Implementing and evaluating a fictitious electron dynamics method for the calculation of electronic structure

Christina Hester Claassens

A dissertation presented in fulfillment of the requirements for the degree

MAGISTER SCIENTAE

in the

Faculty of Natural and Agricultural Sciences

Department of Physics

at the

University of the Free State

Bloemfontein

Promoter: Dr. M.J.H. Hoffman

Co-promoter: Dr. A.P. Greeff

November 2002

The important thing is not to stop questioning. Curiosity has its own reason for existing. One cannot help but be in awe when he contemplates the mysteries of eternity, of life, of the marvelous structure of reality. It is enough if one tries merely to comprehend a little of this mystery everyday. Never lose a holy curiosity.

(Albert Einstein, 1879 - 1955)

This thesis is dedicated to my father, who devoted his life to providing his children with the best he was able to give.

Acknowledgements

Hereby I would like to express my sincere gratitude and appreciation to:

- Dr. M.J.H. Hoffman, as promoter, for his support, guidance and invaluable expertise. Especially for his patience and understanding.
- Dr. A.P. Greeff, for his useful advice and assistance.
- The lecturers at the University of the Free-State's Physics department for their support and guidance, as well as for providing me with the best physics education I could wish for.
- My mother, Ina Claassens, for giving me endless love and support. Most of all, for believing in me.
- My brother, Nico Claassens, for being there when I needed him.
- The NRF for providing me with financial support.
- The Faculty of Natural and Agricultural Sciences for financial support.
- Above all, my Heavenly Father who has blessed me with the knowledge, insight and strength to complete this study to the best of my ability.

Opsomming

Kwantum chemiese berekeninge is 'n onmisbare hulpmiddel in die berekening van elektroniese struktuur. Tog, die grootte van sisteme wat bestudeer kan word is grootliks beperk vanweë die hoogs ongewenste skaling van berekeningstyd - tipies die aantal atome in die sisteem tot die derde of vierde orde. Molekulêre dinamika berekeninge, in teenstelling, kan sisteme wat duisende atome bevat modelleer. Ongelukkig is molekulêre dinamika nie in staat om chemiese reaksies te beskryf nie - hiervoor is kwantum chemiese berekeninge nodig.

Die strewe na $O(N)$ metodes om elektroniese struktuur te bereken het gelei tot die ondersoek van 'n onlangs voorgestelde fiktiewe elektron dinamika metode vir die berekening van elektroniese struktuur wat die idempotensie van die digtheidsmatriks gebruik om eerste en tweede orde bewegingsvergelykings te ontwikkel. Hierdie vergelykings is geïmplementeer in 'n semi-empiriese omgewing, soos verskaf deur die MOPAC sagteware pakket. Die snelheids Verlet skema is gebruik om hierdie vergelyking te integreer en die afdwing van beperkings is bewerkstellig deur McWeeny suiwering en die RATTLE algoritme. Die essensiële rol wat parameters in die effektiwiteit van die bewegingsvergelykings speel, is ook ondersoek en voorstelle is gemaak vir hierdie parameters. Die vereiste van energiebehoud in die bewegingsvergelykings, tesame met die stabiliteit van die snelheids Verlet integreerder, is aangespreek en die waardes wat parameters moet hê sodat daar aan hierdie vereistes voldoen word, is hersien.

Die belangrikheid van die Si(100)2x1:H sisteem as 'n toetssisteem is benadruk en die versperrings vir waterstof atoom diffusie is bereken deur gebruik te maak van bestaande parameters verskaf deur MOPAC. Dié sisteem is dan gebruik om die effektiwiteit van die fiktiewe elektron dinamika metode te bepaal om die berekende minimum energie naby die Born-Oppenheimer energievlak te hou.

Aangeleenthede van belang vir verdere ontwikkeling van hierdie metode is bespreek. Dit sluit in die potensiële kombinasie van hierdie metode met atoomdinamika om chemiese

reaksies op kristal oppervlaktes suksesvol te beskryf asook om gebruik te maak van die bysiendheidsbeginsel van die digtheidsmatriks om lineêre skaling met die aantal atome in die sisteem te bereik.

Summary

Quantum chemical calculations are an invaluable tool in the determination of electronic structure. However, the size of systems studied using these calculations are severely limited due to the highly unfavourable scaling of computational time - typically to the third or fourth order of the number of atoms in the system. Molecular dynamics calculations, on the other hand, can model systems consisting of thousands of atoms. They are, however, insufficient in describing chemical events - quantum chemical calculations are necessary for this.

The quest for $O(N)$ electronic structure calculation methods led to the investigation of a recently proposed fictitious electron dynamics method for calculating electronic structure which uses the idempotency of the density matrix to develop first and second order equations of motion. These equations are implemented in a semi-empirical environment, supplied by the MOPAC software package. The velocity Verlet scheme is used to integrate these equations and the enforcement of constraints is accomplished through McWeeny purification and the RATTLE algorithm. The essential role that parameters play in the effectiveness of the equations of motion is investigated and suggestions are made for these parameters. The requirements of energy conservation of the equations of motion, as well as the stability of the velocity Verlet integrator are addressed and the parameters are revised in order to comply to these requirements.

The importance of the Si(100)2x1:H system as a test system is emphasized and the barriers for hydrogen atom diffusion are calculated using an existing parameter set in MOPAC. This system is used to determine the efficiency of the fictitious electron dynamics method to keep the calculated minimum energy close to the Born-Oppenheimer energy surface.

Issues of relevance for further development of this method are discussed. These include the potential combination of this method with atomic dynamics to successfully describe chemical reactions on crystal surfaces as well as making use of the principle of

nearsightedness of the density matrix to achieve linear scaling.

Keywords: **P**-dynamics, **C**-dynamics, density matrix, configuration matrix, idempotency, orthonormality, velocity Verlet algorithm, McWeeny purification, RATTLE algorithm

Contents

Chapter 1

Introduction

1.1 Background

Newton's *Principia Mathmatica* was the climax of a revolution in man's perception of the Universe, resulting in the acceptance of mathematical physics as a reliable and powerful tool for describing nature. The same laws which accurately predicted the motion of planets around the sun also accounted for the trajectories of terrestrial projectiles, including that legendary windfall apple. So remarkable was the enormous range of scales over which the laws were observed to work, that many believed that they applied universally. Two centuries later, however, a second revolution took place in which classical Newtonian mechanics was found to be inadequate for explaining phenomena on the atomic scale, and a new theory was required. This theory was quantum mechanics.

Despite the philosophical questions of interpretation[1] which arise from the new theory, few question the astounding accuracy with which quantum mechanics describes the world around us. The favourite example cited is that of relativistic quantum field theory's prediction of the gyromagnetic ratio of the electron,[2] which agrees with experiment[3] to better than one part in a million. Today there is little doubt that quantum theory applied to electrons and atomic nuclei provides the foundation for all

low-energy physics, chemistry and the physics underlying biology. If it is wished to describe complex processes occuring in real materials precisely, it should be attempted to solve the equations of quantum mechanics.

Unfortunately, the equations are too complicated to be solved analytically for all but the simplest (and hence most trivial) of systems.[4] The only hope of bringing the power of quantum mechanics to bear on real phenomena of genuine interest to contemporary scientists, and of relevance to our society in general, is to solve the equations numerically by modeling the processes of interest computationally.

Many aspects of computational modeling make it a worthy partner of experimental science.[5] The chemist studying a particular reaction can reach into the computer simulation, alter bond lengths or angles, and then observe the effect of such changes on the process taking place. The geophysicist interested in phase transitions occurring deep inside the earth can model pressures and temperatures which could never be reached in a laboratory. All this can be achieved with a single piece of apparatus - the computer itself.

Quantum-mechanical calculations stand out because, in principle, they can be by design ab initio i.e. from first principles, calculations.[6] They do not depend upon any external parameters except the atomic numbers of the constituent atoms to be modeled and cannot therefore be biased by preconceptions about the final result. Such calculations are reliable and can be used with confidence to predict the behaviour of nature.

Nevertheless, the same complexity which precludes exact analytical solution also results in the highly unfavourable scaling of computational effort and resources required.[7] A alternative to ab initio calculations is to perform semi-empirical calculations. Semi-empirical methods are normally formulated within the same conceptual framework as ab initio methods, but they neglect smaller integrals to speed up the calculations.[5],[8]–[10] In order to compensate for the errors caused by these approximations, empirical parameters are introduced into the remaining integrals and calibrated

against reliable experimental or theoretical reference data. Compared with ab initio or density functional methods, semiempirical calculations are much faster, typically several orders of magnitude,[8] although they are somewhat less accurate.

The last couple of years, however, have seen an upsurge of interest in $O(N)$ electronic structure methods, where the computational time scale linearly with the number of atoms in the system, for treating condensed matter both within tight-binding theory and within density functional theory.[13]-[20] These methods are possible because electronic phase coherence is localized.[13] This localization property can be expressed by saying that the density matrix decays to zero with increasing distance. The result is that the electronic structure of an atom depends only on its local environment, so that if the overall size of the system changes, there is no effect on the local electronic structure; thus the effort required to solve for the whole system should be proportional to the number of atoms, N. This is the foundation of all linear scaling electronic structure methods.

1.2 Motivation for study

A new fictitious electron dynamics method has been proposed by Hoffman.[21] The implementation of this method within the semi-empirical environment of MOPAC shows promise for combination with molecular dynamics in the philosophy of the Car-Parinello method.[50] This can then ultimately be used in the determination of reaction and absorption barriers on, for example, the Si(100)2x1 surface. Furthermore, this new method presents an equation of motion for the density matrix by making use of a theory of idempotency conserving changes. As discussed above, the direct use of the density matrix in the calculation of electronic structure can lead to linear scaling in computational time with the number of atoms in a system. This theory thus shows promise of development towards $O(N)$ scaling. With this in mind, the following aims are intended for this study:

- Comprehend the theory of idempotency conserving changes.

- Implement the fictitious electron dynamics method in the semi-empirical environment of the MOPAC software package[10] - this includes the dynamics for both the density and configuration matrix. The enforcement of constraints on the density and configuration matrix should also be accounted for.

- Create a software environment convenient for further development and evaluation of the method implemented in this study.

- Demonstrate, using simple model systems, that the idempotency conserving equations of motion can indeed be used, with the appropriate molecular dynamics algorithm, to observe the ground state electronic configuration by benchmarking the results against a well established method in MOPAC.

- Identify an appropriate model system for implementation of this method within a semi-empirical environment.

- Understanding the Si(100)2x1:H system with the aim of modeling the diffusion of hydrogen on the surface in future using the fictitious electron dynamics method combined with atomic dynamics. The Si(100)2x1:H system is chosen since numerous studies[22]–[28] have been done using similar methods; thus it serves as an ideal test system.

- Identify issues of relevance for further development of the method in order to become competitive with other methods and possibly achieve a linear scaling method.

- In situations of changing atomic structure, test the efficiency of this method to keep the calculated minimum electronic energy close enough to the Born-Oppenheimer energy surface.

- Compare $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics.

1.3 Layout of thesis

Chapter 2 outlines the theory underlying some of the methods currently used to calculate electronic structure.

Chapter 3 briefly summarizes the mathematical scheme behind a recently proposed fictitious dynamics method,[21] together with the derivation of the equations of motion for this particular method.

Chapter 4 focuses on the implementation of the equations of motion within a semi-empirical framework, as well as the factors that need to be considered during implementation.

Chapter 5 contains an introduction to the Si(100)2x1:H system, including the determination of the barriers of hydrogen atom diffusion and how this system is used to determine the efficiency of the fictitious dynamics method to keep the calculated minimum energy close to the Born-Oppenheimer energy.

Chapter 6 identifies relevant issues for future development of the fictitious dynamics method such as the combination with atomic dynamics and the possible achievement of $O(N)$ scaling.

Chapter 7 summarizes the conclusions that arised from this study.

Chapter 2

Calculation of electronic structure

Using computer programs to derive and analyze chemical properties has become an important tool of study in theoretical chemistry. Solving the Schrödinger equation which describes the relationship between molecular structure and energy is the basis of theoretical chemistry. Hartree-Fock theory, as an immediate task, provides a successful and thoroughly tested framework for molecular calculation.[4],[9],[11] The rigorous ab initio methods (these include Hartree-Fock theory, configuration interaction, density functional theory and perturbation theory) calculate the molecular system using extensive basis sets and provide accurate results. Including correlation, ab initio methods can provide results comparable with experiment. Still, the computer memory capacity and power are limited resources. With the high complexity of calculation, the maximum molecule size modern computers can handle is limited to about 100 basis functions.[29],[30] Molecular mechanics methods replace the complicated calculation with empirical potentials to simplify the calculation.[5],[6] These methods sacrifice the generality and accuracy for speed and can easily handle molecular structures consisting of thousands of molecules on modern computers. Semi-empirical methods[8]–[10] are intermediate to ab initio and molecular mechanics methods. Like molecular mechanics, they use experimentally derived parameters to strive for accuracy; like ab initio methods they are quantum mechanical in nature. Semi-empirical methods are relatively

fast, because many of the difficult integrals, as discussed in section 2.5, are neglected. They are typically several orders of magnitude faster than ab initio methods.[8] The errors introduced by the neglect of integrals are compensated for by the use of parameters. Thus, semi-empirical calculations sometimes produce greater accuracy than ab initio calculations at the same level.

In this chapter the many-electron problem will be put forth followed by a discussion of some of the approximations used to solve this problem. These include the Hartree-Fock method and Roothaan approximation, the latter providing the foundation for many semi-empirical and ab initio methods. Special attention is given to the semi-empirical methods and some background regarding how it is incorporated into the MOPAC software package used in this study, is included. An overview of molecular dynamics are also given and to conclude the chapter ab initio molecular dynamics are briefly discussed.

2.1 The many-electron problem

The challenge of modern electronic structure theory is to go beyond the independent electron approximation and to find a solution to the many-body Schrödinger equation for a fully interacting system of electrons bound by positively charged nuclei. Within the Born-Oppenheimer approximation (Appendix A), the time-independent Hamiltonian operator, $\hat{H}$, of a generic system acting upon the many-body electronic wavefunction, Ψ , is given by [46]

$$\begin{aligned} \hat{H}\Psi\left(\{\mathbf{r}_i\}\right) &= \sum_{i=1}\left[-\frac{1}{2}{\nabla_i}^2 - \sum_{\alpha}\frac{Z_\alpha}{\mathbf{r}_i-\mathbf{R}_\alpha} + \frac{1}{2}\sum_{i=1}\sum_{j\neq i}\frac{1}{|\mathbf{r}_i-\mathbf{r}_j|}\right]\Psi\left(\{\mathbf{r}_i\}\right) \\ &= E\Psi\left(\{\mathbf{r}_i\}\right) , \end{aligned} \tag{2.1}$$

where $\mathbf{r}_i$ and $\mathbf{R}_\alpha$ are the coordinates of the i^{th} electron and the α^{th} nucleus respectively, the latter carrying charge Z_α. The state Ψ has energy E, and both of these quantities have an implicit dependence on the nuclear coordinates, $\mathbf{R}_\alpha$, which has been omitted

for clarity of notation. The first term in the square brackets in equation (2.1) is the electronic energy kinetic operator, the second term is the electron-nucleus Coulomb interaction, and the remaining term is the electron-electron Coulomb interaction. In equation (2.1) and the rest of this section, atomic units are used throughout: $\hbar = m_e = \epsilon = 4\pi\epsilon_0 = 1$.

Now, the primary concern is often with the ground state of the system, and the variational principle of quantum mechanics[46] states that the expectation value of the Hamiltonian operator,

$$\left\langle \hat{H} \right\rangle_{\Psi} = \langle \Psi | \, \hat{H} \, | \Psi \rangle \ , \tag{2.2}$$

is minimised by the wavefunction Ψ that corresponds to the groundstate wavefunction. Thus, by minimising the functional of equation (2.2) with respect to all many-electron wavefuctions that are antisymmetric under particle exchange, the ground state energy will be obtained. This, however, cannot be done exactly except in the very simplest of cases. Further progress can be made by instead of minimising equation (2.2) with respect to all wavefunctions that satisfy the antisymmetry requirement, to just perform the minimisation over antisymmetric wavefunctions that may be written as sums of products of one-electron wavefunctions $\Psi_i(r)$ that are constrained to be normalised. The Euler-Lagrange equations will yield one-body differential equations of the form,

$$\left(-\frac{1}{2}\nabla^2 + U(\mathbf{r}) \right) \psi_i(\mathbf{r}) = \epsilon_i \psi_i(\mathbf{r}) \ , \tag{2.3}$$

for some Lagrange multipliers ϵ_i and mean field potential $U(\mathbf{r})$[46]. Thus, by performing the search for the minimum of the expectation value of equation (2.2) to a smaller subspace of the Hilbert space of anti-symmetric many-body wavefunctions, namely to those that can be expressed in terms of one-electron wavefunctions, the difficult many-body problem can be reduced to a finite set of simpler one-body problems. What is lost out on though is that the true ground state wavefunction will most likely not be in the subspace the search is performed in, and as a result the ground state energy that is found will be an upper bound to the true ground state energy of the system.

2.2 Hartree-Fock Approximation

An example of the approach above, which is known as the Hartree-Fock approximation, is briefly presented because it lies at the root of the methods employed in this study. First, the Hamiltonian of equation (2.1) is written as

$$\begin{aligned} \hat{H} &= \sum_{i=1}\left[-\frac{1}{2}\nabla^2{}_i - \sum_{\alpha}\frac{Z_\alpha}{|\mathbf{r}_i - \mathbf{R}_\alpha|}\right] + \frac{1}{2}\sum_{i=1}\sum_{j\neq i}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} \\ &= \hat{H}_0 + \hat{V}_{ee}\ , \end{aligned} \tag{2.4}$$

where $\hat{H}_0$ represents the term in square brackets and $\hat{V}_{ee}$ is the electron-electron interaction term. Furthermore, the assumption is made that the N-body electronic wavefunction, Ψ_N may be written as a Slater determinant of orthonormal one-electron orbitals,

$$\Psi_N(\mathbf{r}_1 s_1, ..., \mathbf{r}_N s_N) = \frac{1}{\sqrt{N!}}\begin{vmatrix} \psi_1(\mathbf{r}_1 s_1) & \psi_1(\mathbf{r}_2 s_2) & ... & \psi_1(\mathbf{r}_N s_N) \\ \psi_2(\mathbf{r}_1 s_1) & \psi_2(\mathbf{r}_2 s_2) & ... & \psi_2(\mathbf{r}_N s_N) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ \psi_N(\mathbf{r}_1 s_1) & \psi_N(\mathbf{r}_2 s_2) & ... & \psi_N(\mathbf{r}_N s_N) \end{vmatrix} \tag{2.5}$$

where s_i is the spin co-ordinate of the i^{th} electron. This Slater determinant of one-electron wavefunctions automatically accounts for the antisymmetry under particle exchange. The expectation value of the Hamiltonian is given by $\left\langle \Psi_N | \hat{H}_0 + \hat{V}_{ee} | \Psi_N \right\rangle$. First consider the term involving $\hat{H}_0$. By the orthogonality of the ψ_i, each of the $N!$ terms in $\langle \Psi_N |$ only survives when $\hat{H}_0$ operates on the corresponding term in $|\Psi_N\rangle$, thus

$$\langle \Psi_N | \hat{H}_0 | \Psi_N \rangle = N! \frac{1}{N!} \sum_{i=1} \langle \psi_i(\mathbf{r}_i) | -\frac{1}{2}\nabla_i{}^2 - \sum_{\alpha}\frac{Z_\alpha}{\mathbf{r}_i - \mathbf{R}_\alpha} | \psi_i(\mathbf{r}_i) \rangle\ , \tag{2.6}$$

where the spin degree of freedom has been integrated out and use has been made of the fact that $N!$ equal terms arises. The term involving $\hat{V}_{ee}$ requires a little more thought. Each term in $\hat{V}_{ee}$ acts upon a pair of orbitals $\{\psi_i, \psi_j\}$, where $i \neq j$. Thus for each term in $\langle \Psi_N |$, there are two non-zero terms in the expectation value of $\hat{V}_{ee}$ and they

have opposite signs due to the antisymmetry of the Slater determinant. All other terms vanish by orthogonality. Thus it follows that

$$\langle \Psi_N | \hat{V}_{ee} | \Psi_N \rangle = \frac{1}{2} \sum_{i=1}^{N} \sum_{j \neq i} \int \int d\mathbf{r} d\mathbf{r}' \left[\frac{|\psi_i(\mathbf{r})|^2 |\psi_j(\mathbf{r}')|^2}{|\mathbf{r} - \mathbf{r}'|} - \delta_{s_i s_j} \psi_i^*(\mathbf{r}) \psi_j^*(\mathbf{r}') \frac{\psi_j(\mathbf{r}) \psi_i(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \right] \quad (2.7)$$

where the Krönecker delta arises from integrating over the spin co-ordinates. Now, minimizing the sum of equations (2.6) and (2.7) with respect to $\psi_k^*(\mathbf{r})$ subject to the constraint that the one-electron wavefunctions remain normalized, results in

$$\epsilon_k \psi_k(\mathbf{r}) = \left[-\frac{1}{2} \nabla_i^2 - \sum_\alpha \frac{Z_\alpha}{\mathbf{r} - \mathbf{R}_\alpha} \right] \psi_k(\mathbf{r}) + \sum_j \int d\mathbf{r}' \frac{|\psi_j(\mathbf{r}')|^2}{|\mathbf{r} - \mathbf{r}'|} \psi_k(\mathbf{r}) - \delta_{s_i s_j} \sum_j \int d\mathbf{r}' \frac{\psi_j^*(\mathbf{r}) \psi_k(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \psi_j(\mathbf{r}) \,, \quad (2.8)$$

where the $\{\epsilon_k\}$ are Lagrange multipliers used to enforce the normalization constraint upon the one-electron orbitals. Expressions (2.8) are known as the Hartree-Fock equations. From equations (2.6), (2.7) and (2.8) it can be seen that the expectation value of the Hamiltonian, known as the Hartree-Fock energy, is given by

$$\left\langle \hat{H} \right\rangle_{\Psi_N} = \sum_i^N \epsilon_i - \frac{1}{2} \sum_{i,j}^N \left[\langle ij | \frac{1}{r_{ij}} | ij \rangle - \langle ij | \frac{1}{r_{ij}} | ji \rangle \delta_{s_i s_j} \right] \,, \quad (2.9)$$

where $r_{ij} = |r_i - r_j|$, and bra-ket notation has been used for clarity. The first term in square brackets is the *Hartree term.* This is just minus the classical electrostatic energy of a charge distribution $\sum_i |\psi_i(\mathbf{r})|^2$, and it includes the self-interaction energy (i.e the $i = j$ terms). The second term in square brackets is what is known as the *exchange term.* It is purely quantum mechanical in nature, arising from the antisymmetry restriction imposed upon the wavefunction. It acts only upon pairs of electrons with parallel spins, and does so in such a way as to keep them apart, as would be expected from the Pauli exclusion principle. An important feature of the Hartree-Fock theory is that it is free of self-interaction: the $i = j$ contributions in the Hartree and exchange terms in equation (2.9) identically cancel. As mentioned earlier, the Hartree-Fock energy is an upper bound on the true ground state energy of the Hamiltonian, and the difference

(by definition negative) is known as the *correlation energy.* It is mentioned in passing that by taking linear combinations of Slater determinants of single particle orbitals as the trial N-body wavefunction the magnitude of this energy difference can be reduced. Such schemes are known as *configuration interaction* methods and they can in principle provide exact answers, but at massive computational cost.[39],[40]

2.3 Roothaan Approximation

The differential equations for finding the Hartree-Fock orbitals from equation (2.8) can be simply expressed as[45]

$$\hat{F}(1)\psi_i(\mathbf{r}_1) = \epsilon_i\psi_i(\mathbf{r}_1), \qquad i = 1, 2, ...N \ , \tag{2.10}$$

where ψ_i is the i^{th} spin-orbital, the operator $\hat{F}$, called the Fock (or Hartree-Fock) operator, is the effective Hartree-Fock Hamiltonian, and the eigenvalue ϵ_i is the energy of the spin-orbital i. However, the Hartree-Fock operator $\hat{F}$ has extra terms as compared to the effective Hartree Hamiltonian, as set out in equation (2.14).

Equation (2.10) is a one-electron differential equation in which $\hat{F}$ depends on its own eigenfunctions, which are not known initially. Thus Hartree-Fock equations have to be solved self-consistently.

For the closed shell case, the sole criterion required to obtain a minimum expectation value for the total energy, the Hartree-Fock equation, may be expressed entirely as a functional of the spatial orbitals, resulting in the integro-differential equation (2.10). Like the Schrödinger equation, this is an eigenvalue equation that must be solved for the orbital energy, ϵ_i, and the wave function, ψ_i. The subscript on the wave function serves to indicate that it is only one of several spatial orbitals required to construct a complete Slater determinant and accurately model a closed-shell molecule or atomic cluster. In a closed-shell molecule every spatial orbital is occupied by a pair of electrons with opposite spins; therefore, to form a complete Slater determinant description a Hartree-

Fock equation must be solved for each doubly occupied spatial orbital. The contribution of Roothaan[47] to the Hartree-Fock method was to determine how to simultaneously solve these equations by writing the spatial orbitals as a linear combination of a set of basis functions. Roothaan's method is elegant in its simplicity because the differential equation is reduced to a set of algebraic equations that may be solved using matrix methods which can, in turn, be efficiently implemented on modern computers. The first step in the reduction is to expand the unknown spatial orbital ψ_i in terms of a generic basis set, $\{\phi_\mu\}$:

$$\psi_i = \sum_\mu c_{\mu i}\phi_\mu \,, \tag{2.11}$$

with $\{c_{\mu i}\}$ the expansion coefficients and substitution of this expansion into equation (2.10) leads to

$$\sum_\mu c_{\mu i}\hat{F}\phi_\mu = \epsilon_i \sum_\mu c_{\mu i}\phi_\mu \,. \tag{2.12}$$

Multiplication by ϕ_ν^* and integration of the above equation gives

$$\sum_\nu c_{\nu i}\left(F_{\mu\nu} - \epsilon_i S_{\mu\nu}\right) = 0 \,, \tag{2.13}$$

where

$$\begin{aligned} F_{\mu\nu} &= \langle\phi_\mu|\,\hat{F}\,|\phi_\nu\rangle \\ &= H_{\mu\nu}^{core} + \sum_{\lambda=1}^{N}\sum_{\sigma=1}^{N} P_{\lambda\sigma}\left[\langle\mu\nu \mid \lambda\sigma\rangle - \frac{1}{2}\langle\mu\lambda \mid \nu\sigma\rangle\right] . \end{aligned} \tag{2.14}$$

$\mathbf{F}$ is the Fock matrix in the representation of the generic basis set. The quantity $P_{\lambda\sigma} = 2\sum_i^{occ} c_{\lambda i}c_{\sigma i}$ is known as either the density matrix or charge-density bond-order matrix because it recurs in the determination of these properties. Furthermore $\mathbf{S}$ is defined as the overlap matrix, which has elements

$$S_{\mu\nu} = \langle\phi_\nu \mid \phi_\mu\rangle \,. \tag{2.15}$$

and represents the overlap between ϕ_ν and ϕ_μ as a positive or negative fraction between zero and one. The closer the overlap is to one, the closer the two basis functions

approach linear dependence. The overall sign depends on the spatial orientation and signs of the individual basis functions. In the case where the basis functions ϕ_ν and ϕ_μ are atomic orbitals of the constituent atoms, the so-called LCAO[9] (linear combinations of atomic orbitals) is obtained.

The expression of the core Hamiltonian, $H_{\mu\nu}^{core}$, in equation (2.14) is given by

$$H_{\mu\nu}^{core} = \langle\mu| -\frac{1}{2}\nabla^2 |\nu\rangle + \sum_B Z_B \langle\mu| -\frac{1}{R_B} |\nu\rangle . \tag{2.16}$$

The electronic energy can also be obtained from the following expression

$$E_{el} = \frac{1}{2}\sum_{\mu\nu} P_{\mu\nu}\left(H_{\mu\nu}^{core} + F_{\mu\nu}\right) . \tag{2.17}$$

The equations (2.13) are a set of simultaneous linear homogeneous equations in the unknowns $\{c_{\mu i}\}$. For a nontrivial solution

$$det(\mathbf{F} - \mathbf{S\Lambda}) = \mathbf{0} , \tag{2.18}$$

with $\mathbf{\Lambda}$ given by

$$\mathbf{\Lambda} = \begin{pmatrix} \varepsilon_1 & & 0 \\ & \ddots & \\ & & \\ 0 & & \varepsilon_N \end{pmatrix} .$$

This is a secular equation whose roots give the orbital energies. These equations are called the *Roothaan equations* and must be solved self-consistently, since the $F_{\mu\nu}$ integrals depend on the orbitals ϕ_μ, which in turn depend on the unknown coefficients $c_{\mu i}$. In a typical self-consistent field calculation an initial guess is made of the electron distribution. This is then used to calculate the average field of the rest of the electrons. The coefficients are then calculated to produce a set of one-electron molecular orbitals and energies. The molecular orbitals are filled according to the Aufbau principle and the total electronic energy is thus obtained. This set of molecular orbitals are then

used as the next approximation to the electron distribution. The process is repeated, gradually improving the molecular orbitals, until the electronic energy converges, i.e. it changes by less than some pre-set limit.

The secular equation can also be written in matrix form. Let $\mathbf{C}$ be a square matrix whose elements are the expansion coefficients $\{c_{\mu i}\}$. Let $\mathbf{F}$ be a square matrix whose elements are $\{F_{\nu\mu}\}$ as given by equation (2.14). Let $\mathbf{S}$ be a square matrix with elements $\langle\phi_\nu \mid \phi_\mu\rangle$ as given by equation (2.15) and ϵ be the diagonal square matrix whose diagonal elements have the orbital energies $\epsilon_1, \epsilon_2, \ldots$.

Equation (2.13) can thus be written as

$$\mathbf{FC} = \mathbf{SC}\epsilon \,. \tag{2.19}$$

This matrix equation, with $\mathbf{F}$ as a symmetric matrix, is the matrix equivalent of the secular equation and is convenient for implementation on modern computers to calculate electronic structure.

2.4 Ab Initio Methods

The term ab initio means from first principles. This does not imply the exact solution of the Schrödinger equation, but instead suggests the selection of a method that can, in principle, lead to a reasonable approximation to the solution of the Schrödinger equation, as well as a basis set that will implement that method in a reasonable way. Ab initio methods include Hartree-Fock and density functional theory.[46],[49]

An ab initio method would calculate the self-consistent field (SCF) rigorously and thus solve equation (2.19) explicitly. In other words, no experimental data are used in solving equation (2.19). However, the full inclusion of a large set of basis functions and rigorous calculations introduce huge numbers of calculations which require $O(N^4)$[7] computational time, with N the number of basis functions in the system.

Ab initio methods are best used when small systems (tens of atoms) need to be modelled and where electronic transitions are involved. It is also useful for systems or molecules where no or limited experimental data are available.

2.5 Semi-empirical Methods

Instead of introducing a large set of atomic orbitals and doing everything rigorously, semi-empirical methods simplify the SCF calculations by using some carefully selected pieces of experimental information. Semi-empirical methods refer to computational techniques known as the modified neglect of diatomic overlap (MNDO) approach and related methods. The majority of these methods are incorporated into the popular code "Molecular Orbital PACkage" (MOPAC).[10]

The reason why semi-empirical methods have been developed was to find a compromise between the computational efficiency and physical correctness of the approximations used. The usefulness of these methods comes from the balance between several characteristics, namely theoretical rigor and pragmatism, speed and accuracy, specific parameterization and generality. These methoda capture the essential aspects of a quantum mechanical approach including electronic levels, charge transfer, and spin polarization; it allows making and breaking of chemical bonds; it provides geometric structures, binding energies, infrared spectra, electronic charges, electrostatic potentials, dipole moments, etc. The heavy computational effort of ab initio calculations is avoided by a number of algorithmic simplifications such as minimal basis sets and the elimination of difficult integrals, which are either made small by proper mathematical transformations and then ignored or by using them as parameters to fit experimental data such as geometries, heats of formation, and ionization potentials.[8] The latter aspect limits the generality and transferability of the methods. There are various semiempirical methods. Six distinct methods are available within MOPAC namely MINDO/3, MNDO, AM1, PM3, PM5 and MNDO-d.

All these methods have roughly the same structure. They all are self-consistent field methods, they take into account electrostatic repulsion and exchange interaction. However, instead of rigorous calculations of the Fock matrix elements, formulas with predetermined coefficients are used. In order to provide a superficial understanding of these methods and their relationship to ab initio methods, it will be briefly dwelled on the approximations made.

Approximations used in semi-empirical methods [8],[9]

- Explicitly consider only the valence electrons while the core electrons are subsumed into the nuclear core.

- Use of a minimal basis set (a basis set that contains functions only pertaining those atomic orbitals that are occupied (s,p,d), for example, for carbon the STO-3G minimal basis set contains only 1s, 2s, 2p functions).

- Assume the atomic orbital basis has already been orthogonalized, i.e. set the overlap matrix, $S_{\mu\nu}$, equal to the identity matrix. Thus equation (2.19) can be rewritten as

$$\mathbf{FC} = \mathbf{C}\epsilon. \tag{2.20}$$

- Omission of some two electron integrals, where these integrals are given by

$$\begin{aligned} V_{\mu\nu\lambda\sigma} &= \langle \mu\nu \mid \lambda\sigma \rangle \\ &= \int \phi_\mu^*(\mathbf{r}_1)\phi_\nu^*(\mathbf{r}_2)\frac{1}{\mathbf{r}_{12}}\phi_\lambda(\mathbf{r}_1)\phi_\sigma(\mathbf{r}_2)\mathrm{d}\mathbf{r}_1\mathrm{d}\mathbf{r}_2 \, . \end{aligned}$$

  In the MINDO/3 method, for example, the three- and four-center two electron integrals are neglected.

- Practically all matrix elements are approximated by analytical functions of interatomic separation and atom environment. Parameters are chosen to reproduce the characteristics of reference systems.

- Core-core Coulombic repulsion is replaced with a parameterized function.

Semi-empirical methods can be used to model systems where electronic transitions are involved as is the case for ab initio methods. The advantage of semi-empirical methods is that the system-size under study can be in the range of hundreds of atoms in comparison to the tens of atoms for ab initio methods.

2.6 MOPAC

MOPAC is a general-purpose semi-empirical molecular orbital package for the study of solid state and molecular structures and reactions. The semiempirical Hamiltonians MNDO, MINDO/3, AM1, PM3 , PM5, and MNDO-d as discussed above are used in the electronic part of the calculation to obtain molecular orbitals, the heat of formation and its derivative with respect to molecular geometry. Using these results MOPAC calculates the vibrational spectra, thermodynamic quantities, isotopic substitution effects and force constants for molecules, radicals, ions, and polymers.[10] For studying chemical reactions, a transition state location routine[10] and two transition state optimizing routines[10] are available. MOPAC also makes provision for geometry optimization. For users to get the most out of the program, they must understand how the program works, how to enter data, how to interpret the results, and what to do when things go wrong.

The simplest description of how MOPAC works is that the user creates a data-file which describes a molecular system and specifies what kind of calculations and output are desired. The user then commands MOPAC to carry out the calculation using that data-file. Finally the user extracts the desired output on the system from the output files created by MOPAC. This method is used in the calculation the diffusion barriers for hydrogen on the Si(100)2x1 surface, which is discussed in chapter 5.

The original version of the MOPAC package was not designed for the study of solid

state materials. However, later versions of MOPAC incorporate this capability. Since the solid-state method used in MOPAC is unusual, some details are thus given.

Should a particular polymer unit cell be considered which is large enough, a single point in k-space, the Gamma point, is sufficient to specify the entire Brillouin zone. The secular determinant for this point can be constructed by adding together the Fock matrix for the central unit cell plus those for the adjacent unit cells. The Born-von-Karman cyclic boundary conditions are thus satisfied, and diagonalization yields the correct density matrix for the Gamma point.

At this point in the calculation, conventionally, the density matrix for each unit cell is constructed. Instead, the Gamma-point density and one-electron density matrices are combined with a "Gamma-point-like" Coulomb and exchange integral strings to produce a new Fock matrix. The calculation can be visualized as being done entirely in reciprocal space, at the Gamma point.

Most solid-state calculations take a very long time. These calculations, called "cluster" calculations, require between 1.3 and 2 times the equivalent molecular calculation.

A minor "fudge" is necessary to make this method work. The contribution to the Fock matrix element arising from the exchange integral between an atomic orbital and its equivalent in the adjacent unit cells is ignored. This is necessitated by the fact that the density matrix element involved is invariably large.

The unit cell must be large enough that an atomic orbital in the center of the unit cell has an insignificant overlap with the atomic orbitals at the ends of the unit cell. In practice, a translation vector of more that about 7 or 8 Angstroms is sufficient. For one rare group of compounds a larger translation vector is needed. Polymers with delocalized pi-systems, and polymers with very small band-gaps will require a larger translation vector, in order to accurately sample k-space. For these systems, a translation vector in the order of 15-20 Angstroms is needed.

2.7 Molecular Dynamics

In principle, equation (A.6) (Appendix A) could be solved for $E_{el}(\mathbf{R})$, and then equation (A.8) could be solved for the nuclear wavefunctions from which the atomic dynamics can be obtained. However, solution of equation (A.6) requires a large amount of computation (using ab initio quantum chemistry codes such as Gaussian or semiempirical codes such as MNDO in MOPAC as discussed in section 2.5). Thus, usually an empirical fit to $E_{el}(\mathbf{R})$ (a forcefield) is used.

Solution of equation (A.8) is called quantum dynamics and also requires a large amount of computation. Since the nuclei are relatively heavy, the quantum mechanical effects are often insignificant, so equation (A.8) can be replaced with the Newton's equations of motion

$$\mathbf{F}_i = m\mathbf{a}_i \, . \tag{2.21}$$

Equation (2.21) is one statement of the classical equations of motion. A second and completely equivalent statement can be obtained from a function called the Lagrangian. In this case, the Lagrangian, which is the difference between the kinetic and the potential energy, can be written as

$$\mathcal{L}(\mathbf{r}_i, \dot{\mathbf{r}}_i) = K(\dot{\mathbf{r}}_i) - V(\mathbf{r}_i), \tag{2.22}$$

where $\mathcal{L}$ is the Lagrangian, K is the kinetic energy, V is the potential energy and $\mathbf{r}_i = (x_i, y_i, z_i)$ represents the coordinates of the i^{th} particle. The equation of motion is then obtained by evaluating[12]

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{r}}_i}\right) - \left(\frac{\partial \mathcal{L}}{\partial \mathbf{r}_i}\right) = 0. \tag{2.23}$$

Solving equation (2.23) for the nuclei moving classically on a single potential surface is called classical molecular dynamics. If the time evolution of the system is of no interest, then $V(\mathbf{r})$ can be used to calculate static properties such as equilibrium structures, transition states, relative energies, etc. This is called molecular mechanics. Typical assumptions of classical molecular dynamics are:

- The Born-Oppenheimer approximation[9] is invoked, in other words the nuclei move in the average field of the electrons.

- The nuclei also move on a single potential surface (i.e. a single electronic state).

- This potential surface can be approximated by an empirical fit.

- The nuclear motion can be described by classical mechanics.

A question that can arise is how Newton's law can be used to move atoms, when it is known that systems at the atomistic level obey quantum laws rather than classical laws, and that Schrödinger's equation is to be followed.

A simple test of the validity of the classical approximation is based on the de Broglie thermal wavelength[62] defined as

$$\Lambda = \sqrt{\frac{h^2}{2\pi m k_B T}}\,, \tag{2.24}$$

where m is the atomic mass and T the temperature. The classical approximation is justified if $\Lambda \ll a$, where a is the mean nearest neighbour separation. If one considers for instance liquids at the triple point, $\frac{\Lambda}{a}$ is of the order of 0.1 for light elements such as Li and Ar, decreasing further for the heavier elements. The classical approximation is thus poor for very light systems such as H_2, He and Ne.

Moreover, quantum effects become important in any system when T is sufficiently low. The drop in the specific heat of crystals below the Debye temperature[31], or the anomalous behaviour of the thermal expansion coefficient, are well known examples of measurable quantum effects in solids.

Molecular dynamics results should thus be interpreted with caution in these regions.

2.8 Ab initio molecular dynamics

An accurate force field is an important element in the molecular dynamics method, as it permits large systems to be studied at relatively little computational cost. However, current force field technology is not capable of describing chemical events involving bond breaking and forming.[32] Another deficiency of current force fields is their failure to include polarization effects. The technique known as ab initio molecular dynamics (AIMD)[33],[34] solves these problems by combining "on the fly" electronic structure calculations with finite temperature dynamics. Not surprisingly, AIMD simulations are substantially more expensive than calculations based on empirical force fields. However, recent advances in electronic structure theory as well as readily available high-speed computers have begun to render the AIMD approach a viable one for studying chemical processes in the condensed phase.

The most important element in an AIMD calculation is the representation of the electronic structure. Clearly, calculation of the exact ground-state electronic wavefunction is intractable, and approximations must be used. The electronic structure theory employed should be reasonably accurate yet not too computationally demanding. One formulation of the electronic structure problem that satisfies these criteria is density functional theory (DFT).[49] DFT formulates the many-electron problem in terms of the electron density, $n(\mathbf{r})$, rather than the many-body wavefunction. Thus, in principle, the central quantity is a function of just three rather than of $3N$ variables, with N the number of electrons in the system, a fact that renders calculations based on DFT computationally tractable. The basic tenet of DFT is that the energy of the quantum many-body system can be expressed as a unique functional of the electron density. By minimizing the density functional over all electron densities that give rise to a particular number of electrons, the ground state density and energy for a given system are obtained. Unfortunately, the explicit and unique form of this functional is not known. However, in the orbital-based formulation of DFT by Kohn and Sham,[49] reasonable approximations to the density functional have been developed. In the Kohn-Sham for-

mulation, the energy is expressed in terms of a set of n occupied single-particle orbitals, $\{\psi_1(\mathbf{r}), ..., \psi_n(\mathbf{r})\}$, and the N nuclear positions, $\{\mathbf{R}_1, ..., \mathbf{R}_N\}$, and takes the form

$$E_i\left[\{\psi_i\}\{\mathbf{R}_i\}\right] = -\frac{1}{2}\sum_{i=1}^{n}\int d\mathbf{r}\psi_i^*(\mathbf{r})\nabla^2\psi_i(\mathbf{r}) + \frac{1}{2}\int d\mathbf{r}d\mathbf{r}'\frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} + E_{xc}\left[n\right] + \int d\mathbf{r}V_{ext}\left(\mathbf{r}, \mathbf{R}_1, ..., \mathbf{R}_N\right)n(\mathbf{r}) \quad (2.25)$$

where the density is related to the orbitals by

$$n(\mathbf{r}) = \sum_{i=1}^{n}|\psi_i(\mathbf{r})|^2 \quad (2.26)$$

and the orbitals are required to be mutually orthonormal,

$$\langle\psi_i\,|\psi_j\rangle = \delta_{ij}\ . \quad (2.27)$$

In equation (2.25), V_{ext} represents the external potential due to the N nuclei and is given exactly by $V_{ext} = -\sum_{i=1}^{N}\frac{q_i}{|\mathbf{r}-\mathbf{R}_i|}$, where q_i is the charge on each nucleus. The first term in equation (2.25) is the kinetic energy of non-interacting electrons having a density of $n(\mathbf{r})$. All the quantum mechanical electron-electron interactions appear in the functional $E_{xc}\left[n\right]$ which contains the interacting part of the kinetic energy included in what is termed the exchange and correlation energy. Physically interpreted, $E_{xc}\left[n\right]$ lowers the total energy by including the fact that electrons of like spin will avoid one another because of the Pauli principle (exchange) and that all electrons will keep apart because of Coulomb repulsions (correlated motion). In this way, $E_{xc}\left[n\right]$, makes quantum terms to the second term in equation (2.25), which is the purely classical electrostatic energy of the charged particles in the system (electrons and nuclei); because this term contains the self-energy of the electron density, averaged over its motions (Hartree energy), it does not include the lowering due to correlation. This exchange is a purely quantum mechanical phenomena and has no analogue in classical theory.

The functional $E_{xc}\left[n\right]$, called the exchange-correlation functional, is unknown and must be approximated. For certain classes of systems, it has proved sufficient to approximate this functional by taking the exchange and correlation energies of a homogeneous electron gas and substituting for the constant density, n, the inhomogeneous density $n(\mathbf{r})$.

