceleration for the 3-cycle pulse, the amplitude of the high frequency oscillations increases towards the end of the pulse. The dipole acceleration follows the laser field as well as for the shorter pulses despite the longer propagation time. The main features of the harmonic spectrum are again present, with the plateau populated by more frequencies. The fact that these features are present indicates that coherence is maintained using the CCS method for longer simulations, thus answering one of the questions raised at the beginning of this section.

The steady increase in complexity of the spectra as the number of cycles increases echoes the results from ref. [165] shown in figure 4.2. A direct comparison of the spectra shows that as the spectrum builds up over more laser cycles, not only is more complexity seen in the plateau but the position of the cutoff moves closer and closer to the calculated value of $I_p + 3.17U_p = 51.57\omega_0$, with the cutoff for the 2-cycle pulse occurring well before

this value, around $46.5\omega_0$, the cutoff for the 3-cycle pulse occurring around $49\omega_0$ and the cutoff for the 5-cycle pulse agreeing almost completely with the calculated value. This can be seen clearly in figure 4.17.

4.6 Conclusions

The work presented here shows that when using a large grid of coherent states as a basis, trajectory decoupling can be prevented through reprojection of the wavefunction, thus removing the requirement of short time scales for CCS simulations while also improving the ability of the CCS method to accurately describe coherent phenomena. This is demonstrated through the ability of the CCS method to accurately simulate strong field phenomena such as the HHG spectrum of an electron in an intense laser field in one dimension for both steady fields and for laser pulses of various lengths. It has also been shown that computational cost can be reduced by adaptively reducing the basis set size which has allowed the method to account for wavefunctions which spread out to cover large areas in phase space without being prohibitively computationally expensive. This reprojection process applied to the CCS method has also shown itself to be capable of maintaining coherence for at least a 5-cycle laser pulse with no signs that longer pulses would be problematic. The reprojection of the wavefunction relies on a large regular grid which scales exponentially with the number of degrees of freedom of a system. Since only a small part of the grid is kept during the reprojection however, the cost of the computation can be greatly reduced, with very accurate results obtained with a $> 60\%$ speedup, and a good reproduction of the majority of the spectrum with a $> 98\%$ speedup.

Furthermore, for multi-dimensional systems we expect the reduction of the grid used here to play an even greater role, further reducing computational cost. It has been shown previously [69, 71, 235], that the CCS method can make use of Monte-Carlo sampling, ideally scaling quadratically with degrees of freedom. In the direction of the laser field, due to the motion in phase space of the wavefunction and the need for high accuracy in this dimension, this is not possible. A Monte-Carlo sampling, however, can be applied to the remaining two directions for consideration of a realistic three-dimensional system resulting in much less than exponential scaling. Preliminary testing has been carried out using this basis set structure, and it has been found that if using the inverted Gaussian potential, a smaller grid spacing is necessary to properly describe the relevant area of phase space than was needed for the 1D simulation, and as such the size of the grid must be increased to cover the same area, changing the grid dimensions from 80×20 with a spacing of $\Delta = 1.75$ in the direction of the laser

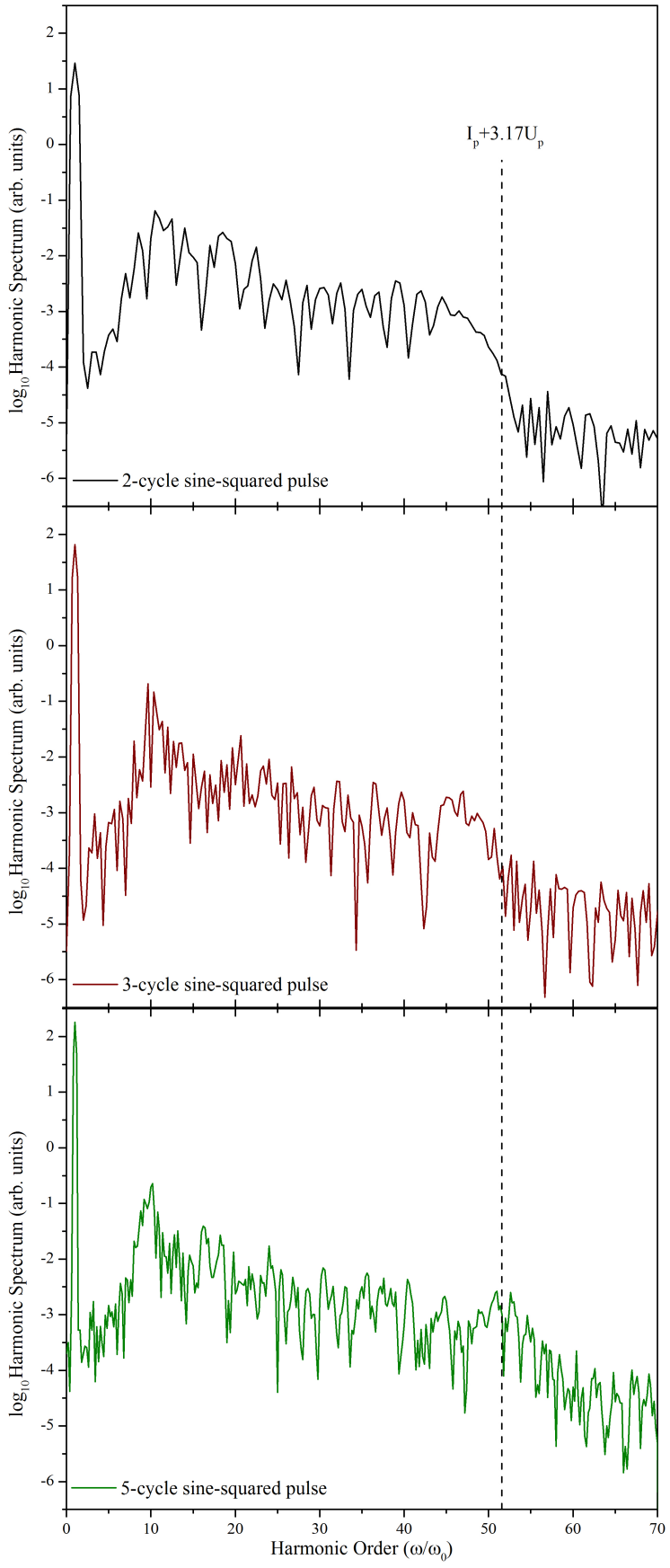


FIGURE 4.17: Comparison of the HHG spectra generated by 2-, 3-, and 5-cycle sine-squared pulses. Also marked on the graphs is the calculated location of the HHG cutoff at $I_p + 3.17U_p = 51.57\omega_0$

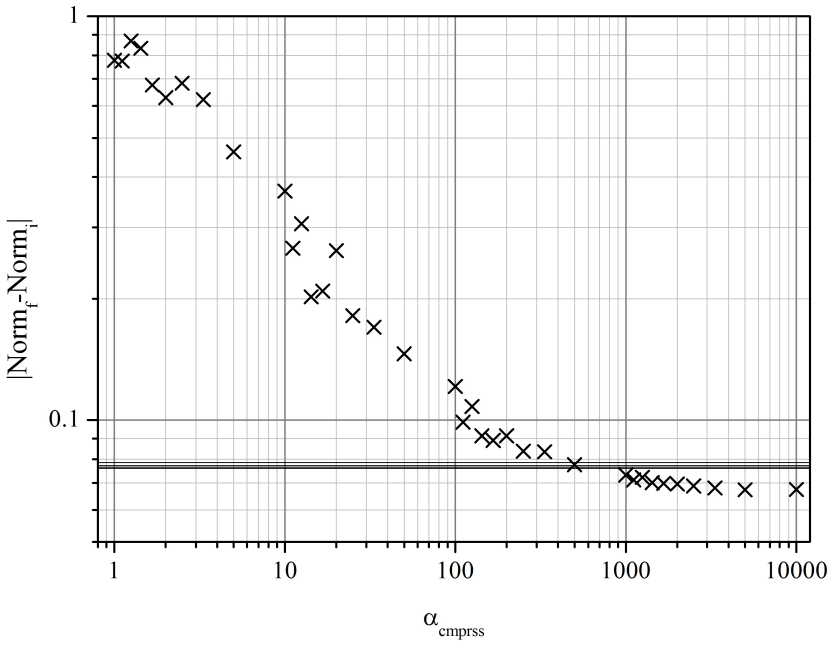


FIGURE 4.18: Calibration graph showing the dependence of the difference between initial and final norm on the compression parameter for a 3D high harmonic generation simulation with a regular grid in one dimension and random swarms in the other two.

field to 100×26 with a spacing of $\Delta = 1.40$. It was found also that the random swarm must have a high compression parameter, equating to a small width for the Gaussian envelope of the swarm distribution, in order to preserve the norm. Some calibration tests were carried out (see figure 4.18), which show that a compression parameter of ≥ 500 was necessary to preserve the norm to the same standard as the 1D simulations reported earlier (shown by horizontal lines). Such a high compression factor could cause problems in correctly describing the behaviour of the wavefunction in these dimensions. In some cases however, including HHG with an external elliptically polarized field, it is possible that these swarms would be ill-suited to the system and so regular grids would be necessary in more than one dimension, which would of course affect the scaling. In addition, the CCS method using the reprojection scheme can be used in the future to treat only the most important degrees of freedom in a system with the rest treated by a single Gaussian, in the spirit of the MCE method [23, 24, 236], which will be the subject of future work aiming at more challenging multi-electron systems.

Chapter 5

Conclusions and Outlook

The Coupled Coherent States (CCS) and Multiconfigurational Ehrenfest (MCE) methods have been shown in the past to be capable of simulating many quantum interactions. There remained the case though that if the basis set does not cover the correct area in phase space, the simulation falls into the semiclassical regime and becomes incapable of accurately simulating behaviour of the wavefunction in classically forbidden regions. While in many cases this can happen due to too large a spreading out of the basis set at longer timescales resulting in a loss of coupling between the basis functions, there are some systems where instead the basis set remains too compressed, causing the basis functions to follow paths which are too similar to each other resulting in a basis set which does not cover a large enough area in phase space. With the addition of extra basis set propagation methods and basis set samplings, it has been shown in this work that steps can be taken to account for the behaviour of such systems, and thus the range of applications to which the CCS methods can be applied is increased to systems where previously the motion of the wavefunction prohibited their use.

The work in Chapter 3 has shown that a discrepancy exists between the two formulations of the MCE method due to the separation of the intraconfigurational coupling and the interconfigurational coupling in the second formulation of this method. As a result of this separation, simulation of a high dimensional system can result in a lack of spreading of the basis set, returning results which do not properly describe the quantum mechanics of the system. With the introduction of swarms of trains during the basis set sampling process and basis function cloning during propagation, it is shown that for a system in which the basis set remains too closely compressed during propagation, such as the high dimensional spin boson model, successful simulation using the MCEv2 equations can be performed with a good level of agreement to the MCTDH benchmark calculations. This is a very important result in the context of wider research using the MCE method, as

simulations using these modifications have been carried out for the non-radiative decay of the first excited state of ethylene[1] and the total kinetic energy release spectrum for the photodisociation of pyrrole[2]. The results in Chapter 3 therefore support the methods used in these two publications. The importance of these results goes further though, as the original motivation for developing a second formulation of the MCE method was to allow use of *ab initio* "on-the-fly" direct dynamics, and with these modifications this aim is realised.

The work in Chapter 4 shows that with the introduction of adaptive reprojection during propagation, a system in which interconfigurational coupling is lost due to behaviour of the coherent state basis such as a single electron acted upon by a strong monochromatic linearly polarised laser field can be successfully simulated using the CCS equations giving good agreement to TDSE benchmark calculations. Using this technique, a large grid of coherent states can be reduced to only the area of interest, while the basis set is kept in the region of this area of interest. It has also been shown that this still holds even if the equations of motion tend to move basis functions far away from this area. This allows the CCS method to simulate quantum effects on larger timescales than was previously possible while also being capable of accurately describing coherent phenomena.

The development of these modifications to the CCS methods significantly widens the range of applications to which the methods can be applied. In light of the results presented in Chapter 3 further cases of the spin boson model can be investigated, including cases which use non-Ohmic spectral densities. Investigations into other cases of the Ohmic bath spin boson model would provide information on the limitations of this method. One expected limitation would be in the case of localisation due to the lack of population transfer between the states, removing the possibility for cloning. Work considering this and other cases of the Ohmic bath spin boson model are currently ongoing. The ability to treat super- and sub-Ohmic baths would be of great use also. As is mentioned in section 3.2 the range of applications of the spin boson model is large, including quantum computation, condensed matter physics, and chemical physics, and many of these applications require careful selection of an appropriate spectral density to describe the harmonic bath. The success of the AIMC method to simulate the spin boson model for an Ohmic bath indicates that application to these problems may be possible, however further work to verify this supposition would be necessary.

The results presented in Chapter 4 show the success of the CCS method in simulating the phenomenon of high harmonic generation in a one-dimensional pseudo potential acted on by either a steady field or a laser pulse, however by drawing on previous work this could be expanded to a three-dimensional Coulombic potential. It has been shown before [71, 75, 235] that upon converting the three dimensional Coulomb potential to coherent

states z-notation the Coulomb singularity is removed. As such not only is it possible to simulate high harmonic generation for a realistic atomic hydrogen potential using the CCS method, but there are benefits in doing so from a computational perspective. Work to implement this into the CCS codes is currently ongoing. More challenging cases, such as for molecular hydrogen or atomic helium, can also be considered by using the Fermionic CCS method [75], which would allow multiple electrons to be considered with the correct permutation symmetry by using a fermionic coherent state which for example would combine two electrons in a singlet state with $S = 0$ as

$$|\alpha\rangle = \frac{|\mathbf{z}_1\rangle |\mathbf{z}_2\rangle + |\mathbf{z}_2\rangle |\mathbf{z}_1\rangle}{\sqrt{2\left(1 + |\langle \mathbf{z}_1 | \, \mathbf{z}_2\rangle|^2\right)}},$$

as opposed to the standard CCS combination which would be a simple product state $|\mathbf{Z}\rangle = |\mathbf{z}_1\rangle |\mathbf{z}_2\rangle$. In addition to this the new two-layer CCS approach [236] could be a very useful tool in the simulation of high harmonic generation, as it would allow the wavefunction to be treated more rigorously in the direction of the laser field than in the other two directions through the use of a modified ansatz

$$|\Psi(t)\rangle = \sum_k D_k \left(\sum_j d_{j,k} e^{iS_{j,k}} \left| z_{j,k}^{(q)} \right\rangle \right) \left| \mathbf{z}_k^{(c)} \right\rangle$$

in which the direction of the laser field is treated by the more quantum $\left|z^{(q)}\right\rangle$ coherent states while the other degrees of freedom are treated by the more classical $\left|\mathbf{z}^{(c)}\right\rangle$ coherent states. This could be implemented allowing larger systems to be within the reach of the CCS methods. Further to this, other strong field phenomena such as Above Threshold Ionisation or Non-Sequential Double Ionisation could be investigated using the CCS method making it potentially a very useful tool in the area of strong field physics.

Further to extensions and expansions of the systems presented in this work, the confirmation of the ability of the MCEv2 equations to simulate high dimensional systems allows further work in the direction of *ab initio* "on-the-fly" direct dynamics, with simulations of interactions such as the photodissociation of hydrogen from various azoles and other ultrafast processes in small organic molecules. Work is already ongoing studying the dynamics of pyrazole, imidazole and 2-ethylpyrrole in conjuction with Prof. Vasilos Stavros, with further work planned considering methylpyrrole and other small molecules.

Appendix A

Algorithmic and Programming Details

A.1 Program Overview

A.1.1 Program Design

A program was written in order to undertake the investigations presented in the earlier chapters. This program was designed with the following requirements in mind:

- The program must be able to run successfully using different Hamiltonians, numbers of electronic states or equation sets without too much extra effort involved in changing between running conditions.
- The program must be able to carry out simulations to a high level of accuracy.
- The program must be efficient with little to no unnecessary or redundant code.
- The program must be either written in a parallelised format or be parallelisable.
- The program must be robust with good error checking and handling.

While points 2-5 are fairly self explanatory, the first point bears some explanation. Flexibility in Hamiltonian is, of course, a basic requirement for a generalised program, as is flexibility in the number of electronic states allowed. Flexibility in the equation set is required as there are many methods within the CCS family, from the original single surface CCS [31] and the MCE methods [23, 24], to more specialised versions such as Fermionic CCS[75, 235] and Cartesian CCS [237], to the recent *ab initio* MCE [26, 27]

and Multiple Cloning MCE methods [1, 2]. It is therefore desirable that the program be able to use many of these methods, although due to differences in the required program structure some of these methods may not be entirely compatible.

A modular design for the program is very important in light of the requirement for flexibility. By using a modular design, different systems can be modelled or different equation sets can be used without the need for any major revision of the code, and without creating multiple versions of the same program. It was decided that the program should have the following main sections:

- ***Main Program***, which determines program flow,
- ***Basis Set Generation Module***, which contains code to generate the initial basis set and initial values of the wavefunction parameters,
- ***Hamiltonian Module***, which contains code to calculate the norm and overlap and the general structure of the Hamiltonian,
- ***System Redirection Module***, containing case statements which choose the correct system specific module to read in system specific parameters and calculate the initial wavefunction, $|\mathbf{z}_0\rangle$, the elements of the Hamiltonian and those of the time derivative of the Hamiltonian, allowing easy expansion of the program to different applications,
- ***Time Derivative Module***, which contains code to solve the time propagation equations, chosen by a similar redirection case statement,
- ***Propagation Module***, which contains code to implement the time derivative equations from the previous module and take a single timestep,
- ***Input Module***, containing subroutines to read in parameters from the input files, and
- ***Output Module***, containing subroutines which produce data outputs from the simulation.

Using this structure it is easy to change the Hamiltonian or swap the MCEv1 equations for the MCEv2 equations by simply calling a different module from the redirection statement. In addition, expanding the program is easy as the modules were written in a general way so as to allow new modules to be added easily. A basic layout of the program is illustrated in figures A.1 and A.2. Some further explanation is required at various points on the flowchart. These points are labelled and itemised below:

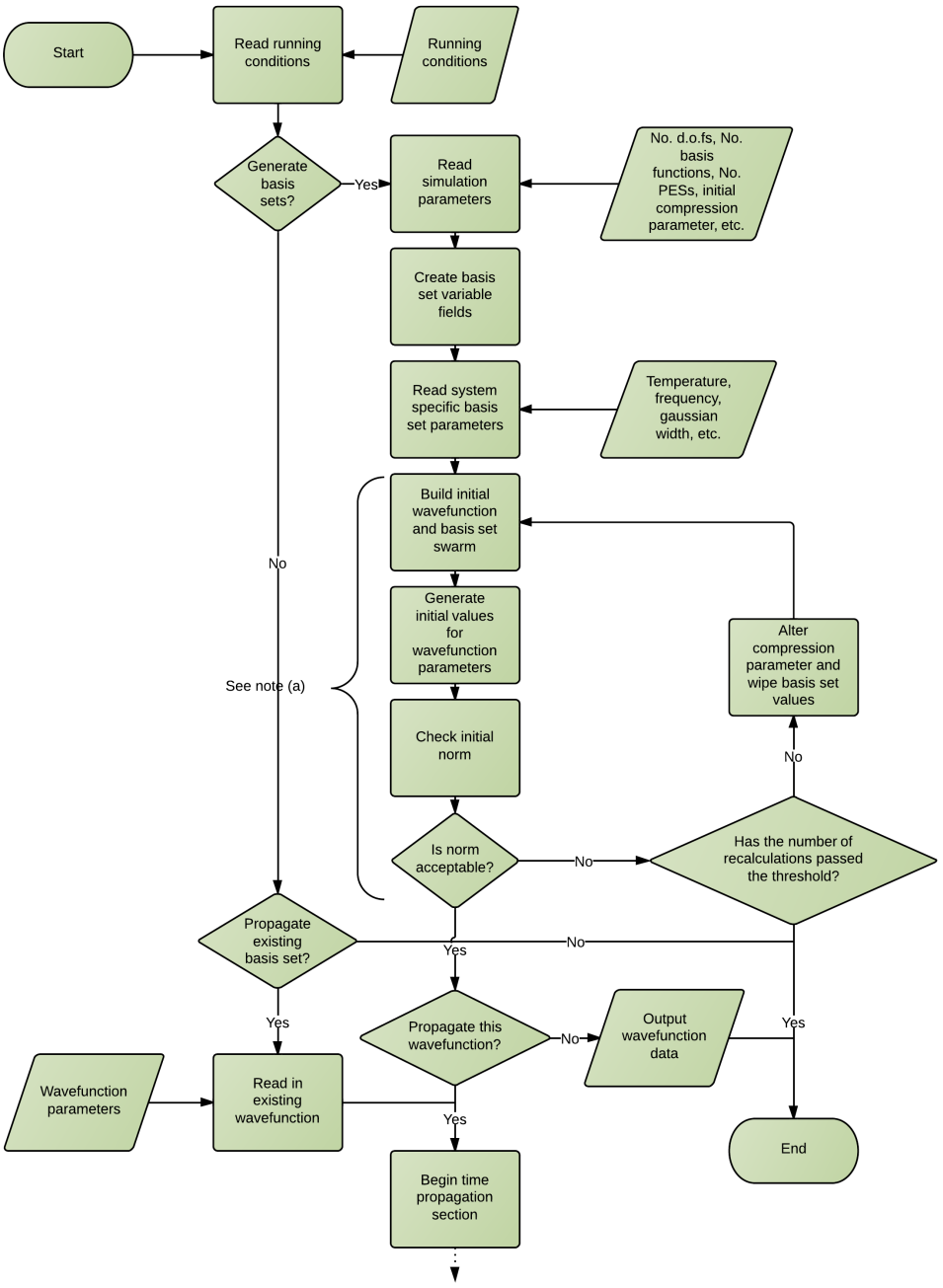


FIGURE A.1: Flow Chart showing the layout of the first half of the operation of the program. The program continues in figure A.2 starting from the "Begin time propagation" process box.

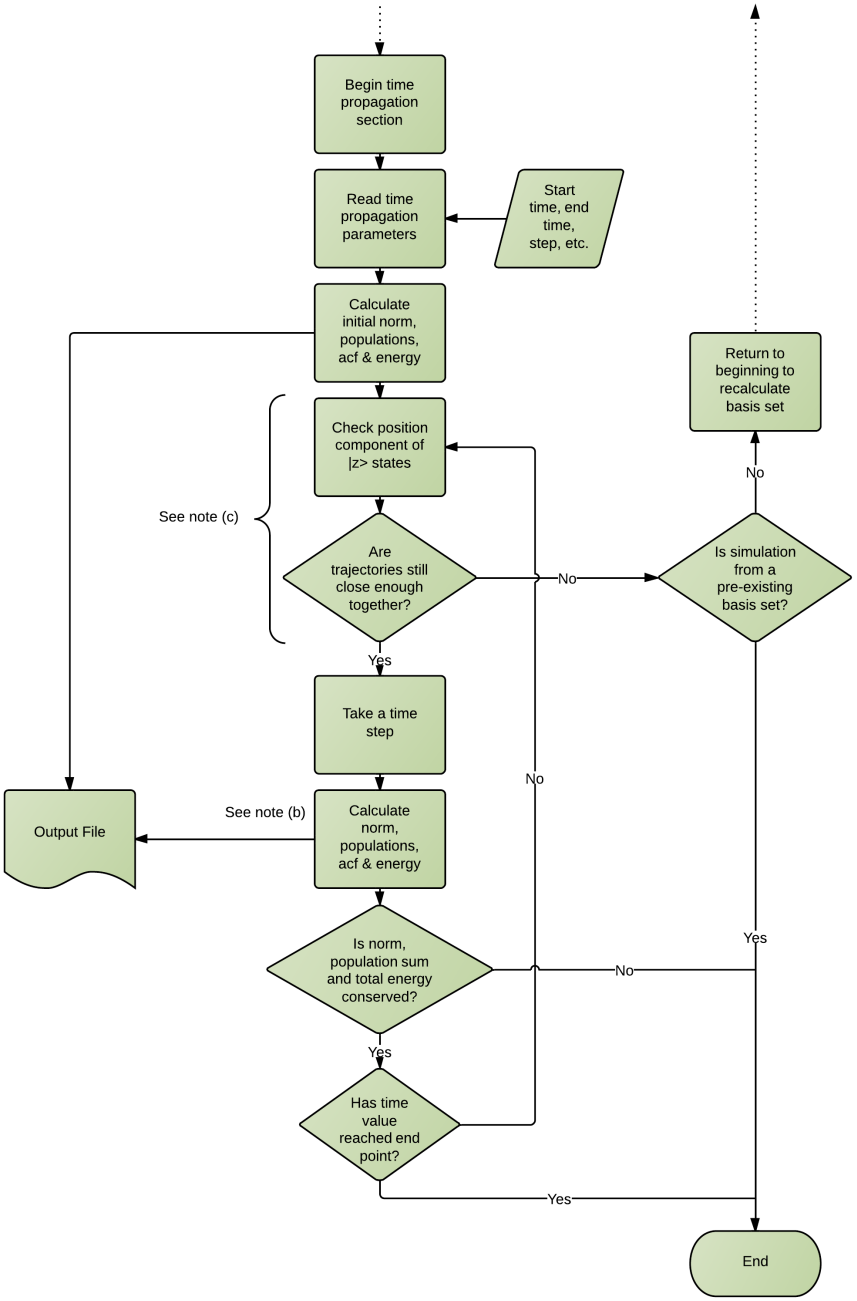


FIGURE A.2: Flowchart showing the layout of the second half of the operation of the program carrying on from figure A.1.

(a) We consider here the case where a random swarm is used for the initial basis set. As such for both the initial wavefunction $|\mathbf{z}_0\rangle$ and basis set $|\mathbf{z}_k(0)\rangle$, the initial values are selected from a set of Gaussian distributed complex random numbers. Due to the requirement that the coherent states must satisfy equation (2.3.5), the real part of these random numbers are multiplied by $\sqrt{\gamma/2}$ and the imaginary part by $\sqrt{1/2\gamma}$. The calculation of the basis set is optionally recursive, as can be seen from the flowchart, recalculating the basis set with adjustments to the compression parameter using the norm as an indicator as to the suitability of the basis set. A similar process can be used for a grid-based basis set as is used in chapter 4. In this case the grid spacing parameter is automatically adjusted until a norm within an acceptable range is calculated. This acceptable range is determined by the input parameters which are discussed more in Appendix C.

(b) The norm, the autocorrelation function and the populations of the wavefunction on different potential energy surfaces are used to keep track of the evolution of the wavefunction. For all propagation systems, the norm and the sum of populations should be conserved and numerically identical within a tolerance on the order of 10^{-6}, and, in addition, for the MCEv2 system the Ehrenfest energy, calculated as the sum of the single configuration Ehrenfest Hamiltonian values, should also be conserved to a similar level throughout propagation. The autocorrelation function is used in many applications as the primary result of a simulation, or acted on by a Fourier transform to generate a Franck-Condon spectrum [24].

(c) The position of the coherent states is checked before each time step to ensure that the coherent states are neither too far apart nor too closely packed. This is done because, if coherent states move too far away from each other, the overlap becomes negligible, effectively removing the coupling between the configurations and causing the simulation to fall into the semi-classical regime. If, however, the coherent states are too closely packed, the norm will increase exponentially, causing the simulation to fail. If this check fails, the initial basis set can be recalculated without the need to restart the program manually.

A.1.2 Program Implementation

The program itself is written mainly in Fortran95, with some of the external subroutines, for example those for generating the random numbers, written in a Fortran77 style. The source code is split into various modules which group together related subroutines and functions, such as a module for data input subroutines, and another for output subroutines. The list of these modules and their dependencies are given in table A.1.

Module Name	Module Purpose	Module Dependencies
GlobalVars	Contains global variables	None
Allocation	Allocates and de-allocates arrays	GlobalVars
Output	Generates data output files	GlobalVars
SysRedirect	Switchboard to redirect the correct system-specific functions to the main program	GlobalVars Various system specific modules
Ham	Calculates, among other things, the Hamiltonian, overlap, autocorrelation function and norm	GlobalVars Allocation SysRedirect
Chks	Runs checks on the basis set to ensure proper propagation	GlobalVars Ham
BsetAlter	For basis set cloning, adaptive grids, etc.	GlobalVars Allocation Output Ham
ReadPars	Read in the simulation parameters from input files	GlobalVars Allocation SysRedirect Ham
Derivs	Time derivative functions for MCE and CCS propagation	GlobalVars Allocation Output SysRedirect Ham ReadPars
Prop	Propagates the wavefunction by a single timestep	GlobalVars Allocation Derivs
BsetGen	Generates the initial basis set	GlobalVars Allocation Output SysRedirect Ham Chks Prop
Main	Controls program flow	GlobalVars Allocation Output SysRedirect Ham Chks BsetAlter ReadPars BsetGen

Table A.1: Table of modules of the CCS/MCE program with their dependencies

Throughout the program, actions and calculations are carried out on the wavefunction. To aid this a defined type is set up in which all parameters of the wavefunction are kept, as shown in listing A.1.

```
type basisfn
   complex(kind=8), dimension(:), allocatable::z
   complex(kind=8)                            ::D_big
   complex(kind=8), dimension(:), allocatable::d_pes
   real   (kind=8), dimension(:), allocatable::s_pes
   complex(kind=8), dimension(:), allocatable::a_pes
end type basisfn
```

LISTING A.1: Definition of the "basisfn" data type

The basis set is made up of an array of this data type of a size equal to the number of basis functions. This has been done to aid the transfer of the basis set to dummy variables in subroutines, which is done often in the propagation section of the program. The z part of the basisfn type has dimensions equal to the number of degrees of freedom, and the parts with the suffix "_pes" have dimensions equal to the number of basis states in the quantum system (or number of potential energy surfaces in the interaction), which in the case of the simple spin boson model is two. "D_big" is the D_k pre-factor. This wavefunction structure can be applied to MCEv1 simulations if "D_big" is set to 1 and kept constant, or applied to CCS simulations if "d_pes" is set to 1 and kept constant.

The simulation parameters such as the number of basis functions "nbf", the number of degrees of freedom "ndim" and the number of quantum states "npes" are all global variables, and are stored in a separate module which only has variable definitions and initialisations in it. The declaration for the "basisfn" defined-type is in this module, as is the declaration of a constant i equal to the imaginary unit and of the defined type "hamiltonian", the structure of which is found in listing A.2.

```
type hamiltonian
   complex(kind=8), dimension(:,:), allocatable::Hjk
end type hamiltonian
```

LISTING A.2: Definition of the "hamiltonian" data type

The "hamiltonian" defined type allows a 2D array of size $nbf \times nbf$ to be made where each element is a 2D array of size $npes \times npes$. This means the entire Hamiltonian can be held in a single variable in a way which is more intuitive than the alternative 4D array. Also in the global variable module are error checking variables and the arrays for the real and imaginary parts of $|\mathbf{z}_0\rangle$ and the time propagation variables.

