

VMC CALCULATIONS OF TWO-DIMENSIONAL QUANTUM DOTS

by

LARS EIVIND LERVÅG

THESIS

for the degree of

MASTER OF SCIENCE

(Master in Computational Physics)



*Faculty of Mathematics and Natural Sciences
Department of Physics
University of Oslo*

November 2010

*Det matematisk- naturvitenskapelige fakultet
Universitetet i Oslo*

Acknowledgements

First and foremost, my supervisor Morten Hjorth-Jensen is owed a huge thanks. You have been a big support during the last two-years, always optimistic, and believing in me; being a source of much inspiration and enthusiasm. Thanks for providing an interesting topic, and for much advice and good discussions.

My fellow students, Håvard Sandsdalen, Magnus Pedersen Lohne and Sigurd Wenner. Thank you for friendship during the last two years. It is good to have people to talk to when the work seems endless, and people to discuss problems with.

My family should also be thanked. My mother and father, for believing in me and supporting me. Thanks to my brother Karl Yngve Lervåg for a lot of help with proof-reading this thesis. I would also like to thank my brothers Alf and Karl Yngve Lervåg for help with emacs and LaTeX, and my brother Jon Vegard Lervåg and his wife Stine Walderhaug for continuous support.

In addition, I have had some help with object orienting the code, as I did not have any previous knowledge of this in C++. My thanks go out to Kyrre Ness Sjøbæk.

Lars Eivind Lervåg

Contents

1	Introduction	9
I	Theory	11
2	Quantum Mechanics	13
2.1	Historical overview	13
2.1.1	Planck's law of radiation	14
2.1.2	The photoelectric effect	15
2.2	Bra-Ket notation	16
2.3	The fundamental postulates of quantum mechanics	16
2.3.1	Postulate 1	16
2.3.2	Postulate 2	17
2.3.3	Postulate 3	18
2.3.4	Postulate 4	19
2.3.5	Postulate 5	19
2.4	Quantum Mechanics in the Schrödinger Picture	20
2.5	Singe-Particle Quantum Mechanics	21
2.5.1	Spin	22
2.5.2	Total single-particle wave function	24
2.6	The particle in a box	25
3	Many-Body Theory	29
3.1	The Many-Body Problem	29
3.2	Identical particles	31
3.3	The Non-Interacting System	34
3.3.1	The ground state of the non-interacting system	35
3.4	The Interacting System	36
4	Quantum Dots - The Artificial Atoms	37
4.1	Semiconductors and quantum dots	38
4.2	Electrical properties resulting from quantum confinement	40
4.3	Optical properties resulting from quantum confinement	41
4.4	Applications of quantum dots	43
5	Theoretical approximations to quantum dots in two dimensions	47
5.1	One-electron quantum dot	48
5.2	Two-electron quantum dot	55
5.3	The N -electron Hamiltonian	59

5.4	Scaling the N -electron Hamiltonian	60
6	Quantum Monte Carlo	61
6.1	Random numbers	62
6.1.1	Generation of uniform random numbers	63
6.1.2	Generation of Gaussian distributed random number	63
6.2	Monte Carlo integration	63
6.2.1	Importance sampling	65
6.3	Markov chains and the Metropolis-Hastings algorithm	66
6.4	The variational principle	69
6.5	The quantum variational Monte Carlo method (QVMC)	71
6.5.1	Calculating the energy and variance of our system	71
6.5.2	Simple Metropolis sampling	73
6.5.3	Generalized Metropolis sampling	73
6.6	Energy minimization using DFP	76
6.7	Blocking	78
6.8	Time-step extrapolation	79
7	The trial wave function	81
7.1	Slater determinants	81
7.1.1	Splitting up our Slater determinant	83
7.1.2	Single-particle orbitals	84
7.1.3	Derivatives of the single-particle orbitals	84
7.2	The Jastrow function	86
7.2.1	Cusp conditions	87
7.3	The total trial wave function	89
II	Implementation and results	91
8	Implementation of the QVMC method	93
8.1	Optimizing the calculations	93
8.1.1	The R_D -ratio	95
8.1.2	The R_J -ratio	97
8.1.3	The G_D -ratio	98
8.1.4	The G_J -ratio	99
8.1.5	The L_D -ratio	100
8.1.6	The L_J -ratio	101
8.2	Class implementation	102
8.2.1	The initialization of the program	102
8.2.2	Implementation of the Metropolis-Hastings algorithm	107
8.3	Parallel computing	112
8.4	Implementing the DFP algorithm	113
8.5	Implementation of blocking	114
8.6	Implementation of time-step extrapolation	116

9	Computational results	117
9.1	Validating the code	117
9.1.1	The non-interacting system	117
9.1.2	The two-particle case	118
9.2	Variational plots — Finding the optimal parameters	122
9.2.1	Simulation of a six-electron quantum dot	122
9.2.2	Simulation of a twelve-electron quantum dot	122
9.2.3	Simulation of a twenty-electron quantum dot	124
9.2.4	Minimization with DFP	124
9.3	Time-step analysis with blocking	127
9.3.1	Two electrons	127
9.3.2	Six electrons	127
9.3.3	Twelve electrons	129
9.3.4	Twenty electrons	130
9.4	Changing the oscillator frequency	130
9.5	Splitting up the energy	130
9.6	Discussions of the results	132
9.7	Quantum dots with more than twenty electrons	134
10	Conclusion	137
A	Statistics	141
	Bibliography	149

Chapter 1

Introduction

In this thesis we study systems consisting of several interacting electrons. More specific, we study systems named quantum dots. Quantum dots are in the literature often dubbed artificial atoms, due to their many similarities with atoms, even though they do not have a nucleus. Quantum dots are of interest because of their many applications in modern technology. Furthermore they are fundamentally interesting because they allow us to study electronic systems without the presence of a nucleus that affects the electrons. Moreover, they allow us to easily study effects of quantum confinement. We will give a longer introduction and motivation of quantum dots in Chapter 4.

The aim of this thesis is to do numerical studies of closed-shell quantum dots, using the Quantum Variational Monte Carlo (QVMC) *ab initio* method (See Chapter 5 and 6 for details). We consider the so-called parabolic (or circular) quantum dot in two dimensions. An object of essential importance for the QVMC method is the *trial wave function*. This will be our guess at how the functional form of the true many-body wave function looks like. In order to construct a good trial wave function, we need to understand the physics that govern the system of interest. In this thesis we use a Jastrow-Slater wave function (see Chapter 7 for details). We want to see if a trial wave function with only one Slater determinant (see Chapter 3) is a good approximation for a closed-shell system. We aim to develop a code that can do computations on closed-shell quantum dots with more than twenty electrons. As far as we know, there has not been done any *ab initio* calculations on systems with this many particles.

Another aim is to develop a good QVMC machinery for studying electronic systems. We will not actually do calculations for other systems in this work, but we know that when the code works for the quantum dot system, it can easily be changed to work for atomic or molecular systems.

Overview

The thesis is structured into two main parts. In the first part we present the theoretical foundation of this thesis. We have organized this into six chapters:

- Chapter 2 gives a short review of non-relativistic quantum mechanics, including the fundamental postulates of the theory. We will focus on single-particle theory, and give a short example of a quantum mechanical system; the infinite square well.

- In Chapter 3 we will present non-relativistic many-body theory; quantum mechanics for systems consisting of more than one particle. We will present the insolvable many-body problem. Then we discuss the basic properties a fermionic wave function must exhibit. We will solve the non-interacting system, which introduces the important concept of Slater determinants. We will briefly discuss the interacting system.
- Chapter 4 gives a presentation of quantum dots; what they are, and why we like them. We will present some theory of semiconductors and quantum dots such as electronic band structure and excitations. We will then briefly go through some of the interesting applications of quantum dots.
- In Chapter 5 we solve the one-particle problem related to the parabolic quantum dot. This is of importance when we construct the Slater determinant of the trial wave function. We will also solve the two-particle problem for a specific case. We then construct the many-particle Hamiltonian and scale it to a dimensionless form.
- Chapter 6 presents the theory of quantum variational Monte Carlo (QVMC) methods. We first discuss random numbers and pseudo-random numbers. We then discuss the basic idea behind Monte-Carlo integration and importance sampling. We will discuss the theory of Markov chains which leads us to the important Metropolis-Hastings algorithm. We then present the variational principle. This principle is what actually enables us to do QVMC in the first place. We are then ready to discuss the QVMC method. Finally we will discuss some methods used to improve the results: The Davidson-Fletcher-Powell (DFP) algorithm, blocking and time-step extrapolation (see sections 6.6 - 6.8).
- Chapter 7 deals with constructing our trial wave function. We will construct our Slater determinant out of the single-particle orbitals from Chapter 5. We then discuss the correlation function used in this thesis, the Pade-Jastrow function. Finally, we give the functional form of our trial wave function.

In the second part of this thesis, we explain the implementation of the QVMC method, and present our numerical results. This part is organized in three chapters:

- In Chapter 8 we present the structure of the code developed in this thesis. We have developed an entire QVMC machinery for solving closed-shell parabolic quantum dot systems. We will explain in detail how the QVMC algorithm is implemented. We also go through the implementation of the DFP, blocking and time-step extrapolation techniques.
- In Chapter 9 we present our results. We will mostly consider parabolic quantum dots with two, six, twelve and twenty particles. The results are discussed and analyzed. We will also present results for 30 and 42 electrons.
- In Chapter 10 we conclude the thesis.

Part I
Theory

Chapter 2

Quantum Mechanics

Mechanics is the field of physics concerned with the behaviour of physical bodies under the influence of forces, and how these bodies affect their environment. There are two major sub-fields: *Classical mechanics* and *quantum mechanics*. The first field is used for describing the dynamics of macroscopic objects. The second field is used for describing the dynamics of microscopic objects.

It is assumed that the reader of this thesis is well acquainted with the fundamentals of quantum physics. Even so, we feel it is important to give a quick review of the basic aspects of the theory in order to clarify some of the notation and to help freshen up the reader's memory.

Put simply, quantum mechanics is a theoretical framework which has been able to describe, correlate and predict the behaviour of a vast number of physical systems. In this chapter, we first give a short historical overview explaining why quantum mechanics was needed. We then move on to describe some of the quantum mechanical formalism. We will go through the fundamental postulates of quantum mechanics, and give a brief explanation to each of them. We then move on to the formalism of quantum mechanics in the Schrödinger picture, and go through the basic aspects of quantum mechanics for a single-particle system. We conclude with discussing and solving the *particle in a box*, a famous and fundamental problem in quantum mechanics. If the reader is in need of a deeper introduction to the field, we refer to [1, 2].

2.1 Historical overview

The theoretical description of Nature called quantum mechanics was created by European physicists in the early 20th century. It was an intellectual revolution that introduced several concepts that had been totally unknown in the classical theory of physics:

- **Quantization:** Many physical quantities can only take certain discrete values.
- **Wave-particle dualism:** A material particle and electromagnetic radiation is not as different as in classical physics. They both show particle- and wave-like properties.
- **Probability interpretation:** The quantum mechanical description can only give the probability for finding a particle at a certain place.

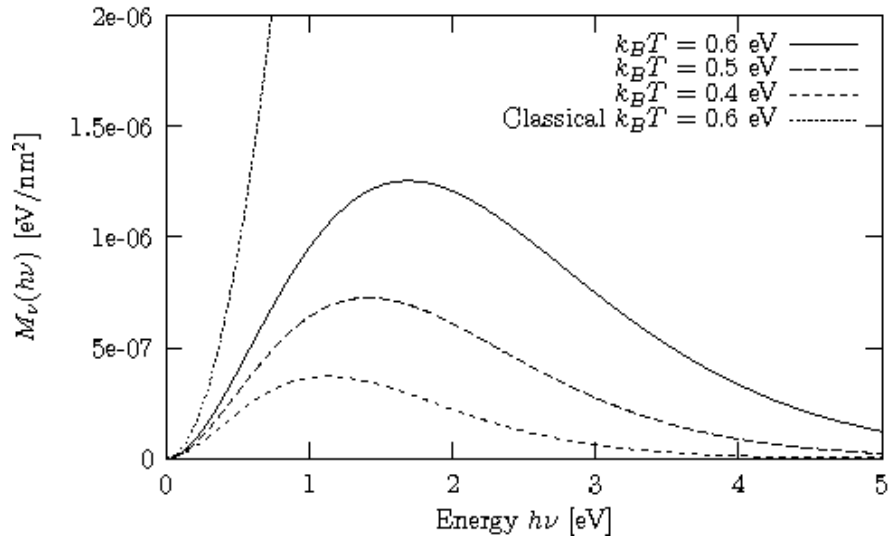


Figure 2.1: Shows the experimental spectral radiation emittance for a black body at different temperature. The classical result is also plotted. We see that the classical result for $k_B T = 0.6\text{eV}$ deviates fundamentally from the experimental result for the same temperature.

- **Uncertainty principle:** Nature puts fundamental limitations on the precision certain physical quantities may be measured with.
- **Annihilation and creation:** Every particle may be annihilated and created.

Quantum mechanics was needed because a number of experimental results were fundamentally inconsistent with classical physics. We will only give a very short introduction of a couple of these problems, and the way two brilliant physicists was able to resolve those problems.

2.1.1 Planck's law of radiation

Every body radiates electromagnetic radiation. The intensity and frequency distribution of the radiation depends on the temperature of the body. In Figure 2.1 we see the experimental result of black body radiation. A black body is an idealized object that absorbs all electromagnetic radiation falling on it. Such a body will also re-emit radiation in a characteristic pattern called a spectrum.

We will not go into detail on the calculation in this short introduction. Suffice it to say that the classical calculations for the spectral emittance results in a divergence such as we see in Figure 2.1. This was labelled the ultraviolet catastrophe. If the classical theory was correct, the total energy current from a black body (over all frequencies) would be infinite!

Max Planck was able to come up with a formula that fit with the experimental results

$$I(\nu, T) = \frac{2\pi\nu^2}{c^2} \frac{h\nu}{e^{\frac{h\nu}{k_B T}} - 1},$$

where $I(\nu, T)$ is the spectral emittance, ν is the frequency, T is the temperature, c is the speed of light, k_B is Boltzmann's constant and h was a new constant which was later

named Planck's constant. The new constant had to be about $6.6 \cdot 10^{-34}$ Js to get agreement with the data.

Planck was able to later explain this empirical formula by assuming that the energy for radiation with frequency ν was quantized in units $h\nu$, i. e. only taking discrete values

$$E = 0, h\nu, 2h\nu, 3h\nu, \dots$$

This energy quantizing was a totally new idea, and it marked the beginning of quantum physics.

Planck himself never fully accepted that this was a revolutionary idea. To him it was only "an act of desperation." Even so, Planck received the Nobel prize in 1918 for "introducing the quantum" as a real physical entity.

2.1.2 The photoelectric effect

The photoelectric effect is a phenomenon in which electrons are emitted from matter as a consequence of their absorption of energy from electromagnetic radiation. It was first observed by Heinrich Hertz in 1887. Experiments showed that if the frequency was below a certain limit ν_0 , no electrons would be emitted. Experiments also showed that the kinetic energy of the electrons that were emitted was independent of the intensity of the radiation, but varied linearly with the frequency of the radiation:

$$E_{kin} = h(\nu - \nu_0), \quad \nu > \nu_0.$$

That electrons may be emitted from matter by electromagnetic radiation was understandable in the classical theory. But in the classical theory energy is proportional to the intensity of the radiation. The frequency dependency was therefore baffling for the physicists in the 19th century.

Albert Einstein solved the problem by assuming that light consists of discrete quanta, photons, with energy

$$E = h\nu,$$

determined by the frequency. When an electron absorbs a photon, the energy of the electron increases by $h\nu$. Some of this energy, W , is needed to liberate the electron from the matter, and the rest is just converted to kinetic energy:

$$E_{kin} = h\nu - W.$$

This is consistent with experiments: The kinetic energy of the electrons varies linearly, with a cutoff frequency $\nu_0 = W/h$.

There were several other problems which classical physics could not resolve, like the Compton-effect, diffraction of electrons and the heat capacity of a diatomic gas. A deeper discussion of these topics can be found in ref. [1].

From the two cases discussed above, we see that some of the problems that classical physics could not explain was resolved by "introducing the quantum." This was, of course, why the new physical theory was labelled quantum physics.

2.2 Bra-Ket notation

To make the notation more concise, we introduce some standard notation. Integrals often take up a lot of space in formulas, and as they are very much used in quantum mechanics through expressions for expectation values, it is very useful to introduce a compact way to write such integrals. Say we have a state Ψ . This state can be expressed through the state vector $|\Psi\rangle$, also called a *ket*. The dual state (complex conjugate) is then represented by $\langle\Psi|$, called a *bra*.

We now define the expectation value of an operator \hat{O} through

$$\langle\hat{O}\rangle = \int \Psi^* \hat{O} \Psi dx \quad (2.1)$$

$$\equiv \langle\Psi|\hat{O}|\Psi\rangle. \quad (2.2)$$

For an arbitrary operator we have that

$$\langle\Psi|\hat{O}|\Psi\rangle = \langle\hat{O}^\dagger\Psi|\Psi\rangle \equiv \langle\Psi|\hat{O}|\Psi\rangle, \quad (2.3)$$

where the dagger means

$$\hat{O}^\dagger \equiv \hat{O}^{*T}, \quad (2.4)$$

i. e. the operators Hermitian conjugate. The standard notation in quantum mechanics for the expectation value is then

$$\langle\hat{O}\rangle = \langle\Psi|\hat{O}|\Psi\rangle. \quad (2.5)$$

2.3 The fundamental postulates of quantum mechanics

Every fundamental physical theory is based on a set of postulates, quantum mechanics being no exception. In this section we present the postulates of quantum mechanics, and give a brief explanations on each of them.

Note that different learning books on the subject presents the postulates in different order, and with different phrasing. In the following we follow closely the text of ref. [2].

2.3.1 Postulate 1

The physical state Ψ of an isolated quantum mechanical system is represented by a vector $|\Psi\rangle$ in a Hilbert space.

Notes on postulate 1: A Hilbert space, \mathcal{H} , is a complex vector space with an inner product. It is also complete. In the bra-ket formalism, for every quantum state $|\Psi\rangle$ in a Hilbert space, there exists a dual state $\langle\Psi|$ in a dual vector space. An inner product is a structure which associates each pair of vectors in the space with a scalar quantity called the inner product of the vectors. In the bra-ket formalism we denote the inner product.

$$\langle\alpha|\beta\rangle. \quad (2.6)$$

It satisfies the following three axioms for all vectors $|\alpha\rangle, |\beta\rangle, |\nu\rangle \in \mathcal{H}$ and all scalars $c \in \mathbb{C}$:
Conjugate symmetry:

$$\langle\alpha|\beta\rangle = \langle\alpha|\beta\rangle^*. \quad (2.7)$$

Linearity in the right argument:

$$\langle \alpha | c\beta \rangle = c \langle \alpha | \beta \rangle \quad (2.8)$$

$$\langle \alpha | \beta + \nu \rangle = \langle \alpha | \beta \rangle + \langle \alpha | \nu \rangle. \quad (2.9)$$

Positive definiteness:

$$\langle \alpha | \alpha \rangle \geq 0. \quad (2.10)$$

For more details into linear algebra, we refer to [3].

Assume we have a discrete basis which is orthonormal

$$\langle i | j \rangle = \delta_{mn}, \quad (2.11)$$

and complete

$$\sum_i |i\rangle \langle i| = \mathbb{1}, \quad (2.12)$$

with δ_{mn} being the Kronecker delta, and $\mathbb{1}$ the identity operator. The quantum state can then be developed in this basis

$$|\psi\rangle = \sum_i |i\rangle \langle i|\Psi\rangle = \sum_i c_i |i\rangle, \quad (2.13)$$

where $c_i \equiv \langle i|\Psi\rangle$ is a complex number. If the basis is continuous, the orthonormality condition is given as

$$\langle x|x'\rangle = \delta(x - x'), \quad (2.14)$$

with $|x\rangle$ being a position basis vector. The completeness relation is given as

$$\int dx |x\rangle \langle x| = \mathbb{1}. \quad (2.15)$$

The quantum state can then be written

$$|\Psi\rangle = \int dx |x\rangle \langle x|\Psi\rangle = \int dx \Psi(x) |x\rangle, \quad (2.16)$$

with $\Psi(x) \equiv \langle x|\Psi\rangle$.

2.3.2 Postulate 2

To each observable F there is in quantum mechanics a Hermitian linear operator \hat{F} . If an observable in classical mechanics takes the form $F(q_1, q_2, \dots, q_f, p_1, p_2, \dots, p_f)$, the operator in quantum mechanics is $\hat{F}(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_f, \hat{p}_1, \hat{p}_2, \dots, \hat{p}_f)$, where the operators representing the generalized momentum p_n and generalized coordinate q_n are

$$\hat{p}_n = \frac{\hbar}{i} \frac{\partial}{\partial q_n} \quad \text{and} \quad \hat{q}_n = q_n. \quad (2.17)$$

They satisfy the commutation relation

$$[\hat{p}_n, \hat{q}_n] = i\hbar, \quad (2.18)$$

where \hbar is the reduced Planck constant, defined as

$$\hbar \equiv \frac{h}{2\pi}, \quad (2.19)$$

where h is Planck's original constant.

Notes on postulate 2: There are quantum mechanical operators that cannot be constructed from a classical expression for the physical entity. *Spin* is the most important example.

The expectation value of an observable can be expressed very neatly in inner-product notation as shown in eq. (2.5). The outcome of a measurement has to be real. This can be expressed through

$$\langle O \rangle = \langle O \rangle^*. \quad (2.20)$$

This means that

$$\langle \Psi | \hat{O} \Psi \rangle = \langle \hat{O} \Psi | \Psi \rangle, \quad (2.21)$$

or, expressed in another way

$$\hat{O} = \hat{O}^\dagger. \quad (2.22)$$

An operator having this quality is called Hermitian. All observables in quantum mechanics are represented by Hermitian operators.

There is a theorem called *the spectral theorem* (see ref. [3]), which states that the eigenvectors of a Hermitian operator \hat{A} in a Hilbert space form a complete set in that Hilbert space, and that its eigenvalues must be real. The completeness is expressed through

$$\hat{I} = \sum_i^d |a_i\rangle \langle a_i|, \quad (2.23)$$

with $|a_i\rangle$ being the eigenvectors of \hat{A} , and d the dimensionality of the vector space. The operator can then be expressed through its spectral decomposition

$$\hat{A} = \sum_i^d a_i |a_i\rangle \langle a_i|. \quad (2.24)$$

2.3.3 Postulate 3

The time evolution of the quantum state of a system is (in the Schrödinger picture) fully described by a time dependent state vector $\Psi(t)$, called a wave function. This wave function obeys the Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle, \quad (2.25)$$

with \hat{H} as the Hamiltonian of the system.

Notes on postulate 3: We define the Hamiltonian (in one dimension) as

$$\hat{H} = -\frac{\hbar}{2m} \frac{\partial^2}{\partial x^2} + V(x). \quad (2.26)$$

What exactly is the wave function? In classical physics we are used to think of a particle as being localized at a point. The wave function, however, is spread out in space. How do we interpret such an object? The answer is provided by Born's *statistical interpretation* of the wave function, which states that $|\Psi(x, t)|^2$ gives the *probability* of finding the particle at point x , at time t . More precisely:

$$\int_a^b |\Psi(x, t)|^2 dx = \begin{cases} \text{probability of finding the particle} \\ \text{between } a \text{ and } b, \text{ at time } t. \end{cases} \quad (2.27)$$

An important concept tied to the wave function is *normalization*. Since we interpret $|\Psi(x, t)|^2$ as a probability density, it follows that

$$\int_{-\infty}^{+\infty} |\Psi(x, t)|^2 dx = 1. \quad (2.28)$$

Without this, the statistical interpretation would be nonsense. This extra condition on the wave function is called *normalizing* the wave function. Any physically realizable state has to be normalizable. It follows that any solution to the Schrödinger equation that is non-normalizable, cannot represent a real physical state. Such a solution will thereby be rejected.

It is a lucky fact that if we normalize the wave function at time $t = 0$, the wave function will still be normalized as time passes by. This is a remarkable property of the Schrödinger equation, and without it the whole quantum theory would have collapsed.

2.3.4 Postulate 4

If a system is in a state $|\Psi\rangle$ and a measurement of the observable A is done, the result will be limited to one of the eigenvalues of the operator \hat{A} .

Notes on postulate 4: Assume the system is in a quantum state Ψ , and we do a measurement on the system. The result of the measurement will be limited to an eigenvalue a_i , which will appear with probability

$$p_i = \sum_{n=1}^l |\langle a_{i,n} | \Psi \rangle|^2, \quad (2.29)$$

where l denotes the degeneracy of the eigenvalue, i. e. how many eigenvectors share the same eigenvalue. In the non-degenerate case we have $l = 1$. If the eigenvalue x of an operator \hat{x} is a continuous variable, p_i will be a probability density.

2.3.5 Postulate 5

In an ideal measurement, when the measured value of an observable \hat{A} is a_i , the quantum state is immediately changed to the corresponding eigenstate.

$$|\Psi\rangle \rightarrow |a_i\rangle. \quad (2.30)$$

Notes on postulate 5: This is what is usually referred to as *the collapse of the wave function*.

Furthermore, if two operators \hat{Q} and \hat{P} are not compatible, $[\hat{Q}, \hat{P}] \neq 0$, there is no measurement that can precisely determine both Q and P simultaneously. There will then be a relation between them that relates how precise we can know each observable. This is the famous uncertainty principle:

$$\sigma_Q^2 \sigma_P^2 \geq \left(\frac{1}{2i} \langle [\hat{Q}, \hat{P}] \rangle \right)^2, \quad (2.31)$$

where σ_Q is the uncertainty in the measurement of observable Q and σ_P is the uncertainty in the measurement of observable P . The most well known example is the Heisenberg uncertainty principle

$$\sigma_x \sigma_p \geq \frac{\hbar}{2}, \quad (2.32)$$

which relates the position and momentum observables. This principle states that if we want to know the exact position of a particle, it comes at the expense of not knowing a thing about the momentum of the same particle.

2.4 Quantum Mechanics in the Schrödinger Picture

We solve problems in quantum mechanics using vectors and operators, which are quite abstract objects. This has led to several different pictures in which to view quantum mechanics, and several different representations of the vectors and operators, depending on which basis one wishes to use. For more information we refer to [4].

In the Schrödinger picture, the time evolution of the system is defined by the Schrödinger equation (eq. (2.25)). Originally this was formulated as a wave equation (here in one dimension)

$$i\hbar \frac{\partial \Psi}{\partial t} = -\frac{\hbar^2}{2m} \frac{\partial^2 \Psi}{\partial x^2} + V(x)\Psi, \quad (2.33)$$

with $V(x)$ as an arbitrary potential. Equation (2.33) can be reformulated as a differential equation in the abstract Hilbert space of ket-vectors such as in eq. (2.25).

Since the Schrödinger equation is a linear differential equation with a first order time derivative, if we know the quantum state at one time t_0 , then the state is uniquely determined for all other times t , as long as the system stays isolated. The information about the dynamics of the system is contained in the Hamiltonian \hat{H} , which usually can be identified as the energy observable of the system.

The dynamical evolution of the state vector can be expressed in terms of a *time evolution operator* $\hat{U}(t, t_0)$, which is a unitary operator that relates the state vector of the system at time t with that of time t_0 ,

$$|\Psi(t)\rangle = \hat{U}(t, t_0) |\Psi(t_0)\rangle. \quad (2.34)$$

The time evolution operator is determined by the Hamiltonian through

$$i\hbar \frac{\partial}{\partial t} \hat{U}(t, t_0) = \hat{H} \hat{U}(t, t_0). \quad (2.35)$$

If \hat{H} is a time-independent operator, we get a closed form expression for the time evolution operator

$$\hat{U}(t, t_0) = e^{i\hat{H}(t-t_0)/\hbar}. \quad (2.36)$$

Furthermore, when \hat{H} is time-independent, we can solve the Schrödinger equation by means of *separation of variables*. We will not go into the details in this thesis, but if the reader is interested we refer to [1] for an explanation of the process. The method of separation of variables leads us to what is called the *time-independent Schrödinger equation*:

$$\hat{H} |\psi_i\rangle = \epsilon_i |\psi_i\rangle. \quad (2.37)$$

The general solution of the Schrödinger equation can be obtained as a *linear combination* of the solutions of eq. (2.37), because these solutions form a complete basis of the Hilbert space

$$\hat{I} = \sum_i |\psi_i\rangle \langle \psi_i|. \quad (2.38)$$

2.5 Singe-Particle Quantum Mechanics

This thesis involves systems containing several particles. But before we move on to such systems, it is instructive to look at systems consisting of only one particle.

We consider an isolated single-particle system with Hamiltonian

$$\hat{H} = \hat{T} + \hat{V}, \quad (2.39)$$

with \hat{T} as the kinetic energy operator (in one dimension)

$$\hat{T} = \frac{\hat{p}^2}{2m} = -\frac{\hbar}{2m} \frac{\partial^2}{\partial x^2}, \quad (2.40)$$

and \hat{V} as an arbitrary potential. The dynamics of the system is provided by the Schrödinger equation (eq. (2.25)), and if the Hamiltonian is time-independent, the time evolution operator reads as in eq. (2.36).

We can then use the solutions of the time-independent Schrödinger equation (eq. (2.37)) to obtain an analytical expression of $|\Psi(t)\rangle$. The initial state vector can be written as

$$|\Psi(t_0)\rangle = \sum_i \langle \psi_i | \Psi(t_0) \rangle |\psi_i\rangle. \quad (2.41)$$

We then write the general wave function as

$$\begin{aligned} |\Psi(t)\rangle &= \hat{U}(t, t_0) |\Psi(t_0)\rangle \\ &= e^{i\hat{H}(t-t_0)/\hbar} \sum_i \langle \psi_i | \Psi(t_0) \rangle |\psi_i\rangle \\ &= \sum_i \langle \psi_i | \Psi(t_0) \rangle e^{i\epsilon_i(t-t_0)/\hbar} |\psi_i\rangle. \end{aligned} \quad (2.42)$$

Given an initial state vector and a time-independent Hamiltonian, we can, in principal, always determine the state vector at time $t > t_0$ through eq. (2.42). The problem then lies in solving the time-independent Schrödinger equation, and finding the weights $\langle \psi_i | \Psi(t_0) \rangle$.

2.5.1 Spin

In the classical theory of central forces, energy and angular momentum are the fundamental conserved quantities. We have already seen how to find the energy of a quantum mechanical system. It is not surprising that angular momentum also plays a significant role in quantum theory.

In classical mechanics, a rigid object admits two kinds of angular momenta. The first one is the *orbital* momentum, defined as

$$\mathbf{L} = \mathbf{r} \times \mathbf{p}, \quad (2.43)$$

where \mathbf{r} is the position vector, and \mathbf{p} is the momentum vector. Orbital momentum is associated with the motion of the center of mass of the object in question. In quantum mechanics, orbital momentum is associated with the motion of particles in space. The quantum theory of orbital momentum is derived from the classical theory in a straight forward fashion, but also shows quantum effects of profound importance. We will not, however, go into detail here. If the reader is interested we refer to [1, 2]

The second type of angular momentum is the *spin*

$$\mathbf{S} = \mathbf{I}\omega, \quad (2.44)$$

where \mathbf{I} is the moment of inertia, and ω is the angular velocity. In classical theory, the distinction between orbital momentum and spin is just a matter of convenience. Spin is really just the sum total of the orbital angular momenta of all the individual parts of the object as the parts circle around the objects axis. There is an analogue to spin in quantum mechanics, and here the distinction between spin momenta and orbital momenta is absolutely fundamental. Consider the case of a hydrogen atom. In addition to orbital angular momentum, associated with the motion of the electron around the nucleus, the electron also carries another form of angular momentum, which has nothing to do with motion in space. As far as we know, the electron is a structureless point particle and its spin angular momentum cannot be decomposed into orbital angular momenta of constituent parts. We therefore say that elemental particles carry *intrinsic* angular momentum, called spin, in addition to the extrinsic angular momentum, \mathbf{L} .