This approximation, known as the local density approximation (LDA), has proved useful and reasonably accurate in many problems of interest in metallic and semiconductor solids. However, it fails, for example, in hydrogen-bonded systems, where spatial variations in the electron density are too rapidly varying to be described adequately by LDA. The most common approach in such cases is to extend the dependence of this functional to include the density $n(\mathbf{r})$ and its gradient $\nabla n(\mathbf{r})$ given by $E_{xc} = E_{xc}[n, \nabla n]$. This approximation is known as the generalized gradient approximation (GGA).[36],[37]

A possible strategy for combining electronic structure with molecular dynamics is the following: for a given set of initial nuclear positions, $\{\mathbf{R}_1, ..., \mathbf{R}_N\}$, minimize the energy functional in equation (2.25) to obtain the ground state density $n^{(0)}(\mathbf{r})$ and corresponding orbitals $\{\psi_1^{(0)}(\mathbf{r}), ..., \psi_n^{(0)}(\mathbf{r})\}$. Given these quantities, the forces between the nuclei are given by the Hellman-Feynman theorem:[48]

$$\mathbf{F}_i = -\frac{\partial}{\partial \mathbf{R}_i} E\left[\left\{\psi^{(0)}\right\}\{\mathbf{R}\}\right]. \tag{2.28}$$

The forces are then fed into a numerical integration procedure together with a set of initial velocities for the nuclei, and a step of molecular dynamics is carried out, yielding a new set of positions and velocities. At the new nuclear positions, the energy functional is minimized again and a new set of forces is obtained and used to perform another step of MD propagation. This procedure is repeated until an entire trajectory has been generated. It is evident that this method can indeed be very time-consuming. An elegant alternative formulation of this procedure was proposed by Car and Parrinello,[50] in which, rather than minimizing the functional at each new nuclear configuration, a fictitious dynamics for the electronic orbitals is introduced, which allows them to follow the motion of the nuclei adiabatically. This dynamical procedure is constructed in such a way that if the orbitals are initially chosen corresponding to the ground state density at the initial nuclear configuration, they will remain approximately in the ground state as the nuclear configuration evolves in time. In the original formulation of the Car-

Parrinello scheme, the orbitals are expanded in a plane wave basis,

$$\psi_i(\mathbf{r}) = \sum_g c_g^i e^{i\mathbf{g}\cdot\mathbf{r}} \tag{2.29}$$

where c_g^i are the expansion coefficients. The fictitious adiabatic dynamics is then formulated for the coefficients by introducing a set of velocities $v_{cg}^i = \dot{c}_g^i$ and an associated mass parameter μ. In a Newtonian scheme, the equations of motion for the particles and coefficients then take the form

$$\mu\ddot{c}_g^i = -\frac{\partial E}{\partial c_g^{i*}} + \sum_j \Lambda_{ij} c_g^j \tag{2.30}$$

$$m_i\ddot{\mathbf{R}}_i = -\frac{\partial E}{\partial \mathbf{R}_i} \tag{2.31}$$

where Λ_{ij} is a set of Lagrange multipliers for enforcing the orthonormality constraint on the orbitals.

An important, developing area of MD methodology is the combination of the ab initio approach with empirical force fields.[42] Such a combined scheme is expected to be of considerable utility in the treatment of large systems, in which chemical processes occur in a relatively localized region, e.g., at the active site of an enzyme or a chemical reaction in solution. Chemical reactions on crystal surfaces, including adsorption and desorption, also provide an ideal environment for application of such methods. In such systems, ab initio MD can be used to treat the chemically active region and a force field employed to describe the rest of the system. One of the difficult problems associated with a combined scheme is specifying how the electrons and nuclei in the ab initio region interact with the atoms in the force-field region. This is an especially challenging problem when it is necessary to "cut" bonds within a molecule, for example, in treating a reaction at the active site of an enzyme.[43],[44]

Chapter 3

Fictitious Electron Dynamics

The combination of molecular dynamics with electronic structure calculations to provide a description of atomic dynamics in a solid on a quantum mechanical level, was discussed in section 2.8. It was also mentioned that the calculation of electronic structure in AIMD, as well as the fictitious dynamics method of Car and Parinello,[50] is done in the framework of DFT. In this study however, the emphasis will be on the development of a fictitious dynamics method[21] for electronic calculation within a semi-empirical framework. The reason for this being the potential to model larger systems faster than ab initio methods, such as DFT, would do. This links up with the desire to eventually develop this method in such a way that $O(N)$ scaling can be achieved. Although the implementation of the fictitious electron dynamics method for the calculation of electronic structure for a given atomic configuration was the primary focus of this study, some aspects of this method that are of relevance for combined atomic dynamics are explored.

The heart of the fictitious dynamics theory, as discussed in this thesis, lies in the expression of equations of motion for both the density and configuration matrix. These matrices are sufficient to describe the electronic structure of a system under study.

A brief description of these two matrices and their properties, which are used in the derivation of the equations of motion, are therefore given.

3.1 Density Matrix and Configuration Matrix

Consider the electron density in a molecule, which can be written as a sum of contributions of Ψ_i by individual electron orbitals; in the case of a closed-shell molecule, each molecular orbital is either doubly occupied or empty, thus

$$\begin{aligned} D &= 2\sum_{i}^{occ}\Psi_i^2 \\ &= 2\sum_{i}^{occ}\sum_{\lambda}\sum_{\sigma} c_{\lambda i}c_{\sigma i}\phi_\lambda\phi_\sigma \\ &= \sum_{\lambda}\sum_{\sigma} P_{\lambda\sigma}\phi_\lambda\phi_\sigma \,, \end{aligned} \tag{3.1}$$

where only real wave functions are considered.

The one-particle density matrix $P_{\lambda\sigma} = 2\sum_{i}^{occ} c_{\lambda i}^* c_{\sigma i}$ is the same as defined in equation (2.14). The quantities, $\{P_{\lambda\sigma}\}$, can be considered as measures of the extent to which the various distributions $\phi_\lambda\phi_\sigma$ contribute to the overall electron density distribution of the molecule. If equation (3.1) is followed, the density matrix can be interpreted as an element in a vector space spanned by the wave function products $\{\phi_\lambda\phi_\sigma\}$ with the matrix elements $\{P_{\lambda\sigma}\}$ as the expansion coefficients.

The density matrix is itself a projection matrix except for the case of restricted Hartree Fock theory for closed shell systems where a spinless density matrix is defined which differs by a factor two from the projection matrix i.e $\mathbf{P} = \frac{1}{2}\mathbf{P}'$, where $\mathbf{P}'$ is the density matrix and $\mathbf{P}$ the projection matrix. The projection matrix for a subspace is defined as follows:

Let $W = \mathrm{sp}(\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_k)$ be a k-dimensional subspace of R^n, and let $\mathbf{A}$ have as columns the vectors $\{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_k}\}$. The projection matrix for the subspace W is

given by

$$\mathbf{P} = \mathbf{A}(\mathbf{A^T A})^{-1}\mathbf{A^T}. \tag{3.2}$$

The density and thus projection matrix represents a complete and unique description of an electronic state. It specifies the "orientation" of the subspace associated with the electronic state within the vector space spanned by all basis functions.

The density matrix can now be written in the following form, given equation (2.14):

$$\mathbf{P}' = 2\underline{\mathbf{C}}\,\underline{\mathbf{C}}^{\dagger}\,, \tag{3.3}$$

with $\underline{\mathbf{C}}^{\dagger}$ hermitian and $\underline{\mathbf{C}}^{\dagger} = \tilde{\underline{\mathbf{C}}}$ if $\underline{\mathbf{C}}$ is real. $\underline{\mathbf{C}}$ is given by

$$\underline{\mathbf{C}} = \begin{pmatrix} c_{11} & \cdots & c_{1M} \\ \vdots & \cdots & \vdots \\ c_{N1} & \cdots & c_{NM} \end{pmatrix}\,, \tag{3.4}$$

with N the number of basis functions and M the number of occupied orbitals. The above matrix is known as the electron configuration matrix. The electron configuration matrix is not the same as the matrix, $\mathbf{C}$, defined in equation (2.19). There $\mathbf{C}$ is a square matrix containing the expansion coefficients $c_{\mu i}$. By taking only the lowest filled orbitals, the electron configuration matrix (which is a rectangular matrix indicated by the underscore) can be constructed, as given by equation (3.4). Each column in the electron configuration matrix thus represents a single particle orbital.

Now that the density (and thus projection) and configuration matrix have been defined, it is necessary to take a look at the properties of these matrices as this will assist in the development of the equations of motion in the next chapter.

3.1.1 Properties of the projection and configuration matrix of relevance in this study

In this section a number of well known properties of an NxN projection matrix, $\mathbf{P}$, and an $N \times M$ configuration matrix, $\underline{\mathbf{C}}$, are listed.

Properties of the projection matrix

- The projection matrix is both hermitian an idempotent, thus $\mathbf{P}^\dagger = \mathbf{P}$ and $\mathbf{P}^2 = \mathbf{P}$.
- The projection matrix for a specific subspace is unique.
- The trace of the projection matrix equals its rank, i.e $\mathrm{tr}(\mathbf{P}) = \mathrm{rank}(\mathbf{P}) = \frac{N}{2}$ with N the dimensionality of the subspace and can, for example, be the number of electrons in the system. N is even for closed shell systems.
- The eigenvalues of $\mathbf{P}$ are either 0 or 1. The multiplicity of the eigenvalue 1 is equal to the rank of $\mathbf{P}$.

Properties of the configuration matrix

- The columns of the electron configuration matrix are orthonormal i.e $\mathbf{C}^\dagger\mathbf{C} = \mathbf{I}$ with $\mathbf{I}$ the identity matrix.

In the next section a brief overview of a theory of idempotency conserving changes, together with the derivation of the equations of motion, is given. For a complete discussion of this theory see [21].

3.2 Mathematical Scheme

The fictitious dynamics method under discussion in this thesis, was developed in a mathematical environment in order to have access to geometrical analogies which could prove useful in suggesting methods and approximations for actual calculations. This also assists in gaining better insight into the meaning of the equations of motion. Figure 3.1 illustrates the global mathematical framework in which this theory was developed.

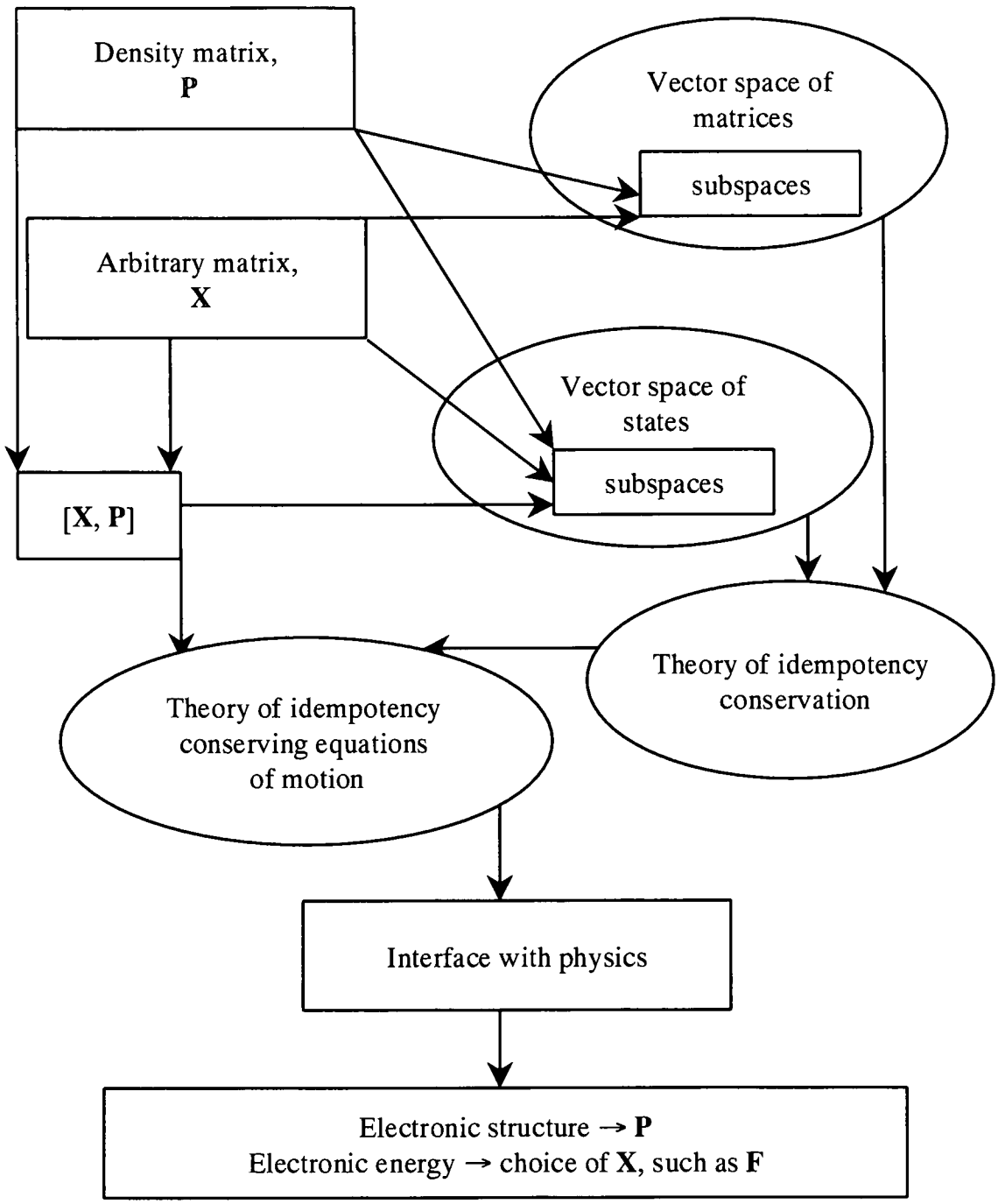

Figure 3.1: Mathematical scheme in which the fictitious electron dynamics theory was developed.

In figure 3.1, a density matrix, $\mathbf{P}$, is considered, which is an element of the vector space of matrices while its columns are elements of the vector space of states. An arbitrary matrix, $\mathbf{X}$, can also be considered, which acts as a generating matrix for idempotency conserving changes (this is explained in more detail in the section 3.3.1). The columns of $\mathbf{X}$ are elements of the vector space of states while $\mathbf{X}$ itself is an element of the vector space of matrices. There also exists an important commutation relationship between the generating matrix and the density matrix. Furthermore, the vector space of states and the vector space of matrices are connected to the theory of idempotency conservation, the latter providing the necessary background for the derivation of the idempotency conserving equations of motion. To use these equations of motion in, for example, a simulated annealing experiment, some interface with physics is necessary to calculate useful quantities such as the electronic energy. This interface is provided by making the appropriate choice for the generating matrix, $\mathbf{X}$. In the following two sections the vector space of states and the vector space of matrices are explained in detail.

3.2.1 The direct sum decomposition of the vector space of states in terms of subspaces associated with a projection matrix

In this section, the idea of a vector space and the subspaces contained within this vector space, is introduced. Expressions are also obtained for the form that matrices have in these different subspaces contained within the vector space.

Consider a column space, $V_{\mathbf{A}}$, of the $(N \times N)$ matrix, $\mathbf{A}$. $V_{\mathbf{A}}$ is a subspace of an N-dimensional vector space, V, for example $\mathcal{R}^N$ or $\mathcal{C}^N$. Let $\mathbf{P}$ be a $(N \times N)$ projection matrix of rank M and let $\mathbf{X}$ be a non-singular $(N \times N)$ matrix, where a matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]$ is non-singular only if $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ is a linearly independent set. The vector space, V, can thus be written as a direct sum of the orthogonal subspaces

$V_{\mathbf{P}}$ and $V_{\mathbf{I-P}}$:

$$V = V_{\mathbf{P}} \oplus V_{\mathbf{I-P}}. \tag{3.5}$$

Now, let

$$\mathbf{X}_1 = \mathbf{PXP}\ , \tag{3.6}$$

$$\mathbf{X}_2 = \mathbf{PX(I-P)}\ , \tag{3.7}$$

$$\mathbf{X}_3 = \mathbf{(I-P)XP}\ , \tag{3.8}$$

and

$$\mathbf{X}_4 = \mathbf{(I-P)X(I-P)}\ . \tag{3.9}$$

Thus,

$$\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2 + \mathbf{X}_3 + \mathbf{X}_4\ . \tag{3.10}$$

The geometric interpretation of these matrices becomes much clearer from the considerations below.

Let

$$V_1 = V_{\mathbf{(I-P)XP}}\ ; \tag{3.11}$$

$$V_2 = V_{\mathbf{(I-P)X(I-P)}}\ ; \tag{3.12}$$

$$V_3 = V_{\mathbf{PXP}}\ ; \tag{3.13}$$

$$V_4 = V_{\mathbf{PX(I-P)}}\ . \tag{3.14}$$

The subspaces introduced above are schematically illustrated in figure 3.2. It can be shown that the vector space of states can be written as the direct sum decomposition[21]

$$V = V_{\mathbf{PXP}} \oplus V_{\mathbf{PX(I-P)}} \oplus V_{\mathbf{(I-P)XP}} \oplus V_{\mathbf{(I-P)X(I-P)}}\ . \tag{3.15}$$

Thus, each column of a matrix $\mathbf{X}$, being an element of V, can be decomposed in accordance with equation (3.15). This statement provides a deeper geometric significance of the almost trivial looking equation (3.10). The decomposition of equation (3.10) is important for the theory of idempotency conservation discussed in the next section.

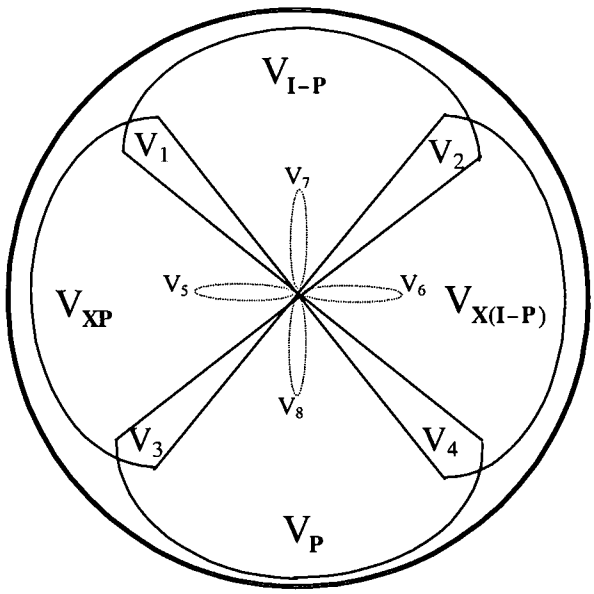

Figure 3.2: Schematic illustration[21] of the subspaces referred to in the text. The zero vector, common to all the subspaces, is represented at the center of the diagram. $\mathbf{X}$ *is a non-singular matrix. The intersection subspaces,* V_1 *to* V_4 *are given by* $V_1 = V_{\mathbf{(I-P)XP}}$, $V_2 = V_{\mathbf{(I-P)X(I-P)}}$, $V_3 = V_{\mathbf{PXP}}$, *and* $V_4 = V_{\mathbf{PX(I-P)}}$. *The complement subspaces,* V_5 *to* V_6, *are shown to be empty subspaces containing only the zero vector.* V *is the direct sum of the intersection subspaces,* V_1 *to* V_4.

3.2.2 The direct sum decomposition of the vector space of matrices in terms of subspaces associated with a projection matrix

Consider a vector space $\mathcal{M}_{N,N}$ consisting of all real $(N \times N)$ matrices. Let $\mathbf{X}$ be an arbitrary $(N \times N)$ in $\mathcal{M}_{N,N}$ with a direct sum decomposition $\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2 + \mathbf{X}_3 + \mathbf{X}_4$, with $\{\mathbf{X}_i\}$ given by equations (3.6) to (3.9) as derived in section 3.2.1.

An operator, $P_{\mathcal{M}_i}$, can now be defined which maps a matrix $\mathbf{X}$ onto its component $\mathbf{X}_i$:

$P_{\mathcal{M}_i}\mathbf{X} = \mathbf{X}_i$, with $\mathcal{M}_i$ the projection subspace of $P_{\mathcal{M}_i}$ in $\mathcal{M}_{N,N}$. It can thus be written for the operator $P_{\mathcal{M}_i}$:

$$P_{\mathcal{M}_1}\mathbf{X} = \mathbf{PXP}\ ; \tag{3.16}$$

$$P_{\mathcal{M}_2}\mathbf{X} = \mathbf{PX(I-P)}\ ; \tag{3.17}$$

$$P_{\mathcal{M}_3}\mathbf{X} = \mathbf{(I-P)XP}\ ; \tag{3.18}$$

$$P_{\mathcal{M}_4}\mathbf{X} = \mathbf{(I-P)X(I-P)}\ . \tag{3.19}$$

These operators are both linear and idempotent and also exhibit the property that they are disjoint i.e $P_{\mathcal{M}_i}\ P_{\mathcal{M}_j} = 0$, provided that $i \neq j$. Furthermore, the subspaces $\mathcal{M}_i$ can be considered to be the projection subspaces of the operator $P_{\mathcal{M}_i}$ in the vector space $\mathcal{M}_{N,N}$. There can also be symmetric projection matrices which are contained in $\mathcal{M}_i^S$ which is a subspace of $\mathcal{M}_j$.

These subspaces, $\mathcal{M}_i$, as introduced above, associated with a projection matrix, $\mathbf{P}$, of rank M further have the following properties:

- They are mutually disjoint;
- $\mathrm{Dim}(\mathcal{M}_1) = M^2$;
- $\mathrm{Dim}(\mathcal{M}_2) = \mathrm{Dim}(\mathcal{M}_3) = M(N-M)$;
- $\mathrm{Dim}(\mathcal{M}_4) = (N-M)^2$.
- $\mathrm{Dim}(\mathcal{M}_1) + \mathrm{Dim}(\mathcal{M}_2) + \mathrm{Dim}(\mathcal{M}_3) + \mathrm{Dim}(\mathcal{M}_4) = \mathrm{Dim}(\mathcal{M}_{N,N}) = N^2$;
- $$\mathcal{M}_{N,N} = \mathcal{M}_1 \oplus \mathcal{M}_2 \oplus \mathcal{M}_3 \oplus \mathcal{M}_4\ ; \tag{3.20}$$
- The subspaces $\{\mathcal{M}_i\}$ are mutually orthogonal.

For proofs of these properties see [21].

Some subspaces in the vector space, $\mathcal{M}_{N,N}$, which can be constructed from the direct sums of the existing subspaces, $\mathcal{M}_1$ - $\mathcal{M}_4$, can also be introduced. In other words:

$$\mathcal{M}_{(1;4)} = \mathcal{M}_1 \oplus \mathcal{M}_4 \,, \tag{3.21}$$

$$\mathcal{M}_{(2;3)} = \mathcal{M}_2 \oplus \mathcal{M}_3 \,. \tag{3.22}$$

It is clear that $\mathcal{M}_1$ and $\mathcal{M}_4$ are each other's orthogonal complements in $\mathcal{M}_{(1;4)}$ and $\mathcal{M}_2$ and $\mathcal{M}_3$ are each other's orthogonal complements in $\mathcal{M}_{(2;3)}$. From equation (3.20) it follows that

$$\mathcal{M}_{N,N} = \mathcal{M}_{(1;4)} \oplus \mathcal{M}_{(2;3)} \,. \tag{3.23}$$

The subspaces introduced above are schematically illustrated in figure 3.3. The elements of the subspace $\mathcal{M}_{(2;3)}$ are all of the form, $\mathbf{PA(I-P)} + \mathbf{(I-P)BP}$, with $\mathbf{A}$ and $\mathbf{B}$ arbitrary $(N \times N)$ matrices in $\mathcal{M}_{N,N}$.

Having define the subspaces $\mathcal{M}_1 - \mathcal{M}_4$, some interesting and useful properties of these subspaces, as well as the matrices contained in them, are pointed out in the next section.

3.3 Theory of idempotency conserving changes

The concept of idempotency which has been enunciated in the previous chapter will now be further explored by introducing changes in the projection matrix that conserve idempotency. It is necessary to investigate these idempotency conserving changes as it provides a necessary background for the derivation of the idempotency conserving equations of motion in the next section. In the next section general forms for first and second order equations of motion for the projection matrix that conserve the idempotency requirements, $\mathbf{P}^2 = \mathbf{P}$ and $\mathbf{P} = \tilde{\mathbf{P}}$ respectively, with appropriate initial conditions, are obtained. An idempotent matrix is usually considered to be symmetric, but situations can exist where "idempotent" matrices are non-symmetric and still satisfy the requirement, $\mathbf{P^2} = \mathbf{P}$. This is therefore referred to as the idempotency requirement and the symmetry is considered as a separate requirement.

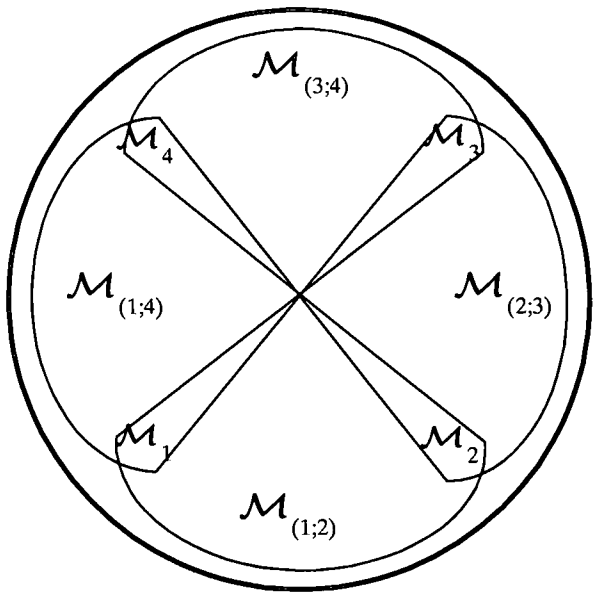

Figure 3.3: A schematic illustration[21] of the various subspaces of $\mathcal{M}_{N,N}$ associated with a projection matrix, $\mathbf{P}$, and referred to in the text. $\mathcal{M}_{N,N}$ is the vector space of all real $(N \times N)$ matrices. $\mathcal{M}_2$, for example, contains all matrices of the form $\mathbf{PX(I-P)}$. $\mathcal{M}_{(2;3)} = \mathcal{M}_2 \oplus \mathcal{M}_3$ is the "first order idempotency conserving subspace associated with $\mathbf{P}$" in $\mathcal{M}_{N,N}$. $\mathcal{M}_{(1;4)}$ is the commuting subspace of $\mathbf{P}$ in $\mathcal{M}_{N,N}$ as shown in the next section.

For a given equation of motion for $\mathbf{P}$, a small time interval, Δt, is associated with a small change, $\Delta\mathbf{P}$, in $\mathbf{P}$. Before the implications that the idempotency constraints have on the equation of motion are considered, it is instructive to first consider the general form for a first order change, $\Delta\mathbf{P}$, that conserves idempotency and symmetry.

To first order in $\Delta\mathbf{P}$ the idempotency condition, $(\mathbf{P}+\Delta\mathbf{P})^2 = \mathbf{P}+\Delta\mathbf{P}$, leads to the condition

$$\Delta\mathbf{P} = \mathbf{P}\Delta\mathbf{P} + \Delta\mathbf{P}\mathbf{P}. \tag{3.24}$$

This condition is both necessary and sufficient for conservation of idempotency to first

order. From this condition it then follows that $\Delta\mathbf{P}\mathbf{P} = (\mathbf{I} - \mathbf{P})\Delta\mathbf{P}$ and $\Delta\mathbf{P}(\mathbf{I} - \mathbf{P}) = \mathbf{P}\Delta\mathbf{P}$.

A choice for $\Delta\mathbf{P}$ that satisfies these conditions, as well as the additional symmetry requirement, is $\Delta\mathbf{P} = \mathbf{P}\mathbf{X}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\tilde{\mathbf{X}}\mathbf{P}$, where $\mathbf{X}$ could be an arbitary square matrix of the same dimension as $\mathbf{P}$. Consider now the arbitrary $(N \times N)$ matrix $\mathbf{X}$. If $\mathbf{X}$ is multiplied from the left and the right by $\mathbf{P} + (\mathbf{I} - \mathbf{P})$ its decomposition in terms of its "projected parts" is then given by[21]

$$\mathbf{X} = \mathbf{P}\mathbf{X}\mathbf{P} + \mathbf{P}\mathbf{X}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\mathbf{X}\mathbf{P} + (\mathbf{I} - \mathbf{P})\mathbf{X}(\mathbf{I} - \mathbf{P}) \,, \tag{3.25}$$

in accordance with the direct sum decomposition of equation (3.15).

3.3.1 Theory of first order idempotency conserving changes

The changes, $\Delta\mathbf{P}$ on $\mathbf{P}$, that conserve idempotency to at least the first order in $\Delta\mathbf{P}$ can now be considered. The idempotency requirement $(\mathbf{P} + \Delta\mathbf{P})^2 = \mathbf{P} + \Delta\mathbf{P}$, leads to the condition given by equation (3.24). An choice for $\Delta\mathbf{P}$ that satisfies this equation, is given by

$$\Delta\mathbf{P} = \mathbf{P}\mathbf{X}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\mathbf{Y}\mathbf{P}, \tag{3.26}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are arbitrary $(N \times N)$ matrices.
If $\Delta\mathbf{P}$ is required to be symmetric then

$$\Delta\mathbf{P} = \mathbf{P}\mathbf{X}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\tilde{\mathbf{X}}\mathbf{P} \,, \tag{3.27}$$

with $\mathbf{X}$ an arbitrary $(N \times N)$ matrix.

The change, $\Delta\mathbf{P} = \mathbf{P}\mathbf{X}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\tilde{\mathbf{X}}\mathbf{P}$, is referred to as the symmetric first order idempotency conserving change generated by $\mathbf{X}$. $\Delta\mathbf{P}$ should of course be scaled appropriately to ensure that the higher order terms are as small as required by the accuracy of a calculation.

It was seen in section 3.2.2 that the matrices contained in $\mathcal{M}_{(2;3)}$ are of the form $\mathbf{P}\mathbf{A}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{B}\mathbf{P}$. Now it is clear that $\Delta\mathbf{P} = \mathbf{P}\Delta\mathbf{P}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\Delta\mathbf{P}\mathbf{P}$. It can

thus be seen that $\Delta\mathbf{P} \epsilon \mathcal{M}_{(2;3)}$. If the additional symmetry requirement is considered it can be seen that $\mathbf{P}$ will only be symmetric if $\Delta\mathbf{P}$ is symmetric. Thus $\Delta\mathbf{P} \epsilon \mathcal{M}^{S}_{(2;3)}$. Following the above $\mathcal{M}_{(2;3)}$ can be referred to as the first order idempotency conserving subspace associated with $\mathbf{P}$.

It is thus clear that a first order idempotency conserving change, $\Delta\mathbf{P}$ on $\mathbf{P}$, can be simply obtained from an arbitrary matrix, say $\mathbf{X}$, in $\mathcal{M}_{N,N}$, as the projection of $\mathbf{X}$ onto the "first order idempotency conserving subspace" associated with $\mathbf{P}$, that is onto $\mathcal{M}_{(2;3)}$. That is

$$\Delta\mathbf{P} = \mathbf{PX}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{XP} \tag{3.28}$$

$$= (\mathbf{I}-2\mathbf{P})[\mathbf{X},\mathbf{P}]\,. \tag{3.29}$$

It can indeed be shown in [21] that all first order idempotency conserving changes, $\Delta\mathbf{P}$ on $\mathbf{P}$, can be written in this form. $\Delta\mathbf{P}$ should of course be appropriately scaled to ensure that the higher order terms are as small as required by the accuracy of a calculation. Since the projection matrix, $\mathbf{P}_{\mathcal{M}_{(2;3)}} = \mathbf{P}_{\mathcal{M}_2} + \mathbf{P}_{\mathcal{M}_3}$, for $\mathcal{M}_{(2;3)}$, is singular, the relationship between $\Delta\mathbf{P}$ and the generating matrix, $\mathbf{X}$, is non-unique. This non-uniqueness is analyzed in [21].

3.3.2 A geometric analogy for the idempotency conserving conditions

In the previous section the theory of first order idempotency conserving changes was considered. In order to get a clear picture of what this entails, the idempotency requirement can be interpreted geometrically. This can be done by first considering the exact idempotency conserving condition on $\Delta\mathbf{P}$ which is given by

$$\Delta\mathbf{P} = \mathbf{P}\,\Delta\mathbf{P} + \Delta\mathbf{P}\,\mathbf{P} + \Delta\mathbf{P}^2\,. \tag{3.30}$$

It is known that the rank of a projection matrix is equal to its trace. Thus, the change in the rank of $\mathbf{P}$, is given by $\mathrm{tr}(\Delta\mathbf{P})$. When a rank conserving change, for which

$\mathrm{tr}(\Delta\mathbf{P}) = 0$, is considered, the necessary though not sufficient condition $\mathrm{tr}((2\mathbf{P} + \Delta\mathbf{P})\Delta\mathbf{P}) = 0$ can be obtained by using the property, $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$. This condition can also be written as

$$\begin{aligned} \mathrm{tr}((\mathbf{P}+\mathbf{P}')\Delta\mathbf{P}) &= 0 \\ \langle(\mathbf{P}+\mathbf{P}'),\Delta\mathbf{P}\rangle &= 0\,, \end{aligned} \tag{3.31}$$

where $\mathbf{P}' = \mathbf{P} + \Delta\mathbf{P}$ and the above condition is valid to any order. The first order version of this condition is

$$\begin{aligned} \mathrm{tr}(\mathbf{P}\Delta\mathbf{P}) &= 0 \\ \langle\mathbf{P},\Delta\mathbf{P}\rangle &= 0\,. \end{aligned} \tag{3.32}$$

In the above equations, use have been made of the trace scalar product. The trace scalar product between two matrices, $\mathbf{A}$ and $\mathbf{B}$, is defined as $\langle\mathbf{A},\mathbf{B}\rangle = \mathrm{tr}(\mathbf{A}^\dagger\mathbf{B})$. For the case where $\mathbf{A}$ is hermitian, then $\langle\mathbf{A},\mathbf{B}\rangle = \mathrm{tr}(\mathbf{AB})$. Associated with this scalar product, is the Euclidian norm, $\|\mathbf{A}\| = \langle\mathbf{A}^\dagger,\mathbf{A}\rangle^{\frac{1}{2}}$.

For a rank conserving change, $\mathrm{tr}(\mathbf{P}) = \mathrm{tr}(\mathbf{P}')$, and thus $\mathrm{tr}(\mathbf{P}^2) = \mathrm{tr}((\mathbf{P}')^2)$. It then follows that

$$\langle\mathbf{P},\mathbf{P}\rangle = \langle\mathbf{P}',\mathbf{P}'\rangle\,. \tag{3.33}$$

Equation (3.33) thus implies that $\mathbf{P}$ and $\mathbf{P}'$ should be of the same length. This emphasizes the difference between the first order and exact conditions. The conditions (3.32) to (3.33) are schematically illustrated in figure 3.3.2. Another geometric interpretation, related to the necessary conditions as discussed above, can be given to the idempotency conserving condition. For an idempotent matrix, $\mathbf{P}$, it follows that $\mathrm{tr}(\mathbf{P}(\mathbf{I}-\mathbf{P})) = 0$. Thus, the necessary condition,

$$\langle\mathbf{P},(\mathbf{I}-\mathbf{P})\rangle = 0\,, \tag{3.34}$$

is obtained.

To supply the sufficient condition for $\mathbf{P}$ to be idempotent, equation (3.34) as well as the eigenvalues, p_i of $\mathbf{P}$, are considered. The eigenvalues of $\mathbf{P}^2$ are then $\{p_i^2\}$. Furthermore,

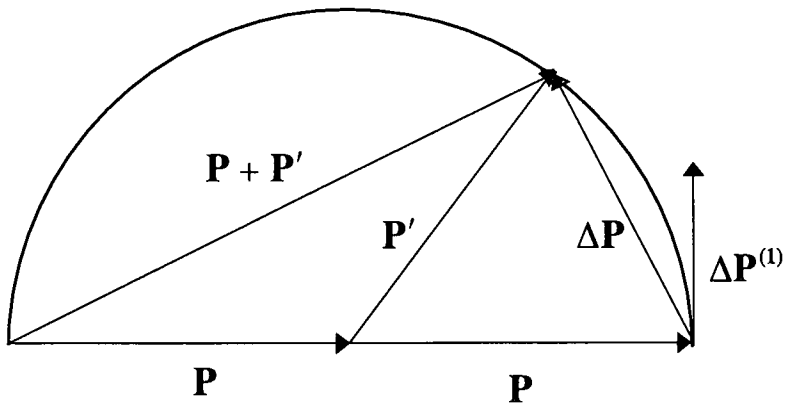

Figure 3.4: Geometric interpretation[21] of the conditions, $\|\mathbf{P}\| = \|\mathbf{P}'\|$, $\langle(\mathbf{P} + \mathbf{P}'), \Delta\mathbf{P}\rangle = 0$, *and* $\langle \mathbf{P}, \Delta\mathbf{P}^{(1)}\rangle = 0$.

$\mathrm{tr}(\mathbf{P}) = \sum_i p_i$, and $\mathrm{tr}(\mathbf{P}^2) = \sum_i p_i^2$. If $\mathrm{tr}(\mathbf{P} - \mathbf{P}^2) = 0$, it follows that $\sum_i p_i(1 - p_i) = 0$. If the eigenvalues, $\{p_i\}$, are now restricted to lie in the interval [0,1], this can only be possible if the $\{p_i\}$ are either zero or one, which is a sufficient condition for $\mathbf{P}$ to be idempotent.

In the above, some differences in the geometric content of the first order and exact idempotency conserving conditions were discussed. An analysis of the implications of the exact condition can be found in [21].

3.4 Idempotency conserving equations of motion

A time-dynamic generalization of the theory of idempotency conserving changes, will clarify the implications of the idempotency constraints for the time derivatives of $\mathbf{P}$. In this section, general forms for first and second order equations of motion for $\mathbf{P}$ that conserve the idempotency requirements, $\mathbf{P}^2 = \mathbf{P}$ and $\mathbf{P} = \tilde{\mathbf{P}}$, with appropriate initial

conditions, were obtained. Equations for the first time derivatives and subsequently for the second time derivatives are considered.

3.4.1 General form of solutions for $\dot{\mathbf{P}}$

Let $\mathbf{P}^2 = \mathbf{P}$ at $t = 0$. Taking the derivative of $\mathbf{P}$ with respect to time results in

$$\dot{\mathbf{P}} = \frac{d}{dt}\left(\mathbf{P}^2\right) = \mathbf{P}\dot{\mathbf{P}} + \dot{\mathbf{P}}\mathbf{P} \; . \tag{3.35}$$

For idempotency conservation as well as the additional requirement of symmetry to the first order, it is thus required that equation (3.35) together with $\dot{\mathbf{P}} = \tilde{\dot{\mathbf{P}}}$ be valid for all t, and $\mathbf{P} = \tilde{\mathbf{P}}$ must be valid at $t = 0$.

It can be seen from equation (3.35) that the condition for $\dot{\mathbf{P}}$ is of the same form as that of the first order condition of equation (3.24). Thus, the theory of first order idempotency conserving changes as discussed in section 3.3.1 can be applied to $\dot{\mathbf{P}}$. Since the condition of equation (3.35) is exact, $\dot{\mathbf{P}}$ need not be small. A solution for $\dot{\mathbf{P}}$ can now be obtained through the following argumentation:

First equation (3.35) is rewritten as $\dot{\mathbf{P}}\mathbf{P} = (\mathbf{I} - \mathbf{P})\dot{\mathbf{P}}$. Using the symmetry of $\mathbf{P}$ and $\dot{\mathbf{P}}$, it follows that

$$\mathbf{P}\dot{\mathbf{P}} = [(\mathbf{I} - \mathbf{P})\dot{\mathbf{P}}]^T \; . \tag{3.36}$$

Now, let $\mathbf{A} = \mathbf{P}\dot{\mathbf{P}}$, with $\mathbf{A}$ a matrix in the subspace $\mathcal{M}_2$. Substituting the expression for $\mathbf{A}$ into equation (3.36) results in

$$\tilde{\mathbf{A}} = (\mathbf{I} - \mathbf{P})\dot{\mathbf{P}} \; . \tag{3.37}$$

Thus,

$$\dot{\mathbf{P}} = \mathbf{A} + \tilde{\mathbf{A}} \; . \tag{3.38}$$

The columns of $\mathbf{A}$ lie in the column space, $V_{\mathbf{P}}$, of $\mathbf{P}$ while the columns of $\tilde{\mathbf{A}}$ lie in the orthogonal complement, $V_{\mathbf{P}}^{\perp}$, of $V_{\mathbf{P}}$. Thus, $(\tilde{\mathbf{A}})^T\mathbf{A} = \mathbf{0}$, or $\mathbf{A}^2 = \mathbf{0}$. $\mathbf{A}$ furthermore has

the following two properties:

$$\mathbf{PA} = \mathbf{A}\,; \tag{3.39}$$

$$\mathbf{P}\tilde{\mathbf{A}} = \mathbf{0}\,. \tag{3.40}$$

Equation (3.38) represents the general form of solutions for $\dot{\mathbf{P}}$, consistent with idempotency conservation, with $\mathbf{A}$ any matrix in $\mathcal{M}_2$. $\mathbf{A}$ is nilpotent of degree two.

From equation (3.35) the commutation relationship between $\mathbf{P}$ and $\dot{\mathbf{P}}$ can also be derived. Premultiplying equation (3.35) with $\dot{\mathbf{P}}$ results in

$$\dot{\mathbf{P}}^2 = \dot{\mathbf{P}}\mathbf{P}\dot{\mathbf{P}} + \dot{\mathbf{P}}^2\mathbf{P}\,, \tag{3.41}$$

and postmultiplying gives

$$\dot{\mathbf{P}}^2 = \mathbf{P}\dot{\mathbf{P}}^2 + \dot{\mathbf{P}}\mathbf{P}\dot{\mathbf{P}}\,. \tag{3.42}$$

By comparing the right hand sides of equations (3.41) and (3.42), it can be concluded that $\mathbf{P}\dot{\mathbf{P}}^2 = \dot{\mathbf{P}}^2\mathbf{P}$ i.e $\left[\mathbf{P}, \dot{\mathbf{P}}^2\right] = 0$.

Thus, $\dot{\mathbf{P}}^2$, which is proportional to $\Delta\mathbf{P}^2$, lies in the commuting subspace of $\mathcal{M}_{(1;4)}$ of $\mathbf{P}$ as mentioned in section 3.3.1. The following identities are introduced since they are used in the discussion below. Using the nilpotency of $\mathbf{A}$, it follows that

$$\dot{\mathbf{P}}^2 = \tilde{\mathbf{A}}\mathbf{A} + \mathbf{A}\tilde{\mathbf{A}}\,. \tag{3.43}$$

Using equations (3.39) and (3.40) leads to

$$\mathbf{P}\dot{\mathbf{P}}^2 = \mathbf{A}\tilde{\mathbf{A}}\,; \tag{3.44}$$

$$(\mathbf{I} - \mathbf{P})\dot{\mathbf{P}}^2 = \tilde{\mathbf{A}}\mathbf{A}\,. \tag{3.45}$$

A general form of solutions for $\dot{\mathbf{P}}$, consistent with idempotency conservation, was considered in this section and consequently a general solution to the second order equation of motion are derived in the next section.

3.4.2 General form of solutions for $\ddot{\mathbf{P}}$

Let $\mathbf{P}^2 = \mathbf{P}$ and $\dot{\mathbf{P}} = \mathbf{P}\dot{\mathbf{P}} + \dot{\mathbf{P}}\mathbf{P}$ at $t = 0$. Taking the second derivative of $\mathbf{P}$ with respect to time leads to

$$\ddot{\mathbf{P}} = \frac{d^2}{dt^2}\left(\mathbf{P}^2\right) = \mathbf{P}\ddot{\mathbf{P}} + \ddot{\mathbf{P}}\mathbf{P} + 2\dot{\mathbf{P}}^2 \tag{3.46}$$

For idempotency conservation as well as the additional requirement of symmetry to the second order, it is thus required that equation (3.46) together with $\ddot{\mathbf{P}} = \tilde{\ddot{\mathbf{P}}}$ be valid for all t, and equation (3.35), $\dot{\mathbf{P}} = \tilde{\dot{\mathbf{P}}}$ and $\mathbf{P} = \tilde{\mathbf{P}}$ must be valid at $t = 0$.

A solution for $\ddot{\mathbf{P}}$ can now be found following the compact derivation of [21] as a starting point. A more intuitive, though much longer, treatment can also be found in [21]. The similarity between equations (3.30) and (3.46) is emphasised.