It should be noted that throughout the course of writing the program, error checking code has been included. These error checking routines can be split into two types: fatal errors and non-fatal errors. The fatal errors are usually caused by problems with the machine, the code, or the input files and are handled by checking for the error and if it is present returning an error message and setting the global variable `errorflag` to equal 1, then escaping the subroutine or function to the main program. At the beginning of each subroutine or function in the program, there is the following check and so once an

```
1  if (errorflag==1) return
```

Listing A.3: This check for previous errors starts each subroutine or function

error has been detected every subsequent process is skipped until the end of the program where if the "errorflag" value is not zero an output of the status of the program is generated. This was written in, rather than simply placing a "stop" command on detected errors so that it was easier to determine from where any unexpected errors might originate.

Non-fatal errors are those which may be due to the distribution of the basis set generated. These errors would be things like the energy of the trajectories being too large or the position components of the coherent states being too widely spaced. Such problems, while not resulting in severe problems straight away may cause problems later during the time propagation of the wavefunction. In situations like this the basis set can be re-calculated by setting a second flag, `restart` to equal 1, which is the condition to carry on to the next iteration of a loop encompassing the wavefunction generation subroutines. An example of a check for these non-fatal errors is the Energy check subroutine, which is called every time a coherent state is generated. This calculates the Ehrenfest Hamiltonian, which initially is equal to $H_{\text{ord}}^{(rr)}$, where r is the initial quantum state, and returns the real part. A too large/small value of this would indicate a basis function which may cause problems in the calculation of C_k or D_k, or during time propagation.

A.2 Basis Set Creation

The basis set creation process is very important to the success or failure of a simulation. There are two main parts to generating the basis set correctly : generating the coherent states, and calculating their amplitudes. In this section the basis set generation procedure will be discussed in detail, starting with the generation of the coherent states followed by their amplitudes before exploring how the validity of the basis set can be

confirmed. When talking about the coherent states, only the case of swarms will be considered, as the process of sampling a regular grid is trivial and the process of generating trains is explored in the main text. Also omitted are the assignments of the action, which is always initially at $S_k = 0 \; \forall \; k$, and of the values for the single configuration amplitudes in MCEv2 as these are $d_k^{(r)} = (1 + 0i) \; \forall \; k$ where r is the initial electronic state.

A.2.1 Monte Carlo Sampling of Coherent State Basis Set

When constructing the basis set as a swarm, the values for the coherent states are selected from pre-defined distributions. In real terms this means that as $F(\mathbf{z}_k)$ from equation (2.3.20) is a set of Gaussian distributed values, the values for the quantity $\mathbf{z}_k$ can be found using random numbers selected for a set upon which a Gaussian distribution has been imposed, hence equation (2.3.20) gives

$$\mathbf{z}_k^{(m)} = z^{(m)} + \alpha_c(\text{C.R.G}) \tag{A.2.1}$$

where C.R.G denotes a complex random Gaussian distributed value. It should be noted that these values must also correspond to the general equation for a coherent state with momentum and position data given in equation (2.3.5), and so the real part is multiplied by $\sqrt{\gamma/2}$ and the imaginary part by $\sqrt{1/2\gamma}$. As for all simulations $\gamma = 1$, this can be largely ignored if the system is divided by $\sqrt{2}$.

Two methods of creating the swarm of normally distributed random numbers were considered. The first uses a subroutine labelled "GAUSS_RANDOM" modelled upon one supplied in "Numerical Recipes" by Press *et al.*[238]. This subroutine generates a Gaussian random number by applying a Box-Muller transformation to put these random numbers within the envelope of a Gaussian distribution with a width of 1 and a mean value of zero. Once this subroutine returns a value, it can be divided by $\alpha_c\sqrt{2}$ to get the real part of the $\mathbf{z}_k^{(m)}$ value, after which the subroutine is called again and the result divided by $\alpha_c\sqrt{2}$ and multiplied by i to get the imaginary part of $z^{(m)}$. The second method uses a library of random number generation subroutines written by Richard Chandler and Paul Northrop[239], and later modified by Stuart Reed, originally designed to extend and supersede the more commonly used NAG library. The relevant subroutine, labelled "ZBQLNOR", takes as its arguments a mean and a width parameter and returns a Gaussian distributed random number from the required distribution. In the case of the initial basis set $\mathbf{z}_k^{(m)}$, the width is $\alpha_c/\sqrt{2}$ while the centre of the real

and imaginary parts will be the real and imaginary parts of $z_0^{(m)}$, which is found by different methods depending on the system being simulated.

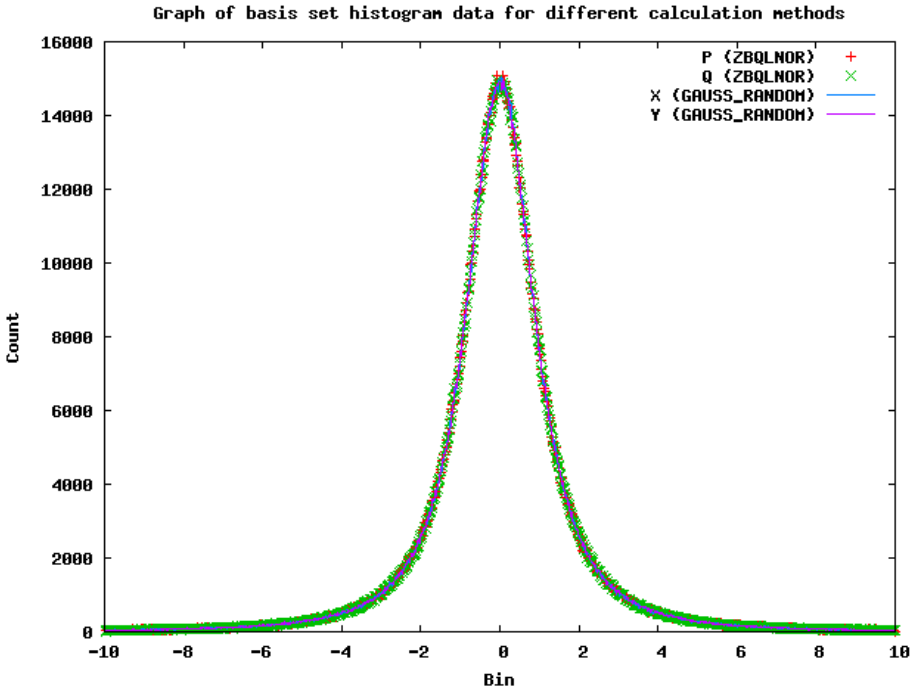


FIGURE A.3: Comparison between the distributions of the initial basis function $\mathbf{z}_k^{(m)}$ showing the real and imaginary parts as separate data sets. As the FG CS width parameter is set to $\gamma = 1$, the real and imaginary parts coincide almost exactly. The difference between the GAUSS_RANDOM and ZBQLNOR subroutines is shown to be undetectable, and so from a mathematical point of view the two are equivalent. From a computing and robustness point of view however, the ZBQLNOR subroutine is still superior.

```
sigb=0.5
gamma=1.0d0
sigx=gamma/sqrt(2.0)
sigy=1/(sqrt(2.0)*gamma)

do i=1,ndim
mup(i)=0.0d0                          ! This loop gives the
muq(i)=0.0d0                          ! values for the initial CS
end do

do j=1,ndim
call GAUSS_RANDOM (xtemp)             ! This loop generates the
call GAUSS_RANDOM (ytemp)             !values for z_k
x(j)=muq(j)+(xtemp*sigb*sigx)         ! The width and mean for x&y
y(j)=mup(j)+(ytemp*sigb*sigy)         ! are applied after initial
q(j)=ZBQLNOR(muq(j),sigb*sigx)        ! call while mean & width values
p(j)=ZBQLNOR(mup(j),sigb*sigy)        ! are input parameters for p&q
end do
```

LISTING A.4: Code to generate the initial coherent state and initial basis state using GAUSS_RANDOM and ZBQLNOR

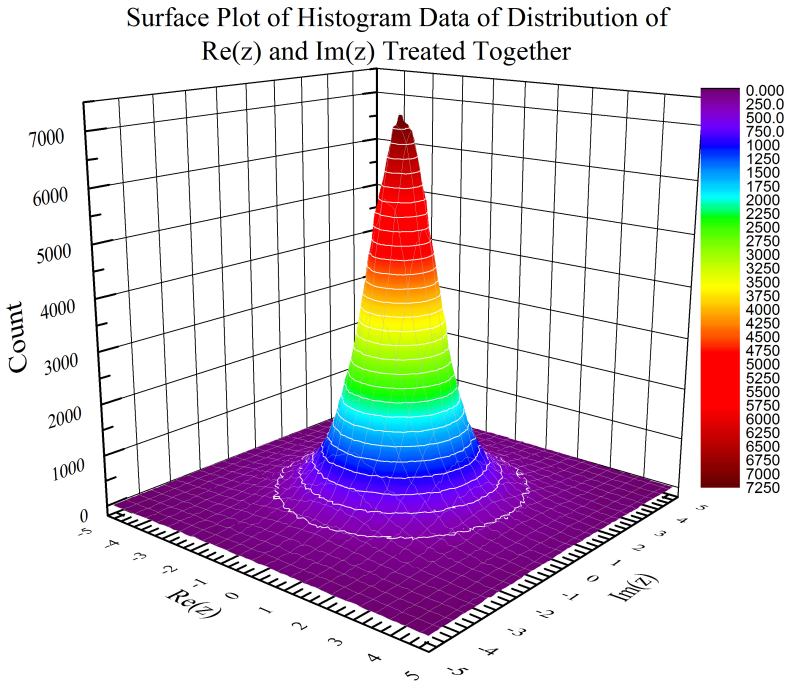


FIGURE A.4: Surface plot showing the two dimensional distribution of the complex z values on the real and imaginary axes for a single coherent state with 10000000 degrees of freedom. As can be seen not only do the independent distributions coincide, but the dependent distributions show a 3 dimensional Gaussian peak, as would be expected.

The less important difference between the two subroutines in their running is that while the ZBQLNOR subroutine and its dependants are built upon the super-wizz-o algorithm of Marsaglia & Zaman and so have a periodicity of 2^{1376}, the GAUSS_RANDOM subroutine uses the intrinsic set of random number subroutines which has a much lower periodicity of 2^{57}. This however is not a problem for the purposes of this program since it is unlikely that a basis set of more than 7.2×10^{16} values will be used. The more important difference between the two subroutines is that while the GAUSS_RANDOM subroutine is easily implementable due to its simplicity, the ZBQL* family of subroutines have in built various fail-safes and error catching procedures making it more robust and more likely to show an error if one is present, and in a recognisable way. It is unfortunate that the ZBQL* family of subroutines is not entirely thread safe, and so can cause problems with parallelisation.

To check the running of the two subroutines in comparison to each other, a small checking program was made which creates two large basis functions of around 10000000 degrees of freedom each (ensuring a very small contribution from shot noise), and plots the values of the real and imaginary parts of the coherent states as histogram data. This plot is shown in figure A.3, and the code snippets used to generate this plot is given in listing

A.4. To ensure that the real and imaginary parts of the data sets coincided in reality as well as they appeared to, a surface plot was created which plotted both the real and imaginary parts together as histogram data. This is shown in figure A.4. The values for the initial coherent state $z_0^{(m)}$ used are set at $(q, p) = (0, 0)$ in all degrees of freedom.

A.2.2 Generation of the Normalised D Prefactor

Once the basis set coherent states are assigned it is possible to find the values for the set of D_k prefactors (or in the case of MCEv1, the $d_k^{(r)}$ single configuration amplitude where r is the initial electronic state). This is done through calculation of the initial C_k prefactor which was used often in the earlier simulations of the CCS method as the main prefactor leading the coherent state basis. As mentioned earlier,

$$C_k = \langle \mathbf{z}_k | \, \Psi(0) \rangle = \langle \mathbf{z}_k | \, \mathbf{z}_0 \rangle \,, \tag{A.2.2}$$

and using equation (2.3.11) this can be calculated using a simple loop such as in listing

```
zinit(1:ndim) = cmplx(muq(1:ndim),mup(1:ndim),kind=8)

do i=1,nbf
  zpq(1:ndim) = bs(i)%z(1:ndim)
  C_k(i) = ovrlpij(zpq, zinit)
  C_kinp(i) = C_k(i)
end do
```

Listing A.5: Loop to generate values of C_k

A.5, where "mup" and "muq" are the imaginary and real parts of the initial coherent state, the form of which is dependent upon the system being simulated. The distribution of values generated by this process is unusual in that for a single degree of freedom, the imaginary and real parts of the C_k value have different distributions, with an approximate Gaussian in the distribution of the real values centred on zero, but an exponential growth from zero to one in the imaginary values. As the number of degrees of freedom in the system increases, the real and imaginary parts get closer together and both tend towards a Gaussian peak, as shown in figure A.5 which shows the real and imaginary parts of the distribution of C_k independently of each other, and also in figure A.6 which shows the C_k distribution on both the real and imaginary axes. The distribution of the D_k values is more regular. It starts out as a wide Gaussian on the real and imaginary axes, but very quickly smooths and narrows to a very tight Gaussian peak centred around zero (see figure A.7). As mentioned earlier, if the overlap matrix Ω_{jk} is calculated, D_k can be calculated by use of equation (2.3.25). There are two ways of calculating D_k

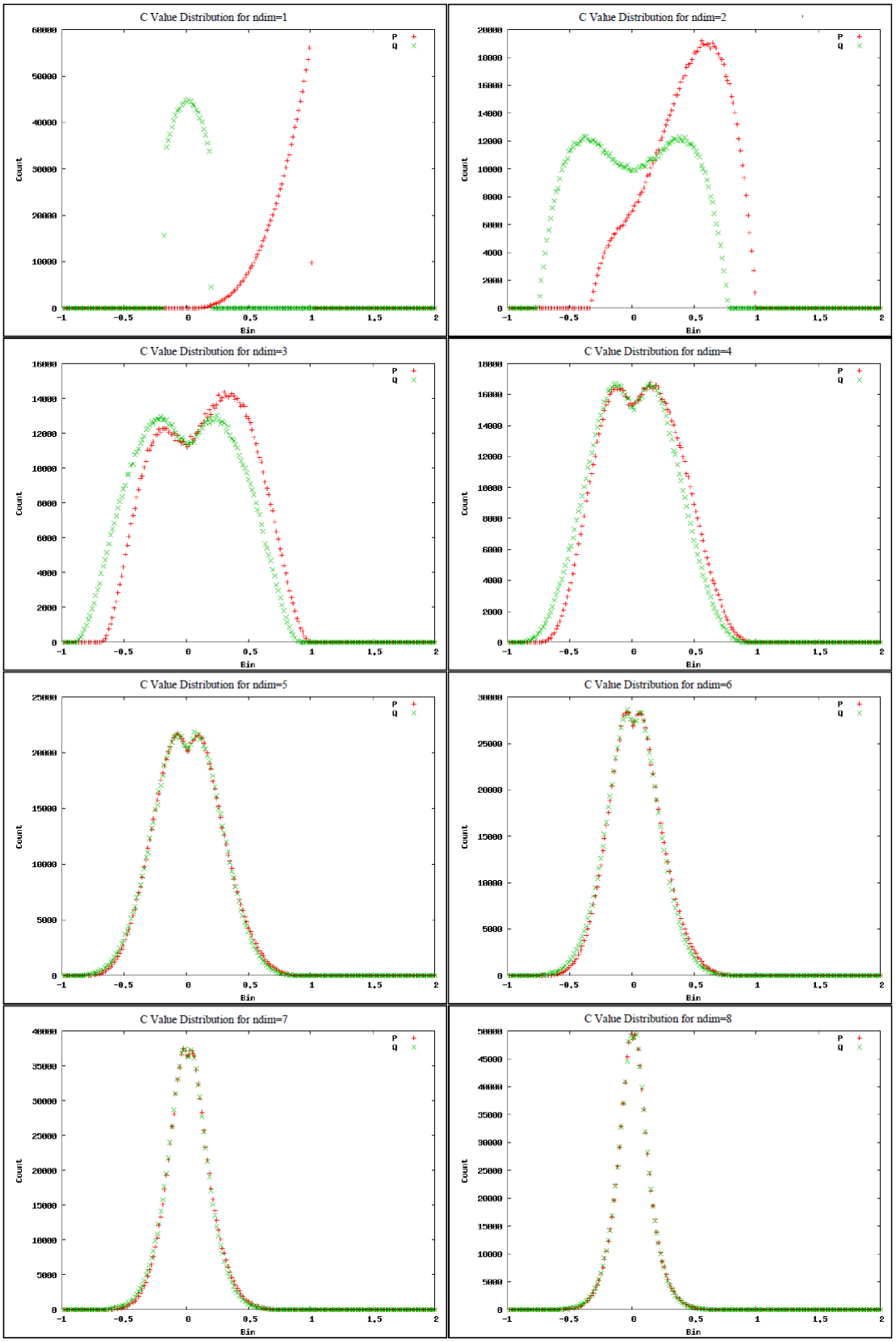


FIGURE A.5: Set of graphs showing the Real and Imaginary parts of the distribution of values for the C_k prefactor independently of each other for the number of degrees of freedom in the system, ndim=1 to 8. It can easily be seen how as the number of degrees of freedom increases, the closer together the shapes of the real and imaginary parts and the closer both tend towards a Gaussian distribution

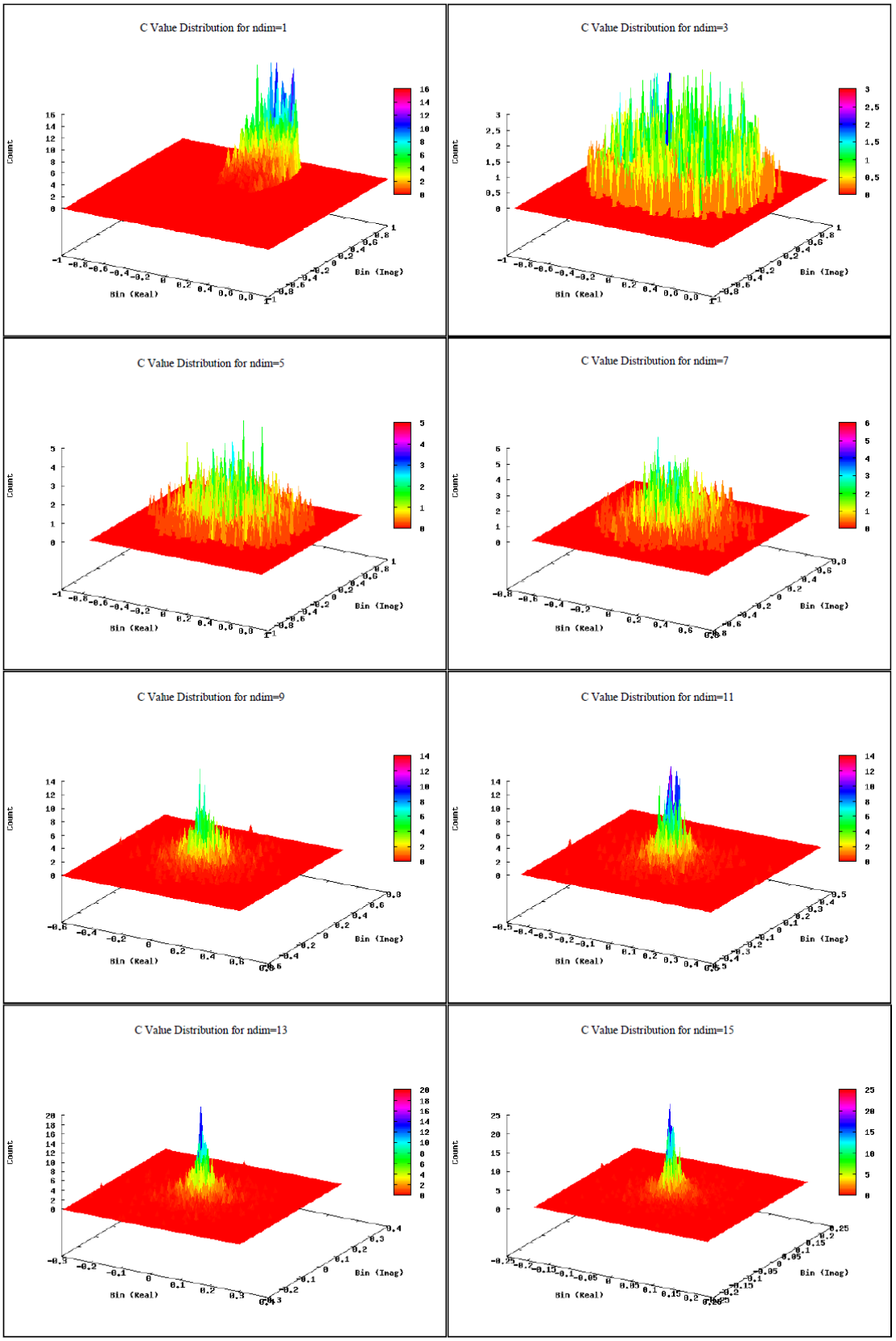


FIGURE A.6: Set of graphs showing the Real and Imaginary parts of the distribution of values for the C_k prefactor as 3D surface plots for the number of degrees of freedom in the system, ndim=1 to 15. These graphs show how the real and imaginary parts combine to form the overall distribution and also show how the distribution becomes narrower and more smooth.

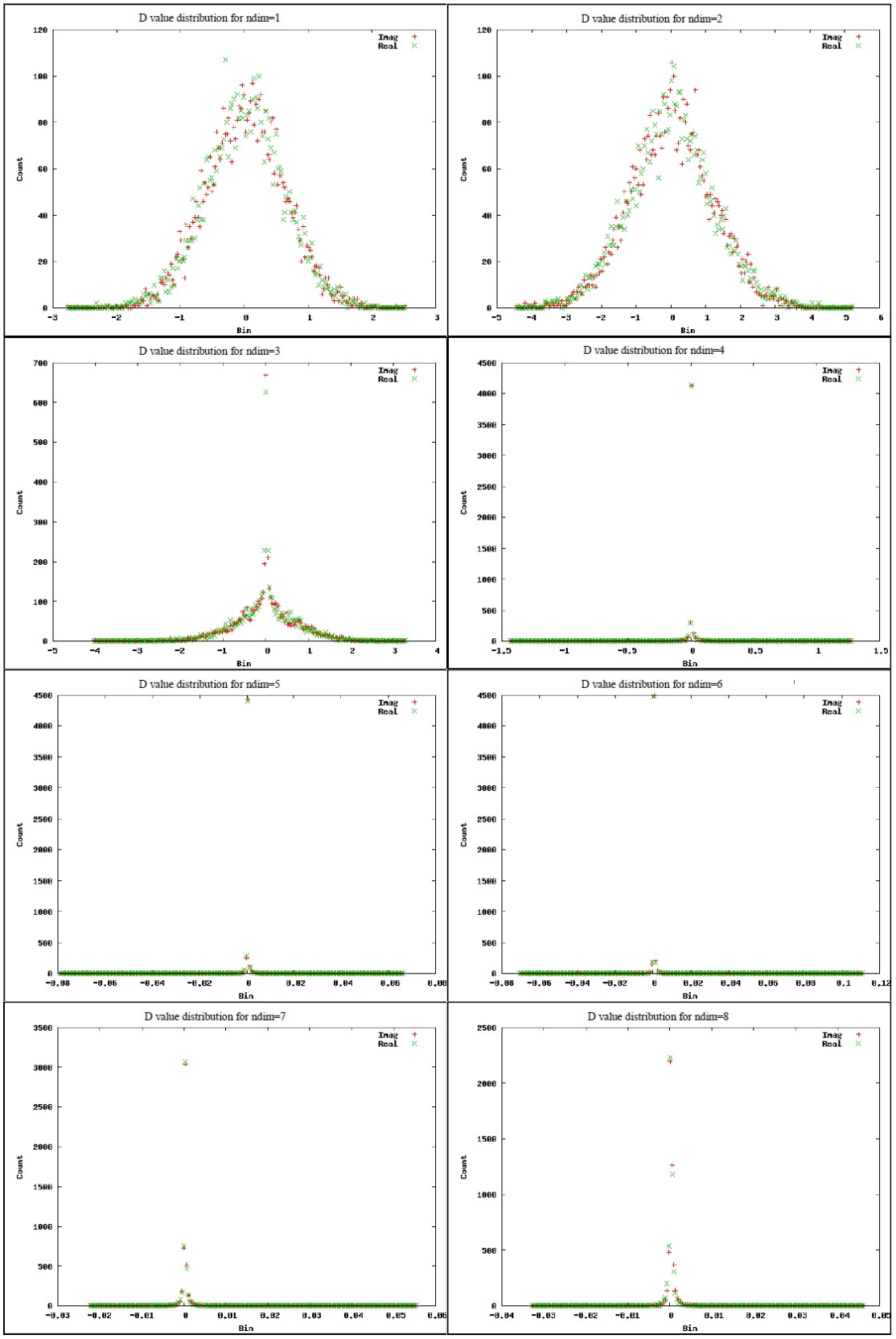


FIGURE A.7: Set of graphs showing the Real and Imaginary parts of the distribution of values for the D_k prefactor independent of each other for the number of degrees of freedom in the system, ndim=1 to 8. It can be easily seen how the distribution becomes narrower and more smooth much more quickly than the distribution for C_k and exhibits none of the unusual distribution shape that was found for C_k.

from this relationship - firstly by finding the solution to the set of linear equations

$$\Omega_{jk} D_j = C_k \tag{A.2.3}$$

which can be carried out by use of the "zgesv" subroutine which is part of the LAPACK library of matrix manipulation subroutines, or secondly by inverting the overlap matrix and calculating the matrix multiplication

$$D_j = \Omega_{jk}^{-1} C_k \tag{A.2.4}$$

which can be achieved by finding the eigenvalues and eigenvectors of the overlap matrix using the "zheev" subroutine, which is also a part of the LAPACK library. The inverted matrix is found from the eigenvalues, λ and eigenvectors, ν using the following transform:

$$\begin{aligned} \Omega\nu &= \lambda\nu \\ \nu &= \Omega^{-1}\lambda\nu \\ \lambda^{-1}\nu &= \Omega^{-1}\nu \\ \Omega^{-1} &= \frac{\nu\nu^{-1}}{\lambda} \end{aligned} \tag{A.2.5}$$

and so this can be implemented in the code as shown in listing A.6

```
call zheev('V','U',nbf,ovrlp,nbf,om_eigen,WORK,LWORK,RWORK,info)

! zheev returns an array of nbf eigenvalues ''om_eigen'' and overwrites the ''
    ovrlp'' matrix with nbf eigenvectors for a nbf x nbf overlap matrix

do i=1,nbf
  do j=1,nbf
    do k=1,nbf
      if (abs(om_eigen(k))/maxval(abs(om_eigen)).gt.1.0d-3) then
        om_inv(i,j)=om_inv(i,j) + ovrlp(i,k)*dconjg(ovrlp(j,k))/om_eigen(k)
      end if     !om_inv is the inverse overlap matrix
    end do
  end do
end do

! This loop will only assign values to the inverse matrix if the eigenvalue is
    large enough. This removes the possibility of extremely large values arising
    from small overlaps

D = matmul(om_inv,C_k)    ! Generates the D array.

return
```

Listing A.6: Subroutine to generate D_k through matrix inversion

Various error checks have been removed from the above listing, which check that no zero eigenvalues exist, that the inverse matrix is hermitian, and that the zheev subroutine executed properly. If any of these checks fail a flag is returned to the main program which causes it to recalculate the wavefunction. The code to generate the D_k array by solving a set of linear equations is much simpler, and is given in listing A.7, again with error checking procedures omitted.

```
call zgesv(nbf,nrhs,ovrlp,nbf,IPIV,C_k,nbf,info)

! zgesv overwrites the C_k array with the values for the D array.

! Error checking code omitted

D = C_k

return
```

LISTING A.7: Subroutine to generate D_k through linear equations

After various tests with known input basis sets, it was found that both methods returned identical answers to an acceptable level of precision. The benefit of the zheev method however is that in cases with very sparse overlap matrices it is less likely generate errors due to the check for small eigenvalues. The zgesv method has no such safety net which can be implemented. To see if there were other benefits of one method over the other, a set of tests were carried out in which a wavefunction was calculated using one or the other of the methods and the norm $|\langle\Psi|\ \Psi\rangle|^2$ and the population sum $\sum_r |\langle\Psi|\ r\rangle\langle r|\ \Psi\rangle|^2$ was calculated, and the amount of time taken to calculate the wavefunction was recorded. This was done for many values of the number of degrees of freedom and the number of basis functions with ten repeat recalculations. The plots of the averages resulting from these tests are in figure A.8.