The algebraic theory of spin is identical to the theory of orbital momentum (see ref. [1] or ref. [2]). We start with the fundamental commutation relations:

$$\left[\hat{S}_x, \hat{S}_y \right] = i\hbar \hat{S}_z, \quad \left[\hat{S}_y, \hat{S}_z \right] = i\hbar \hat{S}_x, \quad \left[\hat{S}_z, \hat{S}_x \right] = i\hbar \hat{S}_y. \quad (2.45)$$

The eigenvectors of \hat{S}^2 and \hat{S}_z satisfy ref. [1]

$$\hat{S}^2 |s, m_s\rangle = \hbar^2 s(s+1) |s, m_s\rangle, \quad (2.46)$$

$$\hat{S}_z |s, m_s\rangle = \hbar m |s, m_s\rangle, \quad (2.47)$$

where s is the principal spin quantum number, and m_s is the quantum number associated with the z -projection of the spin.

Since the different components of the spin does not commute (have no common set of eigenfunctions), they are incompatible observables. This means that we cannot determine

two components at the same time. However, we can determine one component and \hat{S}^2 simultaneously, so the components share eigenfunctions with \hat{S}^2 . In the literature, one generally choose the z -component, which is what we will do.

The spin quantum numbers can take on the values:

$$s = 0, \frac{1}{2}, 1, \frac{3}{2}, 2, \dots \quad (2.48)$$

$$m_s = -s, -s + 1, \dots, s - 1, s. \quad (2.49)$$

Every elemental particle has a specific and fixed value of s , which we call the spin of that particular species: π -mesons have spin 0; electrons have spin $\frac{1}{2}$; photons have spin 1; Δ -isobars have spin $\frac{3}{2}$; gravitons, if they exist, must have spin 2; and so on.

In the rest of this section, we consider the spin $\frac{1}{2}$ case. This is by far the most important case, for this is the spin of the particles that make up ordinary matter (protons, neutrons, and electrons), as well as all quarks and all leptons. Since $s = \frac{1}{2}$, we have

$$m_s = \pm \frac{1}{2}. \quad (2.50)$$

The measured value of \hat{S} will be

$$\hat{S}^2 \left| \frac{1}{2}, m_s \right\rangle = \frac{3}{2} \hbar^2 \left| \frac{1}{2}, m_s \right\rangle. \quad (2.51)$$

The two eigenstates of \hat{S}_z , we label

$$\left| \frac{1}{2}, \frac{1}{2} \right\rangle \equiv |+\rangle, \quad (2.52)$$

$$\left| \frac{1}{2}, -\frac{1}{2} \right\rangle \equiv |-\rangle. \quad (2.53)$$

We will refer to these states as *spin up* ($|+\rangle$), and *spin down* ($|-\rangle$). The measurable values are

$$\hat{S}_z |+\rangle = \frac{\hbar}{2} |+\rangle \quad (2.54)$$

$$\hat{S}_z |-\rangle = -\frac{\hbar}{2} |-\rangle. \quad (2.55)$$

So the Hilbert space of the spin for $s = \frac{1}{2}$ is two-dimensional. We use the eigenstates of \hat{S}_z as basis vectors. The general spin-state can be expressed as a two-element column vector (or spinor)

$$\chi = \begin{pmatrix} a \\ b \end{pmatrix} = a\chi_+ + b\chi_-, \quad (2.56)$$

where we have defined

$$\chi_+ = |+\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad (2.57)$$

and

$$\chi_- = |-\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (2.58)$$

The matrix representation of \hat{S}^2 , \hat{S}_x , \hat{S}_y and \hat{S}_z can be determined by considering the eigenvalue equations (for \hat{S}_z and \hat{S}^2), and algebraic relations (for \hat{S}_x and \hat{S}_y). We obtain (ref. [1])

$$\hat{S}_x = \frac{\hbar}{2} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \hat{S}_y = \frac{\hbar}{2} \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \hat{S}_z = \frac{\hbar}{2} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (2.59)$$

and

$$\hat{S}^2 = \frac{3}{2}\hbar^2 \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \quad (2.60)$$

In the literature, one often uses the famous *Pauli spin matrices* to express these operators. These matrices are defined as (ref. [1])

$$\hat{\sigma}_x \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \hat{\sigma}_y \equiv \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \hat{\sigma}_z \equiv \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (2.61)$$

yielding

$$\hat{S}_x = \frac{\hbar}{2}\hat{\sigma}_x, \quad \hat{S}_y = \frac{\hbar}{2}\hat{\sigma}_y, \quad \hat{S}_z = \frac{\hbar}{2}\hat{\sigma}_z. \quad (2.62)$$

2.5.2 Total single-particle wave function

We have seen that the wave function has several degrees of freedom. The solution to the time-independent Schrödinger equation (eq. (2.37)) does not, however, include the spin degrees of freedom, but the total wave function of the system must also include these degrees of freedom.

We first note that the Hilbert space of spin, and the Hilbert space spanned by the energy-eigenvectors (the solutions of eq. (2.37)) are two distinct spaces. We label the Hilbert space of the energy-eigenvectors \mathcal{H}_e , and the Hilbert space of the spin \mathcal{H}_s . The total Hilbert space of the system is then

$$\mathcal{H} = \mathcal{H}_e \otimes \mathcal{H}_s. \quad (2.63)$$

The \otimes symbol represents a so called tensor product (see ref. [5]). The total wave function becomes (in three dimension, omitting time)

$$|\Psi(\mathbf{x})\rangle = |\Psi(x, y, z)\rangle \otimes |\chi\rangle, \quad (2.64)$$

where the vector $\mathbf{x} = (\mathbf{r}, \mathbf{s})$ now contains both the spatial coordinates, and the spin degrees of freedom. $|\Psi(x, y, z)\rangle$ is the spatial part of the wave-function, and $|\chi\rangle$ is the spin part.

Operators must also be modified. An operator \hat{A} acting on the spatial part of the wave function becomes

$$\hat{A} \otimes \hat{I}, \quad (2.65)$$

where \hat{I} is the identity operator. An operator \hat{B} acting on the spin becomes

$$\hat{I} \otimes \hat{B}. \quad (2.66)$$

A general operator working on the total wave function is then

$$(\hat{A} \otimes \hat{B}) |\Psi(\mathbf{x})\rangle = (\hat{A} \otimes \hat{B}) (|\Psi(x, y, z)\rangle \otimes |\chi\rangle) \quad (2.67)$$

$$= \hat{A} |\Psi(x, y, z)\rangle \otimes \hat{B} |\chi\rangle. \quad (2.68)$$

2.6 The particle in a box

We now go through the famous *particle in a box* problem. This problem is quite easy to solve, but gives valuable insight into the nature of quantum mechanics, and introduces us to some of the important new ideas that are fundamental in the theory. We will show how *size quantization* arises naturally as a consequence of the confinement of a particle. In the following calculations and discussions we do not consider the spin of the particle, and we use the coordinate representation of quantum mechanics. Furthermore, we consider just one dimension.

Suppose we have a potential

$$V(x) = \begin{cases} 0, & \text{if } 0 \leq x \leq L \\ \infty, & \text{otherwise} \end{cases} \quad (2.69)$$

This is called *the infinite square well* potential (see Figure 2.2). The particle is completely

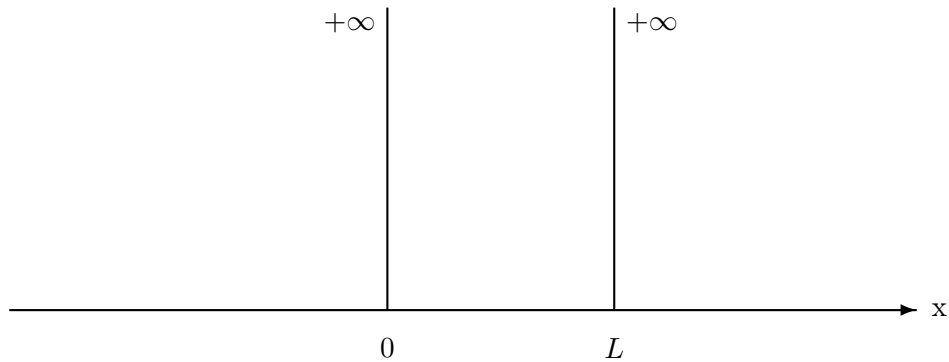


Figure 2.2: The infinite square well potential. The potential is: $V(x) = \infty$ when $x \leq 0$ and $x \geq L$, and $V(x) = 0$ elsewhere.

free, except at the two ends $x = 0$ and $x = L$, where an infinite force prevents it from escaping. A classical analogue would be a cart on a frictionless horizontal air track, with perfectly elastic bumpers, bumping back and forth forever between two walls. We write the time-independent Schrödinger equation for the system

$$\frac{d^2\psi(x)}{dx^2} = \frac{2m}{\hbar^2} [V(x) - E] \psi(x). \quad (2.70)$$

Outside the well, the probability of finding the particle is zero, because of the infinite potential barrier. This means that $\psi(x) = 0$, outside the well. Inside the well we have $V = 0$, and eq. (2.70) reads

$$-\frac{2m}{\hbar^2} \frac{d^2\psi(x)}{dx^2} = E\psi(x), \quad (2.71)$$

or

$$\frac{d^2\psi(x)}{dx^2} = -k^2\psi(x), \quad k \equiv \frac{\sqrt{2mE}}{\hbar}. \quad (2.72)$$

This is the same as the classical simple harmonic oscillator equation. The general solution is

$$\psi(x) = A \sin kx + B \cos kx, \quad (2.73)$$

where A and B are arbitrary constants. These constants are fixed by the boundary conditions of the problem. We require that $\psi(0) = \psi(L) = 0$. Together with eq. (2.73) this gives that

$$\psi(0) = A \sin 0 + B \cos 0 = B. \quad (2.74)$$

So $B = 0$, hence

$$\psi(L) = A \sin kL. \quad (2.75)$$

We must have either $A = 0$ which gives us the trivial and non-normalizable solution $\psi(x) = 0$. Otherwise $\sin kL = 0$, which yields

$$kL = 0, \pm\pi, \pm2\pi, \dots \quad (2.76)$$

But $kL = 0$ again implies that $\psi(x) = 0$, and the negative solutions give nothing new, since $\sin -\theta = -\sin \theta$, and the negative sign can be absorbed into A . This means the distinct solutions are

$$k_n = \frac{n\pi}{L}, \quad n = 1, 2, 3, \dots \quad (2.77)$$

We conclude that the possible wave functions are

$$\psi_n(x) = A \sin \frac{n\pi x}{L}. \quad (2.78)$$

We find A by the normalization requirement

$$\int_{-\infty}^{+\infty} |\psi(x)|^2 dx = 1, \quad (2.79)$$

which yields

$$A = \sqrt{\frac{2}{L}}. \quad (2.80)$$

So the energy eigenfunctions of the time independent Schrödinger equation is

$$\psi_n(x) = \sqrt{\frac{2}{L}} \sin \frac{n\pi x}{L}. \quad (2.81)$$

The energy eigenvalues are found by combining eq. (2.72) and eq. (2.77)

$$E_n = \frac{n^2\pi^2\hbar^2}{2mL^2}, \quad n = 1, 2, 3, \dots \quad (2.82)$$

In Figure 2.3 we have plotted the first few eigenfunctions of the particle in a box. The integer n is what we call the energy quantum number of the system.

The conclusion is that the energy of a particle in a box can only take discrete values. This is what we call quantization, and it is a direct result of the confinement of the particle. We could remove the confinement condition by making the box size L tend to infinity. The discrete energy spectrum then becomes continuous in the limit. This means that at some point, when the box size is large enough, we are not able to discern quantization effects (although they are still there, just smaller than we can observe). Another important observation is that a particle in a box has a non zero minimum energy, known as a zero point energy. Both these properties (quantization and zero point energy) are entirely new in quantum mechanics compared to classical mechanics.

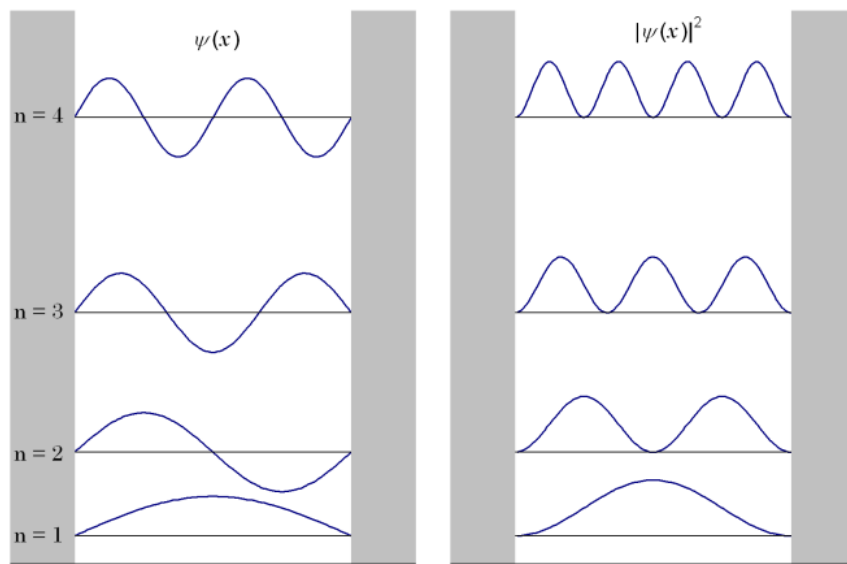


Figure 2.3: The particle in a box wave functions is shown to the left for the quantum numbers $n = 1, 2, 3, 4$. To the right is shown the corresponding probability densities. Image courtesy of Christian Hill.

Chapter 3

Many-Body Theory

In Chapter 2 we discussed basic quantum mechanics, and how to solve problems in quantum mechanics involving just one particle. This is, of course, the natural starting point. Single-particle problems are instructive and shows us many of the important features of quantum mechanics. However, one-particle problems are not very realistic. Almost every realistic physical system will consist of several particles which are interacting with each other. We therefore need ways to solve the many-body problem, i. e. solving problems involving two or more interacting particles. It turns out that such problems are impossible to solve analytically. We must use our knowledge and understanding of quantum mechanics in order to make good approximations, and then use numerical methods to solve the problems.

In this chapter we will discuss the basic quantum mechanics of many-body systems. We start out with presenting the many-body problem. We then consider the wave function for an electronic system, and discuss some of the features this wave function must have. Then we will present the non-interacting system and construct wave function solutions to this problem. We conclude with seeing how one can use the solutions of the non-interacting system to make a general solution to the interacting system.

3.1 The Many-Body Problem

We will consider an isolated system consisting of N particles. In this thesis we exclusively use the non-relativistic description of quantum mechanics, and we do not allow the number of electrons N of the system to vary. To go beyond either of these assumptions would require significant and non-trivial modifications of the mathematical formalism. It has generally been shown that in the study of ground states or low-energy excited states of electronic systems, both of these approximations are quite safe, and the resulting simplification of the model is highly desirable ref. [6].

As discussed in Chapter 2, every physical system is represented by a wave function. When the physical system consists of N particles, we can write the wave function as $\Psi(\mathbf{x}_1, \dots, \mathbf{x}_N, t)$. The coordinates $\mathbf{x} \equiv (\mathbf{r}, s)$ contains both the space and spin degrees of freedom of the particles. The wave function exists in a potential energy landscape described by the function $\hat{V}(\{\mathbf{x}\}, t)$. There is also a set of observables for the system that are represented by operators $\{\hat{O}(\{\mathbf{x}\}, t)\}$. From these observables all the possible

information about the system can be obtained at any point in time, so long as the wave function and observables are completely known. The potential \hat{V} and operators \hat{O} can in addition be dependent on the wave function, Ψ , and on derivatives of Ψ , but we will not consider such potentials and observables in this thesis. In addition, we assume that \hat{V} and \hat{O} are independent of spin. The wave function Ψ is sometimes represented in the bra-ket notation as $|\Psi\rangle$. We use this notation where it is convenient.

Given a system, the first thing we need to do is find the Hamiltonian of the system. We define the Hamiltonian as

$$\hat{H} = \hat{T} + \hat{V}, \quad (3.1)$$

where \hat{T} is the total kinetic energy operator and \hat{V} is the total potential energy operator. We define the kinetic energy operator as

$$\hat{T} = \sum_{i=1}^N \hat{t}_i, \quad (3.2)$$

where \hat{t}_i is the kinetic energy of particle i . The potential energy operator is a bit messier:

$$\hat{V} = \sum_{i=1}^N \hat{v}_i^{(1)} + \frac{1}{2!} \sum_{ij}^N \hat{v}_{ij}^{(2)} + \dots + \frac{1}{N!} \sum_{ij..q}^N \hat{v}_{ij..q}^{(N)}, \quad (3.3)$$

where $\hat{v}_i^{(1)}$ is a one-body potential operator, $\hat{v}_{ij}^{(2)}$ is a two-body operator, and $\hat{v}_{ij..q}^{(N)}$ is an N -body potential operator. In electronic systems, like atoms and quantum dots, the potential energy operator only contains two-body interactions. This would not be the case in nuclear physics, where the fundamental strong interaction seems to exhibit three-body behaviour.

Luckily, in our thesis, we consider quantum dots, which are electronic systems. This means we only need to include two-body interactions, and our Hamiltonian is a two-body operator. Our Hamiltonian reads

$$\hat{H} = \sum_{i=1}^N \hat{t}_i + \sum_{i=1}^N \hat{v}_i^{(1)} + \frac{1}{2!} \sum_{ij}^N \hat{v}_{ij}^{(2)}. \quad (3.4)$$

The time independent Schrödinger equation (2.37) reads

$$\hat{H} |\Psi\rangle = E |\Psi\rangle, \quad (3.5)$$

where E is the energy eigenvalue, and $|\Psi\rangle$ is the eigenfunction. This equation is what is usually referred to as the quantum mechanical many-body problem. It is highly non-trivial due to the interaction between the particles, and can generally not be solved exactly. Even in our case, when the Hamiltonian is a two-body operator, we cannot solve it exactly. The topic of this thesis is to solve this problem in a numerical fashion.

In the following we will see how one goes about constructing the wave function for a system consisting of many particles. The first important observation is that an electronic system such as ours consist of *identical* particles.

3.2 Identical particles

Identical particles are particles which have all physical properties in common. In classical mechanics, identical particles are distinguishable; in the physics of microscopic particles where we must use quantum mechanics, the situation is quite different. We can distinguish two identical particles which are separated by a large distance. For example, one electron at the moon can be distinguished from one on the earth. However, the case is different when two identical particles interact with each other. Heisenberg's uncertainty principle kicks in and as a consequence, microscopic particles which interact with each other are completely indistinguishable.

As discussed in Chapter 2, the measurable quantities of a quantum mechanical system is found by taking the expectation values of operators that represent the observables of the system. We denote the wave function of a system consisting of N identical particles

$$\Psi(\mathbf{x}_1, \dots, \mathbf{x}_N). \quad (3.6)$$

Now, if the system consists of identical particles, the expectation values must not change when the coordinates of two particles are interchanged in the wave function. Thus, we require that for any possible state Ψ of the system and for all observables O

$$\begin{aligned} & \int d\tau \Psi^*(\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_j, \dots, \mathbf{x}_N) \hat{O} \Psi(\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_j, \dots, \mathbf{x}_N) \\ &= \int d\tau \Psi^*(\mathbf{x}_1, \dots, \mathbf{x}_j, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N) \hat{O} \Psi(\mathbf{x}_1, \dots, \mathbf{x}_j, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N), \end{aligned} \quad (3.7)$$

with $d\tau = d\mathbf{x}_1 \dots d\mathbf{x}_N$. This must hold for all pairs (i, j) . We use the notation

$$\int d\mathbf{x} = \sum_s \int d\mathbf{r}. \quad (3.8)$$

We define the permutation operator

$$\hat{P}_{ij} \Psi(\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_j, \dots, \mathbf{x}_N) \equiv \Psi(\mathbf{x}_1, \dots, \mathbf{x}_j, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N), \quad (3.9)$$

which swaps the coordinates of particle i and particle j . Applying this operator twice restores the original wave function. Hence:

$$\hat{P}_{ij}^2 = \mathbb{1} \quad \Rightarrow \quad \hat{P}_{ij}^{-1} = \hat{P}_{ij}. \quad (3.10)$$

We can now rewrite equation (3.7):

$$\langle \Psi | \hat{O} | \Psi \rangle = \langle \hat{P}_{ij} \Psi | \hat{O} | \hat{P}_{ij} \Psi \rangle = \langle \Psi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Psi \rangle \quad \forall (i, j). \quad (3.11)$$

This must hold for all wave functions in the Hilbert space under consideration. It follows that

$$\begin{aligned} \langle \Phi | \hat{O} | \Psi \rangle &= \frac{1}{4} \left(\langle \Phi + \Psi | \hat{O} | \Phi + \Psi \rangle - \langle \Phi - \Psi | \hat{O} | \Phi - \Psi \rangle \right. \\ &\quad \left. - i \langle \Phi + i\Psi | \hat{O} | \Phi + i\Psi \rangle + i \langle \Phi - i\Psi | \hat{O} | \Phi - i\Psi \rangle \right). \end{aligned}$$

Using equation (3.11) on the right hand side, this yields

$$\begin{aligned} \langle \Phi | \hat{O} | \Psi \rangle &= \frac{1}{4} \left(\langle \Phi + \Psi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Phi + \Psi \rangle - \langle \Phi - \Psi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Phi - \Psi \rangle \right. \\ &\quad \left. - i \langle \Phi + i\Psi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Phi + i\Psi \rangle + i \langle \Phi - i\Psi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Phi - i\Psi \rangle \right) \\ &= \langle \Phi | \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij} | \Psi \rangle \quad \forall (i, j). \end{aligned} \quad (3.12)$$

This must hold for arbitrary wave functions Φ and Ψ in the Hilbert space. Equation (3.12) implies the operator identity

$$\hat{O} = \hat{P}_{ij}^\dagger \hat{O} \hat{P}_{ij}. \quad (3.13)$$

In particular, if we take \hat{O} to be the identity operator, it follows that

$$\hat{P}_{ij} \hat{P}_{ij}^\dagger = \mathbb{1} \quad \Rightarrow \quad \hat{P}_{ij}^\dagger = \hat{P}_{ij}. \quad (3.14)$$

Thus, the permutation operators are self-adjoint and unitary (when they operate on the space of state functions of identical particles). If we multiply equation (3.13) from the left with \hat{P}_{ij} , we obtain

$$\hat{P}_{ij} \hat{O} = \hat{O} \hat{P}_{ij}, \quad (3.15)$$

which yields

$$[\hat{O}, \hat{P}_{ij}] = 0 \quad \forall (i, j). \quad (3.16)$$

Let Ψ be an eigenfunction of \hat{P}_{ij} . We know that this function must simultaneously be an eigenfunction of the Hamiltonian \hat{H} of the system under consideration, because equation (3.16) must hold for \hat{H} . We can find the eigenvalues a_{ij} of the permutation operator through:

$$\hat{P}_{ij} \Psi = a_{ij} \Psi. \quad (3.17)$$

It follows that

$$\Psi = \hat{P}_{ij}^2 \Psi = a_{ij}^2 \Psi \quad \Rightarrow \quad a_{ij}^2 = 1. \quad (3.18)$$

Since the operators \hat{P}_{ij} are self-adjoint, their eigenvalues are real. Thus

$$a_{ij} = \pm 1. \quad (3.19)$$

It can be shown (ref. [7]) that if a function Ψ is an eigenfunction of all \hat{P}_{ij} , then the eigenvalues of all \hat{P}_{ij} must be identical. We make the following definitions:

- If $\hat{P}_{ij} \Psi = +\Psi \quad \forall (i, j)$, Ψ is *symmetric*.
- If $\hat{P}_{ij} \Psi = -\Psi \quad \forall (i, j)$, Ψ is *antisymmetric*.

The state function of a system of identical particles must be either symmetric or antisymmetric. In this thesis we only work with antisymmetric state functions.

We now define the permutation \hat{P} as a product of permutation operators \hat{P}_{ij}

$$\hat{P} = \prod \hat{P}_{ij}. \quad (3.20)$$

For an antisymmetric wave function Ψ_A , we then have

$$\hat{P} \Psi_A = \text{sgn}(P) \cdot \Psi_A, \quad (3.21)$$

where $\text{sgn}(P) = +1$ if \hat{P} contains an even number of transpositions, and $\text{sgn}(P) = -1$ if \hat{P} contains an odd number of transpositions.

We define an anti-symmetrization operator \hat{A} by

$$\hat{A} = \frac{1}{\sqrt{N!}} \sum_P \text{sgn}(P) \hat{P}, \quad (3.22)$$

where N is the number of particles in the system that is begin permuted. The factor $\frac{1}{\sqrt{N!}}$ is present because of normalization issues (when we later use the anti-symmetrization operator on a wave function). Each sum runs over all possible permutations. If $f(\mathbf{x}_1, \dots, \mathbf{x}_N)$ is an arbitrary function of N variables, we can use \hat{A} to make an antisymmetric function

$$\Psi_A(\mathbf{x}_1, \dots, \mathbf{x}_N) \equiv \hat{A}f(\mathbf{x}_1, \dots, \mathbf{x}_N). \quad (3.23)$$

There is a principle stating (ref. [7])

The Hilbert space of state functions of a system of identical particles contains either *only* symmetric or *only* antisymmetric functions.

In the first case, the particles are called *bosons*, and in the second case *fermions*. This principle is generally known as the *symmetry postulate*, and we will not prove it here. A proof can be found in ref. [7]. In this thesis we will work with only fermions.

The Schrödinger equation in the coordinate representation reads

$$\hat{H}\Psi_\lambda(\mathbf{x}_1, \dots, \mathbf{x}_N) = E_\lambda\Psi_\lambda(\mathbf{x}_1, \dots, \mathbf{x}_N), \quad (3.24)$$

with E_λ as the energy eigenvalue. We can write the state function as in eq. (2.70)

$$\Psi_\lambda(\mathbf{x}_1, \dots, \mathbf{x}_N) = \Psi_\alpha(\mathbf{r}_1, \dots, \mathbf{r}_N) \otimes |\chi_\sigma\rangle, \quad (3.25)$$

where $\Psi_\alpha(\mathbf{r}_1, \dots, \mathbf{r}_N)$ is the spatial part of the state function, $|\chi_\sigma\rangle$ is the spin part and λ denotes the set of quantum numbers (α, σ) . The state function is antisymmetric with respect to the interchange of two particles. Since the state function consists of two parts, existing in two distinct Hilbert spaces, we have two possibilities. Either

$$\Psi_\lambda^{(AS)}(\mathbf{x}_1, \dots, \mathbf{x}_N) = \Psi_\alpha^{(AS)}(\mathbf{r}_1, \dots, \mathbf{r}_N) \otimes |\chi_\sigma\rangle^{(S)}, \quad (3.26)$$

or

$$\Psi_\lambda^{(AS)}(\mathbf{x}_1, \dots, \mathbf{x}_N) = \Psi_\alpha^{(S)}(\mathbf{r}_1, \dots, \mathbf{r}_N) \otimes |\chi_\sigma\rangle^{(AS)}. \quad (3.27)$$

Here (AS) denotes antisymmetric, and (S) denotes symmetric.

In the following section we consider the non-interacting many-body system. While this system is not physically realistic, it none the less serves as a starting point for most many-body methods.

3.3 The Non-Interacting System

Firstly, we will consider a system of non-interacting identical particles, which are acted upon by an external potential. The Hamiltonian for this system is.

$$\hat{H}_0 = \sum_{i=0}^N \hat{h}_i, \quad (3.28)$$

with N being the number of particles in the system. We define

$$\hat{h}_i = \hat{t}_i + \hat{u}_i, \quad (3.29)$$

with \hat{t}_i as the kinetic energy operator

$$\hat{t}_i = -\frac{\hbar^2}{2m} \nabla_i^2, \quad (3.30)$$

and \hat{u}_i as the external potential

$$\hat{u}_i = u(x_i). \quad (3.31)$$

The time-independent Schrödinger equation for the problem reads

$$\hat{H}_0 |\Phi_i\rangle = E_i |\Phi_i\rangle, \quad (3.32)$$

where $|\Phi_i\rangle$ is the energy eigenfunctions of the non-interacting Hamiltonian. We assume that the associated one-particle problem

$$\hat{h} |\phi_\nu\rangle = \epsilon_\nu |\phi_\nu\rangle, \quad (3.33)$$

has been solved. Here $|\phi_\nu\rangle$ is the state function, or *single-particle orbital*, of the one-particle problem, with ν denoting its quantum state. ϵ_ν is the energy of the orbital.

If electrons had been distinguishable, the energy eigenfunctions of eq. (3.32) would have been given as

$$|\Phi_i\rangle = |\phi_\alpha\rangle \otimes |\phi_\beta\rangle \otimes \cdots \otimes |\phi_\delta\rangle, \quad (3.34)$$

where the i subscript denotes the set of quantum numbers $(\alpha, \beta, \dots, \delta)$. However, as we have discussed, electrons are fundamentally indistinguishable particles, and the simple product form of eq. (3.34) assumes that we can tell the particles apart. These functions can therefore not represent a system of fermions.

There is another way to construct functions from the single-particle orbitals. We apply the anti-symmetrization operator (eq. (3.22)) to a product of single-particle states:

$$\begin{aligned} \Phi^{(A)} &= \hat{A} (\phi_{\nu_1}(x_1) \phi_{\nu_2}(x_2) \dots \phi_{\nu_N}(x_N)) \\ &= \frac{1}{\sqrt{N!}} \sum_P \text{sgn}(P) \hat{P} (\phi_{\nu_1}(x_1) \phi_{\nu_2}(x_2) \dots \phi_{\nu_N}(x_N)). \end{aligned}$$

This function can also be written as a determinant

$$\Phi^{(A)} = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{\nu_1}(x_1) & \cdots & \phi_{\nu_1}(x_N) \\ \vdots & & \vdots \\ \phi_{\nu_N}(x_1) & \cdots & \phi_{\nu_N}(x_N) \end{vmatrix}. \quad (3.35)$$

Such functions are called *Slater determinants*. They are eigenfunctions of

$$\hat{H}_0\Phi^{(A)} = E\Phi^{(A)} \quad (3.36)$$

with eigenvalues

$$E = \sum_{i=1}^N \epsilon_{\nu_i} \quad (3.37)$$

We can, by inspecting the wave functions, deduce the following from the Slater determinants:

1. If two particles are in the same state, $\nu_i = \nu_j$ for some $i \neq j$, we have $\Phi^{(A)} = 0$.
2. If two particles have the same position, so that $x_i = x_j$ for some $i \neq j$, we have $\Phi^{(A)} = 0$.

The wave function vanishes in both cases, and as a result, the probability of finding such a state vanishes as well. This has two consequences

1. It is impossible to have two fermions in the same state. i. e. one state cannot be occupied by more than one particle.
2. It is impossible to bring two fermions with the same spin projection to the same point.

These two statements comprise the *Pauli principle* which is a fundamental postulate in quantum mechanics.

It is important to note that the antisymmetric state $\Phi^{(A)}$ is only determined to within a sign by the single-particle states that it contains. For example, we can define two particle states from ϕ_{ν_1} and ϕ_{ν_2}

$$\Phi_{(\nu_1, \nu_2)}^{(A)} = \frac{1}{\sqrt{2}} \begin{vmatrix} \phi_{\nu_1}(x_1) & \phi_{\nu_1}(x_2) \\ \phi_{\nu_2}(x_1) & \phi_{\nu_2}(x_2) \end{vmatrix}, \quad (3.38)$$

or by

$$\Phi_{(\nu_1, \nu_2)}^{(A)} = \frac{1}{\sqrt{2}} \begin{vmatrix} \phi_{\nu_2}(x_1) & \phi_{\nu_2}(x_2) \\ \phi_{\nu_1}(x_1) & \phi_{\nu_1}(x_2) \end{vmatrix}. \quad (3.39)$$

As long as we remember this and make sure to always use the same order of single-particle states (and particles) in our Slater determinants, this is not a problem.

3.3.1 The ground state of the non-interacting system

The aim of this thesis is to compute the ground-state energy of quantum dots. The ground state of a system is defined as the state with the lowest energy eigenvalue. Assume the system has N particles. By considering eq. (3.37), we see that the ground state can be correctly represented by a Slater determinant built up by the N single-particle orbitals with the lowest energy eigenvalues.

3.4 The Interacting System

Consider a system of N fermions. As we have discussed, the state function of the system must be antisymmetric. We denote the N -particle Hilbert space of the antisymmetric states as \mathcal{H}_N^{AS} .