By adding and subtracting a term $4\mathbf{P}\dot{\mathbf{P}}^2$ on the right hand side of equation (3.46), using the commutivity relationship between $\mathbf{P}$ and $\dot{\mathbf{P}}^2$ (as derived in the previous section), as well as an appropriate grouping of terms, it follows that

$$\ddot{\mathbf{P}} = \mathbf{P}\ddot{\mathbf{P}} + 2\mathbf{P}\dot{\mathbf{P}}^2 + (\mathbf{P}\ddot{\mathbf{P}} + 2\mathbf{P}\dot{\mathbf{P}}^2)^T + 2(\mathbf{I} - 2\mathbf{P})\dot{\mathbf{P}}^2 \,. \tag{3.47}$$

Now, let $\mathbf{Y}$ be a nilpotent matrix of index 2 (with columns in $V_{\mathbf{P}}$) in the subspace $\mathcal{M}_2$ with the following form:

$$\mathbf{Y} = \mathbf{P}\ddot{\mathbf{P}} + 2\mathbf{P}\dot{\mathbf{P}}^2 \,. \tag{3.48}$$

By substituting $\mathbf{P}\ddot{\mathbf{P}} + 2\mathbf{P}\dot{\mathbf{P}}^2$ with $\mathbf{Y}$ into equation (3.47) it follows that

$$\ddot{\mathbf{P}} = \mathbf{Y} + \tilde{\mathbf{Y}} + 2(\mathbf{I} - 2\mathbf{P})\dot{\mathbf{P}}^2 \,. \tag{3.49}$$

It is clear that $\mathbf{PY} = \mathbf{Y}$ and $\tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}\mathbf{P}$. Taking the $\mathbf{P}$-projection of equation (3.49), $\mathbf{P}\ddot{\mathbf{P}} = \mathbf{Y} + \mathbf{P}\tilde{\mathbf{Y}} - 2\mathbf{P}\dot{\mathbf{P}}^2$ is obtained. From this and equation (3.48), it follows that $\mathbf{Y} + \mathbf{P}\tilde{\mathbf{Y}} = \mathbf{P}\ddot{\mathbf{P}} + 2\mathbf{P}\dot{\mathbf{P}}^2 = \mathbf{Y}$ and that $\mathbf{P}\tilde{\mathbf{Y}} = \mathbf{0}$ or $\mathbf{YP} = \mathbf{0}$. It is thus clear that all the columns of $\tilde{\mathbf{Y}}$ lie in the space $V_{\mathbf{P}}^{\perp}$. Thus, $\mathbf{Y}^2 = \mathbf{0}$.

Conversely a matrix, $\ddot{\mathbf{P}} = \mathbf{Y} + \tilde{\mathbf{Y}} + 2(\mathbf{I} - 2\mathbf{P})\dot{\mathbf{P}}^2$, can be considered, with $\mathbf{Y}$ any matrix that satisfies the conditions $\mathbf{PY} = \mathbf{Y}$ or $\mathbf{P}\tilde{\mathbf{Y}} = \mathbf{0}$. Then $\ddot{\mathbf{P}}$ satisfies the condition $\ddot{\mathbf{P}} = \mathbf{P}\ddot{\mathbf{P}} + \ddot{\mathbf{P}}\mathbf{P} + 2\dot{\mathbf{P}}^2$.

From the discussion following equation (3.49) it is thus clear that the general form of solutions of $\ddot{\mathbf{P}}$ is given by equation (3.49), where $\mathbf{Y}$ can be any matrix in $\mathcal{M}_2$.

Other than conserving idempotency and symmetry, the equation of motion (equation (3.49)) should also conserve the rank of the density matrix. The rank of the density matrix corresponds to the dimension of the occupied subspace associated with the occupied orbitals. It is shown in [21] that this equation of motion indeed conserves the rank of $\mathbf{P}$.

Until thus far, the conservation of idempotency and symmetry were treated as mathematical requirements. Some physical content has to be introduced into equation (3.49) so as to render it useful for calculations of electronic structure. This can be done by focussing on an appropriate choice for the matrix, $\mathbf{Y}$.

3.5 Appropriate choice for the matrix Y

In the derivation of the equation of motion for second order dynamics it was seen that the matrix $\mathbf{Y}$ should be an element of $\mathcal{M}_2$ satisfying the conditions $\mathbf{PY} = \mathbf{Y}$ and $\mathbf{YP} = \mathbf{0}$. The general expression for $\mathbf{Y}$ is given by:

$$\mathbf{Y} = \mathbf{PB}\,(\mathbf{I} - \mathbf{P}) \ , \tag{3.50}$$

where $\mathbf{B}$ is an arbitrary square matrix of the same dimension as $\mathbf{P}$. The equation of motion (3.49) can thus be written as

$$\ddot{\mathbf{P}} = \mathbf{PB}(\mathbf{I} - \mathbf{P}) + (\mathbf{I} - \mathbf{P})\tilde{\mathbf{B}}\mathbf{P} + 2(\mathbf{I} - 2\mathbf{P})\dot{\mathbf{P}}^2 \ . \tag{3.51}$$

It is shown in reference [21] that, without loss of generality, $\mathbf{B}$ can be considered to be symmetric.

By using equation (3.50), the symmetry for $\mathbf{B}$, and by adding and subtracting the term, $\mathbf{PB}$, the following expression for $\mathbf{W} = \mathbf{Y} + \tilde{\mathbf{Y}}$ is obtained:

$$\mathbf{W} = (\mathbf{I} - 2\mathbf{P})[\mathbf{B}, \mathbf{P}] \,. \tag{3.52}$$

Combining this with equation (3.49), leads to the following elegant form for the equation of motion:

$$\ddot{\mathbf{P}} = (\mathbf{I} - 2\mathbf{P}) \left([\mathbf{B}, \mathbf{P}] + 2\dot{\mathbf{P}}^2 \right) . \tag{3.53}$$

In a simulated annealing calculation the eventual equilibrium configuration for $\mathbf{P}$, where $\dot{\mathbf{P}} = \mathbf{0}$ and $\ddot{\mathbf{P}} = \mathbf{0}$, is of fundamental importance since this should correspond to the ground state electronic structure. As far as the equilibrium configuration associated with equation (3.53) is concerned, it needs to be evaluated when $\ddot{\mathbf{P}} = \mathbf{0}$.

If it is now supposed that $\dot{\mathbf{P}} = \mathbf{0}$ and $\ddot{\mathbf{P}} = \mathbf{0}$, it follows by inspection that $(\mathbf{I} - 2\mathbf{P})[\mathbf{B}, \mathbf{P}] = \mathbf{0}$. However, $(\mathbf{I} - 2\mathbf{P})$ is a reflection matrix and is thus non-singular. Thus, $(\mathbf{I} - 2\mathbf{P})^{-1} = (\mathbf{I} - 2\mathbf{P})$. From this it then follows that $[\mathbf{B}, \mathbf{P}] = \mathbf{0}$. Conversely, should $\dot{\mathbf{P}} = \mathbf{0}$ and $[\mathbf{B}, \mathbf{P}] = \mathbf{0}$, it follows that $\ddot{\mathbf{P}} = \mathbf{0}$. Thus, if $\dot{\mathbf{P}} = 0$, $\ddot{\mathbf{P}} = \mathbf{0}$ if and only if $[\mathbf{B}, \mathbf{P}] = \mathbf{0}$.

The above statement can thus be used to add the necessary physical content to the equation of motion in order to use it successfully in, for example, simulated annealing experiments.

3.6 A fictitious dynamics equation of motion for simulated annealing experiments

In section 2.3 the Roothaan approximation, where the Hartree-Fock approximation is applied by expanding molecular wavefunctions in terms of a set of basis functions, was discussed. The interdependence between the Fock matrix and the density matrix was

also explicitly shown in equation (2.14). In a self-consistent solution of the electronic structure, these two matrices commute i.e $[\mathbf{P},\mathbf{F}] = \mathbf{0}$.[101]

Considering that $[\mathbf{P},\mathbf{F}] = \mathbf{0}$ and $[\mathbf{B},\mathbf{F}] = \mathbf{0}$ at equilibrium, it can be seen that if $\mathbf{P}$ commutes with $\mathbf{F}$ it should also commute with $\mathbf{B}$. It can thus be argued that an appropriate choice for $\mathbf{B}$ in equation (3.53) would therefore be $\mathbf{B} = b\mathbf{F}$, with b an arbitrary constant.

To identify the most effective choice for $\mathbf{B}$ as far as approaching the equilibrium electronic configuration is concerned, the electronic energy, E, can be considered as a function of $\mathbf{F}$ and $\mathbf{P}$, that is $E = E(\mathbf{F}(\mathbf{P}),\mathbf{P})$. For a given change $\Delta\mathbf{P}$ in the electronic configuration there is an associated change ΔE in electronic energy, with $\Delta\mathbf{P}$ given by equation (3.28). The general form for $\Delta\mathbf{P}$ that would conserve idempotency and symmetry for $\mathbf{P}$ to the first order is

$$\Delta\mathbf{P} = \mathbf{PB}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{BP}\,, \tag{3.54}$$

with $\mathbf{B}$ an arbitrary, though appropriately scaled matrix.

Now, an expression for the associated change in the electronic energy within the Roothaan method is given by[21]

$$\begin{aligned}\Delta E &= 2\mathrm{tr}(\mathbf{F}\Delta\mathbf{P}) \\ &= 2\langle\mathbf{F},\Delta\mathbf{P}\rangle\,,\end{aligned} \tag{3.55}$$

where use has been made of the trace scalar product as introduced in section 3.3.2.

ΔE will be an extremum when $\Delta\mathbf{P} = b\mathbf{F}$ [21], with b an arbitrary constant. In terms of the concept of N^2-dimensional "rolled out" vectors as introduced in [103], for example, this result can be interpreted as implying that the "vectors" $\Delta\mathbf{P}$ and $\mathbf{F}$ should be parallel or anti-parallel.

$\mathbf{F}$ can now be written in terms of its projected parts

$$\mathbf{F} = \mathbf{PFP} + \mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP} + (\mathbf{I}-\mathbf{P})\mathbf{F}(\mathbf{I}-\mathbf{P})\,,$$

as explained in section 3.2.1.

Now, using the expression for $\Delta\mathbf{P}$ in equation (3.54), the fact that $\mathbf{P}(\mathbf{I}-\mathbf{P})=0$ and $\langle\mathbf{A},\mathbf{B}\rangle = \langle\mathbf{B},\mathbf{A}\rangle$, the following is obtained

$$\begin{aligned}\Delta E &= 2\mathrm{tr}\left((\mathbf{PF}(\mathbf{I}-\mathbf{P})+(\mathbf{I}-\mathbf{P})\mathbf{FP})\times(\mathbf{PB}(\mathbf{I}-\mathbf{P})+(\mathbf{I}-\mathbf{P})\mathbf{BP})\right)\\ &= 2\langle\mathbf{Q},\Delta\mathbf{P}\rangle\ ,\end{aligned} \tag{3.56}$$

with

$$\mathbf{Q}=\mathbf{PF}(\mathbf{I}-\mathbf{P})+(\mathbf{I}-\mathbf{P})\mathbf{FP}\ , \tag{3.57}$$

which is nothing else than the projection of the Fock matrix onto the first order idempotency conserving subspace, $\mathcal{M}_{(2;3)}$. ΔE will be an extremum with respect to $\Delta\mathbf{P}$, as constrained by the idempotency condition, when [21]

$$\Delta\mathbf{P}=b\mathbf{Q}\ , \tag{3.58}$$

or

$$P_{\mathcal{M}_{(2;3)}}\mathbf{B}=bP_{\mathcal{M}_{(2;3)}}\mathbf{F}\ , \tag{3.59}$$

with b an arbitrary constant. Thus, $\mathbf{B}$ and $\mathbf{F}$ have the same components in $\mathcal{M}_{2;3)}$, the first order idempotency conserving subspace. Thus

$$\mathbf{B}=b\mathbf{F}+\mathbf{C}\ , \tag{3.60}$$

where $\mathbf{C}$ is any matrix such that $P_{\mathcal{M}_{(2,3)}}\mathbf{C}=0$. The change, $\Delta\mathbf{P}$, associated with $\mathbf{B}$ above is given by

$$\Delta\mathbf{P}=b\left(\mathbf{PF}(\mathbf{I}-\mathbf{P})+(\mathbf{I}-\mathbf{P})\mathbf{FP}\right)\ . \tag{3.61}$$

It is clear that the matrix $\mathbf{C}$ is of no consequence in the resulting expression for $\Delta\mathbf{P}$.

It is concluded that

$$\mathbf{B} = b\mathbf{F} \,, \tag{3.62}$$

with b an arbitrary constant. Since $\mathbf{F}$ is symmetric, this choice of $\mathbf{B}$ is also symmetric. Thus, the appropriate change in the electronic configuration, $\Delta\mathbf{P}$, conserving idempotency to the first order, is "generated" specifically by the Fock matrix, $\mathbf{F}$.

It is necessary to determine what the sign of b should be in order to ensure that the fictitious acceleration is directed towards the equilibrium configuration. From $\Delta E = 2\langle \mathbf{Q}, \Delta\mathbf{P} \rangle$ and $\Delta\mathbf{P} = b\mathbf{Q}$ it follows that

$$\Delta E = \frac{2}{b} tr(\Delta\mathbf{P})^2 \,. \tag{3.63}$$

Since $\Delta\mathbf{P}$ is symmetric, it follows that $tr(\Delta\mathbf{P})^2 \geq 0$.

To ensure that the fictitious acceleration, $\ddot{\mathbf{P}}$, is directed towards the equilibrium configuration, it is required that $\Delta E < 0$, and thus $b < 0$.

This then leads to the expression of the equation of motion obtained from equation (3.53) which can be used in a simulated annealing calculation of the electronic structure:

$$\ddot{\mathbf{P}} = (\mathbf{I} - 2\mathbf{P}) \left(b[\mathbf{F}, \mathbf{P}] + 2\dot{\mathbf{P}}^2 \right) , \tag{3.64}$$

with $b < 0$. This can be written as

$$\mu\ddot{\mathbf{P}} = (\mathbf{I} - 2\mathbf{P})([\mathbf{F}, \mathbf{P}] + 2\mu\dot{\mathbf{P}}^2) \,, \tag{3.65}$$

where $\mu = \frac{1}{b}$ can be interpreted as a fictitious mass, with the right hand side of the above equation representing the fictitious forces that induce the constraint forces.

Equation (3.65) will conserve the idempotency and symmetry of $\mathbf{P}$ and ensure the correct ground state electronic structure at equilibrium. Equation (3.64) can also be written in the more symmetric form

$$\ddot{\mathbf{P}} = b(\mathbf{FP} + \mathbf{PF} - 2\mathbf{PFP}) + 2(\mathbf{I} - 2\mathbf{P})\dot{\mathbf{P}}^2 \,. \tag{3.66}$$

Another useful form that will be used later, is obtained by using equation (3.43):

$$\ddot{\mathbf{P}} = b(\mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP}) + 2(\tilde{\mathbf{A}}\mathbf{A} - \mathbf{A}\tilde{\mathbf{A}}) \; . \tag{3.67}$$

Now, from $\Delta E = 2\langle \mathbf{Q}, \Delta\mathbf{P}\rangle$, other useful forms for $\Delta\mathbf{P}$ are now obtained considering the change $\Delta\mathbf{P}$ that occurs over a time interval Δt due to the equation of motion (3.51), with $b\mathbf{F}$ substituted for $\mathbf{B}$, for the case $\dot{\mathbf{P}} = \mathbf{0}$. It follows that

$$\begin{aligned}
\Delta\mathbf{P} &= \frac{1}{2}(\Delta t)^2 b\,(\mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP}) \\
&= \frac{1}{2}(\Delta t)^2 b\,(\mathbf{PF} + \mathbf{FP} - 2\mathbf{PFP}) \\
&= \frac{1}{2}(\Delta t)^2 b(\mathbf{I}-2\mathbf{P})[\mathbf{F},\mathbf{P}] \\
&= b'(\mathbf{I}-2\mathbf{P})[\mathbf{F},\mathbf{P}] \; .
\end{aligned} \tag{3.68}$$

To obtain the associated first order energy change use can be made of equation (3.55), and the above expression for $\Delta\mathbf{P}$. It is noted that the trace of a matrix is not changed on taking the transpose of the matrix, and the trace of a matrix product is not changed under a cyclic permutation of the matrices. It follows that

$$\Delta E = 2(\Delta t)^2 b \; \mathrm{tr}(\mathbf{F}(\mathbf{I}-\mathbf{P})\mathbf{FP}) \; . \tag{3.69}$$

Consider the expression for ΔE in equation (3.69). The trace of the matrix product will be unchanged under a unitary transformation of the matrices. Since $\mathbf{F}$ is symmetric, it is similar to a diagonal matrix. It can therefore be assumed without loss of generality that the basis set is chosen so that $\mathbf{F}$ is diagonal, that is

$$F_{ij} = \epsilon_i \delta_{ij}, \tag{3.70}$$

with $\{\epsilon_i\}$ the real eigenvalues of $\mathbf{F}$.

Now, let $b'' = 2(\Delta t)^2 b = 4\mathrm{b}'$. If the expression for F_{ij} and b'' is substituted into equation (3.69) it follows that

$$\Delta E = b'\left(\sum_i \epsilon_i^2 P_{ii} - \sum_{ik} \epsilon_i \epsilon_k P_{ik}^2\right)$$

for the first order energy change.

From the symmetry and idempotency conditions it follows that $P_{ii} = \sum_k P_{ik}^2$. Thus

$$\Delta E = b'' \sum_{ik} \epsilon_i (\epsilon_i - \epsilon_k) P_{ik}^2 \, .$$

On splitting the summation into a sum with $i < k$ and a sum with $i > k$, and using the symmetry of $\mathbf{P}$, it follows that

$$\Delta E = \frac{b''}{2} \sum_{ik} (\epsilon_i - \epsilon_k)^2 P_{ik}^2 \, . \tag{3.71}$$

It is shown in [21] that this can also be written in the forms:

$$\Delta E = N^2 b'' \langle P_{ik}^2 \rangle_{w;i \neq k} \langle (\Delta \epsilon_i)^2 \rangle \; ; \tag{3.72}$$

$$\Delta E \geq N^2 b'' \langle P_{ik}^2 \rangle_{w;i \neq k} \langle (\Delta F_{ii})^2 \rangle \, , \tag{3.73}$$

with $\langle P_{ik}^2 \rangle_{w;i \neq k}$ a weighted average of the non-diagonal elements of $\mathbf{P}$ with weighting factors $\{(\epsilon_i - \epsilon_k)^2\}$.

It can thus be concluded that the change in ΔE is determined by the following[21]:

- The extent to which $\mathbf{P}$ can be transformed into a diagonal matrix by the same unitary transformation that diagonalizes $\mathbf{F}$.

- The mean squared deviation in the eigenvalues of $\mathbf{F}$ which is greater than or equal to the mean squared deviation of its diagonal elements.

- If it is assumed that $\langle (\Delta \epsilon_i)^2 \rangle > 0$, it is clear that ΔE is zero if and only if $\mathbf{P}$ is diagonal. This is just a recovery of the equilibrium condition, $[\mathbf{P}, \mathbf{F}] = \mathbf{0}$, according to which $\mathbf{P}$ and $\mathbf{F}$ can be diagonalized by the same similarity transformation.

3.7 $\underline{\mathbf{C}}$-dynamics equations of motion

It is often convenient to represent the electronic state in terms of molecular orbitals as expressed by the configuration matrix, $\underline{\mathbf{C}}$, as introduced in section 3.1. Each column of $\underline{\mathbf{C}}$ represents an occupied molecular orbital and contains the expansion coefficients of this orbital in terms of the chosen set of basis functions. It was also shown in section 3.1 that the associated projection matrix, $\mathbf{P}$, is given by $\mathbf{P} = \underline{\mathbf{C}}\underline{\tilde{\mathbf{C}}}$ and that it is indeed unique for a given subspace, $V_{\mathbf{P}}$.

The $\mathbf{P}$-dynamics generated by the equation of motion (3.66) has an associated $\underline{\mathbf{C}}$-dynamics. Contrary to $\mathbf{P}$, $\underline{\mathbf{C}}$ is however not unique for a given subspace V_P. This non-uniqueness should thus be taken into account. The $\mathbf{P}$-dynamics is incapable of providing information concerning the projection of the first derivative, $\underline{\dot{\mathbf{C}}}$, onto the space $V_{\mathbf{P}}$. This component of the $\underline{\mathbf{C}}$-dynamics is however of no concern, since it is not associated with a change in the orientation of the occupied subspace and therefore not with an actual change in the electronic configuration.

Following [21], the following connecting equations between the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics are used as a starting point to obtain an equation of motion for the $\underline{\mathbf{C}}$-dynamics

$$\mathbf{P} = \underline{\mathbf{C}}\underline{\tilde{\mathbf{C}}}\,; \tag{3.74}$$

$$\dot{\mathbf{P}} = \underline{\mathbf{C}}\underline{\dot{\tilde{\mathbf{C}}}} + \underline{\dot{\mathbf{C}}}\underline{\tilde{\mathbf{C}}}\,; \tag{3.75}$$

$$\ddot{\mathbf{P}} = \underline{\ddot{\mathbf{C}}}\underline{\tilde{\mathbf{C}}} + \underline{\mathbf{C}}\underline{\ddot{\tilde{\mathbf{C}}}} + 2\underline{\dot{\mathbf{C}}}\underline{\dot{\tilde{\mathbf{C}}}}\,. \tag{3.76}$$

To ensure that the only changes allowed in the $\underline{\mathbf{C}}$-matrix are those associated with an actual change in the electronic configuration, the first time derivative, $\underline{\dot{\mathbf{C}}}$, is restricted to be orthogonal to the occupied subspace, $V_{\mathbf{P}}$, in other words

$$\mathbf{P}\underline{\dot{\mathbf{C}}} = \underline{\mathbf{0}}\,, \tag{3.77}$$

or equivalently, $\underline{\dot{\tilde{\mathbf{C}}}}\,\underline{\mathbf{C}} = \underline{\tilde{\mathbf{C}}}\,\underline{\dot{\mathbf{C}}} = \mathbf{0}$.

By taking the first time derivative of $\mathbf{P}\underline{\mathbf{C}} = \underline{\mathbf{C}}$, and using equation (3.77), an expression

for the fictitious velocity is obtained

$$\dot{\mathbf{P}}\underline{\mathbf{C}} = \dot{\underline{\mathbf{C}}} \,. \tag{3.78}$$

Consider now the equation of motion for $\mathbf{P}$ in the form of equation (3.67):

$$\ddot{\mathbf{P}} \;=\; b\,(\mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP}) + 2(\underline{\tilde{\mathbf{A}}}\,\underline{\mathbf{A}} - \underline{\mathbf{A}}\,\underline{\tilde{\mathbf{A}}}). \tag{3.79}$$

Now, the solution for $\dot{\mathbf{P}}$ from equation (3.38) is given by $\dot{\mathbf{P}} = \mathbf{A} + \tilde{\mathbf{A}}$, with $\mathbf{A} = \mathbf{P}\dot{\mathbf{P}}$ an arbitrary element in $\mathcal{M}_2$. Keeping the restriction $\mathbf{P}\dot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$ or $\tilde{\underline{\mathbf{C}}}\,\dot{\underline{\mathbf{C}}} = \mathbf{0}$ in mind, it follows from equation (3.75) that $\mathbf{A} = \underline{\mathbf{C}}\,\tilde{\dot{\underline{\mathbf{C}}}}$. The above equation can thus be written as

$$\ddot{\mathbf{P}} = b\,(\mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP}) + 2(\dot{\underline{\mathbf{C}}}\,\tilde{\dot{\underline{\mathbf{C}}}} - \underline{\mathbf{C}}\,\tilde{\dot{\underline{\mathbf{C}}}}\dot{\underline{\mathbf{C}}}\,\tilde{\underline{\mathbf{C}}}) \,. \tag{3.80}$$

In order to extract an equation of motion for $\underline{\mathbf{C}}$ from the connection between equations (3.76) and (3.80), the projection of the equations onto the space, $V_{\mathbf{P}}^{\perp}$ can be considered. If equation (3.80) is multiplied from the left with $(\mathbf{I}-\mathbf{P})$ and from the right with $\mathbf{C}$ and the restriction $\mathbf{P}\dot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$ or $\tilde{\underline{\mathbf{C}}}\,\dot{\underline{\mathbf{C}}} = \mathbf{0}$, is used, it follows that

$$(\mathbf{I}-\mathbf{P})\ddot{\underline{\mathbf{C}}} = b(\mathbf{I}-\mathbf{P})\mathbf{F}\underline{\mathbf{C}} \,. \tag{3.81}$$

Consider now the component, $\mathbf{P}\ddot{\underline{\mathbf{C}}}$, in the above equation. This component lies in the column space, $V_{\mathbf{P}}$, of $\underline{\mathbf{C}}$, and can thus be written as $\mathbf{P}\ddot{\underline{\mathbf{C}}} = \underline{\mathbf{C}}\mathbf{G}$, with $\mathbf{G} = \mathbf{G}(\underline{\mathbf{C}}, \dot{\underline{\mathbf{C}}}, \mathbf{F})$ a matrix that depends on $\mathbf{C}$ and $\dot{\underline{\mathbf{C}}}$. Now, let

$$\ddot{\underline{\mathbf{C}}} = b(\mathbf{I}-\mathbf{P})\mathbf{F}\underline{\mathbf{C}} + \underline{\mathbf{C}}\mathbf{G} \,. \tag{3.82}$$

By substituting this expression into equation (3.76) and comparing the resulting expression for $\ddot{\mathbf{P}}$ with equation (3.80), it follows that

$$\underline{\mathbf{C}}(\mathbf{G}+\tilde{\mathbf{G}})\tilde{\underline{\mathbf{C}}} + 2\dot{\underline{\mathbf{C}}}\,\tilde{\dot{\underline{\mathbf{C}}}} = 2(\dot{\underline{\mathbf{C}}}\,\tilde{\dot{\underline{\mathbf{C}}}} - \underline{\mathbf{C}}\,\tilde{\dot{\underline{\mathbf{C}}}}\dot{\underline{\mathbf{C}}}\,\tilde{\underline{\mathbf{C}}}) \,.$$

It thus follows from this equation and the restriction $\mathbf{P}\dot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$ or $\tilde{\underline{\mathbf{C}}}\,\dot{\underline{\mathbf{C}}} = \mathbf{0}$, that $\mathbf{G} + \tilde{\mathbf{G}} = -2\tilde{\dot{\underline{\mathbf{C}}}}\,\underline{\mathbf{C}}$. An appropriate choice for $\mathbf{G}$ is thus given by $\mathbf{G} = -\tilde{\dot{\underline{\mathbf{C}}}}\,\underline{\mathbf{C}}$, which is

independent of $\mathbf{F}$. By substituting this expression for $\mathbf{G}$ back into equation (3.82), an equation of motion for $\underline{\mathbf{C}}$ can be obtained:

$$\ddot{\underline{\mathbf{C}}} = b(\mathbf{I} - \mathbf{P})\mathbf{F}\underline{\mathbf{C}} - \underline{\mathbf{C}}\,\tilde{\underline{\mathbf{C}}}\,\dot{\underline{\mathbf{C}}}\,. \tag{3.83}$$

The $\underline{\mathbf{C}}$-dynamics generated by this equation, and the $\mathbf{P}$-dynamics generated by equation (3.66) correspond to the same path for the electronic configuration within accuracy of numerical calculations. Furthermore, it is shown in [21] that the equation of motion for the $\underline{\mathbf{C}}$-dynamics, given by equation (3.83), is indeed compatible with the above mentioned restriction, that is $\tilde{\underline{\mathbf{C}}}\,\dot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$

Equation (3.83) can also be written as

$$\ddot{\underline{\mathbf{C}}} = b\mathbf{F}\underline{\mathbf{C}} - \underline{\mathbf{C}}\boldsymbol{\Lambda}\,, \tag{3.84}$$

with

$$\underline{\mathbf{C}}\boldsymbol{\Lambda} = b\mathbf{P}\mathbf{F}\underline{\mathbf{C}} - \underline{\mathbf{C}}\,\tilde{\underline{\dot{\mathbf{C}}}}\,\dot{\underline{\mathbf{C}}}\,, \tag{3.85}$$

and

$$\boldsymbol{\Lambda} = b\tilde{\underline{\mathbf{C}}}\mathbf{F}\underline{\mathbf{C}} - \tilde{\underline{\dot{\mathbf{C}}}}\,\dot{\underline{\mathbf{C}}}\,. \tag{3.86}$$

where $b < 0$ and Λ the Lagrange multipliers that represent the constraint force.

The first term in equation (3.84) thus corresponds to the applied fictitious force acting on the electronic degrees of freedom. The second term in these equations corresponds to the constraint force induced by the orthonormality constraint.

It was shown above that the equilibrium configuration associated with the $\mathbf{P}$-dynamics equation of motion corresponds to the groundstate electronic configuration. The $\mathbf{P}$-dynamics and $\underline{\mathbf{C}}$-dynamics are physically equivalent, thus this is true for the $\underline{\mathbf{C}}$-dynamics equation of motion as well. In [21] it is explicitly shown that this is indeed the case: $\ddot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$, if and only if $\dot{\underline{\mathbf{C}}} = \underline{\mathbf{0}}$ and

$$\mathbf{F}\underline{\mathbf{C}} = \underline{\mathbf{C}}\boldsymbol{\Lambda}\,,$$

which is within a unitary transformation of the standard eigenvalue problem for the Fock matrix, but only with the occupied molecular orbitals considered. It would be interesting to follow the arguments that led to the above equation "backwards" for the case where orbitals associated with excited states replace the lowest energy orbitals. The associated **P**-dynamics will most probably reveal the excited states as an energy plato as discussed in the next chapter.

Chapter 4

Implementation of the equations of motion for electronic calculations

A successful implementation of the equation of motion for the density matrix (equation (3.64) into the MOPAC package, combined in future with a molecular dynamics algorithm for the atomic dynamics, will have considerable potential of modeling reaction and adsorption dynamics on, for example, silicon surfaces. With this in mind, the MOPAC package was implemented on a Pentium computer equipped with FORTRAN90 and a software interface with the subroutines of MOPAC that calculate the Fock, Hamiltonian and two electron matrices as well as the electronic energy, was provided. An application program interface between FORTRAN and MATLAB was also created in order to extract graphical information.

The implementation of the equations of motion (within the semi-empirical environment as supplied by the MOPAC package) can be divided into the following stages:

1. Integration of the equations of motion using the appropriate integration algorithm;

2. Enforcement of constraints on the projection or configuration matrix to correct for errors introduced by the numerical integration;

3. Parameter optimization to ensure maximum efficiency of the equations of motion.

4.1 Integration of the equations of motion

In section 2.7, molecular dynamics were briefly discussed and the equations of motion for particle dynamics, which are second order differential equations, were explicitly shown. These equations can be solved by means of numerical integration. One such a numerical integration technique is the Verlet algorithm from which a variety of variants were developed, such as the velocity Verlet algorithm.

4.1.1 The Verlet and velocity Verlet algorithm

Perhaps the most widely used method of integrating the equations of motion (section 2.7) is that initially adopted by Verlet.[52] This method is a direct solution of the second-order differential equations. It is based on the positions $\mathbf{r}(t)$, $\mathbf{r}(t+\delta t)$, and the positions $\mathbf{r}(t-\delta t)$ from the previous step. The equation for advancing the positions reads as follows:

$$\mathbf{r}(t+\delta t) = 2\mathbf{r}(t) - \mathbf{r}(t-\delta t) + \delta t^2 \mathbf{a}(t) \; . \tag{4.1}$$

There are several points to note about equation (4.1). It can be seen that the velocities do not appear at all. These have been eliminated by addition of the equations obtained by Taylor expansion about $\mathbf{r}$(t):

$$\mathbf{r}(t+\delta t) = \mathbf{r}(t) + \delta t \mathbf{v}(t) + \tfrac{1}{2}\delta t^2 \mathbf{a}(t) + ...$$
$$\mathbf{r}(t-\delta t) = \mathbf{r}(t) - \delta t \mathbf{v}(t) + \tfrac{1}{2}\delta t^2 \mathbf{a}(t) - .....$$

As it can be immediately seen, the truncation error of the algorithm when evolving the system by δt is of the order δt^4.

The use of the Verlet algorithm as an integrator has several advantages, which are:

1. The advancement of positions occurs all in one step (using equation (4.1));
2. The algorithm is compact and simple to program;
3. The algorithm displays the property of time-reversibility;
4. The algorithm conserves energy well, even with relatively long time steps.

However, the Verlet integrator has some disadvantages which are:

1. The velocities at time t can only be calculated after $\mathbf{r}(t+\delta t)$ are known;
2. To start a trajectory, $\mathbf{r}(t)$ and $\mathbf{r}(t-\delta t)$ must be known rather than $\mathbf{r}(t)$ and $\mathbf{v}(t)$.

A problem with this version of the Verlet algorithm is that the velocities are not directly generated. Although they are not needed for the time evolution, knowledge of them is sometimes necessary. Moreover, they are required to compute the kinetic energy, whose evaluation is necessary to test the conservation of the total energy. This is one of the most important tests to verify that a molecular dynamics simulation is proceeding correctly.

The velocities can be computed from the positions by

$$\mathbf{v}(t) = \frac{\mathbf{r}(t+\delta t) - \mathbf{r}(t-\delta t)}{2\delta t} . \tag{4.2}$$

Whereas equation (4.1) is correct except for errors of order δt^4 (the local error) the velocities from equation (4.2) are subject to errors of order δt^2.

To overcome this difficulty, some variants of the Verlet algorithm have been developed. They give rise to exactly the same trajectory, and only differ in what variables are stored in memory and at what times. A Verlet-equivalent algorithm which stores positions, velocities, and accelerations all at the same time, t, and minimizes round-off error,

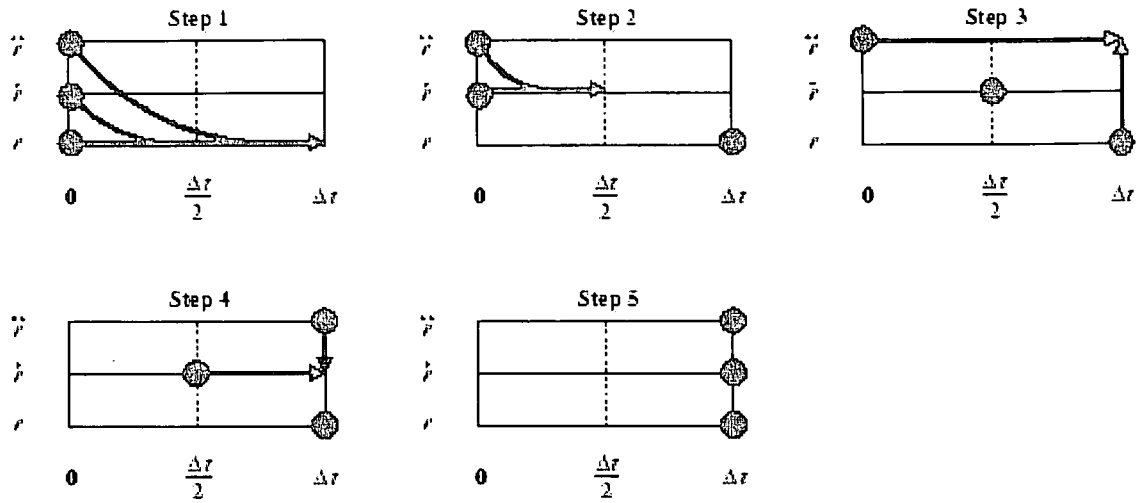

Figure 4.1: A graphical depiction of the velocity Verlet algorithm.[108] *In step 1,* $\mathbf{r}(t_0 + \delta t)$ *is calculated using* $\mathbf{v}(t_0)$ *and* $\mathbf{a}(t_0)$*. In step 2,* $\mathbf{v}(t_0 + \frac{1}{2}\delta t)$ *is calculated using* $\mathbf{a}(t_0)$*. In step 3,* $\mathbf{a}(t_0 + \delta t)$ *is calculated using* $\mathbf{r}(t + \delta t)$*. In step 4,* $\mathbf{v}(t)$ *is calculated using* $\mathbf{a}(t + \delta t)$*.*

has been proposed by Andersen et al.[53] This "velocity Verlet" algorithm takes the form

$$\mathbf{r}(t + \delta t) = \mathbf{r}(t) + \delta t \mathbf{v}(t) + \frac{1}{2}\delta t^2 \mathbf{a}(t) \tag{4.3}$$

$$\mathbf{v}(t + \delta t) = \mathbf{v}(t) + \frac{1}{2}\delta t(\mathbf{a}(t) + \mathbf{a}(t + \delta t)) \tag{4.4}$$

with the implementation thereof illustrated in figure 4.1.

The step for calculating the velocities can also be divided into two: firstly the velocities are calculated at mid-step using equation (4.4)

$$\mathbf{v}(t + \frac{1}{2}\delta t) = \mathbf{v}(t) + \frac{1}{2}\delta t \mathbf{a}(t) \ , \tag{4.5}$$

and secondly the forces and accelerations at time $t + \delta t$ are then computed to complete the velocity move

$$\mathbf{v}(t + \delta t) = \mathbf{v}(t + \frac{1}{2}\delta t) + \frac{1}{2}\delta t \mathbf{a}(t + \delta t) \ . \tag{4.6}$$

Use of this integrator has several advantages which are:

1. The kinetic energy at time $t + \delta t$ is available;
2. The calculation can start with positions and velocities at time t (i.e $\mathbf{r}(t)$ and $\mathbf{v}(t)$);
3. It is numerically stable and simple to program;
4. It has the property of time-reversibility;
5. It conserves energy well even with relatively long time steps.

Still, the velocity Verlet scheme has the disadvantage that the advancement of velocities takes two steps, thereby increasing computational demand.

Several other algorithms for numerical integration exist, for example Gear-Predictor Corrector, leapfrog and Beeman's algorithm.[55] In light of the advantages of the velocity Verlet algorithm as mentioned above, together with its widespread use, led to the selection of this algorithm as integrator for fictitious dynamics' equations of motion in this study.

4.1.2 Integration of the equations of motion through the use of the velocity Verlet algorithm

Although fictitious and not atomic dynamics are dealt with, the velocity Verlet scheme can be applied to solve any second order differential equation, such as the equations of motion for the **P**-dynamics and $\underline{\mathbf{C}}$-dynamics.

The equation of motion for the **P**-dynamics is given by equation (3.64):

$$\ddot{\mathbf{P}} = (\mathbf{I} - 2\mathbf{P})\left(b[\mathbf{F}, \mathbf{P}] + 2\dot{\mathbf{P}}^2\right) , \tag{4.7}$$

where $\ddot{\mathbf{P}}$ can be interpreted as the fictitious acceleration.

The equation of motion for the $\underline{\mathbf{C}}$-dynamics is given by equation (3.83):

$$\ddot{\underline{\mathbf{C}}} = b(\mathbf{I} - \mathbf{P})\mathbf{F}\underline{\mathbf{C}} - \underline{\mathbf{C}}\,\tilde{\dot{\underline{\mathbf{C}}}}\,\dot{\underline{\mathbf{C}}}\,, \tag{4.8}$$

where $\ddot{\underline{\mathbf{C}}}$ can be interpreted as the fictitious acceleration.

Given now the expressions for the advancement of position (equation (4.4)) and velocity (equations (4.5) and (4.6)), expressions for the advancement of fictitious position and velocity can be written respectively.

Application of the velocity Verlet algorithm to the fictitious position leads to the expression

$$\mathbf{P}(t + \delta t) = \mathbf{P}(t) + \delta t\dot{\mathbf{P}}(t) + \frac{1}{2}\delta t^2\ddot{\mathbf{P}}(t)\,, \tag{4.9}$$

with $\ddot{\mathbf{P}}$ given by equation (4.7).

Application of the velocity Verlet algorithm to the fictitious velocity leads to the expressions

$$\dot{\mathbf{P}}(t + \frac{1}{2}\delta t) = \dot{\mathbf{P}}(t) + \frac{1}{2}\delta t\ddot{\mathbf{P}}(t) \tag{4.10}$$

and

$$\dot{\mathbf{P}}(t + \delta t) = \dot{\mathbf{P}}(t) + \frac{1}{2}\delta t\ddot{\mathbf{P}}(t + \delta t)\,, \tag{4.11}$$

with $\ddot{\mathbf{P}}$ given by equation (4.7).

However, it can be seen from equation (4.11) that, in order to calculate the fictitious velocity at a full time step, the fictitious acceleration at a full time step must be known, but from equation (4.7) the fictitious velocity is required to calculate the fictitious acceleration. In order to solve this empasse, the fictitious velocity is calculated at only mid-step. The neglect of the fictitious acceleration at a full time step does not have an effect on whether energy convergence is reached - it can be speculated that it only

influences the "form" of the energy trajectory. It can also influence the conservation of energy due to the neglect of a term of $O(\delta t)$. More attention to this matter will be given in section 4.5.

In order to solve a differential equation, some starting configurations are necessary. Thus, initial values for $\dot{\mathbf{P}}$ and $\mathbf{P}$, consistent with the constraints, are needed in order to start with a calculation. The general form of $\dot{\mathbf{P}}$ is $\dot{\mathbf{P}} = \mathbf{A} + \tilde{\mathbf{A}}$, with $\mathbf{A}$ any matrix in $\mathcal{M}_2$ as given by equation (3.38). As an initial value for $\mathbf{P}$ an $N \times N$ matrix with $N/2$ one's on the diagonal is chosen in order to conserve the number of electrons as mentioned in section 3.1.1. Remember that N is the number of electrons in the system and $M = N/2$ the number of filled, closed orbitals. It turns out that the initial value of $\dot{\mathbf{P}}$ can be generated randomly from equation (3.38) and should only be appropriately scaled so as not to introduce a too high fictitious kinetic energy in the system under study.

Similar expressions can be written for the $\underline{\mathbf{C}}$-dynamics. Application of the velocity Verlet algorithm to the fictitious position, leads to the expression

$$\underline{\mathbf{C}}(t+\delta t) = \underline{\mathbf{C}}(t) + \delta t \dot{\underline{\mathbf{C}}}(t) + \frac{1}{2}\delta t^2 \ddot{\underline{\mathbf{C}}}(t) \; . \tag{4.12}$$

with $\ddot{\underline{\mathbf{C}}}$ given by equation (4.8).

Application of the velocity Verlet algorithm to the fictitious velocity leads to the expressions

$$\dot{\underline{\mathbf{C}}}(t+\frac{1}{2}\delta t) = \dot{\underline{\mathbf{C}}}(t) + \frac{1}{2}\delta t \ddot{\underline{\mathbf{C}}}(t) \tag{4.13}$$

and

$$\dot{\underline{\mathbf{C}}}(t+\delta t) = \dot{\underline{\mathbf{C}}}(t) + \frac{1}{2}\delta t \ddot{\underline{\mathbf{C}}}(t+\delta t) \; , \tag{4.14}$$

with $\ddot{\underline{\mathbf{C}}}$ given by equation (4.8).

In order to start a fictitious dynamics run for the $\underline{\mathbf{C}}$-dynamics, starting configurations for

$\underline{\mathbf{C}}$ and $\dot{\underline{\mathbf{C}}}$, consistent with the constraints, are also needed. In fact, $\dot{\underline{\mathbf{C}}}$ should be orthogonal to the space V_C, that is $\mathbf{P}\dot{\underline{\mathbf{C}}} = 0$. An $N \times M$ matrix with $N/2$ one's on the diagonal can be chosen for $\underline{\mathbf{C}}$ to ensure that its columns are orthonormal. It turns out that the initial value for $\dot{\underline{\mathbf{C}}}$ can be a randomly selected $N \times M$ matrix which should be appropriately scaled to ensure that a not too high fictitious kinetic energy is introduced into the system. The action of RATTLE as discussed below soon gets $\underline{\mathbf{C}}$ and $\dot{\underline{\mathbf{C}}}$ in line with the constraints.

The initial values for $\dot{\mathbf{P}}$, $\mathbf{P}$, $\underline{\mathbf{C}}$ and $\dot{\underline{\mathbf{C}}}$ as supplied above are not necessarily the most effective choices regarding the time necessary for energy convergence. In section 4.4 the influence of the initial values on the resulting energy convergence, is discussed in more detail.

4.2 Enforcement of constraints on the equations of motion

For a given model which characterizes the physical system, it is the integrator which is responsible for the accuracy of the simulation results. If the integrator could work without any error, the simulation would provide *exact model results* within the errors occuring due to finite number representation. However, any finite difference integrator is naturally an approximation for a system developing continuously in time.

It has been emphasized before that the equations of motion conserve idempotency and orthonormality of the projection and configuration matrix respectively. However, the existence of the errors introduced by the integrator will cause deviation from idempotency and orthonormality. It is therefore necessary to treat the idempotency and orthonormality as constraints that need to be imposed on the equations of motion. In the following two sections two algorithms are discussed that can be used to ensure that the above constraints are satisfied.

4.2.1 Forces of constraint in the P-dynamics equation of motion

The requirement that the density matrix should be a projector i.e. it is the operator which projects onto the space of occupied states, necessitates the enforcement of idempotency or equivalently the requirement that the eigenvalues of the density matrix be all either one or zero.

In section 3.1.1 it was mentioned that one of the properties of the projection matrix is that $\mathrm{tr}(\mathbf{P}) = \frac{N}{2}$, with N the number of electrons in the system. Thus, for the density matrix, where $\mathbf{P}' = 2\mathbf{P}$ for spinless particles, $\mathrm{tr}(\mathbf{P}') = \mathrm{rank}(\mathbf{P}') = N$. Equivalently, this can be expressed as

$$\begin{aligned} N &= tr(\mathbf{P}') \\ &= \sum_n f_n(\epsilon_n) \,, \end{aligned} \tag{4.15}$$

where ϵ_n denotes the eigenvalues of the density matrix and f_n the Fermi-distribution function:

$$f(\epsilon) = \frac{1}{1 + e^{\frac{\epsilon - \mu}{k_B T}}} \,, \tag{4.16}$$

with k_B the Boltzmann constant, μ the chemical potential and T the electronic temperature.