As is immediately apparent, the norm and population sum is identical apart from statistical fluctuations, and so as expected the quality of the data generated by the two methods is the same. The amount of runtime, which is directly related to the amount of computational power needed to run the calculation, shows very large differences however. The lower two plots of figure A.8 show that if using the zgesv subroutine for 1 to 1000 basis functions the time ranges from 0.1 seconds to 27.4 seconds, while using the zheev subroutine under the same conditions ranges from 0.1 seconds up to 3170 seconds. As solving equation (2.4.21) in the propagation of the wavefunction also requires a similar equation to be solved to obtain $\dot{D}_k$, this amount of time is far too large for the zheev subroutine to be usable. A direct comparison between the run times for the zgesv and zheev subroutines is given in figure A.9 for 1 to 1000 basis functions with a single

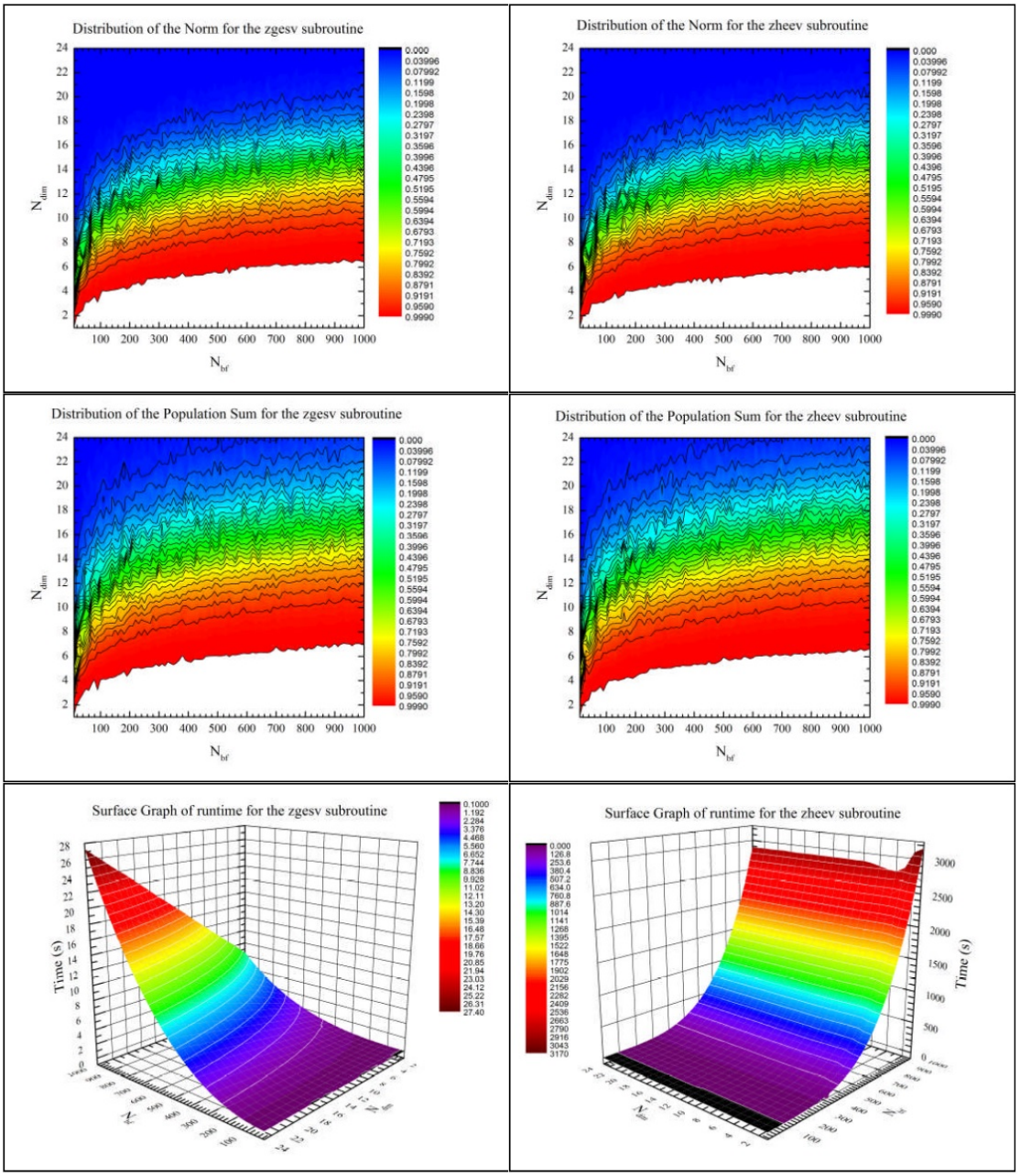


FIGURE A.8: Set of graphs showing the results from the tests on the zgesv and zheev subroutines used for calculation of the D_k array. In the norm and population sum graphs, the white area indicates the part where the norm/popsum is ≥ 0.999, which is used in the previous research as the area where the wave function is valid.

degree of freedom. This plot confirms the increased computational power needed to run the zheev subroutine, and also seems to show a fourth-order polynomial relationship, indicating that the amount of computational power needed is likely to grow much faster than the size of the basis set. As such the zgesv subroutine was used, and in an attempt to protect against very small non-zero overlaps which would result in large terms in an inverse matrix, a check was placed in the overlap matrix generation subroutine which keeps the overlap value at zero if the calculated value is less than 1×10^{-5} (this cutoff value is purely empirical).

A.2.3 Basis Set Verification by way of the Initial Norm

The norm of the initial wavefunction is one of the best indicators of the usability of the wavefunction, as it incorporates data from all the parameters of the wavefunction using the equation

$$\langle \Psi | \ \Psi \rangle = \sum_{j,k=1}^{N} \left[D_j^* \langle \mathbf{z}_j | \ \mathbf{z}_k \rangle D_k \left(\sum_r a_j^{*(r)} a_k^{(r)} \right) \right]. \tag{A.2.6}$$

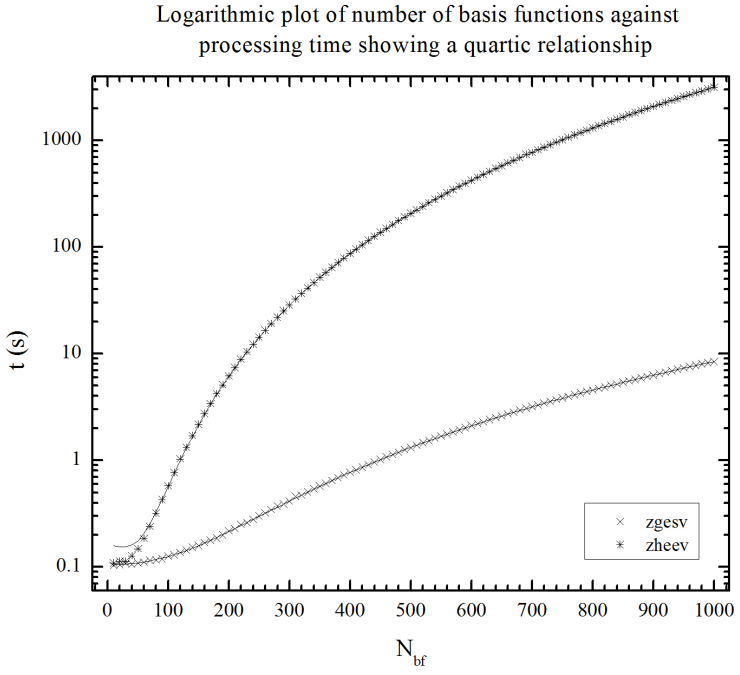


FIGURE A.9: Graph directly comparing the calculation time for the two methods against the number of basis functions for 1 degree of freedom. The time was plotted against the number of basis states as this was the greatest contributor to the calculation time. The trends seem to follow a quartic polynomial curve, and so increases in the number of basis functions will have a large effect on the runtime of the program, and hence the computational power needed.

for a system of r quantum states and N basis functions. It is also a measure of the accuracy of the method because, as mentioned earlier, the only approximation used in the MCE method is the discrete sum over all basis functions in the ansatz which will

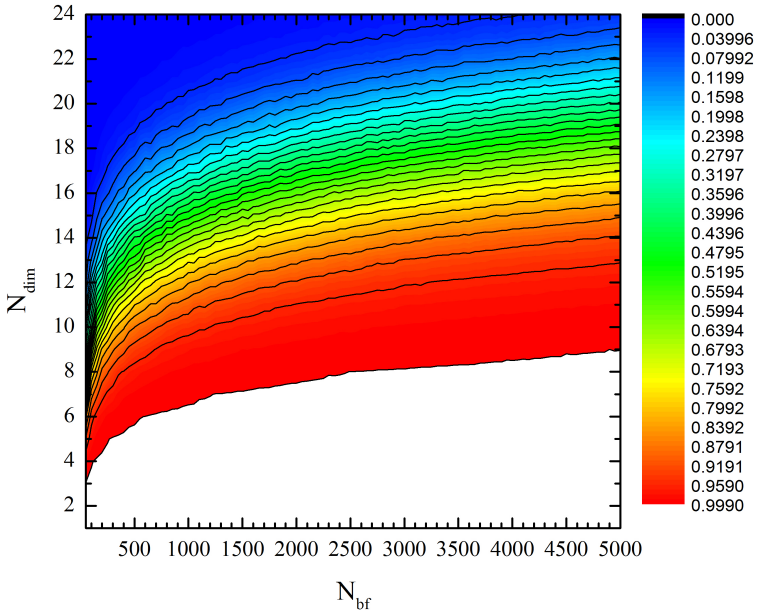


FIGURE A.10: Contour graph showing the norm of the initial wavefunction for 50-5000 basis functions and 1-24 degrees of freedom. Unlike previous graphs for the norm, this shows a linear scale for the number of basis functions and has a much larger range. The white area again is the area in which the norm is ≥ 0.999, and it can be seen that with an increasing number of basis functions the "acceptable" area still does not show any signs of being able to reach the larger numbers of degrees of freedom for the system.

approach the value given from integration at a large enough total number of basis functions. As shown in figure A.10, and in the plots in figure A.8, even with a large number of basis functions the norm may not be close enough to 1 for larger numbers of degrees of freedom. As such a second method of achieving an acceptable norm value is needed, and this comes in the form of the compression parameter α_c.

As was mentioned earlier, the α_c parameter supplies the width of the Gaussian distribution from which the set of basis states $|\mathbf{z}_k\rangle$ is selected. As such decreasing the value of the compression parameter will decrease the spread of the basis functions making the overlap matrix less sparse and thus increasing the norm. To find the maximum values of α_c for a range of different values of N_{bf} and N_{dim} a test was carried out which decreased the value of α_c by a factor of 0.9 and recalculated the basis set until a norm was found of ≥ 0.999. The results of this test are seen in figures A.11 and A.12.

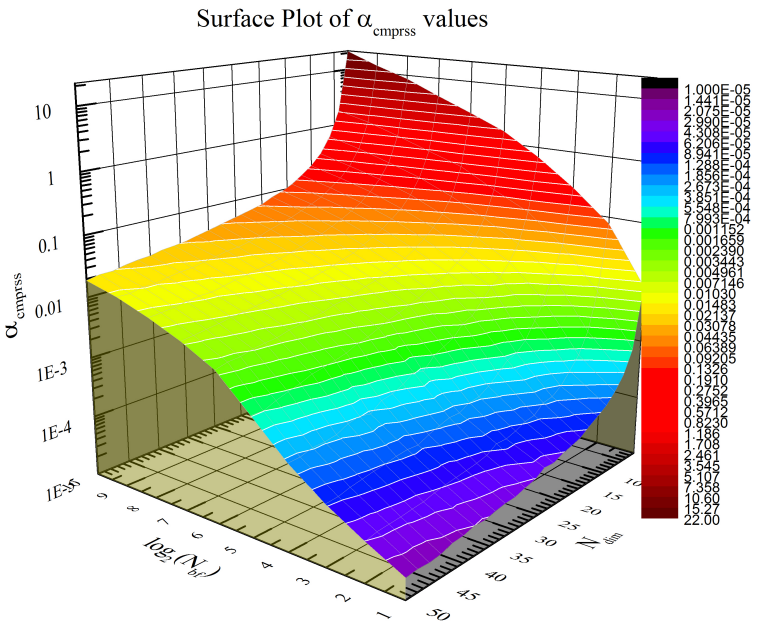


FIGURE A.11: Surface plot showing the average values for the compression parameter α_c needed to generate a wavefunction with a norm of $\geq$0.999. Due to the large spread of values the results are displayed on a logarithmic scale.

As can be seen from these plots there is definite dependence on both the number of basis functions and the number of degrees of freedom in the system. A plot of α_c against the ratio of these two variables, shown in figure A.13, did not show any discernible pattern or trend, and so the best option for ensuring an acceptable description of the wavefunction is a recursive algorithm similar to that used to generate the data for figures A.11 and A.12.

If using a grid structure for the initial basis set a similar procedure can be used with the grid spacing parameter Δ. While the relationship between the number of basis functions and the grid spacing is not the same as for a swarm, to properly describe the wavefunction the grid must have a certain density. If this condition is not met, the initial norm will be below the minimum allowed level which, for the high harmonic generation simulations, was in the region of $|\langle\Psi|\,\Psi\rangle \geq 0.99999$. Listing A.8 shows the conditional governing the this recursive algorithm.

Here, if the relevant parameter results in a norm which is outside the allowed limits, that parameter is raised or lowered by 5% and a flag named `restart` is given a value of 1, then passed back to the main module. Once this happens the basis set is recalculated using this new parameter and checked again, being changed again in the same way if necessary, and so on. This process will result always in a basis set which can adequately describe the wavefunction.

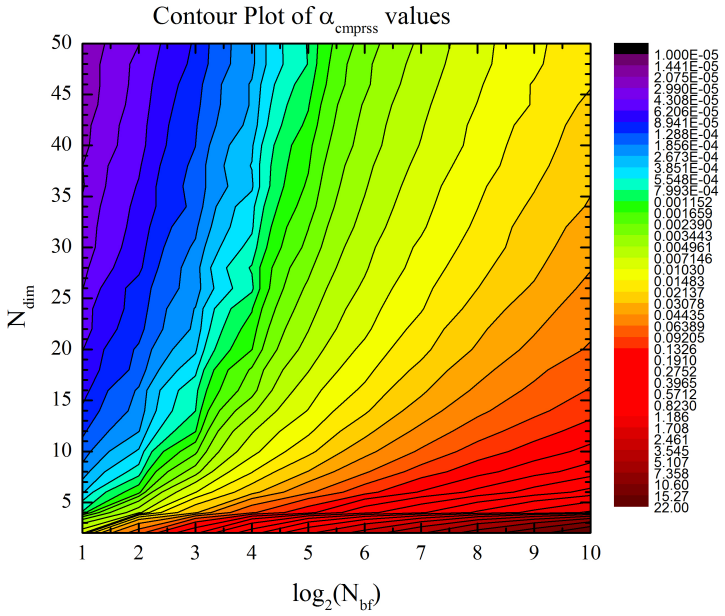


FIGURE A.12: Contour plot showing the average values for the compression parameter α_c needed to generate a wavefunction with a norm of $\geq$0.999. Due to the large spread of values the spread of colours correspond to a logarithmic scale.

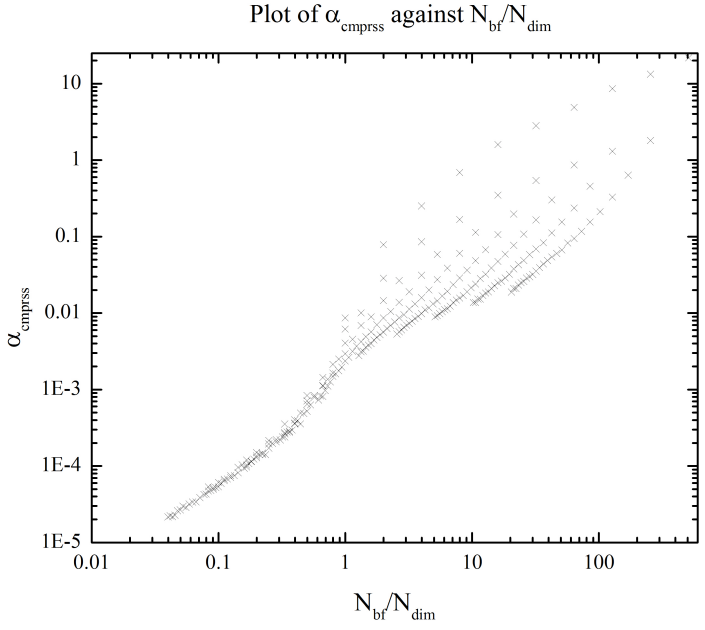


FIGURE A.13: Plot of the ratio of N_{bf} and N_{dim} against the average maximum allowed values for the compression parameter. Despite a short approximately linear section towards the beginning, the values diverge once N_{bf} becomes greater than N_{dim}. Unfortunately N_{bf} is almost always greater than N_{dim} in simulations and so it appears that any direct relationship by which the optimal value of α_c can be calculated is not available.

```
if (absnorm.lt.lowlimnorm) then
  write(6,'(a,es16.8e3)'), "Initial Norm too low with a value of ", absnorm
  if (basis.eq."GRID") then
    gridsp = gridsp * 0.95d0
  else if (basis.eq."SWARM") then
    alcmprss = alcmprss * 0.95d0
  end if
  restart = 1
else if (absnorm.gt.uplimnorm) then
  write(6,'(a,es16.8e3)'), "Initial Norm too high with a value of ", absnorm
  if (basis.eq."GRID") then
    gridsp = gridsp * 1.05d0
  else if ((basis.eq."SWARM").or.(basis.eq."SWTRN")) then
    alcmprss = alcmprss * 1.05d0
  end if
  restart = 1
end if
```

LISTING A.8: Condition governing the recursive algorithm for finding the correct compression factor for a swarm basis set, or finding the correct spacing for a grid basis set

A.3 Basis Set Propagation

The process labelled "Take a time step" in figure A.2 is fairly complex as it is one of the most important parts of the program. The simplest method of taking a time step would be to multiply the time derivative of a variable by the time step and in the Euler method of numerical integration, this would be all that would be needed, as in general

$$\begin{aligned} y_{n+1} &= y_n + \Delta x \frac{dy}{dx} \\ \therefore y &= \lim_{\Delta x \to 0} \sum \Delta x \frac{dy}{dx} = \int \frac{dy}{dx} \mathrm{d}x. \end{aligned} \tag{A.3.1}$$

While this is mathematically accurate, it is not really a viable method of propagating the wavefunction as it relies very heavily on the size of the step for the limit of the accuracy. An alternative to this is to use one of the Runge-Kutta methods. Runge-Kutta methods propagate a function by taking multiple points along the curve and using the derivatives at that point to achieve a better estimate of the position of the function at the end of the time step. This done through a series of equations such as

$$\begin{aligned} k_1 &= dtf(x_n + a_1, y_n) \\ k_2 &= dtf(x_n + a_2, y_n + b_{2,1}k_1) \\ &\vdots \\ k_m &= dtf(x_n + a_m, y_n + b_{m,1}k_1 + b_{m,2}k_2 + \cdots + b_{m,m-1}k_{m-1}) \end{aligned} \tag{A.3.2}$$

which are combined by

$$y_{n+1} = y_n + c_1k_1 + c_2k_2 + \cdots + c_mk_m \tag{A.3.3}$$

where $f(x, y)$ is the function used to calculate the derivative of y. The coefficients for a Runge-Kutta method are usually given in a Butcher tableau, such as in table A.2 below. The most widely used and simplest of the Runge-Kutta methods is the "rk4" method,

a_1					
a_2	$b_{2,1}$				
a_3	$b_{3,1}$	$b_{3,2}$			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
a_m	$b_{m,1}$	$b_{m,2}$	$\cdots$	$b_{m,m-1}$	
	c_1	c_2	$\cdots$	c_{m-1}	c_m

TABLE A.2: Butcher tableau for a general Runge-Kutta method.

so called because it uses four calculation points to take a step between t_0 and t_1. The comparative simplicity shows in the fact that four terms are calculated and all four terms are used to calculate the step, while most other Runge-Kutta methods require more values of k_m to be calculated than are actually used in the final calculation of y_{n+1}; for example reference [240] gives a sixth order Runge-Kutta method and requires seven values of k_m to be calculated. The Butcher tableau that dictates the rk4 method is given in table A.3, which shows how only the previous value of k_m is used to calculate the next, rather than *all* previous values as is the case for higher order Runge-Kutta methods, and figure A.14 shows the rk4 graphically for variables labelled v and t, where the k_m values are labelled a_n.

While the rk4 method is simple and effective in many cases, it does not account for

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	1/3	1/3	1/6

TABLE A.3: Butcher tableau for the classical Runge-Kutta (RK4) method.

the possibility of fast moving functions. To spot a fast moving function without any unintentional aliasing effects the best system is one which can change the step size to account for fast motion in the function. In their "Numerical Recipes" book [238], Press *et al.* suggest the use of the Runge-Kutta-Cash-Karp system which is a set of coefficients that can be used for both fourth- and fifth-order propagation, shown in the Butcher tableau in table A.4. By comparing the difference between the two results a measure of the error in the fourth-order result can be found which in the case of this method scales as $(dt)^5$. If the error found from this process is Δ_1, and there is a desired margin of error Δ_0, then the time step needed to give that desired error will be

$$dt_0 = dt_1 \left| \frac{\Delta_0}{\Delta_1} \right|^{\frac{1}{5}}. \tag{A.3.4}$$

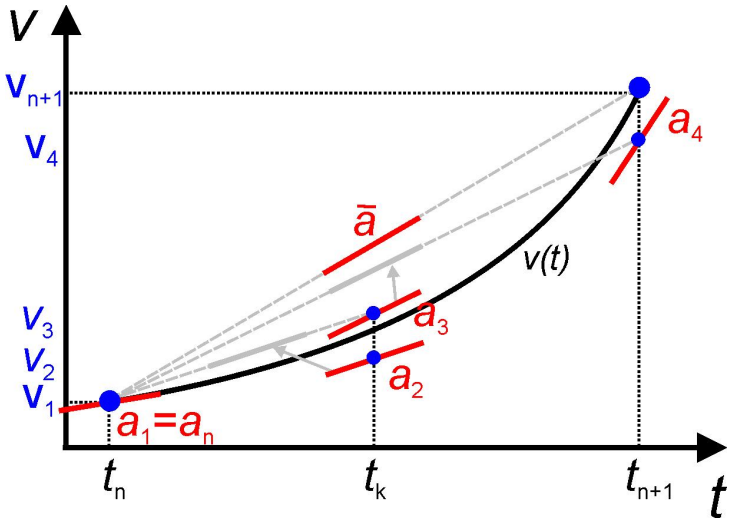


FIGURE A.14: Example plot showing the data points used to calculate a time step in the RK4 method. The locations of these points in the t direction relate to the value of the a_m values in the butcher tableau. This figure also illustrates how the solution of one part step is used to calculate the next (grey arrows). This picture is courtesy of http://www.ipf.uni-stuttgart.de

0						
1/5	1/5					
3/10	3/40	9/40				
3/5	3/10	-9/10	6/5			
1	-11/54	5/2	-70/27	35/27		
7/8	1631/55296	175/512	575/13824	44275/110592	253/4096	
c_i	37/378	0	250/621	125/594	0	512/1771
c_i^*	2825/27648	0	18575/48384	13525/55296	277/14336	1/4

TABLE A.4: Butcher tableau for embedded Runge-Kutta-Cash-Karp (RKCK45) method. The lowest row contains c_i^* which is used to calculate the fifth-order result. Subtracting the fifth-order result from the fourth-order result gives an estimate of the error.

Setting the values of Δ_0 is done by the assignment of fractional errors and, as suggested by Numerical Recipes [238], a slight deviation is built in to protect against points where the function is passing through zero, hence

$$\Delta_0 = \varepsilon \left(|y_n| + \left| \frac{dy_n}{dt} dt \right| \right) \tag{A.3.5}$$

where ε is the factor by which the value of the function is multiplied to get the fractional error for example 1 in 10^6. By this procedure the time step can be kept at the correct size to accurately describe the evolution of the variables without losing any accuracy.

In the implementation of the Runge-Kutta propagation scheme both the static time-step (RK4) and the adaptive time-step (Runge-Kutta-Cash-Karp) methods were encoded. As

such propagation is carried out over four subroutines. The first is a driving subroutine which chooses the propagation scheme and drives the process, the second checks the error and adapts the time-step if needed for the adaptive time-step case, the third carries out the calculations of the Runge-Kutta-Cash-Karp method for the adaptive time-step case and the fourth carries out the calculations of the RK4 method for static time-steps. All these subroutines are based upon those given in the "Numerical Recipes" book [238].

The driving subroutine, shown in listing A.9 starts by performing a check to ensure that the time-step selected does not overshoot the pre-defined end time, and to reduce it if needed. Following this the first set of time derivatives is calculated which will be used in propagation and also to generate Δ_0.

```
1
2  if ((time+dt-timeend)*(time+dt-timestrt) .gt. 0.0d0) then
3    dt = timeend - time     !ensure time parameter does not overshoot
4  end if
5
6  call deriv(bs, dbs_dt1, 1)  !calculates derivatives. the ''1'' is the
7                              !runge-kutta step index (i.e. first step)
8
9  if (step == "A") then       !adaptive step size
10   call allocbs(bserr0)      !the error margin
11   do k=1,nbf
12     bserr0(k)%z(1:ndim) = bs(k)%z(1:ndim) &
13       + dbs_dt1(k)%z(1:ndim) + tiny(0.0d0)
14     bserr0(k)%d_pes(1:npes) = bs(k)%d_pes(1:npes) &
15       + dbs_dt1(k)%d_pes(1:npes) + tiny(0.0d0)
16     bserr0(k)%s_pes(1:npes) = bs(k)%s_pes(1:npes) &
17       + dbs_dt1(k)%s_pes(1:npes) + tiny(0.0d0)
18     bserr0(k)%D_big = bs(k)%D_big + dbs_dt1(k)%D_big + tiny(0.0d0)
19   end do
20
21   call rkstpctrl (bs, dbs_dt1, dt, bserr0, dtfin, dtout, tempbs)
22
23 else if (step == "S") then  !static step size
24
25   call rk4 (bs, dbs_dt1, dt, dtfin, dtout, tempbs)
26
27 end if
28
29 bs = tempbs                 !write new wavefunction
30
31 return
```

LISTING A.9: Code for the driving subroutine for the numerical integration system.

This is done by calling a subroutine which in turn calls all the time derivative functions and returns all the numerical results together in a `basisfn` type variable, here called `dbs_dt1`. After that a check is made to see if the simulation is to use the adaptive step size subroutines or the static step size subroutine. If static is selected then the rk4 subroutine is called however if the adaptive step size is selected then a basis set type variable is populated as required by equation (A.3.5), after which the step control subroutine is called. The final line, which runs after the time propagation is complete,

```
1
2  eps = 1.0d-6
3  dt = dtprev
4  adap = 0
5
6  do
7
8    call rkck45(bs,dbs_dt1,dt,tempbs,bserr1) !single step
9
10   do k=1,nbf
11     err1z(k) = sum(realpart(dconjg(bserr1(k)%z(1:ndim))&
12       *(bserr1(k)%z(1:ndim))))
13     err0z(k) = sum(realpart(dconjg(bserr0(k)%z(1:ndim))&
14       *(bserr0(k)%z(1:ndim))))
15   end do
16
17   errmin = 1.0d10
18
19   do k=1,nbf         !calculates minimum error ratio
20     do r=1,npes     !over each of the derived values
21       errmin = min(abs(bserr0(k)%d_pes(r)/bserr1(k)%d_pes(r)), &
22         abs(bserr0(k)%s_pes(r)/bserr1(k)%s_pes(r)), &
23         errmin)
24     end do
25     errmin = min(abs(bserr0(k)%D_big/bserr1(k)%D_big),&
26       abs(err0z(k)/err1z(k)),errmin)
27   end do
28
29   errmin = errmin * eps
30
31   if ((errmin.lt.1).and.(adap.eq.0)) then
32     dt = sign(max((0.9*abs(dt)*(errmin**0.2)),dtmin),dt)
33     if (dt==dtmin) dtnext=dtmin
34     if (time+dt.eq.time) then
35       print "(a,e12.5,a,e12.5)", "Error! Underflow in step size at time = ", time
           , " for dt value of ", dt
36       errorflag = 1
37       return
38     end if
39     adap = 1    !ensures time is only changed once
40     cycle
41   else if (errmin.ge.1) then
42     dtnext = sign(min(0.9*abs(dt)*(errmin**0.2),dtmax),dt)
43     exit
44   else
45     exit
46   end if
47
48 end do
49
50 dtfin = dt
51
52 return
```

LISTING A.10: Code for the adaptive step control subroutine used in the numerical integration system.

overwrites the existing wavefunction `bs` with the new wavefunction.

The adaptive time-step control subroutine `rkstpctrl` deals with the changing of the time-step, the code for which can be found in listing A.10. The majority of the subroutine is enclosed in a loop which is controlled by a flag such that the loop will experience at most two cycles. Once the subroutine to calculate the step using the Runge-Kutta-Cash-Karp parameters is called, the error (calculated as the difference between the fourth-

and fifth-order results) is compared against the desired error Δ_0 using equation (A.3.4)

for all parameters, and a minimum value is selected. If this value is less than one then the time-step is reduced and the the step recalculated. This recalculation is only carried out once because upper and lower limits are pre-defined for the time-step and so if the calculated new time-step is lower than the minimum an infinite loop could form. If the minimum error ratio is greater than one however then the result is kept and an estimate is made of the next time-step, which is calculated using equation (A.3.4) and the subroutine returns.

The Runge-Kutta-Cash-Karp time-step calculation subroutine calculates equations (A.3.2-A.3.3) using the coefficients in the Butcher tableau in table A.4, the code for which is given in listing A.11. This is done by populating arrays with the Butcher tableau values (this process is omitted in the listing) and running through loops to generate the values for k_m, which are themselves stored in a 2d array of the derived `basisfn` type. A temporary basis set is used to supply values so that the Hamiltonian and overlap matrices can be recalculated for each intermediate step. Once all the values of k_m are calculated, they are combined as required by equation (A.3.3) and the error basis set `errbs`, corresponding to Δ_1 in equation (A.3.4), is generated using equation (A.3.3) but instead of using the c_m values, i.e. the upper values in the Butcher tableau, it uses the difference between the upper and lower values. Once this is complete, control is passed back to the step control subroutine so that the quality of the data can be examined by the step control system. When adaptive stepsizes are used, to obtain usable data, the raw data must undergo interpolation to get evenly spaced points which can then be averaged over multiple repeat simulations. This is done at the end of propagation. The subroutines used for interpolating this data are taken verbatim from the "Numerical Recipes" book [238] and so will not be reproduced here.