The Hamiltonian of the system is the same as eq. (3.1). We assume that the associated single-particle problem

$$\hat{h}\phi_\nu(\mathbf{x}) = \epsilon_\nu\phi_\nu(\mathbf{x}), \quad (3.40)$$

has been solved. We know from single-particle quantum mechanics that the solutions to eq. (3.40) constitute a complete and orthonormal set. We then utilize the following *completeness theorem* (ref. [7]):

If the family $\{\phi_\nu(x)\}$ is complete, so too are the families $\{\Phi^{(A)}\}$ and $\{\Phi^{(s)}\}$ of many-particle functions in the corresponding Hilbert spaces of antisymmetric and symmetric many-particle functions, respectively.

A proof of this can be found in ref. [7]. This means that the energy eigenfunctions of the interacting N -fermion system (eq. (3.24)) can be written as a linear combination of the eigenfunctions of the non-interacting system in eq. (3.32). We write the expansion as

$$\Psi_\lambda(\mathbf{x}_1, \dots, \mathbf{x}_N) = \sum_{\alpha\beta..\delta} C_{\alpha\beta..\delta}^\lambda \Phi_{\alpha\beta..\delta}(\mathbf{x}_1, \dots, \mathbf{x}_N). \quad (3.41)$$

It seems as though we may have solved the many-body problem, but this is not the case. There is usually an infinite number of solutions to the single-particle problem, which means that the basis of Slater determinants is also infinite. In addition there is the problem of deciding the correct expansion coefficients.

Chapter 4

Quantum Dots - The Artificial Atoms

Low-dimensional nanometer sized systems have defined a new research in condensed-matter physics during the last 20-25 years (ref. [8]). Modern semiconductor processing techniques allowed the artificial creation of quantum confinement of only a few electrons. The new systems have much in common with atoms, but are man-made structures; designed and fabricated in the laboratory. Because of their similarity to atoms they are often dubbed *designer atoms* or *artificial atoms* in the literature. Usually, though, they are called *quantum dots*; this name also indicates some of the properties of the system. The word “dot” refers to the systems spacial structure, which is much like a small dot, confined in all three spatial dimensions. The word “quantum” indicates that the actual size of the system is very small, and reveals the physics that governs the systems behaviour — quantum mechanics. The term quantum dot is applied both to localized nanoscopic semiconductor systems with an unknown number of electrons (ref. [9]), and to systems with a countable number of electrons. We consider in this thesis only the latter.

Quantum dots are of fundamental interest because, while they are man-made objects, manufactured and designed artificially at the laboratory, they are at the same time small enough for us to observe quantum mechanical behaviour. This makes them an excellent component for studying quantum effects, not the least because we can tune them to our need by changing the dots geometric shape and the number of particles in the dot. Quantum mechanical behaviour such as shell structure (ref. [10]), entanglement (ref. [11]), tunneling (ref. [12]) and magnetization (ref. [13]) have all been observed in quantum dots. In addition, investigating many-electron interaction in an atom is difficult, due to interactions with the nucleus. Quantum dots gives us the opportunity to study “electron clouds” without the presence of the nucleus.

Even though we give quantum dots the name artificial atoms, there are significant differences between dots and atoms. Firstly, as previously stated, quantum dots are fabricated in the laboratory. Secondly, the typical length scale of a dot is about $1 - 1000nm$. Atoms are much smaller, ranging from about $0.05nm$ to $0.4nm$. Thirdly, the confining forces are different. In atoms, the attractive forces is set up by the nucleus. In quantum dots we typically have some external field or potential confining the particles.

In spite of these differences, there are important similarities between dots and atoms.

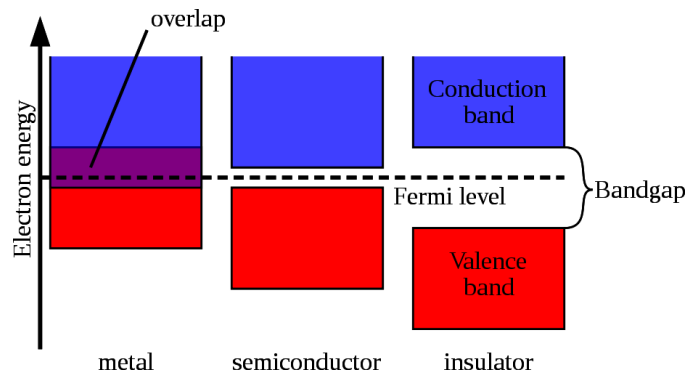


Figure 4.1: A simplified diagram of the electronic band structure of metals, semiconductors, and insulators. Image courtesy of P. Kuiper

The confinement of the particles in the quantum dot lead to size and energy quantization. The only thing in nature that behaves like this is the atom. Quantum dots experience energy bands and excitation in energy just like atoms, and they even exhibit shell structures and magic numbers as seen in atoms, although with different values. We can even control the number of electrons in the dot by controlling the external potential. This means that we in principle can build a periodic table for quantum dots.

If the reader is in need of a more thorough introduction to quantum dots, we refer to [8, 14–16]. In ref. [14] some manufacturing techniques of quantum dots are described.

In this chapter we discuss the basic physics of quantum dots. We start out by discussing semiconductors. Semiconductor materials is important to us because all quantum dots are made out of such materials. We then continue with discussing the properties of a quantum dot, and conclude with presenting some of the main theoretical and practical applications of quantum dots. Much of the discussion in Sections 4.1 to 4.3 follows closely the outline of ref. [17]. This is also where we have borrowed most of the figures from.

4.1 Semiconductors and quantum dots

Semiconductors are generally classified by their electrical resistivity at room temperature, with values in the range of 10^{-2} to $10^9 \Omega cm$ (ref. [18]). This resistivity is strongly dependent on temperature - perfect crystals of most semiconductors are insulators at zero Kelvin.

Devices based on semiconductors include transistors, switches, diodes, photovoltaic cells, detectors and thermistors. Such devices are the basic constituents of modern electronics devices, like radio, computers, telephones etc.

An important concept in all solids is what we call *energy bands*. As discussed in the particle in a box example, confined electrons experience discretized energies. However, compared to the particle in a box, electrons in solids does not show narrow discrete energy levels. Rather, they are arranged in energy bands (see fig. 4.1. We refer to [18] for an explanation as to why energy bands occur.) separated by regions in energy where no

electrons can exist. Such regions are called *bandgaps*. The solid behaves as an insulator if the allowed bands are either full or empty, for then no electrons are free to move in an electric field. The solid behaves as a metal if one or more bands are partly filled. The solid is a semiconductor or a semimetal if one or two bands are slightly filled or slightly empty.

The quantum states in a solid are populated up to a particular band called the *valence band*. The band immediately above the valence band is called the *conduction band*. The ease with which electrons in a semiconductor can be excited from the valence band to the conduction band depends on the bandgap between these bands. The gap size serves as an arbitrary dividing line between semiconductors and insulators.

In an ordinary metallic conductor, the current is carried by the flow of electrons. This is not so in semiconductors; the current can also be carried by the flow of positively charged *holes* in the electron structure of the material. When an electron is excited from a band, it leaves behind a vacant orbital in the same band. This vacant orbital is called a hole. A hole acts in applied electric and magnetic field as if it has a positive charge $+e$. The holes themselves does not exactly move, but a neighbouring electron can move in to fill a hole, it then leaves behind a hole at the place it just came from. This way the holes appears to move, and behaves like positively charged particles. An excited electron paired with the hole it left behind is a bound state called an *exciton*.

It can be shown (ref. [18]) that electrons and holes in a crystal respond to electric and magnetic fields almost as if they were particles with a different mass. In a simplified picture that ignores crystal anisotropies, they behave as free particles in a vacuum, but with a different mass. The result is what we call the *effective masses* of the electrons and holes. We denote the effective mass as m^* . Physically, the effective mass incorporates the complicated periodic potential felt by the charge carriers in the lattice. This approximation allows us to completely ignore the semiconductor atoms in the lattice, and treat the electron and hole as free particles (ref. [19]).

We now return to quantum dots. Quantum dots are semiconductor *nanocrystals* whose excitons are confined in all three spatial direction. Nanocrystals are crystals with at least one dimension less than 100nm. We are going to use the term for crystals where all dimensions are small. More properly one should use the term nanoparticle for these systems, but we are going to use both terms in order to remember that quantum dots behave like particles, but are made out of semiconductor crystals. The main reason quantum dots are so interesting and useful is that they experience size quantization. This enables them to have quantized energy levels, which in turn enables them to absorb and emit light at different frequencies; this ability is again closely related to the optical and electrical qualities of the dots. In Section 2.6, we saw how size quantization arises naturally in small enough systems. A question arises then: What is the limit size of confinement so that quantization effects are still visible at our scale? In general, quantum mechanics is relevant when the de Broglie wavelength of the particle in question is greater than the characteristic size of the system (ref. [1]). The de Broglie wavelength is defined

$$\lambda_B = \frac{h}{p}, \quad (4.1)$$

with λ_B as the de Broglie wavelength, h as Planck's constant, and p as the momentum

of the particle.

The size limit for quantum confinement of charge carriers in solids can be approximated from a modified version of the de Broglie wavelength equation for a free electron (ref. [17])

$$\lambda_B = \frac{h}{p} = \frac{h}{m^*v}, \quad (4.2)$$

where λ_B is the de Broglie wavelength, and p , v and m^* are the momentum, velocity and effective mass of the excited electron, respectively. Modification is required because the semiconductor materials does not contain truly free electrons. With the effective mass approximation, we can treat the charge carriers as free particles. In addition, the effective masses can be quite small for charge carriers in nanocrystals. For instance, in a InSb crystal, an excited electron has an effective mass of $0.015m_e$ (ref. [18]), where m_e is the electron mass. This in turn makes the de Broglie wavelength (eq. (4.2)) large. In the InSb example it is almost 100 times larger than for a free electron. This is actually very lucky for us. It is very difficult to control and create small enough electronic devices so that we can control quantum effects. Since the de Broglie wavelength of charge carriers in quantum dots is much larger than for a free electron, quantum dots can be made relatively large (but still in the nm regime), and still exhibit quantization effect.

Quantum dot semiconductors have properties between those of bulk semiconductors and those of small molecules (fig. 4.2). In bulk semiconductors, the allowed energy levels are organized in bands, as discussed above. When examining a system consisting of only two atoms, the molecular orbitals formed create discrete potential energy states (ref. [17]). Then the electrons will only be excited if the energy absorbed corresponds to specific discrete quantities. In quantum dots, the particle contains less atoms than bulk semiconductors, but more than small molecules. As the number of atoms in the particle is reduced, the energy bands split and shrink, but not to the point of being exactly discrete. This means that electrons in quantum dots can be excited by energies in discrete intervals, rather than a continuum.

We have until now neglected to discuss the Coulombic attraction between the electron and the positively charged hole. By using the *strong confinement approximation* it is possible to show that when the quantum dot is smaller than the size of the bulk exciton, the electron and hole can be treated independently. We will not show this here, but refer to ref. [19] for the interested reader.

For a more theoretical introduction to semiconductors and nanoparticles we refer to [17, 18].

4.2 Electrical properties resulting from quantum confinement

As discussed above, the band structure of quantum dot systems are size dependent. The splitting and shrinking of the bands produces an increase in bandgap with decreasing particle size. The gap energies are therefore size dependent, and it follows that electrical properties that depend on the gap energies also display size dependence. One such property is electron transfer (ref. [20]). As all things in nature, electrons prefer to move

towards states of lower energy. An electron in the conduction band will decay until it reaches the lowest energy state in the band. Since there are no states in the bandgap, it will then return to the valence band through another mechanism (i. e. electron-hole recombination, non-radiative energy loss, etc.). However, if the electron encounters a material with available states that has energy values within the bandgap of the first material, it can transfer its electron to that material, as shown in Figure 4.3. This process is dependent upon the bandgap. If we increase the band gap, excited electrons occupy higher energy levels, and can decay to a greater number of lower state values. Because of the tunability of the bandgaps, we can optimize the electron transfer to many materials.

Another property resulting from small nanocrystal sizes is the presence of large excited state dipole moments. When electrons are excited it is possible for the electron to get trapped at the particle surface (ref. [17]). This results in a fixed charge separation between the electron and the hole, producing a dipole moment that depends on the size of the crystal. The dipole moments are of interest in electronic applications because they can be used to influence ions outside of the particle (ref. [17]).

4.3 Optical properties resulting from quantum confinement

Quantum confinement also affects the optical properties of quantum dots. We denote the bandgap energy ΔE . For a photon to excite an electron, it needs to have at least this much energy. We can then set up the following equation

$$\Delta E = \hbar\omega = \frac{hc}{\lambda}, \quad (4.3)$$

where c is the speed of light, and ω and λ is the frequency and wavelength of the incident light, respectively. Since the energy is inversely proportional to the wavelength of the incident light, it follows that quantum dots will only absorb light of wavelength shorter than that determined by eq. (4.3). As particle size decreases, the bandgap increases, and the absorbance onset shifts to shorter wavelengths. Thus, onset of absorbance is directly related to particle size. In Figure 4.4 this is shown for a CdS quantum dot.

The particle size also influences the particle fluorescence properties. When an electron is excited, it will lose some energy due to atom vibrations (satisfying the second law of thermodynamics). This energy is typically converted to heat energy. When the electron decays back to the valence band, it will emit light (fluoresce) at a longer wavelength due to the energy lost. This is shown in Figure 4.5A. As the bandgap decreases, less energy will dissipate through fluorescent emission when the electron decays. In this way, the wavelength of emitted light will shift to the red as shown in Figure 4.5B. This redshift is dependent on the size of the nanocrystals. In addition, the energy lost to heat decreases in a size-dependent manner.

Nanoparticles can also exhibit a unique type of fluorescent emission resulting from the trapping of an electron at the crystal surface. By introducing a defect into the crystal, it can introduce a potential energy state in the bandgap as shown in Figure 4.6. Electrons that enter in to this state can decay back to the valence band (albeit at a low probability (ref. [17])). Because of this low probability, the life time of an exciton in such a trapped state is significantly longer than that of an ordinary exciton. In addition, the shift between

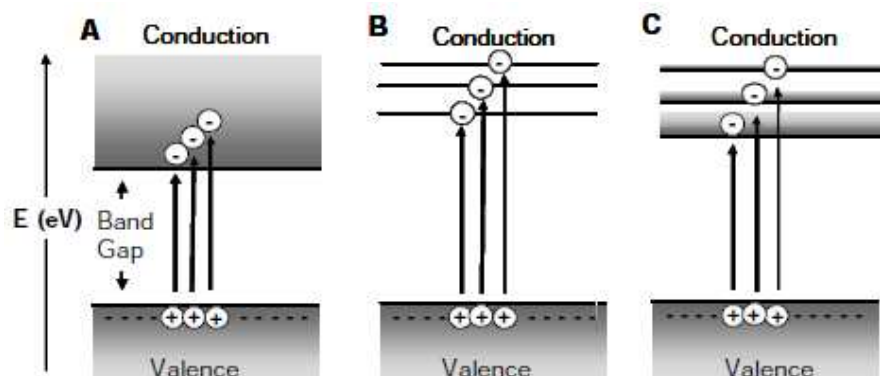


Figure 4.2: (A) Bulk materials have continuous energy bands and absorb energy at a value greater than the band gap. (B) Molecular materials possess discrete energy levels and only absorb energy with certain values. Also, the bandgap is greater than that of a bulk material as a result of shrinking and splitting of the energy bands. (C) Quantum dots lie between the extremes (A, B). They possess discrete energy bands and absorb energy in discrete intervals. The bandgap is greater than that of a bulk material, but less than that of a molecular material. Image courtesy of Jessica O. Winter (ref. [17])

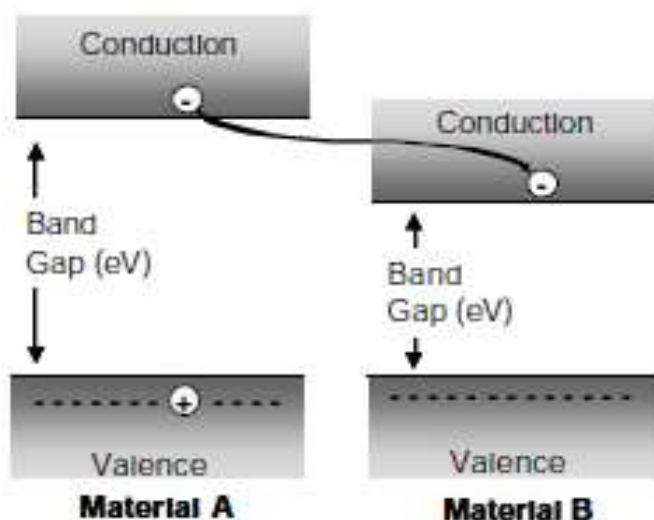


Figure 4.3: The figure shows electron transfer between materials with different bandgap size. An electron in one material (A) can be transferred to another material (B) with a lower bandgap. Image courtesy of Jessica O. Winter (ref. [17]).

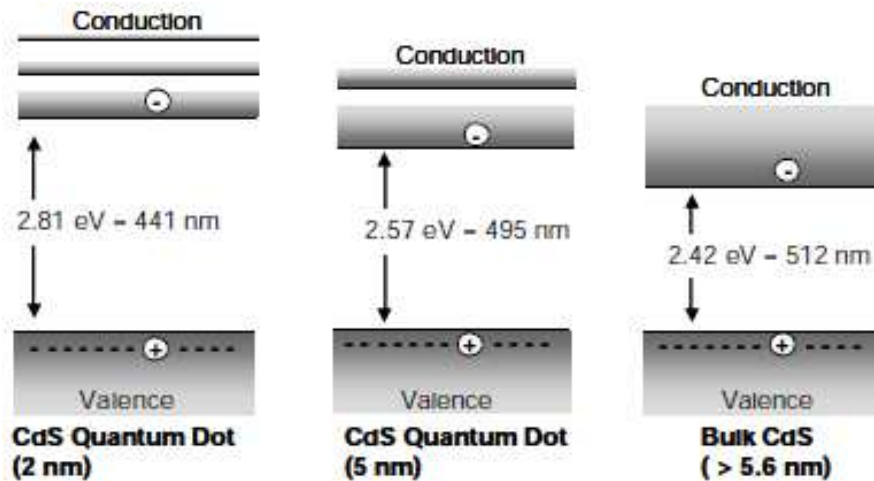


Figure 4.4: In this figure is shown the bandgap energy for different CdS quantum dot sizes as well as Bulk CdS. The bandgap energy is inversely related to the absorbance onset (λ). Smaller particles (the leftmost figure) begin to absorb at shorter wavelengths. Image courtesy of Jessica O. Winter (ref. [17]).

the absorbance wavelength and the emitted wavelength will be larger as a result of the energy lost in decaying to the trapped state.

4.4 Applications of quantum dots

Because of our ability to tune the quantum dots in a precise fashion, they offer a wide variety of usage. As discussed, quantum dots have excellent optical and electrical properties. They are therefore attractive components for integration into electronic devices. One advantage they have over traditional optoelectronic materials is that they exist in the solid state; solids tend to be more compact, easily cooled and allow for direct charge injection. In addition, quantum dots can interconvert light and electricity in a tunable manner dependent on crystal size. This allows for easy wavelength selection. This is a significant improvement over silicon-based materials. These materials require modification of their chemical composition to alter optical properties.

Quantum dots can be used to absorb and emit light efficiently at any wavelength. This property enables them to form new kinds of lighting and improve the current laser technologies. Another problem with conventional lasers is the need for cooling, and that the pulses are relatively slow. These parameters can be improved by the use of quantum dots. Quantum dots can also be used to produce efficient white light (ref. [21]).

The main idea in conventional computer circuits is to create a system that can handle voltage differences, controlled by some external device. During recent years, microchips have been made smaller and smaller. At some point we will no longer be able to make smaller circuits because of quantum effects. Scientists dream of creating quantum computers where the fundamental information unit is the quantum bit (or qbit). Quantum dots have been proposed as the building blocks in such quantum computers, and the idea is to manipulate the electron spin states. A computer built on quantum principles will

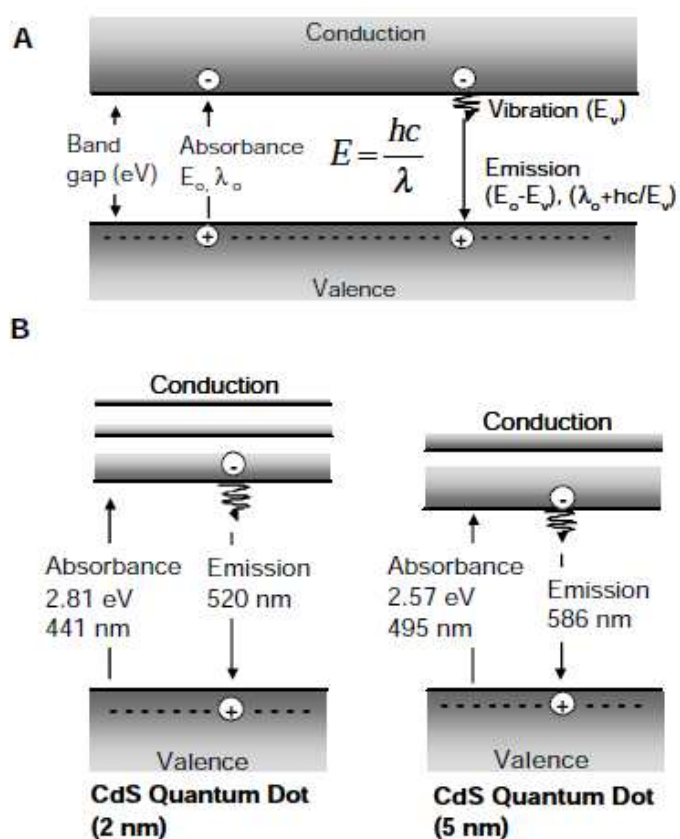


Figure 4.5: The figure shows fluorescent emission and particle bandgap. (A) Photon absorption creates an excited electron. This electron loses some energy to heat, then decays back to the valence band. The emitted photon has a longer wavelength than the absorbed photon because of the energy lost to heat. (B) As the bandgap decreases, the particle will absorb at longer wavelengths. This will produce an associated red-shift in particle fluorescent emission. Image courtesy of Jessica O. Winter (ref. [17]).

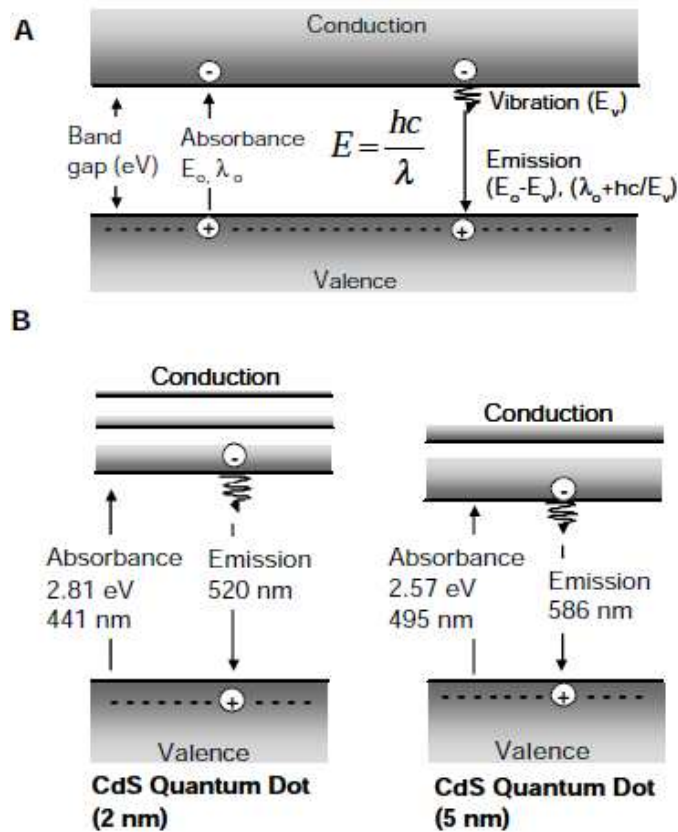


Figure 4.6: The figure shows Trapped State Fluorescent Emission. (A) After absorption of a photon, an excited electron loses some energy to vibration and then decays back to the valence band. (B) In a system with a trap, the electron first decays into the trap, then decays to the valence band producing a photon at a longer wavelength than in (A). Image courtesy of Jessica O. Winter (ref. [17]).

manipulate information in a qualitatively different way than a classical computer in such a way that it will be able to solve certain types of problems much more efficiently. In ref. [4] can be found an introduction to qbits and quantum information theory.

Researchers have also experimented with quantum dots in photovoltaics and new generations of transistors. An exciting possible application lies towards solar cells. It has been shown (ref. [22]) that by creating intermediate bandgaps, the theoretical efficiency of the solar cell is as high as 63.2%. The tunability of quantum dots makes them great candidates for this purpose. Indeed, quantum dot intermediate band solar cell material triads (quantum-dot/barrier/substrate) have been uncovered that yield thermodynamic conversion efficiencies of over 60% at maximum concentration (ref. [23]). This is a great deal higher than conventional solar cells with an efficiency of about 30%. If the reader is interested, ref. [24] goes a bit more into detail on this subject.

In addition, quantum dots can be used for biological applications. Again, it is their unique optical properties that make them appealing. For the interested reader we refer to [25, 26].

Chapter 5

Theoretical approximations to quantum dots in two dimensions

In this chapter we aim at developing a theoretical framework for the two-dimensional quantum dot. Quantum dots are complex devices, and it is not easy to give a simple and complete theoretical model. We need to account for the electron-electron potential, the Pauli exclusion principle, edge effects, and external fields. We should also have an exact analytical expression for the confining potential. The complexity of a model accounting for all this would quickly reach the limit of our computational resources. We therefore need to introduce some approximations.

In this thesis we will completely disregard edge effects. We will model the electron-electron interaction with the Coulomb interaction. Another common approximation that we will use is to reduce the number of spatial dimensions from three to two. The third dimension is usually fixed by manufacturing techniques, which forces the electrons to occupy a planar region. We approximate the confining potential with the harmonic oscillator potential. In two dimensions this potential takes the form

$$V(x, y) = \frac{1}{2}m^*\omega_x^2x^2 + \frac{1}{2}m^*\omega_y^2y^2, \quad (5.1)$$

where m^* is the reduced mass of the electron as we discussed in the previous chapter, x and y are the coordinates of the electron with respect to the point in space where $V(x, y) = 0$, and ω_x and ω_y are the oscillator frequencies in their respective directions. This potential can be tuned by varying the oscillator frequencies. For simplicity we will use a spherical symmetric potential, which means we require $\omega_x = \omega_y \equiv \omega_0$. The potential then takes the form

$$V(x, y) = \frac{1}{2}m^*\omega_0^2(x^2 + y^2). \quad (5.2)$$

Numerical (refs. [27, 28]) and experimental (refs. [29, 30]) results show that this approximation is quite good. Quantum dots with this confining potential are often dubbed parabolic quantum dots, due to the parabolic nature of the potential.

We will in our calculations also include the effect of an electromagnetic field, where the magnetic field is constant and uniform in the z -direction, i. e. the direction perpendicular to the two-dimensional plane.

In this chapter we first consider a quantum dot with only one particle. This system can be analytically solved, and will give us a single-particle orbital basis for later calculations. We move on to construct the N -particle Hamiltonian that is the target of our numerical calculations. We finally bring this Hamiltonian over to a dimensionless form, in order to simplify the numerical calculations.

5.1 One-electron quantum dot

We start out with considering the one-particle problem. The solutions to this problem is important because it gives us the one-particle orbitals that we use to construct Slater determinants in later chapters.

We first need set up the Hamiltonian for the system. We assume that the potential is given as in eq. (5.2). We further assume that the particle is under the influence of a magnetic field. The classical Hamiltonian of a charged electron in an electromagnetic field reads as (ref. [31])

$$H = \frac{1}{2m} (\mathbf{p} - e\mathbf{A})^2 + e\phi, \quad (5.3)$$

with \mathbf{A} and ϕ as the electromagnetic potentials, m as the mass of the electron, and \mathbf{p} as the classical momentum. The electromagnetic potentials are related to the electromagnetic fields through

$$\mathbf{E} = -\frac{1}{c} \frac{\partial \mathbf{A}}{\partial t} - \nabla \phi, \quad (5.4)$$

$$\mathbf{B} = \nabla \times \mathbf{A}, \quad (5.5)$$

where \mathbf{E} is the electric field, and \mathbf{B} is the magnetic field. As previously discussed, electrons carries intrinsic spin. Because of this spin, the electron has a magnetic moment, μ , which can couple with the magnetic field. This leads to an additional energy contribution, $-\boldsymbol{\mu} \cdot \mathbf{B}$. The Hamiltonian of a single-electron parabolic quantum dot subjected to an electromagnetic field then reads

$$\hat{H} = \frac{1}{2m^*} (\hat{\mathbf{p}} - e\mathbf{A})^2 + \frac{1}{2}m^*\omega_0^2(\hat{x}^2 + \hat{y}^2) + e\phi - \hat{\boldsymbol{\mu}} \cdot \mathbf{B}. \quad (5.6)$$

Let us now assume that the magnetic field is constant and uniform along the z -axis, $\mathbf{B} = B_0\mathbf{k}$, where \mathbf{k} is the unit vector in z -direction. We further assume that the electric field is nonexistent. From eqs. (5.4) and (5.5), it follows that \mathbf{A} must be constant in time (if \mathbf{B} is constant, then \mathbf{A} must be as well). We then must have that

$$\nabla \phi = 0, \quad (5.7)$$

in other words, ϕ must be constant. This means that the term $e\phi$ in eq. (5.3) is just a constant addition to the energy.

Let us expand the first term in the Hamiltonian (eq. (5.6))

$$(\hat{\mathbf{p}} - e\mathbf{A})^2 = \hat{\mathbf{p}}^2 - e(\hat{\mathbf{p}} \cdot \mathbf{A} + \mathbf{A} \cdot \hat{\mathbf{p}}) + e^2\mathbf{A}^2. \quad (5.8)$$

In general we have that

$$\mathbf{A}(x, y, z) = A_x(x, y, z)\mathbf{i} + A_y(x, y, z)\mathbf{j} + A_z(x, y, z)\mathbf{k}, \quad (5.9)$$

with \mathbf{i} , \mathbf{j} and \mathbf{k} as unit vectors in respectively x -, y - and z -direction. We can make \mathbf{A} and $\hat{\mathbf{p}}$ commute by demanding that

$$\mathbf{A}(x, y, z) = A_x(y, z)\mathbf{i} + A_y(x, z)\mathbf{j} + A_z(x, y)\mathbf{k}. \quad (5.10)$$

In this case we also have $\nabla \cdot \mathbf{A} = 0$. This is called the Coulomb gauge (ref. [31]). It can be shown (ref. [31]) that the Coulomb gauge only affect the choice of potentials, \mathbf{A} and ϕ , but do not constrain the electromagnetic fields \mathbf{E} and \mathbf{B} . One possible choice for the vector potential is

$$\mathbf{A} = \frac{B_0}{2} (-\hat{y}\mathbf{i} + \hat{x}\mathbf{j}). \quad (5.11)$$

By inserting this into equation (5.8) we get

$$(\hat{\mathbf{p}} - e\mathbf{A})^2 = \hat{\mathbf{p}}^2 - eB_0(\hat{x}\hat{p}_y - \hat{y}\hat{p}_x) + \frac{e^2 B_0^2}{4}(\hat{x}^2 + \hat{y}^2). \quad (5.12)$$

We note that $\hat{x}\hat{p}_y - \hat{y}\hat{p}_x = \hat{L}_z$ (ref. [1]), where \hat{L}_z is the angular momentum operator in the z -direction. We insert this back into the Hamiltonian, resulting in

$$\hat{H} = \frac{1}{2m^*} \left(\hat{\mathbf{p}}^2 - eB_0\hat{L}_z + \frac{e^2 B_0^2}{4}(\hat{x}^2 + \hat{y}^2) \right) + \frac{1}{2}m^*\omega_0^2(\hat{x}^2 + \hat{y}^2) + e\phi - \hat{\boldsymbol{\mu}} \cdot \mathbf{B}. \quad (5.13)$$

We now define

$$\omega_B = \frac{eB_0}{2m^*}, \quad (5.14)$$

$$\omega^2 = \omega_B^2 + \omega_0^2. \quad (5.15)$$

This yields

$$\hat{H} = \frac{1}{2m^*} \left(\hat{\mathbf{p}}^2 - eB_0\hat{L}_z \right) + \frac{1}{2}m^*\omega^2(\hat{x}^2 + \hat{y}^2) + e\phi - \hat{\boldsymbol{\mu}} \cdot \mathbf{B}. \quad (5.16)$$

We split the Hamiltonian up in two:

$$\hat{H} = \hat{H}_\Omega - \hat{H}_s + e\phi, \quad (5.17)$$

where we have defined

$$\hat{H}_\Omega = \frac{1}{2m^*} \left(\hat{\mathbf{p}}^2 - eB_0\hat{L}_z \right) + \frac{1}{2}m^*\omega^2(\hat{x}^2 + \hat{y}^2) \quad (5.18)$$

$$\hat{H}_s = \hat{\boldsymbol{\mu}} \cdot \mathbf{B}. \quad (5.19)$$

We then have that \hat{H}_Ω is dependent on spatial coordinates, and \hat{H}_s is a spin dependent part. We then have

$$H\psi(\mathbf{x}) = H_\Omega\psi(\mathbf{x}) - H_s\psi(\mathbf{x}) + e\phi\psi(\mathbf{x}) \quad (5.20)$$

$$= E_\Omega\psi(\mathbf{x}) - E_s\psi(\mathbf{x}) + e\phi\psi(\mathbf{x}), \quad (5.21)$$

where $\psi(\mathbf{x})$ is the energy eigenfunction, \mathbf{x} contains both the spatial coordinates and the spin degrees of freedom. The total energy of the system can then be written

$$E = E_\Omega - E_s + e\phi. \quad (5.22)$$

From Chapter 2 we know that we can write the total wave function in two dimensions as

$$\psi(\mathbf{x}) = \psi(x, y) \otimes \left| \frac{1}{2}, m_s \right\rangle, \quad (5.23)$$

where \mathbf{x} contains both the spatial coordinates and the spin degrees of freedom. In this representation, $\psi(x, y)$ is the eigenfunction of the spatial part of the Hamiltonian, and $\left| \frac{1}{2}, m_s \right\rangle$ is the eigenfunction of the spin part. We know the form of the spin part since we are working with fermions.