In the coordinate space representation the one-particle density matrix can be written as[38]

$$\mathbf{P}'(\mathbf{r}, \mathbf{r}') = \sum_n f_n(\epsilon_n) \psi_n{}^*(\mathbf{r}) \psi_n(\mathbf{r}') \,, \tag{4.17}$$

where n runs over all the eigenstates of the Hamiltonian. From the functional form of the Fermi distribution function it thus follows that the eigenvalues of $f(\epsilon_n)$ are always in the interval [0,1]. At $T = 0$ the density matrix of an insulating system containing N electrons will have N eigenvalues of one, all the others being zero. Thus, the density

matrix does not have full rank, but only rank N. Hence

$$\mathbf{P}'(\mathbf{r},\mathbf{r}') = \sum_{n}^{occ} \psi_n^*(\mathbf{r})\psi_n(\mathbf{r}') \,, \tag{4.18}$$

where the summation runs only over the occupied states.

The eigenvalues of the density matrix can therefore be interpreted as the occupancy of the particular eigenfunction. Thus, the enforcement of idempotency ensures that the occupation numbers, f_n, are either zero or one so that the correct zero-temperature ground state is reached, namely the state where all the orbitals below the Fermi level have $f_n = 1$ and all those above have $f_n = 0$. If this idempotency constraint was not enforced, then the energy of the system could be lowered by overfilling the lowest energy orbitals, whilst still satisfying electron number conservation. This is however not a physical situation since, according to the variational theorem,[46] no energy can be lower than the ground state energy.

The expression for the **P**-dynamics equation of motion, given by equation (3.67) can also be considered:

$$\ddot{\mathbf{P}} = b[(\mathbf{PF}(\mathbf{I}-\mathbf{P}) + (\mathbf{I}-\mathbf{P})\mathbf{FP})] + 2(\tilde{\mathbf{A}}\mathbf{A} - \mathbf{A}\tilde{\mathbf{A}}) \,,$$

with $\mathbf{A} = \mathbf{P}\dot{\mathbf{P}}$. The term in square brackets in the above equation can be interpreted as a necessary constraint to project the Fock matrix, **F**, onto the first order idempotency conserving subspace, $\mathcal{M}_{(2,3)}$. Thus, components outside of the idempotency conserving subspace are subtracted so that only those inside the idempotency conserving subspace are left. The last term can be interpreted as the constraints induced by the fictitious velocities.

In the next section an algorithm, which can be used to enforce idempotency on the density/projection matrix, thus correcting for numerical errors in the application of equation (3.67), is reviewed.

4.2.1.1 McWeeny purification

The McWeeny purification scheme[56] provides a way of achieving idempotency of the projection matrix without diagonalisation of the Hamiltonian matrix. (The diagonalisation of the Hamiltonian matrix is used in SCF calculations and is extremely time consuming). The purification algorithm for mapping a nearly idempotent matrix $\tilde{\rho}$ into one that is more idempotent is given by:

$$\rho = 3\tilde{\rho}^2 - 2\tilde{\rho}^3 \; . \tag{4.19}$$

The way this mapping works can be understood by considering the eigenvalues of ρ and $\tilde{\rho}$, denoted by λ and $\tilde{\lambda}$ respectively. Since ρ is diagonal in any representation that diagonalises $\tilde{\rho}$, the relationship between their eigenvalues is

$$\lambda = 3\tilde{\lambda}^2 - 2\tilde{\lambda}^3 \; . \tag{4.20}$$

From the form of the function $f(\tilde{\lambda}) = 3\tilde{\lambda}^2 - 2\tilde{\lambda}^3$ (see figure 4.2), it follows that if $-\frac{1}{2} < \tilde{\lambda} < \frac{3}{2}$ then $0 \leq \lambda \leq 1$. Furthermore, if $0 < \tilde{\lambda} < 1$, then, for $\tilde{\lambda} < \frac{1}{2}$, $\lambda < \tilde{\lambda}$ and for $\tilde{\lambda} > \frac{1}{2}$, $\lambda > \tilde{\lambda}$, so that iteration of the mapping drives the eigenvalues towards 0 or 1. As this process continues iteratively, the final approach to idempotency accelerates rapidly. If $\tilde{\lambda}$ deviates from zero by a small amount, then the deviation of λ is proportional to the square of that amount, and similarly for deviations from unity. This quadratic convergence to idempotency can be summarised by noting that the matrices $\rho^2 - \rho$ and $\tilde{\rho}^2 - \tilde{\rho}$ are related by:

$$\rho^2 - \rho = 4(\tilde{\rho}^2 - \tilde{\rho})^3 - 3(\tilde{\rho}^2 - \tilde{\rho})^2 \; . \tag{4.21}$$

For a given initial matrix ρ^0, iteration of the purification algorithm therefore generates a sequence ρ^1, ρ^2, ... that converges to an idempotent matrix ρ^∞. It has been emphasised recently by Palser and Manolopoulos[51] that McWeeny purification, combined with the appropriate equation of motion, automatically delivers the correct ground state after a sufficient amount of iterations. The implementation of the purification algorithm (in terms of programming) is discussed in more detail in Appendix C.

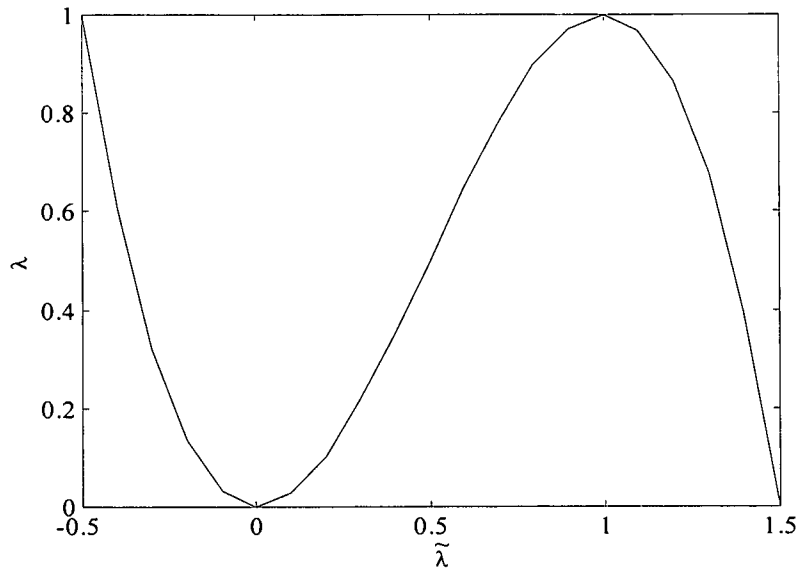

Figure 4.2: Illustration of the function $\lambda = 3\tilde{\lambda}^2 - 2\tilde{\lambda}^3$. *To produce a matrix that is more idempotent than the one that is to be purified, the latter must have eigenvalues in the range between* $-\frac{1}{2}$ *and* $\frac{3}{2}$. *Instability occurs outside this range.*

4.2.2 Forces of constraint in the $\underline{\mathbf{C}}$-dynamics equation of motion

In section 3.7 it was shown that, within the Roothaan treatment, the applied fictitious force on the electronic configuration, $\underline{\mathbf{C}}$, is given by $\underline{\mathbf{F_a}} = b\mathbf{F}\underline{\mathbf{C}}$. The constraint force is given by $\underline{\mathbf{F_c}} = \underline{\mathbf{C}}\mathbf{\Lambda}$, with $\underline{\mathbf{C}}\mathbf{\Lambda}$ given by equation (3.85). As could be expected, the columns of the constraint force matrix lie in the columnspace V_C of $\underline{\mathbf{C}}$. It is obvious that the term $\underline{\mathbf{C}}\,\underline{\tilde{\mathbf{C}}}\,\underline{\dot{\mathbf{C}}}$ represents the constraint forces induced by the velocities contained in $\underline{\dot{\mathbf{C}}}$. This term is present even when the columns of $\underline{\dot{\mathbf{C}}}$ are orthogonal to the columnspace, V_C.

The origin of this term can easily be understood in terms of an anology of a plane rotation in three dimensions. Consider two orthonormal vectors, $\mathbf{r_1}$ and $\mathbf{r_2}$, lying in the xy-plane at a given moment with velocities orthogonal to the xy-plane. The vector,

$\mathbf{r_1}$, will rotate in the ($\mathbf{r_1}$-z-plane) towards the z-axis, and the vector, $\mathbf{r_2}$, will rotate (in the $\mathbf{r_2}$-z-plane) towards the z-axis. The angle between these two vectors will decrease except if some constraint force counteract this effect.

The constraint force term, $-b\mathbf{PF}\underline{\mathbf{C}}$ can also be considered. This constraint force cancels the component of the applied force in the columnspace, V_C, leaving only the contribution orthogonal to V_C as is evident from equation (3.84). Thus, the only term of the netto force acting on the electronic configuration that lies in the columnspace, V_C, is the one induced by the velocities.

In the next section an algorithm that can be used to enforce orthonormality on the configuration matrix is reviewed.

4.2.2.1 RATTLE algorithm

With the aim of imposing the orthonormality constraint on the configuration matrix, the constraint dynamics in molecular dynamics are first considered.

In molecular dynamics simulations the integration time step must be sufficiently small in order to handle the fastest motion in a system accurately. Typically, a time step of 1 femtosecond is necessary to handle bond stretches involving hydrogen atoms.[55] A popular approach for increasing time steps is to fix the fastest degrees of freedom (bond stretches and angles) and to solve the equations of motion for the slower dihedral degrees of freedom. Such an approach is especially justified for studies of large biological molecules, where bond lengths and angles vary little from one structure to another and nearly all important conformational transitions are due to dihedral angle motions.

A special technique has been developed to handle the dynamics of a molecular system in which certain arbitrarily selected degrees of freedom (such as bond lengths) are constrained, while others remain free to evolve under the influence of intermolecular and intramolecular forces. This constraint dynamics approach which was termed SHAKE[57] in effect uses a set of undetermined multipliers to represent the magnitudes

of forces directed along the bonds, which are required to keep the bond lengths constant. The technique is to solve the equations of motion for one time step in the absence of the constraint forces, and subsequently determine their magnitudes and correct the atomic positions. The method can be applied equally well to totally rigid and non-rigid molecules. Its great appeal is that it reduces even a complex polyatomic liquid simulation to the level of diffculty of an atomic calculation plus a constraint package based on molecular geometry. The SHAKE algorithm works well for time steps up to 5 fs,[55] thereby enabling a five-fold speedup in computational time as long as the process of iteratively solving the constraint equations does not consume too much time.

To avoid the computational drawbacks of the basic Verlet scheme, Andersen[58] used instead the velocity Verlet algorithm to impose bond-stretch constraints, and termed the resulting algorithm RATTLE. Holonomic constraints can be imposed on the RATTLE algorithm. The possibility of imposing holonomic constraints in molecular dynamics simulations provides the ability to selectively freeze particular degrees of freedom, without having to interfere with others.

An introduction into the RATTLE algorithm for general holonomic constraints are given in the following discussion. This algorithm is discussed as if it is being applied to atomic posistions. However, by a mere change of notation, this algorithm can equally well be applied to the electronic degrees of freedom subject to the orthonormality constraints.

The equations for the advancement of position and velocity as derived in section 4.1.1 can be written in the following form:

$$\mathbf{r}_i'(t_0+\delta t) = \mathbf{r}_i(t_0) + [\delta t]\,\dot{\mathbf{r}}_i(t_0) + \frac{[\delta t]^2}{2m_i}\mathbf{F}_i(t_0)\ , \tag{4.22}$$

$$\dot{\mathbf{r}}_i'(t_0+\delta t) = \dot{\mathbf{r}}_i(t_0) + \frac{[\delta t]}{2m}\left\{\mathbf{F}_i(t_0) + \mathbf{F}_i(t_0+\delta t)\right\} \tag{4.23}$$

where the acceleration, $\mathbf{a_i}$, has been replaced by the force on each particle divided by its mass: $\frac{\mathbf{F}_i}{m_i}$.

Consider now a system of N interacting particles subject to l general holonomic constraints

$$\sigma_k\left(\left\{\mathbf{r}_{i(t)}\right\}\right) = 0\,, \quad (k = 1, ..., l) \tag{4.24}$$

where $\mathbf{r}_i(t)$ denotes the coordinates of the subset of n_k particles involved in σ_k. The constrained coordinates are given by[57],[60],[61]

$$\mathbf{r}_i\left(t_0 + \delta t, \{\gamma\}\right) = \mathbf{r'}_i\left(t_0 + \delta t\right) - \frac{[\delta t]^2}{2m_i}\sum_{k=1}^{l}\gamma_k\left[\nabla_i\sigma_k\right](t_0)\,. \tag{4.25}$$

The unconstrained coordinates $\mathbf{r}'_i\left(t_0 + \delta t\right)$ are given by equation (4.22). The parameters $\{\gamma_k\}$ are chosen such that the constrained coordinates at time $(t_0 + \delta t)$ satisfy the constraint equations (within a desired tolerance), and either the "matrix method" or the SHAKE procedure can be used[57],[60],[61] to obtain $\{\gamma_k\}$. The constrained velocities are given by[61]

$$\dot{\mathbf{r}}_i\left(t_0 + \delta t, \{\eta_k\}\right) = \dot{\mathbf{r}}'_i\left(t_0 + \delta t\right) - \frac{[\delta t]}{2m_i}\sum_{k=1}^{l}\eta_k\left[\nabla_i\sigma_k\right](t_0 + \delta t)\,. \tag{4.26}$$

where the unconstrained velocities are given by equation (4.23). The parameters $\{\eta_k\}$ are chosen such that the constrained velocities at time $(t_0 + \delta t)$ satisfy the constraint equations, more specifically their time derivatives. Accordingly, differentiating equation (4.24) with respect to time we get

$$\frac{d}{dt}\sigma_k\left(\{\mathbf{r}_i\left(t_0 + \delta t\right)\}\right) = \sum_{i=1}^{n_k}\dot{\mathbf{r}}_i\left(t_0 + \delta t\right)\cdot\left[\nabla_i\sigma_k\right]\left(\{\mathbf{r}_i\left(t_0 + \delta t\right)\}\right) = 0\,, \quad k = 1, ..., l \tag{4.27}$$

where $\dot{\mathbf{r}}_i\left(t_0 + \delta t\right)$ is inserted from equation (4.26). Again, either numerical matrix inversion or the SHAKE procedure can be used to solve the set of l linear equations (4.27) for the $\{\eta_k\}$. Since the solution for the $\{\gamma_k\}$ and $\{\eta_k\}$ by the matrix techniques becomes computationally expensive for systems with large numbers of coupled constraints, the emphasis will be on the solution by the SHAKE procedure, namely RATTLE.

The first of the two stages of the RATTLE formulation of general holonomic constraints, described here, is identical to the generalized SHAKE scheme.[60],[61] The SHAKE algorithm consists of an iterative loop inside which the constraints are considered individually and successively. During an iteration, the algorithm successively

selects every constraint and corrects the positions of the subset of particles involved in that constraint, to satisfy it.

Consider now a certain iteration and a particular constraint, σ_k. Let $\{\mathbf{r}_i^{old}(t_0+\delta t)\}$ be a subset of n_k particle positions involved in σ_k, with values including all changes up to this point in the iteration. The new positions for the particles obtained $\{\mathbf{r}_i^{new}(t_0+\delta t)\}$ in the current iteration are computed as

$$\mathbf{r}_i^{new}(t_0+\delta t) = \mathbf{r}_i^{old}(t_0+\delta t) - \frac{[\delta t]^2}{2m_i}\sum_{k=1}^{l}\gamma_k^{new}\left[\nabla_i\sigma_k\right](t_0)\ , \quad (i=1,...,n_k) \tag{4.28}$$

where the starting value of $\mathbf{r}_i^{old}(t_0+\delta t)$ is given by equation (4.22). These new positions should satisfy the constraint equation for σ_k, leading to

$$\sigma_k\left(\{\mathbf{r}_i^{new}(t_0+\delta t)\}\right) = \sigma_k\left(\left\{\mathbf{r}_i^{old}(t_0+\delta t)\right\} - \left\{\frac{[\delta t]^2}{2m}\gamma_k^{new}\left[\nabla\sigma_k\right](t_0)\right\}\right) = 0\ . \tag{4.29}$$

Equation (4.29) is usually nonlinear in γ_k^{new}, even for a bond-stretch constraint. Taylor expanding $\sigma_k\left(\{\mathbf{r}_i^{new}(t_0+\delta t)\}\right)$ about $\{\mathrm{r}_i^{old}(t_0+\delta t)$, equation (4.29) becomes

$$\begin{aligned}&\sigma_k\left(\left\{\mathbf{r}_i^{old}(t_0+\delta t)\right\} - \left\{\tfrac{[\delta t]^2}{2m}\sum_{k=1}^{l}\gamma_k^{new}\left[\nabla\sigma_k\right](t_0)\right\}\right) = \\ &\sigma_k\left(\left\{\mathbf{r}_i^{old}(t_0+\delta t)\right\}\right) - \sum_{i=1}^{n_k}\tfrac{[\delta t]^2}{2m_i}\gamma_k^{new}\left[\nabla_i\sigma_k\right](t_0)\cdot\left[\nabla_i\sigma_k\right]\left(\left\{\mathrm{r}_i^{old}(t_0+\delta t)\right\}\right) + ... = 0\ ,\end{aligned} \tag{4.30}$$

where the nonlinear terms are not shown explicitly. For computational efficiency, all terms higher than first order in equation (4.30) are usually neglected, the iterative process over constraints ensuring the resulting solution satisfies equation (4.30). From equation (4.30) it follows that

$$\gamma_k^{new} = [\delta t]^{-2}\frac{\sigma_k\left(\left\{\mathbf{r}_i^{old}(t_0+\delta t)\right\}\right)}{\sum_{i=1}^{n_k}(1/2m_i)\left[\nabla_i\sigma_k\right](t_0)\cdot\left[\nabla_i\sigma_k\right]\left(\left\{\mathbf{r}_i^{old}(t_0+\delta t)\right\}\right)}\ . \tag{4.31}$$

Exactly as with SHAKE, iterations over the general holonomic constraints continue until all are satisfied, within some tolerance. With all the constraints satisfied and the constrained coordinates at $(t_0+\delta t)$ available, the potential energy forces $\mathbf{F}_i(t_0+\delta t)$ are computed for use in the second stage of the generalized RATTLE.

During an iteration of this second stage, the algorithm again successively selects every constraint and corrects the velocities of the subset of particles involved in that constraint, to satisfy its time derivative. Considering again some iteration and a particular constraint σ_k, the new velocities of the particles $\dot{\mathbf{r}}_i^{new}(t_0+\delta t)$ obtained in the current iteration are given by

$$\dot{\mathbf{r}}_i^{new}(t_0+\delta t) = \dot{\mathbf{r}}_i^{old}(t_0+\delta t) - \frac{[\delta t]}{2m}\eta_k^{new}\left[\nabla_i\sigma_k\right](t_0+\delta t)\ , \quad (i=1,...,n_k) \tag{4.32}$$

where the starting value of $\dot{\mathbf{r}}_i^{old}(t_0+\delta t)$ is given by equation (4.23). These new velocities should satisfy the time derivative of the constraint equition. Accordingly, inserting equation (4.32) into equation (4.27) and solving the resulting linear equation for η_k^{new} gives

$$\eta_k^{new} = [\delta t]^{-1}\frac{\sum_{i=1}^{n_k}\dot{\mathbf{r}}_i^{old}(t_0+\delta t)\cdot\left[\nabla_i\sigma_k\right](\{\mathbf{r}_i(t_0+\delta t)\})}{\sum_{i=1}^{n_k}(1/2m_i)\left[\nabla_i\sigma_k\right](\{\mathbf{r}_i(t_0+\delta t)\})\cdot\left[\nabla_i\sigma_k\right](\{\mathbf{r}_i(t_0+\delta t)\})}\ . \tag{4.33}$$

As with the first stage of RATTLE, iteration over constraints continues until all the constraints on the velocities have been satisfied within a selected tolerance. The entire RATTLE procedure is then repeated at the next molecular dynamics step.

The RATTLE procedure can now be used to constrain the columns of the configuration matrix to be orthonomal. These orthonormality constraints have their origin in the fact that, in the derivation of a manageable expression for the electronic energy, orthonormality of the molecular orbitals is assumed. However, in principle, the molecular orbitals need not be orthonormal.

The requirement for the columns of the configuration matrix to be orthonormal can be expressed as

$$\underline{\tilde{\mathbf{C}}}\,\underline{\mathbf{C}} = \mathbf{I}\,, \tag{4.34}$$

with $\underline{\mathbf{C}}$ a NxM matrix and $\mathbf{I}$ the MxM identity matrix with $N = 2M$. As mentioned before, N is the number of electrons and M the number of filled, closed orbitals in the system.

Equation (4.34) can be rewritten as

$$\left(\sum_{k=1}^{N} c_{ki}c_{kj}\right) - \delta_{ij} = 0 \,. \tag{4.35}$$

Now, let

$$\sigma_{ij}\left(\{c_{mn}\}\right) = \left(\sum_{k=1}^{N} c_{ki}c_{kj}\right) - \delta_{ij} \,. \tag{4.36}$$

The following orthonormality constraints (which are holonomic constraints) are thus available:

$$\sigma_{ij}\left(\{c_{mn}\}\right) = 0 \,. \quad i = 1, ..., N \quad j = 1, ..., i \tag{4.37}$$

The number of constraint equations, l, are the number of elements in the lower or upper triangle of a MxM matrix, including the diagonal, thus

$$l = \frac{M^2}{2} + \frac{M}{2} = \frac{1}{2}M(M+1). \tag{4.38}$$

From the derivation of the RATTLE algorithm above, it can be seen that another variable needed is $\{\nabla\sigma_k\}$. In other words

$$\begin{aligned}
\frac{\partial\sigma_{ij}}{\partial c_{mn}} &= \frac{\partial}{\partial c_{mn}}\left(\sum_{k=1}^{N} c_{ki}c_{kj} - \delta_{ij}\right) \\
&= \frac{\partial}{\partial c_{mn}}\left(\sum_{k=1}^{N} c_{ki}c_{kj}\right) \\
&= \frac{\partial}{\partial c_{mn}}\left(c_{mi}c_{mj}\right) \\
&= c_{mj}\frac{\partial}{\partial c_{mn}}\left(c_{mi}\right) + c_{mi}\frac{\partial}{\partial c_{mn}}\left(c_{mj}\right) \\
&= c_{mj}\delta_{ni} + c_{mi}\delta_{nj} \,.
\end{aligned} \tag{4.39}$$

The above expressions, equations (4.36) and (4.39), are sufficient to ensure orthonormality of the configuration matrix. The implementation of the RATTLE algorithm (in terms of programming) is explained further in Appendix C.

4.3 Role of parameters

The appropriate choice for the parameters in the equations of motion are imperative as this will permit the efficient operation of these equations. Therefore, optimization of the parameters forms an integral part in the implementation process.

Recall the expression for the equation of motion for the **P**-dynamics from equation (4.9). Combining this with equation (4.7) gives

$$\mathbf{P}(t+\delta t) = \mathbf{P}(t) + \delta t\dot{\mathbf{P}}(t) + \frac{1}{2}\delta t^2(\mathbf{I} - 2\mathbf{P}(t))(b[\mathbf{F}(t), \mathbf{P}(t)] - 2\dot{\mathbf{P}}(t)^2) \; . \tag{4.40}$$

Similarly, combining the equation of motion for the $\underline{\mathbf{C}}$-dynamics (equation (4.12)) with equation (4.8) gives

$$\underline{\mathbf{C}}(t+\delta t) = \underline{\mathbf{C}}(t) + \delta t\dot{\underline{\mathbf{C}}}(t) + \frac{1}{2}\delta t^2 b[(\mathbf{I} - \mathbf{P}(t))\mathbf{F}(t)\underline{\mathbf{C}}(t) - \underline{\mathbf{C}}(t)\,\tilde{\underline{\mathbf{C}}}(t)\dot{\underline{\mathbf{C}}}(t)] \; . \tag{4.41}$$

The parameters that occur in equations (4.40) and (4.41) are the time step, δt, as well as b, which is related to the fictitious mass through the expression: $b = \frac{1}{\mu}$. These two parameters also define the fictitious mass parameter, $b' = \frac{1}{2}b\delta t^2$, which is related to the energy change, ΔE, given by equation (3.69). As yet, these values are unknown and only intuitive guesses can be made. Firstly, the time step in molecular dynamics can be considered, which is limited by the fastest vibration in a system and is typically in the order of femtoseconds.[55] However, fictitious and not atomic dynamics are dealt with, thus the time step should not necessarily be of this order. Secondly, the fictitious mass used in ab initio molecular dynamics can be considered, which are typically 0.1 amu bohr^2 or 182 atomic units.[35] However, it does not necessarily mean that the same applies to the fictitious dynamics method in this study. The last two parameters that are of concern are the degree to which idempotency of the projection matrix and orthonormality of the configuration matrix are enforced.

The role of the parameters is investigated by making use of a model system: a regular

hexagon of hydrogen atoms in which the H-H distance is 0.98314 Å. This system is of course not stable; the only reason this system was used is because it is a simple to study, with parameters that are readily available.[10] The optimized geometry of this system was obtained by defining all bond lengths to be equal, constraining all bond angles to be 120 degrees and defining the system as being planar. The Hamiltonian used during the simulation was approximated using the MNDO[10] method, as supplied by the MOPAC package. In the rest of this chapter, all results obtained will refer to this particular system, except when mentioned otherwise. The different parameters for both the **P**-dynamics and $\underline{\mathbf{C}}$-dynamics are discussed in the following sections.

4.3.1 Idempotency tolerance

The idempotency requirement for the projection/density matrix is given by $\mathbf{P^2}-\mathbf{P}=\mathbf{0}$, as mentioned in section 3.1.1. However, the extent to which this requirement is enforced has a significant influence on the resultant energy convergence.

The idempotency tolerance can be defined as the Euclidean norm of $\mathbf{P^2}-\mathbf{P}$:

$$I_{tol}=\left(\sum_{i=1}^{N}\left((\mathbf{P}^2-\mathbf{P})^T(\mathbf{P^2}-\mathbf{P})\right)_{ii}\right)^{\frac{1}{2}}. \tag{4.42}$$

To investigate the effect of the idempotency tolerance on the energy convergence, values for b and δt are needed, since these are required during integration of equation (4.9). Guesses for these values can be made by considering the change in the projection matrix, $\Delta\mathbf{P}$ that conserves idempotency (equation (3.68)). If it is now assumed (which is entirely plausible) that the projection matrix at time t are idempotent, say about 10^{-4} (this results in a matrix $(\mathbf{P^2}-\mathbf{P})$ with elements very close to zero), then the change in the projection matrix at time $t+\delta t$ should be of the same order to ensure that $\mathbf{P}+\Delta\mathbf{P}$ is also idempotent. Since $\Delta\mathbf{P}$ is proportional to the fictitious mass parameter, $b'=\frac{1}{2}b\delta t^2$, this combination should be more or less 10^{-4}. A time step of 10^{-1} and

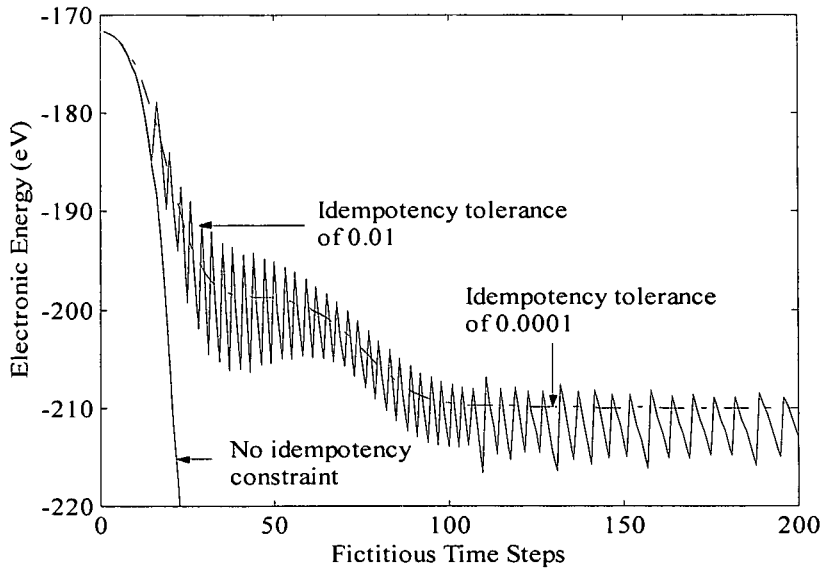

Figure 4.3: Illustration of the effect that the enforcement of idempotency has on the energy convergence. The influence of the McWeeny purification (to ensure idempotency) can be clearly seen.

a value for b of 10^{-2} can thus be chosen. However, the fictitious mass parameter has not been optimized yet and here it only serves as an aid in determining the effect the idempotency tolerance has on the energy convergence, as illustrated in figure 4.3.

From figure 4.3 it can be seen that, if no idempotency is enforced on the projection matrix, there is a "run-away" decrease in the electronic energy. This correlates with the discussion in section 4.2.1 on how the energy of the system is lowered by overfilling the lower energy levels. It is also evident from figure 4.3 that, for a too large idempotency tolerance, the resultant minimum energy is still lower than the true ground state, which is of course unphysical due to the variational theorem.[46] For a strong enough idempotency constraint, the ground state energy obtained with the equations of motion agrees with the ground state energy of -210.085 eV calculated with the MNDO SCF method.[10]

4.3.2 Orthonormality tolerance

The requirement that the columns of the configuration matrix must be orthonormal is given by $\underline{\tilde{\mathbf{C}}}\ \underline{\mathbf{C}} = \mathbf{I}$ for real $\underline{\mathbf{C}}$, as mentioned in section 3.1.1. As was the case for the **P**-dynamics, the extent to which this constraint is enforced will have a significant effect on the energy convergence.

The orthonormality tolerance is defined as the Euclidian norm of $\underline{\mathbf{C}}\ \underline{\tilde{\mathbf{C}}} - \mathbf{I}$:

$$O_{tol} = \left(\sum_{i=1}^{N} \left((\underline{\mathbf{C}}\ \underline{\tilde{\mathbf{C}}} - \mathbf{I})^{\mathbf{T}} (\underline{\mathbf{C}}\ \underline{\tilde{\mathbf{C}}} - \mathbf{I}) \right)_{ii} \right)^{\frac{1}{2}} . \tag{4.43}$$

A similar expression as equation (3.68) can be written for $\Delta\underline{\mathbf{C}}$ from

$$\Delta\underline{\mathbf{C}} = \dot{\underline{\mathbf{C}}}\Delta t\ , \tag{4.44}$$

where only first order changes are considered. Substituting the expression for $\dot{\underline{\mathbf{C}}}$, given by equation (3.78), as well as the expression for $\Delta\mathbf{P}$, given by equation (3.68), into equation (4.44) leads to

$$\Delta\underline{\mathbf{C}} = b'(\mathbf{I} - 2\mathbf{P})[\mathbf{F}, \mathbf{P}]\mathbf{C}\ , \tag{4.45}$$

with $b' = \frac{1}{2}b\delta t^2 = \frac{1}{2\mu}\delta t^2$. Following a similar argument as for the idempotency tolerance, the same values for b and δt can be used so that equation (4.12) can be integrated. The effect that the orthonormality tolerance has on the energy convergence is illustrated in figure 4.4.

From figure 4.4 it can be seen that if no orthonormality is enforced an unphysical situation arises which results in a vicious oscillation of energy. For a too large orthonormality tolerance the oscillations reduce significantly, but still convergence to the correct ground state energy is not reached. For a sufficient orthonormality tolerance the correct ground state energy of -210.085 eV is reached which correlates with that from a MNDO SCF calculation.[10]

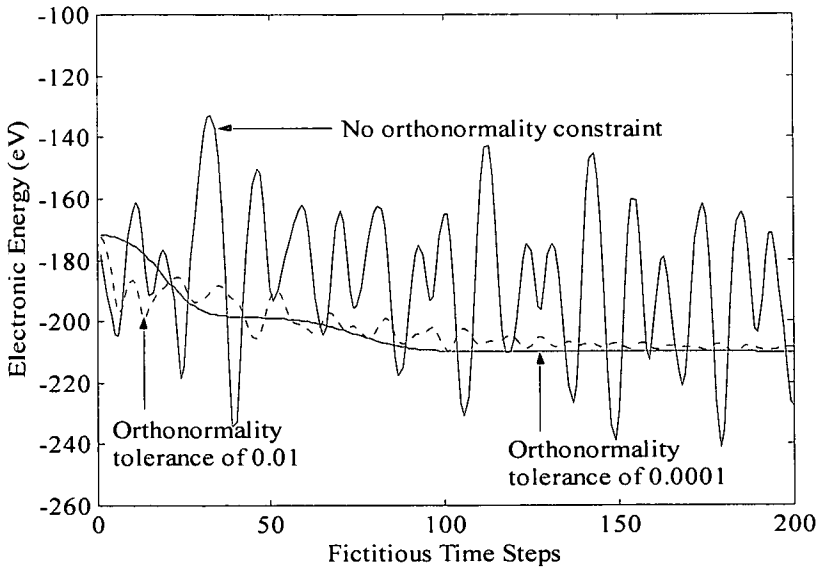

Figure 4.4: Illustration of the effect that the enforcement of orthonormality has on the energy convergence. The influence of the RATTLE algorithm (to ensure orthonormality) can be clearly seen.

4.3.3 Fictitious mass parameter, b'

It is best to view b and δt in terms of the fictitious mass parameter, b'. From equation (3.68) the fictitious mass parameter, b', is related to the fictitious mass, μ, through the expression $b' = \frac{1}{2\mu}\delta t^2$. From equations (3.68) and (4.45) it is thus clear that, for a given δt, a large fictitious mass, μ, would result in a small b' and thus a small $\Delta\mathbf{P}$ or $\Delta\underline{\mathbf{C}}$. The system is thus inert in reacting to the fictitious forces and moving to the ground state. For a small μ, b' is large and thus $\Delta\mathbf{P}$ or $\Delta\underline{\mathbf{C}}$ will be large. Thus, the system will react rapidly to the fictitious forces. However, $\Delta\mathbf{P}$ or $\Delta\underline{\mathbf{C}}$ is too large to conserve idempotency or orthonormality sufficiently, thus too much fictitious time steps are spent on purification or applying RATTLE. It thus follows that b' should be sufficiently small so that not too much fictitious time steps are used to enforce the constraints.

To determine what the value for b' should be for the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics respectively, a reasonably small value for b' was chosen which was consecutively increased by a certain

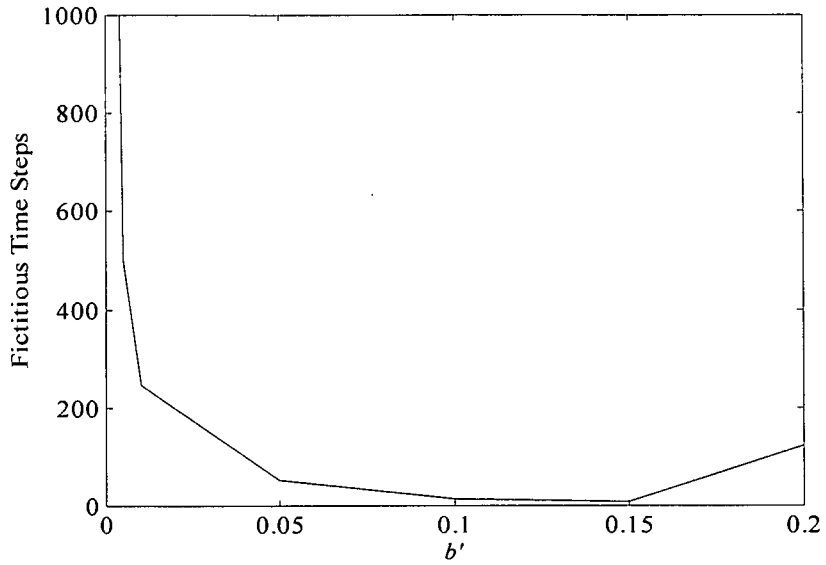

Figure 4.5: Illustration of the number of fictitious time steps necessary to reach energy convergence for a given value of b' for the **P***-dynamics.*

amount, each time noting the number of fictitious time steps necessary to reach energy convergence. In figure 4.5 it is clear that the number of fictitious time steps to reach energy convergence for the $\mathbf{P}$ dynamics decreases as b' increases, up to about where $b' = 0.15$, after which the fictitious time steps necessary for energy convergence increase again. For $b' > 0.2$, a large $\Delta\mathbf{P}$ is produced, resulting in a matrix $\mathbf{P} + \Delta\mathbf{P}$ with eigenvalues outside the range -$\frac{1}{2}$ and $\frac{3}{2}$, as shown in figure 4.2. McWeeny purification is thus insufficient in transforming $\mathbf{P} + \Delta\mathbf{P}$ into an idempotent matrix.

In figure 4.6 it can be seen that the number of fictitious time steps necessary to reach energy convergence for the $\underline{\mathbf{C}}$-dynamics also decreases as b' increases. For $b' > 0.01$, however, the RATTLE algorithm cannot successfully enforce orthonormality on $\underline{\mathbf{C}}$.

It can thus be concluded that, in the case of the $\mathbf{P}$-dynamics, to reach energy convergence in the least amount of fictitious time steps, b' should be between 0.05 and 0.15. Values for b' smaller than 0.05 can be used, but this will require much more ficititious time steps to reach energy convergence. It is best not to choose b' larger than 0.15 for

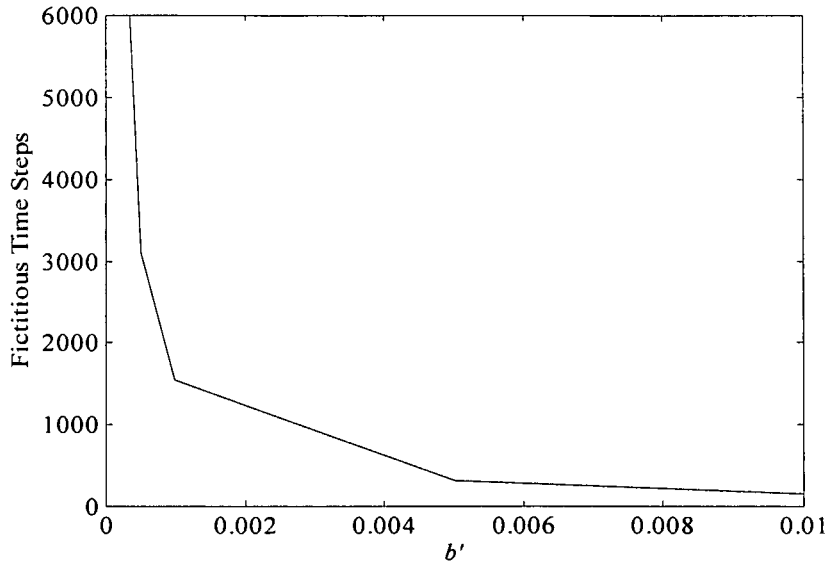

Figure 4.6: Illustration of the number of fictitious time steps necessary to reach energy convergence for a given value of b' for the $\underline{\mathbf{C}}$-dynamics.

the reason discussed in the previous paragraph. In the case of the $\underline{\mathbf{C}}$-dynamics, a value for b' between 5×10^{-3} and 0.01, will result in the fastest energy convergence. Values for b' smaller than 5×10^{-3} can be used, but this again will result in longer energy convergence. Choosing a value for b' larger than 0.01 is not advisable for the reason discussed in the previous paragraph.

Remark: *A fictitious time step refers to an iteration during the integration procedure whilst a time step refers to δt appearing in the equations of motion.*

Unfortunately, the fictitious time steps necessary to reach convergence are not the only consideration when selecting a fictitious mass parameter. If this was the case, it is obvious that the choice for b' would be the one that results in the fastest energy convergence, as discussed above. However, the conservation of the total energy in the system is an important factor that will be influenced by the choice for b'. Similar to the case in molecular dynamics where a large time step will result in the non-conservation of the total energy,[52] a large b' would most likely not guarantee the conservation of the

total energy. The value that b' should have to satisfy energy conservation, is discussed in more detail in section 4.5.

4.3.4 Effect of b and δt on the energy convergence

In the beginning of this section it was mentioned that the change in energy, ΔE, is dependent on the fictitious mass parameter, $b' = \frac{1}{2}b\delta t^2$, as is $\Delta \mathbf{P}$ from equation (3.68) and $\Delta \underline{\mathbf{C}}$ from equation (4.45). It is however necessary to confirm that the computer program used to perform the calculations, indeed reflects this behaviour.

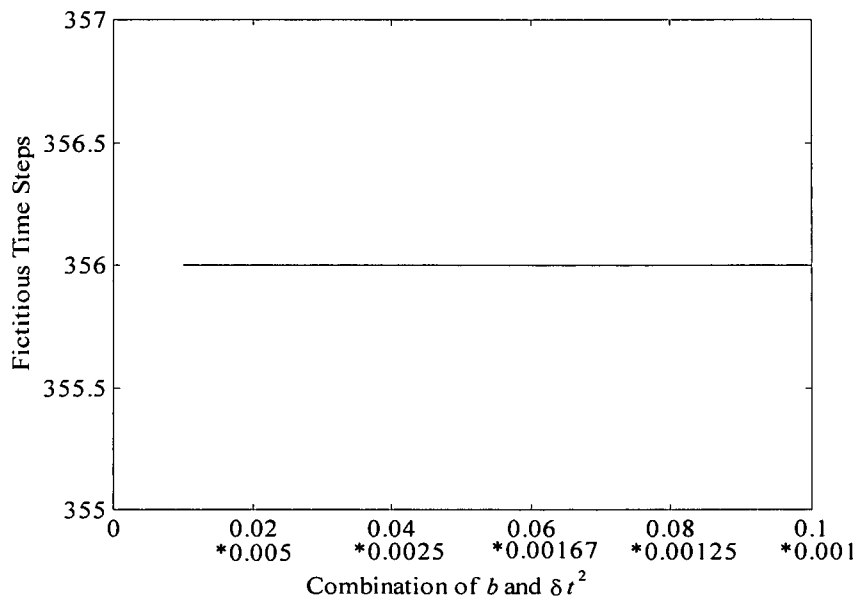

Figure 4.7: Illustration of the effect the combination $b\delta t^2$ has on the number of fictitious time steps necessary to reach energy convergence. For the first value on the x-axis, $b = 0.02$ and $\delta t^2 = 0.005$.

In order to do so, b is chosen as 0.01 and δt as 10^{-1} so that b' are in the order of 10^{-4} (as argued in section 4.3.1). Keeping this product constant, b can be decreased by a small amount whilst δt^2 is increased by the same amount, each time noting the

number of fictitious time steps necessary to reach energy convergence for this particular combination. From figure 4.7 it can be seen that the number of fictitious time steps necessary to reach energy convergence remains constant for a variety combinations of δt^2 and b. This thus confirms equation (3.73): the energy change is dependent on the combination of $b\delta t^2 = \frac{\delta t^2}{\mu}$ and not on them individually. Of course, this result also applies to the $\underline{\mathbf{C}}$-dynamics.

4.4 Influence of starting configurations

As mentioned in section 4.1.2, initial values for $\mathbf{P}$, $\dot{\mathbf{P}}$, $\mathbf{C}$ and $\dot{\mathbf{C}}$ are necessary to start the numerical integration. Suggestions for these initial values were also made. However, the effect of the initial values on the resultant energy convergence needs to be investigated more closely. It could for instance be asked if a "bad" selection of, for example the initial projection/density matrix, would result in failure to converge as is the case for SCF calculations used for example in MOPAC.[107] In this section, different initial values for $\mathbf{P}$, $\dot{\mathbf{P}}$, $\mathbf{C}$ and $\dot{\mathbf{C}}$ are chosen and the effect that each of these have on the energy convergence is observed.

Randomly selected projection matrix

In order to generate a random projection matrix of order N and rank M, associated with a random orientation of the corresponding subspace, M normalized, N dimensional vectors, $\{\mathbf{a}_1, ..., \mathbf{a}_M\}$, each with a random orientation, need to be generated and consequently the projection matrix

$$\mathbf{P} = \underline{\mathbf{A}}(\tilde{\underline{\mathbf{A}}}\underline{\mathbf{A}})^{-1}\tilde{\underline{\mathbf{A}}}$$

can be calculated with $\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_M]$. This can be done by generating a vector, $\{x_1, x_2, ..., x_N\}$, with each x_i selected from normal distribution:

$$f(x_i) = e^{-x_i^2/(2\sigma^2)} .$$

Then the distribution function for the vector $(x_1, x_2, ..., x_N)$ is given by

$$\begin{aligned} f(x_1, x_2, ..., x_N) &= e^{-x_1^2/(2\sigma^2)} e^{-x_2^2/(2\sigma^2)} ... e^{-x_N^2/(2\sigma^2)} \\ &= e^{(-x_1^2+...+x_N^2)/(2\sigma^2)} \\ &= e^{(-r^2)/(2\sigma^2)}, \end{aligned}$$

which is independent of direction. The corresponding vector, $\{x_1, x_2, ..., x_N\}$, thus have a random direction.