The static step size subroutine calculates the step using the fourth-order Runge-Kutta method. The implementation is very similar to that of the time adaptive step subroutine which uses the Runge-Kutta-Cash-Karp method, however this subroutine is a bit simpler due to the lack of an error function or off diagonal terms in the Butcher tableau (table A.3). As before arrays are populated with the values from the Butcher tableau, after which a two-dimensional array of the derived `basisfn` type is generated and allocated to be used for holding the derivative terms, and the first time derivative set of variables are written to this array. Following this the different steps are calculated as required by equation (A.3.2) and the time step is calculated from these steps by equation (A.3.3).

```
1
2  allocate (dbs_dt(6,nbf))    !to hold time derivative values at each step
3
4  do l=h,6
5    do k=1,nbf
6      call allocbf(dbs_dt(h,k))
7    end do
8  end do
9
10 do k=1,nbf
11   dbs_dt(1,k)=dbs_dt1(k)    !first derivative step
12 end do                      !calculated in driving subroutine
13
14 do n=2,6                    !n=2-6 as n=1 already calculated
15   tempbs = bsin             !bsin is the existing wavefunction
16     do k=1,nbf              !b(n,h) is the values from the butcher tableau
17       do h=1,n-1
18         tempbs(k)%z(1:ndim)=tempbs(k)%z(1:ndim)&
19           +(b(n,h)*dt*dbs_dt(h,k)%z(1:ndim))
20         tempbs(k)%d_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
21           +(b(n,h)*dt*dbs_dt(h,k)%d_pes(1:npes))
22         tempbs(k)%s_pes(1:npes)=tempbs(k)%s_pes(1:npes)&
23           +(b(n,h)*dt*dbs_dt(h,k)%s_pes(1:npes))
24         tempbs(k)%a_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
25           *exp(i*tempbs(k)%s_pes(1:npes))
26         tempbs(k)%D_big=tempbs(k)%D_big+(b(n,h)*dt*dbs_dt(h,k)%D_big)
27       end do
28     end do
29
30   call deriv(tempbs, dbs_dt(n,:), n)
31
32   !the arguments for the derivs subroutine are (input, output, step index)
33
34   end do
35
36 tempbs = bsin
37
38 do k=1,nbf     !c(h) is the fourth-order values for the butcher tableau
39   do h=1,6
40     tempbs(k)%z(1:ndim)=tempbs(k)%z(1:ndim)&
41       +(c(h)*dt*dbs_dt(h,k)%z(1:ndim))
42     tempbs(k)%d_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
43       +(c(h)*dt*dbs_dt(h,k)%d_pes(1:npes))
44     tempbs(k)%s_pes(1:npes)=tempbs(k)%s_pes(1:npes)&
45       +(c(h)*dt*dbs_dt(h,k)%s_pes(1:npes))
46     tempbs(k)%a_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
47       *exp(i*tempbs(k)%s_pes(1:npes))
48     tempbs(k)%D_big=tempbs(k)%D_big+(c(h)*dt*dbs_dt(h,k)%D_big)
49   end do
50 end do
51
52 do k=1,nbf          !d(h) is the difference between the fourth- and
53   do h=1,6         !and fifth-order values from the butcher tableau
54     errbs(k)%z(1:ndim)=errbs(k)%z(1:ndim)&
55       +(d(h)*dbs_dt(h,k)%z(1:ndim))
56     errbs(k)%d_pes(1:npes)=errbs(k)%d_pes(1:npes)&
57       +(d(h)*dbs_dt(h,k)%d_pes(1:npes))
58     errbs(k)%s_pes(1:npes)=errbs(k)%s_pes(1:npes)&
59       +(d(h)*dbs_dt(h,k)%s_pes(1:npes))
60     errbs(k)%D_big=errbs(k)%D_big+(d(h)*dbs_dt(l,k)%D_big)
61   end do
62 end do
63
64 return
```

Listing A.11: Code for the Runge-Kutta-Cash-Karp numerical integration.

```
1
2  allocate (dbs_dt(4,nbf))
3  do h=1,4
4    do k=1,nbf
5      call allocbf(dbs_dt(h,k))
6    end do
7  end do
8
9  do k=1,nbf
10   dbs_dt(h,k)=dbs_dt1(k)   !first derivative step
11 end do                     !calculated in driving subroutine
12
13 do n=2,4                   !n=2-6 as n=1 already calculated
14   do k=1,nbf               !h=n-1 as only last value needed.
15     tempbs(k)%z(1:ndim)=bsin(k)%z(1:ndim)&
16       +(b(n)*dt*dbs_dt(n-1,k)%z(1:ndim))
17     tempbs(k)%d_pes(1:npes)=bsin(k)%d_pes(1:npes)&
18       +(b(n)*dt*dbs_dt(n-1,k)%d_pes(1:npes))
19     tempbs(k)%s_pes(1:npes)=bsin(k)%s_pes(1:npes)&
20       +(b(n)*dt*dbs_dt(n-1,k)%s_pes(1:npes))
21     tempbs(k)%a_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
22       *exp(i*tempbs(k)%s_pes(1:npes))
23     tempbs(k)%D_big=bsin(k)%D_big+(b(n)*dt*dbs_dt(n-1,k)%D_big)
24   end do
25   call deriv(tempbs, dbs_dt(n,:), n)
26   !the arguments for the derivs subroutine are (input, output, step index)
27 end do
28
29 tempbs=bsin
30
31 do k=1,nbf
32   do h=1,4
33     tempbs(k)%z(1:ndim)=tempbs(k)%z(1:ndim)&
34       +(c(h)*dt*dbs_dt(h,k)%z(1:ndim))
35     tempbs(k)%d_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
36       +(c(h)*dt*dbs_dt(h,k)%d_pes(1:npes))
37     tempbs(k)%s_pes(1:npes)=tempbs(k)%s_pes(1:npes)&
38       +(c(h)*dt*dbs_dt(h,k)%s_pes(1:npes))
39     tempbs(k)%a_pes(1:npes)=tempbs(k)%d_pes(1:npes)&
40       *exp(i*tempbs(k)%s_pes(1:npes))
41     tempbs(k)%D_big=tempbs(k)%D_big+(c(h)*dt*dbs_dt(h,k)%D_big)
42   end do
43 end do
44
45 dtfin = dt     !needed for compatibility with adaptive step system
46 dtout = dt
47
48 return
```

Listing A.12: Code for the Fourth-Order Runge-Kutta (RK4) numerical integration subroutine.

A.4 Parallelisation

A.4.1 Parallel Environments in High Performance Computing

For many applications, confining computation to a single processor can make execution times unwieldy, and as even modern PCs often contain many cores using only a single processor can be a waste of available resources, even for standard consumer software. The alternative is to construct code in such a way as to make use of multiple processors simultaneously. This is often achieved using one of two methods: the first is a shared

memory system such as used by OpenMP, the second is a distributed memory system such as MPI.

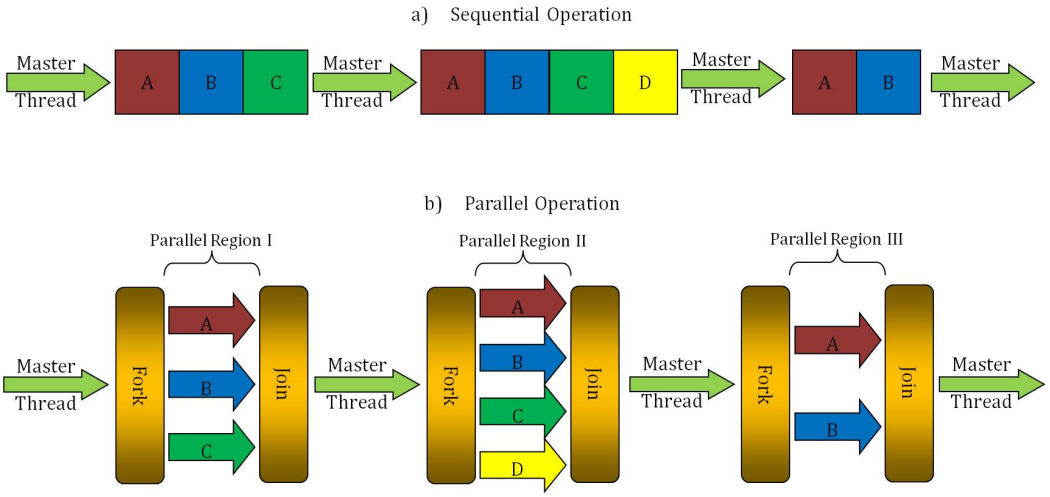


FIGURE A.15: Illustration of a sequential program (a) compared to a program using the the fork/join parallel function of OpenMP (b).

OpenMP, or Open Multi-Processing, is a multi-threading system which allows sections of a program to be run on multiple parallel threads simultaneously through the use of shared memory architecture.This is achieved by introducing parallel sections in a program where formerly serial sections are split by *forks*, executed and then recombined at *joins*, as illustrated by figure A.15.

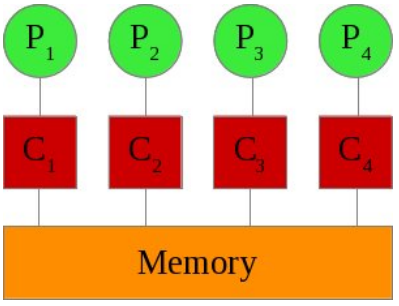


FIGURE A.16: Representation of the architecture of a single shared memory multiple processor node where P_x indicates a processor, and C_x the corresponding cache.

Each thread is executed independently of its fellows, meaning that no communication is needed between the threads. This means that sections of program which previously ran sequentially can be executed simultaneously, with the runtime environment assigning threads to different processors. The shared memory architecture used by OpenMP consists of multiple processors which are all connected to the same physical memory chip, as illustrated in figure A.16. This allows data from the same memory address to be copied into the cache for multiple processors and so used in different processing threads without the need for message passing. Combined with the independent nature of the

parallel threads in OpenMP, this results in a parallel environment with little overhead. This creates an upper limit on the number of simultaneous processes allowed however, which cannot exceed the number of logical processors sharing the memory chip, and as such purely OpenMP parallel implementations are unsuited for programs which require very many simultaneous calculations.

In cases where a larger system is required, one would instead turn to a distributed memory system such as MPI, or Message Passing Interface. This is a system whereby separate processes in a program communicate by means of explicit messages. Usually this requires a master process which broadcasts data to slave processes. These slave processes then carry out calculations, sometimes communicating between themselves, and produce a result. The "farming out" of the workload to the slave processes can be thought of in a similar manner to the *fork/join* model used by OpenMP, however there are differences in that the master process remains running, and often acts as a controller for the communication between the slave processes. One of the main benefits of this system is that the program is not limited to a single physical memory block. Instead, the local cache memory assigned to a single processor can exchange data with a memory address located in a physical memory block on a different node through the interconnect between nodes, and thereby local cache memories can communicate with each other. This can be more easily understood by considering figure A.17. This

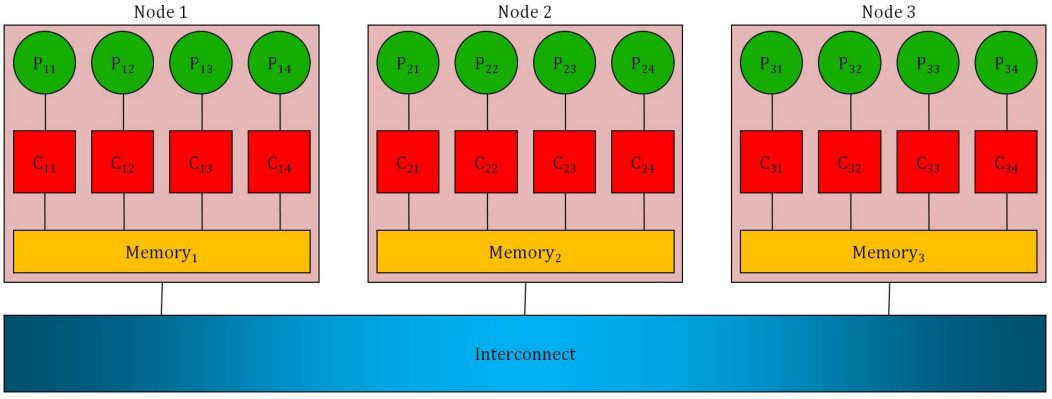


FIGURE A.17: Representation of the architecture of a cluster computer with multiple nodes, where P_{xx} indicates a processor, and C_{xx} indicates the cache assigned to the processor.

architecture is similar in structure to that of most available cluster computers, such as those used for the simulations of the spin boson model and for high harmonic generation in the preparation of this work. Here, data calculated by processor P_{13} can be passed to processor P_{34} through the interconnect by means of a message. As such there is no requirement that the processors used be on the same node - a single execution of the program requesting four processors could use processors P_{11}, P_{23}, P_{31} and P_{33} for

example. There is also no limitation in the number of processing nodes used, unlike OpenMP where the maximum number of slave processes is equal to the number of logical processors attached to a single memory chip.

A.4.2 Choosing a Parallel system

Running the MCE program in a serial fashion can have serious drawbacks when it comes to doing large simulations or large numbers of repeats. Using a parallel system is the logical alternative, spreading the workload over many processors. The structure of the program does not lend itself easily to fine-grain parallelisation. It is well known that recursive integration systems such as Runge-Kutta cannot be parallelised in any meaningful way since each part of the calculation depends on the previous and so independent threads or processes cannot be constructed, and as the majority of matrix operations use Level 2 BLAS (matrix-vector) operations there is an impediment to parallelisation here also, as these operations show little improvement using the PBLAS system except for very large matrices. As a consequence, coarse-grain parallelisation is the best option, and as such running repeat calculations from within the program and executing these repeat calculations in parallel should significantly improve the running time of the program.

While the use of MPI would allow very many processors to be used simultaneously, the addition of MPI directives to a serial program is difficult and very time consuming. In contrast adding OpenMP directives to a serial program is relatively simple, and requires little rewriting of the existing code. OpenMP limits the number of simultaneous threads that can be used effectively depending on available hardware, and so a combination of OpenMP and a shell script which splits the repeat running of the simulation into many separate instances of the program would be an acceptable compromise which would increase the available number of cores for a single simulation with many repeat calculations. In the future however, if very large systems were needed, for example a simulation which uses many thousands of basis functions to represent the wavefunction, it is possible that the inclusion of MPI may become necessary.

A.4.3 Implementation of Parallel coding

As mentioned previously, two levels of load splitting are added into the running of the program. Firstly, separate instances of the program are created in a group of individual folders, run, and then a separate program gathers, combines and averages the output

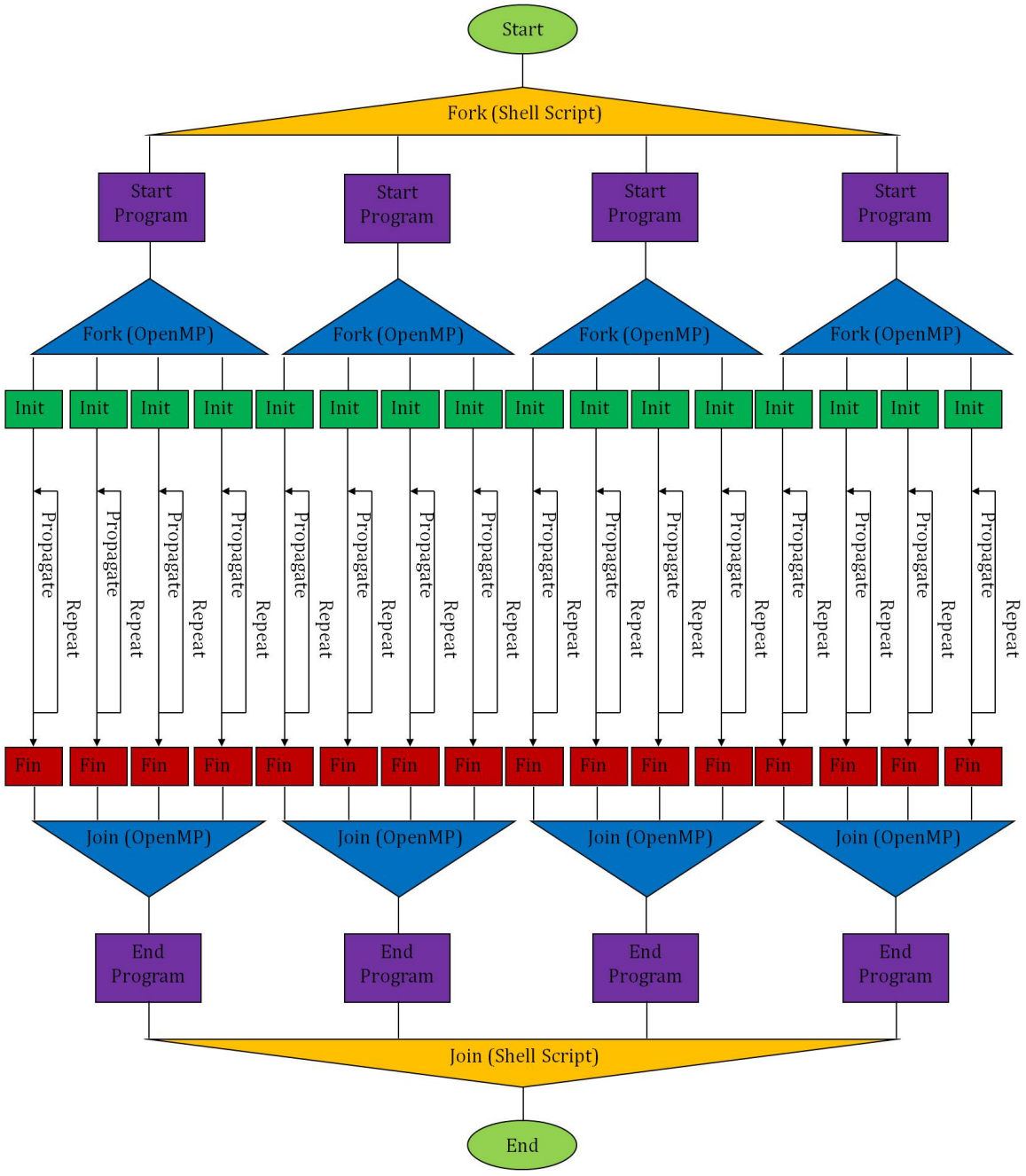


FIGURE A.18: Flow chart showing the splitting of the operation of the program into OpenMP threads and separate executions.

data from all instances of the program.

The second layer of load splitting occurs inside the program by way of OpenMP directives. The combination of these two levels of load splitting is illustrated in figure A.18, wherein it can be seen that after the shell script forking described above, the program begins, input parameters are read and initialised, and then the program passes through a second forking, labelled "Fork (OpenMP)". Once this fork has split the program load,

the program initialises the different threads (labelled "Init") before beginning a process of "generate basis set, propagate, repeat" until the required number of repeat calculations are completed. Following this, the outputs from the threads are finalised and averaged before being combined at the OpenMP join, after which the combined output files are written and the program ends. Once the program has ended, a second join is undertaken wherein the combining and averaging of data detailed above is undertaken.

OpenMP works through the use of *directives* - preprocessor commands which instruct the compiler to interpret the following code in a certain way. These directives are compiler extensions, and as such are language dependent, meaning that the directives used for C or C++ are different to those used for Fortran. In Fortran, these directives are always preceded by the flag `!$omp`. A parallel section of code is enclosed between the directives `!$omp parallel` and `!$omp end parallel`, often with a list of variables to be kept private, or particular to each individual thread, accompanying the starting command. It is important to include this list as, due to the shared memory nature of OpenMP programs, any variables not explicitly set to be private will be shared amongst all threads meaning that errors due to two threads overwriting the same value could be introduced. The list of private variables for this program is extensive as the majority of the main operation of the program exists within the parallel section of code. Included in this list are all the output parameters (norm, autocorrelation function, populations etc.), the basis set defined type variable, the time and various parameters and indices.

Often a parallel section in OpenMP enabled code will contain a loop, with the individual results from that loop combined after completion. This is the case for this program, with arrays of the output parameters combined through the use of the `reduction` directive, which is given at the begining of the `!$omp do` loop, as shown in listing A.13. A

```
1 !$omp do reduction (+:pops,absnorm,absnorm2,absehr,acf_t, extra)
2 ⋮
3 !$omp end do
```

Listing A.13: Initialisation of the parallel loop with declaration of the variables to be reduced.

significant consideration for parallel computing occurs when attempting to write to a file or variable which needs to be accessible from all threads. In these cases it is necessary to block other threads from accessing the memory address or file while one thread writes to it, so as to limit conflicts between the threads. To this end, two directives are used : `!$omp critical` which stops all threads executing a block of code effectively serializing this section, and `!$omp atomic` which limits access to a particular line of code as it

accesses a given memory address. The `!$omp critical` environment is very general and is often used for writing to files or for accessing third party subroutines which may not be thread safe. The `!$omp atomic`, being more specific, has much lower overhead cost and can only be applied to a single instruction. In the implementation of the parallelisation, both of these directives are used. The `critical` directive is used to call subroutines which use the UCL random number generator (see section A.2 for a discussion of the random number generator), which is not thread safe due to the method of acquiring the seed values, and is also used to write timestep information to a file when the adaptive timestep functionality is used. These are shown in listing A.14. The `atomic` directive

```
1
2  !$omp critical            ! Critical block stops subroutine running
3                            ! on multiple threads simultaneously
4  call genzinit(mup, muq)   ! This is needed as the UCL random library
5                            ! is not thread safe
6  !$omp end critical
7
8  ⋮
9
10 !$omp critical            ! again, the random number generator called in
11                           ! this sub requires the critical block
12 call genbasis(bset, mup, muq, alcmprss, gridsp, time, initgrid)
13
14 !$omp end critical
15 ⋮
16 !$omp critical
17
18 write(1710,"(e12.5)") dtdone  ! Outputs the stepsizes of all repetitions
19
20 !$omp end critical
```

Listing A.14: Usage of the critical directive to prevent access conflicts.

is also used when incrementing the counters tracking when the time step size has been changed for adaptive step size functionality. As these counters are incremented in all threads and over all repeats of the program, this becomes necessary to prevent memory access conflicts. By the use of these directives, the running of repeat calculations in the

```
1
2  if (dtdone.eq.dt) then
3
4  !$omp atomic              ! nsame and nchange are used to keep track of
5  nsame = nsame + 1         ! changes to the stepsize. The atomic directive
6  else                      ! is used to ensure two threads do not write to
7  !$omp atomic              ! the same memory address simultaneously as
8  nchange = nchange + 1     ! these counts are taken over all repeats.
9
10 end if
```

Listing A.15: Usage of the atomic directive to prevent memory access conflicts.

program is successfully parallelised without large overheads and without major changes to the serial program, providing a significant speedup compared to the serial code.

Appendix B

Derivations of Equations for the Coupled Coherent States and Multi-Configurational Ehrenfest methods

In this appendix we will endeavour to present a more complete derivation of the equations used in chapters 3 and 4. The time derivative equations derived in sections B.2-B.4 are found through a combination of the time dependent variational principle (TDVP) of Kramer and Saraceno[79], the time dependent Schrödinger equation, and classical Hamiltons equations. To begin with the equations of the CCS method will be derived, and many of the results of this derivation will be subsequently used for the derivation of the equations for the MCE methods. Following this the Hamiltonian and the acceleration form of the dipole moment are derived for the high harmonic generation system. For the purposes of all the following derivations, $\hbar = 1$.

B.1 Properties of the Coherent States

The derivations below make use of various properties of coherent states. The first is that the overlap of two coherent states, being non-orthogonal, is non-zero and can be found by

$$\langle \alpha | \, \beta \rangle = \exp\left[\alpha^* \beta - \frac{\alpha^* \alpha}{2} - \frac{\beta^* \beta}{2}\right]. \tag{B.1.1}$$

This is well known (see for example [40, 41] or [37]) and need not be confirmed here.

A second property is that of the overlap between a coherent state and the time derivative of a coherent state, which is given as

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \left(\alpha^* \dot{\beta} - -\frac{\dot{\beta}^* \beta}{2} - \frac{\beta^* \dot{\beta}}{2} \right) \langle \alpha | \beta \rangle . \tag{B.1.2}$$

This may need some confirmation which is done via

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \lim_{t \to 0} \frac{\langle \alpha | \beta + \Delta\beta \rangle - \langle \alpha | \beta \rangle}{\Delta t}. \tag{B.1.3}$$

If equation (B.1.1) is used here, this becomes

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \lim_{t \to 0} \frac{1}{\Delta t} \left(\exp \left[\alpha^* \beta + \alpha^* \Delta\beta - \frac{\alpha^* \alpha}{2} - \frac{\beta^* \beta}{2} - \frac{\beta^* \Delta\beta}{2} - \frac{\Delta\beta^* \beta}{2} - \frac{\Delta\beta^* \Delta\beta}{2} \right] - \langle \alpha | \beta \rangle \right). \tag{B.1.4}$$

By taking out the terms which together equal $\langle \alpha | \beta \rangle$ this becomes

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \lim_{t \to 0} \frac{1}{\Delta t} \left(\exp \left[\alpha^* \Delta\beta - \frac{\beta^* \Delta\beta}{2} - \frac{\Delta\beta^* \beta}{2} - \frac{\Delta\beta^* \Delta\beta}{2} \right] - 1 \right) \langle \alpha | \beta \rangle , \tag{B.1.5}$$

which upon taking a first order Taylor expansion and neglecting $\mathcal{O}(\Delta\beta^2)$ terms becomes

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \lim_{t \to 0} \frac{1}{\Delta t} \left(1 + \alpha^* \Delta\beta - \frac{\beta^* \Delta\beta}{2} - \frac{\Delta\beta^* \beta}{2} - 1 \right) \langle \alpha | \beta \rangle . \tag{B.1.6}$$

Taking the limit $\lim_{t \to 0}$ then finally returns

$$\left\langle \alpha \middle| \dot{\beta} \right\rangle = \left(\alpha^* \dot{\beta} - \frac{\dot{\beta}^* \beta}{2} - \frac{\beta^* \dot{\beta}}{2} \right) \langle \alpha | \beta \rangle . \tag{B.1.7}$$

A third property is that of the time derivative of the overlap, which follows from the previous, such that

$$\begin{aligned}
\frac{d \langle \alpha | \beta \rangle}{dt} &= \left\langle \dot{\alpha} \middle| \beta \right\rangle + \left\langle \alpha \middle| \dot{\beta} \right\rangle \\
&= \left[\left(\dot{\alpha}^* \beta - \frac{\dot{\alpha}^* \alpha}{2} - \frac{\alpha^* \dot{\alpha}}{2} \right) + \left(\alpha^* \dot{\beta} - \frac{\dot{\beta}^* \beta}{2} - \frac{\beta^* \dot{\beta}}{2} \right) \right] \langle \alpha | \beta \rangle \\
&= \left[\left(\left(\beta - \frac{\alpha}{2} \right) \dot{\alpha}^* - \frac{\alpha^* \dot{\alpha}}{2} \right) + \left(\left(\alpha^* - \frac{\beta^*}{2} \right) \dot{\beta} - \frac{\dot{\beta}^* \beta}{2} \right) \right] \langle \alpha | \beta \rangle \\
&= \left[\left((\alpha^* - \beta^*) \dot{\beta} - \frac{\dot{\alpha}^* \alpha}{2} + \frac{\alpha^* \dot{\alpha}}{2} \right) + \left((\beta - \alpha) \dot{\alpha}^* - \frac{\beta^* \dot{\beta}}{2} + \frac{\dot{\beta}^* \beta}{2} \right) \right] \langle \alpha | \beta \rangle .
\end{aligned} \tag{B.1.8}$$

B.1.1 Conversion into z-notation

The reformulating of a Hamiltonian into the coherent states z-notation is often not trivial. In some cases one can merely substitute Eq(2.3.5). In many other cases, to find the z-notation version of an operator in M dimensions,

$$\left\langle \mathbf{z}_j \right| \hat{O}(\mathbf{x}) \left| \mathbf{z}_k \right\rangle, \tag{B.1.9}$$

one must use the identity

$$\mathbb{I} = \int_{-\infty}^{\infty} |\mathbf{x}\rangle \langle \mathbf{x}| \, d^M\mathbf{x}, \tag{B.1.10}$$

such that

$$\left\langle \mathbf{z}_j \right| \hat{O}(\mathbf{x}) \left| \mathbf{z}_k \right\rangle = \int_{-\infty}^{\infty} \left\langle \mathbf{z}_j \right| \mathbf{x} \left\rangle \right\langle \mathbf{x} \left| \hat{O} \right| \mathbf{z}_k \right\rangle d^M\mathbf{x} = \int_{-\infty}^{\infty} \langle \mathbf{z}_j | \, \mathbf{x} \rangle \langle \mathbf{x} | \, \mathbf{z}_k \rangle \hat{O}(\mathbf{x}) d^M\mathbf{x}. \tag{B.1.11}$$

As such, it is necessary to have an expression for $\langle \mathbf{z}_j | \, \mathbf{x} \rangle \langle \mathbf{x} | \, \mathbf{z}_k \rangle$. Starting with equation (2.3.2) and its conjugate,

$$\begin{aligned} \langle \mathbf{z}_j | \, \mathbf{x} \rangle \langle \mathbf{x} | \, \mathbf{z}_k \rangle = \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\Big[& -\frac{\gamma}{2}\left((\mathbf{x}-\mathbf{q}_k)^2 + (\mathbf{x}-\mathbf{q}_j)^2 \right) \\ & + i\left(\mathbf{p}_k (\mathbf{x}-\mathbf{q}_k) - \mathbf{p}_j (\mathbf{x}-\mathbf{q}_j) \right) + \frac{i\mathbf{p}_k\mathbf{q}_k}{2} - \frac{i\mathbf{p}_j\mathbf{q}_j}{2} \Big]. \end{aligned} \tag{B.1.12}$$

This can be rearranged such that

$$\begin{aligned} \langle \mathbf{z}_j | \, \mathbf{x} \rangle \langle \mathbf{x} | \, \mathbf{z}_k \rangle &= \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\Big[-\frac{\gamma}{2}\left((\mathbf{x}-\mathbf{q}_k)^2 + (\mathbf{x}-\mathbf{q}_j)^2 \right) + i\mathbf{x}(\mathbf{p}_k - \mathbf{p}_j) \\ &\qquad - \frac{i}{2}(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j) \Big] \\ &= \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\Big[-\frac{\gamma}{2}\left(2\mathbf{x}^2 - 2\mathbf{x}(\mathbf{q}_k + \mathbf{q}_j) + \mathbf{q}_k^2 + \mathbf{q}_j^2 \right) + i\mathbf{x}(\mathbf{p}_k - \mathbf{p}_j) \\ &\qquad - \frac{i}{2}(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j) \Big] \\ &= \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\Big[-\gamma\left(\mathbf{x}^2 - \mathbf{x}(\mathbf{q}_k + \mathbf{q}_j) - \frac{i}{\gamma}\mathbf{x}(\mathbf{p}_k - \mathbf{p}_j) \right) - \frac{\gamma}{2}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2\right) \\ &\qquad - \frac{i}{2}(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j) \Big]. \end{aligned} \tag{B.1.13}$$

If we then introduce

$$\boldsymbol{\rho}_{jk} = \frac{\mathbf{z}_j^* + \mathbf{z}_k}{\sqrt{2\gamma}} = \frac{1}{2}\left[(\mathbf{q}_k + \mathbf{q}_j) + \frac{i}{\gamma}(\mathbf{p}_k - \mathbf{p}_j) \right], \tag{B.1.14}$$

this then becomes

$$\begin{aligned}\langle \mathbf{z}_j|\,\mathbf{x}\rangle\,\langle \mathbf{x}|\,\mathbf{z}_k\rangle &= \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\left[-\gamma\left(\mathbf{x}^2 - 2\mathbf{x}\boldsymbol{\rho}_{jk}\right) - \frac{\gamma}{2}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2\right) - \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j\right)\right] \\ &= \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\left[-\gamma\left(\mathbf{x} - \boldsymbol{\rho}_{jk}\right)^2 + \gamma\boldsymbol{\rho}_{jk} - \frac{\gamma}{2}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2\right) - \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j\right)\right].\end{aligned} \tag{B.1.15}$$

By substitution of

$$\begin{aligned}\gamma\boldsymbol{\rho}_{jk}^2 =& \frac{\gamma}{4}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2 + 2\mathbf{q}_k\mathbf{q}_j\right) - \frac{1}{4\gamma}\left(\mathbf{p}_k^2 + \mathbf{p}_j^2 - 2\mathbf{p}_k\mathbf{p}_j\right) \\ &+ \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_k - \mathbf{p}_j\mathbf{q}_j\right) + \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_j - \mathbf{p}_j\mathbf{q}_k\right),\end{aligned} \tag{B.1.16}$$

this can be rewritten as

$$\begin{aligned}\langle \mathbf{z}_j|\,\mathbf{x}\rangle\,\langle \mathbf{x}|\,\mathbf{z}_k\rangle = \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\Big[&-\gamma\left(\mathbf{x} - \boldsymbol{\rho}_{jk}\right)^2 + \frac{\gamma}{2}\left(\mathbf{q}_k\mathbf{q}_j\right) - \frac{\gamma}{4}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2\right) \\ &+ \frac{1}{2\gamma}\left(\mathbf{p}_k\mathbf{p}_j\right) - \frac{1}{4\gamma}\left(\mathbf{p}_k^2 + \mathbf{p}_j^2\right) + \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_j - \mathbf{p}_j\mathbf{q}_k\right)\Big].\end{aligned} \tag{B.1.17}$$