We first find the energy of the spin part. The magnetic moment is given as

$$\hat{\boldsymbol{\mu}} = \frac{eg}{2m^*} \hat{\mathbf{S}}, \quad (5.24)$$

where g is the Landé g-factor, which is $g \approx 2$, e is the electron charge, and \hat{S} is the spin operator as discussed in Chapter 2. We now rewrite the spin Hamiltonian as

$$\begin{aligned} \hat{H}_s &= \hat{\boldsymbol{\mu}} \cdot \mathbf{B} \\ &= \frac{eg}{2m^*} \hat{\mathbf{S}} \cdot B_0 \mathbf{k} \\ &= \frac{egB_0}{2m^*} \hat{S}_z \\ &= g\omega_B \hat{S}_z. \end{aligned} \quad (5.25)$$

This spin Hamiltonian will act on the spin part of the wave function

$$\begin{aligned} \hat{H}_s \left| \frac{1}{2}, m_s \right\rangle &= g\omega_B \hat{S}_z \left| \frac{1}{2}, m_s \right\rangle \\ &= m_s \hbar g \omega_B \left| \frac{1}{2}, m_s \right\rangle, \end{aligned} \quad (5.26)$$

where m_s is the spin projection on the z -axis. For electrons, we have that $m_s = \pm \frac{1}{2}$. Do *not* mistake m_s for m^* , which is the effective mass of the electron. In the calculation above we have used what we learned about spin in Chapter 2. Equation (5.25) gives us the following energy eigenvalues for the spin part

$$E_s = m_s \hbar g \omega_B. \quad (5.27)$$

We now look at the spatial part of the total Hamiltonian. The time-independent Schrödinger equation reads

$$\hat{H}_\Omega \psi(x, y) = E_\Omega \psi(x, y). \quad (5.28)$$

We will solve this equation in polar coordinates, as it is much easier than using Cartesian coordinates. In ref. [32] can be found how the transformations are done, we will not show how to do them in this thesis, as it is quite tedious and not really relevant; we just need the results. In polar coordinates the z -projection of the angular momentum reads

$$\hat{L}_z = -i\hbar \frac{\partial}{\partial \theta}. \quad (5.29)$$

We do the substitution $\hat{p} \rightarrow -i\hbar \nabla$, which yields

$$\hat{p}^2 = \hbar^2 \nabla^2. \quad (5.30)$$

The two-dimensional Laplacian in polar coordinates reads as

$$\nabla^2 = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2}, \quad (5.31)$$

in polar coordinates. The spatial wave function is rewritten as

$$\psi(x, y) = \psi(r, \theta), \quad (5.32)$$

and the Hamiltonian is rewritten as

$$\hat{H}_\Omega = -\frac{\hbar^2}{2m^*} \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2} - \frac{eiB_0}{\hbar} \frac{\partial}{\partial \theta} \right) + \frac{1}{2} m^* \omega r^2. \quad (5.33)$$

We want to solve the following eigenvalue equation

$$-\frac{\hbar^2}{2m^*} \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2} - \frac{eiB_0}{\hbar} \frac{\partial}{\partial \theta} \right) \psi(r, \theta) + \frac{1}{2} m^* \omega r^2 \psi(r, \theta) = E \psi(r, \theta). \quad (5.34)$$

We try with separable solutions on the form

$$\psi(r, \theta) = R(r)Y(\theta). \quad (5.35)$$

What form should the angular part have? We try with an *ansatz*

$$Y(\theta) = e^{im\theta}. \quad (5.36)$$

We then have

$$\psi(r, \theta) = R(r)e^{im\theta}. \quad (5.37)$$

By substituting this into eq. (5.34) we get

$$-\frac{\hbar^2}{2m^*} \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right) - \frac{m^2}{r^2} + \frac{emB_0}{\hbar} \right) R(r) + \frac{1}{2} m^* \omega r^2 R(r) = ER(r). \quad (5.38)$$

The solution to this equation is derived in ref. [33]. The eigenfunctions are

$$R_{nm}(r) = \sqrt{\frac{2n!}{(n+|m|)!}} \beta^{\frac{1}{2}(|m|+1)} r^{|m|} e^{\frac{1}{2}\beta r^2} L_n^{|m|}(\beta r^2), \quad (5.39)$$

where $L_n^{|m|}(\beta r^2)$ is the associated Laguerre polynomials, and β is defined as

$$\beta = \frac{m^* \omega}{\hbar}. \quad (5.40)$$

The eigenfunctions in eq. (5.39) are already normalized. We have not, however, normalized the functions $Y(\theta)$. This is rather easy. We require that

$$\int_0^{2\pi} |Y(\theta)|^2 d\theta = 1. \quad (5.41)$$

The normalized functions are

$$Y(\theta) = \frac{1}{\sqrt{2\pi}} e^{im\theta}. \quad (5.42)$$

The total normalized eigenfunctions to \hat{H}_Ω , including the angular part, is then

$$\begin{aligned}\psi_{nm}(r, \theta) &= R(r)Y(\theta) \\ &= \sqrt{\frac{n!}{\pi (n + |m|)!}} \beta^{\frac{1}{2}(|m|+1)} r^{|m|} e^{-\frac{\beta}{2}r^2} L_n^{|m|}(\beta r^2) e^{im\theta}.\end{aligned}\quad (5.43)$$

The corresponding energy eigenvalues are

$$E_{nm} = (2n + |m| + 1) \hbar\omega + m\hbar\omega_B. \quad (5.44)$$

What are the values of n and m ? The quantum number n is found in the derivation of eq. (5.39). It can only take positive integer values

$$n = 0, 1, 2, .. \quad (5.45)$$

The quantum number m is related to the angular part of the wave function. Our system should be invariant under a 2π -rotation, so we make the demand that

$$\frac{1}{\sqrt{2\pi}} e^{im\theta} = \frac{1}{\sqrt{2\pi}} e^{im(\theta+2\pi)} \quad (5.46)$$

$$= \frac{1}{\sqrt{2\pi}} (e^{im\theta} e^{im2\pi}). \quad (5.47)$$

This indicates that

$$e^{i2m\pi} = 1 \quad (5.48)$$

This is always true if m is an integer. We therefore demand that

$$m = 0, \pm 1, \pm 2, .. \quad (5.49)$$

The total energy of the system in eq. (5.22) can now be written as

$$E_{nmm_s} = (2n + |m| + 1) \hbar\omega + m\hbar\omega_B - m_s \hbar g \omega_B + e\phi. \quad (5.50)$$

The last term in this equation is not really important. We know that the potential ϕ is constant, so the term $e\phi$ is just a constant addition to the energy. We therefore ignore this term in all calculations from now on. This gives us an energy

$$E_{nmm_s} = (2n + |m| + 1) \hbar\omega + m\hbar\omega_B - m_s \hbar g \omega_B. \quad (5.51)$$

We now consider the case $\omega_B = 0$. This is the same as saying there is no external magnetic field, i. e. $B = 0$. This leads to the energy

$$E_{nm}^{B=0} = (2n + |m| + |) \hbar\omega_0. \quad (5.52)$$

These energies are degenerate with a degeneracy

$$d = 2n + |m| + 1. \quad (5.53)$$

Remembering that each electron has two spin options, giving a total degeneracy of $D = 2d$. This is what we call a *shell structure*. It is often seen in nature, f. ex. in the Hydrogen atom. We define the *shell number*

$$R = 2n + |m| + 1. \quad (5.54)$$

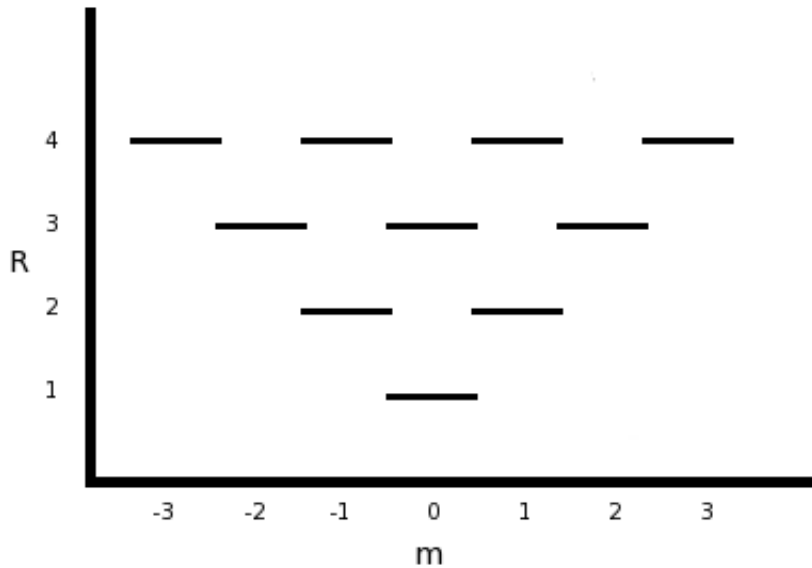


Figure 5.1: Shows the shell structure of a one-electron quantum dot. The vertical axis shows the shell number R , the horizontal axis shows the angular quantum number m . Each vertical bar can contain two particles due to spin degeneracy. This means that the first shell can contain two particles, the next can contain four, and so on. Generally, each shell can contain $2R$ particles (in two dimensions).

Table 5.1: The table shows the degeneracy of each shell number R , and how many particles it takes to fill up all shells up to, and including that shell number.

Shell number R	Degeneracy $D = 2d$	Shell filling S
1	2	2
2	4	6
3	6	12
4	8	20

The shell structure is depicted in Table 5.1 and Figure 5.1 for the first four shell numbers.

The shell filling S shown in Table 5.1 represent the number of non-interacting electrons that are needed to fill up all one-particle orbitals up to the relevant shell number. They are so-called *magic numbers*, as seen in the periodic table for atoms. Quantum dots that are filled up with a number of particles equivalent to one of the magic numbers define what we call the *closed-shell* quantum dots.

Turning on the magnetic field, we move from eq. (5.52) to eq. (5.51). We now rewrite eq. (5.51), using eq. (5.15), for simplicity we leave out the spin dependent part:

$$\frac{E_{nm}}{\hbar\omega_0} = (2n + |m| + 1) \sqrt{1 + \frac{\omega_B^2}{\omega_0^2}} + m \frac{\omega_B}{\omega_0}. \quad (5.55)$$

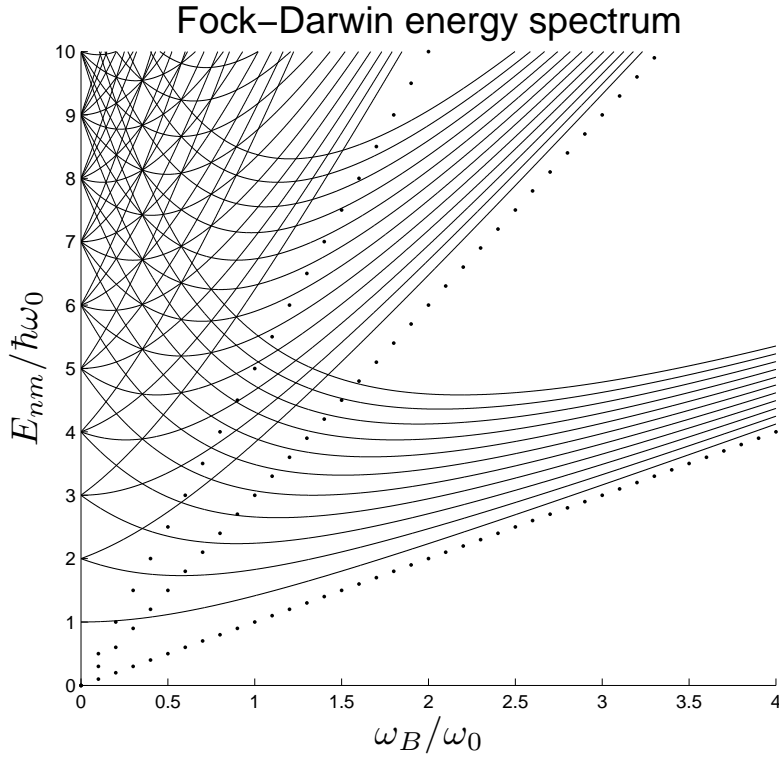


Figure 5.2: Shows the two-dimensional Fock-Darwin energy spectrum for a single-electron quantum dot. When ω_B is increased from $\omega_B = 0$, the degeneracy in eq. (5.39) is split up. This is because of the energy dependence on the sign of m in eq. (5.34). Comparing with 5.1, the shell number $R = 2$ has two possibilities for m , namely $m = \pm 1$. This is picture in the figure above as a splitting in the energy for $E_{nm}/\hbar\omega_0 = 2$ when ω_B increases from zero. One path is for $m = 1$, the other is for $m = -1$. If we were to follow one of the energy lines as ω_B increases from zero, the “energy path” of the particle would become a zig-zag line. One line will eventually cross another line, forming a sudden degeneracy. When ω_B is increased further, the particle will choose the path with the lowest energy (assuming the state is available). Another feature that is clearly visible is the forming of Landau bands. As the magnetic field is increased to the high-field limit, energy states converge to certain Landau levels shown as dotted lines in the figure.

This leads to what is called the Fock-Darwin energy spectrum shown in Figure 5.2. When the magnetic field is off, the energy is degenerate, as shown in Table 5.1 and Figure 5.1. As we turn on the magnetic field, the term $m\hbar\omega_B$ in eq. (5.51) causes a splitting of the degenerate states. The energy of the states increases or decreases, depending on the sign of m . If we increase ω_B further, it causes sudden degeneracies to appear with states of different shell numbers. If the magnetic field is increased more, a particle in such a degenerate state will choose the state with the most favorable energy (if it is available). Incrementing ω_B further leads to new degeneracies, and so on. In the high-field limit ($\omega_B \rightarrow \infty$), the energy lines seem to gather in bands. These bands are called Landau bands (ref. [2]). In this limit, eq. (5.51) reads (without spin energy)

$$\lim_{\omega_B \rightarrow \infty} E_{nm} = (2n + |m| + 1 + m)\hbar\omega_B. \quad (5.56)$$

We split this equation up in two parts, one for $m > 0$ and one for $m \leq 0$. In the first case

we have

$$\lim_{\omega_B \rightarrow \infty} E_{nm} = (2(n+m) + 1)\hbar\omega_B, \quad (5.57)$$

in the second case, the energy becomes independent from m

$$\lim_{\omega_B \rightarrow \infty} E_n = (2n + 1)\hbar\omega_B. \quad (5.58)$$

We define the Landau band quantum number

$$N_L = 0, 1, 2, .. \quad (5.59)$$

With this, we can write the high field approximation of the energy as

$$E_{N_L} = (2N_L + 1)\frac{\omega_B}{\omega_0}, \quad (5.60)$$

called Landau energies. We have plotted the first three of these energies in Figure 5.2 as dotted lines. When the magnetic field increases, the Landau bands in the figure converge to these Landau energies.

Since we have not included the spin energy in eq. (5.55), the lines in Figure 5.2 are degenerate. Including the spin energy would break each of the lines into two, making the energy spectra even more complicated.

5.2 Two-electron quantum dot

It has been shown (ref. [34]) that there exists particular analytical solutions of the two-dimensional quantum dot in a homogeneous magnetic field. The solutions exist only for some particular values of the oscillator frequency in eq. (5.15). This is of importance for us because these analytical solutions give us a good benchmark for our code when we do our numerical simulations. Our code must be able to reproduce these results.

The Hamiltonian for the two-particle system is

$$\hat{H} = \sum_{i=1}^2 \left[\frac{1}{2m^*} (\mathbf{p}_i - e\mathbf{A}_i)^2 + \frac{1}{2}m^*\omega_0^2\epsilon_r r_i^2 \right] + \frac{e^2}{4\pi\epsilon_0\epsilon_r r_{12}}, \quad (5.61)$$

where ϵ_0 is the vacuum permittivity, ϵ_r is the relative permittivity, $r_i^2 = x_i^2 + y_i^2$, and $r_{12} = |\mathbf{r}_2 - \mathbf{r}_1|$. We have not included the spin, because it gives the same addition in the energy as in the previous section. The only difference now is that we have to sum the contributions from both of the particles. We assume that the magnetic field is given as $B_0\mathbf{k}$, where B_0 is a constant and \mathbf{k} is the unit vector in z -direction. We want to introduce center-of-mass coordinates. The transformation is given by

$$\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1, \quad (5.62)$$

and

$$\begin{aligned} \mathbf{R} &= \frac{m_1\mathbf{r}_1 + m_2\mathbf{r}_2}{m_1 + m_2} \\ &= \frac{\mathbf{r}_1 + \mathbf{r}_2}{2}, \end{aligned} \quad (5.63)$$

where \mathbf{R} is the center-of-mass term and \mathbf{r} is the relative term. We have used that the particles are identical, and therefore have identical mass. The momenta transforms in almost the same way as the coordinates

$$\mathbf{p} = \frac{\mathbf{p}_2 - \mathbf{p}_1}{2}, \quad (5.64)$$

$$\mathbf{P} = \mathbf{p}_1 + \mathbf{p}_2. \quad (5.65)$$

Since the magnetic field is constant, Maxwell's equations implies that the vector field \mathbf{A} must be linear. This means we transform it as

$$\mathbf{A}(\mathbf{r}) = \mathbf{A}(\mathbf{r}_2) - \mathbf{A}(\mathbf{r}_1) \quad (5.66)$$

$$\mathbf{A}(\mathbf{R}) = \frac{\mathbf{A}(\mathbf{r}_1) + \mathbf{A}(\mathbf{r}_2)}{2} \quad (5.67)$$

We need the following relations when we transform the Hamiltonian:

$$\mathbf{A}_R = 2\mathbf{A}(\mathbf{R}), \quad (5.68)$$

$$\mathbf{A}_r = \frac{1}{2}\mathbf{A}(\mathbf{r}), \quad (5.69)$$

$$\mathbf{p}_1^2 + \mathbf{p}_2^2 = \frac{1}{2}(\mathbf{P}^2 + 4\mathbf{p}^2), \quad (5.70)$$

$$r_1^2 + r_2^2 = \frac{1}{2}(4R^2 + r^2), \quad (5.71)$$

$$r_{12} = r, \quad (5.72)$$

$$\mathbf{p}_1 \cdot \mathbf{A}(\mathbf{r}_1) + \mathbf{p}_2 \cdot \mathbf{A}(\mathbf{r}_2) = \frac{1}{2}\mathbf{P} \cdot \mathbf{A}_R + 2\mathbf{p} \cdot \mathbf{A}_r, \quad (5.73)$$

$$\mathbf{A}(\mathbf{r}_1)^2 + \mathbf{A}(\mathbf{r}_2)^2 = \frac{1}{2}\mathbf{A}_R^2 + 2\mathbf{A}_r^2, \quad (5.74)$$

$$\omega_R = 2\omega_0, \quad (5.75)$$

$$\omega_r = \frac{1}{2}\omega_0. \quad (5.76)$$

We write out the Hamiltonian

$$\begin{aligned}
 \hat{H} &= \frac{1}{2m^*} [\mathbf{p}_1^2 + \mathbf{p}_2^2 - 2e(\mathbf{p}_1 \cdot \mathbf{A}(\mathbf{r}_1) + \mathbf{p}_2 \cdot \mathbf{A}(\mathbf{r}_2)) + e^2(\mathbf{A}(\mathbf{r}_1)^2 + \mathbf{A}(\mathbf{r}_2)^2)] \\
 &\quad + \frac{1}{2}m^*\omega_0^2\epsilon_r(r_1^2 + r_2^2) + \frac{e^2}{4\pi\epsilon_0\epsilon_rr} \\
 &= \frac{1}{2m^*} \left[\frac{1}{2}\mathbf{P}^2 + 2\mathbf{p}^2 - 2e\left(\frac{1}{2}\mathbf{P} \cdot \mathbf{A}_R + 2\mathbf{p} \cdot \mathbf{A}_r\right) + e^2\left(\frac{1}{2}\mathbf{A}_R^2 + 2\mathbf{A}_r^2\right) \right] \\
 &\quad + \frac{1}{2}m^*\omega_0^2\epsilon_r\frac{1}{2}(4R^2 + r^2) + \frac{e^2}{4\pi\epsilon_0\epsilon_rr} \\
 &= \frac{1}{2} \left\{ \frac{1}{2m^*} [\mathbf{P}^2 - 2e\mathbf{P} \cdot \mathbf{A}_R + e^2\mathbf{A}_R^2] + \frac{1}{2}m^2\omega_R^2R^2 \right\} \\
 &\quad + 2 \left\{ \frac{1}{2m^*} [\mathbf{p}^2 - 2e\mathbf{p} \cdot \mathbf{A}_r + e^2\mathbf{A}_r^2] + \frac{1}{2}m^*\omega_r^2r^2 + \frac{e^2}{4\pi\epsilon_0\epsilon_rr} \right\} \\
 &= \frac{1}{2} \left\{ \frac{1}{2m^*} [\mathbf{P}^2 - e\mathbf{A}_R]^2 + \frac{1}{2}m^2\omega_R^2R^2 \right\} \\
 &\quad + 2 \left\{ \frac{1}{2m^*} [\mathbf{p} - e\mathbf{A}_r]^2 + \frac{1}{2}m^*\omega_r^2r^2 + \frac{e^2}{4\pi\epsilon_0\epsilon_rr} \right\} \\
 &= \frac{1}{2}\hat{H}_R + 2\hat{H}_r, \tag{5.77}
 \end{aligned}$$

where we in the last step have defined

$$\hat{H}_R = \frac{1}{2m^*} [\mathbf{P}^2 - e\mathbf{A}_R]^2 + \frac{1}{2}m^2\omega_R^2R^2, \tag{5.78}$$

and

$$\hat{H}_r = \frac{1}{2m^*} [\mathbf{p} - e\mathbf{A}_r]^2 + \frac{1}{2}m^*\omega_r^2r^2 + \frac{e^2}{4\pi\epsilon_0\epsilon_rr}. \tag{5.79}$$

This indicates we can write the wave function in a separable form as

$$\Psi(\mathbf{R}, \mathbf{r}) = \psi_R(\mathbf{R})\psi_r(\mathbf{r}), \tag{5.80}$$

where the separable solutions are eigenfunctions of the respective Hamiltonians:

$$\hat{H}_R\psi_R(\mathbf{R}) = E_R\psi_R(\mathbf{R}), \tag{5.81}$$

and

$$\hat{H}_r\psi_r(\mathbf{r}) = E_r\psi_r(\mathbf{r}). \tag{5.82}$$

The time independent Schrödinger equation for the whole system is

$$\begin{aligned}
 \hat{H}\Psi(\mathbf{R}, \mathbf{r}) &= \left(\frac{1}{2}\hat{H}_R + 2\hat{H}_r \right) \psi(\mathbf{R})\psi(\mathbf{r}) \\
 &= \frac{1}{2}\hat{H}_R\psi(\mathbf{R})\psi(\mathbf{r}) + 2\psi(\mathbf{R})\hat{H}_r\psi(\mathbf{r}) \\
 &= \left(\frac{1}{2}E_R + 2E_r \right) \psi(\mathbf{R})\psi(\mathbf{r}). \tag{5.83}
 \end{aligned}$$

The total energy is then

$$E = \frac{1}{2}E_R + 2E_r. \tag{5.84}$$

We make the following definitions

$$\begin{aligned}\bar{\omega}_r &= \sqrt{\omega_r^2 + \frac{\omega_B^2}{4}} \\ &= \frac{\omega}{2}\end{aligned}\tag{5.85}$$

$$\begin{aligned}\bar{\omega}_R &= \sqrt{\omega_R^2 + 4\omega_B^2} \\ &= 2\omega,\end{aligned}\tag{5.86}$$

where ω_B and ω is defined as in the previous section. We see that eq. (5.81) is just the same as the one-particle problem, but with different parameters. We are only interested in the energy, which is given as

$$\begin{aligned}E_R = E_{NM} &= (2N + |M| + 1)\hbar\bar{\omega}_R + 2M\hbar\omega_B \\ &= 2(2N + |M| + 1)\hbar\omega + 2M\hbar\omega_B.\end{aligned}\tag{5.87}$$

The ground state is given by $N = M = 0$, which gives an energy

$$E_{R,0} = 2\hbar\omega.\tag{5.88}$$

Solving eq. (5.82) is in general not possible because of the $\frac{1}{r}$ term in the Hamiltonian. However, for particular values of $\bar{\omega}_r$ it is possible to get closed-form expressions. We refer to [34] for a derivation of these solutions. The energies are given as

$$\begin{aligned}E_r = E_{nm} &= (n + |m|)\hbar\bar{\omega}_r + \frac{1}{2}m\hbar\omega_B \\ &= \frac{1}{2}(n + |m|)\hbar\omega + \frac{1}{2}m\hbar\omega_B.\end{aligned}\tag{5.89}$$

The possible solutions are found by first choosing a value for n . This in turn gives a value for ω . One problem is that solutions found this way is not always ground states. The simplest solutions are generated by $n = 2$. The frequency is then (ref. [34])

$$\bar{\omega}_r = \frac{1}{2(2|m| + 1)},\tag{5.90}$$

or

$$\omega = \frac{1}{2|m| + 1},\tag{5.91}$$

and luckily for us, these are all ground states (ref. [34]). We choose the state with $|m| = 0$, this gives $\omega = 1$. The energy is then

$$E_{r,0} = \hbar.\tag{5.92}$$

The total ground-state energy for $\omega = 1$ is found by eq. (5.84). We get $E_0 = 3\hbar$.

For the non-interacting case, the two-particle energy can be found by summing up the two lowest energy eigenvalues in eq. (5.44). This gives a ground-state energy of $E_0 = 2\hbar$ when $\omega = 1$. This means that in the interacting system, the interaction energy is one third of the total energy of the system.

5.3 The N -electron Hamiltonian

For an N -electron system, we need to solve the time-independent Schrödinger equation numerically. As before, we assume a magnetic field $\mathbf{B} = B_0\mathbf{k}$, and we approximate the confining potential with the harmonic oscillator potential. The N -electron Hamiltonian then reads

$$\hat{H} = \sum_{i=1}^N \left(\frac{1}{2m^*} (\hat{\mathbf{p}}_i - e\mathbf{A})^2 + \frac{1}{2}m^*\omega_0^2(\hat{x}_i^2 + \hat{y}_i^2) + e\phi - \hat{\boldsymbol{\mu}}_i \cdot \mathbf{B} \right) + \frac{e^2}{4\pi\epsilon_0\epsilon_r} \sum_{i<j}^N \frac{1}{r_{ij}}. \quad (5.93)$$

This is almost the same as eq. (5.6), but we have added the electron-electron interaction, and we now sum over several particles. Again, we assume that $\mathbf{B} = B_0\mathbf{k}$, and do the same definitions as in Section 5.1. For brevity we set, $\kappa \equiv \frac{4\pi\epsilon_0\epsilon_r\hbar}{e^2}$. The Hamiltonian can then be written as

$$\hat{H} = \sum_{i=1}^N \left(-\frac{\hbar^2}{2m^*} \nabla_i^2 - \omega_B \hat{L}_z^{(i)} + \frac{1}{2}m^*\omega^2(\hat{x}_i^2 + \hat{y}_i^2) - g\omega_B \hat{S}_z^{(i)} \right) + \frac{\hbar}{\kappa} \sum_{i<j}^N \frac{1}{r_{ij}}, \quad (5.94)$$

where we have set the term $e\phi = 0$ like before, for the same reason. Both L_z and S_z commute with the Hamiltonian, m and m_s are therefore good quantum numbers. The effects of these operators are therefore only constant (dependent only upon the relevant quantum number) additions to the energy (ref. [35]), this means we can write eq. (5.94) as

$$\hat{H} = \sum_{i=1}^N \left(-\frac{\hbar^2}{2m^*} \nabla_i^2 + \frac{1}{2}m^*\omega^2(\hat{x}_i^2 + \hat{y}_i^2) - \omega_B (m^{(i)} + gm_s^{(i)}) \right) + \frac{\hbar}{\kappa} \sum_{i<j}^N \frac{1}{r_{ij}}. \quad (5.95)$$

The contributions from the angular and spin part of the Hamiltonian are as stated really just constant additions. This means that there is no reason for us to carry them with us through all the calculations; we can just add them after the simulation. This means the Hamiltonian we need to simulate is just

$$\hat{H} = \sum_{i=1}^N \left(-\frac{\hbar^2}{2m^*} \nabla_i^2 + \frac{1}{2}m^*\omega^2(\hat{x}_i^2 + \hat{y}_i^2) \right) + \frac{\hbar}{\kappa} \sum_{i<j}^N \frac{1}{r_{ij}}. \quad (5.96)$$

Furthermore, in our simulations, we will only look at closed-shell system. This means that all states up to a certain shell is filled. Within a shell, there is a symmetry to the spin and angular quantum number

$$m = 0, \pm 1, \pm 2, \dots, \pm m_{max}, \quad (5.97)$$

$$m_s = \pm \frac{1}{2}, \quad (5.98)$$

where m_{max} is the maximum value of the angular quantum number in the relevant shell. When we add the angular and spin energy, the sum goes over all these quantum numbers. The positive and negative terms will then cancel each other out. Therefore, in a closed-shell system, there will be no spin and angular addition to the energy. The effect of the external magnetic field is then just to tune the oscillator frequency $\omega_0 \rightarrow \omega$.