The effect of a number of randomly selected projection matrices on the energy convergence is illustrated in figure 4.8. It can be seen that energy convergence is reached regardless what the choice for the initial projection matrix is. However, it is clear that for some choices, less fictitious time steps are needed to reach energy convergence than for others. The choices for which more fictitious time steps are needed to reach energy convergence (the ones that lie on the plato) can probably be attributed to excited states. In section 3.1 it was mentioned that the configuration matrix contains only the lowest filled molecular orbitals and that the projection matrix is then constructed from the configuration matrix. If, say, a higher filled molecular orbital (corresponding to an excited state) replaces one of the lower filled molecular orbitals, the projection matrix constructed from this configuration matrix will most probably be one that lies on the plato illustrated in figure 4.8. In contrast to this, the projection matrices for which energy convergence were reached in the least amount of fictitious time steps, correspond closely to the prescribed selection for $\mathbf{P}$ as given in section 4.4.

The important thing to note from this analysis is that energy convergence is reached regardless of what the initial projection and thus density matrix is. The SCF method, on the other hand, is known to fail in reaching convergence for certain systems on bad choices of the starting density matrices.[107] The relative strength of the fictitious dynamics method in this regard versus the SCF method need to be explored in more detail in future.

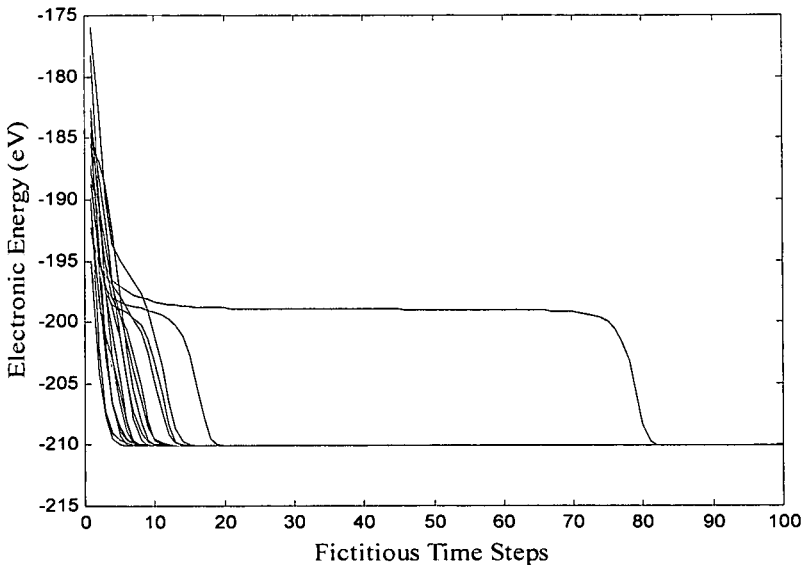

Figure 4.8: Schematic illustration of number of fictitious time steps needed to reach convergence for a number of randomly selected initial projection matrices with $b' = 0.01$.

Randomly selected $\dot{\mathbf{P}}$

By making use of equation (3.38), together with an appropriate scaling factor the following expression for $\dot{\mathbf{P}}$ can be obtained

$$\dot{\mathbf{P}} = k(\mathbf{I} - \mathbf{P})(\mathbf{A} + \tilde{\mathbf{A}}) \, . \tag{4.46}$$

If $\mathbf{A}$ is now a random $N \times N$ matrix, a random $\dot{\mathbf{P}}$, consistent with the constraints, can be generated. To make sure that a not too high fictitious kinetic energy is introduced to the system, the constant k must be such that $\dot{\mathbf{P}}$ is small enough. The effect of a number of randomly selected initial $\dot{\mathbf{P}}$'s on the energy convergence is illustrated in figure 4.9. It is evident from figure 4.9 that energy convergence is reached regardless of what the initial fictitious velocities are. Furthermore, it can also be seen that approximately the same amount of fictitious time steps are required to reach energy convergence for all the random fictitious velocities.

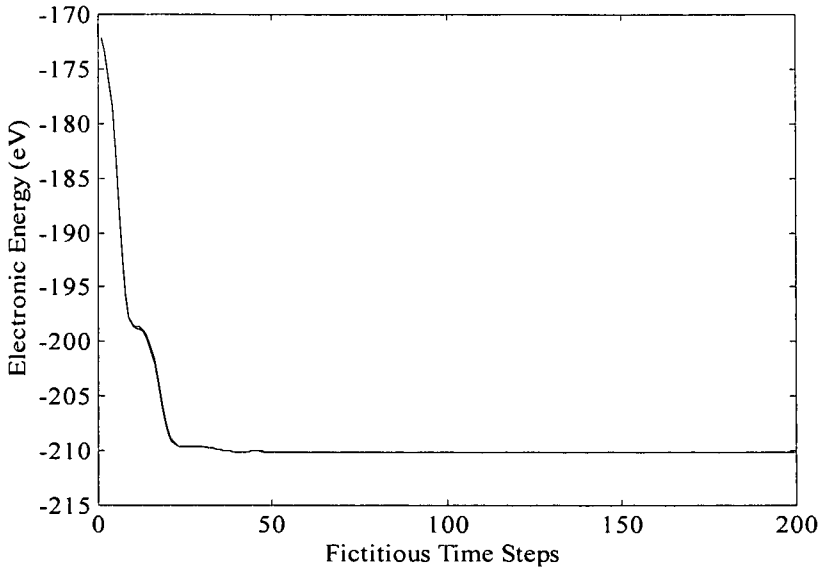

Figure 4.9: Schematic illustration of number of time steps needed to reach energy convergence for a number of randomly selected initial fictitious velocities with $b' = 0.01$.

Randomly selected $\underline{\mathbf{C}}$

A random $N \times M$ matrix can be generated and orthonormalized to serve as a random $\underline{\mathbf{C}}$. The effect of a number of randomly selected $\underline{\mathbf{C}}$ on the resultant energy convergence is illustrated in figure 4.10. Similar to before, energy convergence is reached, in this case, regardless of the what the configuration matrix is initially. It is also apparent that more or less the same number of fictitious time steps are needed to reach energy convergence for all the random configuration matrices. It is worthy to mention that, if the initial $\underline{\mathbf{C}}$ is not orthonormal, the RATTLE algorithm will enforce orthonormality on $\underline{\mathbf{C}}$ and this will result in energy convergence on the correct ground state energy.

Randomly selected $\underline{\dot{\mathbf{C}}}$

A randomly generated $N \times M$ matrix can be used as a random $\underline{\dot{\mathbf{C}}}$. This random matrix should of course be appropriately scaled to avoid a too high fictitious kinetic energy being introduced to the system. The effect that a number of randomly selected $\dot{\mathbf{C}}$'s has on the resultant energy convergence is illustrated in figure 4.11 from which it is clear that

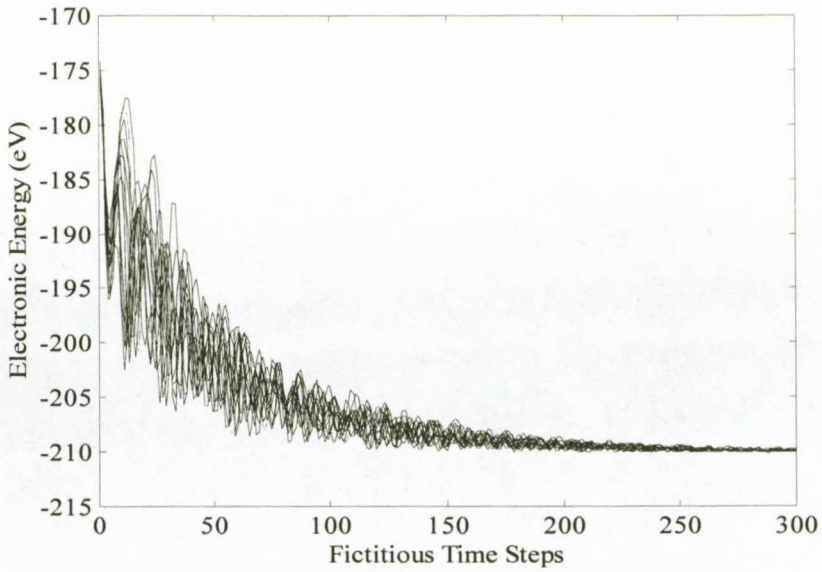

Figure 4.10: Schematic illustration of number of fictitious time steps needed to reach energy convergence for a number of randomly selected initial configuration matrices with $b' = 0.01$.

the energy convergence is not dependent on the initial values for the fictitious velocity. Furthermore, the number of fictitious time steps necessary for energy convergence is more or less the same for all random $\dot{\mathbf{C}}$'s.

The conclusion that can be drawn from the above analysis is that energy convergence is reached regardless of what the initial values for $\mathbf{P}$, $\dot{\mathbf{P}}$, $\mathbf{C}$ and $\dot{\mathbf{C}}$ are. For all of these different initial values, except for those of $\mathbf{P}$, more or less the same number of fictitious time steps were required to reach energy convergence. Furthermore, the equations of motion also exhibit the property that, should a "bad" selection be made for one of the initial values, they correct the initial values in such a way that eventual energy convergence is reached. As an example, a "bad" selection for $\mathbf{P}$ was one that led to the plato being created in figure 4.8. In this case, energy convergence was still reached, it just took significantly longer than for "better" selections for $\mathbf{P}$. It would thus be most advantageous to follow the guidelines as provided in section 4.1.2 since these require the least amount of fictitious time steps to reach energy convergence.

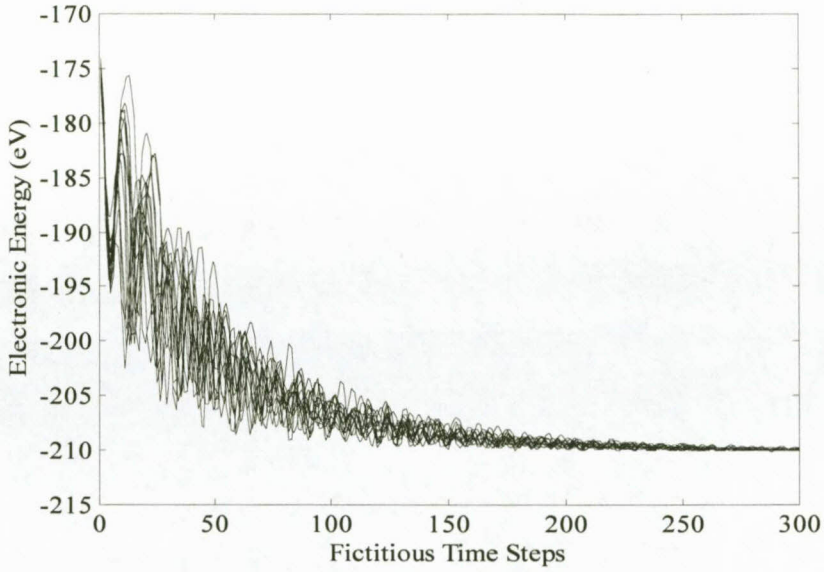

Figure 4.11: Schematic illustration of number of fictitious time steps needed to reach energy convergence for a number of randomly selected initial fictitious velocities with $b' = 0.01$.

4.5 Stability of the integrator and energy conservation

It was mentioned in section 4.2 that the integrator is the determining factor in the accuracy of a molecular dynamics run. Similarly, the integration of the equations of motion will determine how accurate the fictitious dynamics run are. One of the criteria that can be used to determine the accuracy of a simulation is the time-reversibility of the integrator. This means that, by changing the signs of all the velocities, the energy trajectory will be retraced. In other words, given the expression for the equations of motion (4.9) and (4.10), the expression for the fictitious position can be written as

$$\mathbf{P}(t-\delta t) = \mathbf{P}(t) - \delta t \mathbf{P}(t) - \frac{1}{2}\delta t^2[(\mathbf{I} - 2\mathbf{P}(t))(b[\mathbf{F}(t), \mathbf{P}(t)] - 2\dot{\mathbf{P}}(t)^2)] \qquad (4.47)$$

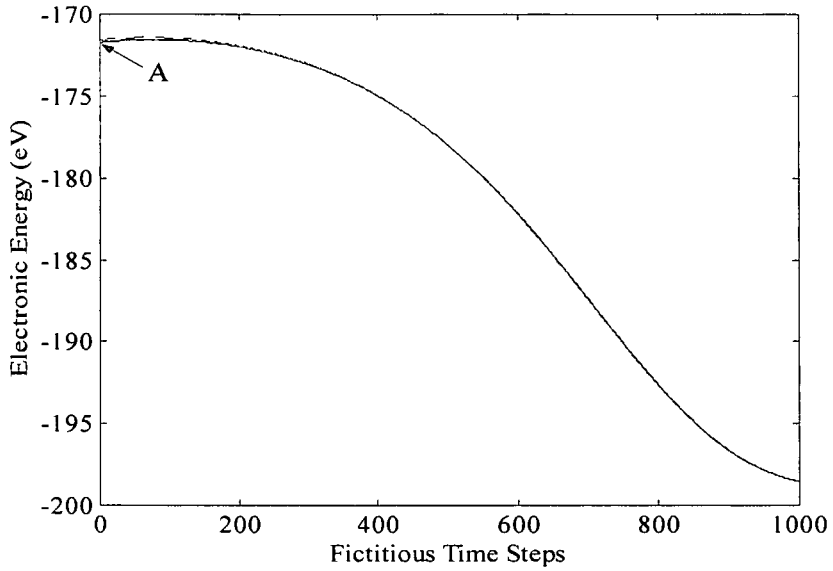

Figure 4.12: Time-reversibility for the $\mathbf{P}$*-dynamics for the* H_6 *ring with* $b' = 10^{-6}$*. The energy difference between the two trajectories at point A is 0.14 eV.*

and for the fictitious velocity

$$\dot{\mathbf{P}}(\mathbf{t} - \delta\mathbf{t}) = \dot{\mathbf{P}}(t - \frac{1}{2}\delta t) - \frac{1}{2}\delta t[(\mathbf{I} - 2\mathbf{P}(t))(b[\mathbf{F}(t), \mathbf{P}(t)] - \dot{\mathbf{P}}(t)^2)]. \tag{4.48}$$

Should the simulation proceed correctly, the above equations should retrace the energy trajectory as traced out by equations (4.9) and (4.10), thus the forward and reverse energy trajectory should be superimposed. A similar argumentation of course applies to the $\underline{\mathbf{C}}$-dynamics.

An investigation of the time-reversibility of the integrator thus serves as a convenient test to confirm the correctness of the programming. It can also indicate what the effect of the neglected fictitious acceleration term in equation (4.11) is.

Figures 4.12 and 4.13 illustrate the time-reversibility of the velocity Verlet integrator for the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics respectively. The accuracy of the time-reversibility is determined by calculating the energy difference of the two trajectories (the one forward in time and the other reversed in time) at point A as indicated in figures 4.12 and 4.13.

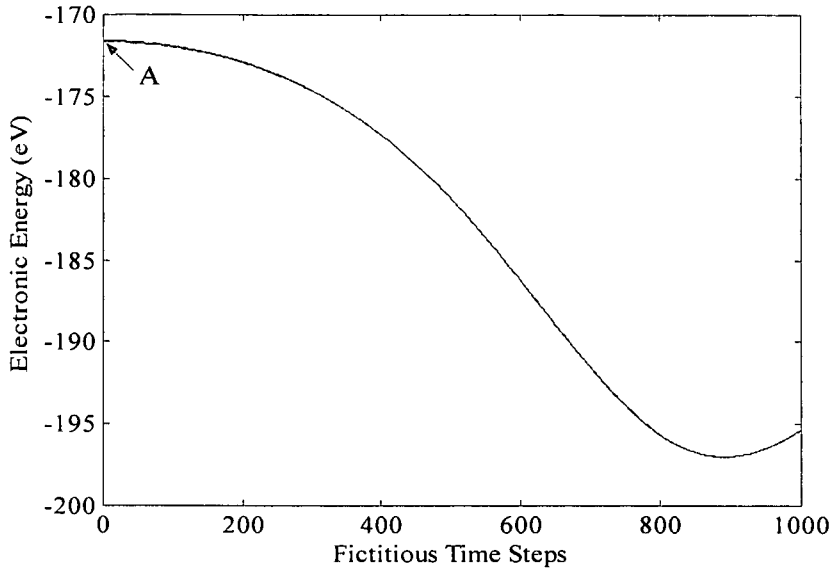

Figure 4.13: Time-reversibility for the $\underline{\mathbf{C}}$-dynamics for the H_6 ring with $b' = 10^{-6}$. The energy difference between the two trajectories at point A is 0.085 eV.

Another criteria which can be used to determine the accuracy of a simulation is the conservation of the total energy. Thus, the sum of the kinetic and potential energy should remain constant during the course of the simulation. This requirement carries more weight than the time-reversibility of the integrator and should thus be considered more carefully. This being because time-reversibility does not necessarily guarantee energy conservation. Figures 4.14 - 4.17 illustrates the energy conservation, using either a fictitious mass parameter of 10^{-4} or one of 10^{-6}, for the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics respectively.

It can be concluded from the above analysis that the error in the total energy for both the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics, taken at point B in figures 4.14 - 4.17, over a number of fictitious time steps, is roughly 5.6 % and 0.25 % for $b' = 10^{-4}$ and $b' = 10^{-6}$ respectively. It is thus evident that the percentage error in the total energy decreases for a decreasing fictitious mass parameter, b'. Should better accuracy be required, a smaller value for b' can be chosen, for example, $b' = 10^{-10}$ would result in a percentage error in the

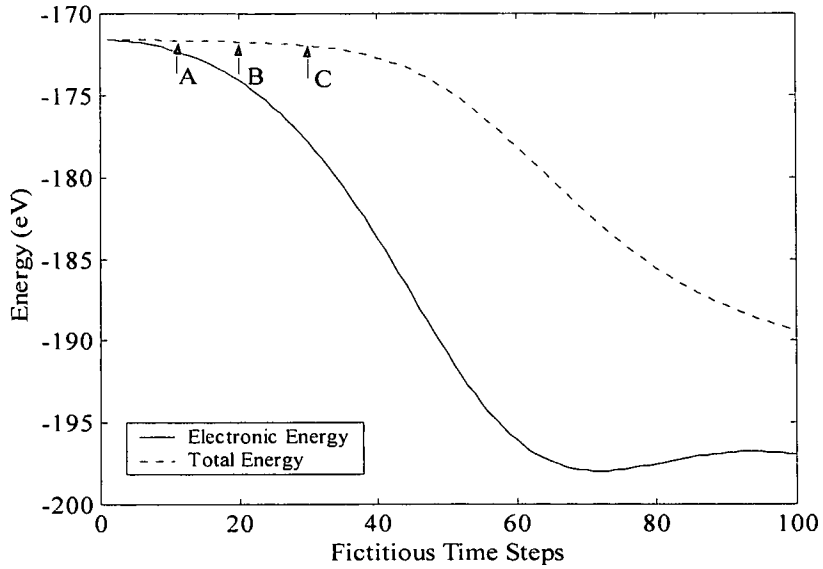

Figure 4.14: Energy conservation for the $\mathbf{P}$*-dynamics for the* H_6 *ring with* $b' = 10^{-4}$*. The differences in energy between the initial value of the total energy and point A, B and C are 0.053 eV, 0.14 eV and 0.37 eV respectively.*

total energy of about 10^{-4} %. This, however, will significantly increase the number of fictitious time steps necessary to reach energy convergence. It is thus imperative to find a balance between the amount of accuracy required during a calculation and the computational resources that are available. It can be noted in passing that a possible strategy for increasing the speed of a calculation can be to start with a large value for b', say 10^{-3} and then, as soon as the error in the total energy exceeds some pre-set limit, b' can be appropriately scaled so that the error in the total energy satisfies the pre-set limit. The extent to which energy should be conserved in a fictitious dynamics run can however only be properly addressed when the fictitious dynamics are combined with the atomic dynamics.

It is worth mentioning that the fictitious parameter, $b' = 10^{-6}$, necessary to satisfy energy conservation for a certain amount of time steps, also proves adequate for the velocity Verlet integrator to be time-reversible, as can be seen in figures 4.12 and 4.13.

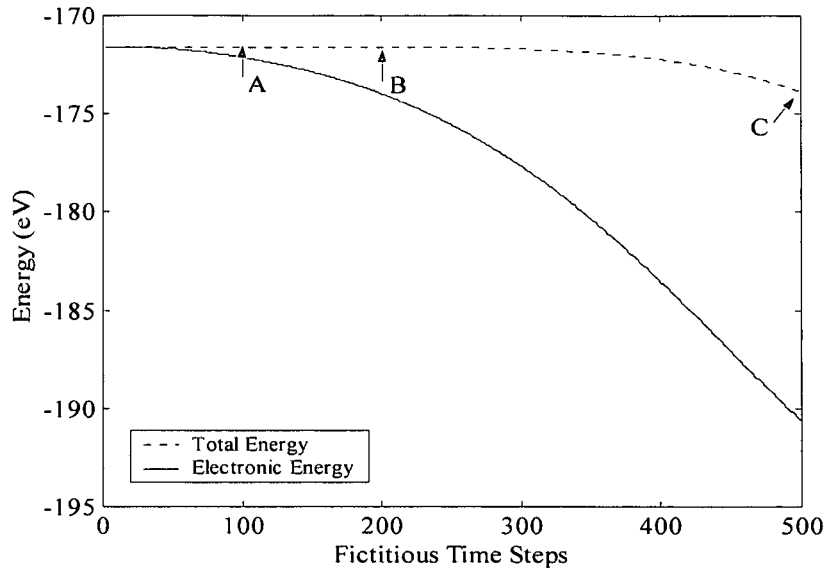

Figure 4.15: Energy conservation for the $\mathbf{P}$*-dynamics for the* H_6 *ring with* $b' = 10^{-6}$*. The differences in energy between the initial value of the total energy and point A, B and C are 0.0058 eV, 0.021 eV and 2.29 eV respectively.*

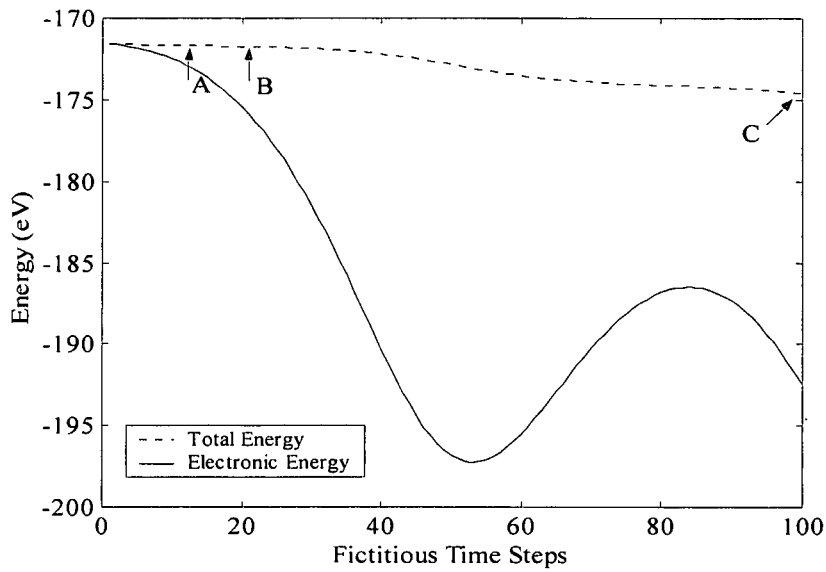

Figure 4.16: Energy conservation for the $\underline{\mathbf{C}}$*-dynamics for the* H_6 *ring with* $b' = 10^{-4}$*. The differences in energy between the initial value of the total energy and point A, B and C are 0.06 eV, 0.15 eV and 0.28 eV respectively.*

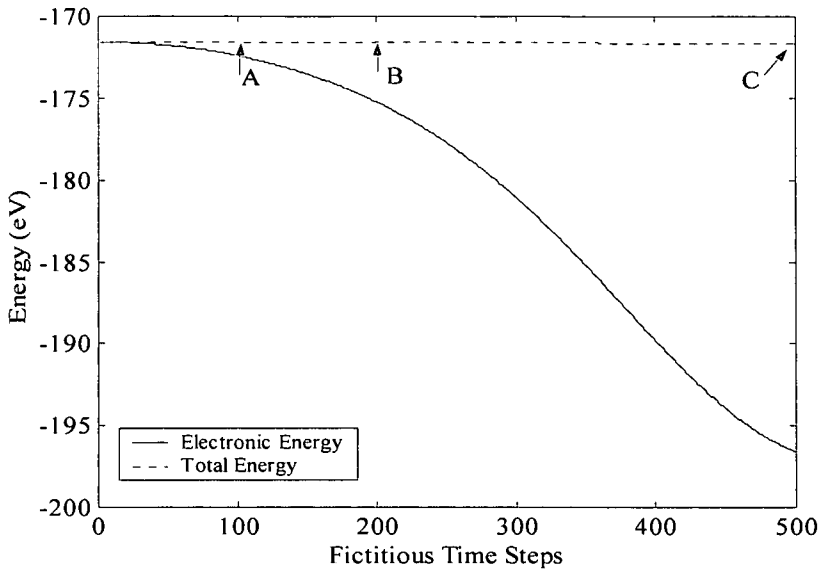

Figure 4.17: Energy conservation for the $\underline{\mathbf{C}}$*-dynamics for the* H_6 *ring with* $b' = 10^{-6}$*. The differences in energy between the initial value of the total energy and point A, B and C are 0.0034 eV, 0.016 eV and 0.13 eV respectively.*

It can be illustrated that the choice of the fictitious mass parameter does not only apply to this particular system (the H_6 ring). This is done by considering the energy conservation during the calculation of the electronic structure of a Si_9H_{12} cluster (this is discussed in the next chapter) were also considered. For both the **P**-dynamics and $\underline{\mathbf{C}}$-dynamics a fictitious mass parameter of 10^{-6} were chosen. Figures 4.18 and 4.19 show that the conservation of total energy, given the same b' that was used during the calculation of the elctronic structure of the H_6 ring, still applies, even though the system under study differs. The error in total energy for both the **P** and $\underline{\mathbf{C}}$-dynamics, taken at point B in figures 4.18 and 4.19, is roughly 0.5%.

It is now clear that the neglect of the fictitious acceleration at a full time step does not significantly effect the conservation of energy in the equations of motion. As long as a small enough fictitious mass parameter is used, this term can be ignored. In future,

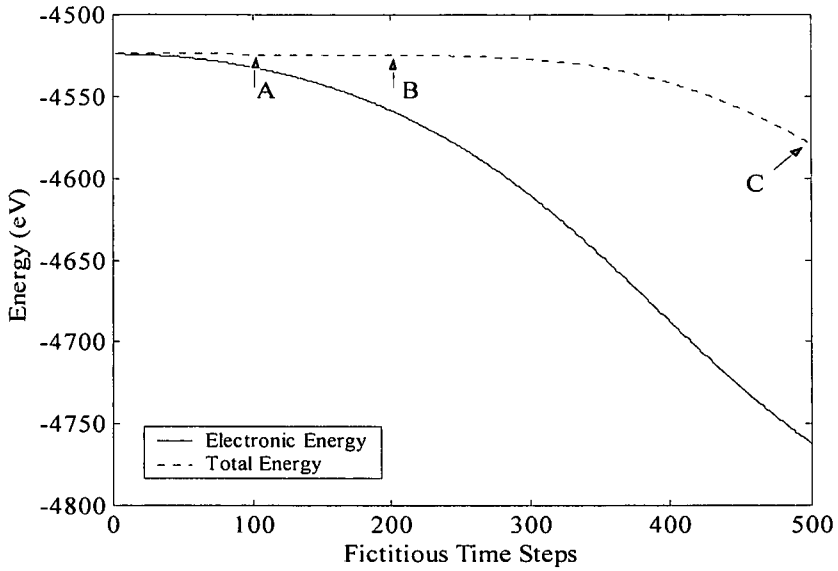

Figure 4.18: Energy conservation for the **P***-dynamics for the* Si_9H_{12} *cluster with* $b' = 10^{-6}$. *The difference between the initial value of the total energy and A, B and C is 0.078 eV, 0.37 eV and 5.5 eV respectively.*

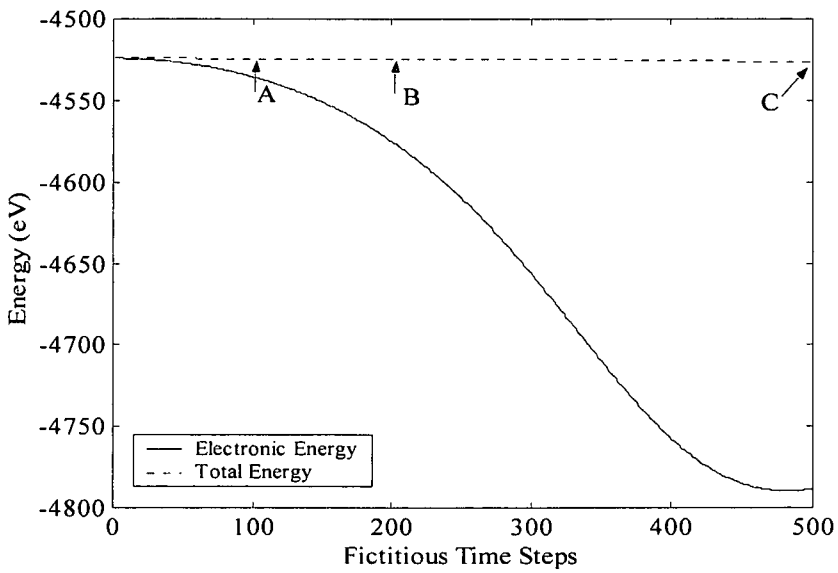

Figure 4.19: Energy conservation for the **C***-dynamics for the* Si_9H_{12} *cluster with* $b' = 10^{-6}$. *The difference between the initial value of the total energy and A, B and C is 0.11 eV, 0.24 eV and 2.51 eV respectively.*

however, an approximation to this term can be made in order to use a larger fictitious mass parameter which would lead to faster energy convergence.

4.6 Velocity quenching in the equations of motion

Molecular systems can exist in different conformational geometries, corresponding to different three-dimensional arrangements of atoms in a structure. The number of conformations allowed increases with molecular size. In particular, macromolecules of biological interest present a large number of local minima (or local equilibrium conformations). Indeed, geometry optimization is a crucial first step in any molecular modeling strategy. The main obstacle in this field is to find a global minima and not to get trapped in one of the many local minima. The usual optimization methods applied to quantum and classical molecular physics are based on the gradient descent approach, which indistinctly selects both global and local minima. Therefore, in locating the global minima, brute force has been the usual tool. In other words, a single geometry optimization will not necessarily deliver the global minima on the first trail - this will most likely be a local minima. Furthermore, different initial geometries will deliver different local minima. Thus, to find a global minimum, (or at least, to be more confident about it) many minimizations using different initial coordinates for each run, should be done.

The so-called simulated annealing methods have demonstrated important successes in the description of a variety of global extremization problems. Simulated annealing methods have attracted significant attention as suitable for large-scale optimization problems, especially for those where a desired global minimum is hidden among many local minima. The basic aspect of the simulated annealing method is its analogy with thermodynamics, especially in regard to the way that liquids freeze and crystallize, or that metals cool and anneal. The first nontrivial solution along this line was provided

by Kirkpatrick et al[99] for classical systems, and further extended by Ceperley et al[100] to quantum systems.

Thus, in simulated annealing, MD calculations are started with a relative high temperature (high kinetic energy) and this is then slowly reduced during the course of the simulation. This allows the simulation to move far from the starting geometry before a local minimization is performed. If the cooling is done very slowly, the resulting minimum is the global minimum. Practically, simulations are done on a very short time scale and only the local area is probed.

In the calculation of the ground state energy for the H_6 ring, the expectation was that the kinetic energy gained as the potential energy (electronic energy) decreased, should be quenched on approaching equilibrium. However, this was not necessary: the equations of motion automatically quench the fictitious velocity and thus kinetic energy as the equilibrium is approached, as illustrated in figures 4.20 and 4.21. From figures 4.20 and 4.21 it can be seen that, as the equilibrium is approached, both the fictitious acceleration velocity tends to zero. A fictitious mass parameter of 10^{-2} was used for both the $\mathbf{P}$ and $\underline{\mathbf{C}}$-dynamics. In figure 4.14 it was seen that, for $b' = 10^{-4}$, energy is not conserved for long - at most for about 40 fictitious time steps, depending on the accuracy required. This non-conservation of energy can be associated with the automatic velocity quenching for large enough b', which is convenient if the primary interest is calculating the electronic ground state for a fixed atomic geometry.

The effect of the velocity quenching on the energy conservation can be illustrated by considering figure 4.22. For this case, a fictitious mass parameter of 10^{-4} is used, thus figure 4.22 can be compared with figure 4.15. In figure 4.22 the fictitious velocity is expressed (in arbitrary units) in terms of the Euclidian norm of $\dot{\mathbf{P}}$. The fictitious kinetic energy, K_f, can indeed be expressed in terms of the Euclidian norm of $\dot{\mathbf{P}}$, $\left\|\dot{\mathbf{P}}\right\|$.

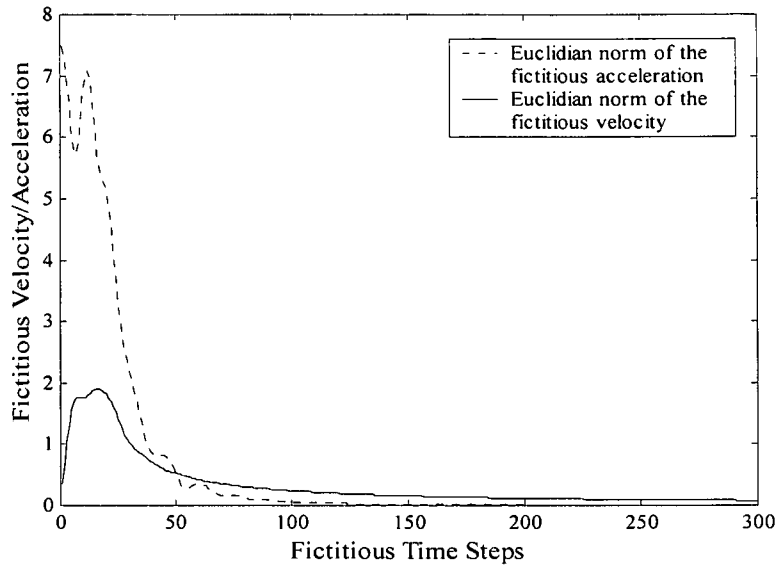

Figure 4.20: Illustration of the automatic velocity quenching in the equations of motion for the $\mathbf{P}$*-dynamics for the* H_6*-ring with* $b' = 10^{-2}$.

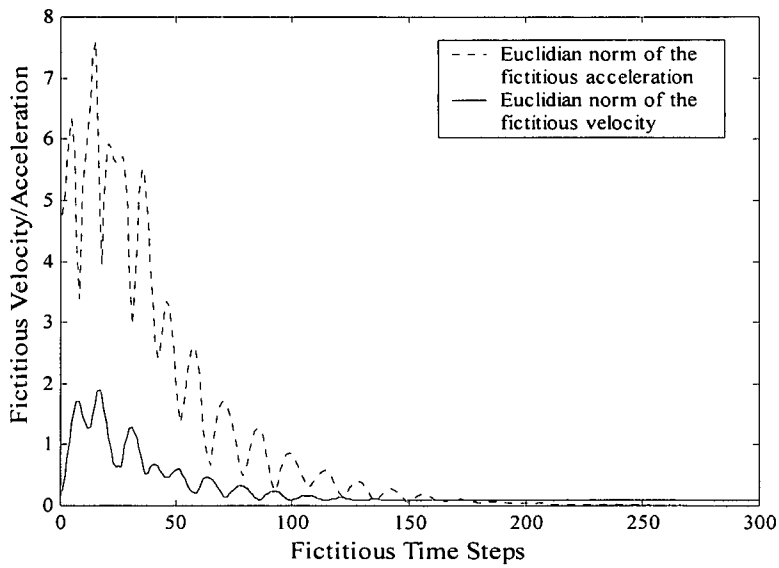

Figure 4.21: Illustration of the automatic velocity quenching in the equations of motion for the $\underline{\mathbf{C}}$*-dynamics for the* H_6*-ring with* $b' = 10^{-2}$.

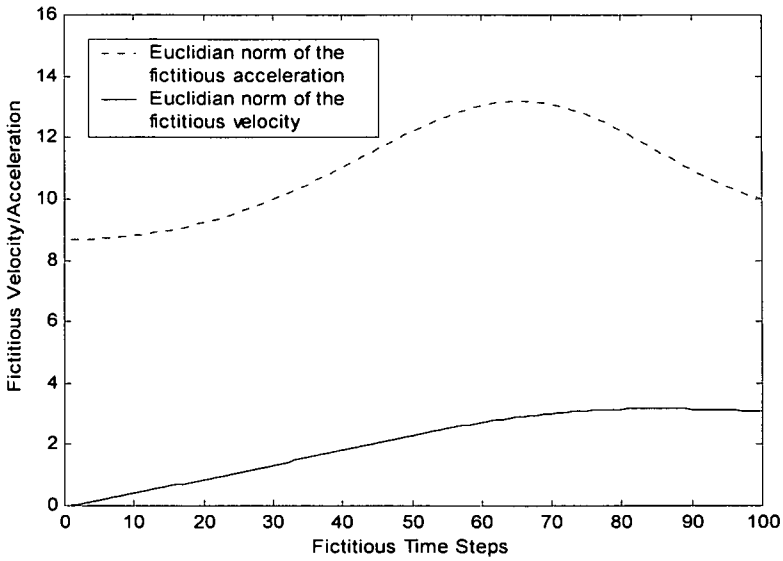

Figure 4.22: Illustration of the automatic velocity quenching in the equations of motion for the $\mathbf{P}$ *dynamics for the* H_6*-ring with* $b' = 10^{-4}$.

In other words

$$\begin{aligned} K_f &= \frac{1}{2}\mu_{ij}\sum_{ij}\dot{P}_{ij}^2 \\ &= \frac{1}{2}\mu_{ij}\sum_{ij}\dot{P}_{ij}\dot{P}_{ij} \\ &= \frac{1}{2}\mu_{ij}\sum_{ij}\tilde{\dot{P}}_{ji}\dot{P}_{ij} \\ &= \frac{1}{2}\mu\left\|\dot{\mathbf{P}}\right\| . \end{aligned}$$

The fictitious kinetic energy is thus proportional to the Euclidian norm of $\dot{\mathbf{P}}$.

From figure 4.22 it can be seen that, at about 40 fictitious time steps, the fictitious acceleration decreases and the fictitious velocity reaches a constant value for about 10 fictitious time steps after which it decreases. This explains why the total energy (figure 4.15) is not conserved from about 40 fictitious time steps onwards: the fictitious velocity and thus kinetic energy reaches a constant value between 50 and 60 fictitious time steps decreasing after that, whilst the electronic (potential) energy continues to

decrease. The potential energy that is lost is clearly not gained as kinetic energy. Energy is thus not conserved.

Chapter 5

The Si(100)2x1:H system

It still remains to be tested whether the fictitious dynamics method keeps the calculated minimum electronic energy close enough to the Born-Oppenheimer energy for situations of changing atomic geometry. With the future endeavor to describe the dynamics of, for example, hydrogen on the Si(100)2x1 surface, this system was selected as an ideal test system to study the deviation from the Born-Oppenheimer energy.

Furthermore, the intent in this study was also to benchmark results obtained with the fictitious dynamics method with that of the semi-empirical methods used in MOPAC. It is thus necessary to explore whether MOPAC can sufficiently describe the energy surface underlying the dynamics of hydrogen on the Si(100)2x1 surface. Although MOPAC has been used before to study the nature of the dimer bond of Si(100)2x1,[65] it has not yet been intensively applied in the study of hydrogen diffusion on this particular surface. Thus, a semi-empirical investigation of the diffusion barriers on the Si(100)2x1 surface is required to verify that MOPAC can indeed be used to describe the above mentioned energy surface.

5.1 The Silicon surface

Where would we be without silicon? This abundant element - it comprises 27 percent of the Earth's crust - provides the starting material for nearly all the semiconductor devices, from VCRs to the Space Shuttle, that have transformed the way we live. In computing and communications, 40 years of continuing innovation in smaller, faster circuitry, with silicon as the base material, is driving us headlong into the Information Age.

Silicon remains the choice for many applications, despite other materials that promise faster electronics, because it is durable and reliable, and over the years many problems associated with producing chip-quality crystals in commercial quantity have been solved. That, however, hardly means that no challenges remain in silicon chip technology. Whether it is to solve important problems in science or for the rapidly evolving new world of communications and entertainment, we need faster circuitry and smaller, more perfect chips.

In order to address the challenges in silicon technology, some basic understanding of this semi-conductor and particularly of its surface is needed. The Si(100) surface was first observed in 1959 in a LEED (Low Energy Electron Diffraction) study by Schlier and Farnsworth.[77] They found evidence for a (2x1) periodicity, which they attributed to a reconstruction based on dimer formation. Further investigations showed evidence of higher periodicities (p(2x2) and c(4x2) where two adjacent dimers buckle in the same and the opposite direction respectively) which depended on sample preparation and treatment. A variety of models were proposed, which featured dimer models, and various other more drastic reconstructions (e.g. the dimer plus chain model of Northrup[72] or the vacancy model of Poppendieck et al.[74])

With the advent of scanning tunneling microscopy (STM),[85] a real space image of the electronic structure of the silicon surface could be obtained, an example of which is shown in figure 5.1. It should be kept in mind that an STM image is a represen-

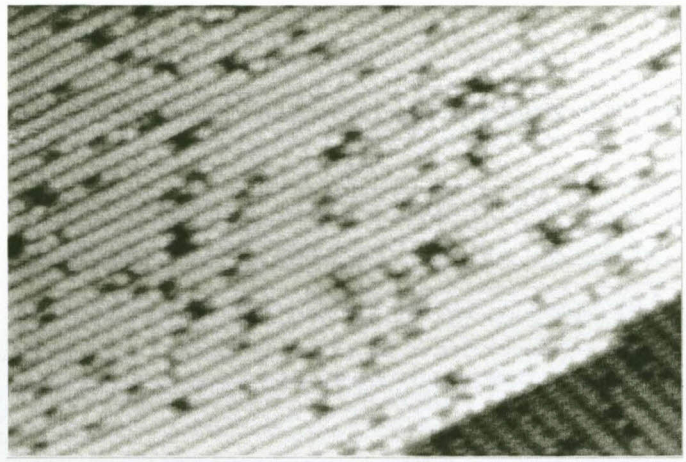

Figure 5.1: An STM image[106] of the Si(100)2x1 reconstruction.

tation of the electronic density of the surface and is not necessarily a morphological image. Tromp et Al[78] did just this and followed it up with a more comprehensive paper [70] in which they compared the various models to their pictures. This effectively established the dimer as the basic unit of the surface reconstruction. However, further controversy ensued. There was significant doubt about whether the dimers were symmetric or buckled (as first suggested by Chadi[67]) and also about the high level of defects on the surface. A schematic diagram of the reconstructions is shown in figures (5.2) and (5.3). Various different authors predicted that symmetric (Pandey,[73] Ihara et al,[71] Verwoerd[65]) or buckled (Chadi,[68] Roberts and Needs[76]) dimers were more stable - the essential point being that the energy difference between the two was small. Symmetric dimers seemed unlikely, as the surface would become metallic due to the degeneracy of surface states. The question as to why dimers might appear symmetric in the STM if they were really buckled was answered with the idea that they were flipping very quickly between the two possible buckled states (Dabrowski and Scheffler[69]). The breakthrough in understanding came when Wolkow[79] imaged the Si(100)2x1 surface as he cooled it to 120 K. He saw a significant increase in the number of buckled dimers as he cooled, finding a maximum of 80% buckling at 120 K. He also observed (as have others) that buckling at room temperature occurs next to step edges and defects, where

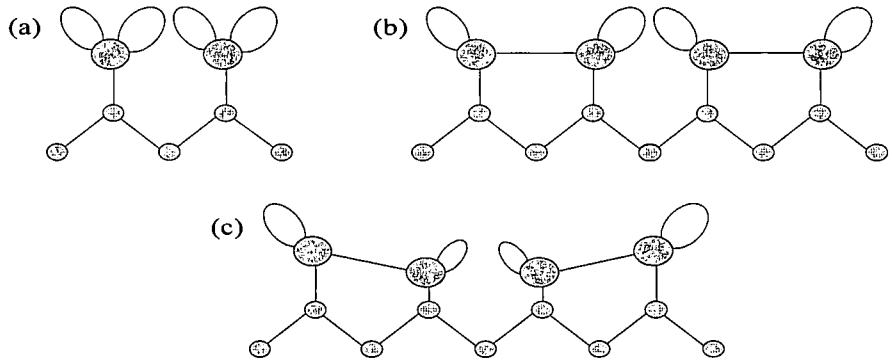

Figure 5.2: The figure[106] above shows a cross section through the bulk terminated surface. If a Si crystal is cleaved along the (100) plane each surface atom has two dangling bonds (a). To reduce the number of dangling bonds each surface atom bonds with a neighbour, forming a dimer (b). The resulting reconstruction has a 2x1 periodicity. The dimers buckle to minimize the surface energy (c).

pinning will raise the barrier for flipping.