By considering the coherent state overlap

$$\begin{aligned}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle &= \exp\left(\mathbf{z}_j^*\mathbf{z}_k - \frac{|\mathbf{z}_j|^2}{2} - \frac{|\mathbf{z}_k|^2}{2}\right) \\ &= \exp\left[\frac{\gamma}{2}\left(\mathbf{q}_k\mathbf{q}_j\right) + \frac{1}{2\gamma}\left(\mathbf{p}_k\mathbf{p}_j\right) + \frac{i}{2}\left(\mathbf{p}_k\mathbf{q}_j - \mathbf{p}_j\mathbf{q}_k\right)\right. \\ &\qquad \left. -\frac{\gamma}{4}\left(\mathbf{q}_k^2 + \mathbf{q}_j^2\right) - \frac{1}{4\gamma}\left(\mathbf{p}_k^2 + \mathbf{p}_j^2\right)\right],\end{aligned} \tag{B.1.18}$$

we can finally get

$$\langle \mathbf{z}_j|\,\mathbf{x}\rangle\,\langle \mathbf{x}|\,\mathbf{z}_k\rangle = \left(\frac{\gamma}{\pi}\right)^{\frac{M}{2}} \exp\left[-\gamma\left(\mathbf{x} - \boldsymbol{\rho}_{jk}\right)^2\right]\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle\,. \tag{B.1.19}$$

B.2 Derivation of the Equations for the Coupled Coherent States method

The CCS method uses the variational principle to find the time derivative equation for the amplitudes. Starting from the ansatz

$$|\Psi\rangle = \sum_k A_k\,|\mathbf{z}_k\rangle\,, \tag{B.2.1}$$

we construct the Lagrangian

$$\mathcal{L} = \left\langle \Psi \middle| i \overleftrightarrow{\partial}_t - \hat{H} \middle| \Psi \right\rangle = \frac{i}{2}\left[\left\langle \Psi \middle| \dot{\Psi} \right\rangle - \left\langle \dot{\Psi} \middle| \Psi \right\rangle\right] - \left\langle \Psi \middle| \hat{H} \middle| \Psi \right\rangle, \tag{B.2.2}$$

where the differential operator is defined as

$$\overleftrightarrow{\partial}_t = \frac{1}{2}\left(\overrightarrow{\partial}_t - \overleftarrow{\partial}_t\right) \text{ where } \overrightarrow{\partial}_t |\cdots\rangle, \ \langle\cdots| \overleftarrow{\partial}_t. \tag{B.2.3}$$

In equation (B.2.2), the term $\left\langle \Psi \middle| \dot{\Psi} \right\rangle$ is found as

$$\left\langle \Psi \middle| \dot{\Psi} \right\rangle = \frac{i}{2}\sum_{jk}\left[A_j^* \langle \mathbf{z}_j | \mathbf{z}_k\rangle \dot{A}_k + A_j^* \langle \mathbf{z}_j | \dot{\mathbf{z}}_k\rangle A_k\right], \tag{B.2.4}$$

and so, in light of equation (B.1.2) the Lagrangian becomes

$$\begin{aligned}\mathcal{L} =& \frac{i}{2}\sum_{jk}\left(A_j^*\dot{A}_k - \dot{A}_j^* A_k\right)\langle \mathbf{z}_j | \mathbf{z}_k\rangle + \\ & \frac{i}{2}\sum_{jk} A_j^* A_k\left[\left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}} *_k \mathbf{z}_k}{2}\right) - \left(\dot{\mathbf{z}}_j^*\mathbf{z}_k - \frac{\mathbf{z}_j^*\dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}} *_j \mathbf{z}_j}{2}\right)\right]\langle \mathbf{z}_j | \mathbf{z}_k\rangle - \\ & \sum_{jk} A_j^* A_k H_{\text{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right)\langle \mathbf{z}_j | \mathbf{z}_k\rangle .\end{aligned} \tag{B.2.5}$$

From this Lagrangian, the Euler-Lagrange equations can be determined, and by obtaining the Euler-Lagrange equation for A^* we can find the time derivative equation for $\dot{A}$. This is given as

$$\frac{\partial \mathcal{L}}{\partial A_j^*} - \frac{\partial}{\partial t}\frac{\partial \mathcal{L}}{\partial \dot{A}_j^*} = 0. \tag{B.2.6}$$

The first term can be found as

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial A_j^*} =& \frac{i}{2}\sum_k \Big[\dot{A}_k + A_k\left(2iH_{\text{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) + \mathbf{z}_j^*\dot{\mathbf{z}}_k - \dot{\mathbf{z}}_j^*\mathbf{z}_k\right) \\ & -\frac{1}{2}\left(\mathbf{z}_k^*\dot{\mathbf{z}}_k + \dot{\mathbf{z}}_k^*\mathbf{z}_k - \mathbf{z}_j^*\dot{\mathbf{z}}_j - \dot{\mathbf{z}}_j^*\mathbf{z}_j\right)\Big]\langle \mathbf{z}_j | \mathbf{z}_k\rangle ,\end{aligned} \tag{B.2.7}$$

and the second as

$$\begin{aligned}-\frac{\partial}{\partial t}\frac{\partial \mathcal{L}}{\partial \dot{A}^*} &= \frac{\partial}{\partial t}\frac{i}{2}\sum_k A_k\langle \mathbf{z}_j | \mathbf{z}_k\rangle \\ &= \frac{i}{2}\sum_k \Big[\dot{A}_k + A_k\left(2iH_{\text{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) + \mathbf{z}_j^*\dot{\mathbf{z}}_k + \dot{\mathbf{z}}_j^*\mathbf{z}_k\right) \\ &\quad -\frac{1}{2}\left(\mathbf{z}_k^*\dot{\mathbf{z}}_k + \dot{\mathbf{z}}_k^*\mathbf{z}_k + \mathbf{z}_j^*\dot{\mathbf{z}}_j + \dot{\mathbf{z}}_j^*\mathbf{z}_j\right)\Big]\langle \mathbf{z}_j | \mathbf{z}_k\rangle .\end{aligned} \tag{B.2.8}$$

This finally gives

$$\begin{aligned} &i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, \dot{A}_k - \sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, A_k H_{\mathrm{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) \\ &+ i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, A_k \left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right) = 0, \end{aligned} \tag{B.2.9}$$

which, from comparison with equation (B.1.2) can be seen as being equivalent to the Schrödinger equation. If we now make the substitution

$$A_k = D_k e^{iS_k}, \tag{B.2.10}$$

where S_k is the action, given by

$$S = \int \left[\frac{i}{2}\left(\mathbf{z}^*\frac{d\mathbf{z}}{dt} - \mathbf{z}\frac{d\mathbf{z}^*}{dt}\right) - H_{\mathrm{ord}}(\mathbf{z}^*, \mathbf{z})\right] \mathrm{d}t, \tag{B.2.11}$$

we obtain the equation

$$\begin{aligned} &i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, \dot{D}_k e^{iS_k} - \sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, D_k e^{iS_k}\left[i\left(\frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\mathbf{z}_k\dot{\mathbf{z}}_k^*}{2}\right) - H_{\mathrm{ord}}(\mathbf{z}_k^*, \mathbf{z}_k)\right] \\ &- \sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, D_k e^{iS_k} H_{\mathrm{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) \\ &+ i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, D_k e^{iS_k}\left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right) = 0, \end{aligned} \tag{B.2.12}$$

which then becomes

$$i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, \dot{D}_k e^{iS_k} - \sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\left[H_{\mathrm{ord}}(\mathbf{z}_j^*, \mathbf{z}_k) - H_{\mathrm{ord}}(\mathbf{z}_k^*, \mathbf{z}_k) - i\left(\mathbf{z}_j^* - \mathbf{z}_k^*\right)\dot{\mathbf{z}}_k\right] D_k e^{iS_k} = 0, \tag{B.2.13}$$

and so

$$\sum_i \langle \mathbf{z}_j|\ \mathbf{z}_i\rangle\, \frac{\partial D_i}{\partial t} = -i\sum_k \langle \mathbf{z}_j|\ \mathbf{z}_k\rangle\, \delta^2 H'_{\mathrm{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) D_k e^{i(S_k - S_i)} \tag{B.2.14}$$

where

$$\delta^2 H'_{\mathrm{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) = \left[H_{\mathrm{ord}}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) - H_{\mathrm{ord}}\left(\mathbf{z}_k^*, \mathbf{z}_k\right) - i\left(\mathbf{z}_j^* - \mathbf{z}_k^*\right)\dot{\mathbf{z}}_k\right]. \tag{B.2.15}$$

The time derivative equations for $\dot{\mathbf{z}}$ can also be found through the variational principle using the Euler-Lagrange equations for $\mathbf{z}^*$, however as the centre of a coherent state moves classically through phase space the Hamilton's equations can be used, and indeed if variational calculations are performed this is the result. As such, in the coherent state

z notation,

$$i\dot{\mathbf{z}}_k = \frac{\partial H_{\text{ord}}\left(\mathbf{z}_k^*, \mathbf{z}_k\right)}{\partial \mathbf{z}^*}. \tag{B.2.16}$$

B.3 Derivation of the Equations for the First Formulation of the Multiconfigurational Ehrenfest Method

The time derivative equations for the MCEv1 method are found in a similar fashion to those of the CCS method. Here we present the derivation for two electronic states, but the equations can be easily expanded to more states. We use a Hamiltonian of the form

$$\left\langle \mathbf{z} \right| \hat{H} \left| \mathbf{z} \right\rangle = \begin{vmatrix} H_{\text{ord}}^{(11)}\left(\mathbf{z}^*, \mathbf{z}\right) & H_{\text{ord}}^{(12)}\left(\mathbf{z}^*, \mathbf{z}\right) \\ H_{\text{ord}}^{(21)}\left(\mathbf{z}^*, \mathbf{z}\right) & H_{\text{ord}}^{(22)}\left(\mathbf{z}^*, \mathbf{z}\right) \end{vmatrix}. \tag{B.3.1}$$

Starting with the ansatz

$$\begin{aligned} |\Psi\rangle &= \sum_k |\varphi_k\rangle \\ &= \sum_k \left(a_k^{(1)} |1\rangle + a_k^{(2)} |2\rangle \right) |\mathbf{z}_k\rangle , \end{aligned} \tag{B.3.2}$$

to find the time derivative equation for the coherent state basis we first construct the single configuration Lagrangian

$$\begin{aligned} \mathcal{L}_{sc} &= \left\langle \varphi \right| i \overleftrightarrow{\partial}_t - \hat{H} \left| \varphi \right\rangle \\ &= i \left[\frac{a^{(1)*}\dot{a}^{(1)}}{2} - \frac{\dot{a}^{(1)*}a^{(1)}}{2} \right] + i \left[\frac{a^{(2)*}\dot{a}^{(2)}}{2} - \frac{\dot{a}^{(2)*}a^{(2)}}{2} \right] \\ &\quad + i \left[\left(\frac{\mathbf{z}^*\dot{\mathbf{z}}}{2} - \frac{\dot{\mathbf{z}}^*\mathbf{z}}{2} \right) \left(a^{(1)*}a^{(1)} + a^{(2)*}a^{(2)} \right) \right] \\ &\quad - H_{\text{ord}}^{(11)}\left(\mathbf{z}^*, \mathbf{z}\right) a^{(1)*}a^{(1)} - H_{\text{ord}}^{(22)}\left(\mathbf{z}^*, \mathbf{z}\right) a^{(2)*}a^{(2)} \\ &\quad - H_{\text{ord}}^{(12)}\left(\mathbf{z}^*, \mathbf{z}\right) a^{(1)*}a^{(2)} - H_{\text{ord}}^{(21)}\left(\mathbf{z}^*, \mathbf{z}\right) a^{(2)*}a^{(1)}. \end{aligned} \tag{B.3.3}$$

The Euler-Lagrange equation for $\mathbf{z}^*$ can then be found as

$$\begin{aligned} \frac{\partial \mathcal{L}_{sc}}{\partial \mathbf{z}^*} - \frac{\partial}{\partial t}\frac{\partial \mathcal{L}_{sc}}{\partial \dot{\mathbf{z}}^*} =& i\frac{\dot{\mathbf{z}}}{2}\left(a^{(1)*}a^{(1)} + a^{(2)*}a^{(2)} \right) \\ &- \frac{\partial H_{\text{ord}}^{(11)}\left(\mathbf{z}^*, \mathbf{z}\right)}{\partial \mathbf{z}^*} a^{(1)*}a^{(1)} - \frac{\partial H_{\text{ord}}^{(22)}\left(\mathbf{z}^*, \mathbf{z}\right)}{\partial \mathbf{z}^*} a^{(2)*}a^{(2)} \\ &- \frac{\partial H_{\text{ord}}^{(12)}\left(\mathbf{z}^*, \mathbf{z}\right)}{\partial \mathbf{z}^*} a^{(1)*}a^{(2)} - \frac{\partial H_{\text{ord}}^{(21)}\left(\mathbf{z}^*, \mathbf{z}\right)}{\partial \mathbf{z}^*} a^{(2)*}a^{(1)} \\ &+ i\frac{\mathbf{z}}{2}\frac{d\left(a^{(1)*}a^{(1)} + a^{(2)*}a^{(2)} \right)}{dt} + i\frac{\dot{\mathbf{z}}}{2}\left(a^{(1)*}a^{(1)} + a^{(2)*}a^{(2)} \right) = 0. \end{aligned} \tag{B.3.4}$$

By introducing the Ehrenfest Hamiltonian

$$H^{\mathrm{Ehr}} = \frac{\sum_{rs}\left(H^{(rs)}_{\mathrm{ord}}\left(\mathbf{z}^*,\mathbf{z}\right)a^{(r)*}a^{(s)}\right)}{\sum_{r}\left(a^{(r)*}a^{(r)}\right)}, \tag{B.3.5}$$

where in this case $r,s \in \{1,2\}$, we can then say

$$i\dot{\mathbf{z}} = \frac{\partial H^{\mathrm{Ehr}}}{\partial \mathbf{z}^*} + i\frac{\mathbf{z}}{2}\frac{d\,\ln\left(a^{(1)*}a^{(1)} + a^{(2)*}a^{(2)}\right)}{dt}, \tag{B.3.6}$$

and since the second term here is equal to zero for a single configuration, this becomes

$$i\dot{\mathbf{z}} = \frac{\partial H^{\mathrm{Ehr}}}{\partial \mathbf{z}^*} \tag{B.3.7}$$

which, as with the CCS method, is equivalent to the Hamiltons equations in z notation.

To find the time derivative equations for the amplitudes we use the multi-configurational Lagrangian

$$\begin{aligned}
\mathcal{L} &= \left\langle \Psi \middle| \, i\overleftrightarrow{\partial}_t - \hat{H} \, \middle| \Psi \right\rangle \\
&= \sum_{jk}\frac{i}{2}\left[a_j^{(1)*}\dot{a}_k^{(1)} - \dot{a}_j^{(1)*}a_k^{(1)}\right]\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle + \sum_{jk}\frac{i}{2}\left[a_j^{(2)*}\dot{a}_k^{(2)} - \dot{a}_j^{(2)*}a_k^{(2)}\right]\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad + \sum_{jk}\frac{i}{2}\left[\left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right) - \left(\dot{\mathbf{z}}_j^*\mathbf{z}_k - \frac{\mathbf{z}_j^*\dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}}_j^*\mathbf{z}_j}{2}\right)\right]a_j^{(1)*}a_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad + \sum_{jk}\frac{i}{2}\left[\left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right) - \left(\dot{\mathbf{z}}_j^*\mathbf{z}_k - \frac{\mathbf{z}_j^*\dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}}_j^*\mathbf{z}_j}{2}\right)\right]a_j^{(2)*}a_k^{(2)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad - \sum_{jk}H^{(11)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_j^{(1)*}a_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle - \sum_{jk}H^{(22)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_j^{(2)*}a_k^{(2)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad - \sum_{jk}H^{(12)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_j^{(1)*}a_k^{(2)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle - \sum_{jk}H^{(21)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_j^{(2)*}a_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle,
\end{aligned} \tag{B.3.8}$$

to find the Euler-Lagrange equation for $a_j^{(1)*}$,

$$\begin{aligned}
\frac{\partial\mathcal{L}}{\partial a_j^{(1)*}} - \frac{\partial}{\partial t}\frac{\partial\mathcal{L}}{\partial \dot{a}_j^{(1)*}} &= \frac{i}{2}\sum_k \dot{a}_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad + \frac{i}{2}\sum_k\left[\left(\mathbf{z}_j^*\dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right) - \left(\dot{\mathbf{z}}_j^*\mathbf{z}_k - \frac{\mathbf{z}_j^*\dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}}_j^*\mathbf{z}_j}{2}\right)\right]a_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad - \sum_k H^{(11)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle - \sum_k H^{(12)}_{\mathrm{ord}}\left(\mathbf{z}_j^*,\mathbf{z}_k\right)a_k^{(2)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle \\
&\quad + \frac{i}{2}\sum_k a_k^{(1)}\frac{d\,\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle}{dt} + \frac{i}{2}\sum_k \dot{a}_k^{(1)}\langle \mathbf{z}_j|\,\mathbf{z}_k\rangle = 0.
\end{aligned} \tag{B.3.9}$$

This can be rewritten, thanks to equation B.1.8, as

$$\begin{aligned}
&\frac{\partial \mathcal{L}}{\partial a_j^{(1)*}} - \frac{\partial}{\partial t}\frac{\partial \mathcal{L}}{\partial \dot{a}_j^{(1)*}} = i\sum_k \dot{a}_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \\
&+ \frac{i}{2}\sum_k \left[\left(\mathbf{z}_j^* \dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right) - \left(\dot{\mathbf{z}}_j^* \mathbf{z}_k - \frac{\mathbf{z}_j^* \dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}}_j^* \mathbf{z}_j}{2} \right) \right] a_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \\
&+ \frac{i}{2}\sum_k \left[\left(\mathbf{z}_j^* \dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right) + \left(\dot{\mathbf{z}}_j^* \mathbf{z}_k - \frac{\mathbf{z}_j^* \dot{\mathbf{z}}_j}{2} - \frac{\dot{\mathbf{z}}_j^* \mathbf{z}_j}{2} \right) \right] a_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \\
&- \sum_k H_{\mathrm{ord}}^{(11)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle - \sum_k H_{\mathrm{ord}}^{(12)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(2)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle = 0,
\end{aligned} \tag{B.3.10}$$

and so

$$\begin{aligned}
&i\sum_k \dot{a}_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle + i\sum_k \left[\mathbf{z}_j^* \dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right] a_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \\
&- \sum_k H_{\mathrm{ord}}^{(11)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(1)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle - \sum_k H_{\mathrm{ord}}^{(12)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) a_k^{(2)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle = 0.
\end{aligned} \tag{B.3.11}$$

Introducing the substitution that

$$a_k^{(r)} = d_k^{(r)} e^{iS_k^{(r)}}, \tag{B.3.12}$$

where

$$S_k^{(r)} = \int \left[i\frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k - \dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} - H_{\mathrm{ord}}^{(rr)} \left(\mathbf{z}_k^*, \mathbf{z}_k \right) \right] dt, \tag{B.3.13}$$

this then becomes

$$\begin{aligned}
&i\sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \dot{d}_k^{(1)} e^{iS_k^{(1)}} - \sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle d_k^{(1)} e^{iS_k^{(1)}} \left[i\left(\frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\mathbf{z}_k \dot{\mathbf{z}}_k^*}{2} \right) - H_{\mathrm{ord}}^{(11)} (\mathbf{z}_k^*, \mathbf{z}_k) \right] \\
&+ i\sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle d_k^{(1)} e^{iS_k^{(1)}} \left[\mathbf{z}_j^* \dot{\mathbf{z}}_k - \frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right] \\
&- \sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle d_k^{(1)} e^{iS_k^{(1)}} H_{\mathrm{ord}}^{(11)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) - \sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle d_k^{(1)} e^{iS_k^{(1)}} H_{\mathrm{ord}}^{12} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) = 0,
\end{aligned} \tag{B.3.14}$$

and finally

$$\begin{aligned}
i\sum_k \dot{d}_i^{(1)} e^{iS_i^{(1)}} \langle \mathbf{z}_j |\, \mathbf{z}_i \rangle = &\sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \delta^2 H^{(11)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) d_k^{(1)} e^{iS_k^{(1)}} \\
&+ \sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle H_{\mathrm{ord}}^{(12)} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) d_k^{(2)} e^{iS_k^{(2)}},
\end{aligned} \tag{B.3.15}$$

or more generally

$$\begin{aligned} i\sum_k \dot{d}_i^{(r)} e^{iS_i^{(r)}} \langle \mathbf{z}_j |\, \mathbf{z}_i\rangle = &\sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k\rangle\, \delta^2 H^{(rr)}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) d_k^{(r)} e^{iS_k^{(r)}} \\ &+ \sum_k \langle \mathbf{z}_j |\, \mathbf{z}_k\rangle \sum_{s\neq r} H_{\text{ord}}^{(rs)}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) d_k^{(s)} e^{iS_k^{(s)}}, \end{aligned} \tag{B.3.16}$$

where, similarly to with the CCS method,

$$\delta^2 H^{(rr)}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) = \left[H_{\text{ord}}^{(rr)}\left(\mathbf{z}_j^*, \mathbf{z}_k\right) - H_{\text{ord}}^{(rr)}\left(\mathbf{z}_k^*, \mathbf{z}_k\right) - i\left(\mathbf{z}_j^* - \mathbf{z}_k^*\right)\dot{\mathbf{z}}_k\right]. \tag{B.3.17}$$

It is possible to find the equation for $\dot{\mathbf{z}}$ from the multi-configurational Lagrangian using a fully variational method. As shown by Shalashilin in ref [23], this leads to a formulation equivalent to the G-MCTDH equations [112, 113, 115]. A discussion of this is given in appendix A of [23].

B.4 Derivation of the Equations for the Second Formulation of the Multiconfigurational Ehrenfest Method

The process for finding the time derivative equations for the MCEv2 method is very similar to that of the MCEv1 method. The ansatz for the MCEv2 method is

$$\begin{aligned} |\Psi\rangle &= \sum_k D_k\, |\varphi_k\rangle \\ &= \sum_k D_k \left(\sum_r a_k^{(r)}\, |r\rangle\right) |\mathbf{z}_k\rangle\,. \end{aligned} \tag{B.4.1}$$

An important consideration is that in the MCEv2 method the configurations are normalised, and so unlike in the case of the MCEv1 method, within a single configuration k,

$$\langle \varphi_k |\, \varphi_k\rangle = \langle \mathbf{z}_k |\, \mathbf{z}_k\rangle \sum_r \left(a_k^{(r)*} a^{(r)}\right) = 1. \tag{B.4.2}$$

The condition still stands from the MCEv1 method that

$$\langle r |\, s\rangle = \delta_{rs}. \tag{B.4.3}$$

Again the single configuration Lagrangian is used to find the equation for the evolution of the coherent states, giving the same result for $\dot{\mathbf{z}}$. As the single configuration amplitudes $a_k^{(r)}$ are not coupled across configurations in this method, the single configuration Lagrangian is also used to find the time derivative equations for these also. The single

configuration Lagrangian is unchanged from equation B.3.3, and the Euler-Lagrange equation for $a^{(r)}$ can be found as

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial a_k^{(r)*}} - \frac{\partial}{\partial t}\frac{\partial \mathcal{L}}{\partial \dot{a}_k^{(r)*}} =& i\dot{a}_k^{(r)} + i\left(\frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right)a_k^{(r)} \\ &- H_{\text{ord}}^{(rr)}\left(\mathbf{z}_k^*, \mathbf{z}_k\right)a_k^{(r)} - \sum_{s\neq r} H_{\text{ord}}^{(rs)}\left(\mathbf{z}_k^*, \mathbf{z}_k\right)a_k^{(s)} = 0,\end{aligned} \tag{B.4.4}$$

which after introduction of the substitution (B.3.12) becomes,

$$\dot{d}_k^{(r)} = -i\sum_{s\neq r} H_{\text{ord}}^{(rs)}\left(\mathbf{z}_k^*, \mathbf{z}_k\right)d_k^{(s)}e^{i\left(S_k^{(s)} - S_k^{(r)}\right)}. \tag{B.4.5}$$

The equation for the time evolution of the D_k prefactor can be found through the variational method as earlier, it can also be found much more simply by substitution of the ansatz (B.4.1) into the Schrödinger equation

$$\frac{d\left|\Psi\right\rangle}{dt} = -i\hat{H}\left|\Psi\right\rangle, \tag{B.4.6}$$

such that

$$\sum_i \frac{dD_i(t)}{dt}\left|\varphi_i(t)\right\rangle = -i\sum_k D_k(t)\hat{H}\left|\varphi_k(t)\right\rangle - \sum_k D_k(t)\left|\frac{d\varphi_k(t)}{dt}\right\rangle, \tag{B.4.7}$$

and so

$$\sum_i \left\langle\varphi_j\right|\left.\varphi_i\right\rangle\dot{D}_i = -i\sum_k\left\langle\varphi_j\right|\hat{H}\left|\varphi_k\right\rangle D_k - \sum_k\left\langle\varphi_j\right|\left.\dot{\varphi}_k\right\rangle D_k. \tag{B.4.8}$$

In this equation the overlap between two configurations is given as

$$\left\langle\varphi_j\right|\left.\varphi_k\right\rangle = \left\langle\mathbf{z}_j\right|\left.\mathbf{z}_k\right\rangle\sum_r\left(a_j^{(r)*}a_k^{(r)}\right), \tag{B.4.9}$$

the matrix elements of the Hamiltonian are given as

$$\begin{aligned}\left\langle\varphi_j\right|\hat{H}\left|\varphi_k\right\rangle &= \sum_{rs}\left\langle\mathbf{z}_j\right|\hat{H}^{(rs)}\left|\mathbf{z}_k\right\rangle a_j^{(r)*}a_k^{(s)} \\ &= \left\langle\mathbf{z}_j\right|\left.\mathbf{z}_k\right\rangle\sum_{rs} H_{\text{ord}}^{(rs)}\left(\mathbf{z}_j^*, \mathbf{z}_k\right)a_j^{(r)*}a_k^{(s)},\end{aligned} \tag{B.4.10}$$

and by way of equation (B.1.8),

$$\begin{aligned}\left\langle\varphi_j\right|\left.\dot{\varphi}_k\right\rangle &= \left\langle\mathbf{z}_j\right|\left.\mathbf{z}_k\right\rangle\sum_r\left(a_j^{(r)*}\dot{a}_k^{(r)}\right) + \left\langle\mathbf{z}_j\right|\left.\dot{\mathbf{z}}_k\right\rangle\sum_r\left(a_j^{(r)*}a_k^{(r)}\right) \\ &= \left\langle\mathbf{z}_j\right|\left.\mathbf{z}_k\right\rangle\sum_r\left(a_j^{(r)*}\dot{a}_k^{(r)}\right) + \left\langle\varphi_j\right|\left.\varphi_k\right\rangle\left(\left(\mathbf{z}_j^* - \mathbf{z}_k^*\right)\dot{\mathbf{z}}_k + \frac{\mathbf{z}_k^*\dot{\mathbf{z}}_k - \dot{\mathbf{z}}_k^*\mathbf{z}_k}{2}\right).\end{aligned} \tag{B.4.11}$$

Equation (B.4.4) can be introduced to the first term of the above equation, such that

$$\begin{aligned}\langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \sum_r \left(a_j^{(r)*} \dot{a}_k^{(r)} \right) = i \sum_r \Bigg[& i \left(\frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k}{2} - \frac{\dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right) a_j^{(r)*} a_k^{(r)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \\ & - H_{\text{ord}}^{(rr)} \left(\mathbf{z}_k^*, \mathbf{z}_k \right) a_j^{(r)*} a_k^{(r)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \, r \\ & - \sum_{s \neq r} H_{\text{ord}}^{(rs)} \left(\mathbf{z}_k^*, \mathbf{z}_k \right) a_j^{(r)*} a_k^{(s)} \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \Bigg] \\ = & - \langle \varphi_j |\, \varphi_k \rangle \left(\frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k - \dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right) \\ & - i \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \sum_{rs} H_{\text{ord}}^{(rs)} \left(\mathbf{z}_k^*, \mathbf{z}_k \right) a_j^{(r)*} a_k^{(s)} \\ = & - \langle \varphi_j |\, \varphi_k \rangle \left(\frac{\mathbf{z}_k^* \dot{\mathbf{z}}_k - \dot{\mathbf{z}}_k^* \mathbf{z}_k}{2} \right) - i \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \widetilde{H}_{jk}\end{aligned} \tag{B.4.12}$$

for

$$\widetilde{H}_{jk} = \sum_{rs} a_j^{(r)*} \left\langle \mathbf{z}_k \right| \hat{H} \left| \mathbf{z}_k \right\rangle a_k^{(s)} = \sum_{rs} d_j^{(r)*} \left\langle \mathbf{z}_k \right| \hat{H} \left| \mathbf{z}_k \right\rangle d_k^{(s)} e^{i\left(S_k^{(s)} - S_j^{(r)} \right)}. \tag{B.4.13}$$

When substituted into equation (B.4.11), we then get

$$\langle \varphi_j |\, \dot{\varphi}_k \rangle = \langle \varphi_j |\, \varphi_k \rangle \left(\left(\mathbf{z}_j^* - \mathbf{z}_k^* \right) \dot{\mathbf{z}}_k \right) - i \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \widetilde{H}_{jk}. \tag{B.4.14}$$

Returning then to equation (B.4.8),

$$\begin{aligned}\sum_i \langle \varphi_j |\, \varphi_i \rangle \dot{D}_i &= -i \sum_k \left[\left\langle \varphi_j \right| \hat{H} \left| \varphi_k \right\rangle - \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \widetilde{H}_{jk} - i \langle \varphi_j |\, \varphi_k \rangle \left(\left(\mathbf{z}_j^* - \mathbf{z}_k^* \right) \dot{\mathbf{z}}_k \right) \right] D_k \\ &= -i \sum_k \Delta^2 \langle H \rangle_{jk} D_k,\end{aligned} \tag{B.4.15}$$

where

$$\Delta^2 \langle H \rangle_{jk} = \left[\left\langle \varphi_j \right| \hat{H} \left| \varphi_k \right\rangle - \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \widetilde{H}_{jk} - i \langle \varphi_j |\, \varphi_k \rangle \left(\left(\mathbf{z}_j^* - \mathbf{z}_k^* \right) \dot{\mathbf{z}}_k \right) \right]. \tag{B.4.16}$$

This is effectively the same as for the CCS equations, as for a single electronic state

$$\langle \varphi_j |\, \varphi_k \rangle = \langle \mathbf{z}_j |\, \mathbf{z}_k \rangle \, e^{i(S_k - S_j)}, \tag{B.4.17}$$

and so in that situation

$$\Delta^2 \langle H \rangle_{jk} = \delta^2 H'_{\text{ord}} \left(\mathbf{z}_j^*, \mathbf{z}_k \right) e^{i(S_k - S_j)}, \tag{B.4.18}$$

making equations (B.2.14) and (B.4.15) equivalent

Appendix C

Use of the CCS/MCE Program

C.1 Running the Program

The program allows a wide array of simulations using the MCE and CCS equations. The files are separated into three folders : `run`, `src` and `build`. The `src` folder contains all the source files, the `build` folder contains the Makefile and any temporary compilation files, and the `run` folder contains scripts to compile and run the program and collect and combine results in addition to the input files containing the simulation parameters, which are discussed in more detail in section C.2. This section will describe the function of the scripts in the `run` folder.