5.4 Scaling the N -electron Hamiltonian

In order to simplify our calculations later on, we want to rescale our Hamiltonian to a dimensionless form. First of all, we define a new length unit α . We want to make all the length units in the Hamiltonian dimensionless. We define

$$x' \equiv \frac{x}{\alpha}, \quad y' \equiv \frac{y}{\alpha}, \quad r'_{ij} \equiv \frac{r_{ij}}{\alpha}, \quad \nabla' \equiv \alpha \nabla. \quad (5.99)$$

We can then rewrite the Hamiltonian:

$$\hat{H} = \sum_{i=1}^N \left(-\frac{\hbar^2}{2m^*} \nabla_i^2 - \omega_B \hat{L}_z^{(i)} + \frac{1}{2} m^* \omega^2 (x_i^2 + y_i^2) - g \omega_B \hat{S}_z^{(i)} \right) + \frac{\hbar}{\kappa} \sum_{i<j}^N \frac{1}{r_{ij}}, \quad (5.100)$$

$$= \sum_{i=1}^N \left(-\frac{\hbar}{2m^* \alpha^2} \alpha^2 \nabla_i^2 + \frac{1}{2} m^* \omega^2 \alpha^2 \frac{x_i^2 + y_i^2}{\alpha^2} \right) + \frac{\hbar}{\kappa \alpha} \sum_{i<j}^N \frac{\alpha}{r_{ij}} \quad (5.101)$$

$$= \sum_{i=1}^N \left(-\frac{\hbar^2}{2m^* \alpha^2} \nabla_i'^2 + \frac{1}{2} m^* \omega^2 \alpha^2 (x_i'^2 + y_i'^2) \right) + \frac{\hbar}{\kappa \alpha} \sum_{i<j}^N \frac{1}{r'_{ij}}. \quad (5.102)$$

We want to express the Hamiltonian in a dimensionless form. To do this, we multiply the whole equation with $\frac{m^* \alpha^2}{\hbar^2}$:

$$\frac{m^* \alpha^2}{\hbar^2} \hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i'^2 + \frac{1}{2} \left(\frac{m^* \omega \alpha^2}{\hbar} \right)^2 (x_i'^2 + y_i'^2) \right) + \frac{m^* \alpha}{\hbar \kappa} \sum_{i<j}^N \frac{1}{r'_{ij}}. \quad (5.103)$$

For our purpose, we require that

$$\alpha = \frac{\hbar \kappa}{m^*}. \quad (5.104)$$

Putting this back into eq. (5.103), we get

$$\frac{\kappa^2}{m^*} \hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i'^2 + \frac{1}{2} \left(\frac{\hbar \omega \kappa^2}{m^*} \right)^2 (x_i'^2 + y_i'^2) \right) + \sum_{i<j}^N \frac{1}{r'_{ij}}. \quad (5.105)$$

We define

$$\hat{H}' = \frac{\kappa^2}{m^*} \hat{H} \quad (5.106)$$

$$\omega' = \frac{\hbar \omega \kappa^2}{m^*}. \quad (5.107)$$

This gives us the final form of our dimensionless Hamiltonian

$$\hat{H}' = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i'^2 + \frac{1}{2} \omega'^2 (x_i'^2 + y_i'^2) \right) + \sum_{i<j}^N \frac{1}{r'_{ij}} \quad (5.108)$$

where ω' is a dimensionless oscillator frequency. We are later going to vary this when we calculate the energy, and see how this change affects the energy. By varying this frequency we are in effect actually varying the exterior magnetic field.

Using the scaled Hamiltonian, energy is given in units of $\frac{m^*}{\kappa^2} = m^* \left(\frac{e^2}{4\pi\epsilon_0\epsilon_r\hbar} \right)^2$, which is called Hartrees (symbol: E_H or Ha). Lengths are given in units of $\frac{m^*}{\hbar\kappa} = \frac{m^* e^2}{4\pi\epsilon_0\epsilon_r\hbar^2}$.

Chapter 6

Quantum Monte Carlo

As we already have stated a number of times, it is generally impossible to solve the many-body problem analytically. We therefore need numerical methods to get approximated results. Many different many-body techniques have been developed over the years, but in this thesis we are going to use the Monte-Carlo (MC) method. This method (or rather, series of methods) is an example of what we call an *ab initio* method. *Ab initio* is derived from the Latin *ab* (“from”) and *initio* (“beginning”). In science the term means something like “from first principles”. A calculation is said to be *ab initio* if it relies on basic and established laws of nature without additional assumptions or special models. In quantum mechanics this means that we want to solve the many-body Schrödinger equation from fundamental physical principles alone.

MC methods are based on using (pseudo-)random numbers — hence the name “Monte Carlo”. Any numerical technique in which random numbers play an essential role can be called a “Monte Carlo” method, after the famous Mediterranean casino town. Three main types of MC simulations can be distinguished (ref. [36])

Direct Monte Carlo, in which random numbers are used to model the effect of complicated processes, the details of which are not crucial. An example is the modelling of traffic where the behaviour of cars is determined in part by random numbers.

Monte Carlo integration, which is a method for calculating integrals using random numbers. This method is efficient when the integration is over high-dimensional volumes.

Metropolis Monte Carlo, in which a sequence of distributions of a system is generated in a so-called Markov chain. This method allows us to study the static properties of classical and quantum many-particle systems.

In this thesis, the first one is not important. The two others, however, are central to our problem.

By using random numbers, MC methods are considered stochastic, i. e. non-deterministic, unlike other simulation techniques, such as Hartree Fock, Coupled Cluster Theory and Configuration Interaction (ref. [37]). There are several Quantum MC techniques such as diffusion Monte Carlo (DMC), Green’s function Monte Carlo (GFMC), path-integral Monte Carlo (PIMC) and variational Monte Carlo (VMC). The last one is our choice

method.

In this chapter we will discuss the Monte Carlo method at some length. We start out by discussing random number and pseudo-random numbers. We then explain the basic idea behind Monte Carlo integration and importance sampling. We will look at Markov chains and the Metropolis-Hastings algorithm. We then move on to the Variational principle, before we discuss the quantum-variational-Monte-Carlo (QVMC) method, which utilize all of the concepts described before. We briefly discuss another algorithm for finding the energy minimum, called the *Davidon-Fletcher-Powell* algorithm (DFP). At the end of the chapter we discuss the blocking method that deals with correlations in our result, and a method called time-step extrapolation, which we use to get more correct results.

It is advisable that the reader reviews Appendix A before reading this chapter. Some basic concepts like PDFs and sample error are discussed in this appendix, and we will make use of the results in this chapter.

6.1 Random numbers

All MC methods are based upon the use of random numbers. Random numbers are characterised by the fact that their value cannot be predicted. More precisely, if we construct a sequence of random numbers, the probability density for a new number is independent of all the numbers generated so far (ref. [36]). Pure random numbers may occur in experiments; for example the radioactive decay of a nucleus is associated with a pure random number since it is not possible to predict when it will decay. One could therefore in principle build a device that supplies random number based on such radioactive decay, and indeed this has been done. This is, however, a cumbersome method, and one would in many instances prefer a method that can reproduce the same sequence of random numbers when it is necessary to check the consistency and accuracy of the produced results.

In our work, we will use random numbers generated by computers. In all computer generators the new numbers are generated from the previous ones by a mathematical formula (ref. [36]); the new values are in other words fully determined by the previous ones. Luckily, though, the numbers generated by computer algorithms have properties which make them suitable for simulations. They are therefore called *pseudo-random* numbers. The most crucial test of a pseudo-random-number generator is that the mean and variance of the number sequence produced match those of the desired PDF¹.

We will usually treat computer-generated numbers as though they are truly random, but in reality there will always exist a correlation between the numbers, and they will have a period before the same set of random numbers is repeated. The algorithms are usually based on a random seed that starts the sequence of numbers. This means that if you construct a sequence two times with the same initial seed, the same sequence of numbers is produced.

¹We advice that the reader review the appendix on statistics for a brief introduction to statistics and PDFs

6.1.1 Generation of uniform random numbers

We will mostly need random numbers of a uniform distribution. The PDF is given by

$$\rho(x) = \begin{cases} \frac{1}{b-a} & \text{for } a \leq x \leq b, \\ 0 & \text{else.} \end{cases} \quad (6.1)$$

These generators will return a random number

$$r \in [a, b], \quad (6.2)$$

with equal probability for every number in the range. In refs. [36, 38] is described some of the standard methods for generating uniform random numbers.

In most cases, we want a random number, r , in the range

$$r \in [0, 1]. \quad (6.3)$$

In our computations, we will use library functions to get these numbers.

Uniform random numbers are also of importance because the generation of arbitrary PDFs usually begins by transforming from a uniform distribution.

6.1.2 Generation of Gaussian distributed random number

Besides the uniform random numbers, we are also in need of random numbers that follow a Gaussian distribution. They have the PDF

$$\rho(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\nu)^2}{2\sigma^2}}, \quad (6.4)$$

where ν is the mean of the distribution, and σ^2 is the variance (ref. [38]). We will use a library function when we generate these numbers. The standard algorithm for creating Gaussian distributed numbers is based on uniform random numbers. It can be found in ref. [38].

For more information on the subject of random and pseudo-random numbers, we refer to [38, 39].

6.2 Monte Carlo integration

In quantum mechanics (as well as other fields of physics), one often face the problem of solving an integral on the form

$$I = \int_a^b dx f(x). \quad (6.5)$$

There are plenty of numerical methods to solve such integrals. Standard methods usually boil down to evaluating the function on a set of (usually, but not always) equally spaced values x_i and then evaluate the sum

$$I = \frac{b-a}{N} \sum_{i=1}^N w_i f(x_i), \quad (6.6)$$

where N is the number of points evaluated, and w_i are some weights that do not depend on f , but determine the accuracy of the method. Equation 6.6 is also used in MC integration, with all weights $w_i = 1$, but the points x_i chosen *randomly*.

When we use random numbers to calculate the integral by eq. (6.5), we have to remember that the random numbers are uniformly distributed. We need to make a change of variables in order to adapt to the integration limits a and b . This is quite easily done by defining

$$x = a + (b - a)y, \quad (6.7)$$

where y is a random number in the range $y \in [0, 1]$. We further have that

$$dx = (b - a)dy. \quad (6.8)$$

Substituting this in eq. (6.5), assuming $a \neq b$, we get

$$I = \frac{1}{b - a} \int_0^1 dy f(x(y)). \quad (6.9)$$

This together with eq. (6.6) yields the following sum for the MC integration (remembering we set the weights $\omega_i = 1$)

$$I = \frac{1}{N} \sum_{i=1}^N f(x(y_i)). \quad (6.10)$$

If the random points x_i are distributed homogeneously on $[0, 1]$, and if N is sufficiently large, the sum in eq. (6.6) yields a result close to the exact integral. This result has a corresponding variance which we write as

$$\sigma^2 \approx \langle f^2 \rangle - \langle f \rangle^2, \quad (6.11)$$

where we have defined

$$\langle f \rangle = \frac{1}{N} \sum_{i=1}^N f(x(y_i)), \quad \text{and} \quad \langle f^2 \rangle = \frac{1}{N} \sum_{i=1}^N f(x(y_i))^2. \quad (6.12)$$

The error is given as²

$$\begin{aligned} \text{err}^2 &= \frac{\sigma^2}{N} \\ \text{err} &= \sqrt{\frac{\sigma^2}{N}} \end{aligned} \quad (6.13)$$

which means the error scales as $(N)^{-\frac{1}{2}}$. See Appendix A for a derivation of this equation.

Compared to standard numerical integration techniques, this seems unfavourable. Standard quadrature methods yield an error proportional to N^{-k} with $k \geq 1$ (ref. [36]). However, MC integration is more efficient in higher dimensions. To see this, consider integration of a hypercube with sides L and dimension d . Standard quadrature methods

²For the moment we will assume no correlations in the random numbers. We will later discuss a method call blocking which deals with correlation.

has equidistant values for the evaluation points x_i . The separation value is h . The number of integration points in the volume will be

$$N = \left(\frac{L}{h}\right)^d. \quad (6.14)$$

And the error in the result scales as $N^{-\frac{k}{d}}$ (ref. [36]). The error in MC integration, on the other hand, is independent of the dimensionality. Comparing this to eq. (6.14), we see that MC integration is more efficient when $d > 2k$. Monte-Carlo integration is in addition much faster for higher dimensions than standard quadrature methods.

6.2.1 Importance sampling

The scheme described above is very “brute-force”-ish, meaning we do not take into account that the contributions to the integral from different regions in the integration volume might vary strongly. Standard MC integration samples the function homogeneously (meaning that they sample with the uniform distribution), so if the significant contributions to the integral come from a small region within the integration volume, there will be only a few sample points there. This can lead to large statistical errors. One way to improve upon this is by concentrating the sampling in the regions where $|f|$ happens to be large. This is called *importance sampling*.

Let $\rho(x)$ be a PDF on $[a, b]$ that has more or less the shape of $f(x)$ in the sense that

$$\frac{f(x)}{\rho(x)} \approx \text{constant}. \quad (6.15)$$

We require that

$$\int_a^b dx \rho(x) = 1. \quad (6.16)$$

We rewrite eq. (6.5)

$$\begin{aligned} I &= \int_a^b dx f(x) \\ &= \int_a^b dx \rho(x) \left[\frac{f(x)}{\rho(x)} \right]. \end{aligned} \quad (6.17)$$

Random numbers are generated through the uniform distribution, but we want to generate random numbers that follow the distribution of $\rho(x)$. We make a change of variables. Let $p(y)$ be the uniform distribution, then we have

$$p(y)dy = dy = \rho(x)dx, \quad (6.18)$$

which yields

$$y(x) = \int_a^x \rho(x')dx'. \quad (6.19)$$

If $y(x)$ is invertible, we can find $x(y)$ as well. We then rewrite eq. (6.17) as

$$\begin{aligned} I &= \int_a^b dx \rho(x) \left[\frac{f(x)}{\rho(x)} \right] \\ &= \int_0^1 dy \frac{f(x(y))}{\rho(x(y))}, \end{aligned} \quad (6.20)$$

where the integration limits have been changed to suit the uniform distribution. With the integral on the form of eq. (6.20), we can rewrite eq. (6.6)

$$I = \frac{1}{N} \sum_{i=1}^N \frac{f(x(y_i))}{\rho(x(y_i))}. \quad (6.21)$$

We define

$$\bar{f}(x(y_i)) = \frac{f(x(y_i))}{\rho(x(y_i))}. \quad (6.22)$$

The error still scales as in eq. (6.13).

The advantage of such a change of variable (in case $\rho(x)$ follows closely $f(x)$) is that the integrand becomes smooth and we can sample over relevant values for the integrand. This allows us to get a better result while using fewer samples than we would have if we did not use importance sampling.

6.3 Markov chains and the Metropolis-Hastings algorithm

In a MC calculation we first prepare a system in a configuration, calculate an observable, reconfigure the system, calculate the observable again, and so on. This is done several times, and at the end we calculate the mean of the observable. The question is then: How do we choose the configuration of the system for each step?

All MC schemes used are based on Markov processes in order to generate new random states/configurations (ref. [38]). The reason for choosing a Markov process is: When a Markov process is run for a long enough time starting with a random state, we will eventually reach the most likely state of the system (ref. [38]). To reach this distribution, the Markov process needs to obey two important criteria; namely that of *ergodicity* and *detailed balance*. The ergodicity criteria has two conditions (ref. [36]):

1. Every configuration which we want to be included in the ensemble should be accessible from every other configuration within a finite number of steps. This property is called *connectedness* or *irreducibility*.
2. There should be no *periodicity*. Periodicity means that after visiting a particular configuration, it should not be possible to return to the same configuration except after $t = nk$ steps, where n is some positive integer and k is a fixed positive integer.

We will come to the criteria of detailed balance later.

The theory of Markov chains is based on *random walks*. In a random walk, one defines a mathematical entity, called a walker, whose attributes completely define the state of the system in question. In our case the state of the system refers to the state of a many-electron quantum dot. The walker moves in the state space by a combination of deterministic and random displacements from its previous position. This sequence then forms a chain. This chain is a Markov chain if the state-to-state transition probabilities are independent of time and history — that is, they depend only on the current state of the system, not on how or when it got there (ref. [39]).

Assume that the system is described by a PDF $\omega_i(t)$, where the subscript i denotes the state of the system, and t denotes the “time”, or Markov step. This is the same as saying that $\omega_i(t)$ gives us the probability of the configuration i to occur at time t . We define transition probabilities W_{ij} , which brings the system from state j to state i . We can then set up the following equation

$$\omega_i(t + \Delta t) = \sum_j W_{ij} \omega_j(t), \quad (6.23)$$

where Δt is a chosen time step. We require W and ω to be normalized

$$\sum_i \omega_i(t) = 1 \quad (6.24)$$

$$\sum_j W_{ij} = 1, \quad (6.25)$$

since they are both probabilities. The change in the function $\omega_i(t)$ from one step to the other is governed by two processes:

1. Going from a configuration i at time t to some other configuration j at time $t + \epsilon$. This means there is a decrease in the probability of finding the system in state i .
2. Going from a configuration j at time t to some other configuration i at time $t + \epsilon$. This means there is a decrease in the probability of finding the system in state j .

This change can be summed up in the following equation

$$\omega_i(t + \epsilon) - \omega_i(t) = \sum_j \{W_{ij} \omega_j(t) - W_{ji} \omega_i(t)\}. \quad (6.26)$$

This equation is a discretized version of what is called the *master equation* (ref. [36]).

We are trying to find the stationary distribution, also called the steady state of the system. In this distribution there will be no noticeable change in the probability for state i during a change in time ϵ . In other words, we require

$$\omega_i(t + \epsilon) = \omega_i(t). \quad (6.27)$$

We can then rewrite eq. (6.26) to

$$\sum_j W_{ij} \omega_j(t) = \sum_j W_{ji} \omega_i(t). \quad (6.28)$$

We can now omit the t -dependence, since we have assumed ergodicity. It is difficult to find a general solution to eq. (6.28), but there is a particular solution that is immediately recognizable:

$$W_{ij} \omega_j = W_{ji} \omega_i. \quad (6.29)$$

For all pairs of configurations i and j . This solution is called the *detailed balance* solution. Picture this in the following way. The different configurations i are buckets, each containing an amount ω_i of water. We can make connections between each pair of buckets with pumps. Water is pumped from a bucket i to a bucket j with pumping rate $W_{ij} \omega_j$. Equation (6.29) then states that the pumping rates between each pair of buckets are

balanced in such a way that the flow from i to j is equal to the flow from j to i . The volumes ω_i and ω_j does not change. This holds for all pairs of buckets. The whole set of water volumes in the buckets will remain stationary.

We can rewrite eq. (6.29) on the form

$$\frac{W_{ij}}{W_{ji}} = \frac{\omega_i}{\omega_j}. \quad (6.30)$$

We (usually) know the probabilities ω , but unfortunately we do not know the analytic form of the transition probabilities W_{ij} . We are, however, free to model it as

$$W_{ij} = g_{ij}A_{ij}, \quad (6.31)$$

where we interpret g_{ij} as a *trial step probability* and A_{ij} as an *acceptance probability*. There are now two paths to take.

1. We demand that the matrix g is symmetric, this is the same as demanding that the probability for stepping from i to j is equal to the probability for stepping from j to i . We then have that $g_{ij} = g_{ji}$, and we rewrite eq. (6.30) on the form

$$\frac{A_{ij}}{A_{ji}} = \frac{\omega_i}{\omega_j}. \quad (6.32)$$

We then define the ratio

$$R = \frac{\omega_i}{\omega_j}. \quad (6.33)$$

This is the procedure in what we call *simple Metropolis sampling*.

2. Instead of letting each random walker attempt to move to any state with equal probability as in the simple Metropolis sampling, we can try to guide the walkers in the direction of the steady state. This is called *generalized Metropolis sampling*. We then need to find an appropriate form of the trial step probabilities g_{ij} . We rewrite eq. (6.30) as

$$\frac{A_{ij}}{A_{ji}} = \frac{g_{ji}\omega_i}{g_{ij}\omega_j}. \quad (6.34)$$

We then define the ratio

$$R = \frac{g_{ji}\omega_i}{g_{ij}\omega_j}. \quad (6.35)$$

Generalized Metropolis sampling is of greatest use where a transition function/probability g_{ij} is known that generates a PDF close to the steady state ω (ref. [39]).

We now introduce the *Metropolis-Hastings algorithm*. This algorithm was first proposed by Metropolis *et al.* (ref. ([40])) in 1953, then for the special case of the Boltzmann distribution. It was later extended to the more general case in 1970 by W. K. Hastings (ref. [41]). This algorithm can be summarized as follows:

$$A_{ij} = \begin{cases} R & \text{if } R < 1 \\ 1 & \text{if } R \geq 1 \end{cases}, \quad (6.36)$$

or

$$A_{ji} = \begin{cases} \frac{1}{R} & \text{if } R \geq 1 \\ 1 & \text{if } R < 1 \end{cases}, \quad (6.37)$$

with R defined as above (either simple or generalized Metropolis sampling). A step from j to i is accepted with probability A_{ij} , and rejected with probability $1 - A_{ij}$. This is usually done by generating a random number $r \in [0, 1]$. If $r < A_{ij}$, the new state is accepted, else it is rejected.

It is easy to check that the Metropolis algorithm is consistent with eq. (6.30). Say we want to use simple Metropolis sampling (the proof is equivalent for generalized Metropolis sampling). Then we need to check that eq. (6.32) is satisfied with the choice of the acceptance probabilities. We have to check two cases:

1. $R < 1$:

$$\begin{aligned} A_{ij} &= R = \frac{\omega_i}{\omega_j} \\ A_{ji} &= 1 \\ \frac{A_{ij}}{A_{ji}} &= \frac{\frac{\omega_i}{\omega_j}}{1} = \frac{\omega_i}{\omega_j} \quad \text{QED.} \end{aligned}$$

2. $R \geq 1$

$$\begin{aligned} A_{ij} &= 1 \\ A_{ji} &= \frac{1}{R} = \frac{\omega_j}{\omega_i} \\ \frac{A_{ij}}{A_{ji}} &= \frac{1}{\frac{\omega_j}{\omega_i}} = \frac{\omega_i}{\omega_j} \quad \text{QED.} \end{aligned}$$

The Metropolis-Hastings algorithm satisfies the conditions for both ergodicity and detailed balance (ref. [38]). The theory of Markov chains then guarantees that we arrive at the steady state distribution by using the Metropolis-Hastings algorithm.

6.4 The variational principle

The time independent Schrödinger equation reads

$$\hat{H}\Psi_n = E_n\Psi_n, \tag{6.38}$$

where \hat{H} , Ψ_n and E_n is the Hamiltonian, energy eigenfunctions and energy eigenvalues of the system under consideration, respectively. In this thesis we want calculate the ground state energy of the system, E_0 . Unfortunately, it is usually impossible to solve eq. (6.38) analytically. We can, however, make a guess at the ground state wave function. We define a *trial wave function* Ψ_T . The energy associated with this wave function is

$$E_T = \frac{\langle \Psi_T | \hat{H} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle}, \tag{6.39}$$

where the denominator stems from the fact that we have not required Ψ_T to be normalized. The variational principle then states that

$$E_0 \leq \frac{\langle \Psi_T | \hat{H} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle}, \tag{6.40}$$

which means that expectation value of \hat{H} , in the (presumably incorrect) state ψ_T is certain to overestimate the ground state energy. We will in the following prove the variational principle.

Since the eigenstates of \hat{H} form a complete set, we can expand our trial wave function in this basis

$$\Psi_T = \sum_n c_n \Psi_n. \quad (6.41)$$

We then have that

$$\begin{aligned} E_T &= \frac{\langle \Psi_T | \hat{H} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle} = \frac{\sum_{n,m=0}^{\infty} c_n^* c_m \langle \Psi_n | \hat{H} | \Psi_m \rangle}{\sum_{i,j=0}^{\infty} c_i^* c_j \langle \Psi_i | \Psi_j \rangle} \\ &= \frac{\sum_{n,m=0}^{\infty} c_n^* c_m E_m \langle \Psi_n | \Psi_m \rangle}{\sum_{i,j=0}^{\infty} c_i^* c_j \delta_{ij}} = \frac{\sum_{n,m=0}^{\infty} c_n^* c_m E_m \delta_{nm}}{\sum_{i=0}^{\infty} |c_i|^2} = \frac{\sum_{n=0}^{\infty} |c_n|^2 E_n}{\sum_{i=0}^{\infty} |c_i|^2}. \end{aligned} \quad (6.42)$$

where we have used that the energy eigenfunctions are orthogonal. By definition, the ground state energy is the smallest eigenvalue, that is

$$E_0 < E_n \quad \text{for } n > 0, \quad (6.43)$$

which means that

$$E_T = \frac{\sum_{n=0}^{\infty} |c_n|^2 E_n}{\sum_{i=0}^{\infty} |c_i|^2} \geq \frac{\sum_{n=0}^{\infty} |c_n|^2 E_0}{\sum_{i=0}^{\infty} |c_i|^2} = E_0. \quad (6.44)$$

This inequality will only be an equality when $\Psi_T = \Psi_0$.

The variational principle gets us an upper bound of the ground state energy E_0 . The main goal of our thesis is to approximate this energy. The way we do this is by first making a wave function *ansatz*

$$\Psi_T = \Psi_T(\{\mathbf{x}\}, \{\alpha_i\}), \quad (6.45)$$

where $\{\mathbf{x}\}$ is the positions of the particles in the system, and $\{\alpha_i\}$ is a set of variational parameters which we introduce to the wave function. The number of variational parameters usually varies; in this thesis we do not introduce more than two. By introducing these parameters, the energy E_T becomes

$$E_T = E_T(\{\alpha_i\}) \quad (6.46)$$

Our goal is then to find the minimum of this energy by varying the parameters $\{\alpha_i\}$. This is the main concept behind the Variational Monte Carlo (VMC) method. One should note that the degree of accuracy of this method depends on our choice of trial wave function. If we choose a function that does not remotely follow the form of the true wave function, the energy approximation will not be good. Therefore it is important to use our understanding of the physics of the system to construct good trial wave functions.

6.5 The quantum variational Monte Carlo method (QVMC)

In the previous sections, we have discussed basic MC methods. We will now proceed to see how we make use of these methods to get an estimate of the ground state energy of a quantum mechanical many-particle system, using the QVMC method. We start out with seeing how the actual energy calculation is done, then discuss how the sampling is done. We then set up the actual algorithm. At the end we will discuss what is called importance sampling in the QVMC scheme. This is a way to improve upon the basic algorithm.

6.5.1 Calculating the energy and variance of our system

Given a Hamiltonian, \hat{H} , the energy of a system is given as the expectation value of the Hamiltonian

$$\langle E \rangle = \langle \hat{H} \rangle = \frac{\int d\boldsymbol{\tau} \Psi_T^*(\boldsymbol{\tau}, \{\alpha_i\}) \hat{H}(\boldsymbol{\tau}) \Psi_T(\boldsymbol{\tau}, \{\alpha_i\})}{\int d\boldsymbol{\tau} \Psi_T^*(\boldsymbol{\tau}, \{\alpha\}) \Psi_T(\boldsymbol{\tau}, \{\alpha\})}, \quad (6.47)$$

where we define $\boldsymbol{\tau} \equiv (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$, and $d\boldsymbol{\tau} \equiv d\mathbf{x}_1 d\mathbf{x}_2 \dots d\mathbf{x}_N$, with N as the number of particles. We want to calculate this integral for a high number of particles. This integral has a dimensionality of $2N$. When N gets large, MC integration is usually the most efficient way to evaluate this integral. For brevity, we write $\Psi_T(\boldsymbol{\tau}, \{\alpha_i\}) = \Psi_T$. We rewrite eq. (6.47)

$$\langle E \rangle = \frac{\int d\boldsymbol{\tau} \Psi_T^* \hat{H} \Psi_T}{\int d\boldsymbol{\tau} \Psi_T^* \Psi_T} \quad (6.48)$$

$$= \frac{\int d\boldsymbol{\tau} \Psi_T^* (\Psi_T / \Psi_T) \hat{H} \Psi_T}{\int d\boldsymbol{\tau} \Psi_T^* \Psi_T} \quad (6.49)$$

$$= \int d\boldsymbol{\tau} \frac{|\Psi_T|^2}{\int d\boldsymbol{\tau} |\Psi_T|^2} \frac{\hat{H} \Psi_T}{\Psi_T} \quad (6.50)$$

$$= \int d\boldsymbol{\tau} \rho(\boldsymbol{\tau}) \hat{E}_L(\boldsymbol{\tau}), \quad (6.51)$$

where we have defined a probability density function

$$\rho(\boldsymbol{\tau}) = \frac{|\Psi_T(\boldsymbol{\tau})|^2}{\int d\boldsymbol{\tau} |\Psi_T(\boldsymbol{\tau})|^2}, \quad (6.52)$$

and the local energy operator

$$\hat{E}_L(\boldsymbol{\tau}) = \frac{\hat{H} \Psi_T(\boldsymbol{\tau})}{\Psi_T(\boldsymbol{\tau})}. \quad (6.53)$$

The local energy is constant if Ψ_T is an exact eigenfunction of the Hamiltonian. The more closely Ψ_T approaches the exact eigenfunction, the less strongly will E_L vary with $\boldsymbol{\tau}$ (ref. [36]). This means that the variance should go to zero as our trial wave function approaches the correct ground state. In most cases we will never get a variance of exactly zero, because our wave-function *ansatz* never actually takes the exact form of the correct ground state. In Section 9.1.1 we will, however, see that the variance indeed is zero if our wave-function *ansatz* is equal to the exact ground state.

We want to solve the integral in eq. (6.51) by MC integration. We do this with a Metropolis random-walk as discussed above. The energy approximation becomes

$$E \approx \frac{1}{M} \sum_{i=1}^M E_L(\mathbf{X}_i), \quad (6.54)$$

where $\{\mathbf{X}_i\}$ is the set of sample points, and M is the number of samples. There are two ways to sample the state space of the system. We can use simple Metropolis sampling, or we incorporate importance sampling by using generalized Metropolis sampling. The samples are then drawn from the PDF in eq. (6.52).

We also want to calculate the variance. We then need to calculate the energy squared. This is given by

$$E^2 = \int d\tau \rho(\tau) \hat{E}_L^2(\tau) \approx \frac{1}{M} \sum_{i=1}^M E_L^2(\mathbf{X}_i). \quad (6.55)$$

The variance is then given by

$$\sigma^2 \approx \langle E^2 \rangle - \langle E \rangle^2, \quad (6.56)$$

and the error is then (by eq. (6.13), still assuming no correlations)

$$\text{err} = \sqrt{\frac{\sigma^2}{N}}. \quad (6.57)$$

This means that for each cycle in the MC simulation, we need to calculate the local energy at the given particle positions. In these calculations, we use the dimensionless Hamiltonian in (5.108), restated here as

$$\hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega(x_i^2 + y_i^2) \right) + \sum_{i<j}^N \frac{1}{r_{ij}}. \quad (6.58)$$

The local energy is then given as

$$\begin{aligned} E_L(x, y) &= \frac{1}{\Psi_T} \left(\sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega(x_i^2 + y_i^2) \right) + \sum_{i<j}^N \frac{1}{r_{ij}} \right) \Psi_T, \\ &= -\frac{1}{\Psi_T} \sum_{i=1}^N \frac{1}{2} \nabla_i^2 \Psi_T + \sum_{i=1}^N \frac{1}{2} \omega(x_i^2 + y_i^2) + \sum_{i<j}^N \frac{1}{r_{ij}}. \end{aligned} \quad (6.59)$$

From this we calculate both $\langle E \rangle$ and $\langle E^2 \rangle$, giving us an estimate of both the energy and the variance.

In our thesis, we will try to vary the trial wave function to find an energy minimum. As stated, if we find the exact ground state, the variance should be zero. This means that one in principle could vary the trial wave function to find the minimum of the variance. Since we use a wave-function *ansatz*, we will in general not find a minimum for the variance and the energy for the same set of parameters. One could, after the MC simulation, use a linear combination of the parameters that minimize the energy, and the parameters that minimize the variance to construct the wave function. Indeed, this has proven to yield the most efficient wave functions (ref. [42]). We will however not do this, but focus on finding the parameters that minimize the energy.

6.5.2 Simple Metropolis sampling

Our system is described with the PDF in eq. (6.52). The Metropolis-Hastings algorithm is implemented as follows. For each MC cycle we choose new random positions for the particles that are dependent on the previous positions. We call this a *random walk*. For each particle, it is done by setting

$$\mathbf{r}_{new} = \mathbf{r}_{old} + \Delta r \boldsymbol{\eta}, \quad (6.60)$$

where \mathbf{r}_{new} is the new position of the particle, \mathbf{r}_{old} is the old position, Δr is a set step length, and $\boldsymbol{\eta}$ is a two-dimensional vector of uniformly distributed random numbers. When the new position for a particle is set, we perform a so-called *Metropolis test*. This test is done to determine if we should accept or reject the new position. We define $\boldsymbol{\tau}'$ as the set of particle coordinates where one particle has been moved to a new position, and $\boldsymbol{\tau}$ is the set of particle positions where that particle is at the old location. We then define the ratio according to the simple Metropolis method

$$R \equiv \frac{\rho(\boldsymbol{\tau}')}{\rho(\boldsymbol{\tau})} = \frac{|\Psi_T(\boldsymbol{\tau}')|^2}{|\Psi_T(\boldsymbol{\tau})|^2}, \quad (6.61)$$

and we define an acceptance probability

$$A = \begin{cases} R & \text{if } R < 1, \\ 1 & \text{if } R \geq 1. \end{cases} \quad (6.62)$$

Finally we compare this with a uniform random number χ . If χ is less than A , we accept the new position. If χ is larger than A , the new position is rejected, and we move the particle back to the old position. After this procedure is done for all particles, we calculate the local energy, and a new MC cycle can begin.