The conclusion which can be drawn from this work is that the silicon (100) surface consists of alternately buckled dimers which are bistable, and oscillate backwards and forwards between their two states as illustrated in figures 5.2 and 5.3. Previous calculations which found that the symmetric orientation was more stable may well have been insufficiently converged (Dabrowski and Scheffler[69]), and a study by Ramstad et al[75] has found that, with insufficient convergence with respect to plane waves, symmetric dimers are more stable, and that the difference between p(2x2) and c(4x2) is too small to resolve accurately in local density approximation (LDA).

5.2 Hydrogen diffusion on the silicon surface

Hydrogen on silicon is an interesting system to study for two main reasons: firstly, it is a prototypical system for diffusion, which is easy to image in STM, and is therefore good to model as there is comparative data available; secondly, it plays an important

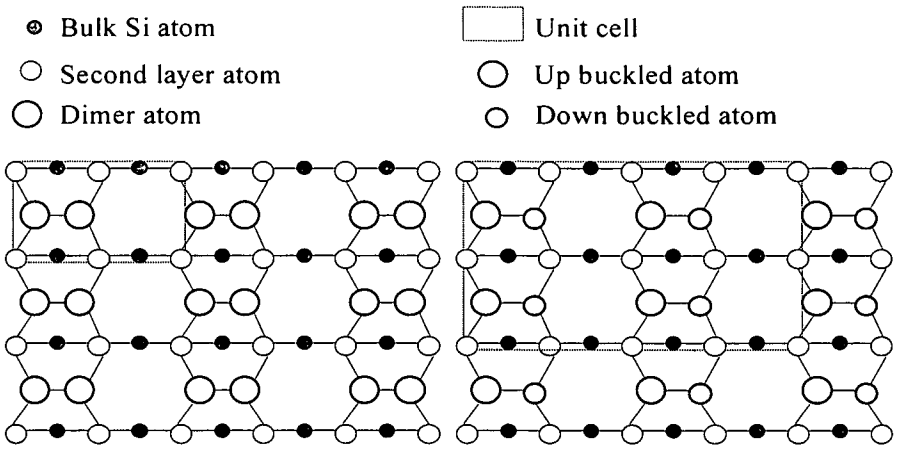

Figure 5.3: The schematic[106] above shows a plane view of the surface for the 2x1 reconstruction and for the c(4x2) reconstruction which can be observed under certain conditions and consists of buckled dimers (as shown above in figure 5.2 (c)).

role in the growth of silicon from disilane. At low temperatures, the silicon growth rate is limited by the hydrogen desorption rate[24][90] since the adsorbed hydrogen can block incoming disilane molecules or fragments, although growth can proceed slowly on a hydrogen saturated surface.[80] It can also influence the rate of diffusion of Si atoms,[84] which then affects the morphology and the rate of growth.

Hydrogen adsorbs on the surface of silicon by breaking the π-bond of the silicon dimer[24] and a bond to one of the dangling bonds is thus created. The remaining half-filled dangling bond appears as a bright blob in the STM, while the hydrogen at the other end is relatively dark,[81] due to the bonding energy gain as can be seen in figure 5.4. Hydrogen adsorbs randomly at coverages of less than 0.1 monolayer,[82],[83] but after that it pairs up onto a single dimer.[88] This arrangement, with one hydrogen on each end of a silicon dimer, is the most stable phase for adsorption, as the silicon atoms are both four coordinate, with no dangling bonds as illustrated in figure 5.5. Widdra et al[88] found that this pairing occurs at adsorption temperatures between

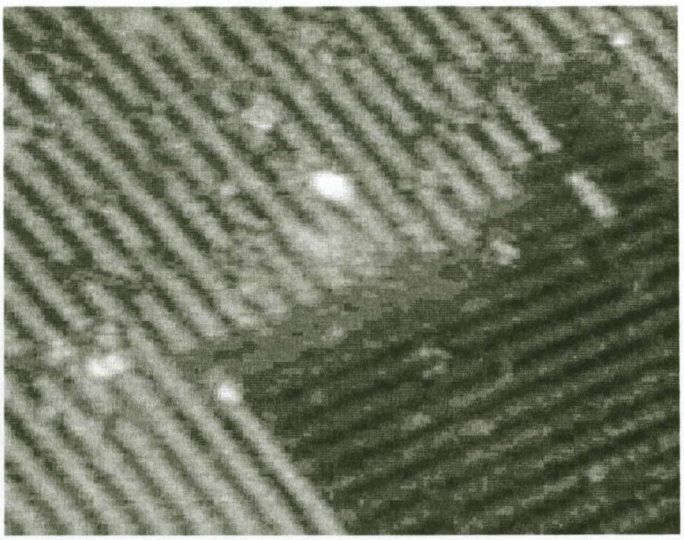

Figure 5.4: This STM image[106] shows the Si(100)-2x1:H that is formed by reacting hydrogen atoms with the clean Si(100)-2x1 surface[82]. The bright spots that are observed on the STM image are regions where the dangling bonds have not been saturated by the hydrogen. These can act as adsorption sites for many types of adsorbate.

150K and 600K, and that the amount of pairing was independent of temperature; they proposed a mobile precursor mechanism (similar to that proposed by Sinniah et al[86] for desorption) to explain this. In this mechanism, the incoming hydrogen atom is in an excited band state with a finite lifetime, which will sample several different adsorption sites. If the lifetime allows it to sample, say, 10 sites, then for coverages below 0.1 ML, the hydrogen distribution will appear random, and above that it will be almost entirely paired. Confirming that the diffusion barrier is sufficiently high to prevent diffusion upon adsorption over a good part of the temperature range used (150-600 K), would lend weight to this argument. These two experiments suggest that the diffusion barrier is sufficiently high to prevent hydrogen diffusion upon adsorption; these are the only experimental evidence available on diffusion.

There have been many theoretical investigations of hydrogen diffusion on Si(100), though none of them have been conclusive. Wu and Carter have performed many cluster calculations using correlation interaction (CI) techniques and initially[89] pre-

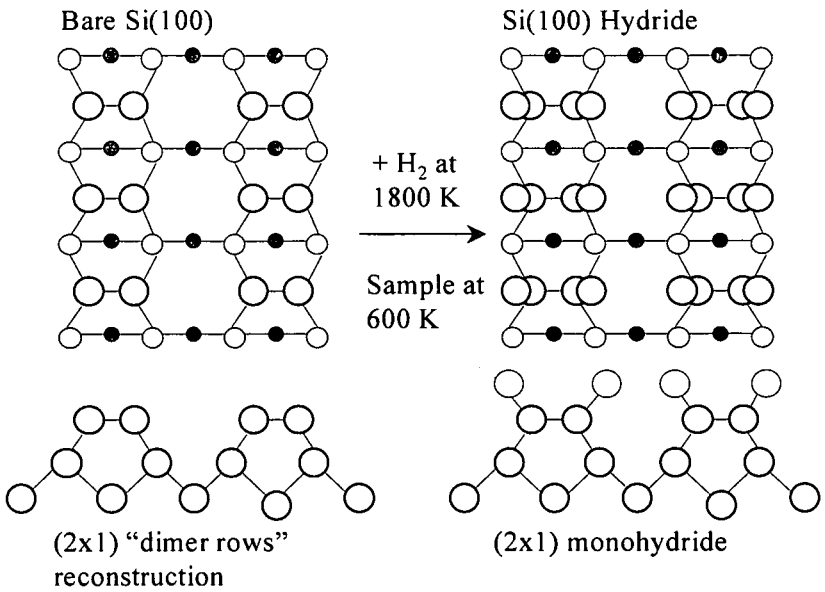

Figure 5.5: Schematic illustration[106] of the Si(100)-2x1:H structure as shown in figure (5.4). Note that the buckling of the dimers are not illustrated for the bare Si(100), but it can occur as shown in figure (5.3). The dimers are buckled at about 15^o*, but upon adsorption of hydrogen the buckling for a singly occupied dimer is about* $2-4^o$ *whereas it disappears for doubly occupied dimers.[87]*

dicted diffusion barriers of 2.0 eV along the dimer rows and 2.7 eV across the dimer rows. However, they used a small cluster consisting of 9 silicon atoms terminated with 12 "pseudo-hydrogen atoms", and did not allow the substrate to relax during diffusion, which suggests that their barrier will give a poor answer compared to the actual barrier. The CI method finds a very accurate answer, but the degree of relevance of the answer found to the physical system is rather small, due to the use of small clusters. When they later performed different calculations[25] using empirical potentials (based on the Stillinger-Weber potential) which they had generated from SCF-LCAO calculations, and allowed the slab substrate to relax, they found a significantly lower barrier of 1.5 eV for diffusion along the dimer row and a slightly lower value of 2.5 eV for diffusion across dimer rows.

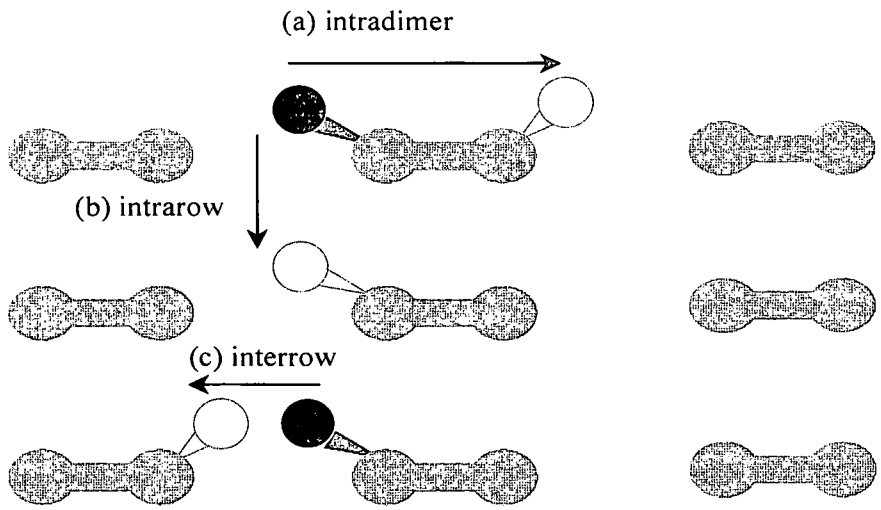

Figure 5.6: The three diffusive pathways of a single hydrogen atom diffusing on the Si(100)2x1 surface.[28] These are intradimer, intrarow and interrow, as explained in the text.

This lower barrier fits with calculations by Vittadini et al[22] who found a barrier of 1.3 eV, though it could be argued that using empirical potentials to model a process where the bond breaking is vital, as Wu, Ionova and Carter did, will only get results by luck. Vittadini et al used local density approximation (LDA), and a repeated slab for their calculations, and allowed the top four layers of the slab to relax. They claim that the major difference between their calculations and those of Wu and Carter[89] is the lack of surface relaxation in Wu and Carter's study. They attempt to estimate the magnitude of this effect by performing their diffusion calculations without allowing the substrate to relax, and conclude that the effect can be as large as 1 eV.

All of these sources of evidence indicate that the diffusion of hydrogen on Si(100)2x1 has an activation barrier of at least 1.3 eV, and that hydrogen is unlikely to hop across the dimer rows as this is likely to occur in a similar temperature regime as desorption.

5.3 Diffusion on perfect terraces

The Si(100)2x1 surface is highly anisotropic, and contains many features such as missing-dimer defects and steps. It is important to understand the behaviour of hydrogen on the perfect surface before considering the effect of such features. There are three different directions in which a hydrogen atom adsorbed onto one end of a terrace dimer may hop in order to reach an equivalent site. These are shown schematically in figure 5.6. The closest site is the empty dangling bond at the other end of the silicon dimer (figure 5.6 (a)). The next closest site is the next dimer within the same dimer row (figure 5.6 (b)). Lastly, the hydrogen atom may hop across dimer rows onto the next dimer row (figure 5.6 (c)), though this hopping is rarely observed.

5.4 Methodology

In this study the Si_9H_{13}, $Si_{15}H_{17}$ and $Si_{23}H_{25}$ clusters, depicted in figure 5.4, are used to model the intradimer, intrarow and interrow H-atom diffusion processes respectively. Hydrogen atoms are used to terminate subsurface Si atoms. In order to obtain initial geometries of the clusters, in the absence of an adsorbed hydrogen atom, the geometry of the Si_9H_{12} cluster is optimized using the BGFS optimization method as provided by the MOPAC software package. This is done by allowing the surface and subsurface atoms to relax. The rest of the cluster is kept rigid. The $Si_{15}H_{17}$ and $Si_{23}H_{25}$ clusters are then constructed from the Si_9H_{12} cluster by means of translational symmetry and optimized by relaxing the surface and subsurface atoms.

Although cluster calculations are not the most efficient choice regarding the modeling of an infinite periodic system, this method is chosen since comparable results with other theoretical calculations, which were done using clusters, are desired. However, a standard option in the MOPAC package is to make use of a translational vector, which is supplied by the utility DEPVAR in which a symmetry-defined bond-length is related

to another by a multiple. The use of such a vector is considered to be superior in the modeling of infinite periodic systems.

The reaction barriers for diffusion are found using a very simple method. In all cases presented in this section, it is easy to find an unambiguous reaction coordinate: a line in real space from the start to the end position. The diffusing hydrogen atom is constrained to lie in a given plane normal to the reaction coordinate, an the energy is minimized with respect to the other atomic coordinates; a series of these minimizations are carried out with the plane at different positions along the diffusion path. This method, which is computationally efficient, allowed a simple determination of the energy barrier. The calculations are carried out using the MINDO/3[10] SCF method which employs a STO-3G (slater type orbitals approximated by three Gaussian functions) minimal basis set. This method is chosen because the nature of the dimer-bond on Si(100)2x1 was successfully studied using the MINDO/3 parameterization[65] and also to test the underlying physics of hydrogen diffusion on the Si(100)2x1 surface, as discussed in section 5.6. However, other parameterizations can be used to model the reaction barriers for hydrogen diffusion, such as PM3, AM1, SAM1, MNDO-d.

5.5 Results and Discussion

Figure 5.8 to 5.10 illustrates the change in the total energy for intrarow, intradimer and interrow diffusion respectively. One would expect the energy curves would have a well-rounded peak, as Bowler et al[27] found using LDA. However, it appears that the barrier calculated for the intradimer diffusion has a cusp at the peak. Normally, this is an indication that the reaction coordinate chosen is incorrect; here, however, it is an artefact caused by the anomalous repulsion at long bond lengths. This makes sense, because the hydrogen atom is bonded to two silicon atoms at this point, which is not an energetically favourable state for a $1s^1$ atom to be in. Bowler et al[27] also found

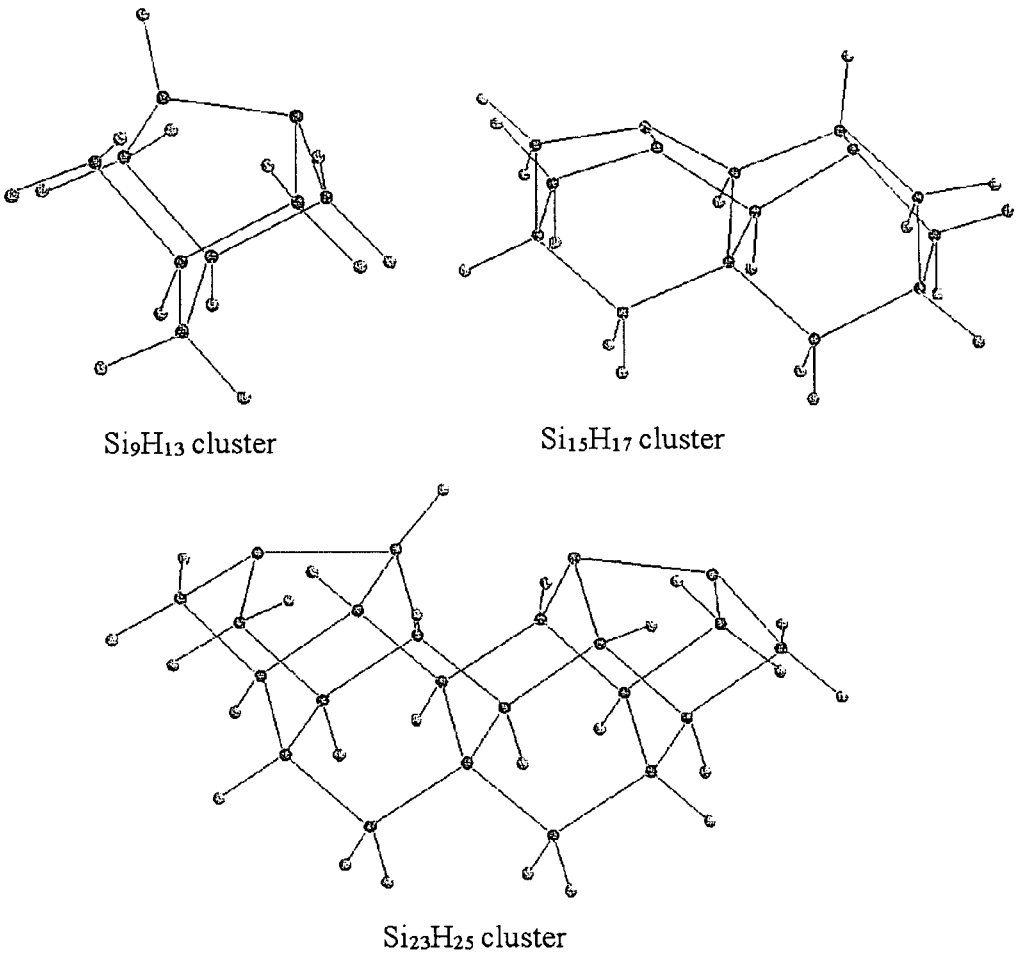

Figure 5.7: The Si_9H_{13}, $Si_{15}H_{17}$ and $Si_{23}H_{25}$ clusters (side view) used to model intradimer, intrarow, and interrow H atom diffusion barriers, respectively. The dark and light grey circles represent the silicon and hydrogen atoms respectively. In the transition state structures, the H atom is located midway between the two silicon atoms. The distances between the two Si atoms in the surface dimer closest to the H atom in the transition state are 2.43 Å, 2.38 Å and 2.4 Å respectively. The distances of the surface H atom to the closest Si atom in the transition state is 1.82 Å, 1.94 Å and 2.82 Å, respectively.

that the barrier calculated with tightbinding for the intradimer diffusion has a cusp at the peak. They estimated the effect of the anomalous repulsion to be about 0.2 eV. However, it needs to be investigated whether the same explanation applies in this case.

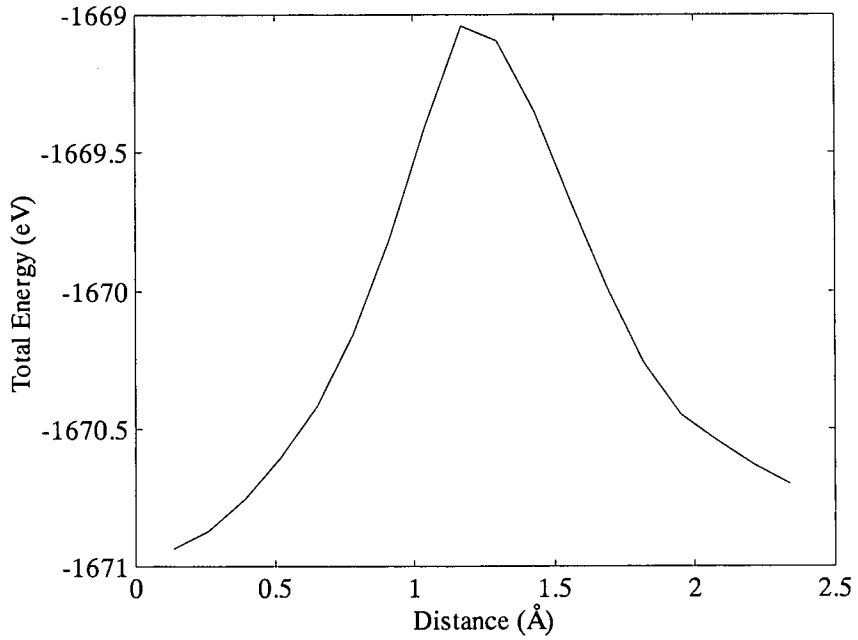

Figure 5.8: Modeling barriers of single-atom diffusion for intra-dimer diffusion using the Si_9H_{13} cluster. The diffusion barrier was determined to be 1.59 eV.

Table 5.1 summarizes the barrier heights obtained from our calculations and those of previous theoretical studies. As can be seen from Table 5.1, the semi-empirical results are in reasonable agreement with other theoretical studies, particularly the results for the BP-cluster which was a DFT study making use of the Becke-Perdrew functional. It also compares fairly well with the experimental results should the experimental error be taken into consideration. A lot of improvement on these results can still be done, for example, making use of larger clusters or even a slab. It could be argued that the use of the latter would lower the diffusion barriers by a small amount. More relaxation

	Intradimer kcal/mol	*Intrarow kcal/mol*	*Interrow kcal/mol*
Quadratic CI SD(T) Cluster (QCISD(T))[23]	1.90	–	–
Becke3-LYP-Cluster[23]	1.87	2.34	3.12
Becke Perdrew (BP) Cluster[23]	1.65	1.90	2.47
Local spin density (LSD) Cluster[23]	1.43	1.82	–
Local density approximation (LDA) Slab[22]	1.30	1.30	–
BP Slab[87]	1.39	1.39	–
General valence bond (GVB) CI Cluster[89]	2.50	1.99	2.69
Force Field Slab[25]	1.58 ± 0.96	2.72 ± 0.27	2.73
Tightbinding[27]	1.44	1.64	2.34
Classical many-body theory (MBT)[26]	1.10	1.80	–
Molecular Dynamics[91]	1.71	1.74	–
DFT Cluster[23]	1.70	2.30	–
Experiment[27]	1.39 ± 0.20	1.68 ± 0.20	–
Experiment[28]	1.00 ± 0.10	1.75 ± 0.05	–
Semi-empirical MINDO/3 (this study)	1.59	1.84	2.60

Table 5.1: Calculated barrier heights for intradimer, intrarow and interrow diffusion. For the QCISD(T) cluster there was no zero-point energy correction, whilst for the LSD cluster a value for the interrow diffusion is unknown since the author encountered difficulties in converging the calculations on the transition structure. The experimental measurements was done making use of hot STM atom tracking.[27],[28] Experimental results for interdimer diffusion are not available since these processes occurs extremely fast and are rarely seen. It is estimated, however, that the barrier should be around 2.4 eV.[27] The semi-empirical results refer to those obtained in this study.

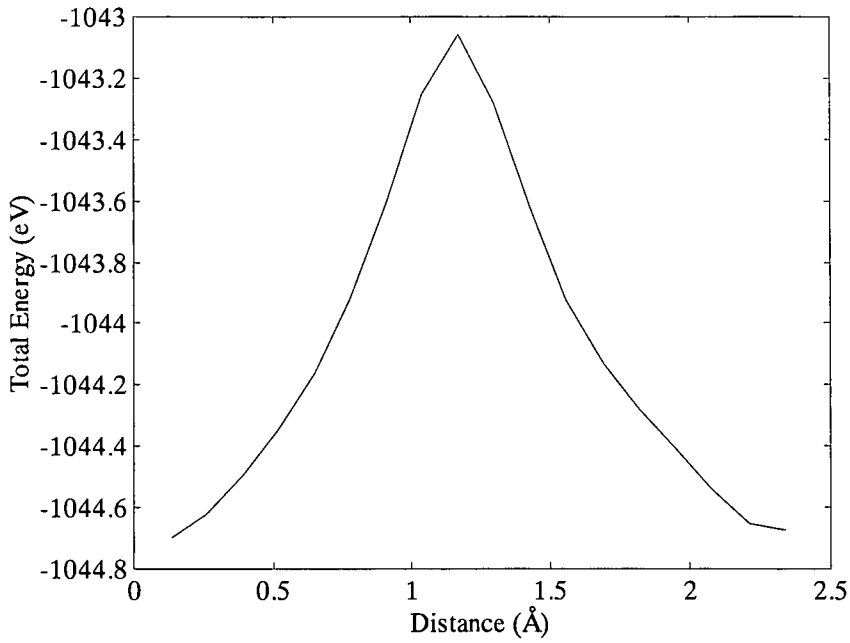

Figure 5.9: Modeling barriers of single-atom diffusion for intra-row diffusion using the $Si_{15}H_{17}$ cluster. The diffusion barrier was determined to be 1.84 eV.

of the cluster can also be included - in this study, only the surface and subsurface atoms were allowed to relax. Vittadini et al[87] estimated that the inclusion of surface relaxation for the Si_9H_{13} cluster results in an decrease of the diffusion barrier height by approximately 0.08 eV from that of a rigid cluster. Jordan et al[23] estimated the inclusion of relaxation for the $Si_{23}H_{25}$ causes the diffusion barrier height to decrease by approximately 0.7 eV. Since the surface and subsurface atoms were relaxed in this study, it can be concluded that the barrier height is not overestimated by such a large amount, though further investigation by means of inclusion of relaxation of the third and fourth layer atoms can clarify this.

It is necessary to emphasize that an intensive study of the diffusion of hydrogen on Si(100)2x1 was not made: the study extended in so far as to gain insight into the system, to validate that MOPAC can be used with success to model this type of process and to use the knowledge gained for further development.

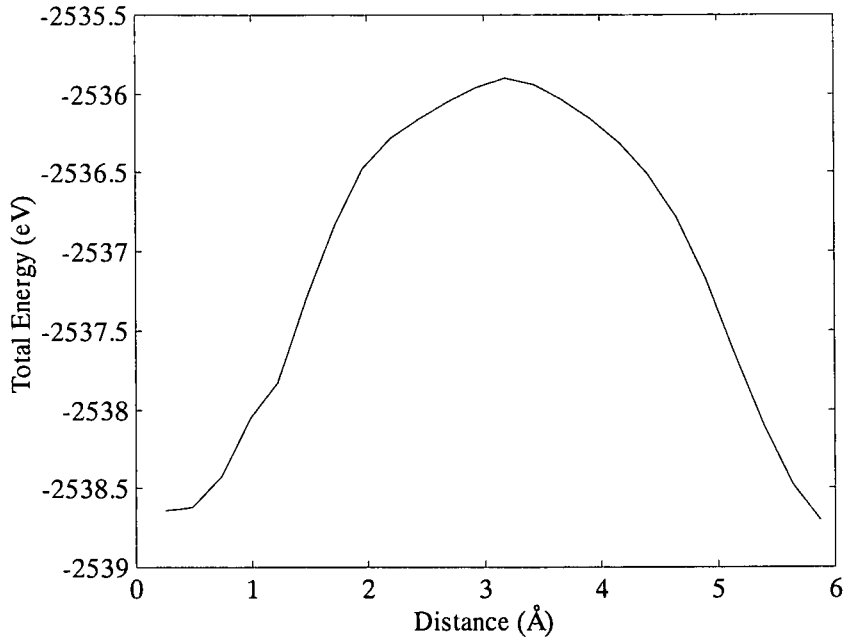

Figure 5.10: Modeling barriers of single-atom diffusion for interrow diffusion using the $Si_{23}H_{25}$ cluster. The diffusion barrier was determined to be 2.60 eV.

5.6 Deviation from Born-Oppenheimer energy

In the previous section it was shown that MOPAC can be successfully used to calculate the diffusion barriers of hydrogen on the Si(100)2x1 surface. It can thus be concluded that MOPAC is sufficient in describing the energy surface underlying the dynamics of hydrogen on the Si(100)2x1 surface.

Considering the future aim of combining fictitious dynamics with atomic dynamics, it is necessary to investigate, for situations of changing atomic geometry, how close the fictitious dynamics keep the electronic energy to the ground state.

Consider figure 5.11. First the ground state density matrix for hydrogen at position 1, in the same plane as the surface dimer, is calculated using the fictitious dynamics method (thus a sufficient number of fictitious time steps is used so that energy convergence is

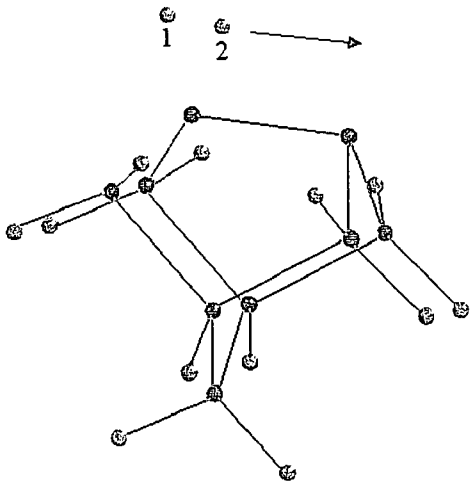

Figure 5.11: The Si_9H_{12} cluster was used as a "mini"-Silicon surface in order to investigate the deviation from the Born-Oppenheimer energy for situations of changing atomic geometry, such as the diffusion of hydrogen on the Si(100)2x1 surface.

reached). Using this density matrix as starting density matrix for position 2, also in the same plane as the surface dimer and in line with position 1, only one fictitious time step is made and the effect on the electronic energy is noted. This resulting density matrix is then used as starting density matrix for the next position and so forth.

In figure 5.12 the deviation from the Born-Oppenheimer energy is illustrated. Ideally, only one fictitious time step should be required to keep the calculated energy close enough to the Born-Oppenheimer energy. However, as illustrated in figure 5.12, it can be seen that one fictitious time step is insufficient to do so whilst three fictitious time steps seem to be necessary to keep the calculated energy close enough to the Born-Oppenheimer energy. A possible strategy to increase the ability of the fictitious electron dynamics method to keep the calculated energy closer to the Born-Oppenheimer energy can be to take smaller "atomic" steps. How small such an "atomic" step must be so that only one fictitious time step is sufficient to do the above, needs further investigation however.

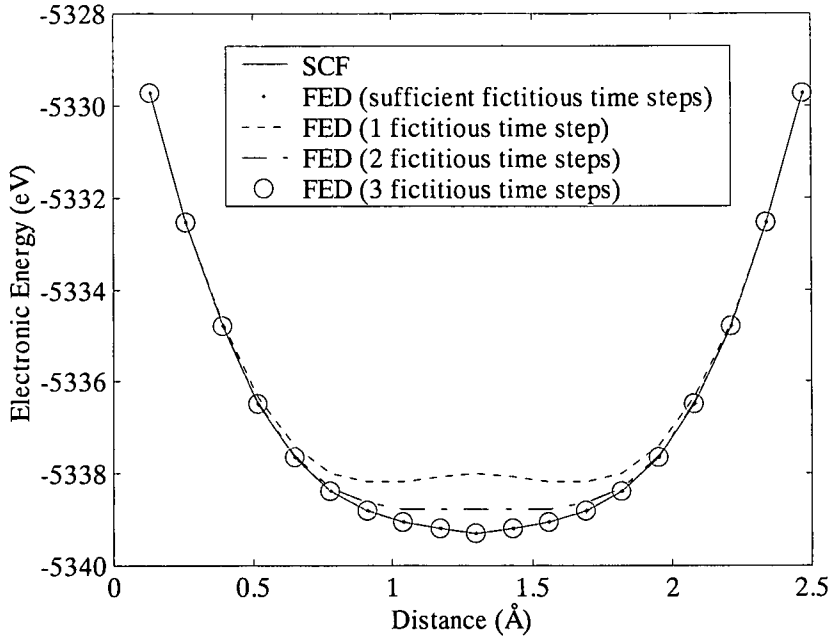

Figure 5.12: Deviation from Born-Oppenheimer energy. The abbreviation FED refers to calculations done with the fictitious dynamics method. A fictitious mass parameter, b', of 0.1 was used. The results are discussed in the text.

From the above discussion it can thus be concluded that the ability of the fictitious dynamics method to keep the calculated energy close to the Born-Oppenheimer energy, opens up the possibility of successfully combining this method with atomic dynamics, although some extra work is required in this regard. It is worth mentioning that the electronic energy calculated using SCF and fictitious dynamics as hydrogen moves from one end of the dimer to another, are in excellent agreement.

Chapter 6

Future Development

In this thesis the primary emphasis was on the implementation and evaluation of the success of the fictitious dynamics method in a semi-empirical environment, as well as the optimization of the parameters of importance. In the next sections the further development of this method will be elaborated on which include the potential combination with atomic dynamics as well as the proposed endeavor to reach $O(N)$ scaling with system size.

6.1 Combination of atomic dynamics with fictitious dynamics

A brief introduction into molecular dynamics was given in section 2.7. It was mentioned that the only tractable prescriptions for the atomic interactions in molecular dynamics are potential models. Although the effects of changes in bonding may be taken into account in an average way so that, for example, phase transitions may be approximately treated, molecular dynamics are insufficient in describing molecular reactions at surfaces.

The idea of ab initio molecular dynamics was also discussed in section 2.8, as well as the unified approach to electronic structure and molecular dynamics by Car and Parinello.[50] This method is efficient enough to be used for the many evaluations of energy and forces which are needed for molecular dynamics. Thus, the molecular dynamics technique can be used even when chemical bonding is changing.

The attractiveness of combining fictitious dynamics with molecular dynamics lies in the following reasons. Firstly, it provides an unbiased way to both sample phase space (to calculate statistical-mechanical quantities) or to explore configuration space (finding the best geometry, a simulated annealing approach). Secondly, it yields time-dependent quantities. For the first reason, because of the small number of degrees of freedom and the low energy barriers encountered, the configuration space of a small molecule on a surface is readily sampled by quite short simulations. This means that adsorption geometries can be found in an automatic way, rather than by guessing the most favourable configuration(s) and relaxing them to mechanical equilibrium as is done in Monte Carlo methods. For the second reason, transport coefficients, including the thermal conductivity and bulk and shear viscosities, are intimately related to the dynamics of the atoms, a fact which is evident through time-correlation expressions of these coefficients.

The combination of fictitious dynamics with atomic dynamics can thus ultimately lead to a successful description of, for example, the dynamics of hydrogen on the Si(100)2x1 surface, which include the adsorption, diffusion and desorption from the surface. Since an introductory study of this system has already been undertaken, the prospect of further investigation using the combined method is promising.

6.2 Combined quantum and molecular mechanics schemes

The modeling of large, complex chemical systems is still a daunting challenge. This has compelled the development of the so-called hybrid QM/MD (combined quantum mechanics and molecular mechanics) schemes,[94] where quantum mechanics are used to describe the region in a system where chemical reactions occur, whereas molecular dynamics are used to describe the rest of the system. The study of large biological systems are thus possible using a QM/MD approach.

The combination of quantum mechanics with molecular dynamics is an important developing area of research in ab initio molecular dynamics, as discussed in section 2.8. However, semi-empirical methods can also be combined with empirical potentials as laid out in a seminal paper by Levitt and Warshel.[94]

To see how this can be the case, the potential energy (in semi-empirical methods such as those employed in MOPAC) associated with two atoms within a larger system containing N atoms can be expressed as[10]

$$U(\mathbf{R}_1, ..., \mathbf{R}_N) = \frac{1}{2} \sum_{AB} V_{AB}(R_{AB}, \{P_{\lambda\sigma}\}) \, . \tag{6.1}$$

Thus, the energy of the system can be written in a "classical" form as a sum of pair potentials. However, the potential energy, associated with atoms A and B with relative position vector $\mathbf{R}_{AB}$, depends strongly on the electronic state represented by the density matrix elements, $P_{\lambda\sigma}$. It thus follows that the environmental dependence of the potential energy is contained within the density matrix.

In light of the discussion in the previous paragraph, it can thus be proposed that by the very nature of the semi-empirical methods employed in this study, promise towards the strategy of combining these methods with empirical potentials, can be offered.

6.3 Achievement of $O(N)$ scaling

6.3.1 Locality in quantum mechanics

Locality in quantum mechanics means that the properties of a certain observation region comprising one or a few atoms are only weakly influenced by factors that are spatially far away from this observation region. This fundamental characteristic of insulators is well established within independent-electron theories and it can even be carried over into the many-electron framework.[13] Traditional chemistry is based on local concepts. Covalently bonded materials are described in terms of bonds and lone electron pairs. It is standard textbook knowledge that the properties of a bond are mainly determined by its immediate neighbourhood. The decisive factors are what type of atoms and how many of them (the coordination number) are surrounding it. Second-nearest neighbours and other more distant atoms have a very small influence. As an example, the total energy of a hydrocarbon chain molecule C_nH_{2n+1} can be considered.

In this case each CH_2 sub-unit is, from an energetical point of view, practically an independent unit. As one adds one CH_2 sub-unit, the energy increases by an amount that is nearly independent of the chain length. Already the insertion of a CH_2 sub-unit into the smallest chain C_2H_6 gives an energy gain that agrees within 10^{-4} atomic units with the asymptotic value of the insertion energy for very long chains. This means that the electrons belonging to this inserted sub-unit no longer "see" the end of the chain for very short chain lengths. This example is a drastic illustration of a principle sometimes termed "nearsightedness".[13]

In other insulating materials the influence of the neighbouring atoms decays more slowly. An example is shown in figure 6.1, where the total energy per silicon atom is plotted as a function of the size of its crystalline environment. Even in metallic systems, where the elementary bond concept is no longer valid, locality still exists. This is supported by the well-known fact that the total charge density in a metal is given with reasonable

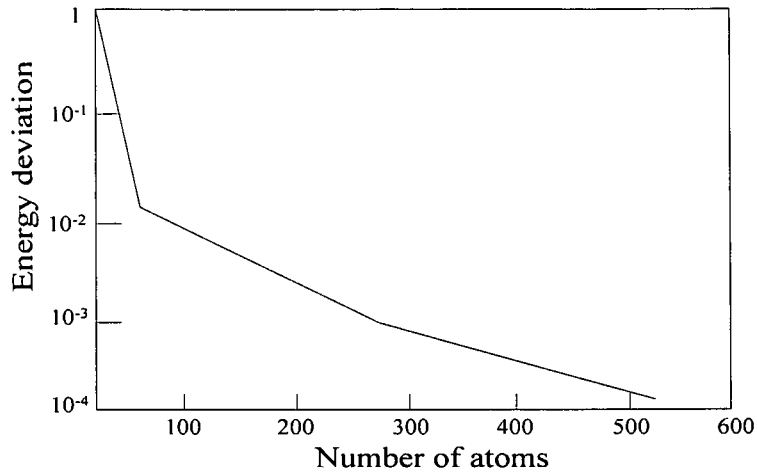

Figure 6.1: The deviation of the total energy per silicon atom from its asymptotic bulk value as a function of the size of the periodic volume in which its embedded. The calculation was done with a tight binding scheme using exact diagonalization[38].

accuracy by the superposition of the atomic charge densities. Since atomic charge densities decay rapidly, this implies that the charge density at the midpoint between two neighbouring atoms is mainly determined by the two closest atoms and influenced very little by other more distant atoms.

To give more substance to the above discussion, the one-particle density matrix from equation (4.18) is written as

$$\mathbf{P}'(\mathbf{r},\mathbf{r}') = \langle \mathbf{r}|\,\hat{\rho}\,|\mathbf{r}'\rangle \ , \tag{6.2}$$

where $\hat{\rho}$ is the density operator given by

$$\hat{\rho} = \sum_{n}^{occ} |\psi_n\rangle\,\langle\psi_n| \tag{6.3}$$

and $|\psi_n\rangle$ is the molecular or crystal orbital.

Using the closure relation of quantum mechanics,[46] equation (6.2) can be transformed to the atomic orbital representation

$$\begin{aligned} P'_{\lambda\sigma} &= \langle\phi_\lambda|\,\hat{\rho}\,|\phi_\sigma\rangle \\ &= \int d\mathbf{r}\int d\mathbf{r}'\,\langle\phi_\lambda\mid\mathbf{r}\rangle\,\langle\mathbf{r}|\,\hat{\rho}\,|\mathbf{r}'\rangle\,\langle\mathbf{r}\mid\phi_\sigma\rangle \\ &= \int d\mathbf{r}\int d\mathbf{r}'\phi^*_\lambda(r)P(\mathbf{r},\mathbf{r}')\phi_\sigma(\mathbf{r}') \end{aligned} \tag{6.4}$$

with $P'_{\lambda\sigma}$ the matrix elements of the density operator ρ in the representation of a set of orthonormal atomic orbitals $\{\phi_\lambda\}$.

Consider now the orbitals ϕ_λ and ϕ_σ which are localized around $\mathbf{r}$ and $\mathbf{r}_\sigma$ respectively. The principle of nearsightedness implies that, if $|\mathbf{r}-\mathbf{r}'| > \mathbf{r}_c$, with $\mathbf{r}_c$ a certain cut-off distance, $\mathbf{P}'(\mathbf{r},\mathbf{r}') \approx 0$. To interpret the implication of this principle on the elements of the density matrix, consider the following:

If $\mathbf{r}$ is close to $\mathbf{r}_\lambda$ and $\mathbf{r}'$ is far from $\mathbf{r}_\sigma$ and $|\mathbf{r}-\mathbf{r}'| < \mathbf{r}_c$, this implies that ϕ_λ and $\mathbf{P}'(\mathbf{r},\mathbf{r}') \neq 0$ and $\phi_\sigma \approx 0$. Similarly, in the case where $\mathbf{r}$ is far from $\mathbf{r}_\lambda$ and $\mathbf{r}'$ is close to $\mathbf{r}_\sigma$ and $|\mathbf{r}-\mathbf{r}'| < \mathbf{r}_c$, $\phi_\lambda \approx 0$. For the case where $\mathbf{r}$ is close to $\mathbf{r}_\lambda$ and $\mathbf{r}'$ is close to $\mathbf{r}_\sigma$, $|\mathbf{r}-\mathbf{r}'| > \mathbf{r}_c$ thus $\mathbf{P}'(\mathbf{r},\mathbf{r}') \approx 0$. Even though the integration is performed over the whole volume $d\mathbf{r}d\mathbf{r}'$, the integrant will always contain at least one very small factor, except if $|\mathbf{r}_\lambda - \mathbf{r}_\sigma| \leq \mathbf{r}_c$.

For insulators the density matrix decays exponentially with distance[95] i.e

$$P'_{\lambda\sigma} \propto e^{-k|\mathbf{r}_\lambda - \mathbf{r}'_\sigma|}\,. \tag{6.5}$$

From the above equation it can be seen that, if $|\mathbf{r}_\lambda - \mathbf{r}_\sigma|$ are small, then $e^{-k|\mathbf{r}_\lambda-\mathbf{r}'_\sigma|} \approx 1$. If $|\mathbf{r}_\lambda - \mathbf{r}_\sigma|$ are very large, then $e^{-k|\mathbf{r}_\lambda-\mathbf{r}'_\sigma|} \approx 0$. This then produces a sparse density matrix which can be taken advantage of to achieve $O(N)$ scaling.

6.3.2 Basic strategies for $O(N)$ scaling

Most $O(N)$ algorithms are built around the density matrix or its representation in terms of Wannier functions and take advantage of its decay properties. To obtain

linear scaling, the exponentially decaying quantities has to be cut of when they are small enough. This introduces the concept of a localization region. Only inside this localization region is the quantity calculated; outside it is assumed to vanish. For simplicity the localization region is usually taken to be a sphere. Usually the boundary of the localization region can be defined by a geometric distance criterion.

It is clear that one can gain significant computational efficiency only if the extent of the system is larger than the size of the localization region. The crossover point depends therefore on the decay properties of the density matrix of the system. However, it also depends on the dimensionality of the system. For a linear-chain molecule with a large band gap, it might be enough to have a localization region containing just two neighbouring atoms on each side, hence just five atoms, and for systems larger than five atoms one might potentially gain computational efficiency by using an $O(N)$ method. If one had a three-dimensional system with a comparable gap, then a spherical localization region extending out to the second neighbours would contain some sixty atoms and the crossover point would already be much larger. For a system with a small gap, such as silicon, or for metallic systems the crossover point would be even larger.

The reason why the crossover point would be much larger for systems with a small bandgap can be sought in the expression for the exponential decay of the density matrix given by equation (6.5). Some simple predictions for the asymptotic decay of the density matrix are available. Ismaj et al predicted that, for insulators in the weak binding limit, the exponential will decay with rate[97]

$$\gamma = \left(\frac{aE_G m_e}{\hbar^2}\right) ,$$

with a an arbitrary constant, E_G the bandgap energy, m_e the electron mass and $\hbar$ Planck's constant. Kohn, on the other hand, suggested that in the tight binding limit this exponential decay rate is given by[96]

$$\gamma = \left(\frac{2E_G m_e}{\hbar^2}\right)^{\frac{1}{2}} .$$

Both of these equations express the relationship between the bandgap energy, E_G,

and the decay rate, γ. Consider two elements, Si with $E_G = 1.1$ eV and C with $E_G = 5.5$ eV. Since Si has a smaller bandgap than C, the decay rate, γ for Si will be approximately a factor $\sqrt{5}$ smaller than that for C in the case of the tight binding limit. Thus, $e^{-\gamma|\mathbf{r}_\lambda - \mathbf{r}_\sigma|}$ and consequently $P_{\lambda\sigma}$ will decay more slowly for Si than for C, resulting in a larger localization region for Si. This then explains why systems with small bandgaps need to be must larger than systems with large bandgaps to benefit from an $O(N)$ scaling method.