As discussed in Appendix A.4, the MCE and CCS methods usually require a large number of repeat executions. These repeat executions are split in two levels - firstly a number of different instances of the program are created in separate folders (these are named `1-run, 2-run,` etc), and secondly the program reruns itself for a number of times, which can be done on multiple processors simultaneously using OpenMP directives. To run the program the `run.sh` file in the run folder is executed, with the following arguments:

$1 - Total number of repeat calculations desired
$2 - Number of parallel threads per folder
$3 - Number of folders to split the job into (referred to as sub-folders).

For example, to run 128 jobs using 4 parallel cores, with the load split into 4 folders, you would use the command

```
./run.sh 128 4 4
```

which would have a total of 16 parallel threads running simultaneously. This allows OpenMP execution to be carried out over many more cores than would be present on a single node. Within the `run.sh` script, the arguments are checked first to ensure they comply with the following conditions:

- There are 3 arguments used;
- All arguments are integers;
- All arguments are greater than 0;
- \$2 is not greater than the number of cores per node of the machine;
- \$3 is less than 100;
- \$1/\$3 is less than 1000 (Less than 1000 repeats per program instance);
- If the conjugate_repeats flag is enabled, that the total number of repeats is an integer multiple of $2 \times \$2 \times \$3$;
- If the conjugate_repeats flag is disabled, that the total number of repeats is an integer multiple of $\$2 \times \$3$;
- The total number of cores requested ($\$2 \times \$3$) is below 100;
- If there is no job management system present, that \$3 is equal to 1.

Following this the folders in which the program will be executed are set up, and the program is compiled by way of the Makefile in the `build` folder. For portability the program is compiled using the GNU fortran compiler, gfortran. An issue encountered during development is that the intel ifort 12.1 compiler results in problems due to a known bug with allocation through subroutines when OpenMP is enabled, however the intel compiler can be used if not version 12.1 and this would require modifications to the makefile in the build folder. Once compilation is completed the input files and the program executable are copied to the execution folders, along with a job script used by the job management system which controls how long the program will run for, how much system memory it can be allocated, how many parallel cores are needed and how many simultaneous instances of the program will be running. By submitting this job script to the job management system all instances of the program will be run.

The `run.sh` script creates a second script, called `result.sh` which when run calls the `collate.sh` script which combines the results from all the completed runs. This script collects all the data from the different subfolders, calls averaging programs to combine

this data, then puts all relevant files in a results folder, before deleting the raw data. By disabling a flag in the script, this raw data can be preserved. The `collate.sh` script requires the following arguments:

$1- The path of the folder in which the raw data exists
$2- The total number of repeats for the entire simulation
$3- The number of sub-folders in the raw data folder
$4- The random number generated by the run.sh script as a unique run identifier
$5- The name of the results file which calls this script (deleted at the end)

Once all the output files are collected and averaged a set of gnuplot scripts are created, and if gnuplot is present on the machine these scripts are run, allowing instant graphical results.

Often a cluster computer with many users will impose time limits on running jobs, and once this time limit is reached the job will be cancelled, aborting the program. If time limits cause a simulation to be aborted prematurely, this simulation can be restarted through the `restart.sh` script which uses the same arguments as the run.sh script. This file should be called from the running folder of the most recent partial run. When this script is run it collects the output files from the execution folders and uses them as input files for a new set of simulations before calling the `run.sh` script. This allows the simulation to restart with a wavefunction in the same state as when the program was aborted.

If multiple partial runs are used, the data from the different partial runs can be combined with the `combine.sh` script which requires the file `folderlist.dat`, a file containing a list of the required folders which should be in order. This file is automatically created by the `restart.sh` script, however care should be taken to ensure that no confusion occurs when there are multiple simulations happening at the same time. This script overwrites the output files of the final partial run with the combined data (after making a backup), averages the data in each sub-folder (as would be done by the main program), and then calls the `collate.sh` script to combine and average the data.

C.2 Input Files

There are three input files containing all the parameters needed : `input.dat`, `inham.dat` and `prop.dat`. This section will contain a description of the various parameters in the

different input files. The `input.dat` file controls most aspects of the program infrastructure and operation, and contains parameters given in tables C.1 and C.2. Unless otherwise stated, it should be assumed that the values given are case sensitive.

Parameter	Values	Effect	Restrictions
System	SB	Spin Boson	Usable only with MCE
	HP	Harmonic Potential	Usable only with CCS
	FP	Free Particle	Usable only with CCS
	MP	Morse Potential	Usable only with CCS
	IV	Inverted Gaussian	Usable only with CCS
	CP	Coulomb Potential	Usable only with CCS
	HH	Hennon-Heiles Potential	Usable only with CCS
Runfolder	default	Runs program in a folder with the pattern <method>-<system>-<rand> ie CCS-HP-31254	Case insensitive
	<any string>	<method>-<system>-<string> ie CCS-HP-withgrids	
debug	0	No debug output generated	
	1	Debug outputs generated	Will not work with dynamic basis set sizes
gen	YES/NO	Basis set generated	
prop	YES/NO	Basis set propagated	
restart	0	Standard simulation	
	1	Restart a prior simulation	Set by restart.sh
cmprss	YES/NO	Automatically change the compression parameter or the grid spacing to ensure an acceptable norm	
method	CCS	Runs using CCS equations	Only valid for 1 PES
	MCEv1	Runs using MCEv1 eqns	Only valid for >1 PES
	MCEv2	Runs using MCEv2 eqns	Only valid for >1 PES
	MCE12	Two jobs are submitted, 1 using MCEv1 eqns, the other using MCEv2 eqns	Only valid for >1 PES
repeats	<number>	Number of repeats for this instance of the program	Integer. Set automatically by the run.sh script
Conjugate_Repeats	YES/NO	If yes, constructs the basis set for every other repeat around the conjugate of z_0 of the previous run.	If enabled must have an even no. repeats for each instance of the program. Also incompatible with restarting a prior simulation.
in_nbf	<number>	Initial number of basis functions.	Integer
ndim	<number>	Number of degrees of freedom in the system	Integer
in_PES	<number>	Initial electronic state of the wavefunction	Integer

Table C.1: Parameters in the input.dat file, part 1

Parameter	Values	Effect	Restrictions
npes	<number>	Number of electronic states in the system	Integer. Must be 1 for CCS and >1 for MCE
basis	SWARM	Basis set is a swarm	
	SWTRN	Basis set is a swarm of trains	
	TRAIN	Basis set is a single train	
	GRID	Basis set is a regular grid	Only valid for $ndim \in \{1, 3\}$
	GRSWM	Basis set is a regular grid in one dimension with swarms in the other two	Only valid for $ndim = 3$
ALCMP	<number>	Seed number for α_c	Double precision real number
nbfadapt	yes/no	Enables adaptive basis sets	Only valid with GRID or GRSWM basis
nbfepsilon	<number>	Adaptive basis set cutoff ζ parameter	Double precision real number in the form `1.0d-x`
gridsp	<number>	Spacing Δ between adjacent grid points	Real number in the range $0.8 \leq \Delta \leq 2.0$
qsizex	<number>	Size of a 3D grid in x_q	Integer
psizex	<number>	Size of a 3D grid in x_p	Integer
qsizey	<number>	Size of a 3D grid in y_q	Integer
psizey	<number>	Size of a 3D grid in y_p	Integer
qsizez	<number>	Size of a 1D or 3D grid in z_q	Integer
psizez	<number>	Size of a 1D or 3D grid in z_p	Integer
Cloning	yes/no	Enables basis set cloning	
max_cloning	<number>	Maximum number of cloning events	Integer
clon_freq	<number>	Minimum number of timesteps between cloning events	Integer ≥ 50
trainsp	<number>	Number of timesteps between adjacent basis functions in a train-type or swarm-train basis	Integer
def_stp	<number>	Number of basis functions per train in a swarm-train basis set	Integer. May be increased or decreased by 1 by program
matfun	zgesv/zheev	Allows change of function used for linear equations	
SEED	<number>	Seed of random number generator	Defaults to 0 which takes the seed from `/bin/urandom`
gamma	<number>	Width parameter for CSs	Default value 1.0d0
mu	<number>	Centre of z_0	Double precision real number
hbar	<number>	Value of $\hbar$.	Defaults to $\hbar = 1$.

TABLE C.2: Parameters in the input.dat file, part 2

The `inham.dat` file contains parameters specific to the available systems. Different parameters are read depending on the value of the "System" parameter in the `input.dat` file. Table C.3 gives these parameters, and unless otherwise stated all parameters are double precision real numbers.

Parameter	Description
Spin Boson Model Parameters	
SBDelta	Tunnelling amplitude between states Δ, with default value of 1.0d0
SBEps	Bias detuning parameter ϵ
SBw	Cutoff frequency ω_c
SBwmax	Largest frequency of the bath modes, which should be $\omega_{max} = 5\omega_c$
SBkondo	Kondo parameter α_k
SBBeta	Thermal parameter $\beta = 1/kT$
SBupnorm	Largest allowed value for the initial norm
SBdownnorm	Smallest allowed value for the initial norm
Harmonic Potential Parameters	
HPw	Frequency of the harmonic oscillator
HPupnorm	Largest allowed value for the initial norm
HPdownnorm	Smallest allowed value for the initial norm
Free Particle Parameters	
FPmass	Mass of the free particle
FPupnorm	Largest allowed value for the initial norm
FPdownnorm	Smallest allowed value for the initial norm
Morse Potential Parameters	
MPw	Frequency of the Morse oscillator
MPmass	Mass of the particle in a Morse oscillator
MPDissEn	Dissociation Energy for Morse ocillator
MPWellParam	Width of potential well for Morse oscillator
MPupnorm	Largest allowed value for the initial norm
MPdownnorm	Smallest allowed value for the initial norm
Inverted Gaussian Parameters for High Harmonic Generation	
IVm	Mass of particle (electron) in strong field
IVw	Frequency of laser field
IVlambda	Width of Gaussian potential well
IVIntensity	Intensity of laser field
IVupnorm	Largest allowed value for the initial norm
IVdownnorm	Smallest allowed value for the initial norm
Coulomb Potential Parameters for High Harmonic Generation	
CPm	Mass of particle in a strong laser field
CPfrequency	Frequency of laser field
CPRc	Initial distance between particle and the potential well
CPIntensity	Intensity of the laser field
CPupnorm	Largest allowed value for the initial norm
CPdownnorm	Smallest allowed value for the initial norm
Henon-Heiles Parameters	
HHcoupling	Coupling between Henon-Heiles oscillators
HHupnorm	Largest allowed value for the initial norm
HHdownnorm	Smallest allowed value for the initial norm
Energy Checking Parameters	
ECheck (YES/NO)	Flag to enable checking of energy $p^2/2$ of single initial basis functions to ensure they will not run away from each other too fast (YES/NO)
Ntries (Integer)	Number of recalculations before program is aborted if energy check fails
Ebfmax	Maximum allowed energy of single initial basis function
Ebfmin	Minimum allowed energy of single initial basis function

Table C.3: Parameters in the inham.dat file

The `prop.dat` file contains parameters specific to the propagation of the wavefunction. Table C.4 lists the parameters contained within this file. All values in the file are in atomic units, and all but the final parameter are double precision real numbers.

Parameter	Description
dtmin	Minimum allowed step size, used for adaptive step size propagation
dtmax	Maximum allowed step size, used for adaptive step size propagation
dtinit	For adaptive step size propagation, this is the size of the first time step taken. For static step size propagation this is the size of every time step taken
time_end	End time of propagation
time_start	Start time of propagation. For restarted simulations this is set automatically
step	(adaptive/static) Determines the type of propagation (See section A.3 for details)

TABLE C.4: Parameters in the prop.dat file

C.3 Output Files

There are three types of output files generated by the program. The first is the raw data, which is a set of files specific to each individual repeat simulation. The second is the combined data which is data that has been combined and averaged over a set or over all repeat simulations. The third is plotting outputs which contains gnuplot commands to graphically display data from other output files. This section will give a description of all the possible output files, their data fields and under what conditions the file will be created.

C.3.1 Raw Data Outputs

The raw data outputs are created in the running sub-folder (named `1-run, 2-run,` etc). Most of them refer to a single repeat simulation of the program, and have the naming convention "<name>-#.out", where # denotes the repeat number.

(a) `normpop-#.out` contains data for each time step for a single repeat and is always created. This is the main data file for the simulation, and for most simulations this will be the only output file used in data processing. The `normpop-#.out` file contains the time, norm, autocorrelation function (real, imaginary and absolute), population in each electronic state and an extra field with real, imaginary and absolute parts which can be determined by the system (for example dipole acceleration for HHG simulations, or cross-correlation function). If there are two electronic states then this file will also contain the population sum and population difference.

(b) `Outbs-#.out` contains the wavefunction data for the most recent time step, and can be used to restart a simulation. If the program is run only to generate an initial basis set, this is also the main output file. It contains all the wavefunction data needed to restart a simulation and is laid out in three sections. Firstly there is a preamble containing the number of degrees of freedom, number of electronic states, number of basis functions, initial electronic state of the wavefunction, the linear algebra equation used (zgesv/zheev) and the current simulation time. Secondly there is the real and imaginary parts of the $|\mathbf{z}_0\rangle$ state in all dimensions. Finally there are the wavefunction parameters D_k, $d_k^{(r)}$, $S_k^{(r)}$, and $z_k^{(m)}$ for all values of k, r, and$m$. This file does not however contain the information contained in `prop.dat` or `inham.dat` or any information about cloning so a restarted simulation will still need the main input files.

(c) `timesteps.out` contains a list of all time step sizes and is only generated when adaptive step size propagation is enabled

(d) `clonearr-#.out` contains a list of the basis function indices ($k = 1 \ldots N_{bf}$) with both the number of times that particular basis function has undergone cloning and the most recent time step in which that basis function was cloned. This file is only generated when basis function cloning is enabled, and can be used to restart a simulation which has used cloning to ensure that N_{cln} is not exceeded and that two cloning events do not occur too close together in time.

(e) `Clonetrack-#.out` is written to every time a cloning event occurs, and contains the index of the basis function which has been cloned, the index to which the new cloned basis function has been written, the time step number at which cloning occurred and the absolute values of the single configuration amplitudes $a_k^{(r)}$. The data in this output file can be used to reconstruct a cloning history of the wavefunction.

(f) `normpop-#_interp.out` is an interpolated form of the `normpop-#.out` file, created when adaptive step size propagation is used. This is needed for averaging all the data from each repeat execution in such cases as it ensures that each file being averaged will have the same size time step.

(g) `wavefn-#.out` gives the D_k amplitude and the real and imaginary parts of the $z_k^{(m)}$ values for all basis functions. This is only generated when a 1D adaptive grid is used, and the data within it can be used to visualise the wavefunction such as is done for figure 4.9. As a regular grid removes the need for repeat executions, the number in the filename is the time step number, with one file being created at each time step.

(h) `PropVars-#.out` contains all the wavefunction parameters D_k, $d_k^{(r)}$, and $S_k^{(r)}$ for all values of k, and$r$ at each time step. This file is only created when the debug flag is enabled in the `input.dat` file.

(i) `Traj-#.out` contains all the real and imaginary parts of the coherent state variables $z_k^{(m)}$ for each value of k and m at each time step. This file is also only created if the debug flag in the `input.dat` file is enabled.

(j) `normpop.out` is an averaged combination of all the `normpop-#.out` files for a single instance of the program (or if adaptive step sizes are used, an averaged combination of all the `normpop-#_interp.out` files). This file is created by the main program itself at the end of propagation, but only when the program has completed execution uninterrupted. If the program has restarted at any point this file will not be created and is instead made by a small external program called `subavrg.exe` (with the help of a separate interpolation program if needed)

C.3.2 Combined Outputs

The combined outputs are created in the output folder upon running of the `collate.sh` script. Within this folder also is the output files from the job management system if any exist, which gives the status of the program as it runs and contains any error messages generated as the program runs. In these files, for the most part, # will denote the sub-folder number from which the data file originated.

(a) `normpop_#.out` is a copy of the file `normpop.out` from each sub-folder, containing the time, norm, autocorrelation function (real, imaginary and absolute), population in each electronic state and an extra field with real, imaginary and absolute parts which can be determined by the system (for example dipole acceleration for HHG simulations, or cross-correlation function). If there are two electronic states then this file will also contain the population sum and population difference.

(b) `normpop_cumul_#.out` contains a cumulative average of the data from the `normpop_#.out` files. There are always as many files of the `normpop_cumul_#.out` type as there is of the `normpop_#.out`, and are numbered by the number of repeat executions considered, so if each subfolder has 32 repeats then the first will be named `normpop_cumul_32.out` and be identical to `normpop_1.out`, the second will be named `normpop_cumul_64.out` and be an average of the data from the `normpop_1.out` and `normpop_2.out` files, the third will be named `normpop_cumul_72.out` and be an average of the data from the `normpop_1.out`, `normpop_2.out` and `normpop_3.out`

files, and so on. The highest numbered file will be an average over all repeat executions. The way in which these files can be used to test convergence is detailed in section C.4.

(c) `Outbs-#_#.out` is a copy of the `Outbs-#.out` file from each subfolder, with the first numerical identifier being the repeat number and the second being the sub-folder number. These files are used for restarting a prematurely aborted simulation.

(d) `timesteps.out` is a concatenation of all the `timesteps-#.out` files from each sub-folder, containing the sizes of time steps for all repeat executions of the program. This is only generated if adaptive step size propagation has been used.

(e) `timehist_#.out` contains histogram data generated from the individual `timesteps-#.out` files. This is also only created if adaptive step size propagation is used.

(f) `timehist_all.out` contains histogram data generated from the `timesteps.out` file, and so is a histogram of time steps for all instances of the simulation.

(g) `popdiffresiduals.out` contains the differences between the population difference from the highest numbered `normpop_cumul_#.out` file and that from each of the lower numbered `normpop_cumul_#.out` files. This can be used to assess convergence as is shown in section C.4.

C.3.3 Plotting Outputs

The plotting outputs can be divided into two categories. The first, used mainly as a preliminary check on the validity of a simulation and to ensure that no major errors have occurred, plots data from the files enumerated in sectionC.3.1. The second plots data from the files enumerated in section C.3.2 and can be used to assess the combined data and in the process of finding the optimal running parameters. The second type will be discussed in detail in section C.4, while the first type will be discussed here. As in section C.3.1, unless otherwise stated it should be assumed that the numerical identifiers # refer to the repeat number.

(a) `plotacf-#.gpl` plots the real, imaginary and absolute values of the autocorrelation function from the data in the `normpop-#.out` file.

(b) `plotdif-#.gpl` plots the population difference from the `normpop-#.out` file

(c) `plotext-#.gpl` plots the real, imaginary and absolute values from the "extra" output data field in the `normpop-#.out` file, which can be the dipole acceleration in the case of HHG simulations or the cross-correlation function for example.

(d) `plotnrm-#.gpl` plots the norm from the `normpop-#.out` file.

(e) `plotamps-#.gpl` plots the real values of the D_k prefactors against their imaginary counterparts for each basis function k. The data for these plots are taken from the `PropVars-#.out` file and as with the data file, this plotting file will only be created if the debug flag is enabled in the `input.dat` file

(f) `plotd-#.gpl` plots the real values of the $d_k^{(r)}$ single configuration amplitudes against their imaginary counterparts for each basis function k and each electronic state r. The data for this plot also comes from the `PropVars-#.out` file and so again this file is only created when the debug flag is enabled

(g) `plotact-#.gpl` plots the action $S_k^{(r)}$ against time for each of the basis functions k and each electronic state r. Again the data for this plot originates in the `PropVars-#.out` file and so the plotting file is only created when the debug flag is enabled

(h) `plot-Traj-#.out` plots the real and imaginary parts of the coherent state parameters $z_k^{(m)}$ for each basis function k and each degree of freedom m. The data used is taken from the `Traj-#.out` file and so this plotting file is only created when the debug flag is enabled.

C.4 Finding Optimal Running Parameters

When running a simulation it is important that the correct size of time step is used. As discussed in section A.3 a useful way of ensuring that the correct time step is always used is by using the adaptive time step system. Unfortunately if the basis set is constructed using trains at all this is not an option, as all basis functions must be equally temporally spaced at all times during simulation, else instabilities could result. A good way of ensuring that an adequate time step is used is by running the simulation first with a swarm-type basis set and adaptive step size propagation. Once the simulation data is collected a histogram of all timestep sizes can be generated by the files `plothist.gpl` and `plothistall.gpl`. The `plothistall.gpl` file uses data from the `timehist_all.out` file and so returns a histogram of all time steps taken by all instances of the program and as a check the `plothist.gpl` file plots the individual histograms from the time step data from each of the sub-folders to ensure that no outliers are effecting the combined data. The importance of carrying out this preliminary run can be seen in figures C.1 and C.2, which show the histogram data for adaptive step size simulations of the symmetric and asymmetric cases of the spin boson model respectively. For the symmetric case of the spin boson model the peak of the distribution of time steps is in the region of

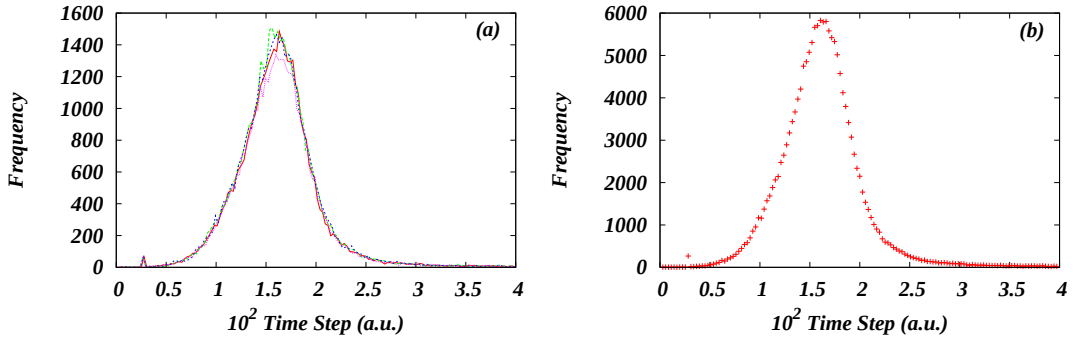


FIGURE C.1: Histograms of the time steps taken during adaptive time step propagation for the symmetric case of the spin boson model, with individual histograms from each of the sub-folders in (a) and a histogram of all the data combined in (b)

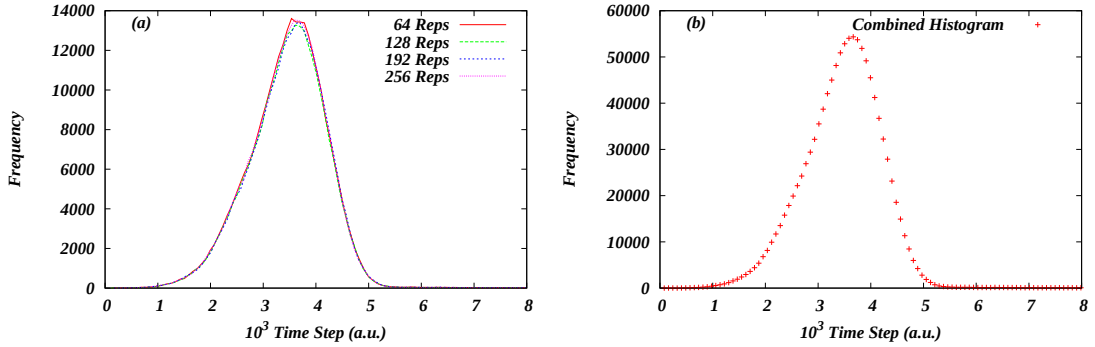


FIGURE C.2: Histograms of the time steps taken during adaptive time step propagation for the asymmetric case of the spin boson model, with individual histograms from each of the sub-folders in (a) and a histogram of all the data combined in (b)

$dt = 0.016$ and considering the width of the distribution an acceptable value of the static step size would be $dt = 0.01$. This value however would be completely unsuitable for the asymmetric case of the spin boson model, which peaks around $dt = 0.0036$ giving an acceptable value of the static step size at $dt = 0.002$.

A second important consideration is that the number of repeat executions of the program be sufficient to ensure an acceptable level of convergence. To aid in this, two plotting files are created, which compare the population difference from each of the cumulatively averaged data files `normpop_cumul_#.out` and the residuals between these population differences and that from the highest numbered `normpop_cumul_#.out` file. Examples of these plots are given in figures C.3 and C.4 for the symmetric and asymmetric cases of the spin boson model, using uncloned swarm-type basis sets and the MCEv2 equations. In some cases a lack of convergence will be readily apparent in the population difference plot (a), however sometimes the residual plot (b) would need to be considered. In the cases above it can be easily seen that convergence is approached, with the instability in the residuals greatly reduced by $N_{rpt} = 192$ in the asymmetric case and the overestimation

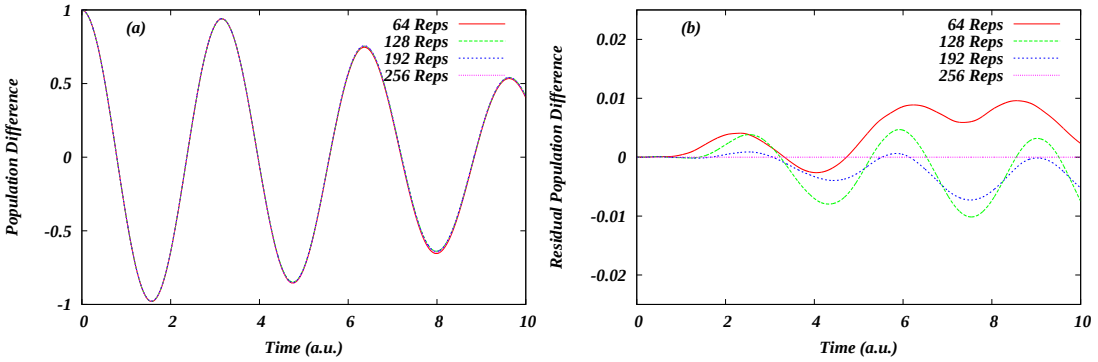


FIGURE C.3: Plots used to test convergence in the population difference for the symmetric case of the spin boson model, where (a) is the population difference averaged over different numbers of repeats, and (b) is the difference between the those population differences and that averaged over 256 repeats

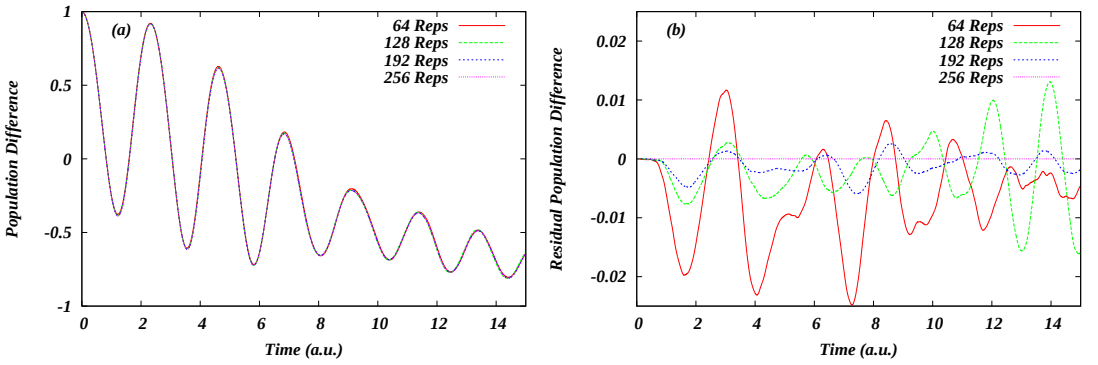


FIGURE C.4: Plots used to test convergence in the population difference for the asymmetric case of the spin boson model, where (a) is the population difference averaged over different numbers of repeats, and (b) is the difference between the those population differences and that averaged over 256 repeats

of the oscillations reduced in the symmetric case, although it should be noted that the residuals for the symmetric case are never as large as is seen for the asymmetric case, indicating that a larger number of repetitions may be needed for the asymmetric case than for the symmetric case.

Bibliography

[1] D. V. Makhov, W. J. Glover, T. J. Martínez, and D. V. Shalashilin. Ab Initio Multiple Cloning Algorithm For Quantum Nonadiabatic Molecular Dynamics. *J. Chem. Phys.*, 141(5):54110, 2014. doi: 10.1063/1.4891530.

[2] D. V. Makhov, K. Saita, T. J. Martínez, and D. V. Shalashilin. Ab Initio Multiple Cloning Simulations Of Pyrrole Photodissociation: TKER Spectra And Velocity Map Imaging. *Phys. Chem. Chem. Phys.*, 17:3316–3325, 2015. doi: 10.1039/C4CP04571H.

[3] E. J. Heller. Time-Dependent Approach To Semiclassical Dynamics. *J. Chem. Phys.*, 62(4):1544–1555, 1975. doi: 10.1063/1.430620.

[4] E. J. Heller. Frozen Gaussians: A Very Simple Semiclassical Approximation. *J. Chem. Phys.*, 75(6):2923–2931, 1981. doi: 10.1063/1.442382.

[5] E. J. Heller. The Semiclassical Way To Molecular Spectroscopy. *Acc. Chem. Res.*, 14(12):368–375, 1981. doi: 10.1021/ar00072a002.

[6] S.-I. Sawada, R. Heather, B. Jackson, and H. Metiu. A Strategy For Time Dependent Quantum Mechanical Calculations Using A Gaussian Wave Packet Representation Of The Wave Function. *J. Chem. Phys.*, 83(6):3009–3027, 1985. doi: 10.1063/1.449204.

[7] S.-I. Sawada and H. Metiu. A Multiple Trajectory Theory For Curve Crossing Problems Obtained By Using A Gaussian Wave Packet Representation Of The Nuclear Motion. *J. Chem. Phys.*, 84(1):227–238, 1986. doi: 10.1063/1.450175.

[8] M. F. Herman and E. Kluk. A Semiclasical Justification For The Use Of Nonspreading Wavepackets In Dynamics Calculations. *Chem. Phys.*, 91(1):27–34, 1984. doi: 10.1016/0301-0104(84)80039-7.

[9] K. G. Kay. Semiclassical Initial Value Treatments Of Atoms And Molecules. *Annu. Rev. Phys. Chem.*, 56:255–280, 2005. doi: 10.1146/annurev.physchem.56.092503.141257.

[10] D. J. Tannor and S. Garashchuk. Semiclassical Calculation Of Chemical Reaction Dynamics Via Wavepacket Correlation Functions. *Annu. Rev. Phys. Chem.*, 51: 553–600, 2000. doi: 10.1146/annurev.physchem.51.1.553.