6.5.3 Generalized Metropolis sampling

The algorithm described above is not optimal. We do not discriminate when we choose new positions. Typical many-particle wave functions are very small in large parts of configuration space, and very large in other parts ([36]). This means that we might have problems in finding regions where the wave function is large, and that attempted moves of walkers from regions where the wave function is large will be rejected with high probabilities.

There exist more efficient approaches. The simplest form of the *diffusion equation* can be written as

$$\frac{\partial \rho(\boldsymbol{\tau}, t)}{\partial t} = D \nabla^2 \rho(\boldsymbol{\tau}, t), \quad (6.63)$$

where D is the diffusion constant. This equation describes how the probability density $\rho(\boldsymbol{\tau}, t)$ of a random walker evolves in time (ref. [36]). We want to construct a Markovian diffusion process that has a predefined distribution as its stationary distribution. Then stationary distribution we want to simulate is eq. (6.52). We then modify the simple diffusion equation in eq. (6.63) by adding a drift term that guides the distribution towards more likely regions. The Fokker-Planck equation

$$\frac{\partial \rho(\boldsymbol{\tau}, t)}{\partial t} = \sum_{i=1}^N D \nabla_i (\nabla_i - \mathbf{F}_i) \rho(\boldsymbol{\tau}, t), \quad (6.64)$$

describes this Markovian process (ref. [36]). To see this, assume we have reached the invariant distribution $\rho(\boldsymbol{\tau})$. Since this is an unchanging state, it is time independent. We must have that the left-hand side of eq. (6.64) is zero, namely

$$\sum_{i=1}^N \nabla_i (\nabla_i - \mathbf{F}_i) \rho(\boldsymbol{\tau}) = 0. \quad (6.65)$$

This has the trivial but uninteresting (and unnormalizable) solution $\rho = 0$. The simplest solution that makes sense is that each term in the sum is zero. This yields

$$\nabla_i (\nabla_i - \mathbf{F}_i) \rho(\boldsymbol{\tau}) = 0 \quad (6.66)$$

$$\nabla_i^2 \rho(\boldsymbol{\tau}) = \nabla_i (\mathbf{F}_i \rho(\boldsymbol{\tau})) \quad (6.67)$$

$$\nabla_i^2 \rho(\boldsymbol{\tau}) = (\nabla_i \mathbf{F}_i) \rho(\boldsymbol{\tau}) + \mathbf{F}_i \nabla_i \rho(\boldsymbol{\tau}). \quad (6.68)$$

This equation has the solution (ref. [39])

$$\mathbf{F}_i = \frac{1}{\rho(\boldsymbol{\tau})} \nabla_i \rho(\boldsymbol{\tau}). \quad (6.69)$$

This drift term causes the diffusion to be biased by Ψ_T . Equation (6.64) will guide $\rho(\boldsymbol{\tau}, t)$ towards the invariant solution $\rho(\boldsymbol{\tau})$ in a biased way, actually incorporating importance sampling.

Our stationary state is given by eq. (6.52). We then get the following drift term³

$$\begin{aligned} \mathbf{F}_i &= \frac{1}{\Psi_T^2} \nabla_i \Psi_T^2 \\ &= \frac{1}{\Psi_T^2} 2\Psi_T \nabla_i \Psi_T \\ &= 2 \frac{1}{\Psi_T} \nabla_i \Psi_T \end{aligned} \quad (6.70)$$

This vector field is usually called the *quantum force* of the system. Think of it like this: The gradient of the wave function tells us which direction the particles are moving. The quantum force term then steers our sampling in directions that are more relevant to our system, rather than just sampling random points that may actually be of very little interest. It actually takes us a bit of time to calculate the quantum force for each MC cycle. However, this is more than made up for by the fact that we need much fewer samples to get a good estimate of the energy.

We now have a diffusion equation which gives the desired distribution. The next problem is to implement it using MC. In Statistical mechanics, Fokker-Planck trajectories are generated by means of a Langevin equation (ref. [39]). The Langevin equation corresponding to eq. (6.64) is

$$\frac{\partial \mathbf{r}(t)}{\partial t} = D\mathbf{F}(\mathbf{r}(t)) + \boldsymbol{\eta}, \quad (6.71)$$

³We use that our trial state is real, and that the normalization integral is just a constant that is canceled out in the fraction.

where $\mathbf{r}(t)$ is the position of one particle and $\boldsymbol{\eta}$ is a randomly fluctuating force which is distributed according to a multidimensional Gaussian with a mean of zero and a variance of $2D$. We integrate this equation over a short time interval Δt , obtaining a discretized form suitable for our MC calculations. The particle is moved from an old position to a new through

$$\mathbf{r}_{new} = \mathbf{r}_{old} + D\mathbf{F}(\mathbf{r}_{old})\Delta t + \chi\sqrt{\Delta t}, \quad (6.72)$$

where χ is a Gaussian random variable with a mean value of zero and a variance of $2D$. So moving to a new position requires that we have an algorithm for creating random numbers that follow the distribution in eq. (6.4), with $\sigma^2 = 2D$, and $\nu = 0$.

The Fokker-Planck equation yields a step probability for the Markov chain given by the Green's function (in two dimensions with N particles)

$$G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t) = \frac{1}{(4\pi D \Delta t)^N} e^{-\frac{1}{4D\Delta t}(\boldsymbol{\tau}' - \boldsymbol{\tau} - D\Delta t\mathbf{F}(\boldsymbol{\tau}))^2}. \quad (6.73)$$

This function is derived in refs. [36, 39]. If the reader is in need of a deeper introduction to diffusion equations and Green's functions, we refer to the same references. The diffusion constant will in our calculations be $D = \frac{1}{2}$. This is found from considering the Schrödinger equation with our dimensionless units.

When we previously discussed Markov chains, we split the transition probability up in two like this:

$$W_{ij} = g_{ij}A_{ij}, \quad (6.74)$$

which is the same as saying

$$W(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}') = g(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}')A(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}'), \quad (6.75)$$

where $\boldsymbol{\tau}$ is the old positions and $\boldsymbol{\tau}'$ is the new positions. We defined g as a trial step probability, and A as an acceptance probability. We now want to use the generalized Metropolis method as described above. We identify the trial step probabilities g with the greens function G . We then have

$$W(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}') = G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t)A(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}'), \quad (6.76)$$

and we can rewrite eq. (6.30) as

$$\frac{A(\boldsymbol{\tau} \rightarrow \boldsymbol{\tau}')}{A(\boldsymbol{\tau}' \rightarrow \boldsymbol{\tau})} = \frac{G(\boldsymbol{\tau}', \boldsymbol{\tau}; \Delta t)\rho(\boldsymbol{\tau}')}{G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t)\rho(\boldsymbol{\tau})}. \quad (6.77)$$

We then define the ratio

$$\begin{aligned} R &= \frac{G(\boldsymbol{\tau}', \boldsymbol{\tau}; \Delta t)\rho(\boldsymbol{\tau}')}{G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t)\rho(\boldsymbol{\tau})} \\ &= \frac{G(\boldsymbol{\tau}', \boldsymbol{\tau}; \Delta t)|\Psi_T(\boldsymbol{\tau}')|^2}{G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t)|\Psi_T(\boldsymbol{\tau})|^2}. \end{aligned} \quad (6.78)$$

The fraction involving the greens functions becomes

$$\begin{aligned} \frac{G(\boldsymbol{\tau}', \boldsymbol{\tau}; \Delta t)}{G(\boldsymbol{\tau}, \boldsymbol{\tau}'; \Delta t)} &= \frac{e^{-\frac{1}{2\Delta t}(\boldsymbol{\tau} - \boldsymbol{\tau}' - \frac{\Delta t}{2}\mathbf{F}(\boldsymbol{\tau}'))^2}}{e^{-\frac{1}{2\Delta t}(\boldsymbol{\tau}' - \boldsymbol{\tau} - \frac{\Delta t}{2}\mathbf{F}(\boldsymbol{\tau}))^2}} \\ &= e^{-\frac{1}{2}(\mathbf{F}(\boldsymbol{\tau}) + \mathbf{F}(\boldsymbol{\tau}'))(\frac{\Delta t}{4}(\mathbf{F}(\boldsymbol{\tau}) - \mathbf{F}(\boldsymbol{\tau}')) + (\boldsymbol{\tau} - \boldsymbol{\tau}'))} \end{aligned} \quad (6.79)$$

Table 6.1: A summary of the QVMC strategy.

Put N walkers at random initial positions.
 REPEAT for all MC cycles
 REPEAT for all walkers
 Select next walker, and move that walker to a new position (either by eq. (6.60) or by eq. (6.72)).
 Compute the ratio R (according to either eq. (6.61) or by eq. (6.80)).
 Accept or reject the new position according to eq. (6.81).
 UNTIL we have tried moving all walkers.
 Update expectation values.
 UNTIL all MC cycles is done.

We then rewrite eq. (6.78) as

$$R = \frac{|\Psi_T(\boldsymbol{\tau}')|^2}{|\Psi_T(\boldsymbol{\tau})|^2} e^{-\frac{1}{2}(\mathbf{F}(\boldsymbol{\tau})+\mathbf{F}(\boldsymbol{\tau}'))(\frac{\Delta t}{4}(\mathbf{F}(\boldsymbol{\tau})-\mathbf{F}(\boldsymbol{\tau}'))+(\boldsymbol{\tau}-\boldsymbol{\tau}'))}. \quad (6.80)$$

The acceptance probability is then defined as before

$$A = \begin{cases} R & \text{if } R < 1, \\ 1 & \text{if } R \geq 1. \end{cases} \quad (6.81)$$

This probability is compared with a uniform random number. If the random number is less than A , we accept the step. If not, we reject it.

In Table 6.1 we have summarized the QVMC algorithm. The strategy is equivalent for both simple Metropolis sampling and for generalized Metropolis sampling, but the equations used are different.

6.6 Energy minimization using DFP

We wish to minimize the energy using our QVMC machinery. To do this, we need to find the optimal parameters for our trial wave function, that is, the parameters that minimizes the expectation value of the energy.

The first strategy that comes to mind is to manually vary the parameters and calculate the energy for each set of parameters. However, this is rather brute-force-ish; it does not utilize any knowledge of the trial wave function, and it may take us several runs of the program to actually target the correct parameters.

Another method we will try out, is the Davidon-Fletcher-Powell algorithm (DFP), given in ref. [43]. This method builds on the conjugate gradient method (CGM), and the steepest descent method (both of these also described in ref. [43]).

The algorithm is based on approximating a function of variational parameters, $f(\boldsymbol{\alpha})$, by a quadratic form, where $\boldsymbol{\alpha}$ is a vector of variational parameters. We write f as a Taylor series at around some point \mathbf{P} (we choose \mathbf{P} as the origin of the coordinate system

with coordinates $\boldsymbol{\alpha}$)

$$\begin{aligned} f(\boldsymbol{\alpha}) &= f(\mathbf{P}) + \sum_i \left. \frac{\partial f}{\partial \alpha_i} \right|_{\mathbf{P}} \alpha_i + \frac{1}{2} \sum_{ij} \left. \frac{\partial^2 f}{\partial \alpha_i \partial \alpha_j} \right|_{\mathbf{P}} \alpha_i \alpha_j + \dots \\ &\approx a - \mathbf{b} \cdot \boldsymbol{\alpha} + \frac{1}{2} \boldsymbol{\alpha} \cdot \mathbf{A} \cdot \boldsymbol{\alpha}, \end{aligned} \quad (6.82)$$

where we have defined

$$a \equiv f(\mathbf{P}), \quad \mathbf{b} \equiv \nabla f|_{\mathbf{P}}, \quad \mathbf{A}_{ij} = \left. \frac{\partial^2 f}{\partial \alpha_i \partial \alpha_j} \right|_{\mathbf{P}}. \quad (6.83)$$

The matrix \mathbf{A} is called the Hessian matrix of the function f at \mathbf{P} . In this second order approximation, the gradient of f is easily calculated as

$$\nabla f = \mathbf{A} \cdot \boldsymbol{\alpha} + \mathbf{b}. \quad (6.84)$$

Calculating the Hessian \mathbf{A} is generally hard work. The basic idea the DFP procedure is to build up, iteratively, a good approximation to the inverse Hessian, \mathbf{A}^{-1} . This means we want to construct a sequence of matrices, \mathbf{H}_i , such that

$$\lim_{i \rightarrow \infty} \mathbf{H}_i = \mathbf{A}^{-1}. \quad (6.85)$$

DFP is a variant of a *variable metric method*. Such methods are sometimes called *quasi-Newton methods* as well. Consider finding a minimum by using Newton's method to search for a zero of the gradient of the function. Near the current point $\boldsymbol{\alpha}_i$, we have to second order (ref. [43])

$$f(\boldsymbol{\alpha}) \approx f(\boldsymbol{\alpha}_i) + (\boldsymbol{\alpha} - \boldsymbol{\alpha}_i) \cdot \nabla f(\boldsymbol{\alpha}_i) + \frac{1}{2} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_i) \cdot \mathbf{A} \cdot (\boldsymbol{\alpha} - \boldsymbol{\alpha}_i), \quad (6.86)$$

so

$$\nabla f(\boldsymbol{\alpha}) = \nabla f(\boldsymbol{\alpha}_i) + \mathbf{A} \cdot (\boldsymbol{\alpha} - \boldsymbol{\alpha}_i). \quad (6.87)$$

Using Newton's method, we set $\nabla f(\boldsymbol{\alpha}) = 0$ to find the next iteration point, which yields

$$\boldsymbol{\alpha} - \boldsymbol{\alpha}_i = -\mathbf{A}^{-1} \cdot \nabla f(\boldsymbol{\alpha}_i). \quad (6.88)$$

DFP is called a quasi-Newton method because we do not use the actual Hessian matrix of f , but instead use our current approximation for it. This approach is actually often-times *better* than using the true Hessian, because we can choose \mathbf{H} such that it is *always* positive definite and symmetric. The true Hessian need not always be so, which may lead to moving towards *increasing* values of f , as explained in ref. [43].

In ref. [43] there is only given a heuristic motivation for the algorithm which takes \mathbf{H}_i to \mathbf{H}_{i+1} . The algorithm given for updating the matrix is

$$\begin{aligned} \mathbf{H}_{i+1} = \mathbf{H}_i &+ \frac{(\boldsymbol{\alpha}_{i+1} - \boldsymbol{\alpha}_i) \otimes (\boldsymbol{\alpha}_{i+1} - \boldsymbol{\alpha}_i)}{(\boldsymbol{\alpha}_{i+1} - \boldsymbol{\alpha}_i) \cdot (\nabla f(\boldsymbol{\alpha}_{i+1}) - \nabla f(\boldsymbol{\alpha}_i))} \\ &- \frac{[\mathbf{H}_i \cdot (\nabla f(\boldsymbol{\alpha}_{i+1}) - \nabla f(\boldsymbol{\alpha}_i))] \otimes [\mathbf{H}_i \cdot (\nabla f(\boldsymbol{\alpha}_{i+1}) - \nabla f(\boldsymbol{\alpha}_i))]}{(\nabla f(\boldsymbol{\alpha}_{i+1}) - \nabla f(\boldsymbol{\alpha}_i)) \cdot \mathbf{H}_i \cdot (\nabla f(\boldsymbol{\alpha}_{i+1}) - \nabla f(\boldsymbol{\alpha}_i))}, \end{aligned} \quad (6.89)$$

where \otimes denotes the “outer” or “direct” product of two vectors, which will be a matrix. The ij component of $\mathbf{u} \otimes \mathbf{v}$ is $u_i v_j$. This algorithm is taken from ref. [43].

Minimization of the function f then entails that we must have algorithms that evaluates the function itself, evaluates the gradient of the function, $\nabla f(\boldsymbol{\alpha})$, with respects to the variational parameters. We then choose a matrix \mathbf{H} , and update it according to eq. (6.89). We then use the same procedure as in Newton’s method, but with \mathbf{H} instead of the exact Hessian \mathbf{A} .

6.7 Blocking

In most of the discussion in this chapter we have completely ignored correlations in the MC results. We have approximated the error in the sample mean with

$$\text{err}^2 \approx \frac{1}{n} \text{Var}(x). \quad (6.90)$$

The derivation if this equation is shown in Appendix A, and it assumes that the measurements are uncorrelated. This is in reality a bad approximation. We are using (pseudo-)random-number generators in many parts of our calculations, and as discussed in Section 6.1 there will always be correlations between these random numbers. Even though we have assumed that the random numbers have little correlations, it turns out that we cannot completely ignore the fact that correlations are present. Functions evaluated with respects to correlated variables will also be correlated. This means that each calculated value of the local energy will be correlated to other values. In addition, our algorithm is based on Markov chains to find new trial positions for the random walkers. A Markov chain, as discussed in Section 6.3, contains inherent correlations, because every new Markov step depends on the previous position. This is *not* an independent stochastic process, and we need to account for the correlations.

As discussed in Appendix A, the effect of correlations in a sample is to increase the error of the sample mean. When we take correlations in the sample in to considerations, the statistical error for a set of stochastic values is given by

$$\text{err}^2 \approx \frac{1}{n} \text{Cov}(x) = \frac{\tau}{n} \text{Var}(x), \quad (6.91)$$

with τ as the autocorrelation time, defined as in eq. (A.39). We have that $\tau \geq 1$, where $\tau = 1$ means that the sample is uncorrelated. Calculating τ is rather time consuming, and it is therefore not the best approach for dealing with the correlations. We will instead use the *blocking* technique, described in ref. [44].

The procedure is to group the data into several equal-sized blocks, and then treat the mean of each block as our measurements. This will leave the mean of the sample untouched, but the error will change.

Say we have 10000 measurements. We could then start with an initial block size of 10, this means we have 1000 blocks with 10 measurements in each of the blocks. We treat the mean of each block as the new measurements. The total mean of the sample will then be unchanged. We calculate the error of the sample mean, assuming the new measurements

to be uncorrelated, by eq. (6.90). We then increase the block size, and do a new calculation of the error. This process is repeated. And at some point, we will reach a plateau (we will see this later when we present our results) where the error does not increase any more as the block size increases. The error at this plateau will then be our true estimate for the sample error, meaning the measurements (means of each blocks) are in effect uncorrelated.

6.8 Time-step extrapolation

In the generalized Metropolis method, we use the equation

$$\mathbf{r}_{new} = \mathbf{r}_{old} + D\mathbf{F}(\mathbf{r}_{old})\Delta t + \chi\sqrt{\Delta t}. \quad (6.92)$$

to generate trajectories for the particles. Unfortunately, this equation is only exact at the limit $\Delta t \rightarrow 0$. However, it is not possible to use $\Delta t = 0$ in the calculations, because then the particle would not move at all. This introduces a bias into the dynamics for any $\Delta t > 0$. The distribution of trajectories, and therefore the measured energy, will deviate increasingly from the exact as Δt increases (ref. [39]).

It is possible to extrapolate towards $\Delta t = 0$ by plotting results for non-zero Δt and fitting a curve to the data plot. In this work we will use a simple linear fitting that also accommodates for the standard error of the data. We use a least-squares method called Singular-Value decomposition, described in Section 15.4 in ref. [43].

Chapter 7

The trial wave function

We want to find the ground state energy of an N -electron quantum dot. The Hamiltonian of the system is defined in eq. (5.108). We restate it here as

$$\hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega^2 (x_i^2 + y_i^2) \right) + \sum_{i < j}^N \frac{1}{r_{ij}}. \quad (7.1)$$

As stated before, we simply *cannot* solve this problem analytically. We therefore need to do numerical calculations. Our method of choice is the QVMC method described in the previous chapter. But before we can start any calculations, we need a trial wave function. To get a good approximation of the energy, our trial wave function cannot be just any function. We need to employ our understanding of the system to make a good guess at how the analytical form of the wave function actually looks like.

The most commonly used correlated wave function in many-fermion physics is the Jastrow-Slater wave function (ref. [9]). This is the wave function we will use in our thesis. The initial assumption is that the wave function can be separated into two parts

$$\Psi_T = SJ, \quad (7.2)$$

where S is a Slater determinant part, and J is some correlation function which we will call the Jastrow factor.

In this chapter we will find an analytical expression for our trial wave function. We start out with finding a Slater determinant part that we assume is a good physical description of our system. We will find the single-particle orbitals associated with the Slater determinant, and calculate the gradient and Laplacian of these. We then choose a Jastrow function, and discuss the cusp conditions this function must satisfy. At the end we will give the total trial wave function for our system.

7.1 Slater determinants

We discussed Slater determinants in Chapter 3. The result was that a Slater determinant is an exact solution of the non-interacting system, and that the exact solution to the interacting system is a linear expansion of Slater determinants. However, this expansion is infinite, and therefore we cannot do it in computer experiments. In this chapter, and the rest of the thesis, we use the notation $|A|$ for determinants. The matrix associated

with the determinant is then written as A .

A general Slater determinant is written

$$|S\rangle = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{\nu_1}(\mathbf{x}_1) & \cdots & \phi_{\nu_1}(\mathbf{x}_N) \\ \vdots & & \vdots \\ \phi_{\nu_N}(\mathbf{x}_1) & \cdots & \phi_{\nu_N}(\mathbf{x}_N) \end{vmatrix}, \quad (7.3)$$

where the functions ϕ_ν are single-particle orbitals, and $\mathbf{x} = (\mathbf{r}, s)$ as before contains both the spatial and spin degree of freedom. We restate eq. (3.41), which says that any general many-body wave function can be written as a linear combination of Slater determinants

$$\Psi(\mathbf{x}_1, \dots, \mathbf{x}_N) = \sum_{\nu_1 \nu_2 \dots \nu_M} C_{\nu_1 \nu_2 \dots \nu_M} \Phi_{\nu_1 \nu_2 \dots \nu_M}(\mathbf{x}_1, \dots, \mathbf{x}_N), \quad (7.4)$$

where M is the number of single-particle orbitals, which is generally infinite. Unfortunately, we cannot solve an infinite dimensional problem, so we have to cut this sum at some point. This is why we need to include some extra correlation function to our trial wave function. If we could do the infinite expansion, we would not have to.

We now define the determinant part of our trial wave function as a limited sum

$$|S'\rangle = \sum_i^c C_i |S_i\rangle, \quad (7.5)$$

where $|S_i\rangle$ represents Slater determinants built up of different sets of single-particle orbitals, $\{C_i\}$ is a set of expansion coefficients, and c is some cutoff that we have defined. To do QVMC, we want to include some variational parameter, α , to each Slater determinant, i. e.

$$|S_i\rangle \rightarrow |S_i(\alpha_i)\rangle. \quad (7.6)$$

Our trial Slater determinant can then be written as

$$|S'(\{\alpha_i\})\rangle = \sum_i C_i |S_i(\alpha_i)\rangle. \quad (7.7)$$

Assume that

$$\{\phi_{\sigma_i}\}, \quad 0 \leq i < N, \quad (7.8)$$

is the N single-particle orbitals with lowest energy, where N is the number of particles in the system. We then define the ground state of the non-interacting system S_0 as the Slater determinant that is built up of these orbitals. We rewrite eq. (7.7) as

$$|S'(\{\alpha_i\})\rangle = C_0 |S_0(\alpha_0)\rangle + \sum_{i=1}^c C_i |S_i(\alpha_i)\rangle. \quad (7.9)$$

The Slater determinant S_0 is as stated the ground state of the non-interacting system. We then interpret the other Slater determinants as including correlation effects to the system. They do this by introducing single-particle state of higher energy, and then in a sense allowing the particles to excite to higher energy levels.

In our calculations, we will actually only use the ground-state Slater determinant, thereby setting

$$|S'(\alpha)| = |S_0(\alpha)|. \quad (7.10)$$

This is because we consider a closed shell system. We know that there is a high energy difference between the highest energy state in one shell and the lowest energy state in the next shell. We make the assumption that it is very hard to excite a particle from the outer-most filled shell, and therefore Slater determinants that include orbitals with even higher energy are very unlikely. This is the same as saying that all the expansion coefficients in eq. (7.9) are approximately zero, except for C_0 , which is approximately 1. Our Slater determinant does therefore not include any correlation effects, making the choice of the Jastrow function very important.

7.1.1 Splitting up our Slater determinant

When we later do our actual MC simulation, we will move one particle at the time and evaluate the PDF at the new position. The Slater determinant can be quite tough to compute, but luckily there are some tricks we can do to make our computations faster.

Our Hamiltonian is spin independent. There are therefore two and two single-particle orbitals that are just the same. It is shown in ref. [9] that when we are calculating the expectation values of operators that do not depend on spin, we get equivalent results if we do the substitution

$$|S| = \frac{1}{(N/2)!} |D^\uparrow| |D^\downarrow|, \quad (7.11)$$

with

$$|D^\uparrow| = \begin{vmatrix} \phi_{\nu_1\uparrow}(\mathbf{r}_1) & \cdots & \phi_{\nu_1\uparrow}(\mathbf{r}_{\frac{N}{2}}) \\ \vdots & & \vdots \\ \phi_{\nu_{\frac{N}{2}}\uparrow}(\mathbf{r}_1) & \cdots & \phi_{\nu_{\frac{N}{2}}\uparrow}(\mathbf{r}_{\frac{N}{2}}) \end{vmatrix} \quad (7.12)$$

$$|D^\downarrow| = \begin{vmatrix} \phi_{\nu_1\downarrow}(\mathbf{r}_{(\frac{N}{2}+1)}) & \cdots & \phi_{\nu_1\downarrow}(\mathbf{r}_N) \\ \vdots & & \vdots \\ \phi_{\nu_{\frac{N}{2}}\downarrow}(\mathbf{r}_{(\frac{N}{2}+1)}) & \cdots & \phi_{\nu_{\frac{N}{2}}\downarrow}(\mathbf{r}_N) \end{vmatrix}. \quad (7.13)$$

With the above substitution, even local values of any observables are unchanged, which is very important for us when we calculate the Local energy at specific points in space. To do the above substitution, we have assumed that we have an even number of particles, with an even number of spin ups and spin downs. This will always be so in a closed-shell system. The strength of this substitution is that when we move one particle, only one of the determinants is changed, and we only have to evaluate this determinant, since we have stored the value of the other one. This reduces greatly the computation time.

However, if we write our wave function as a separated spin-up and spin-down determinant, it will no longer be antisymmetric under particle exchange, as it should be. Even so, the eigenvalues are unchanged since \hat{H} is spin independent. Because we work with ratios of the wave function in this thesis, we neglect the normalization factors from now on. These will just cancel out in the total result anyway.

7.1.2 Single-particle orbitals

We now need to define the single-particle orbitals that make up our Slater determinant. The single-particle orbitals should be the solutions to the one-particle problem associated with the Hamiltonian in eq. (7.1)

$$\hat{h} = -\frac{1}{2}\nabla^2 + \frac{1}{2}\omega^2(x^2 + y^2). \quad (7.14)$$

We found the solutions to this problem in Section 5.1, but then in polar coordinates. In the code we have developed, we use Cartesian coordinates. By examining the one-particle problem above, we see that it is just the ordinary two-dimensional harmonic oscillator equation. The solutions to this problem can be found in most standard text books on quantum mechanics (see f. ex. refs. [1, 2]). We write them as

$$\phi_{n_x, n_y}(x, y) = AH_{n_x}(\sqrt{\omega}x)H_{n_y}(\sqrt{\omega}y)e^{-\frac{\omega}{2}(x^2+y^2)}, \quad (7.15)$$

where A is some normalization constant, and H_{n_x} and H_{n_y} are so-called Hermite polynomials (see refs. [1, 2]). It is easy to confuse the Hermite polynomials with the Hamiltonian. The way to distinguish them is to remember that the Hermite polynomials always have a subscript n , and the Hamiltonian is usually written in operator form with a hat. The single-particle energies are now given as (remember, we use the scaled Hamiltonian, and the oscillator frequency ω is the same as defined in eq. (5.107))

$$E_{n_x, n_y} = (n_x + n_y + 1)\omega. \quad (7.16)$$

with

$$n_x = 0, 1, 2, \dots \quad (7.17)$$

$$n_y = 0, 1, 2, \dots \quad (7.18)$$

This expression for the energy is a bit different than what we found in eq. (5.44), but remember that our Slater determinant represents a closed-shell system. We have therefore filled up all single-particle orbitals up to a given shell. The energy of the the Slater determinant is given in eq. (3.37), it is simply the sum of all the single-particle energies in the determinant. Since we deal with a closed-shell system, the spin and projection of the orbital momentum are zero. Defining the single-particle energy as eq. (7.15) therefore yields the same total energy for the Slater determinant.