In metals the crossover point is even larger because, in this case, the density matrix does not decay exponentially, but rather as a power law. This can be seen from the expression for the density matrix of a free-electron gas at T=0[98]

$$\begin{aligned} \mathbf{P}'(\mathbf{r},\mathbf{r}') &= 2(2\pi)^{-3}\int_k d^3\mathbf{k}e^{-i\mathbf{k}(\mathbf{r}-\mathbf{r}')} \\ &= \frac{3n[\sin(\zeta)-\zeta\cos(\zeta)]}{\zeta^3}, \end{aligned}$$

with $\zeta = \mathbf{k}|\mathbf{r}-\mathbf{r}'|$, $\mathbf{k}$ the Fermi wave vector and n the electron density. The density matrix of a free-electron system has a long range at $T = 0$ K, following from the fact that, if $\zeta \to \infty$, $\mathbf{P}' \propto \zeta^{-2}$. The localization region for metals is thus much larger than for insulators which explains the larger crossover point.

From the above discussion it then follows that the localization property of the density matrix opens up the possibility to develop the fictitious dynamics method to such extend that $O(N)$ scaling can be achieved.

6.3.3 Number of linearly independent parameters in the density matrix

Another potential method of reducing the computational time, is by considering the number of linearly independent parameters in the density matrix. These parameters are sufficient to describe the orientation of the subspace V_P and thus only these parameters can be used during calculations. Although the use of the linear independent parameters

will not necessarily reduce the computational time from, say $O(N^2)$ to $O(N)$, it will influence the prefactor of the scaling time.

Consider now a density or projection matrix of rank M for an M-dimensional subspace, V_P in an N-dimensional vector space, V. The number of independent parameters, N_P, in $\mathbf{P}$ is also the minimum number of parameters needed to describe the orientation of the subspace V_P. V_P can be characterized by a set of M orthonormal basis vectors contained in V_P. In order to describe these M basis vectors in terms of N arbitrary basis vectors in V, MN parameters are needed that are connected by $\frac{M}{2}(M+1)$ independent orthonormality constraints. This then leaves $N'_P = MN - \frac{M}{2}(M+1)$ independent parameters that are needed to specify the orthonormal basis for V_P.

Now, the orientation of this orthonormal basis is completely arbitrary within V_P. It can be described by an $(M \times M)$ orthogonal basis transformation in terms of some standard basis in V_P. Such an orthogonal transformation contains $N_\sigma = \frac{M}{2}(M-1)$ independent parameters. This number represents the redundant information contained in the N'_P parameters. Thus, the number of independent parameters that are exactly needed to uniquely specify the orientation of the subspace V_P, are thus also the number of independent parameters in $\mathbf{P}$, which is given by

$$N_P = N'_P - N_\sigma = M(N-M) \tag{6.6}$$

The proof for equation (6.6) is given in Appendix B. This number is a maximum when $M = \frac{N}{2}$. This situation occurs, for example, in the case of a minimal basis set calculation on a closed shell system where the number of occupied orbitals is one half the even number of valence orbital basis functions as discussed in this thesis.

Typically density matrix methods focus on calculating all N^2 elements of the matrix which is up to four times more than the number of independent elements (for the case where $M = \frac{N}{2}$). Various fascinating relationships between submatrices of the density matrix can be obtained that offers potential of focusing on a subset of elements of $\mathbf{P}$,

still containing all information on the orientation of the subspace, V_p. However, much more work has to be done in this regard.

6.4 Use of method in situations where SCF fails

The attractiveness of the use of the fictitious dynamics method in situations where standard selfconsistent field calculations fail to converge, makes it worthy for further investigation. Not having the selfconsistent results as comparitave medium, some other means are required for testing whether results obtained with the fictitious dynamics method is indeed correct. A $\underline{\mathbf{C}}$-matrix can be obtained from the projection matrix by means of QR-factorization as discussed in the next section. This can then be used to determine whether the results acquired from the fictitious dynamics calculation are indeed correct by focusing on the self-consistency of $\underline{\mathbf{C}}$ and $\mathbf{F}(\underline{\mathbf{C}})$.

6.4.1 QR-factorization of a column submatrix of the density matrix

Let $\mathbf{P}'_M$ contain M linearly independent columns from the density matrix, $\mathbf{P}$, where $\rho(\mathbf{P}) = M$. For simplicity of notation, it is assumed to be the first M columns contained in $\mathbf{P}_M$.

The calculation of the complete density matrix, $\mathbf{P}$, in terms of $\mathbf{P}_M$ is set as a problem. This can in principle be done through the equation, $\mathbf{P} = \mathbf{P}_M(\tilde{\mathbf{P}}_M\mathbf{P}_M)^{-1}\tilde{\mathbf{P}}_M$.

However, calculating the inverse in this expression is not an attractive option. In this section, a straightforward method, based on the QR-factorization of $\mathbf{P}_M$, is described.

$\mathbf{P}$ can be factorized as $\mathbf{P} = \underline{\mathbf{C}}\tilde{\underline{\mathbf{C}}}$, where $\underline{\mathbf{C}}$ contains any set of M orthonormal vectors $\{\underline{\mathbf{c}}_i\}$ in the space $V_{\mathbf{P}}$, say $\underline{\mathbf{C}} = [\underline{\mathbf{c}}_1....\underline{\mathbf{c}}_M]$. Thus $\tilde{\underline{\mathbf{C}}}\,\underline{\mathbf{C}} = \mathbf{I}$.

Now, let $\underline{\mathbf{C}} = \begin{pmatrix} \underline{\mathbf{C}}_M \\ \underline{\mathbf{C}}_{N-M} \end{pmatrix}$. It follows that $\underline{\mathbf{P}}_M = \underline{\mathbf{C}}\tilde{\mathbf{C}}_M$. In the more general case, where $\mathbf{P}'_M$ contain any set of M linearly independent, columns from $\mathbf{P}$, we have $\mathbf{P}'_M = \underline{\mathbf{C}}\tilde{\mathbf{C}}'_M$, where $\mathbf{C}'_M$ contains M linearly independent rows from $\underline{\mathbf{C}}$ as its columns.

A "special set" of orthonormal vectors in $V_{\mathbf{P}}$, say $\{\mathbf{q}_1, ..., \mathbf{q}_M\}$, can be considered, obtained from $\underline{\mathbf{P}}_M$ by means of the QR-factorization,

$$\mathbf{P}_M = \underline{\mathbf{Q}}\mathbf{R}\,, \tag{6.7}$$

where $\underline{\mathbf{Q}} = [\underline{\mathbf{q}}_1 .... \underline{\mathbf{q}}_M]$. $\mathbf{R}$ is a non-singular upper-triangular matrix. $\mathbf{R}^{-1}$ is also an upper-triangular matrix. It also follows that $\underline{\mathbf{Q}} = \underline{\mathbf{P}}_M\mathbf{R}^{-1}$.

Since the columns of $\underline{\mathbf{Q}}$ are orthonormal vectors in $V_{\mathbf{P}}$, it follows that

$$\mathbf{P} = \underline{\mathbf{Q}}\,\tilde{\mathbf{Q}}\,. \tag{6.8}$$

Now, let $\underline{\mathbf{Q}} = \begin{pmatrix} \mathbf{Q}_1 \\ \underline{\mathbf{Q}}_2 \end{pmatrix}$, where $\mathbf{Q}_1 \epsilon \mathcal{M}_{M,M}$ and $\underline{\mathbf{Q}}_2 \epsilon \mathcal{M}_{N-M,M}$. From equation (6.8) it follows that

$$\underline{\mathbf{P}}_M = \underline{\mathbf{Q}}\tilde{\mathbf{Q}}_1\,. \tag{6.9}$$

$\tilde{\mathbf{Q}}_1$ contains as columns M linearly independent rows from $\underline{\mathbf{Q}}$. From equations (6.7) and (6.9) it is obtained that $\underline{\mathbf{Q}}(\mathbf{R} - \tilde{\mathbf{Q}}_1) = \underline{\mathbf{0}}$, and by multiplying from the left with $\tilde{\underline{\mathbf{Q}}}$, it follows that $\mathbf{R} = \tilde{\underline{\mathbf{Q}}}$, in accordance with the uniqueness of the QR-factorization. It can be concluded that $\tilde{\underline{\mathbf{Q}}}$, should be upper-triangular. The diagonal elements of $\underline{\mathbf{Q}}$, are its eigenvalues.

Now, let

$$\underline{\mathbf{P}}_M = \begin{pmatrix} \mathbf{A} \\ \underline{\mathbf{H}} \end{pmatrix}. \tag{6.10}$$

It follows from equations (6.9) and (6.10) that

$$\mathbf{A} = \mathbf{Q}_1\tilde{\mathbf{Q}}_1\,, \tag{6.11}$$

and

$$\underline{\mathbf{H}} = \underline{\mathbf{Q}}_2 \tilde{\mathbf{Q}}_1 \,. \tag{6.12}$$

Since $\underline{\tilde{\mathbf{Q}}}$, is lower-triangular and $\tilde{\mathbf{Q}}_1$ is upper-triangular, equation (6.11) corresponds to the Choleski decomposition of the symmetric matrix $\mathbf{A}$. This decomposition exists if $\mathbf{A}$ is a non-singular matrix. Hoffman[21] shows that a non-singular principle submatrix is a necessary and sufficient condition for the associated columns, contained in $\underline{\mathbf{P}}_M$, to be linearly independent.

It is clear from the above analysis, that calculating the QR-factorization of the column submatrix, $\underline{\mathbf{P}}_M$ amounts to a large extent to the fast Choleski factorization of the symmetric principle submatrix $\mathbf{A}$ in $\mathbf{P}_M$.

A special property of the projection matrix indeed opens up the possibility of an even faster Choleski factorization of $\mathbf{A}$ than for an arbitrary symmetric matrix. Of relevance is the fact that the diagonal elements of $\mathbf{P}$ is already known to be the norm square of the associated column i.e $P_{ii} = \sum_k P_{ki}^2$.

6.5 Reducing computational time

The RATTLE and McWeeny algorithm as discussed in section 4.2 scale as $O(N^2)$. The integration of the equations of motion are solely matrix multiplication which scales as $O(N^3)$. By using more efficient programming strategies, such as replacing the matrices with arrays and using only the lower triangle and diagonal of the matrices (since they are symmetric), the computational time can be significantly decreased, not necessarily the scaling factor, but most definitely the scaling prefactor.

Another matter that will receive a fair amount of attention in future is the storage requirements during a fictitious electron dynamics run. Thus far, the one and two electron arrays calculated by MOPAC are written to output files which are then used by the routines performing the fictitious electron dynamics. Some strategies concerning

reducing the amount of storage required during a run thus need to be developed.

Another possibility of reducing computational time is by making appropriate simplifications of the equations of motion to reduce the number of multiplications needed. For example, including only the most significant constraint effects. In this way an optimal synergy between an approximate accommodation of constraints in the equation of motion and the subsequent refining action of the McWeeny and RATTLE algorithms can be sought.

6.6 Further optimization of parameters

Although the parameters for the equations of motion have been extensively investigated, some extra work is required in this regard. For example, consider the size of the time step used in the fictitious dynamics calculation. From section 4.5 it became apparent that, in order for the total energy of a system to stay conserved, a fictitious mass parameter of roughly 10^{-6} should be used. A possible strategy for increasing the fictitious mass parameter can be as follows: Firstly, a certain error tolerance in the total energy can be specified. The simulation can then be started for a larger fictitious mass parameter than 10^{-6}, say 10^{-3}. The fictitious mass parameter can then be appropriately scaled to meet the specified error tolerance by an ad hoc factor[93]

$$\sqrt{1 + \frac{\Delta E_0 - \Delta E}{\tau \Delta E}},$$

with ΔE_0 and ΔE the target and observed errors in the total energy and τ the response time in units of the fictitious dynamics time step size.

It is also necessary to test as many systems as possible to ascertain the scope of systems for which the choice of parameters can be applied to.

Chapter 7

Conclusion

In section 3.3.1 a review of a theory of idempotency conserving changes for the density matrix was given. The most important result from this section was the general form $\Delta\mathbf{P} = (\mathbf{I} - 2\mathbf{P})[\mathbf{X}, \mathbf{P}]$, for a first order idempotency conserving change with $\mathbf{X}$ an arbitrary, but appropriately scaled matrix. It was also emphasized that $\Delta\mathbf{P}$ is the projection of $\mathbf{X}$ onto the first order idempotency conserving subspace, $\mathcal{M}_{(2;3)}$ of $\mathcal{M}_{N,N}$, that is, $\Delta\mathbf{P} = P_{\mathcal{M}_{(2;3)}}\mathbf{X}$.

In section 3.3.2 a geometric interpretation of the difference between first order and exact idempotency conserving conditions was given. This was done by considering necessary, though not sufficient conditions for idempotency conservation.

The theory of idempotency conserving changes presented in section 3.3.1, provided a sturdy foundation for sections 3.4.1 and 3.4.2. In these sections the implications of the idempotency constraints on the first order and second order time derivatives of the density matrix were considered. General forms of the first and second order equations of motion for the density matrix that implicitly ensure conservation of idempotency were also obtained and are given by equations (3.38) and (3.49).

In sections 3.5 and 3.6 physical content was added to the second order equation of motion. It was concluded that the first order idempotency conserving change, generated

by the Fock matrix, corresponds to the steepest descent change. An appropriate choice for the parameter, b, which figures in the equations of motion, was also made. This parameter is related to a fictitious mass for the electronic degrees of freedom.

In section 3.7 the connection between the $\mathbf{P}$ and related $\underline{\mathbf{C}}$-dynamics was shown. and an equation of motion, that conserves the orthonormality of the columns of $\underline{\mathbf{C}}$ was obtained in equation (3.83). It was also noted that this equation corresponds, at equilibrium, to the standard eigenvalue problem for the Fock matrix.

Chapter 4 focused on the implementation of the equations of motion. In section 4.1.2 the integration of these equations by means of the velocity Verlet scheme was discussed. It was mentioned that, although the equations of motion conserve idempotency and orthonormality of the projection and configuration matrix respectively, the use of a numerical integration technique will cause deviation from idempotency and orthonormality. In sections 4.2.1.1 and 4.2.2.1 two algorithms, the McWeeny purification and RATTLE algorithm, were discussed that would be sufficient to enforce idempotency and orthonormality on the projection and configuration matrix during integration.

Special attention was given in section 4.3 to the optimization of parameters in the equations of motion, as this will ensure efficient operation of these equations. In sections 4.3.1 and 4.3.2 the effect that the constraints has on the energy convergence was discussed. It was concluded that an idempotency and orthonormality tolerance of 10^{-4} would be sufficient to guarantee that the correct ground state energy is reached. In sections 4.3.3 and 4.3.4 the fictitious mass parameter, b', which is related to the fictitious mass through $b' = \frac{1}{2}b\delta t^2 = \frac{1}{2\mu}\delta t^2$, was under investigation. From these sections it was concluded that, firstly, the number of fictitious time steps necessary for energy convergence are dependent on the fictitious mass parameter, b', and not b and δt individually. Secondly, it was concluded that, should fast energy convergence be required, a fictitious mass parameter of about 0.1 for the $\mathbf{P}$-dynamics and 0.01 for the $\underline{\mathbf{C}}$-dynamics would suffice. However, it was mentioned in section 4.3.3 that these choices for b' would not necessarily conserve the total energy. In section 4.5 it was concluded that, to ensure

energy conservation, a fictitious mass parameter of about 10^{-6} should be used for both the **P** and $\underline{\mathbf{C}}$-dynamics, depending on the accuracy required as well as on the duration of the simulation. This value for the fictitious mass parameter also ensured the time-reversibility of the velocity Verlet integrator. It was noted that smaller values for b' can be used, however this will depend on the computational resources available since this will result in much longer energy convergence. The role of the velocity quenching in the non-conservation of energy for large values of b' was also discussed. It is however clear that the extent to which energy should be conserved can only be clarified when the fictitious dynamics are combined with atomic dynamics.

An important conclusion from chapter 4 is how the **P**-dynamics and $\underline{\mathbf{C}}$-dynamics weigh up against each other. From figures 4.8 - 4.11 it is evident that for $b' = 0.01$ energy convergence is reached much faster for the **P**-dynamics than for the $\underline{\mathbf{C}}$-dynamics. For the **P**-dynamics energy convergence was reached in about 20-30 fictitious time steps whereas roughly 250 fictitious time steps was necessary for the $\underline{\mathbf{C}}$-dynamics. Another advantage of the **P**-dynamics over the $\underline{\mathbf{C}}$-dynamics is that the fictitious mass parameter, b', for the **P**-dynamics can be about a factor ten larger than b' for the $\underline{\mathbf{C}}$-dynamics as mentioned in section 4.3.3. However, use of the $\underline{\mathbf{C}}$-dynamics has its advantages - the time-reversibility is better than for **P**-dynamics as can be seen from figures 4.12 and 4.13. Figures 4.14 - 4.17 illustrates that energy is also better conserved for the $\underline{\mathbf{C}}$-dynamics.

In chapter 5, a brief overview of the Si(100)2x1:H system was given, together with the intent to study this system using MOPAC. The barriers for hydrogen atom diffusion, as determined by the MINDO/3 semi-empirical method, proved to be well in agreement with other theoretical, as well as experimental results. It was thus felt that the MINDO/3 method could be used to determine the electronic energy change as hydrogen diffuses over the silicon surface. This then served as comparative medium when the deviation from the Born-Oppenheimer energy during a fictitious dynamics run was investigated. A fictitious mass parameter, b', of 0.1 was used during the fictitious dy-

namics run. It was concluded that the fictitious dynamics method keeps the minimum energy fairly close to the Born-Oppenheimer energy - only three fictitious time steps are necessary to do so. However, it was mentioned that the intent is to reduce this value to only one fictitious time step.

It is believed that this study revealed the success of the fictitious dynamics method to calculate electronic structure. Some work remains to be done to develop it into a competative method to SCF calculations, especially for certain situations where SCF methods find it difficult or even fail to converge. The possibility of developing this density matrix based method to such an extent that $O(N)$ scaling can be achieved, together with the promise of combining it with atomic dynamics, makes it worthy for further study.

Appendix A

The Born-Oppenheimer approximation

If it is assumed that the nuclei and electrons are point masses and neglect spin-orbit and other relativistic interactions, then the molecular Hamiltonian is:

$$\hat{H} = -\frac{1}{2}\sum_i \nabla_i^2 - \sum_\alpha \frac{1}{2m_\alpha}\nabla_\alpha^2 - \sum_i \sum_\alpha \frac{Z_\alpha}{|\mathbf{r}_i - \mathbf{R}_\alpha|} + \frac{1}{2}\sum_i \sum_{j\neq i} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \frac{1}{2}\sum_\alpha \sum_{\beta\neq\alpha} \frac{Z_\alpha Z_\beta}{|\mathbf{R}_\alpha - \mathbf{R}_\beta|}\ , \tag{A.1}$$

in atomic units ($\hbar = m_e = \epsilon = 4\pi\epsilon_0 = 1$), where α and β refer to the nuclei with mass m_α and nuclear charges Z_α, Z_β and i and j refer to the electrons. The first and second terms in equation (A.1) corresponds to the kinetic energy operator for the electrons and nuclei respectively, the third term describes the attractions between electrons and nuclei, the fourth term describes the repulsions between electrons and the last term describes the repulsions between nuclei.

Wavefunctions and energies can be obtained by solving the Schrödinger wave equation:

$$\hat{H}\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_\alpha\}) = E\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_\alpha\})\ , \tag{A.2}$$

where $\{\mathbf{r_i}\}$ and $\{\mathbf{R}_\alpha\}$ symbolize the electronic and nuclear coordinates respectively. The molecular Hamiltonian is formidable enough to strike terror into the heart of any

quantum chemist/physicist; fortunately there exists a highly accurate and simplifying approximation. This approximation is simply a separation of the nuclear and atomic degrees of freedom for a system of atoms. The motivation behind such a decoupling is that the nuclei, being much more massive than the electrons, have smaller velocities. Thus, the electrons will relax to their groundstate configuration almost instantaneously on the timescale of their nuclear motion.

When the nuclei is now consider to be fixed, the nuclear kinetic energy terms can be omitted from equation (A.1) and obtain the Schrödinger wave equation for the electronic wavefunction:

$$(\hat{H}_{el} + V_{NN})\psi_{el}(\{\mathbf{r}_i\}, \{\mathbf{R}_\alpha\}) = U\psi_{el}(\{\mathbf{r}_i\}, \{\mathbf{R}_\alpha\}) \,, \tag{A.3}$$

where the *purely electronic Hamiltonian* $\hat{H}_{el}$ is given by:

$$\hat{H}_{el} = -\frac{1}{2}\sum_i \nabla_i^2 - \sum_i \sum_\alpha \frac{Z_\alpha}{|\mathbf{r}_i - \mathbf{R}_\alpha|} + \frac{1}{2}\sum_i \sum_{j \neq i} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} \,. \tag{A.4}$$

The electronic Hamiltonian including nuclear repulsion is $\hat{H} + V_{NN}$. The nuclear repulsion term V_{NN} is given by

$$V_{NN} = \sum_\alpha \sum_{\beta > \alpha} \frac{Z_\alpha Z_\beta}{|\mathbf{R}_\alpha - \mathbf{R}_\beta|} \,. \tag{A.5}$$

The energy U in equation (A.3) is the electronic energy including the energy of nuclear repulsion.

The variables in the electronic Schrödinger equation (A.3) are the electronic coordinates. The quantity $V_N N$ is independent of these coordinates and is a constant for a given nuclear configuration. Now, it can be easily proved that the omission of a constant term C from the Hamiltonian does not affect the wave functions, and simply decreases each energy eigenvalue by C. Hence if V_{NN} is omitted form equation (A.3)

$$\hat{H}_{el}\psi_{el} = E_{el}\psi_{el} \,, \tag{A.6}$$

where the *purely electronic energy* E_{el} is related to the electronic energy including internuclear repulsion by

$$U = E_{el} + V_{NN} \,. \tag{A.7}$$

Thus, the internuclear repulsion can be omitted from the electronic Schrödinger equation and simply added to E_{el} after solving equation (A.6).

Assuming now that the electronic Schrödinger equation has been solved, the nuclear motions can be considered next. As mentioned the assumption is made that the electrons move much faster than the nuclei; when the nuclei change their configuration slightly, say from $\mathbf{R}_\alpha$ to $\mathbf{R}'_\alpha$, the electrons adjust immediately to the change, with the electronic wave function changing form $\psi(\{\mathbf{r}_i, \mathbf{R}_\alpha\})$ to $\psi(\{\mathbf{r}_i, \mathbf{R}'_\alpha\}$) and the electronic energy changing from $U(\mathbf{R}_\alpha)$ to $U(\mathbf{R}'_\alpha)$. A schematic view of such a nuclear trajectory in given in figure (A.1). Thus, as the nuclei move, the electronic energy varies smoothly as a function of the parameters defining the nuclear configuration, and becomes, in effect, the potential energy for the nuclear motion. The electrons act like a spring connecting the nuclei; as the internuclear distance changes, the energy stored in the spring changes. Hence the Schrödinger equation for nuclear motion is

$$\hat{H}_N \phi_N(\{\mathbf{R}_\alpha\}) = E\phi_N(\{\mathbf{R}_\alpha\}) \,, \tag{A.8}$$

with

$$\hat{H}_N = -\frac{1}{2}\sum_i \nabla^2_\alpha + U(\mathbf{R}_\alpha) \,. \tag{A.9}$$

The variables in the nuclear Schrödinger equation are the nuclear coordinates, symbolized by $\mathbf{R}_\alpha$. The energy eigenvalue E in equation (A.8) is the total energy of the molecule, since the Hamiltonian given by equation (A.9) includes operators for both nuclear energy and electronic energy.

The approximation of separating electronic and nuclear motion is called the *Born-Oppenheimer approximation* and is basic to quantum chemistry. Born and Oppenheimer's mathematical treatment showed that the true molecular wave function is adequately approximated as

$$\Psi(\{\mathbf{r}_i, \mathbf{R}_\alpha\}) = \psi_{el}(\{\mathbf{r}_i, \mathbf{R}_\alpha\})\phi_N(\{\mathbf{R}_\alpha\}) \tag{A.10}$$

if $\left(\frac{m_e}{m_\alpha}\right)^{\frac{1}{4}} \ll 1$, with m_e the electron mass. In ab initio molecular dynamics (section 2.8), such as the Car-Parinello method, atomic dynamics are combined with fictitious

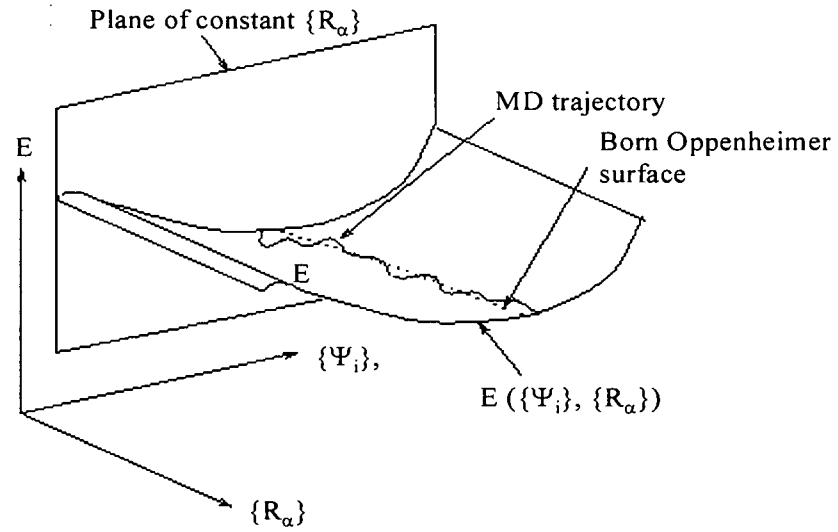

Figure A.1: A schematic view[109] of the potential energy surface $E(\{\Psi_i\}, \{R_\alpha\})$. A typical trajectory is shown by the solid line which must lie close to the Born-Oppenheimer surface (dashed) line for the Hellman-Feynman theorem (theorem to calculate the forces on the atoms) to be implemented without large errors.

dynamics. The need for the system to be kept close to the Born-Oppenheimer surface governs the method in which Car-Parrinello molecular dynamics is used in practice. This is necessary to prohibit the unphysical transfer of energy from the fictitious dynamics to the atomic dynamics.

Appendix B

Some additional proofs

B.1 Independent parameters in density/projection matrix

Let $\mathbf{P}$ be the projection matrix for an M-dimensional subspace, V_P of an N-dimensional vector space, V. The number of independent parameters in $\mathbf{P}$, say N_P, is given by $N_P = M(N - M)$.

Proof:

N_P is also the minimum number of parameters that are necessary to describe the subspace V_P. V_P can for example be characterized by M orthonormal basis vectors, say $\{\mathbf{e}'_1, ..., \mathbf{e}'_M\}$. Each of these basis vectors can be expanded in terms of the chosen basis for V:

$$\mathbf{e}'_i = \sum_j u_{ij}\mathbf{e}_j. \qquad i = 1, ..., N \tag{B.1}$$

The requirement of normalization leads to that only $N - 1$ of these coefficients that,

say, specify $\mathbf{e}_1$ can be chosen independently. Since $\mathbf{e}_2$ is also normalized and is in addition orthonormal to $\mathbf{e}_1$, only $N-2$ of its coefficients can be chosen independently and so forth.

The number of independent parameters, N'_P, which is thus necessary to specify the orthonormal basis $\{\mathbf{e}'_i\}$ of V_P, is thus given by

$$\begin{aligned} N'_P &= (N-1)+(N-2)+....+(N-M) \\ &= MN-(1+2+....+M) \\ &= MN-\frac{M}{2}(2+(M-1)) \\ &= MN-\frac{M}{2}(M+1)\,. \end{aligned} \tag{B.2}$$

The orientation of the orthonormal basis vectors, $\{\mathbf{e}'_1,....,\mathbf{e}'_M\}$, inside the space, V_P could have been chosen arbitrarily without the space, V_P changing. The number of parameters, N'_P thus still contain excess information: It is namely valid that the number of parameters necessary to describe the orientation of the orthonormal base inside the space, V_P, is, say, N_M which is equal to $N'_P - N_P$.

The M basis vectors, $\{\mathbf{e}'_1,....,\mathbf{e}'_M\}$, can be specified with the help of an $M \times M$ orthogonal matrix transformation in terms of an arbitrary orthonormal basis, say $\{\mathbf{e}''_1,....,\mathbf{e}''_M\}$:

$$\mathbf{e}'_i = \sum_j A_{ij}\mathbf{e}''_j$$

with

$$\tilde{\mathbf{A}}\mathbf{A} = \mathbf{I}$$

or

$$\sum_k A_{ik}A_{kj} = \delta_{ij}$$

The number of independent parameters in an MxM orthogonal matrix is given by

$$N_M = \frac{M}{2}(M-1).$$

It thus follows that

$$\begin{aligned} N_P &= N'_P - N_M \\ &= MN - \frac{M}{2}(M+1) - \frac{M}{2}(M-1) \\ &= MN - \frac{M}{2}(M+1+M-1) \\ &= M(N-M) \\ &= Dim(V_P)Dim(V_P^{\perp}) \end{aligned} \tag{B.3}$$

It can be illustrated that equation (B.3) is indeed valid. Consider the following special case - a two-dimensional subspace contained within a three-dimensional vector space. This subspace can be specified using only two vectors, thus $M = 2$. The dimensionality of the vector space is three, thus $N = 3$. The number of parameters thus contained in a projection matrix for this two-dimensional subspace is $N \times M = 6$. However, these two vectors are constrained to be normalized as well as orthonormal to one another, which brings to pass that only three (the six parameters minus the three constraints) parameters are independent in the projection matrix. The orientation of these two vectors also has to be taken into consideration, any orientation can be chosen without changing the three-dimensional vector space. Taking this into account, the number of independent parameters in the projection matrix is thus two (the three parameters minus this last bit of superfluous information). This is in agreement with equation (B.3) where $N_P = M(N-M) = 2(3-2) = 2$.

In the above example the number of parameters are decreased from six to two. These two parameters are still sufficient to describe the orientation of the three-dimensional vector space. This, of course, can be generalized to an N-dimensional subspace. In this case, the number of independent parameters will be sufficient to describe the orientation of the subspace, V_p. It is thus evident that, by making use of the linear independent

parameters in the density matrix, the computational time necessary for calculations will significantly decrease.

Appendix C

Programming

The routines for the calculation of electronic structure by means of fictitious dynamics were incorporated into the existing MOPAC software package. Incorporating these routines into a well-developed program, with the capability of handling geometry optimizations and reaction coordinates, makes sense as this permits the focus to be primarily on the implementation of the fictitious dynamics method. MATLAB was also used intensively as a medium to test the FORTRAN code and as a convenient environment for graphics and some matrix manipulation.

In figure (C.1) a flow diagram of the FORTRAN program is illustrated. The user has the option of either continuing with a standard SCF calculation, or to use the fictitious dynamics method, which can be either $\mathbf{P}$ or $\underline{\mathbf{C}}$-dynamics. Before all, geometry optimization is done using the routines as supplied by the MOPAC package.

Consider now the flow diagram of the program for the case when a calculation is done using the $\mathbf{P}$-dynamics. A calculation starts with updating $\mathbf{P}$ by making use of the desired fictitious mass parameter i.e. a selection for b and δt, as well as the initial values for $\mathbf{P}$ and $\dot{\mathbf{P}}$. The idempotency of $\mathbf{P}$ is then tested and, should it be required, purified using McWeeny purification, as discussed in section 4.2.1.1. If $\mathbf{P}$ is sufficiently idempotent after purification, a new Fock matrix, $\mathbf{F}$ can be constructed using the

MOPAC routines FOCK1 and FOCK2. The fictitious velocity can thus also be updated. The electronic energy can then be calculated by making use of the MOPAC routine HELECT and, if energy convergence is reached, the MATLAB engine can be opened to construct a graph of electronic energy against fictitious time steps. If energy convergence is not reached the whole cycle is repeated, updating $\mathbf{P}$ and $\dot{\mathbf{P}}$ respectively. The same argumentation can be applied to the $\underline{\mathbf{C}}$-dynamics, with the only difference being the application of the RATTLE algorithm to ensure orthonormality of the columns of the configuration matrix.

C.1 Explanation of code

The argumentation followed in the programming (shown at the end of this appendix) can now be explained by considering the FORTRAN routine, PDynamics.

Before the routine for the $\mathbf{P}$ -dynamics is started, the one-electron and two-electron arrays, as determined by MOPAC, are written to four different output files. In MOPAC, all matrices are handled as arrays in order to cut down on computational overhead. In the program written for this study, however, matrices instead of arrays were used to simplify programming. It was thus necessary to convert the arrays as handled by MOPAC to matrices. In future, however, it will be considered working only with arrays due to the desire to reach O(N)-scaling. When the routine for the $\mathbf{P}$ -dynamics is started, the one and two-electron matrices is first read from the output files and the one-electron matrix is then used to construct an initial Fock matrix. An initial projection matrix with rank (N/2) is then generated, N being the number of electrons in the system. The associated reflection matrix, $(\mathbf{I} - \mathbf{P})$, is also generated. The projection matrix is then converted to a density matrix by multiplying it with the factor two and consequently it is converted to a density array. The alpha and beta density arrays are then calculated, which are equal since spinless particles are considered. These arrays, together with the two-electron arrays are then used to calculate the Fock array which

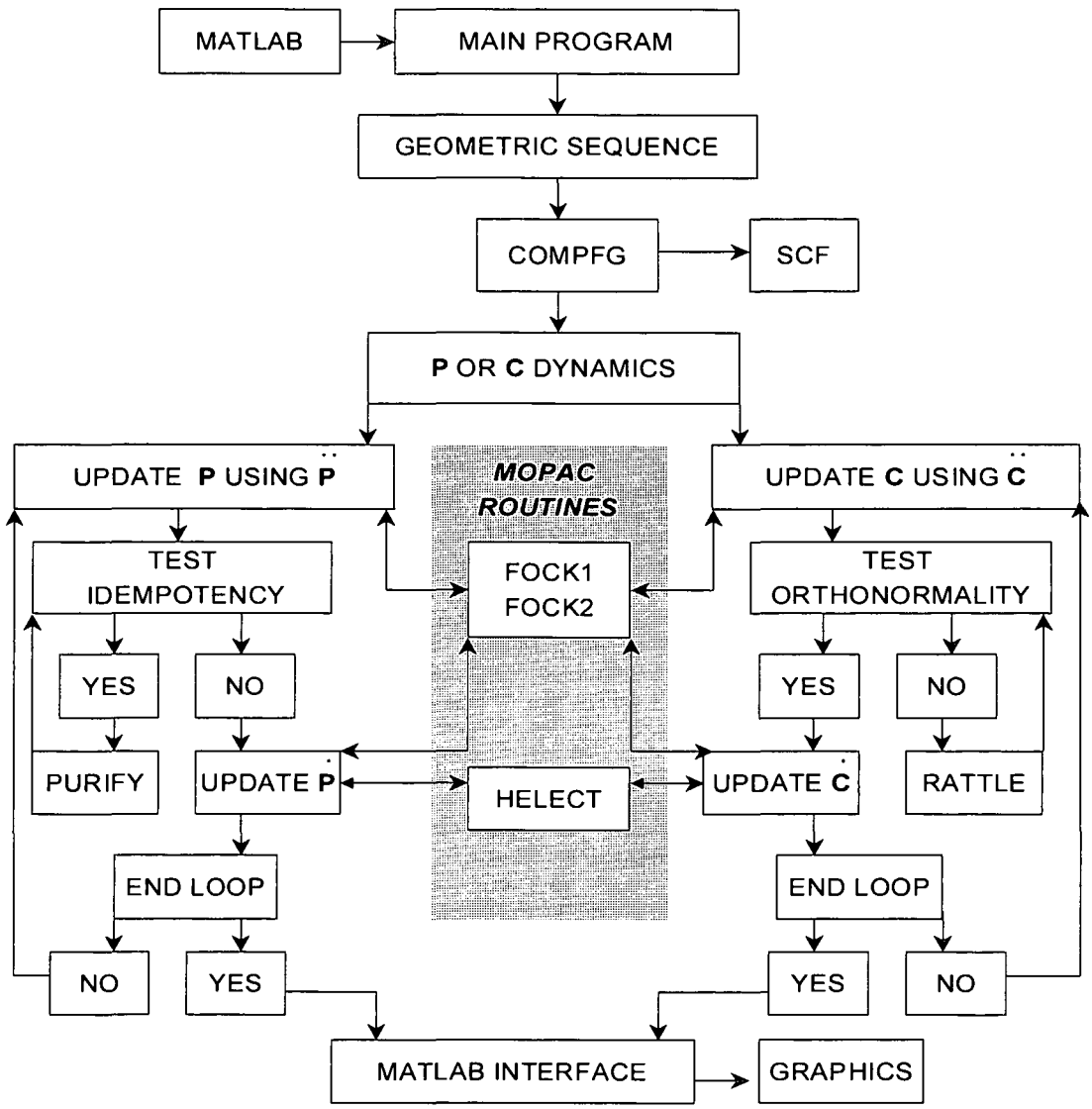

Figure C.1: Flow diagram of the FORTRAN program for fictitious electron dynamics.

is then converted to a Fock matrix. An initial fictitious velocity, appropriately scaled, are also generated.

All the ingredients are now available to start the integration of the equations of motion. Firstly, $\mathbf{P}$ is updated and tested for idempotency. If $\mathbf{P}$ is not idempotent enough, it is purified by making use of the McWeeny purification. As soon as $\mathbf{P}$ is idempotent enough, purification is seized. The projection matrix is then converted to a density matrix and consequently to a density array. The alpha and beta density arrays are then constructed from the density array. Following this, the initial Fock array is set equal to the one electron array and this, together with the two electron and density arrays, are then used to construct the Fock array. The electronic energy is then calculated using the Fock, density and Hamiltonian array. The Fock array and density matrix are then converted to a Fock and projection matrix respectively and used to update the $\dot{\mathbf{P}}$. The whole procedure is then repeated until energy convergence is reached.