[11] T. J. Martínez, M. Ben-Nun, and R. D. Levine. Multi-Electronic-State Molecular Dynamics: A Wave Function Approach With Applications. *J. Phys. Chem*, 100 (19):7884–7895, 1996. doi: 10.1021/jp953105a.

[12] T. J. Martínez, M. Ben-Nun, and R. D. Levine. Molecular Collision Dynamics On Several Electronic States. *J. Phys. Chem. A*, 101(36):6389–6402, 1997. doi: 10.1021/jp970842t.

[13] M. Ben-Nun and T. J. Martínez. Nonadiabatic Molecular Dynamics: Validation Of The Multiple Spawning Method For A Multidimensional Problem. *J. Chem. Phys.*, 108(17):7244–7257, 1998. doi: 10.1063/1.476142.

[14] M. Ben-Nun and T. J. Martínez. Ab Initio Molecular Dynamics Study Of Cis-Trans Photoisomerization In Ethylene. *Chem. Phys. Lett.*, 298(1-3):57–65, 1998. doi: 10.1016/S0009-2614(98)01115-4.

[15] M. Ben-Nun, J. Quenneville, and T. J. Martínez. Ab Initio Multiple Spawning: Photochemistry From First Principles Quantum Molecular Dynamics. *J. Phys. Chem. A*, 104(22):5161–5175, 2000. doi: 10.1021/jp994174i.

[16] H.-D. Meyer, U. Manthe, and L. S. Cederbaum. The Multi-Configurational Time-Dependent Hartree Approach. *Chem. Phys. Lett.*, 165(1):73–78, 1990. doi: 10.1016/0009-2614(90)87014-I.

[17] H.-D. Meyer. Studying Molecular Quantum Dynamics With The Multiconfiguration Time-Dependent Hartree Method. *WIREs Comput. Mol. Sci.*, 2(2):351–374, 2012. doi: 10.1002/wcms.87.

[18] M. H. Beck, A. Jäckle, G. A. Worth, and H.-D. Meyer. The Multiconfiguration Time-Dependent Hartree (MCTDH) Method: A Highly Efficient Algorithm For Propagating Wavepackets. *Phys. Rep.*, 324(1):1–105, 2000. doi: 10.1016/S0370-1573(99)00047-2.

[19] D. V. Shalashilin and M. S. Child. Multidimensional Quantum Propagation With The Help Of Coupled Coherent States. *J. Chem. Phys.*, 115(12):5367–5375, 2001. doi: 10.1063/1.1394939.

[20] M. J. Davis and E. J. Heller. Multidimensional Wave Functions From Classical Trajectories. *J. Chem. Phys.*, 75(8):3916–3924, 1981. doi: 10.1063/1.442548.

[21] W. H. Miller. The Semiclassical Initial Value Representation: A Potentially Practical Way For Adding Quantum Effects To Classical Molecular Dynamics Simulations. *J. Phys. Chem. A*, 105(13):2942–2955, 2001. doi: 10.1021/jp003712k.

[22] D. V. Shalashilin and M. S. Child. Description Of Tunneling With The Help Of Coupled Frozen Gaussians. *J. Chem. Phys.*, 114(21):9296–9304, 2001. doi: 10.1063/1.1367392.

[23] D. V. Shalashilin. Quantum Mechanics With The Basis Set Guided By Ehrenfest Trajectories: Theory And Application To Spin-Boson Model. *J. Chem. Phys.*, 130 (24):244101, 2009. doi: 10.1063/1.3153302.

[24] D. V. Shalashilin. Nonadiabatic Dynamics With The Help Of Multiconfigurational Ehrenfest Method: Improved Theory And Fully Quantum 24D Simulation Of Pyrazine. *J. Chem. Phys.*, 132(24):244111, 2010. doi: 10.1063/1.3442747.

[25] O. Vendrell and H.-D. Meyer. Multilayer Multiconfiguration Time-dependent Hartree Method: Implementation And Applications To A Henon-Heiles Hamiltonian And To Pyrazine. *J. Chem. Phys.*, 134(4):44135, 2011. doi: 10.1063/1.3535541.

[26] K. Saita and D. V. Shalashilin. On-The-Fly Ab Initio Molecular Dynamics With Multiconfigurational Ehrenfest Method. *J. Chem. Phys.*, 137(22):22A506, 2012. doi: 10.1063/1.4734313.

[27] K. Saita, M. G. D. Nix, and D. V. Shalashilin. Simulation Of Ultrafast Photodynamics Of Pyrrole With A Multiconfigurational Ehrenfest Method. *Phys. Chem. Chem. Phys.*, 15(38):16227–16235, 2013. doi: 10.1039/C3CP51199E.

[28] E. Kluk, M. F. Herman, and H. L. Davis. Comparison Of The Propagation Of Semiclassical Frozen Gaussian Wave Functions With Quantum Propagation For A Highly Excited Anharmonic Oscillator. *J. Chem. Phys.*, 84(1):326–334, 1986. doi: 10.1063/1.450142.

[29] M. F. Herman. Dynamics By Semiclassical Methods. *Annu. Rev. Phys. Chem.*, 45:83–111, 1994. doi: 10.1146/annurev.physchem.45.1.83.

[30] W. H. Miller and T. F. George. Analytic Continuation Of Classical Mechanics For Classically Forbidden Collision Processes. *J. Chem. Phys.*, 56(11):5668–5681, 1972. doi: 10.1063/1.1677086.

[31] D. V. Shalashilin and M. S. Child. The Phase Space CCS Approach To Quantum And Semiclassical Molecular Dynamics For High-Dimensional Systems. *Chem. Phys.*, 304(12):103–120, 2004. doi: 10.1016/j.chemphys.2004.06.013.

[32] D. V. Shalashilin. Multiconfigurational Ehrenfest Approach To Quantum Coherent Dynamics In Large Molecular Systems. *Faraday Discuss.*, 153(0):105–116, 2011. doi: 10.1039/C1FD00034A.

[33] W. H. Miller. Semiclassical Theory Of Atom-Diatom Collisions: Path Integrals And The Classical S Matrix. *J. Chem. Phys.*, 53(5):1949–1959, 1970. doi: 10.1063/1.1674275.

[34] W. H. Miller. Classical S Matrix: Numerical Application to Inelastic Collisions. *J. Chem. Phys.*, 53(9):3578–3587, 1970. doi: 10.1063/1.1674535.

[35] R. A. Marcus. Theory Of Semiclassical Transition Probabilities (S Matrix) For Inelastic And Reactive Collisions. *J. Chem. Phys.*, 54(9):3965–3979, 1971. doi: 10.1063/1.1675453.

[36] J. H. Van Vleck. The Correspondence Principle In The Statistical Interpretation Of Quantum Mechanics. *Proc. Natl. Acad. Sci. U.S.A.*, 14(2):178–188, 1928.

[37] A. Perelomov. *Generalized Coherent States And Their Applications*. Springer-Verlag, 1986. URL http://www.springer.com/physics/quantum+physics/book/978-3-540-15912-4.

[38] E. Schrödinger. Der Stetige Übergang Von Der Mikro- Zur Makromechanik. *Naturwissenschaften*, 14:664–666, 1926.

[39] J. von Neumann. *Mathematical Foundations Of Quantum Mechanics*. Springer, 1932.

[40] R. J. Glauber. The Quantum Theory Of Optical Coherence. *Phys. Rev.*, 130(6):2529–2539, 1963. doi: 10.1103/PhysRev.130.2529.

[41] R. J. Glauber. Coherent And Incoherent States Of The Radiation Field. *Phys. Rev.*, 131(6):2766–2788, 1963. doi: 10.1103/PhysRev.131.2766.

[42] G. Campolieti and P. Brumer. Semiclassical Initial Value Theory For Dissociation Dynamics. *J. Chem. Phys.*, 107(3):791–803, 1997. doi: 10.1063/1.474379.

[43] X. Sun and W. H. Miller. Mixed Semiclassical-Classical Approaches To The Dynamics Of Complex Molecular Systems. *J. Chem. Phys.*, 106(3):916–927, 1997. doi: 10.1063/1.473171.

[44] G. Campolieti and P. Brumer. Semiclassical Initial Value Approach For Chaotic Long-Lived Dynamics. *J. Chem. Phys.*, 109(8):2999–3003, 1998. doi: 10.1063/1.476892.

[45] V. Guallar, V. S. Batista, and W. H. Miller. Semiclassical Molecular Dynamics Simulations Of Excited State Double-Proton Transfer In 7-Azaindole Dimers. *J. Chem. Phys.*, 110(20):9922–9936, 1999. doi: 10.1063/1.478866.

[46] V. Guallar, V. S. Batista, and W. H. Miller. Semiclassical Molecular Dynamics Simulations Of Intramolecular Proton Transfer In Photoexcited 2-(2'-Hydroxyphenyl)-Oxazole. *J. Chem. Phys.*, 113(21):9510–9522, 2000. doi: 10.1063/1.1321049.

[47] M. Thoss, W. H. Miller, and G. Stock. Semiclassical Description Of Nonadiabatic Quantum Dynamics: Application To The S_1-S_2 Conical Intersection In Pyrazine. *J. Chem. Phys.*, 112(23):10282–10292, 2000. doi: 10.1063/1.481668.

[48] S. Bonella and D. F. Coker. A Semiclassical Limit For The Mapping Hamiltonian Approach To Electronically Nonadiabatic Dynamics. *J. Chem. Phys.*, 114(18):7778–7789, 2001. doi: 10.1063/1.1366331.

[49] V. S. Batista and P. Brumer. A Direct Approach To One Photon Interference Contributions In The Coherent Control Of Photodissociation. *J. Chem. Phys.*, 114(23):10321–10331, 2001. doi: 10.1063/1.1372713.

[50] H. Wang, D. E. Manolopoulos, and W. H. Miller. Generalized Filinov Transformation Of The Semiclassical Initial Value Representation. *J. Chem. Phys.*, 115(14):6317–6326, 2001. doi: 10.1063/1.1402992.

[51] M. Ceotto, G. F. Tantardini, and A. Aspuru-Guzik. Fighting The Curse Of Dimensionality In First-Principles Semiclassical Calculations: Non-Local Reference States For Large Number Of Dimensions. *J. Chem. Phys.*, 135(21):214108, 2011. doi: 10.1063/1.3664731.

[52] S. Habershon. Linear Dependence And Energy Conservation In Gaussian Wavepacket Basis Sets. *J. Chem. Phys.*, 136(1):14109, 2012. doi: 10.1063/1.3671978.

[53] D. Gelman and S. D. Schwartz. Dissipative Dynamics With The Corrected Propagator Method. Numerical Comparison Between Fully Quantum And Mixed Quantum/Classical Simulations. *Chem. Phys.*, 370(13):62–69, 2010. doi: 10.1016/j.chemphys.2010.01.021.

[54] D. Gelman and S. D. Schwartz. Tunneling Dynamics With A Mixed Quantum-Classical Method: Quantum Corrected Propagator Combined With Frozen Gaussian Wave Packets. *J. Chem. Phys.*, 129(2):24504, 2008. doi: 10.1063/1.2949818.

[55] D. H. Zhang, J. Shao, and E. Pollak. Frozen Gaussian Series Representation Of The Imaginary Time Propagator Theory And Numerical Tests. *J. Chem. Phys.*, 131(4):44116, 2009. doi: 10.1063/1.3190328.

[56] M. Ceotto, S. Atahan, G. F. Tantardini, and A. Aspuru-Guzik. Multiple Coherent States For First-Principles Semiclassical Initial Value Representation Molecular Dynamics. *J. Chem. Phys.*, 130(23):234113, 2009. doi: 10.1063/1.3155062.

[57] E. Martin-Fierro and E. Pollak. Semiclassical Initial Value Series Solution Of The Spin Boson Problem. *J. Chem. Phys.*, 126(16):164108, 2007. doi: 10.1063/1.2714520.

[58] W. H. Miller. On The Relation Between The Semiclassical Initial Value Representation And An Exact Quantum Expansion In Time-Dependent Coherent States. *J. Phys. Chem. B*, 106(33):8132–8135, 2002. doi: 10.1021/jp020500+.

[59] W. H. Miller. An Alternate Derivation Of The Herman-Kluk (Coherent State) Semiclassical Initial Value Representation Of The Time Evolution Operator. *Mol. Phys.*, 100(4):397–400, 2002. doi: 10.1080/00268970110069029.

[60] D. V. Shalashilin and M. S. Child. Time Dependent Quantum Propagation In Phase Space. *J. Chem. Phys.*, 113(22):10028–10036, 2000. doi: 10.1063/1.1322075.

[61] D. V. Shalashilin and B. Jackson. Guiding Paths And Time-dependent Basis Sets For Wavefunction Propagation. *Chem. Phys. Lett.*, 318(4-5):305–313, 2000. doi: 10.1016/S0009-2614(00)00009-9.

[62] K. Husimi. Some Formal Properties Of The Density Matrix. *Proc. Phys. Math. Soc. Jpn.*, 22(4):264–314, 1940.

[63] P. A. J. Sherratt, D. V. Shalashilin, and M. S. Child. Description Of Multidimensional Tunnelling With The Help Of Coupled Coherent States Guided By Classical Hamiltonians With Quantum Corrections. *Chem. Phys.*, 322(12):127–134, 2006. doi: 10.1016/j.chemphys.2005.06.050.

[64] M. S. Child and D. V. Shalashilin. Locally Coupled Coherent States And Herman-Kluk Dynamics. *J. Chem. Phys.*, 118(5):2061–2071, 2003. doi: 10.1063/1.1531997.

[65] D. V. Shalashilin and M. S. Child. Nine-Dimensional Quantum Molecular Dynamics Simulation Of Intramolecular Vibrational Energy Redistribution In The CHD_3 Molecule With The Help Of Coupled Coherent States. *J. Chem. Phys.*, 119(4):1961–1969, 2003. doi: 10.1063/1.1584663.

[66] S. K. Reed, D. R. Glowacki, and D. V. Shalashilin. Quantum Dynamics Simulations Of Energy Redistribution In HO-SO_2. *Chem. Phys.*, 370:223–231, 2010. doi: 10.1016/j.chemphys.2010.02.010.

[67] D. V. Shalashilin and M. S. Child. Basis Set Sampling In The Method Of Coupled Coherent States: Coherent State Swarms, Trains, And Pancakes. *J. Chem. Phys.*, 128(5):54102, 2008. doi: 10.1063/1.2828509.

[68] D. V. Shalashilin, M. S. Child, and D. C. Clary. Quantum Initial Value Representation Simulation Of Water Trimer Far Infrared Absorption Spectrum. *J. Chem. Phys.*, 120(12):5608–5615, 2004. doi: 10.1063/1.1650299.

[69] D. V. Shalashilin and M. S. Child. Real Time Quantum Propagation On A Monte Carlo Trajectory Guided Grids Of Coupled Coherent States: 26D Simulation Of Pyrazine Absorption Spectrum. *J. Chem. Phys.*, 121(8):3563–3568, 2004. doi: 10.1063/1.1776111.

[70] H.-D. Meyer and W. H. Miller. Classical Models For Electronic Degrees Of Freedom: Derivation Via Spin Analogy And Application To $F*+H_2 \rightarrow F+H_2$. *J. Chem. Phys.*, 71(5):2156–2169, 1979. doi: 10.1063/1.438598.

[71] D. V. Shalashilin and M. S. Child. Electronic Energy Levels With The Help Of Trajectory-Guided Random Grid Of Coupled Wave Packets. I. Six-Dimensional Simulation Of H_2. *J. Chem. Phys.*, 122(22):224108, 2005. doi: 10.1063/1.1926268.

[72] T. J. Martínez and R. D. Levine. First-Principles Molecular Dynamics On Multiple Electronic States: A Case Study Of NaI. *J. Chem. Phys.*, 105(15):6334–6341, 1996. doi: 10.1063/1.472486.

[73] M. Ben-Nun, T. J. Martínez, and R. D. Levine. Dynamical Stereochemistry On Several Electronic States: A Computational Study Of Na* + H_2. *J. Phys. Chem. A*, 101(41):7522–7529, 1997. doi: 10.1021/jp971058b.

[74] D. V. Shalashilin and M. S. Child. A Version Of Diffusion Monte Carlo Method Based On Random Grids Of Coherent States. II. Six-Dimensional Simulation Of Electronic States Of H_2. *J. Chem. Phys.*, 122(22):224109, 2005. doi: 10.1063/1.1926269.

[75] A. Kirrander and D. V. Shalashilin. Quantum Dynamics With Fermion Coupled Coherent States: Theory And Application To Electron Dynamics In Laser Fields. *Phys. Rev. A*, 84(3):33406, 2011. doi: 10.1103/PhysRevA.84.033406.

[76] Michal Ben-Nun and Todd. J. Martnez. *Ab Initio Quantum Molecular Dynamics*, pages 439–512. John Wiley & Sons, Inc., 2002. ISBN 9780471264316. doi: 10.1002/0471264318.ch7.

[77] G. D. Billing. On The Use Of Ehrenfest's Theorem In Molecular Scattering. *Chem. Phys. Lett.*, 100(6):535–539, 1983. doi: 10.1016/0009-2614(83)87423-5.

[78] R. Martinazzo, M. Nest, P. Saalfrank, and G. F. Tantardini. A Local Coherent-State Approximation To System-Bath Quantum Dynamics. *J. Chem. Phys.*, 125 (19):194102, 2006. doi: 10.1063/1.2362821.

[79] P. Kramer and M. Saraceno. TDVP For Non-Normalised States. In P. Kramer and M. Saraceno, editors, *Geom. Time-Dependent Var. Princ. Quantum Mech.*, volume 140 of *Lecture Notes in Physics*, pages 3–14. Springer Berlin Heidelberg, 1981. ISBN 978-3-540-10579-4. doi: 10.1007/3-540-10579-4_19.

[80] G. A. Worth, H.-D. Meyer, and L. S. Cederbaum. Relaxation Of A System With A Conical Intersection Coupled To A Bath: A Benchmark 24-Dimensional Wave Packet Study Treating The Environment Explicitly. *J. Chem. Phys.*, 109(9):3518–3529, 1998. doi: 10.1063/1.476947.

[81] A. Raab, G. A. Worth, H.-D. Meyer, and L. S. Cederbaum. Molecular Dynamics Of Pyrazine After Excitation To The S_2 Electronic State Using A Realistic 24-Mode Model Hamiltonian. *J. Chem. Phys.*, 110(2):936–946, 1999. doi: 10.1063/1.478061.

[82] H. Wang. Basis Set Approach To The Quantum Dissipative Dynamics: Application Of The Multiconfiguration Time-Dependent Hartree Method To The Spin-Boson Problem. *J. Chem. Phys.*, 113(22):9948–9956, 2000. doi: 10.1063/1.1323746.

[83] H. Wang and M. Thoss. Multilayer Formulation Of The Multiconfiguration Time-Dependent Hartree Theory. *J. Chem. Phys.*, 119(3):1289–1299, 2003. doi: 10.1063/1.1580111.

[84] S.-Y. Ye, D. V. Shalashilin, and A. Serafini. Modeling Of Quantum-Information Processing With Ehrenfest Guided Trajectories: A Case Study With Spin-Boson-Like Couplings. *Phys. Rev. A*, 86(3):32312, 2012. doi: 10.1103/PhysRevA.86.032312.

[85] S. Mahapatra, G. A. Worth, H.-D. Meyer, L. S. Cederbaum, and H. Köppel. The $\widetilde{A}^2E/\widetilde{B}^2B_2$ Photoelectron Bands Of Allene Beyond The Linear Coupling Scheme: An Ab Initio Dynamical Study Including All Fifteen Vibrational Modes. *J. Phys. Chem. A*, 105(23):5567–5576, 2001. doi: 10.1021/jp003727i.

[86] H. Köppel, M. Döscher, I. Bâldea, H.-D. Meyer, and P. G. Szalay. Multistate Vibronic Interactions In The Benzene Radical Cation. II. Quantum Dynamical Simulations. *J. Chem. Phys.*, 117(6):2657–2671, 2002. doi: 10.1063/1.1491398.

[87] L. J. Doriol, F. Gatti, C. Iung, and H.-D. Meyer. Computation Of Vibrational Energy Levels And Eigenstates Of Fluoroform Using The Multiconfiguration Time-Dependent Hartree Method. *J. Chem. Phys*, 129(22):224109, 2008. doi: 10.1063/1.3020716.

[88] F. Matzkies and U. Manthe. Accurate Reaction Rate Calculations Including Internal And Rotational Motion: A Statistical Multi-Configurational Time-Dependent Hartree Approach. *J. Chem. Phys.*, 110(1):88–96, 1999. doi: 10.1063/1.478128.

[89] M. D. Coutinho-Neto, A. Viel, and U. Manthe. The Ground State Tunneling Splitting Of Malonaldehyde: Accurate Full Dimensional Quantum Dynamics Calculations. *J. Chem. Phys.*, 121(19):9207–9210, 2004. doi: 10.1063/1.1814356.

[90] O. E. Alon, A. I. Streltsov, and L. S. Cederbaum. Multiconfigurational Time-Dependent Hartree Method For Bosons: Many-body Dynamics Of Bosonic Systems. *Phys. Rev. A*, 77(3):33613, 2008. doi: 10.1103/PhysRevA.77.033613.

[91] A. I. Streltsov, O. E. Alon, and L. S. Cederbaum. Formation And Dynamics Of Many-Boson Fragmented States In One-Dimensional Attractive Ultracold Gases. *Phys. Rev. Lett.*, 100(13):130401, 2008. doi: 10.1103/PhysRevLett.100.130401.

[92] J. Zanghellini, M. Kitzler, C. Fabian, T. Brabec, and A. Scrinzi. An MCTDHF Approach To Multielectron Dynamics In Laser Fields. *Laser Phys.*, 13(8):1064–1068, 2003.

[93] T. Kato and H. Kono. Time-Dependent Multiconfiguration Theory For Electronic Dynamics Of Molecules In An Intense Laser Field. *Chem. Phys. Lett.*, 392(4-6):533–540, 2004. doi: 10.1016/j.cplett.2004.05.106.

[94] M. Nest, T. Klamroth, and P. Saalfrank. The Multiconfiguration Time-Dependent Hartree-Fock Method For Quantum Chemical Calculations. *J. Chem. Phys.*, 122 (12):124102, 2005. doi: 10.1063/1.1862243.

[95] K. Sakmann, A. I. Streltsov, O. E. Alon, and L. S. Cederbaum. Exact Quantum Dynamics Of A Bosonic Josephson Junction. *Phys. Rev. Lett.*, 103(22):220601, 2009. doi: 10.1103/PhysRevLett.103.220601.

[96] A. I. Streltsov, O. E. Alon, and L. S. Cederbaum. Swift Loss Of Coherence of Soliton Trains in Attractive Bose-Einstein Condensates. *Phys. Rev. Lett.*, 106(24):240401, 2011. doi: 10.1103/PhysRevLett.106.240401.

[97] M. Kitzler, J. Zanghellini, Ch. Jungreuthmayer, M. Smits, A. Scrinzi, and T. Brabec. Ionization Dynamics Of Extended Multielectron Systems. *Phys. Rev. A*, 70(4):41401, 2004. doi: 10.1103/PhysRevA.70.041401.

[98] J. Caillat, J. Zanghellini, M. Kitzler, O. Koch, W. Kreuzer, and A. Scrinzi. Correlated Multielectron Systems In Strong Laser Fields: A Multiconfiguration Time-Dependent Hartree-Fock Approach. *Phys. Rev. A*, 71(1):12712, 2005. doi: 10.1103/PhysRevA.71.012712.

[99] G. Jordan, J. Caillat, C. Ede, and A. Scrinzi. Strong Field Ionization Of Linear Molecules: A Correlated Three-Dimensional Calculation. *J. Phys. B*, 39(13):S341, 2006. doi: 0.1088/0953-4075/39/13/S07.

[100] F. Remacle, M. Nest, and R. D. Levine. Laser Steered Ultrafast Quantum Dynamics Of Electrons In LiH. *Phys. Rev. Lett.*, 99(18):183902, 2007. doi: 10.1103/PhysRevLett.99.183902.

[101] M. Nest. Quantum Carpets And Correlated Dynamics Of Several Fermions. *Phys. Rev. A*, 73(2):23613, 2006. doi: 10.1103/PhysRevA.73.023613.

[102] M. Nest, R. Padmanaban, and P. Saalfrank. Time-Dependent Approach To Electronically Excited States Of Molecules With The Multiconfiguration Time-Dependent Hartree-Fock Method. *J. Chem. Phys.*, 126(21):214106, 2007. doi: 10.1063/1.2743007.

[103] O. E. Alon, A. I. Streltsov, K. Sakmann, A. U. J. Lode, J. Grond, and L. S. Cederbaum. Recursive Formulation Of The Multiconfigurational Time-Dependent Hartree Method For Fermions, Bosons And Mixtures Thereof In Terms Of One-Body Density Operators. *Chem. Phys.*, 401(0):2–14, 2012. doi: 10.1016/j.chemphys.2011.09.026.

[104] M. Schroder and A. Brown. Realization Of The C-NOT Quantum Gate Operation In Six-Dimensional Ammonia Using The OCT-MCTDH Approach. *J. Chem. Phys.*, 131(3):34101, 2009. doi: 10.1063/1.3168438.

[105] H. Wang and M. Thoss. Quantum-Mechanical Evaluation Of The Boltzmann Operator In Correlation Functions For Large Molecular Systems: A Multilayer Multiconfiguration Time-Dependent Hartree Approach. *J. Chem. Phys.*, 124(3): 34114, 2006. doi: 10.1063/1.2161178.

[106] I. R. Craig, M. Thoss, and H. Wang. Proton Transfer Reactions In Model Condensed-Phase Environments: Accurate Quantum Dynamics Using The Multilayer Multiconfiguration Time-Dependent Hartree Approach. *J. Chem. Phys.*, 127 (14):144503, 2007. doi: 10.1063/1.2772265.

[107] I. Kondov, M. Čížek, C. Benesch, H. Wang, and M. Thoss. Quantum Dynamics Of Photoinduced Electron-Transfer Reactions In Dye-Semiconductor Systems: First-Principles Description And Application To Coumarin 343-TiO_2. *J. Phys. Chem. C*, 111(32):11970–11981, 2007. doi: 10.1021/jp072217m.

[108] H. Wang and M. Thoss. From Coherent Motion To Localization: Dynamics Of The Spin-Boson Model At Zero Temperature. *New J. Phys.*, 10(11):115005, 2008. doi: 10.1088/1367-2630/10/11/115005.

[109] I. Burghardt, H.-D. Meyer, and L. S. Cederbaum. Approaches To The Approximate Treatment Of Complex Molecular Systems By The Multiconfiguration Time-Dependent Hartree Method. *J. Chem. Phys.*, 111(7):2927–2939, 1999. doi: 10.1063/1.479574.

[110] G. A. Worth and I. Burghardt. Full Quantum Mechanical Molecular Dynamics Using Gaussian Wavepackets. *Chem. Phys. Lett.*, 368(3-4):502–508, 2003. doi: 10.1016/S0009-2614(02)01920-6.

[111] S. Romer, M. Ruckenbauer, and I. Burghardt. Gaussian-Based Multiconfiguration Time-Dependent Hartree: A Two-Layer Approach. I. Theory. *J. Chem. Phys.*, 138(6):64106, 2013. doi: 10.1063/1.4788830.

[112] I. Burghardt, M. Nest, and G. A. Worth. Multiconfigurational System-Bath Dynamics Using Gaussian Wave Packets: Energy Relaxation And Decoherence Induced By A Finite-Dimensional Bath. *J. Chem. Phys.*, 119(11):5364–5378, 2003. doi: 10.1063/1.1599275.

[113] I. Burghardt, K. Giri, and G. A. Worth. Multimode Quantum Dynamics Using Gaussian Wavepackets: The Gaussian-Based Multiconfiguration Time-Dependent Hartree (G-MCTDH) Method Applied To The Absorption Spectrum Of Pyrazine. *J. Chem. Phys.*, 129(17):174104, 2008. doi: 10.1063/1.2996349.

[114] G. A. Worth, M. A. Robb, and I. Burghardt. A Novel Algorithm For Non-Adiabatic Direct Dynamics Using Variational Gaussian Wavepackets. *Faraday Discuss.*, 127 (0):307–323, 2004. doi: 10.1039/B314253A.

[115] G. A. Worth, M. A. Robb, and B. Lasorne. Solving The Time-Dependent Schrodinger Equation For Nuclear Motion In One Step: Direct Dynamics Of Non-Adiabatic Systems. *Mol. Phys.*, 106(16-18):2077–2091, 2008. doi: 10.1080/00268970802172503.

[116] D. V. Shalashilin and I. Burghardt. Gaussian-Based Techniques For Quantum Propagation From The Time-dependent Variational Principle: Formulation In

Terms Of Trajectories Of Coupled Classical And Quantum Variables. *J. Chem. Phys.*, 129(8):84104, 2008. doi: 10.1063/1.2969101.

[117] M. Ronto and D. V. Shalashilin. Numerical Implementation And Test Of The Modified Variational Multiconfigurational Gaussian Method For High-Dimensional Quantum Dynamics. *J. Phys. Chem. A*, 117(32):6948–6959, 2013. doi: 10.1021/jp310976d.

[118] M. Ben-Nun, T. J. Martínez, and R. D. Levine. Multiple Traversals Of A Conical Intersection: Electronic Quenching In Na* + H_2. *Chem. Phys. Lett.*, 270(3-4):319–326, 1997. doi: 10.1016/S0009-2614(97)00369-2.

[119] J. C. Tully. Molecular Dynamics With Electronic Transitions. *J. Chem. Phys.*, 93(2):1061–1071, 1990. doi: 10.1063/1.459170.

[120] T. J. Martínez, M. Ben-Nun, and G. Ashkenazi. Classical/Quantal Method For Multistate Dynamics: A Computational Study. *J. Chem. Phys.*, 104(8):2847–2856, 1996. doi: 10.1063/1.471108.

[121] M. D. Hack, A. M. Wensmann, D. G. Truhlar, M. Ben-Nun, and T. J. Martínez. Comparison Of Full Multiple Spawning, Trajectory Surface Hopping, And Converged Quantum Mechanics For Electronically Nonadiabatic Dynamics. *J. Chem. Phys.*, 115(3):1172–1186, 2001. doi: 10.1063/1.1377030.

[122] M. Ben-Nun and T. J. Martínez. A Multiple Spawning Approach To Tunneling Dynamics. *J. Chem. Phys.*, 112(14):6113–6121, 2000. doi: 10.1063/1.481213.

[123] T. J. Martínez and R. D. Levine. Non-Adiabatic Molecular Dynamics: Split-Operator Multiple Spawning With Applications To Photodissociation. *J. Chem. Soc., Faraday Trans.*, 93(5):941–947, 1997. doi: 10.1039/A605958I.

[124] M. Ben-Nun and T. J. Martínez. Exploiting Temporal Nonlocality To Remove Scaling Bottlenecks In Nonadiabatic Quantum Dynamics. *J. Chem. Phys.*, 110(9):4134–4140, 1999. doi: 10.1063/1.478297.