We want to introduce some variational parameter to the Slater determinant. We do this by introducing it in the single-particle orbitals. We then write the orbitals, now omitting the normalization constant¹

$$\phi_{n_x, n_y}(x, y; \alpha) = H_{n_x}(\sqrt{\omega\alpha}x)H_{n_y}(\sqrt{\omega\alpha}y)e^{-\frac{\omega\alpha}{2}(x^2+y^2)}. \quad (7.19)$$

7.1.3 Derivatives of the single-particle orbitals

In the MC calculations, we need the gradient and Laplacian of the wave function. We first state an important property of the Hermite polynomials

$$H'_n(x) = 2nH_{n-1}(x), \quad (7.20)$$

¹We can omit this because all orbitals in the same Slater determinant have the same normalization factor. It will then vanish when we compute ratios of the determinants.

we then have

$$\frac{\partial H_{n_x}(\sqrt{\omega\alpha}x)}{\partial x} = \frac{\partial H_{n_x}(\sqrt{\omega\alpha}x)}{\partial(\sqrt{\omega\alpha}x)} \frac{\partial(\sqrt{\omega\alpha}x)}{\partial x} \quad (7.21)$$

$$= \sqrt{\omega\alpha} H'_{n_x}(\sqrt{\omega\alpha}x) \quad (7.22)$$

$$= 2n_x \sqrt{\omega\alpha} H_{n_x-1}(\sqrt{\omega\alpha}x), \quad (7.23)$$

and the same applies for $H_{n_y}(\sqrt{\omega\alpha}y)$. We also find

$$\frac{\partial}{\partial x} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} = \omega\alpha x e^{-\frac{\omega\alpha}{2}(x^2+y^2)}. \quad (7.24)$$

For the sake of brevity, we set $H_{n_x}(\sqrt{\omega\alpha}x) = H_{n_x}$, and $H_{n_y}(\sqrt{\omega\alpha}y) = H_{n_y}$. The gradient of the single-particle orbitals is then

$$\nabla \phi_{n_x, n_y}(x, y; \alpha) = \left(\mathbf{i} \frac{\partial}{\partial x} + \mathbf{j} \frac{\partial}{\partial y} \right) H_{n_x} H_{n_y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} \quad (7.25)$$

$$\begin{aligned} &= \mathbf{i} \left(\frac{\partial H_{n_x}}{\partial x} H_{n_y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} + H_{n_x} H_{n_y} \frac{\partial}{\partial x} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} \right) \\ &+ \mathbf{j} \left(H_{n_x} \frac{\partial H_{n_y}}{\partial y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} + H_{n_x} H_{n_y} \frac{\partial}{\partial y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} \right). \end{aligned} \quad (7.26)$$

We use eq. (7.23) and eq. (7.24) in the last equation, which yields

$$\begin{aligned} \nabla \phi_{n_x, n_y}(x, y; \alpha) &= \mathbf{i} \left(2n_x \sqrt{\omega\alpha} \frac{H_{n_x-1}}{H_{n_x}} - x\omega\alpha \right) H_{n_x} H_{n_y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} \\ &+ \mathbf{j} \left(2n_y \sqrt{\omega\alpha} \frac{H_{n_y-1}}{H_{n_y}} - y\omega\alpha \right) H_{n_x} H_{n_y} e^{-\frac{\omega\alpha}{2}(x^2+y^2)} \end{aligned} \quad (7.27)$$

$$\begin{aligned} &= \left(\left(2n_x \sqrt{\omega\alpha} \frac{H_{n_x-1}}{H_{n_x}} - x\omega\alpha \right) \mathbf{i} \right. \\ &\left. + \left(2n_y \sqrt{\omega\alpha} \frac{H_{n_y-1}}{H_{n_y}} - y\omega\alpha \right) \mathbf{j} \right) \phi_{n_x, n_y}(x, y; \alpha). \end{aligned} \quad (7.28)$$

We also need to calculate the Laplacian. It is

$$\nabla^2 \phi_{n_x, n_y}(x, y; \alpha) = \nabla \cdot \nabla \phi_{n_x, n_y}(x, y; \alpha) \quad (7.29)$$

$$\begin{aligned} &= \frac{\partial}{\partial x} \left(\left(2n_x \sqrt{\omega \alpha} \frac{H_{n_x-1}}{H_{n_x}} - x\omega \alpha \right) \phi_{n_x, n_y}(x, y; \alpha) \right) \\ &+ \frac{\partial}{\partial y} \left(\left(2n_y \sqrt{\omega \alpha} \frac{H_{n_y-1}}{H_{n_y}} - y\omega \alpha \right) \phi_{n_x, n_y}(x, y; \alpha) \right) \end{aligned} \quad (7.30)$$

$$\begin{aligned} &= \left(\frac{\partial}{\partial x} \left(2n_x \sqrt{\omega \alpha} \frac{H_{n_x-1}}{H_{n_x}} - x\omega \alpha \right) \right. \\ &+ \left(2n_x \sqrt{\omega \alpha} \frac{H_{n_x-1}}{H_{n_x}} - x\omega \alpha \right)^2 \\ &+ \frac{\partial}{\partial y} \left(2n_y \sqrt{\omega \alpha} \frac{H_{n_y-1}}{H_{n_y}} - y\omega \alpha \right) \\ &+ \left. \left(2n_y \sqrt{\omega \alpha} \frac{H_{n_y-1}}{H_{n_y}} - y\omega \alpha \right)^2 \right) \phi_{n_x, n_y}(x, y; \alpha) \end{aligned} \quad (7.31)$$

$$\begin{aligned} &= \omega \alpha \left(4n_x(n_x - 1) \frac{H_{n_x-2}}{H_{n_x}} + 4n_y(n_y - 1) \frac{H_{n_y-2}}{H_{n_y}} + \omega \alpha (x^2 + y^2) \right. \\ &\left. - 4n_x x \frac{H_{n_x-1}}{H_{n_x}} - 4n_y y \frac{H_{n_y-1}}{H_{n_y}} - 2 \right) \phi_{n_x, n_y}(x, y; \alpha). \end{aligned} \quad (7.32)$$

7.2 The Jastrow function

In Chapter 3 we discussed some of the basic properties a fermionic wave function must have. The Jastrow function needs to emulate the same behaviour. The total wave function must be antisymmetric, and we have assumed $\Psi_T = |S|J$, where $|S|$ is antisymmetric. The Jastrow function must therefore be symmetric. We also want J to only affect the wave function when the distance between particles is reasonably small. The most commonly used wave function is the Pade-Jastrow function (ref. [39])

$$J = \prod_{i < j}^N e^{g(r_{ij})} \quad (7.33)$$

$$g(r_{ij}) = \frac{\sum_{k=1}^n a_{ij} r_{ij}^k}{1 + \sum_{k=1}^n \beta_k r_{ij}^k}, \quad (7.34)$$

where r_{ij} is the distance between electron i and electron j , a_{ij} is a factor that is dependent on the spin configuration of the particles, and β_k is a variational parameter. When the distance r_{ij} becomes small, only the lower order terms in the expansion above is relevant, and one usually use only $n = 1$, that is

$$g(r_{ij}) = \frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}, \quad (7.35)$$

which is the analytical form we will use throughout the thesis. Our Jastrow function is then only

$$J = \prod_{i < j} e^{\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}}. \quad (7.36)$$

This function follows the criteria we set in the beginning of the section. In addition it satisfies important cusp conditions that must be followed.

7.2.1 Cusp conditions

We define the total unnormalized trial wave function as

$$\Psi_T = |D^\uparrow| |D^\downarrow| J. \quad (7.37)$$

The Hamiltonian in eq. (7.1) has a singularity when $r_{ij} \rightarrow 0$. Since, for any exact eigenstate, $\hat{H}\Psi$ is proportional to Ψ , this divergence must be cancelled by an equal and opposite divergence in the kinetic energy terms. It is important that this is included in our trial wave function, else the local energy will diverge as r_{ij} approaches zero. The divergency cannot be cancelled by the kinetic energy of the Slater determinant part (ref. [39]), so the cancellation must come from the Jastrow factor.

We consider two electrons, k and l , in close approach, and rewrite the Hamiltonian in terms of their center-of-mass coordinates, that is

$$\mathbf{r}_{kl} = \mathbf{r}_l - \mathbf{r}_k \quad \text{and} \quad \mathbf{R} = \frac{m_k \mathbf{r}_k + m_l \mathbf{r}_l}{m_k + m_l} = \frac{\mathbf{r}_k + \mathbf{r}_l}{2} \quad (7.38)$$

$$\mathbf{r}_k = \mathbf{R} - \frac{\mathbf{r}_{kl}}{2} \quad \text{and} \quad \mathbf{r}_l = \mathbf{R} + \frac{\mathbf{r}_{kl}}{2}, \quad (7.39)$$

where we have used that the particles have equal mass. The gradients in the new coordinates transforms as (ref. [45])

$$\nabla_{\mathbf{r}_{kl}} = \frac{1}{2} \nabla_l - \frac{1}{2} \nabla_k \quad (7.40)$$

$$\nabla_{\mathbf{R}} = \nabla_k + \nabla_l, \quad (7.41)$$

which gives

$$\frac{1}{2} \nabla_k^2 + \frac{1}{2} \nabla_l^2 = \nabla_{\mathbf{r}_{kl}}^2 + \frac{1}{4} \nabla_{\mathbf{R}}^2. \quad (7.42)$$

With this, the Hamiltonian in eq. (7.1) takes the form

$$\hat{H} = - \left(\nabla_{\mathbf{r}_{kl}}^2 + \frac{1}{4} \nabla_{\mathbf{R}}^2 - \frac{1}{r_{kl}} \right) - \sum_{i \neq k, l}^N \frac{1}{2} \nabla_i^2 + \sum_i^N \frac{1}{2} \omega^2 (x_i^2 + y_i^2) + \sum_{(i < j) \neq k, l}^N \frac{1}{r_{ij}}. \quad (7.43)$$

The oscillator energies are finite, as is the kinetic energy of the center-of-mass motion. The Coulomb term $1/r_{kl}$, however, diverges as k approaches l . We must have that the kinetic energy term $-\nabla_{\mathbf{r}_{kl}}^2$ term cancels the Coulomb divergence. We consider the local energy with the these two terms in the limit $r_{kl} \rightarrow 0$

$$\lim_{r_{kl} \rightarrow 0} \left\{ \frac{1}{|D^\uparrow| |D^\downarrow| J} \left(-\nabla_{\mathbf{r}_{kl}}^2 + \frac{1}{r_{kl}} \right) |D^\uparrow| |D^\downarrow| J \right\} < \infty, \quad (7.44)$$

with $r_{kl} = |\mathbf{r}_{kl}|$. We assume that we have a Pade-Jastrow function on the form of eq. (7.35). Since we now consider only one pair of particles (the function for all other pairs of particles will then disappear in the fraction in the expression for the local energy), we write the Jastrow as

$$J = e^{g(r_{kl})} \prod_{(i<j)\neq k,l} e^{g(r_{ij})}, \quad (7.45)$$

with the function g as before. The Laplacian in two dimensions is

$$\nabla_{\mathbf{r}}^2 = \frac{1}{r} \frac{\partial}{\partial r} + \frac{\partial^2}{\partial r^2}. \quad (7.46)$$

We then have

$$\nabla_{\mathbf{r}_{kl}}^2 |D^\uparrow||D^\downarrow| J = \left(\nabla_{\mathbf{r}_{kl}}^2 |D^\uparrow||D^\downarrow| \right) J + 2 \left(\nabla_{\mathbf{r}_{kl}} |D^\uparrow||D^\downarrow| \right) \left(\nabla_{\mathbf{r}_{kl}} J \right) + |D^\uparrow||D^\downarrow| \left(\nabla^2 J \right). \quad (7.47)$$

The derivatives of the Jastrow is

$$\begin{aligned} \nabla_{\mathbf{r}} J &= r_{kl} \hat{r}_{kl} \frac{\partial}{\partial r_{kl}} \left(e^{g(r_{kl})} \prod_{(i<j)\neq k,l} e^{g(r_{ij})} \right) \\ &= g'(r_{kl}) J, \end{aligned} \quad (7.48)$$

and

$$\begin{aligned} \nabla_{\mathbf{r}}^2 J &= \left(\frac{1}{r} \frac{\partial}{\partial r_{kl}} + \frac{\partial^2}{\partial r_{kl}^2} \right) \left(e^{g(r_{kl})} \prod_{(i<j)\neq k,l} e^{g(r_{ij})} \right) \\ &= \left(\frac{g'(r_{kl})}{r_{kl}} + g''(r_{kl}) + g'(r_{kl})^2 \right) J, \end{aligned} \quad (7.49)$$

where \hat{r}_{kl} is a unit vector in the \mathbf{r}_{kl} direction. We insert eq. (7.47), eq. (7.48) and eq. (7.49) back into eq. (7.44)

$$\lim_{r_{kl} \rightarrow 0} \left\{ -\frac{\nabla_{\mathbf{r}_{kl}}^2 |D^\uparrow||D^\downarrow|}{|D^\uparrow||D^\downarrow|} - \frac{2g' r_{kl} \cdot (\nabla_{\mathbf{r}_{kl}} |D^\uparrow||D^\downarrow|)}{|D^\uparrow||D^\downarrow|} - \left(\frac{g'}{r_{kl}} + g'' + g'^2 - \frac{1}{r_{kl}} \right) \right\} < \infty. \quad (7.50)$$

This equation indicates different short-range behavior for anti-parallel spin than for parallel spin. For electrons with anti-parallel spin in contact, in general $|D^\uparrow||D^\downarrow| \neq 0$, but for electrons with parallel spin in contact, two rows are the same in one of the determinants, and we must have that $|D^\uparrow||D^\downarrow| = 0$. We consider the two cases separately.

1. We first consider anti-parallel spin. With anti-parallel spin, all the terms that involve a derivative of the determinants will be finite. We are left with only two divergent terms in eq. (7.50), these are the two terms proportional with $1/r_{kl}$. We find that we must have

$$\lim_{r_{kl} \rightarrow 0} \left\{ \frac{g' + 1}{r_{kl}} \right\} < \infty. \quad (7.51)$$

To have this we demand that

$$\left. \frac{\partial g}{\partial r_{kl}} \right|_{r_{kl}=0} - 1 = 0, \quad (7.52)$$

which gives the cusp condition

$$\left. \frac{\partial g}{\partial r_{kl}} \right|_{r_{kl}=0} = 1 \quad (7.53)$$

2. For parallel spins we expand $|D^\uparrow||D^\downarrow|$ around $\mathbf{r}_{kl} = 0$. We get

$$|D^\uparrow||D^\downarrow| \approx \mathbf{b} \cdot \mathbf{r}_{kl} + \dots, \quad (7.54)$$

where \mathbf{b} is some undetermined vector. The derivatives are

$$\left(\nabla_{\mathbf{r}_{kl}} |D^\uparrow||D^\downarrow| \right) \Big|_{r_{kl} \rightarrow 0} \approx \mathbf{b} \quad (7.55)$$

$$\left(\nabla_{\mathbf{r}_{kl}}^2 |D^\uparrow||D^\downarrow| \right) \Big|_{r_{kl} \rightarrow 0} \approx 0 \quad (7.56)$$

We insert this into eq. (7.50)

$$\lim_{r_{kl} \rightarrow 0} \left\{ -\frac{2g'(\hat{r}_{kl} \cdot \mathbf{b})}{r_{kl}(\hat{r}_{kl} \cdot \mathbf{b})} - \left(\frac{g'}{r_{kl}} + g'' + g'^2 - \frac{1}{r_{kl}} \right) \right\} < \infty. \quad (7.57)$$

Collecting all the divergent terms, this yields

$$\lim_{r_{kl} \rightarrow 0} \left\{ \frac{3g' + 1}{r_{kl}} \right\} < \infty. \quad (7.58)$$

We then demand that

$$3 \frac{\partial g}{\partial r_{kl}} \Big|_{r_{kl}=0} - 1 = 0, \quad (7.59)$$

which yields

$$\frac{\partial g}{\partial r_{kl}} \Big|_{r_{kl}=0} = \frac{1}{3}. \quad (7.60)$$

We want to use the Pade-Jastrow function. To find what the cusp conditions impose on the function we need to find the derivative of g :

$$g'(r_{kl}) = \frac{a_{kl}}{(1 + \beta r_{kl})^2} \quad (7.61)$$

$$g'(0) = a_{kl}. \quad (7.62)$$

To satisfy the cusp condition we demand that

$$a_{kl} = \begin{cases} 1 & \text{when particles } k \text{ and } l \text{ have anti-parallel spin,} \\ \frac{1}{3} & \text{when particles } k \text{ and } l \text{ have parallel spin.} \end{cases} \quad (7.63)$$

7.3 The total trial wave function

We have decided on a trial wave function that obeys much of the physics one would expect from the correct wave function of the system. Our trial wave function has the form

$$\Psi_T(\alpha, \beta) = |D^\uparrow(\alpha)||D^\downarrow(\alpha)|J(\beta), \quad (7.64)$$

where α and β are two variational parameters which we have imposed on the function. The determinantal part of our function is split into a spin-up and spin-down part. This actually destroys the antisymmetric properties of the wave function, but as we have discussed, it still gives us the correct expectation values. We have chosen a Pade-Jastrow

function as the correlation function, and found the cusp conditions this function has to satisfy. The total wave function can then be written as

$$\Psi_T = \begin{vmatrix} \phi_{\nu_1 \uparrow}(\mathbf{r}_1, \alpha) & \cdots & \phi_{\nu_1 \uparrow}(\mathbf{r}_{\frac{N}{2}}, \alpha) \\ \vdots & & \vdots \\ \phi_{\nu_{\frac{N}{2}} \uparrow}(\mathbf{r}_1, \alpha) & \cdots & \phi_{\nu_{\frac{N}{2}} \uparrow}(\mathbf{r}_{\frac{N}{2}}, \alpha) \end{vmatrix} \times \begin{vmatrix} \phi_{\nu_1 \downarrow}(\mathbf{r}_{\frac{N}{2}+1}, \alpha) & \cdots & \phi_{\nu_1 \downarrow}(\mathbf{r}_N, \alpha) \\ \vdots & & \vdots \\ \phi_{\nu_{\frac{N}{2}} \downarrow}(\mathbf{r}_{\frac{N}{2}+1}, \alpha) & \cdots & \phi_{\nu_{\frac{N}{2}} \downarrow}(\mathbf{r}_N, \alpha) \end{vmatrix} \prod_{i < j} e^{\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}}, \quad (7.65)$$

where \mathbf{r}_i represents the coordinates of particle $1 \leq i \leq N$. The spin dependent factor in the Jastrow factor is defined as

$$a_{ij} = \begin{cases} 1 & \text{when particles } i \text{ and } j \text{ have anti-parallel spin,} \\ \frac{1}{3} & \text{when particles } i \text{ and } j \text{ have parallel spin,} \end{cases} \quad (7.66)$$

in order to satisfy the cusp condition.

Part II

Implementation and results

Chapter 8

Implementation of the QVMC method

We implemented the QVMC method by writing a code in C++. This language was chosen because of its efficiency and its opportunity for object-orientation. The first step was to build a brute-force code to handle the two-particle quantum dot. By brute force we mean that we used simple Metropolis sampling (see Section 6.5.2). The second step was to implement generalized Metropolis sampling for the two-electron quantum dot, in order to do a more efficient sampling. The next step was to move on to more particles. We then needed efficient algorithms to calculate the Slater determinant and Jastrow factor, and their respective derivatives. Since we had an analytical form of the trial wave function, we used analytical derivatives instead of numerical. To make the code more general and adaptive, it has been written in an object-oriented manner. The code can then easily be used to do QVMC calculations on closed-shell atoms as well, simply by changing the single-particle orbitals and the confining potential. The last step was to optimize the code in order to make the calculations as fast as possible. The trial wave function used in the code is the same as before

$$\Psi_T(\alpha, \beta) = |D^\uparrow(\alpha)\rangle |D^\downarrow(\alpha)\rangle J(\beta). \quad (8.1)$$

We start this chapter by giving a description of how we optimize the important calculations in our program. We then give a description of the classes in the program and their function. We will then describe how the QVMC algorithm has been implemented. We only describe the final QVMC program, not the intermediate ones. This means we only discuss the generalized Metropolis method, described in Section 6.5.3. We will also give a short description as to how we implemented the Davidon-Fletcher-Powell (DFP) algorithm, the blocking technique and time-step extrapolation. Finally, to make our program more efficient, we have parallelized it. This is shortly discussed at the end of the chapter.

8.1 Optimizing the calculations

If we just raw-coded all the algorithms, the calculations would take a lot of time. We want our calculations to be as fast as possible. There are several measures we can take to achieve this.

Calculating the determinantal part of the wave function scales as N^4 , while the Jastrow part scales as N^3 (ref. [38]). Clearly the Slater determinant scales faster when we simulate systems with a large number of particles. This means that optimization of our code is important in order for us to be able to simulate systems with a large number of electrons.

An important observation is that by moving only one electron, we actually only change *one* column in only *one* of the two determinants in eq. (8.1). This is why we split up the determinant in the previous chapter. The wave function is needed in three cases:

1. Calculating the acceptance probabilities in the Metropolis-Hastings algorithm

$$A = \begin{cases} R & \text{if } R < 1 \\ 1 & \text{if } R \geq 1, \end{cases} \quad (8.2)$$

with R as in eq. (6.80). We concentrate on the part of this ratio that involves only the wave function. We write this as

$$R'^2 = R_{D\uparrow}^2 \cdot R_{D\downarrow}^2 \cdot R_J^2, \quad (8.3)$$

where we have defined

$$R_{D\uparrow} = \frac{|D_{new}^\uparrow|}{|D_{old}^\uparrow|} \quad (8.4)$$

$$R_{D\downarrow} = \frac{|D_{new}^\downarrow|}{|D_{old}^\downarrow|} \quad (8.5)$$

$$R_J = \frac{J_{new}}{J_{old}}. \quad (8.6)$$

The critical observation is that by moving only *one* particle, only *one* of the determinants is changed. This means that one of the ratios $R_{D\uparrow}$, or $R_{D\downarrow}$ is equal to 1. We can therefore write eq. (8.3) as

$$R'^2 = R_D^2 \cdot R_J^2, \quad (8.7)$$

where R_D is

$$R_D = \frac{|D_{new}|}{|D_{old}|}, \quad (8.8)$$

which is the determinantal ratio that is not equal to one. This ratio is also needed when updating the inverse-Slater matrix, as discussed in Section 8.2.2.

2. Calculating the quantum force

$$\mathbf{F}_i = 2 \frac{\nabla_i \Psi_T}{\Psi_T}, \quad (8.9)$$

where the subscript i means differentiation with respects to particle number i . We want to split this up in a Slater part and a Jastrow part. The above equation can be written as

$$\begin{aligned} \mathbf{F}_i &= 2 \frac{\nabla_i (|D^\uparrow| |D^\downarrow| J)}{|D^\uparrow| |D^\downarrow| J} \\ &= 2 \left(\frac{\nabla_i |D^\uparrow|}{|D^\uparrow|} + \frac{\nabla_i |D^\downarrow|}{|D^\downarrow|} + \frac{\nabla_i J}{J} \right). \end{aligned} \quad (8.10)$$

Since we differentiate with respects to particle i , the gradient of one of the determinants will be zero. We then rewrite the above equation to

$$\mathbf{F}_i = 2 \left(\frac{\nabla_i |D|}{|D|} + \frac{\nabla_i J}{J} \right), \quad (8.11)$$

where we have suppressed the spin configuration of the determinant. This means we need algorithms to compute the fractions

$$G_D \equiv \frac{\nabla_i |D|}{|D|} \quad (8.12)$$

$$G_J \equiv \frac{\nabla_i J}{J}. \quad (8.13)$$

3. Calculating the Local energy

$$E_L = \sum_i \frac{1}{\Psi_T} \nabla_i^2 \Psi_T - V, \quad (8.14)$$

with the potential V as the sum of the harmonic oscillator potential and the Coulomb energy associated with the electron-electron repulsion. The potential energy bit is easy to compute, but we need an efficient algorithm to compute the Laplacian

$$\begin{aligned} \frac{\nabla_i^2 \Psi_T}{\Psi_T} &= \frac{\nabla_i^2 (|D^\uparrow| |D^\downarrow| J)}{|D^\uparrow| |D^\downarrow| J} \\ &= \frac{\nabla_i^2 |D^\uparrow|}{|D^\uparrow|} + \frac{\nabla_i^2 |D^\downarrow|}{|D^\downarrow|} + \frac{\nabla_i^2 J}{J} + 2 \frac{(\nabla_i |D^\uparrow|) (\nabla_i |D^\downarrow|)}{|D^\uparrow| |D^\downarrow|} \\ &\quad + 2 \frac{(\nabla_i |D^\uparrow|) (\nabla_i J)}{|D^\uparrow| J} + 2 \frac{(\nabla_i |D^\downarrow|) (\nabla_i J)}{|D^\downarrow| J}. \end{aligned} \quad (8.15)$$

Once again, only one of the determinants contains the particle i . This means that the gradient/Laplacian of the other determinant is zero. The above equation then reduces to

$$\frac{\nabla_i^2 \Psi_T}{\Psi_T} = \frac{\nabla_i^2 |D|}{|D|} + \frac{\nabla_i^2 J}{J} + 2 \frac{\nabla_i |D|}{|D|} \frac{\nabla_i J}{J}. \quad (8.16)$$

This means we need an algorithm to compute the fractions

$$L_D = \frac{\nabla_i^2 |D|}{|D|} \quad (8.17)$$

$$L_J = \frac{\nabla_i^2 J}{J}, \quad (8.18)$$

in addition to the fractions in eq. (8.12) and eq. (8.13).

In the following we will describe the algorithms used to calculate the various fractions.

8.1.1 The R_D -ratio

The inverse of a matrix is related to the determinant of the matrix through (see ref. [3])

$$D^{-1} = \frac{adj(D)}{|D|}, \quad (8.19)$$

where $\text{adj}D$ is the *adjugate* of a matrix. The adjugate of a matrix D is the transpose of the cofactor matrix of D (see ref. [3]). We can rewrite eq. (8.19) on matrix-element form

$$D_{ij}^{-1} = \frac{C_{ji}}{|D|}, \quad (8.20)$$

where C_{ji} is the (i, j) -cofactor of D (see ref. [3]). For the matrix D to be invertible, we must have that $D^{-1}D = I$, with I as the identity matrix. This can be written as

$$\sum_{k=1}^N D_{ik} D_{kj}^{-1} = \delta_{ij}, \quad (8.21)$$

where δ_{ij} is the delta function. We multiply eq. (8.20) from the left with D_{ji} , and sum over i , yielding

$$\sum_{i=1}^N D_{ji} D_{ij}^{-1} = \sum_{i=1}^N \frac{D_{ji} C_{ji}}{|D|}. \quad (8.22)$$

This results in

$$I = \sum_{i=1}^N \frac{D_{ji} C_{ji}}{|D|}, \quad (8.23)$$

or

$$|D| = \sum_{i=1}^N D_{ji} C_{ji}. \quad (8.24)$$

Using eq. (8.24), we express the ratio in eq. (8.8) as

$$R_D = \frac{\sum_{j=1}^N D_{ji}^{\text{new}} C_{ji}^{\text{new}}}{\sum_{j=1}^N D_{ji}^{\text{old}} C_{ji}^{\text{old}}}. \quad (8.25)$$

When moving only one particle, lets say particle j , only the j -th row of the matrix changes. This means that the j -th row of the cofactor matrix will remain unchanged (in fact, only the j -th column is unchanged). This means we set

$$C_{ji}^{\text{new}} = C_{ji}^{\text{old}}. \quad (8.26)$$

Equation (8.25) then becomes

$$R_D = \frac{\sum_{j=1}^N D_{ji}^{\text{new}} C_{ji}^{\text{old}}}{\sum_{j=1}^N D_{ji}^{\text{old}} C_{ji}^{\text{old}}}. \quad (8.27)$$

Using eq. (8.20), we rewrite this as

$$R_D = \frac{\sum_{j=1}^N D_{ji}^{\text{new}} (D_{ij}^{\text{old}})^{-1}}{\sum_{j=1}^N D_{ji}^{\text{old}} (D_{ij}^{\text{old}})^{-1}} \quad (8.28)$$

Using eq. (8.21), the sum in the denominator is equal to one. We then have

$$R_D = \sum_i^N D_{ji}^{new} (D_{ij}^{old})^{-1} \quad (8.29)$$

$$= \sum_i^N \phi_i(r_j^{new}) (D_{ij}^{old})^{-1}, \quad (8.30)$$

where we used that the matrix D has the single-particle orbitals $\phi_j(\mathbf{r}_i)$ as entries (where j contains the quantum numbers of the state, and \mathbf{r}_i is the coordinates of particle i). The sum in eq. (8.30) is then all we need to evaluate when we calculate the Slater ratio. This means we actually need to keep track of the *inverse* of the Slater matrix. We will see how we do this later.

We have implemented eq. (8.30) as shown in the code segment below.

```
double Slater::get_ratio(double* r, int p) {
    R = 0;

    for (int i = 0; i < dim; i++) {
        r_try[i] = r[i];
    }

    particle = p;

    // Get the ratio of the determinant
    for (int i = 0; i < slater_dim; i++) {
        new_slater[i] = orbital[i]->get_wf(r_try);
        R += new_slater[i] * inverse_slater[i][particle];
    }
    return R;
}
```

As its input, this function gets the new particle position and the number of the particle that has been moved. The calculation is done with the old inverse Slater matrix (i. e. the matrix before the move was done). We also store the values of the single-particle orbitals at the new position, these are needed when we update the inverse-Slater matrix.

8.1.2 The R_J -ratio

Using the Jastrow function in eq. (7.33), the ratio R_J in eq. (8.6) becomes

$$\begin{aligned} R_J &= \frac{\prod_{i<j} e^{g_{ij}^{new}}}{\prod_{i<j} e^{g_{ij}^{old}}} \\ &= e^{\Delta J} \end{aligned} \quad (8.31)$$

where we have defined

$$\Delta J \equiv \sum_{i<j} (g_{ij}^{new} - g_{ij}^{old}). \quad (8.32)$$

When we have moved only one particle (say particle k), we have that

$$g_{ij}^{new} = g_{ij}^{old} \quad i, j \neq k, \quad (8.33)$$

which reduces eq. (8.32) to

$$\Delta J = \sum_{i=1}^{k-1} (g_{ik}^{new} - g_{ik}^{old}) + \sum_{i=k+1}^N (g_{ik}^{new} - g_{ik}^{old}) \quad (8.34)$$

To calculate the Jastrow ratio, we then need to calculate eq. (8.34), and input this in eq. (8.31). The following code segment implements this.

```
double Jastrow_exp::get_ratio(double* r, int p) {
    // First copy which particle is moved and its new location
    particle = p;
    for (int i = 0; i < dimension; i++) {
        r_try[i] = r[i];
    }

    // Calculate the new distances between particles,
    // and the new Jastrow function values
    for (int i = 0; i < number_particles; i++) {
        if (i != particle) {
            r_ij_new[i] = calc_r_ij(r_try, r_now[i]);
            g_ij_new[i] = jas_func->get_value(r_ij_new[i], i, particle);
        }
    }

    // Calculate the ratio between new and old Jastrow
    // The matrix g_ij_old has entries only above the diagonal
    // so we only use these
    R = 0;
    for (int i = 0; i < particle; i++) {
        R += g_ij_new[i] - g_ij_old[i][particle];
    }
    for (int i = particle + 1; i < number_particles; i++) {
        R += g_ij_new[i] - g_ij_old[particle][i];
    }

    R = exp(R);

    return R;
}
```

8.1.3 The G_D -ratio

The algorithm we use to compute the ratio in eq. (8.12) is taken from refs. [38, 39]:

$$\frac{\nabla_i |D|}{|D|} = \sum_{j=1}^N (\nabla_i \phi_j(\mathbf{r}_i)) D_{ji}^{-1}. \quad (8.35)$$

Again, we see that it is actually the inverse of the Slater matrix that is used.

Equation (8.35) is implemented in the following code segment:

```
void Slater::get_grad(double* grad_slater, int p) {
    for (int i = 0; i < dim; i++) {
        grad_slater[i] = 0;
    }
}
```

```

for (int i = 0; i < slater_dim; i++) {
    // get the gradient from the orbital class, returns a vector
    orbital[i]->get_grad(grad, r_now[p]);
    for (int j = 0; j < dim; j++) {
        grad_slater[j] += grad[j] * inverse_slater[i][p];
    }
}
}
    
```

This function has as its input a vector where the gradient is stored (`grad_slater`), and the particle number (`p`). The function `orbital[i]->get_grad(grad, r_now[p])` then calculates the gradient of single-particle orbital $\phi_j(\mathbf{r}_p)$ according to eq. (7.28).

8.1.4 The G_J -ratio

We split eq. (8.13) into coordinate form:

$$\frac{1}{J} \frac{\partial J}{\partial x_i}. \quad (8.36)$$

Using the Jastrow in eq. (7.33), the expression above becomes

$$\frac{1}{J} \frac{\partial J}{\partial x_i} = \frac{1}{\prod_{k<l}^N e^{g(r_{kl})}} \frac{\partial \prod_{k<l}^N e^{g(r_{kl})}}{\partial x_i}. \quad (8.37)$$

The differentiation only affects the terms where either k or l is equal to i . The above expression then reduces to

$$\begin{aligned} \frac{1}{J} \frac{\partial J}{\partial x_i} &= \sum_{k=1}^{i-1} \frac{1}{e^{g(r_{ki})}} \frac{\partial e^{g(r_{ki})}}{\partial x_i} + \sum_{k=i+1}^N \frac{1}{e^{g(r_{ik})}} \frac{\partial e^{g(r_{ik})}}{\partial x_i} \\ &= \sum_{k=1}^{i-1} \frac{\partial g(r_{ki})}{\partial x_i} + \sum_{k=i+1}^N \frac{\partial g(r_{ik})}{\partial x_i} \\ &= \sum_{k=1}^{i-1} \frac{\partial g_{ki}}{\partial r_{ki}} \frac{\partial r_{ki}}{\partial x_i} + \sum_{k=i+1}^N \frac{\partial g_{ik}}{\partial r_{ik}} \frac{\partial r_{ik}}{\partial x_i}. \end{aligned} \quad (8.38)$$

We first find

$$\begin{aligned} \frac{\partial r_{ik}}{\partial x_i} &= \frac{\partial}{\partial x_i} \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \\ &= - \frac{x_k - x_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}} \\ &= \frac{x_i - x_k}{r_{ik}} \\ \frac{\partial r_{ki}}{\partial x_i} &= \frac{x_i - x_k}{r_{ki}}. \end{aligned} \quad (8.39)$$

Using the Pade-Jastrow function in eq. (7.35), we find

$$\frac{\partial g_{ik}}{\partial r_{ik}} = \frac{\partial}{\partial r_{ik}} \frac{a_{ik} r_{ik}}{1 + \beta r_{ik}} = \frac{a_{ik}}{(1 + \beta r_{ik})^2}. \quad (8.40)$$

We finally get

$$\frac{1}{J} \frac{\partial J}{\partial x_i} = \sum_{k=1}^{i-1} \frac{a_{ki}(x_i - x_k)}{r_{ki}(1 + \beta r_{ki})^2} + \sum_{k=i+1}^N \frac{a_{ik}(x_i - x_k)}{r_{ik}(1 + \beta r_{ik})^2}. \quad (8.41)$$

The expression will be equal for the other dimension.