The same explanation applies to the $\underline{\mathbf{C}}$-dynamics where the only difference occurs in the determination of the constraint forces to impose orthonormality on the configuration matrix by means of the RATTLE-algorithm.

```
C***********************************************************************
C     ROUTINE TO PERFORM P-DYNAMICS
C     READ THE NUMBER OF ELECTRONS, N, AND THE SIZE OF THE ARRAYS,
C     SIZE, FROM MOPAC
      SUBROUTINE PDYNAMICS(N,SIZE)
      INCLUDE 'SIZES.COPY'
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C     ALLOCATE SPACE FOR MATRICES READ FROM FILE AS WELL AS USED IN
C     CALCULATION
      DOUBLE PRECISION FARRAY(SIZE),PA(SIZE),PARRAY(SIZE),PB(SIZE)
      DIMENSION Z(N,N),D(N,N)
C     DECLARE VARIABLE NECESSARY TO DEFINE THE NUMBER OF ITERATIONS
C     NEEDED FOR PURIFICATION
      INTEGER NPURIFY
      PARAMETER (NPURIFY = 20)
C     DECLARE VARIABLE DEFINING THE NUMBER OF INTEGRATION STEPS
C     INTEGER ITERATION
      PARAMETER (ITERATION = 1000)
      DOUBLE PRECISION ERRORN,ID(N,N),ENERGY(ITERATION),
     1V(N,N),P(N,N),F(N,N),COMM(N,N),A(N,N),C(N,N),RAND2(N,N),RAND(N,N),
     1Y(N,N),X(N,N),SUM,IDEMP(N,N),PPRIME(N,N),B,EE,PMIN(SIZE),
     1FMIN(SIZE),H(SIZE),W(SIZE),WJ(SIZE),WK(SIZE),ETOTAL
C     DECLARE FICTITIOUS MASS VARIABLE
      PARAMETER (B = 0.1)
C     DEFINE DEGREE TO WHICH IDEMPOTENCY IS ENFORCED
      PARAMETER (IDEMTOL = 0.001)
C     DECLARE TIMESTEP VARIABLE
      REAL DT
      PARAMETER (DT = 0.1)
      REAL TIME_BEGIN, TIME_END,TIME_TAKEN
      INTEGER ENGOPEN,ENGGETMATRIX,MXCREATEFULL,MXGETPR
      INTEGER NCOUNT,EP,T,D,RESULT

      COMMON /TWOELE/ GSS(107),GSP(107),GPP(107),GP2(107),HSP(107)
     1               ,GSD(107),GPD(107),GDD(107)
      COMMON /MOLKST/ NUMAT,NAT(NUMATM),NFIRST(NUMATM),NMIDLE(NUMATM),
     1                NLAST(NUMATM), NORBS, NELECS,
     1                NALPHA, NBETA, NCLOSE, NOPEN, NDUMY, FRACT
     1      /MOLORB/ DUMMY(MAXORB),PDIAG(MAXORB)
     1      /KEYWRD/ KEYWRD
     1      /NUMSCF/ NSCF

C     START TAKING TIME TO DETEMINE HOW LONG THE CALCULATION
C     TAKES

      CALL CPU_TIME (TIME_BEGIN )

C     READ THE ONE AND TWO ELECTRON ARRAYS AS DETERMINED BY
C     MOPAC

      REWIND 16
        READ(16,*)(H(I),I=1,SIZE)
      REWIND 17
```

```
      READ(17,*)(W(I),I=1,SIZE)
     REWIND 18
      READ(18,*)(WJ(I),I=1,SIZE)
     REWIND 19
      READ(19,*)(WK(I),I=1,SIZE)

C     SET INITIAL FOCK ARRAY EQUAL TO THE ONE ELECTRON ARRAY

     DO I = 1,SIZE
        FARRAY(I) = H(I)
     END DO

C     GENERATE AN IDENTITY MATRIX

     DO I = 1,N
        DO J = 1,N
           IF (I.EQ.J) THEN
              ID(I,J) = 1.DO
           ELSE
              ID(I,J) = 0.DO
           END IF
        END DO
     END DO

C     GENERATE A STARTING PROJECTION MATRIX

     DO I = 1,N
        DO J = 1,N
           P(I,J) = 0.DO
        END DO
     END DO

     DO I = 1, N/2
        DO J = 1,N/2
           IF (I.EQ.J) THEN
             P(I,J) = 1.DO
           END IF
        END DO
     END DO

C     CONVERT PROJECTION MATRIX INTO DENSITY MATRIX

     P=2*P

C     CONVERT DENSITY MATRIX INTO DENSITY ARRAY

     NCOUNT=0
     DO I = 1,N
        DO J = 1, I
           NCOUNT = NCOUNT+1
           PARRAY(NCOUNT) = P(J,I)
        END DO
     END DO
```

```
C     MAKE ALPHA AND BETA DENSITY ARRAY

      DO I = 1,SIZE
         PA(I) = 0.5D0*PARRAY(I)
         PB(I) = 0.5D0*PARRAY(I)
      END DO

C     CALCULATE THE NEW FOCK ARRAY

      CALL FOCK2(FARRAY,PARRAY,PA,W, WJ,WK,NUMAT,NFIRST,NMIDLE,NLAST)
      CALL FOCK1(FARRAY,PARRAY,PA,PB)

C     CONVERT THE FOCK ARRAY INTO THE FOCK MATRIX

      DO I = 1,N
         K = I
         DO J=1, N
            K = K + J - 1
            IF (K.LE.SIZE)THEN
                F(I,J) = FARRAY(K)
            END IF
         END DO
      END DO

C     MAKE FOCK MATRIX SYMMETRIC

      DO I = 1,N
         DO J = 1,N
            F(J,I) = F(I,J)
         END DO
      END DO

C     CONVERT THE DENSITY MATRIX INTO A PROJECTION MATRIX

      P=0.5*P

C     GENERATE A RANDOM INITIAL VELOCITY AND SCALE TO BE SMALL
C     ENOUGH

      CALL RANDOM_SEED()
      CALL RANDOM_NUMBER(RAND)
      A = ID - P
      C = MATMUL(P,RAND)
      RAND2 = MATMUL(C,A)
      V = 0.001*(RAND2 + TRANSPOSE(RAND2))

C     APPLY VERLET AND IDEMPOTENCY CONSERVING EQUATIONS OF
C     MOTION

      LOOP 1:DO 100 Q=1,ITERATION

C            GENERATE A NEW PROJECTION MATRIX ASSOCIATED WITH SMALL
```

```
C             CHANGE IN TIME

              COMM = MATMUL(F,P) - MATMUL(P,F)

              P = P + DT*V+ 0.5*DT*DT*MATMUL((ID - 2*P),(-B*COMM +
     1        2*MATMUL(V,V)))

C             MAKE THE PROJECTION MATRIX SYMMETRIC

              P = 0.5*(P + TRANSPOSE(P))

C             NOW TEST FOR THE IDEMPOTENCY OF THE PROJECTION MATRIX

              IDEMP = MATMUL(P,P) - P
              X = TRANSPOSE(IDEMP)
              Y = MATMUL(X,IDEMP)
              SUM = 0.00
              DO I = 1, N
                 SUM = SUM + Y(I,I)
              END DO
              ERROR = SQRT(SUM)
              IF (ERROR > IDEMTOL) THEN

C             APPLY MCWEENY PURIFICATION

                  LOOP 2: DO 1 B = 1, NPURIFY
                          PPRIME = 3*MATMUL(P,P) - 2*MATMUL(P,MATMUL(P,P))
                          P = PPRIME
                          IDEMP = MATMUL(P,P) - P
                          X = TRANSPOSE(IDEMP)
                          Y = MATMUL(X,IDEMP)
                          SUM = 0.00
                          DO I = 1, N
                             SUM = SUM + Y(I,I)
                          END DO
                          ERRORN = SQRT(SUM)
                          IF (ERRORN.lt.IDEMTOL) THEN
                              EXIT LOOP 2
                          END IF
     1                1 CONTINUE
                    END IF


C             CONVERT PROJECTION MATRIX INTO DENSITY MATRIX

              P = 2*P

C             CONVERT DENSITY MATRIX INTO DENSITY ARRAY

              NCOUNT=0
              DO  I = 1,N
                  DO J = 1, I
                     NCOUNT = NCOUNT+1
```

```
                  PARRAY(NCOUNT) = P(J,I)
               END DO
            END DO

C           MAKE ALPHA AND BETA DENSITY ARRAY

            DO I = 1,SIZE
               PA(I) = 0.5D0*PARRAY(I)
               PB(I) = 0.5D0*PARRAY(I)
            END DO

C           SET FOCK MATRIX EQUAL TO THE ONE ELECTRON MATRIX

            DO I = 1,SIZE
               FARRAY(I) = H(I)
            END DO

C           CALCULATE THE NEW FOCK ARRAY

            CALL FOCK2(FARRAY,PARRAY,PA,W, WJ,
     1      WK,NUMAT,NFIRST,NMIDLE,NLAST)
            CALL FOCK1(FARRAY,PARRAY,PA,PB)

C           CONVERT THE FOCK ARRAY INTO THE FOCK MATRIX

            DO I = 1,N
               K = I
               DO J=1, N
                  K = K + J - 1
                  F(I,J) = FARRAY(K)
               END DO
            END DO

C           MAKE FOCK MATRIX SYMMETRIC

            DO I = 1,N
               DO J = 1,N
                  F(J,I) = F(I,J)
               END DO
            END DO

C           CALCULATE THE ELECTRONIC ENERGY

            EE = HELECT(N,PARRAY,H,FARRAY)

C           CONVERT DENSITY MATRIX TO PROJECTION MATRIX AND
C           CALCULATE NEW VELOCITY

            P = 0.5*P
            COMM = MATMUL(F,P)-MATMUL(P,F)
            V = V + 0.5*DT*MATMUL((ID - 2*P),((-B*COMM + 2*MATMUL(V,V))

C           PUT ENERGY VALUES IN MATRIX TO BE USED LATER IN
```

```
C           CONSTRUCTING A GRAPH

            ENERGY(Q) = EE

100   CONTINUE

C     DETERMINE THE TIME TAKEN TO DO THE CALCULATION AND WRITE
C     TO AN OUTPUT FILE

      CALL CPU_TIME ( TIME_END)
      TIMETAKEN = TIME_END - TIME_BEGIN
      WRITE(15,*)'ELAPSED TIME:'TIMETAKEN

C     NOW OPEN THE MATLAB ENGINE TO DO THE GRAPHICS

      ep = engOpen('matlab ')

C     TEST IF ENGINE CAN BE OPENED

      if (ep .eq. 0) then
         write(6,*) 'Can''t start MATLAB engine'
         stop
      endif

      T = MXCREATEFULL(1,ITERATION,0)
      CALL MXSETNAME(T,'T')
      CALL MXCOPYREAL8TOPTR(ENERGY,MXGETPR(T),ITERATION)

C     PLACE THE VARIABLE T INTO THE MATLAB WORKSPACE

      call engPutMatrix(ep, T)

C     PLOT THE RESULT

      call engEvalString(ep, 'plot(T);')
      call engEvalString(ep, 'title(''Time vs. fictitious mass'')')
      call engEvalString(ep, 'xlabel(''Fictitious mass'')')
      call engEvalString(ep, 'ylabel(''Time'')')

C     READ FROM THE CONSOLE TO MAKE SURE THAT WE PAUSE LONG
C     ENOUGH TO BE ABLE TO SEE THE PLOT

      print *, 'Type 0 <return> to Exit'
      print *, 'Type 1 <return> to continue'

      read(*,*) temp

      if (temp.eq.0) then
         print *, 'EXIT!'
         stop
      end if

      call engEvalString(ep, 'close;')
```

```
C     FREE THE VARIABLES

      call mxFreeMatrix(T)
      call mxFreeMatrix(result)

C     CLOSE THE MATLAB ENGINE
      stat = engClose(ep)

      STOP
      RETURN
      END


C ***********************************************************************
C     ROUTINE TO PERFORM C-DYNAMICS
C     READ THE NUMBER OF ELECTRONS, N, AND THE SIZE OF THE ARRAYS,
C     SIZE, FROM MOPAC
      SUBROUTINE CDYNAMICS(N,SIZE)
      INCLUDE 'SIZES.COPY'
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C     ALLOCATE SPACE FOR THE MATRICES READ FROM FILES AS WELL AS
C     USED DURING CALCULATION
      DOUBLE PRECISION F(SIZE),PARRAY(SIZE),PA(SIZE),PB(SIZE)
      COMMON /FOKMAT/ F2(MPACK), FB(MPACK)
      COMMON /DENSTY/ P2(MPACK), PA(MPACK), PB(MPACK)
      DIMENSION Z(N,N),D(N,N)
C     DECLARE VARIABLE TO DEFINE NUMBER OF ITERATIONS NECESSARY
C     FOR RATTLE
      INTEGER NRATTLE
      PARAMETER (NRATTLE = 200)
C     DECLARE VARIABLE FOR NUMBER OF FICTITIOUS TIME STEPS
      INTEGER ITERATION
      PARAMETER(ITERATION = 500)
C     DECLARE VARIABLE FOR THE FICTITIOUS MASS
      PARAMETER (B = 0.01)
C     DECLARE VARIABLE FOR THE ORTHONOMALITY TOLERANCE
      REAL ORTHOTOL
      PARAMETER (ORTHOTOL = 0.0001)
C     DECLARE VARIABLE FOR TIME STEP
      REAL DT
      PARAMETER (DT = 0.1)
      DOUBLE PRECISION ERROR,ID(N,N),PTOT(SIZE),ENERGY(ITERATION),
     1SIGMA(N/2,N/2),ETASUM(N/2,N/2),ETASUM2(N/2,N/2),
     1PTEST(N,N),FS(N,N),COMM(N,N),A(N,N),C(N,N/2),RAND2(N,N),
     1RAND(N,N),B,CDOT(N,N/2),CDOTDOT(N,N/2),GAMMASUM(N/2,N/2),
     1ETA(N/2,N/2),COLD(N,N/2),C_NEW(N,N/2),H(SIZE),W(SIZE),WJ(SIZE),
     1WK(SIZE),CMIN(N,N),X(N,N),Y(N,N)
      REAL TIME_BEGIN, TIME_END,TIME_TAKEN
      INTEGER EP,T,D,RESULT
      INTEGER ENGOPEN,ENGGETMATRIX,MXCREATEFULL,MXGETPR
      INTEGER NCOUNT,QQ,U,BB,II,G,MM,NN,R,S,TIME,MTIME

      COMMON /TWOELE/ GSS(107),GSP(107),GPP(107),GP2(107),HSP(107)
```

```
     1                ,GSD(107),GPD(107),GDD(107)
      COMMON /MOLKST/ NUMAT,NAT(NUMATM),NFIRST(NUMATM),NMIDLE(NUMATM),
     1                NLAST(NUMATM), NORBS, NELECS,
     1                NALPHA, NBETA, NCLOSE, NOPEN, NDUMY, FRACT
     1       /MOLORB/ DUMMY(MAXORB),PDIAG(MAXORB)
     1       /KEYWRD/ KEYWRD
     1       /NUMSCF/ NSCF

C     START TO CALCULATE HOW LONG ROUTINE TAKES

      CALL CPU_TIME(TIME_BEGIN)

C     READ THE ONE AND TWO ELECTRON ARRAYS AS CALCULATED BY MOPAC

      REWIND 16
        READ(16,*)(H(I),I=1,SIZE)
      REWIND 17
        READ(17,*)(W(I),I=1,SIZE)
      REWIND 18
        READ(18,*)(WJ(I),I=1,SIZE)
      REWIND 19
        READ(19,*)(WK(I),I=1,SIZE)

C     SET INITIAL FOCK MATRIX EQUAL TO THE ONE ELECTRON MATRIX

      DO I = 1,SIZE
         F2(I) = H(I)
      END DO

C     GENERATE AN IDENTITY MATRIX

      DO I = 1,N
         DO J = 1,N
            IF (I.EQ.J) THEN
               ID(I,J) = 1.D0
            ELSE
               ID(I,J) = 0.D0
            END IF
         END DO
      END DO

C     GENERATE A STARTING PROJECTION MATRIX

      DO I = 1,N
         DO J = 1,N
            P(I,J) = 0.D0
         END DO
      END DO

      DO I = 1, N/2
         DO J = 1, N/2
            IF (I.EQ.J) THEN
               P(I,J) = 1.D0
```

```
            END IF
         END DO
      END DO

C     CONVERT PROJECTION MATRIX INTO DENSITY MATRIX

      P=2*P

C     CONVERT DENSITY MATRIX INTO DENSITY ARRAY

      NCOUNT=0
      DO I = 1,N
         DO J = 1, I
            NCOUNT = NCOUNT+1
            PARRAY(NCOUNT) = P(J,I)
         END DO
      END DO

C     MAKE ALPHA AND BETA DENSITY ARRAY

      DO I = 1,SIZE
         PA(I) = 0.5D0*PARRAY(I)
         PB(I) = 0.5D0*PARRAY(I)
      END DO

C     CALCULATE THE NEW FOCK ARRAY

      CALL FOCK2(FARRAY,PARRAY,PA,W, WJ, WK,NUMAT,NFIRST,NMIDLE,NLAST)
      CALL FOCK1(FARRAY,PARRAY,PA,PB)

C     CONVERT THE FOCK ARRAY INTO THE FOCK MATRIX

      DO I = 1,N
         K = I
         DO J=1, N
            K = K + J - 1
            F(I,J) = FARRAY(K)
         END DO
      END DO

C     MAKE FOCK MATRIX SYMMETRIC

      DO I = 1,N
         DO J = 1,N
            F(J,I) = F(I,J)
         END DO
      END DO

C     CONVERT THE DENSITY MATRIX INTO A PROJECTION MATRIX

      P=0.5*P

C     GENERATE AN INITIAL CONFIGURATION MATRIX
```

```
      DO I = 1, N
         DO J = 1,N/2
            C(I,J) = 0.DO
         END DO
      END DO

      DO I = 1,N/2
         DO J = 1,N/2
            IF (I.EQ.J)THEN
                C(I,J) = 1.DO
            ENDIF
         ENDDO
      ENDDO

C     GENERATE AN INITIAL VELOCITY

      DO I =1 ,N
         DO J = 1,N/2
            CDOT(I,J) = 0.DO
         ENDDO
      ENDDO


C     START THE INTEGRATION PROCEDURE

      DO 100 Q=1,ITERATION

C        CALCULATE THE FICTITIOUS ACCELERATION

         CDOTDOT = (-B*MATMUL((ID-P),MATMUL(F,C)) - MATMUL(C,
     1             MATMUL(TRANSPOSE(CDOT),CDOT)))

C        STORE THE INITIAL VALUE OF C FOR USE IN RATTLE
         DO I = 1, N
            DO J = 1, N/2
               COLD(I,J) = C(I,J)
            ENDDO
         ENDDO

C        GENERATE A NEW C MATRIX ASSOCIATED WITH SMALL CHANGE IN TIME

         C = C + DT*CDOT + 0.5*DT*DT*CDOTDOT

C        DETERMINE ORTHONORMALITY

         X = TRANSPOSE(MATMUL(C,TRANSPOSE(C))-ID)
         Y = MATMUL(X,MATMUL(C,TRANSPOSE(C))-ID)
         SUM = 0.00
         DO I = 1, N
            SUM = SUM + Y(I,I)
         END DO
         ERROR = SQRT(SUM)
```

```
      IF (ERROR > ORTHOTOL)

C     APPLY RATTLE

      LOOP 2: DO 1 Q = 1, NRATTLE

C     GENERATE THE CONSTRAINT EQUATIONS

      DO I = 1,N/2
         DO J = 1,I
            SIGMA(I,J) = 0.0
            DO K = 1,N
               SIGMA(I,J) = SIGMA(I,J) +  C(K,I)*C(K,J)
            ENDDO
               SIGMA(I,J) = SIGMA(I,J) - ID(I,J)

C           DETERMINE THE LANGRANGE MULTIPLIERS

            DO QQ = 1,1
               GAMMASUM(I,J) = 0.0
               DO NN = 1,N/2
                  DO MM = 1,N
                     GAMMASUM(I,J) = GAMMASUM(I,J) + (COLD(MM,j)*
1              ID(NN,I) + COLD(MM,I)*ID(NN,J))*(C(MM,j)*ID(NN,i) +
     1                 C(MM,I)*ID(NN,J))
                  ENDDO
               ENDDO


               DO S = 1,N/2
                  DO R = 1,N
                     C(R,S) = C(R,S) - (SIGMA(I,J)*(COLD(R,J)*
     1               ID(S,I) + COLD(R,I)*ID(S,J)))/(GAMMASUM(I,J))
                  ENDDO
                ENDDO
               ENDDO
             ENDDO
            ENDDO

            X = TRANSPOSE(MATMUL(C,TRANSPOSE(C))-ID)
            Y = MATMUL(X,MATMUL(C,TRANSPOSE(C))-ID)
            SUM = 0.00
            DO I = 1, N
               SUM = SUM + Y(I,I)
            END DO
            ERRORN = SQRT(SUM)
            IF (ERRORN.lt.IDEMTOL) THEN
               EXIT LOOP 2
            END IF
1           CONTINUE
         END IF
```

```
C          CALCULATE THE DENSITY MATRIX FROM C

           P = 2*MATMUL(C,TRANSPOSE(C))

C          CONVERT DENSITY MATRIX INTO A DENSITY ARRAY

           NCOUNT=0
           DO I = 1,N
              DO J = 1, I
                 NCOUNT = NCOUNT+1
                 PARRAY(NCOUNT) = P(J,I)
              END DO
           END DO

C          MAKE ALPHA AND BETA DENSITY ARRAY

           DO I = 1,SIZE
              PA(I) = 0.5D0*PARRAY(I)
              PB(I) = 0.5D0*PARRAY(I)
           END DO

C          SET FOCK MATRIX EQUAL TO THE ONE ELECTRON MATRIX

           DO I = 1,SIZE
              FARRAY(I) = H(I)
           END DO

C          CALCULATE THE NEW FOCK ARRAY

           CALL FOCK2(FARRAY,PARRAY,PA,W, WJ, WK,NUMAT,NFIRST,NMIDLE,NLAST)
           CALL FOCK1(FARRAY,PARRAY,PA,PB)

C          CONVERT THE FOCK ARRAY INTO THE FOCK MATRIX

           DO I = 1,N
              K = I
              DO J=1, N
                 K = K + J - 1
                 F(I,J) = FARRAY(K)
              END DO
           END DO

C          MAKE FOCK MATRIX SYMMETRIC

           DO I = 1,N
              DO J = 1,N
                 F(J,I) = F(I,J)
              END DO
           END DO

C          CALCULATE THE ELECTRONIC ENERGY

           EE = HELECT(N,PARRAY,H,FARRAY)
```

```
C          TEST FOR THE MINIMUM ELECTRONIC ENERGY

           IF (EE < Emin) THEN
              EMIN = EE
              CMIN = C
           END IF

C          WRITE ENERGY TO FILE

           WRITE(15,*)EE

C          PUT ENERGY VALUES IN MATRIX TO BE USED LATER IN CONSTRUCTING A GRAPH

           ENERGY(Q) = EE

C          CONVERT DENSITY MATRIX TO PROJECTION MATRIX AND CALCULATE NEW
C          VELOCITY

           P = 0.5*P

C          CALCULATE ACCELERATION

           CDOTDOT = (-B*MATMUL((ID-P),MATMUL(F,C)) - MATMUL(C,
     1               MATMUL(TRANSPOSE(CDOT),CDOT)))

C          UPDATE THE FICTITIOUS VELOCITY USING VERLET

           CDOT = CDOT +  0.5*DT*CDOTDOT

C          DETERMINE ORHONORMALITY

           X = TRANSPOSE(MATMUL(C,TRANSPOSE(C))-ID)
           Y = MATMUL(X,MATMUL(C,TRANSPOSE(C))-ID)
           SUM = 0.00
           DO I = 1, N
              SUM = SUM + Y(I,I)
           END DO
           ERROR = SQRT(SUM)

           IF (ERROR > ORTHOTOL)

C          APPLY RATTLE

           LOOP 3: DO 10 G = 1, NRATTLE

C          CALCULATE LANGRANGE MULIPLIERS

           DO QQ = 1,1
              DO I = 1,N/2
                 DO J = 1,I
                    ETASUM(I,J) = 0.0
                    ETASUM2(I,J) = 0.0
```

```
                  DO NN = 1,N/2
                     DO MM = 1,N
                        ETASUM(I,J) = ETASUM(I,J) + CDOT(MM,NN)*
1                 (C(MM,j)*ID(NN,I)+ C(MM,I)*ID(NN,J))
1                 ETASUM2(I,J) = ETASUM2(I,J)+ (C(MM,J)*
1                 ID(NN,I) + C(MM,ID)*ID(NN,J))*(C(MM,J)*ID(NN,I)
      1                 + C(MM,I)*ID(NN,J))
                     ENDDO
                  ENDDO

                  DO S = 1,N/2
                     DO R = 1,N
                        CDOT(R,S) = CDOT(R,S) - ((C(R,J)*ID(S,I)+
1                 C(R,I)*ID(S,J))*ETASUM(I,J))/ETASUM2(I,J))
                     ENDDO
                  ENDDO
                ENDDO
              ENDDO
             ENDDO
              X = TRANSPOSE(MATMUL(C,TRANSPOSE(C))-ID)
              Y = MATMUL(X,MATMUL(C,TRANSPOSE(C))-ID)
              SUM = 0.00
              DO I = 1, N
                 SUM = SUM + Y(I,I)
              END DO
              ERRORN = SQRT(SUM)
              IF (ERRORN.lt.IDEMTOL) THEN
                 EXIT LOOP 3
              END IF
10            CONTINUE
           END IF

100   CONTINUE

C     DETERMINE TIME CALCULATION HAS TAKEN

      CALL CPU_TIME (TIME_END )
      TIME_TAKEN = TIME_END - TIME_BEGIN

C     WRITE THE TIME TAKEN FOR THE CALCUALTION TO AN OUTPUT FILE

      WRITE (15,*)'ELAPSED TIME:',TIME_TAKEN,'  SECONDS'

C     WRITE MINIMUM ENERGY TO AN OUTPUT FILE

      WRITE(15,*)'EMIN'
      WRITE(15,*)EMIN

C     OPEN THE MATLAB ENGINE

      EP = ENGOPEN('MATLAB ')

C     TEST IF ENGINE CAN BE OPENED
```

```
      IF (EP .EQ. 0) THEN
          WRITE(6,*) 'Can''t start MATLAB engine'
          STOP
      ENDIF

C     ALLOCATE SPACE FOR MATRIX T

      T = MXCREATEFULL(1,ITERATION,0)
      CALL MXSETNAME(T,'T')

C     COPY ENERGY VALUES INTO MATRIX T

      CALL MXCOPYREAL8TOPTR(ENERGY,MXGETPR(T),ITERATION)

C     PLACE THE VARIABLE T INTO THE MATLAB WORKSPACE

      call engPutMatrix(ep, T)

C     PLOT THE RESULT

      call engEvalString(ep, 'plot(T);')
      call engEvalString(ep, 'title(''Energy vs. no. of iterations'')')
      call engEvalString(ep, 'xlabel(''Iterations'')')
      call engEvalString(ep, 'ylabel(''Energy (eV)'')')

C     READ FROM THE CONSOLE TO MAKE SURE THAT WE PAUSE LONG ENOUGH TO BE
C     ABLE TO SEE THE PLOT

      print *, 'Type 0 <return> to Exit'
      print *, 'Type 1 <return> to continue'

      read(*,*) temp

      if (temp.eq.0) then
          print *, 'EXIT!'
          stop
      end if

      call engEvalString(ep, 'close;')

C     FREE THE VARIABLES

      call mxFreeMatrix(T)
      call mxFreeMatrix(result)

C     CLOSE THE MATLAB ENGINE
      stat = engClose(ep)

      STOP
      RETURN
      END
C ******************************************************************************
```

```
**********************************************************************
DensityMatrix.m
**********************************************************************

clear all
TS = input('Timestep:')
R = input('Iterations:')
count = double(R);
cmin = 0;
N=6;
Npurify = 100;
Niter = 200;
idemtol = 1E-4;
errorn = 0.0;
KE = double(R);
Energy = double(R);
counter = 0;
Emin=0;
mass = double(R);
number = 50;
Ecompare = double(number,R);

%Construct one and two electon matrices
H =[-51.7112   -3.2457   -1.0970   -0.6992   -1.0970  -3.2457;
     -3.2457  -51.7112   -3.2457   -1.0970   -0.6992  -1.0970;
     -1.0970   -3.2457  -51.7112   -3.2457   -1.0970  -0.6992;
     -0.6992   -1.0970   -3.2457  -51.7112   -3.2457  -1.0970;
     -1.0970   -0.6992   -1.0970   -3.2457  -51.7112  -3.2457;
     -3.2457   -1.0970   -0.6992   -1.0970   -3.2457 -51.7112];

Two_Mat = [12.848    9.6583   7.0632  6.3620  7.0632  9.65830;
            9.6583  12.8480   9.6583  7.0632  6.3620   7.0632;
            7.0632   9.6583  12.8480   9.6583  7.0632  6.3620;
            6.3620   7.0632   9.6583  12.8480  9.6583  7.0632;
            7.0632   6.3620   7.0632   9.6583 12.8480  9.6583;
            9.6583   7.0632   6.3620   7.0632  9.6583 12.8480];

%Generate projection matrix
P = eye(N,N/2);
P = P*P';

%Convert projection matrix to density matrix
P=2*P;

%Calculate Fock matrix
F=fockmatrix(N,H,Two_Mat,P);

%Convert density matrix to projection matrix
P=0.5*P;

%Calculate random initial velocity
A_D_P = rand(N,N);
V = InitialVel(A_D_P,P);
```

```
c=0;

%start integration procedure
for i = 1:R
      c = c + 1;

      % calculate projection matrix
      P = DensMatrix(P,V,F,T,fictmass);

      %Test idempotency of projection matrix
      Idemerror = P^2 - P;
      error(i) = norm(Idemerror,'fro');
      if  error(i) > idemtol
          %apply McWeeny purification
          Pn = Purify(P,N,idemtol,errorn,Neig,Npurify);
          error(i) = norm((Pn^2 - Pn),'fro');
          P = Pn;
      end

      %convert projection matrix to density matrix
      P=2*P;

      %calculate the Fock matrix
      F=fockmatrix(6,H,Two_Mat,P);

      %calculate the electronic energy
      E = energy(6,H,F,P) ;

      %put the electronic energy into a matrix
      Energy(c) = E;

      %convert density matrix to projection matrix
      P = 0.5*P;

      %calculate the fictitious velocity
      V = Vel(P,F,V,T,fictmass);

      %Calculate the kinetic energy
      KESUM=0;
      for i = 1:N
          for j = 1:N
              KESUM = KESUM + 0.5*fictmass*V(i,j)*V(i,j);
          end
      end

      %put the kinetic energy in a matrix
      KE(c) = KE;
    end
end
```

```
**************************************************************************
Velocity.m
**************************************************************************

%function to update the fictitious velocity
function V=Vel(P,F,V,T,fictmass)
A = (eye(6,6) - 2*P)*((-fictmass*(F*P-P*F)) + 2*V*V);
V = V + 0.5*T*A;

**************************************************************************
DensMatrix.m
**************************************************************************

%function to update the projection matrix
function P=DensMatrix(P,V,F,T,fictmass)
P = P + T*V + 0.5*T*T*(eye(6,6) - 2*P)*((-fictmass*(F*P-P*F)) + 2*V*V);
P = 0.5*(P + P');

**************************************************************************
Energy.m
**************************************************************************

%function to calculate the electronic energy
function Ee=energy(N,H,F,P)
Ee=0.0 ;
for i = 1:N
      for j = 1:N
       Ee = Ee + 0.5*P(i,j)*(H(i,j)+F(i,j))  ;
      end
end

**************************************************************************
Fockmatrix.m
**************************************************************************

%function to calculate the Fock matrix
function F=fockmatrix(N,H,Two_Mat,P)
sum =0;
for i = 1:N
    for j = 1:N
        if i == j
           for k = 1:N
           sum = sum + Two_Mat(1,k);
           end
           F(i,j) = H(i,j) + 0.5*sum;
        else
           F(i,j) = H(i,j) - 0.5*P(i,j)*Two_Mat(i,j);
        end
    end
end
```

```
**************************************************************************
ConfMatrix.m
**************************************************************************
clear all
T = input('Timestep:')
R = input('Iterations:')
N=6;
sigma = zeros(N/2,N/2)
sigmaold = zeros(N/2,N/2)
KE = double(RS);
etasum = zeros(N/2,N/2)
sigma = zeros(N/2,N/2)
gammasum = zeros(N/2,N/2)
etasum = zeros(N/2,N/2)
fictmass = 2.828;
cmin = 0;
number = double(RS);
orthtol = 1e-4;
errorn = 0.0;
Energy = double(R);
CDOTDOT = zeros(6,3);
orth = double(RS);
Emin=0;

%construct one and two electron matrices
H =[-51.7112   -3.2457   -1.0970   -0.6992   -1.0970  -3.2457;
     -3.2457  -51.7112   -3.2457   -1.0970   -0.6992  -1.0970;
     -1.0970   -3.2457  -51.7112   -3.2457   -1.0970  -0.6992;
     -0.6992   -1.0970   -3.2457  -51.7112   -3.2457  -1.0970;
     -1.0970   -0.6992   -1.0970   -3.2457  -51.7112  -3.2457;
     -3.2457   -1.0970   -0.6992   -1.0970   -3.2457 -51.7112];

Two_Mat = [12.848    9.6583   7.0632  6.3620  7.0632  9.65830;
            9.6583  12.8480   9.6583  7.0632  6.3620   7.0632;
            7.0632   9.6583  12.8480   9.6583  7.0632  6.3620;
            6.3620   7.0632   9.6583  12.8480  9.6583  7.0632;
            7.0632   6.3620   7.0632   9.6583 12.8480  9.6583;
            9.6583   7.0632   6.3620   7.0632  9.6583 12.8480];

%Generate projection matrix
P = eye(N,N/2);
P = P*P';

%Convert projection matrix to density matrix
P=2*P;

%Calculate Fock matrix
F=fockmatrix(N,H,Two_Mat,P);

%convert density matrix to projection matrix
P=0.5*P;

%construct an initial configuration matrix
```

```
for i = 1:N
    for j = 1:N/2
        if (i == j)
            C(i,j) = 1;
         else
            C(i,j) = 0;
         end
    end
end

COLD =zeros(N,N/2);

%construct an initial fictitious velocity
CDOT = zeros(N,N/2);

%start integration procedure
for Q=1:R

%note old value of C to be used in RATTLE
      for I = 1:N
          for J = 1:N/2
              COLD(I,J) = C(I,J);
          end
      end

      %update the configuration matrix
      C = ConfMatrix(C,CDOT,N,fictmass,P,T)

      %test orthonormality
      Ortherror = C'C - eye(6,3);
      error(i) = norm(Ortherror,'fro');
   if  error(i) > orthtol
          %apply RATTLE
          C = RATTLE(Npurify,N,C,COLD)
      end

      %construct the density matrix
      P = 2*C*C';

      %calculate the Fock matrix
      F=fockmatrix(6,H,Two_Mat,P);

      %convert density matrix to projection matrix
      P = 0.5*P;

      %update fictitious velocities
      CDOT = ConfMatrixVel(P,F,C,CDOT,N,fictmass);

      %test orthonormality
      Ortherror = CDOT'CDOT - eye(6,3);
      error(i) = norm(Ortherror,'fro');

   if  error(i) > orthtol
```

```
   %apply RATTLE
         CDOT = RATTLE2(Npurify,P,F,C,CDOT,N)
      end if

      %convert the projection matrix to a density matrix
      P = 2*P;

      %calculate the Fock matrix
      F=fockmatrix(N,H,Two_Mat,P);

      %calculate the electronic energy
      E = energy(6,H,F,P);

      %convert the density matrix to a projection matrix
      P = 0.5*P;

      %put the energy in a matrix
      Energy(Q) = E;

     end
end
```

```
*******************************************************************************
C.m
*******************************************************************************

%function to calculate the new configuration matrix
function C = ConfMatrix(C,CDOT,N,fictmass,P,T)
CDOTDOT = -fictmass*(eye(6,6)-P)*F*C - C*CDOT'*CDOT;
C = C + T*CDOT + 0.5*T*T*CDOTDOT;
```

```
*******************************************************************************
CRATTLE.m
*******************************************************************************

%function to calculate the new configuration matrix including
%the constraint forces as determined by RATTLE
function C = RATTLE(NRATTLE,N,C,COLD)
for G = 1:NRATTLE
   %construct constraint equations
   for i = 1:N/2
       for j = 1:j
       sigma(i,j) = 0;
       for k = 1:N
           sigma(i,j) = sigma(i,j) +  C(k,i)*C(k,j);
       end
       sigma(i,j) = sigma(i,j) - eye(i,j);

       %calculate constraint forces
       for QQ = 1:1
           gammasum(i,j) = 0;
           for nn = 1:3
               for mm = 1:6
```

```
                gammasum(i,j) = gammasum(i,j) +
                (COLD(mm,j)*eye(nn,i) +
                C(mm,i)*eye(nn,j));
            end
        end

        %add effect of the constraint forces
        for s = 1:N/2
            for r = 1:N
                C(r,s) = C(r,s) - (sigma(i,j)*(COLD(r,j)*eye(s,i) +
                COLD(r,i)*eye(s,j)))/(gammasum(i,j));
            end
        end
      end
     end
    end
```

```
*************************************************************************
CDOT.m
*************************************************************************

%function to update the fictitious velocities
function CDOT = ConfMatrixVel(N,F,C,fictmass,CDOT);
CDOTDOT = -fictmass*(eye(N,N)-P)*F*C - C*CDOT'*CDOT;
CDOT = CDOT +  0.5*T*CDOTDOT;
```

```
*************************************************************************
CDOTRATTLE.m
*************************************************************************

%function to determine the new fictitious velocities
%as determined by %RATTLE
function CDOT = RATTLE2(NRATTLE,P,F,C,CDOT,N)
    for G = 1:NRATTLE
        for QQ = 1:1

        %construct the constraint equations
        for i = 1:N/2
            for j = 1:i
                etasum(i,j) = 0;
                etasum2(i,j) = 0;

                %calculate the constraint forces
                for nn = 1:N/2
                    for mm = 1:N
                        etasum(i,j) = etasum(i,j) +
                        C(mm,i)*eye(nn,j));
                        etasum2(i,j) = etasum2(i,j) + (C(mm,j)*eye(nn,i)
                        + C(mm,i)*eye(nn,j))*(C(mm,j)*eye(nn,i) +
                        C(mm,i)*eye(nn,j));
                     end
                end
```

```
            %add the constraint forces
            for s = 1:N/2
                for r = 1:N
                    CDOT(r,s) = CDOT(r,s) - ((C(r,j)*eye(s,i)+
                    C(r,i)*eye(s,j))*etasum(i,j))/(etsum2(i,j));
                end
            end
        end
    end
end

*******************************************************************************
```

Bibliography

[1] Bell, J.S.; *Speakable and unspeakable in quantum mechanics*; Cambridge University Press, 1987.

[2] Kinoshita, T; Lindquist, W.B.; Phys. Rev. Lett. **47** (22), 1573, 1981.

[3] Van Dyck Jr., R.S.; Schwinberg, P.B.; Dehmelt, G.H.; Phys. Rev. Lett. **59** (1), 26 July 1987.

[4] Szabo, A.; Ostlund, N.S.; *Modern Quantum Chemistry* ; McGraw-Hill, New York, 1982.

[5] Hirst, D.M.; *A Computational Approach to Chemistry*; Blackwell Scientific Publications, Oxford, pp. 101-120, 1990.

[6] Jensen, F.; *Introduction to Computational Chemistry*; John Wiley and Sons, 1998.

[7] Hehre, W.J.; Radom, L, Scheleyer, P.; Pople, J.; *Ab Initio Molecular Orbital Theory*; John Wiley & Sons, New York, 1986.

[8] Thiel, W.; Adv. Chem. Phys. **93**, pp 703-757, 1996.

[9] Dewar, M.J.S.; *The Molecular Orbital Theory of Organic Chemistry*; McGraw-Hill Book Company: New York, p 78, 1979.

[10] James J.P. Stewart, J. of Computer-aided Molecular Design, **4**, 1990.

[11] Cox, R.; Williams, D.E.; J. of Comp. Chem. **2**, p 304, 1981.

[12] Barger, V.D.; Olsson, M.G.; *Classical Mechanics: A Modern Perspective*; McGraw-Hill, 1995.

[13] Kohn, W.; Int. J. Quant. Chem. **56**; p 229, 1995.

[14] Pettifor, D.G.; Phys. Rev. Lett. **63**, p 2480, 1989.

[15] Yang, W; Phys. Rev. Lett. **66**, p 1428, 1991.

[16] Galli, G; Parinello, M.; Phys. Rev. Lett. **69**, p 3547, 1992.

[17] Li, X; Nunes, W.; Vanderbilt, D; Phys. Rev. B **47**, p 10891, 1993.

[18] Goedecker, S.; Colombo, L; Phys. Rev. Lett. **73**, p 122, 1994.

[19] Kress, J.D.; Voter, A.F.; Phys. Rev. B **53**, p 12733, 1996.

[20] Horsefield, A.P.; Mat. Sci. Engin. B **37**, p 219, 1996.

[21] Hoffman, M.J.H.; PhD Thesis, University of the Free-State, 1998.

[22] Vittadini A.; Selloni A.; Casarin M.;Surf. Sci. Lett. **289**, L625, 1993.

[23] Nachtigall, P.; Jordon, K.D.; Journal of Chemical Physics **102**, pp 8249 - 8259, 1995.

[24] Wu C.J.; Carter E.A.; Chem. Phys. Lett. **185**, p 172, 1991.

[25] Wu C.J.; Ionova I.I.; Carter E.A.; Phys. Rev. B **49**, p 13488, 1994.

[26] Hansen, U; Vogl, P. Phys. Rev. B **57**, p 13295, 1998.

[27] Bowler, D.R.; Owen, J.H.G.; Goringe, C.M.,; Miki, K; Briggs, G.A.D.; J. Phys.: Condens. Matter **12**, pp 7655-7670, 2000.

[28] Freelon, B; Hill, E; Ganz, E; Surf. Sci. **385**, p146, 1999.

[29] First Principles Simulation of Materials Properties, July 1998 (http://hpcf.nersc.gov/about/ERSUG/workinp/greenbook/node19.html)

[30] Computational Chemistry, March 2002,
(http://www.uscs.edu /llever/Computational%20Chemistry/compchem.html)

[31] Kittel, C; Wiley, 1956.

[32] Forcefields, August 1999
(http://homepages.nyu.edu/~mt33/jpcfeat/node9.html)

[33] Tuckerman, M.E.; Ungar, P.J.; von Rosenvinge, T.; Klein, M.C.; J. Phys. Chem. **100**, pp 12878-12887, 1996.

[34] Ab initio molecular dynamics - Theory and Implementation, John von Neumann Institute for Computing, 2000
(http://www.fz-juelich.de/nic-series/Volume3/marx.pdf)

[35] Schlegel, H.B.; Iyengar, S.S.; Li, X.; Millam, J.M.; Voth, G., Souseria, G.E. Frisch, M.J.; Journal of Chemical Physics **117**, p 8701, 2002.

[36] Nagy, A.; Physics Reports **298**, p 1-79, 1998.

[37] Dal Corso, A.; Pasquarello, A.; Baldereschi, A.; Car, R.; Phys. Rev. B **53**, p 1180, 1996.

[38] Goedecker, S.; Rev. of Mod. Phys. **71**, p 1088, 1999.

[39] Gershgorn, Z.; Shavitt, I.; Int. J. Quantum. Chem. **2**, p 751, 1968.

[40] Knowles, P.J.; Handy, N.C.; Chem. Phys. Lett. **111**, p 315, 1984.

[41] Olsen, J.; Roos, B.O.; Jorgensen, P.; Jensen, H.J.; J. Chem. Phys. **89**, p 2185, 1988.

[42] Gao, J.; Thompson, M.A.; *Combined Quantum Mechanical and Molecular Mechanical Methods*; Oxford University Press, 1999.

[43] Eurenius, K.P.; Chatfield, S.; Hodoscek, B.; Brooks, K.; Int. J. Quant. Chem. **60**, pp 1189-1200, 1996.

[44] Merz, M.; Acc. Chem. Res. **32**, pp 904-911,1999.

[45] Levine, I.N.; *Quantum Chemistry*; Allyn and Bacon Inc., 1983.

[46] Gasiorowicz, S; *Quantum Physics*; John Wiley and Sons, New York, 1974.

[47] Roothaan, C.C.J.; Rev. Mod. Phys. **23**, 69, 1951.

[48] Sutton, A.P.; *Electronic Structure of Materials*; Oxford University Press, 1993.

[49] Parr, R.G.; Yang, W.; *Density Functional Theory for Atoms and Molecules*; Oxford University Press, New York, 1988.

[50] Car, R.; Parrinello, M.; Phys. Rev. Lett. **55**, pp 2471-2474, 1985.

[51] Palser, A.H.R; Manolopoulos, D.E; Phys. Rev. B **58**, p 12704, 1998.

[52] Verlet, L. Phys. Rev. **159**, pp 98 - 103, 1967.

[53] Swope, W.C.; Andersen, H.C.; Berens, P.H; Wilson, K.R.; J. Chem. Phys. **76**, pp 631 - 649, 1982.

[54] Kutteh, R.; *RATTLE recipe for general holonomic constraints: angle and torsion constraints*; October 1998.

[55] Allen, M.P.; Tildesley, D.J.; *Computer Simulation of Liquids*; Oxford University Press, Oxford, 1987

[56] McWeeny, R.; Rev. Mod. Phys. **32**, p 335, 1960.

[57] Ryckaert, J.P.; Ciccotti, G.; Berendsen, H.J.C.; J. Comput. Phys. **23**, p 327, 1977.

[58] Andersen, H.C.; J. Comput. Phys. **52**, p 24, 1983.

[59] Dahlquist, G.; Björk, A.; *Numerical Methods*; Prentice-Hall Englewood Cliffs, N.J., 1974.

[60] Ryckaert, J.P.; Mol. Phys. **55**, p 549, 1958.

[61] Kutteh, R.; Straatsma, T.P.; Reviews in Computational Chemistry **12**, 1998.

[62] Hansen, J.P.; McDonald, I.R.; *Theory of Simple Liquids* 2^{nd} edition, Academic Press, 1986.

[63] Erdahl, R.M.; Smith, V.H. Jr., Eds. *Density Matrices and Density Functionals*; Reidel: Dordrecht, 1987.

[64] Parr, R.G.; Yang, W. *Density Functional Theory of Atoms and Molecules*; Oxford University Press: New York, 1989.

[65] Verwoerd, W.S; Surface Science **103**, pp 404 - 415, 1981.

[66] Carter, L.E.; Carter E.A.; Surf. Sci. **323**, pp 39-50, 1995.

[67] Chadi D.J.; J. Vac. Sci. Technol. **16**, p 1290, 1979.

[68] Chadi D.J.; Phys. Rev. Lett. **59**, p 1691, 1987.

[69] Dabrowski J.; Scheffler M.; Appl. Surf. Sci. **15**, pp 56-58, 1992.

[70] Hamers R.J.; Tromp R.M.; Demuth J.E.; Phys. Rev. B **34**, p 5343, 1986.

[71] Ihara S.; Ho S.L.; Uda T.; Hirao M.; Phys. Rev. Lett. **65**, p 1909, 1990.

[72] Northrup J.E.; Phys. Rev. Lett.**54**, p 815, 1985.

[73] Pandey K.C.; *Proceedings of the Seventeenth International Conference on the Physics of Semiconductors*; Springer-Verlag, New York, p 55 , 1985.

[74] Poppendieck T.D.; Ngoc T.C.; Webb M.B.; Surf. Sci. **75**, p 287, 1978.

[75] Ramstad A.; Brocks G.; Kelly P.J.; Phys. Rev. B **51**, 14504, 1995.

[76] Roberts N.; Needs R.J.; Surf. Sci. **236**, 112, 1990.

[77] Schlier R.E.; Farnsworth H.E.; J. Chem. Phys. **30**, p 917, 1959.

[78] Tromp R.M.; Hamers R.J.; Demuth J.E.; Phys. Rev. Lett. **55**, p 1303, 1985.

[79] Wolkow R.A.; Phys. Rev. Lett. **68**, p 2636, 1992.

[80] Boland J.J.; Phys. Rev. B **44**, p 1383, 1991.

[81] Boland J.J.; Phys. Rev. Lett. **67**, p 1539, 1991.

[82] Boland J.J.; J. Vac. Sci. Technol. A **10**, p 2458, 1992.

[83] Boland J.J.; Advances in Physics **42**, p 129, 1993.

[84] Eaglesham D.J.; Unterwald F.C.; Luftman H.; Adams D.P.; Yalisove S.M.; J.Appl.Phys. **74**, p 6615, 1993.

[85] Chen, J.C.; *Introduction to Scanning Tunneling Microscopy*; Oxford University Press, 1993.

[86] Sinniah K.; Sherman M.G.; Lewis L.B.; Weinberg W.H.; Yates, J.T. Jr.; Janda K.C.; Phys. Rev. Lett. **62**, p 567, 1989.

[87] Vittadini A.; Selloni A.; Casarin M.; Phys. Rev. B. **49**, p 11 191, 1993.

[88] Widdra W.; Yi S.I.; Maboudian R.; Briggs G.A.D.; Weinberg W.H.; Phys. Rev. Lett. **74**, p 2074, 1995.

[89] Wu C.J.; Carter E.A.; Phys. Rev. B **46**, p 4651, 1992.

[90] Mokler, S.M.; Liu W.K.; Ohtani N.; Joyce B.A.; J. Cryst. Growth **120**, p 290, 1992.

[91] Kohen, D; Tully, J; Stillinger, H; *Modeling the interaction of hydrogen with silicon surfaces*; Surface Science **397**, pp 225-236, 1998.

[92] Hehre, W.J.; Radom, L.; Schleyer, P.v.R.; Pople, J.A.; *Ab Initio Molecular Orbital Theory*; John Wiley and Sons: New York, pp 22-24, 1986.

[93] Güntert, P.; Mumenthaler, C.; Wüthrich, K.; J. Mol. Biol. **273**, 1997.

[94] Warshell, A.; Levitt, M.; J. Mol. Biol. **103**, pp 227-249, 1976.

[95] des Cloizeanux, J.; Phys. Rev. **135**, p 685. 1964.

[96] Kohn, W.; Chem. Phys. Lett. **208**, p 167, 1993.

[97] Ismail-Beigi, S.; Arius, T.; Phys. Rev. Lett. **82**, p 2127, 1999.

[98] Gordon, B.; *Lectures on Quantum Mechanics*; The Benjamin Cummings Publishing Company Inc., p 424, 1969.

[99] Kirkpatrick, S.; Gelat, C.D.; Vecchi, M.P.; Science **220**, p 671, 1983.

[100] Ceperly, D.; Alder, B.; Science **231**, p 555, 1986.

[101] McWeeny, R.; Sutcliffe, B.T. *Methods in Molecular Quantum Mechanics*; Academic Press: London,p 129, 1969.

[102] Pulay, P.; J. of Comp. Chem. **3**, p 556, 1982.

[103] Graybill, F.A.; *Matrices with Applications in Statistics, Second Edition*; Wadsworth, Inc.: Belmont, p 309, 1983.

[104] Graybill, F.A. *An Introduction to Linear Statistical Models, Vol 1*; McGraw-Hill: New York, p 12, 1961.

[105] Quick Reference Guide to Computational Chemistry Acronyms, Methods, Parameters, Accuracy, and Software, Chamot Labs, 1999
http://www.chamotlabs.com/Freebies/Table/Methods/SemiEmpirical.htmlMINDO/3

[106] Silicon Surfaces, University of Nottingham Science Group, 2000
(http://www.nottingham.ac.uk/ ppzstm/surfaces)

[107] MOPAC Manual, Fujitsu Limited, 2002
(http://www.cachesoftware.com/mopac/Mopac2002manual.html)

[108] Wilson, N.T.; PhD Thesis, University of Birmingham, 2002
(http://www.hpc.susx.ac.uk/ nickw/research/bham.ac.uk/PhD/node28.html)

[109] Molecular dynamics, October 1996
(http://cmt.dur.ac.uk/sjc/thesis/thesis/node27.html#SECTION00390000000000000000)