[125] B. G. Levine and T. J. Martínez. Ab Initio Multiple Spawning Dynamics Of Excited Butadiene: Role Of Charge Transfer. *J. Phys. Chem. A*, 113(46):12815–12824, 2009. doi: 10.1021/jp907111u.

[126] T. J. Martínez. Insights For Light-Driven Molecular Devices From Ab Initio Multiple Spawning Excited-State Dynamics Of Organic And Biological Chromophores. *Acc. Chem. Res.*, 39(2):119–126, 2006. doi: 10.1021/ar040202q.

[127] B. G. Levine, J. D. Coe, A. M. Virshup, and T. J. Martínez. Implementation Of Ab Initio Multiple Spawning In The MolPro Quantum Chemistry Package. *Chem. Phys.*, 347(1-3):3–16, 2008. doi: 10.1016/j.chemphys.2008.01.014.

[128] A. L. Thompson, C. Punwong, and T. J. Martínez. Optimization Of Width Parameters For Quantum Dynamics With Frozen Gaussian Basis Sets. *Chem. Phys.*, 370(13):70–77, 2010. doi: 10.1016/j.chemphys.2010.03.020.

[129] S. Yang, J. D. Coe, B. Kaduk, and T. J. Martínez. An "Optimal" Spawning Algorithm For Adaptive Basis Set Expansion In Nonadiabatic Dynamics. *J. Chem. Phys.*, 130(13):134113, 2009. doi: 10.1063/1.3103930.

[130] U. Weiss. *Quantum Dissipative Systems*. World Scientific, Singapore, 2nd edition, January 1999.

[131] A. J. Leggett, S. Chakravarty, A. T. Dorsey, M. P. A. Fisher, A. Garg, and W. Zwerger. Dynamics Of The Dissipative Two-State System. *Rev. Mod. Phys.*, 59(1): 1–85, January 1987. doi: 10.1103/RevModPhys.59.1.

[132] M. Thorwart, E. Paladino, and M. Grifoni. Dynamics Of The Spin-boson Model With A Structured Environment. *Chem. Phys.*, 296(2–3):333–344, 2004. doi: 10.1016/j.chemphys.2003.10.007.

[133] M. Thoss, W. Domcke, and H. Wang. Theoretical Study Of Vibrational Wave-Packet Dynamics In Electron-transfer Systems. *Chem. Phys.*, 296(23):217–229, 2004. ISSN 0301-0104. doi: 10.1016/j.chemphys.2003.08.029.

[134] G. Stock. A Semiclassical Self-Consistent-Field Approach To Dissipative Dynamics: The Spin-Boson Problem. *J. Chem. Phys*, 103(4):1561–1573, 1995. doi: 10.1063/1.469778.

[135] K. LeHur. Quantum Phase Transitions In Spin-Boson Systems: Dissipation And Light Phenomena. In Lincoln Carr, editor, *Underst. Quantum Phase Transitions*, pages 217–240. CRC Press, 2010.

[136] F. K. Wilhelm, S. Kleff, and J. von Delft. The Spin-Boson Model With A Structured Environment: A Comparison Of Approaches. *Chem. Phys.*, 296(2–3):345–353, 2004. ISSN 0301-0104. doi: 10.1016/j.chemphys.2003.10.010.

[137] K. Le Hur. Coulomb Blockade Of A Noisy Metallic Box: A Realization of Bose-Fermi Kondo Models. *Phys. Rev. Lett.*, 92(19):196804, May 2004. doi: 10.1103/PhysRevLett.92.196804.

[138] U. Weiss, H. Grabert, and S. Linkwitz. Influence Of Friction And Temperature On Coherent Quantum Tunneling. *J. Low Temp. Phys.*, 68(3-4):213–244, 1987. doi: 10.1007/BF00683900.

[139] A. Suárez and R. Silbey. Properties Of A Macroscopic System As A Thermal Bath. *J. Chem. Phys*, 95(12):9115–9121, 1991. doi: 10.1063/1.461190.

[140] N. Makri. Quantum Dissipative Dynamics: A Numerically Exact Methodology. *J. Phys. Chem. A*, 102(24):4414–4427, 1998. doi: 10.1021/jp980359y.

[141] R. A. Marcus and N. Sutin. Electron Transfers In Chemistry And Biology. *Biochim. Biophys. Acta*, 811(3):265–322, 1985. ISSN 0304-4173. doi: 10.1016/0304-4173(85)90014-X.

[142] M. Thoss, H. Wang, and W. H. Miller. Self-Consistent Hybrid Approach For Complex Systems: Application To The Spin-Boson Model With Debye Spectral Density. *J. Chem. Phys*, 115(7):2991–3005, 2001. doi: 10.1063/1.1385562.

[143] A G Redfield. The Theory Of Relaxation Processes. In John S Waugh, editor, *Adv. Mag. Reson.*, volume 1 of *Advances in Magnetic and Optical Resonance*, pages 1–32. Academic Press, 1965. doi: 10.1016/B978-1-4832-3114-3.50007-6.

[144] J. M. Jean, R. A. Friesner, and G. R. Fleming. Application Of A Multilevel Redfield Theory To Electron Transfer In Condensed Phases. *J. Chem. Phys*, 96 (8):5827–5842, 1992. doi: 10.1063/1.462858.

[145] A. K. Felts, W. T. Pollard, and R. A. Friesner. Multilevel Redfield Treatment of Bridge-Mediated Long-Range Electron Transfer: A Mechanism for Anomalous Distance Dependence. *J. Phys. Chem.*, 99(9):2929–2940, 1995. doi: 10.1021/j100009a057.

[146] D. Egorova, A. Kühl, and W. Domcke. Modeling Of Ultrafast Electron-transfer Dynamics: Multi-Level Redfield Theory And Validity Of Approximations. *Chem. Phys.*, 268(13):105–120, 2001. doi: 10.1016/S0301-0104(01)00293-2.

[147] D. Egorova, M. Thoss, W. Domcke, and H. Wang. Modeling Of Ultrafast Electron-Transfer Processes: Validity Of Multilevel Redfield Theory. *J. Chem. Phys*, 119 (5):2761–2773, 2003. doi: 10.1063/1.1587121.

[148] J. C. Tully and R. K. Preston. Trajectory Surface Hopping Approach To Nonadiabatic Molecular Collisions: The Reaction Of H^+ with D_2. *J. Chem. Phys*, 55(2): 562–572, 1971. doi: 10.1063/1.1675788.

[149] G. D. Billing. On The Applicability Of The Classical Trajectory Equations In Inelastic Scattering Theory. *Chem. Phys. Lett.*, 30(3):391–393, 1975. ISSN 0009-2614. doi: 10.1016/0009-2614(75)80014-5.

[150] D. A. Micha. A Self-Consistent Eikonal Treatment Of Electronic Transitions In Molecular Collisions. *J. Chem. Phys*, 78(12):7138–7145, 1983. doi: 10.1063/1.444753.

[151] G. D. Billing. Quantum Corrections To The Classical Path Theory. *J. Chem. Phys*, 99(8):5849–5857, 1993. doi: 10.1063/1.465938.

[152] M. F. Herman. Generalization Of The Geometric Optical Series Approach For Nonadiabatic Scattering Problems. *J. Chem. Phys*, 76(6):2949–2958, 1982. doi: 10.1063/1.443388.

[153] S. Chapman. The Classical Trajectory-Surface-Hopping Approach To Charge-Transfer Processes. In Michael Baer and Cheuk-Yiu Ng, editors, *Adv. Chem. Phys.*, number 82, chapter 7, pages 423–483. John Wiley & Sons, Inc., 1992. ISBN 9780470141403. doi: 10.1002/9780470141403.ch7.

[154] D. F. Coker and L. Xiao. Methods For Molecular Dynamics With Nonadiabatic Transitions. *J. Chem. Phys*, 102(1):496–510, 1995. doi: 10.1063/1.469428.

[155] R. Egger and U. Weiss. Quantum Monte Carlo Simulation Of The Dynamics Of The Spin-Boson Model. *Z. Phys. B*, 89(1):97–107, 1992. ISSN 0722-3277. doi: 10.1007/BF01320834.

[156] R. Egger and C. H. Mak. Low-temperature Dynamical Simulation Of Spin-Boson Systems. *Phys. Rev. B*, 50(20):15210–15220, November 1994. doi: 10.1103/PhysRevB.50.15210.

[157] N. Makri and D. E. Makarov. Tensor Propagator For Iterative Quantum Time Evolution Of Reduced Density Matrices. I. Theory. *J. Chem. Phys*, 102(11):4600–4610, 1995. doi: 10.1063/1.469508.

[158] E. Sim and N. Makri. Path Integral Simulation Of Charge Transfer Dynamics In Photosynthetic Reaction Centers. *J. Phys. Chem. B*, 101(27):5446–5458, 1997. doi: 10.1021/jp970707g.

[159] L. Mühlbacher and R. Egger. Crossover From Nonadiabatic To Adiabatic Electron Transfer Reactions: Multilevel Blocking Monte Carlo Simulations. *J. Chem. Phys*, 118(1):179–191, 2003. doi: 10.1063/1.1523014523014.

[160] R. P. Feynman and F. L. Vernon Jr. The Theory Of A General Quantum System Interacting With A Linear Dissipative System. *Ann. Phys.*, 24:118–173, 1963. ISSN 0003-4916. doi: 10.1016/0003-4916(63)90068-X.

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

[162] B. Bruognolo, A. Weichselbaum, C. Guo, J. von Delft, I. Schneider, and M. Vojta. Two-Bath Spin-Boson Model: Phase Diagram And Critical Properties. *Phys. Rev. B*, 90(24):245130, December 2014. doi: 10.1103/PhysRevB.90.245130.

[163] H. Wang, M. Thoss, and W. H. Miller. Forward-Backward Initial Value Representation For The Calculation Of Thermal Rate Constants For Reactions In Complex Molecular Systems. *J. Chem. Phys*, 112(1):47–55, 2000. doi: 10.1063/1.480560.

[164] C. Symonds, J. Wu, M. Ronto, C. Zagoya, C. de Morisson Faria, and D. V. Shalashilin. Coupled-Coherent-States Approach For High-Order Harmonic Generation. *Phys. Rev. A*, 91(2):23427, February 2015. doi: 10.1103/PhysRevA.91.023427.

[165] N. A. Papadogiannis, B. Witzel, C. Kalpouzos, and D. Charalambidis. Observation Of Attosecond Light Localization In Higher Order Harmonic Generation. *Phys. Rev. Lett.*, 83(21):4289–4292, 1999. doi: 10.1103/PhysRevLett.83.4289.

[166] O. Smirnova and M. Ivanov. Multielectron High Harmonic Generation: Simple Man On Complex Plane. In Thomas Schultz and Marc Vrakking, editors, *Attosecond XUV Spectrosc. Ultrafast Dyn. Spectrosc.*, pages 201–256. Wiley, 2014. ISBN 978-3-527-41124-5. doi: 10.1002/9783527677689.

[167] K. L. Ishikawa. High-Harmonic Generation. In Mikhail Grishin, editor, *Adv. Solid-State Lasers Dev. Appl.*, pages 439–464. InTech Europe, 2010. ISBN 978-953-7619-80-0.

[168] J. C. Baggesen and L. B. Madsen. On The Dipole, Velocity And Acceleration Forms In High-Order Harmonic Generation From A Single Atom Or Molecule. *J. Phys. B At. Mol. Opt. Phys.*, 44(11):115601, June 2011. ISSN 0953-4075. doi: 10.1088/0953-4075/44/11/115601.

[169] M. Protopapas, D. G. Lappas, C. H. Keitel, and P. L. Knight. Recollisions, Bremsstrahlung, And Attosecond Pulses From Intense Laser Fields. *Phys. Rev. A*, 53(5):R2933–R2936, 1996. doi: 10.1103/PhysRevA.53.R2933.

[170] J. L. Krause, K. J. Schafer, and K. C. Kulander. High-Order Harmonic Generation From Atoms And Ions In The High Intensity Regime. *Phys. Rev. Lett.*, 68(24):3535–3538, 1992. doi: 10.1103/PhysRevLett.68.3535.

[171] P. B. Corkum. Plasma Perspective On Strong Field Multiphoton Ionization. *Phys. Rev. Lett.*, 71(13):1994–1997, 1993. doi: 10.1103/PhysRevLett.71.1994.

[172] K. J. Schafer, B. Yang, L. F. DiMauro, and K. C. Kulander. Above Threshold Ionization Beyond The High Harmonic Cutoff. *Phys. Rev. Lett.*, 70(11):1599–1602, 1993. doi: 10.1103/PhysRevLett.70.1599.

[173] K. C. Kulander, K. J. Schafer, and J. L. Krause. Dynamics Of Short-Pulse Excitation, Ionization And Harmonic Conversion. In B Piraux, A L'Huillier, and K Rzazewski, editors, *Superintense Laser-Atom Phys.*, volume 316 of *NATO Advanced Study Institute Series B: Physics*, pages 95–110. Plenum, New York, 1993. doi: 10.1007/978-1-4615-7963-2_10.

[174] F. Grossmann. *Theoretical Femtosecond Physics*. Springer-Verlag, Berlin, 2008. ISBN 978-3-540-77894-7.

[175] T. Kreibich, M. Lein, V. Engel, and E. K. U. Gross. Even-Harmonic Generation Due To Beyond-Born-Oppenheimer Dynamics. *Phys. Rev. Lett.*, 87(10):103901, August 2001. ISSN 0031-9007. doi: 10.1103/PhysRevLett.87.103901.

[176] G. van de Sand and J. Rost. Irregular Orbits Generate Higher Harmonics. *Phys. Rev. Lett.*, 83(3):524–527, 1999. doi: 10.1103/PhysRevLett.83.524.

[177] C. Zagoya, C.-M. Goletz, F. Grossmann, and J.-M. Rost. Dominant-Interaction Hamiltonians For High-Order Harmonic Generation In Laser-Assisted Collisions. *Phys. Rev. A*, 85(4):41401, 2012. doi: 10.1103/PhysRevA.85.041401.

[178] C. Zagoya, C.-M. Goletz, F. Grossmann, and J.-M. Rost. An Analytical Approach To High Harmonic Generation. *New J. Phys.*, 14(9):93050, 2012. doi: 10.1088/1367-2630/14/9/093050.

[179] S. Baier, C. Ruiz, L. Plaja, and A. Becker. Nonsequential Double Ionization Of The Hydrogen Molecule In A Few-Cycle Laser Pulse. *Phys. Rev. A*, 74(3):33405, 2006. doi: 10.1103/PhysRevA.74.033405.

[180] S. Baier, C. Ruiz, L. Plaja, and A. Becker. Single And Double Ionization Of The Hydrogen Molecule In An Intense Few-cycle Laser Pulse. *Laser Phys.*, 17(4):358–367, 2007. doi: 10.1134/S1054660X07040111.

[181] J. Parker, B. Doherty, K. Taylor, K. Schultz, C. Blaga, and L. F. DiMauro. High-Energy Cutoff In The Spectrum Of Strong-Field Nonsequential Double Ionization. *Phys. Rev. Lett.*, 96(13):133001, 2006. doi: 10.1103/PhysRevLett.96.133001.

[182] F. Lepine, M. Ivanov, and M. J. J. Vrakking. Attosecond Molecular Dynamics: Fact Or Fiction? *Nat. Photon.*, 8(3):195–204, 2014. ISSN 1749-4885. doi: 10.1038/nphoton.2014.25.

[183] J. A. Pérez-Hernández, J. Ramos, L. Roso, and L. Plaja. Harmonic Generation Beyond The Strong-field Approximation: Phase And Temporal Description. *Laser Phys.*, 20(5):1044–1050, 2010. ISSN 1054-660X. doi: 10.1134/S1054660X10090069.

[184] M. Gaarde and K. J. Schafer. Quantum Path Distributions For High-Order Harmonics In Rare Gas Atoms. *Phys. Rev. A*, 65(3):31406, 2002. doi: 10.1103/PhysRevA.65.031406.

[185] L. V. Keldysh. Ionization In The Field Of A Strong Electromagnetic Wave. *Sov. Phys. JETP*, 20:1307–1314, 1965.

[186] F. H. M. Faisal. Multiple Absorption Of Laser Photons By Atoms. *J. Phys. B*, 6 (4):L89, 1973. doi: 10.1088/0022-3700/6/4/011.

[187] H. R. Reiss. Effect Of An Intense Electromagnetic Field On A Weakly Bound System. *Phys. Rev. A*, 22(5):1786–1813, 1980. doi: 10.1103/PhysRevA.22.1786.

[188] O. Smirnova, M. Spanner, and M. Ivanov. Anatomy Of Strong Field Ionization II: To Dress Or Not To Dress? *J. Mod. Opt.*, 54(7):1019–1038, 2007. doi: 10.1080/09500340701234656.

[189] C. de Morisson Faria. High-Order Harmonic Generation In Diatomic Molecules: A Quantum-Orbit Analysis Of The Interference Patterns. *Phys. Rev. A*, 76(4): 43407, 2007. doi: 10.1103/PhysRevA.76.043407.

[190] S. V. Popruzhenko, G. G. Paulus, and D. Bauer. Coulomb-Corrected Quantum Trajectories In Strong-Field Ionization. *Phys. Rev. A*, 77(5):53409, 2008. doi: 10.1103/PhysRevA.77.053409.

[191] T.-M. Yan, S. V. Popruzhenko, M. J. J. Vrakking, and D. Bauer. Low-Energy Structures In Strong Field Ionization Revealed By Quantum Orbits. *Phys. Rev. Lett.*, 105(25):253002, 2010. doi: 10.1103/PhysRevLett.105.253002.

[192] O. Smirnova, M. Spanner, and M. Ivanov. Coulomb And Polarization Effects In Sub-Cycle Dynamics Of Strong-Field Ionization. *J. Phys. B*, 39(13):S307, 2006. doi: 10.1088/0953-4075/39/13/S05.

[193] O. Smirnova, M. Spanner, and M. Ivanov. Analytical Solutions For Strong Field Driven Atomic And Molecular One- And Two-Electron Continua And Applications To Strong-Field Problems. *Phys. Rev. A*, 77(3):33407, 2008. doi: 10.1103/PhysRevA.77.033407.

[194] L. Plaja and J. A. Pérez-Hernández. A Quantitative S-matrix Approach To High-Order Harmonic Generation From Multiphoton To Tunneling Regimes. *Opt. Express*, 15(7):3629–3634, 2007. doi: 10.1364/OE.15.003629.

[195] J. Wu, B. B. Augstein, and C. de Morisson Faria. Bohmian-Trajectory Analysis Of High-Order Harmonic Generation: Ensemble Averages, Nonlocality, And Quantitative Aspects. *Phys. Rev. A*, 88(6):63416, 2013. doi: 10.1103/PhysRevA.88.063416.

[196] Eugene P. Wigner. Resonance reactions and anomalous scattering. *Phys. Rev.*, 70:15–33, Jul 1946. doi: 10.1103/PhysRev.70.15.

[197] E. P. Wigner and L. Eisenbud. Higher angular momenta and long range interaction in resonance reactions. *Phys. Rev.*, 72:29–41, Jul 1947. doi: 10.1103/PhysRev.72.29.

[198] P G Burke, A Hibbert, and W D Robb. Electron scattering by complex atoms. *Journal of Physics B: Atomic and Molecular Physics*, 4(2):153, 1971. doi: 10.1088/0022-3700/4/2/002.

[199] P.G. Burke and W.D. Robb. The r-matrix theory of atomic processes. volume 11 of *Adv. At. Mol. Phys.*, pages 143 – 214. Academic Press, 1976. doi: 10.1016/S0065-2199(08)60030-5.

[200] P G Burke and V M Burke. Time-dependent r -matrix theory of multiphoton processes. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 30(11):L383, 1997. doi: 10.1088/0953-4075/30/11/002.

[201] M. A. Lysaght, H. W. van der Hart, and P. G. Burke. Time-dependent r-matrix theory for ultrafast atomic processes. *Phys. Rev. A*, 79:053411, May 2009. doi: 10.1103/PhysRevA.79.053411.

[202] A. C. Brown and H. W. van der Hart. Influence of multiple ionization thresholds on harmonic generation from ar^+. *Phys. Rev. A*, 86:063416, Dec 2012. doi: 10.1103/PhysRevA.86.063416.

[203] A. C. Brown, S. Hutchinson, M. A. Lysaght, and H. W. van der Hart. Interference between competing pathways in atomic harmonic generation. *Phys. Rev. Lett.*, 108:063006, Feb 2012. doi: 10.1103/PhysRevLett.108.063006.

[204] A. C. Brown and H. W. van der Hart. Enhanced harmonic generation from ar^+ aligned with $m = 1$. *Phys. Rev. A*, 88:033419, Sep 2013. doi: 10.1103/PhysRevA.88.033419.

[205] O. Hassouneh, A. C. Brown, and H. W. van der Hart. Harmonic generation by noble-gas atoms in the near-ir regime using *ab initio* time-dependent r-matrix theory. *Phys. Rev. A*, 90:043418, Oct 2014. doi: 10.1103/PhysRevA.90.043418.

[206] O. Hassouneh, A. C. Brown, and H. W. van der Hart. Multichannel interference in high-order-harmonic generation from ne^+ driven by an ultrashort intense laser pulse. *Phys. Rev. A*, 89:033409, Mar 2014. doi: 10.1103/PhysRevA.89.033409.

[207] Lisa Torlina, Felipe Morales, H G Muller, and Olga Smirnova. Ab initio verification of the analytical r-matrix theory for strong field ionization. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 47(20):204021, 2014. doi: 10.1088/0953-4075/47/20/204021.

[208] Alex G Harvey, Danilo S Brambila, Felipe Morales, and Olga Smirnova. An r-matrix approach to electron??photon??molecule collisions: photoelectron angular distributions from aligned molecules. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 47(21):215005, 2014. doi: 10.1088/0953-4075/47/21/215005.

[209] Danilo S Brambila, Alex G Harvey, Zden?k Man, Jimena D Gorfinkiel, and Olga Smirnova. The role of multichannel effects in the photoionization of the no 2 molecule: an ab initio r -matrix study. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 48(24):245101, 2015. doi: 10.1088/0953-4075/48/24/245101.

[210] A. Baltuska, Th. Udem, M. Uiberacker, M. Hentschel, E. Goulielmakis, Ch. Gohle, R. Holzwarth, V. S. Yakovlev, A. Scrinzi, T. W. Hansch, and F. Krausz. Attosecond Control Of Electronic Processes By Intense Light Fields. *Nature*, 421(6923): 611–615, 2003. doi: 10.1038/nature01414.

[211] G. Sansone, E. Benedetti, F. Calegari, C. Vozzi, L. Avaldi, R. Flammini, L. Poletto, P. Villoresi, C. Altucci, R. Velotta, S. Stagira, S. De Silvestri, and M. Nisoli. Isolated Single-Cycle Attosecond Pulses. *Science (80-.).*, 314(5798):443–446, 2006. doi: 10.1126/science.1132838.

[212] E. Goulielmakis, M. Schultze, M. Hofstetter, V. S. Yakovlev, J. Gagnon, M. Uiberacker, A. L. Aquila, E. M. Gullikson, D. T. Attwood, R. Kienberger, F. Krausz, and U. Kleineberg. Single-Cycle Nonlinear Optics. *Science (80-.).*, 320(5883): 1614–1617, 2008. doi: 10.1126/science.1157846.

[213] R. A. Bartels, A. Paul, H. Green, H. C. Kapteyn, M. M. Murnane, S. Backus, I. P. Christov, Y. Liu, D. Attwood, and C. Jacobsen. Generation Of Spatially Coherent Light At Extreme Ultraviolet Wavelengths. *Science (80-.).*, 297(5580):376–378, 2002. doi: 10.1126/science.1071718.

[214] Z. Zeng, Y. Cheng, X. Song, R. Li, and Z. Xu. Generation Of An Extreme Ultraviolet Supercontinuum In A Two-Color Laser Field. *Phys. Rev. Lett.*, 98(20): 203901, 2007. doi: 10.1103/PhysRevLett.98.203901.

[215] Y. Oishi, M. Kaku, A. Suda, F. Kannari, and K. Midorikawa. Generation Of Extreme Ultraviolet Continuum Radiation Driven By A Sub-10-fs Two-Color Field. *Opt. Express*, 14(16):7230–7237, 2006. doi: 10.1364/OE.14.007230.

[216] V. S. Yakovlev, M. Ivanov, and F. Krausz. Enhanced Phase-Matching For Generation Of Soft X-Ray Harmonics And Attosecond Pulses In Atomic Gases. *Opt. Express*, 15(23):15351–15364, 2007. doi: 10.1364/OE.15.015351.

[217] T. Pfeifer, A. Jullien, M. J. Abel, P. M. Nagel, L. Gallmann, D. M. Neumark, and S. R. Leone. Generating Coherent Broadband Continuum Soft X-Ray Radiation By Attosecond Ionization Gating. *Opt. Express*, 15(25):17120–17128, 2007. doi: 10.1364/OE.15.017120.

[218] M. Klaiber, K. Z. Hatsagortsyan, C. Müller, and C. H. Keitel. Coherent Hard X Rays From Attosecond Pulse Train-Assisted Harmonic Generation. *Opt. Lett.*, 33 (4):411–413, 2008. doi: 10.1364/OL.33.000411.

[219] S. Haessler, J. Caillat, W. Boutu, C. Giovanetti-Teixeira, T. Ruchon, T. Auguste, Z. Diveki, P. Breger, A. Maquet, B. Carre, R. Taieb, and P. Salieres. Attosecond Imaging Of Molecular Electronic Wavepackets. *Nat. Phys.*, 6(3):200–206, 2010. doi: 10.1038/nphys1511.

[220] J. Itatani, J. Levesque, D. Zeidler, H. Niikura, H. Pepin, J. C. Kieffer, P. B. Corkum, and D. M. Villeneuve. Tomographic Imaging Of Molecular Orbitals. *Nature*, 432(7019):867–871, 2004. doi: 10.1038/nature03183.

[221] J. Guo, X.-S. Liu, and S.-I. Chu. Exploration Of Strong-Field Multiphoton Double Ionization, Rescattering, And Electron Angular Distribution Of He Atoms In Intense Long-wavelength Laser Fields: The Coupled Coherent-State Approach. *Phys. Rev. A*, 82(2):23402, 2010. doi: 10.1103/PhysRevA.82.023402.

[222] C. Zagoya, J. Wu, M. Ronto, D. V. Shalashilin, and C. de Morisson Faria. Quantum And Semiclassical Phase-Space Dynamics Of A Wave Packet In Strong Fields Using Initial-Value Representations. *New J. Phys.*, 16(10):103040, 2014. doi: 10.1088/1367-2630/16/10/103040.

[223] K. Burnett, V. C. Reed, J. Cooper, and P. L. Knight. Calculation Of The Background Emitted During High-Harmonic Generation. *Phys. Rev. A*, 45(5):3347–3349, 1992. doi: 10.1103/PhysRevA.45.3347.

[224] J. L. Krause, K. J. Schafer, and K. C. Kulander. Calculation Of Photoemission From Atoms Subject To Intense Laser Fields. *Phys. Rev. A*, 45(7):4998–5010, 1992. doi: 10.1103/PhysRevA.45.4998.

[225] L. Jie, C. Shi-Gang, and B. Hu. Chaotic Behavior Of A One-dimensional Model Atom In An Intense Field. *Acta Phys. Sin. (Overseas Ed.*, 7(2):89, 1998. doi: 10.1088/1004-423X/7/2/002.

[226] S. Cocke and L. Reichl. High-Harmonic Generation In A Driven Triangular Well: The Implications Of Chaos. *Phys. Rev. A*, 53(3):1746–1750, 1996. doi: 10.1103/PhysRevA.53.1746.

[227] J. C. Burant and V. S. Batista. Real Time Path Integrals Using The Herman-Kluk Propagator. *J. Chem. Phys.*, 116(7):2748–2756, 2002. doi: 10.1063/1.1436306.

[228] D. Huber and E. J. Heller. Hybrid Mechanics: A Combination Of Classical And Quantum Mechanics. *J. Chem. Phys.*, 89(8):4752–4760, 1988. doi: 10.1063/1.455669.

[229] D. Huber and E. J. Heller. Generalized Gaussian Wave Packet Dynamics. *J. Chem. Phys.*, 87(9):5302–5311, 1987. doi: 10.1063/1.453647.

[230] N. Takemoto, A. Shimshovitz, and D. J. Tannor. Phase Space Approach To Laser-Driven Electronic Wavepacket Propagation. *J. Chem. Phys.*, 137(1):11102, 2012. doi: 10.1063/1.4732306.

[231] O. I. Tolstikhin and T. Morishita. Adiabatic Theory Of Ionization By Intense Laser Pulses: Finite-Range Potentials. *Phys. Rev. A*, 86(4):43417, 2012. doi: 10.1103/PhysRevA.86.043417.

[232] Y. Okajima, O. I. Tolstikhin, and T. Morishita. Adiabatic Theory Of High-Order Harmonic Generation: One-Dimensional Zero-Range-Potential Model. *Phys. Rev. A*, 85(6):63406, 2012. doi: 10.1103/PhysRevA.85.063406.

[233] J. Wu, B. B. Augstein, and C. de Morisson Faria. Local Dynamics In High-Order Harmonic Generation Using Bohmian Trajectories. *Phys. Rev. A*, 88(2):23415, 2013. doi: 10.1103/PhysRevA.88.023415.

[234] Yang Song, Fu-Ming Guo, Su-Yu Li, Ji-Gen Chen, Si-Liang Zeng, and Yu-Jun Yang. Investigation of the generation of high-order harmonics through bohmian trajectories. *Phys. Rev. A*, 86:033424, Sep 2012. doi: 10.1103/PhysRevA.86.033424.

[235] D. V. Shalashilin, M. S. Child, and A. Kirrander. Mechanisms Of Double Ionization In Strong Laser Field From Simulation With Coupled Coherent States: Beyond Reduced Dimensionality Models. *Chem. Phys.*, 347(1-3):257–262, 2008. doi: 10.1016/j.chemphys.2007.11.006.

[236] James A. Green, Adriano Grigolo, Miklos Ronto, and Dmitrii V. Shalashilin. A two-layer approach to the coupled coherent states method. *The Journal of Chemical Physics*, 144(2):024111, 2016. doi: 10.1063/1.4939205.

[237] S. K. Reed, M. L. Gonzalez-Martinez, J. Rubayo-Soneira, and D. V. Shalashilin. Cartesian Coupled Coherent States Simulations: Ne_nBr_2 Dissociation As A Test Case. *J. Chem. Phys.*, 134(5):54110, 2011. doi: 10.1063/1.3532407.

[238] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes In Fortran 77*, volume 1. Cambridge University Press, 2nd edition, 2006.

[239] R. Chandler and P. Northrop. Random Number Generation, 2003.

[240] H. A. Luther. An Explicit Sixth-Order Runge-Kutta Formula. *Math. Comput.*, 22:434–436, 1968.