We implement eq. (8.41) through the code segment below.

```

void Jastrow_exp::get_grad(double* grad_jastrow, int p) {
    // initialize the array
    for (int i = 0; i < dimension; i++) {
        grad_jastrow[i] = 0;
    }

    // Sum over all particles except particle p
    for (int i = 0; i < p; i++) {
        jas_func->get_grad(grad, r_now[i], r_now[p], i, p);
        for (int j = 0; j < dimension; j++) {
            grad_jastrow[j] += grad[j];
        }
    }
    for (int i = p + 1; i < number_particles; i++) {
        jas_func->get_grad(grad, r_now[i], r_now[p], i, p);
        for (int j = 0; j < dimension; j++) {
            grad_jastrow[j] += grad[j];
        }
    }
}
    
```

The function `jas_func->get_grad(grad, r_now[i], r_now[p], i, p)` calculates the value of the expression

$$\frac{a_{ki}(x_p - x_i)}{r_{pi}(1 + \beta r_{pi})^2}, \quad (8.42)$$

and similarly for the y -dimension.

8.1.5 The L_D -ratio

The algorithm we use to compute the ratio in eq. (8.17) is taken from refs. [38, 39]:

$$\frac{\nabla_i^2 |D|}{|D|} = \sum_{j=1}^N (\nabla_i^2 \phi_j(\mathbf{r}_i)) D_{ji}^{-1}. \quad (8.43)$$

We see once again that it is the inverse of the Slater matrix that is used.

We implement eq. (8.43) through the following code segment:

```

double Slater::get_lap(int p) {
    double ret = 0;
    for (int i = 0; i < slater_dim; i++) {
        // get the Laplacian from the orbital class
        ret += orbital[i]->get_lap(r_now[p]) * inverse_slater[i][p];
    }
    return ret;
}
    
```

As input, this function gets the particle number p . The function `orbital[i]->get_lap(r_now[p])` then calculates the Laplacian of single-particle orbital $\phi_j(\mathbf{r}_p)$ according to eq. (7.32).

8.1.6 The L_J -ratio

The derivation of the algorithm used to compute this ratio can be found in ref. [38]. It is a bit tedious, but not really that different from the derivation of the G_J ratio.

As shown in ref. [38], L_J can be written as

$$L_J = G_J^2 + \sum_{k=1}^{i-1} \left(\frac{d-1}{r_{ki}} \frac{\partial g_{ki}}{\partial r_{ki}} + \frac{\partial^2 g_{ki}}{\partial r_{ki}^2} \right) + \sum_{k=i+1}^N \left(\frac{d-1}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} + \frac{\partial^2 g_{ik}}{\partial r_{ik}^2} \right), \quad (8.44)$$

where d is the number of dimensions, which in our case is $d = 2$. We first find, using eq. (8.40)

$$\frac{\partial^2 g_{ik}}{\partial r_{ik}^2} = \frac{\partial}{\partial r_{ik}} \frac{a_{ik}}{(1 + \beta r_{ik})^2} = -\frac{2a_{ik}\beta}{(1 + \beta r_{ik})^3}. \quad (8.45)$$

Using eq. (8.40) and eq. (8.45), eq. (8.44) becomes

$$\begin{aligned} L_J &= G_J^2 + \sum_{k=1}^{i-1} \left(\frac{1}{r_{ki}} \frac{a_{ki}}{(1 + \beta r_{ki})^2} - \frac{2a_{ki}\beta}{(1 + \beta r_{ki})^3} \right) \\ &\quad + \sum_{k=i+1}^N \left(\frac{1}{r_{ik}} \frac{a_{ik}}{(1 + \beta r_{ik})^2} - \frac{2a_{ik}\beta}{(1 + \beta r_{ik})^3} \right) \\ &= G_J^2 + \sum_{k=1}^{i-1} \frac{a_{ki}(1 - \beta r_{ki})}{r_{ki}(1 + \beta r_{ki})^3} + \sum_{k=i+1}^N \frac{a_{ik}(1 - \beta r_{ik})}{r_{ik}(1 + \beta r_{ik})^3}. \end{aligned} \quad (8.46)$$

We implement eq. (8.46) through the following code segment:

```
double Jastrow_exp::get_lap(int p) {
    double ret = 0;
    // We split the Laplacian into two part. The first one involves
    // the gradient of the Jastrow function
    get_grad(grad_g, p);
    for (int i = 0; i < dimension; i++) {
        ret += grad_g[i]*grad_g[i];
    }

    // The second part involves both the Laplacian and the gradient
    // of the Jastrow function, but both of these are collected into
    // what we call the Laplacian in the Pade Jastrow class
    for (int i = 0; i < p; i++) {
        ret += jas_func->get_lap(r_ij_old[i][p], i, p);
    }
    for (int i = p + 1; i < number_particles; i++) {
        ret += jas_func->get_lap(r_ij_old[p][i], p, i);
    }
    return ret;
}
```

The function `jas_func->get_lap(r_ij_old[i][p], i, p)` calculates the value of the expression

$$\frac{a_{ip}(1 - \beta r_{ip})}{r_{ip}(1 + \beta r_{ip})^3}. \quad (8.47)$$

8.2 Class implementation

The classes of our program are

MC_algo: This is where the Monte Carlo algorithm is implemented. It has a Wavefunction object.

Wavefunction: This is an abstract class that contains members that are associated with the trial wave function.

quantum_dot: This is a derived class from the Wavefunction base class. Specifies a quantum dot wave function. As members it has two Slater objects (one for each spin configuration), a Jastrow object, and a potential object.

Slater: Class for handling Slater determinants. It has an array of Orbital objects.

Orbital: Class for handling the single-particle orbitals in eq. (7.19), and their derivatives.

Jastrow: Abstract class for handling Jastrow functions.

Jastrow_exp: Derived class from the Jastrow class. Specifies an exponential Jastrow function as in eq. (7.33). It has an array of Jastrow_Function objects.

Jastrow_Function: Abstract class that handles the functions in the exponent in an exponential Jastrow factor. It specifies the function $g(r_{ij})$ in eq. (7.33), and the derivatives of this function.

Pade_Jastrow: Derived class from the Jastrow_Function class. Specifies the Pade Jastrow function in eq. (7.35).

potential: Abstract class for handling the confining potential in the Hamiltonian.

harm_osc: Derived class from the potential class. Specifies an harmonic oscillator potential.

In Figure 8.1 we have included a class diagram of our program.

8.2.1 The initialization of the program

In the main of the program, before we start the MC sampling, we create objects of the classes.

```
int main(int argc, char*argv[]) {
    ...
    // Initialization of MPI and variables
    ...

    // Creates a harmonic oscillator potential object
    potential* pot = new harm_osc(omega, dimension, number_particles);

    // Define a matrix of MonteCarlo object pointers
    MC_algo** paramset = new MC_algo**[alpha_steps];

    for (int a = 0; a < alpha_steps; a++) {
        paramset[a] = new MC_algo*[beta_steps];
    }
}
```

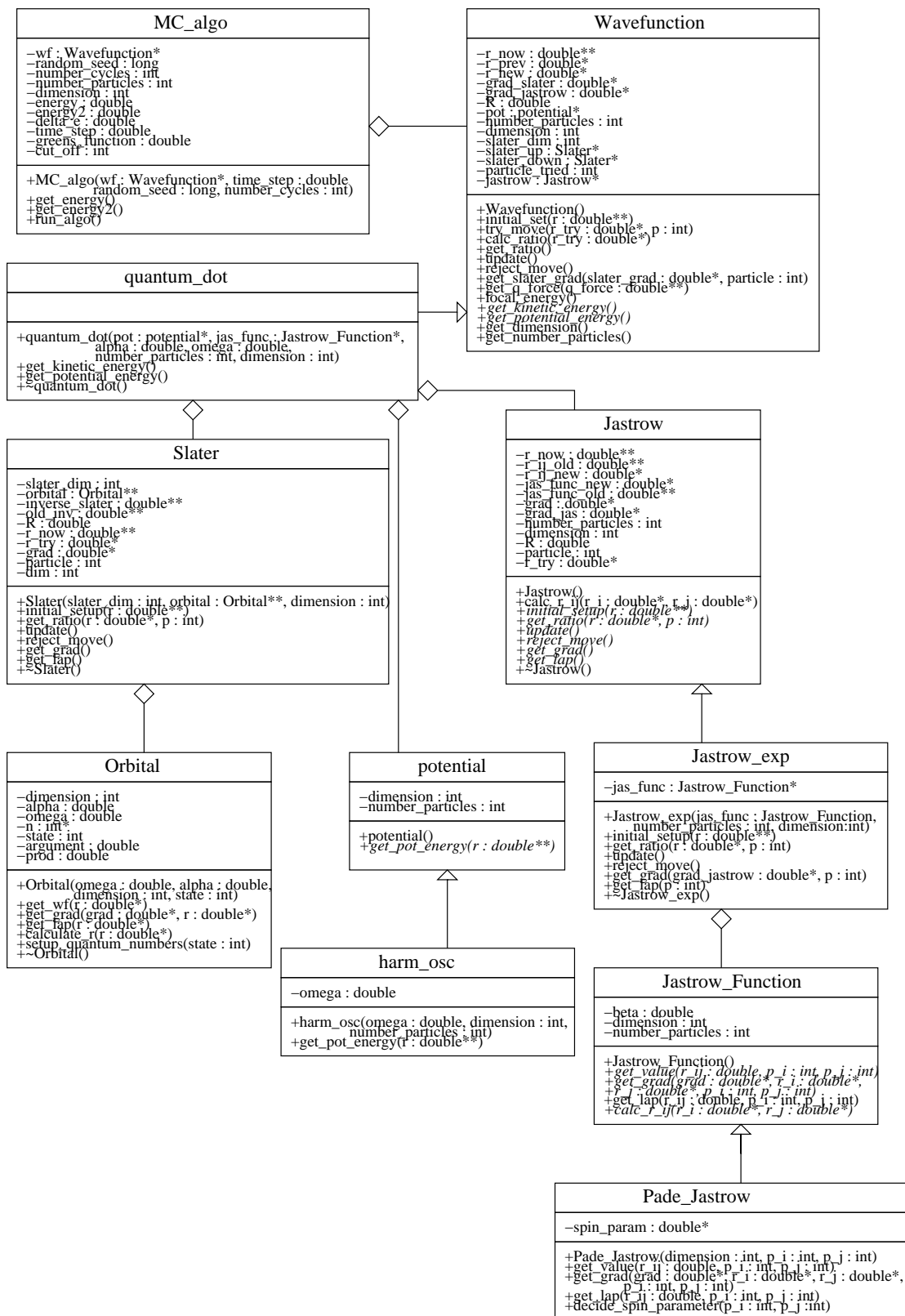


Figure 8.1: Class diagram of our QVMC program

```

for (int b = 0; b < beta_steps; b++) {
    // Create a wavefunction object for the specific alpha and beta
    long idum = -time(NULL);
    #if MPI_ON == 1
    idum -= my_rank;
    #endif
    // Define the correlation function we're using
    Jastrow_Function* jas_func =
        new Pade_Jastrow(dimension, beta_start + b*beta_plus,
                        number_particles);
    Wavefunction* wf =
        new quantum_dot(pot, jas_func, alpha_start + a*alpha_plus,
                       omega, number_particles, dimension);
    paramset[a][b] = new MC_algo(wf, time_step, idum,
                                number_cycles);
}
}
...
}

```

We start by creating an array of MC_algo objects; one for each set of parameters. Each of the MC_algo objects is given a Wavefunction object of type quantum_dot, and each quantum_dot objects has two Slater objects, a Jastrow object and a potential object. In our case, we use the Pade-Jastrow function defined in eq. (7.33) and eq. (7.35), and the Slater determinants use the single-particle orbitals from eq. (7.19). The potential we use is the harmonic oscillator potential

$$V(x, y) = \frac{1}{2}\omega^2(x^2 + y^2). \quad (8.48)$$

Inside the main the MC sampling is done through:

```

int main(int argc, char*argv[]) {
    ...
    // Run the simulation itself
    for (int a = 0; a < alpha_steps; a++) {
        for (int b = 0; b < beta_steps; b++) {
            ...
            // run the MC algorithm
            paramset[a][b]->run_algo();

            // get the energy results from the run
            energy = paramset[a][b]->get_energy();
            energy2 = paramset[a][b]->get_energy2();

            total_energy = energy;
            total_energy2 = energy2;

            #if MPI_ON == 1
            // Collect results from all nodes
            MPI_Reduce(&energy, &total_energy, 1,
                    MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
            MPI_Reduce(&energy2, &total_energy2, 1,
                    MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);

            total_energy /= numprocs;
            total_energy2 /= numprocs;
            #endif
        }
    }
}

```



```

    ...
    // Write out to file and measure the time of the iteration
    ...
}
}
...
// Finalize
...
}

```

We loop over all the different pairs of parameters, and calculate the energy for each pair in the function `paramset[a][b]->run_algo()`. Inside this function we start with the following initializations:

```

void MC_algo::run_algo() {
    ...
    // Initialize arrays and variables
    ...

    // initial trial position
    for (int i = 0; i < number_particles; i++) {
        for (int j = 0; j < dimension; j++) {
            r_old[i][j] = gaussian_deviate(&idum) * sqrt(time_step);
            r_new[i][j] = r_old[i][j];
        }
    }

    // Get the wave function at these coordinates
    wf->initial_set(r_old);
    wf->get_q_force(q_force_old);

    ...
    // Do the MC sampling
    ...
} // end function

```

We start with setting the initial positions of the particles. Then the wave function is initialized and the initial quantum force is calculated.

The initialization of the wave function is done through the function `wf->initial_set(r_old)`:

```

void Wavefunction::initial_set(double** r) {
    ...
    // Copy position
    ...

    // initialize Slater determinants
    double** r_1 = (double **) matrix(slater_dim, dimension,
                                      sizeof(double));
    double** r_2 = (double **) matrix(slater_dim, dimension,
                                      sizeof(double));

    for (int i = 0; i < slater_dim; i++) {
        for (int j = 0; j < dimension; j++) {
            r_1[i][j] = r_now[i][j];
        }
    }
    for (int i = slater_dim; i < number_particles; i++) {
        for (int j = 0; j < dimension; j++) {
            r_2[i - slater_dim][j] = r_now[i][j];
        }
    }
}

```

```

    }
  }
  slater_up->initial_setup(r_1);
  slater_down->initial_setup(r_2);

  free_matrix((void **) r_1); // free memory
  free_matrix((void **) r_2); // free memory

  // initialize Jastrow factor
  jastrow->initial_setup(r_now);
}

```

On input this function gets the initial positions of the particles. The particles are numbered from 0 to $N - 1$. It is important that we keep track of the numbering of these particles. We define that particles 0 through $N/2 - 1$ have spin up, and the rest have spin down. The $|D^\uparrow\rangle$ determinant is then initialized with the first half of the particles, and the $|D^\downarrow\rangle$ determinant is initialized with the other half. The numeration of the particles is also important for the Jastrow factor, in order to decide the value of the parameter in eq. (7.66).

The Slater objects are initialized through:

```

void Slater::initial_setup(double** r) {
  ...
  // Copy positions
  ...

  // First we calculate the Slater matrix
  for (int i = 0; i < slater_dim; i++) {
    for (int j = 0; j < slater_dim; j++) {
      inverse_slater[i][j] = orbital[j]->get_wf(r_now[i]);
    }
  }

  if (slater_dim > 1) {
    // Find the inverse of the matrix. Function inverse can be
    // found in the library file lib.cpp
    inverse(inverse_slater, slater_dim);
  }
  else { // If the Slater matrix only contains one particle, the
        // function inverse will not work. We therefore need to
        // do this check, in order for the code to work for
        // two particles
    inverse_slater[0][0] = 1./inverse_slater[0][0];
  }
}

```

This function first gets a copy of the initial position, then calculates the Slater matrix¹. This is done by evaluating the function `orbital[j]->get_wf(r_now[i])`, which calculates the value of the single-particle orbital $\phi_j(\mathbf{r}_i)$ through eq. (7.19). As we saw in Section 8.1, most of our calculations require the inverse Slater matrix, not the Slater matrix itself. We calculate the inverse matrix using a library function. The Slater matrix itself is not stored, only the inverse matrix.

¹The Slater matrix is defined as the matrix D associated with the determinant $|D\rangle$.

The Jastrow function in eq. (7.33) depends on the inter-electron distances r_{ij} . If we write this as a matrix, it will be symmetrical, which means we only need to store the entries above the main diagonal. When moving one particle, say k , we need to change only the entries along column k and row k . As stated above, we use the Pade-Jastrow function in eq. (7.35). The values of this function are stored in a matrix. This matrix will also be symmetric. The Jastrow object is initialized through:

```
void Jastrow_exp::initial_setup(double** r) {
    ...
    // Copy positions
    ...

    // Set up the r_ij_old and g_ij_old matrices,
    // making them upper triangular.
    // The value of r_ij_old[i][j] is the distance
    // between particle i and particle j.
    for (int i = 0; i < number_particles; i++) {
        for (int j = i + 1; j < number_particles; j++) {
            r_ij_old[i][j] = calc_r_ij(r_now[i], r_now[j]);
            g_ij_old[i][j] = jas_func->get_value(r_ij_old[i][j], i, j);
        }
    }
}
```

As its input, this function gets a copy of the position of all the particles. Then the intermediate distances, r_{ij} , are calculated. The function `jas_func->get_value(r_ij_old[i][j], i, j)` then calculates the value of g_{ij} , according to eq.(7.35).

8.2.2 Implementation of the Metropolis-Hastings algorithm

After the initialization shown in the previous section is done, we are ready to start the MC sampling. This is all done in the function `MC_algo::run_algo()`. We start out by:

```
void MC_algo::run_algo() {
    ...
    // Initializations
    ...
    // loop over monte carlo cycles
    for (int cycles = 1; cycles <= number_cycles + cut_off; cycles++){
        // new position
        for (int i = 0; i < number_particles; i++) {
            for (int j = 0; j < dimension; j++) {
                r_new[i][j] = r_old[i][j] +
                    (D * q_force_old[i][j] * time_step) +
                    (gaussian_deviate(&idum) * sqrt(time_step));
            }
        }
        ...
    } // end function
}
```

The array `q_force_old[i][j]` contains the quantum force (eq. (6.69)) for all the particles at the old positions. The function `gaussian_deviate` creates a Gaussian distributed random number. The new positions of the particles moved is calculated through eq. (6.72). After this is done, we tell the wavefunction to try out the new position. We calculate the new quantum force, the Green's function and the ratio between new and old wave function. This is all done through:

```
void MC_algo::run_algo() {
```

```

...
// Try out the new position
wf->try_move(r_new[i], i);
...
} // end function
\end{lstlisting}
The function \emph{wf->try\_move(r\_new[i], i)} does the following:
\begin{lstlisting}
void Wavefunction::try_move(double* r_try, int p) {
...
// Make a backup of the particles old position
...

// Calculate the ratio of determinants and Jastrow
calc_ratio(r_try);
// update determinant and Jastrow
update();
}
    
```

It gets the number of the particle that has been moved, p , and the new position of this particle on input. The ratio R' from eq. (8.3) is then calculated, before the Wavefunction is updated in the function:

```

void Wavefunction::update() {
// Update slater determinant
if (particle_tried < slater_dim) {
    slater_up->update();
}
else {
    slater_down->update();
}

// Update Jastrow factor
jastrow->update();

// Update position with pointer switching
double* r_temp = r_now[particle_tried];
r_now[particle_tried] = r_new;
r_new = r_temp;
}
    
```

This function updates the relevant Slater determinant (i. e. the determinant that contains the particle that has been moved), and then updates the Jastrow factor.

As we have observed, in the Slater objects we keep track of the *inverse*-Slater matrix, not the actual Slater matrix. When a particle is moved, we update the inverse matrix through an algorithm taken from ref. [39]. We first calculate

$$S_j \equiv \sum_{k=1}^N \phi_k(\mathbf{r}_i^{new})(D_{kj}^{old})^{-1}. \quad (8.49)$$

The new elements in the inverse matrix are then given as

$$(D_{lj}^{new})^{-1} = \begin{cases} (D_{lj}^{old})^{-1} - \frac{S_j(D_{li}^{old})^{-1}}{R_D} & \text{if } j \neq i, \\ \frac{(D_{li}^{old})^{-1}}{R_D} & \text{if } j = i. \end{cases} \quad (8.50)$$

The algorithm above is implemented in the following code:

```

void Slater::update() {
    ...
    // Make a backup of the matrix in case we reject the move
    ...

    // Update the inverse matrix according to the algorithm
    // First we update every column but number p
    for (int i = 0; i < p; i++) {
        double S = 0.0;
        // Calculate S
        for(int j = 0; j < slater_dim; j++) {
            S += new_slater[j] * inverse_slater[j][i];
        }
        for (int j = 0; j < slater_dim; j++) {
            inverse_slater[j][i] -= (S/R) * inverse_slater[j][p];
        }
    }
    for (int i = p + 1; i < slater_dim; i++) {
        double S = 0.0;
        // Calculate S
        for(int j = 0; j < slater_dim; j++) {
            S += new_slater[j] * inverse_slater[j][i];
        }
        for (int j = 0; j < slater_dim; j++) {
            inverse_slater[j][i] -= (S/R) * inverse_slater[j][p];
        }
    }

    // Update column p
    for (int i = 0; i < slater_dim; i++) {
        inverse_slater[i][p] /= R;
    }

    ...
    // Update positions
    ...
}

```

The Jastrow is updated through the function:

```

void Jastrow_exp::update() {
    double memory_switch;
    // First update r_now
    double* temp = r_now[particle];
    r_now[particle] = r_try;
    r_try = temp; // Old position is now stored in r_try

    // Update r_ij_old
    temp = r_ij_old[particle];
    r_ij_old[particle] = r_ij_new;
    r_ij_new = temp; // Old r_ij is now stored in r_ij_new
    // All the new values are stored as a row in the matrix
    // we need to change it so the matrix is still upper diagonal
    for (int i = 0; i < particle; i++) {
        memory_switch = r_ij_old[i][particle];
        r_ij_old[i][particle] = r_ij_old[particle][i];
        r_ij_new[i] = memory_switch;
    }
}

```

```

// Update g_ij_old in the same way
temp = g_ij_old[particle];
g_ij_old[particle] = g_ij_new;
g_ij_new = temp; // Old position is now stored in g_ij_new
for (int i = 0; i < particle; i++) {
    memory_switch = g_ij_old[i][particle];
    g_ij_old[i][particle] = g_ij_old[particle][i];
    g_ij_new[i] = memory_switch;
}
}

```

When a particle is moved, lets say particle number p , the only values r_{ij} and g_{ij} that are changed are the values where either i or j are equal to p . This means that the values along row p , and column p in the matrices for r_{ij} and g_{ij} are changed.

After the wave function is updated, we do the following:

```

void MC_algo::run_algo() {
    ...
    // Get new quantum force
    wf->get_q_force(q_force_new);

    // Metropolis-Hastings algo
    greens_function = 0;
    for (int p = 0; p < number_particles; p++) {
        for (int j = 0; j < dimension; j++) {
            greens_function += 0.5 *
                (q_force_old[p][j] + q_force_new[p][j]) *
                (D * time_step * 0.5 *
                (q_force_old[p][j] - q_force_new[p][j]) +
                r_old[p][j] - r_new[p][j]);
        }
    }
    greens_function = exp(greens_function);
    R = wf->get_ratio();
    R = R*R;
    double random = ran2(&idum);
    double test = greens_function*R;

    // perform the Metropolis test
    if (random <= test) {
        // Accept the move
        for (int j = 0; j < dimension; j++) {
            // Update position
            r_old[i][j] = r_new[i][j];
        }
        // Update quantum force by pointer switching
        double** temp = q_force_old;
        q_force_old = q_force_new;
        q_force_new = temp;
    }
    else {
        // Reject move in the wavefunction
        wf->reject_move();
        for (int j = 0; j < dimension; j++) {
            // Reset position
            r_new[i][j] = r_old[i][j];
        }
    }
}
}

```

```

...
} end function

```

The quantum force at the new position is calculated, then the Green's function (or rather, ratio of Green's functions) is calculated according to eq. (6.79). Then we calculate the ratio in eq. (6.80). The Metropolis test is done as described in Section 6.5.3. If the new position is accepted, we update the positions and quantum force. If it is rejected, we set all the objects back to the old values. This is done in the function `wf->reject_move()`.

After all particles have been tried to be moved, we update the expectation values. This is done through:

```

void MC_algo::run_algo() {
    ...

    if (cycles > cut_off) {
        // compute local energy
        delta_e = wf->local_energy();
        // update energies
        energy += delta_e;
        energy2 += delta_e*delta_e;
    }
    // end of loop over MC trials
    // update the energy average and its square
    energy /= number_cycles;
    energy2 /= number_cycles;
    ...
    // Free up memory
    ...
} // end function

```

The expectation values are calculated as discussed in Section 6.5.1. The local energy is calculated as the sum of the kinetic energy and the potential energy. The kinetic energy term is calculated through

$$\sum_{i=1}^N -\frac{1}{2} \frac{\nabla_i \Psi_T}{\Psi_T} = \sum_{i=1}^N -\frac{1}{2} \left(\frac{\nabla_i^2 |D|}{|D|} + \frac{\nabla_i^2 J}{J} + 2 \frac{\nabla_i |D|}{|D|} \frac{\nabla_i J}{J} \right), \quad (8.51)$$

which is implemented in the following code segment:

```

double quantum_dot::get_kinetic_energy() {
    double ret = 0;
    // Contribution from the Laplacian of the Slater determinant part
    for (int i = 0; i < slater_dim; i++) {
        ret -= slater_up->get_lap(i);
    }
    for (int i = 0; i < slater_dim; i++) {
        ret -= slater_down->get_lap(i);
    }

    // Contribution from the Laplacian of the Jastrow part
    for (int i = 0; i < number_particles; i++) {
        ret -= jastrow->get_lap(i);
    }

    // Contribution from the coupling of the Slater determinant
    // and the Jastrow

```

```

for (int i = 0; i < number_particles; i++) {
    get_slater_grad(grad_slater, i);
    jastrow->get_grad(grad_jastrow, i);
    for (int i = 0; i < dimension; i++) {
        ret -= 2*grad_slater[i]*grad_jastrow[i];
    }
}

return 0.5*ret;
}

```

We could have used numerical derivatives, but since we have an analytical form of the wave function, it is much faster to do analytical derivatives.

The potential energy includes both the harmonic oscillator potential and the Coulomb energy from the electron-electron repulsion. This is calculated in the following code segment:

```

double quantum_dot::get_potential_energy() {
    double e_potential = 0;

    // First calculate energy from the coulomb interaction between
    // electrons
    double r_ij;

    for (int i = 0; i < number_particles; i++) {
        for (int j = i + 1; j < number_particles; j++) {
            r_ij = 0;
            for (int k = 0; k < dimension; k++) {
                // calculates squared distance between electron i and j
                r_ij += (r_now[i][k] - r_now[j][k]) *
                    (r_now[i][k] - r_now[j][k]);
            }
            e_potential += 1./sqrt(r_ij);
        }
    }

    return e_potential + pot->get_pot_energy(r_now);
}

```

The function `pot->get_pot_energy(r_now)` returns the potential energy from the harmonic oscillator potential at the current position. The local energies for each cycle of the MC sampling is summed up. After all cycles have been finished, we divide the expectation values by the number of samples.

8.3 Parallel computing

One of the major strengths of the MC algorithm is that it is very easy to parallelize. The possibility of running the program in parallel has been implemented in our code.

Parallel computing means that large tasks are split up in many smaller tasks that are sent to different processes. Each individual process performs its task, and finally all the results are collected.

In our work, we use the MPI library to parallelize the program. The strategy for our

program is that we decide on how many codes we want to use, and we then let each process calculate the expectations values of the energy, and energy squared. Each process is given different random seeds, so that they have different initial positions for the particles. If each process then do 10^6 MC cycles, and we have 10 processes doing calculations, we are in effect doing 10^7 MC cycles. If we have many processes at our disposal, this will greatly reduce the time it takes to do a desired amount of MC cycles.

In this work we use the *Titan* computer cluster for large parallel jobs. The Titan computer cluster is described on its homepage <http://hpc.uio.no>.

8.4 Implementing the DFP algorithm

We want to implement the DFP algorithm from Section 6.6. We then implement the function *dfpmin* from ref. [43]. This function utilizes the DFP method to find a minimum of a function f . In our case, this function will be the expectation value of the energy discussed in Section 6.5.1.

As input, the DFP function needs a pointer to the function it minimizes, as well as a pointer to a function containing the gradient of the function it minimizes. In our case we want to minimize the energy of a trial wave function. To calculate the energy, we use the QVMC method. We must also calculate the gradient of the expectation value of the energy. The partial derivative of $\langle E \rangle$ with respects to just one variational parameter is

$$\begin{aligned} \langle E \rangle_i &\equiv \frac{\partial \langle E \rangle}{\partial \alpha_i} = \left\langle \frac{\Psi_{T,i}}{\Psi_T} E_L + \frac{\hat{H} \Psi_T}{\Psi_T} - 2 \langle E \rangle \frac{\Psi_{T,i}}{\Psi_T} \right\rangle \\ &= 2 \left\langle \frac{\Psi_{T,i}}{\Psi_T} (E_L - \langle E \rangle) \right\rangle = 2 \left\langle \frac{\Psi_{T,i}}{\Psi_T} E_L \right\rangle - 2 \left\langle \frac{\Psi_{T,i}}{\Psi_T} \right\rangle \langle E \rangle, \end{aligned} \quad (8.52)$$

where we have defined $\Psi_{T,i} = \partial \Psi_T / \partial \alpha_i$. The expectation values in eq. (8.52) cannot easily be calculated using closed form expression. In general we must evaluate them numerically. In this work we use a simple estimate for the first derivative

$$f'(x) = \frac{f(x+h) - f(x-h)}{h}, \quad (8.53)$$

with $h = 0.001$.

We have implemented this through the following code segment:

```
int main(int argc, char*argv[]) {
    ...
    // various declarations and initializations
    ...
    // Do the CGM
    dfpmin(param, number_parameters, g_tol, &iterations, &energy,
    &E_function, &dE_function);
    ...
    // Print out results
    ...
}
```

dfpmin gets the address of the MC object *E_function* and the gradient *dE_function*. The DFP algorithm then uses an MC simulation to calculate the energy. Inside the

MC simulation, we compute derivatives numerically in order to calculate the expectation values in eq. (8.52) as well, by:

```

...
// MC calculations
...
    // compute local energy
    delta_e = wf->local_energy();

    // Partial derivatives
    wf->get_partial_psi_over_psi(partial_psi_over_psi,
                                number_parameters);

    // compute expectation values
    for (int i = 0; i < number_parameters; i++) {
        exp_val_part_psi_over_psi[i] += partial_psi_over_psi[i];
        exp_val_e_local_times_part_psi_over_psi[i] +=
            delta_e*partial_psi_over_psi[i];
    }

    // update energy
    energy += delta_e;
}
} // end of loop over MC trials
// update the expectation values
energy /= number_cycles;

for (int i = 0; i < number_parameters; i++) {
    exp_val_part_psi_over_psi[i] /= number_cycles;
    exp_val_e_local_times_part_psi_over_psi[i] /= number_cycles;
}

for (int i = 0; i < number_parameters; i++) {
    gradient[i] = 2 * (exp_val_e_local_times_part_psi_over_psi[i]
                      - energy*exp_val_part_psi_over_psi[i]);
}
...
// free memory and return the energy
...
}

```

This means that when $E_function$ is called, it calculates all the relevant expectation values, and then stores the gradient so that it can be collected when $dE_function$ is called.

8.5 Implementation of blocking

The implementation of the blocking technique discussed in Section 6.7 is rather easy. The first thing we do is to find the optimal parameters by using the QVMC method discussed above. We then do a single QVMC calculation using the optimal parameters, and store the local value calculated for each MC loop. We write these values to a file. A separate program does the actual blocking. The main of that program looks like this:

```

int main(int nargs, char* args [])
{
    int n_procs, min_block_size, max_block_size, n_block_samples, i;

```

```

// The program needs four comand line arguments
if(nargs > 4) {
    n_procs = atoi(args[1]); // First input is the number of
                            // processors used in the MPI
    min_block_size = atoi(args[2]); // Second input is the minimum
                                    // block size
    max_block_size = atoi(args[3]); // Third input is the maximum
                                    // block size
    n_block_samples = atoi(args[4]); // Fourth input is the number
                                    // of block samples we want
}
else {
    ...
    // quit program and write out an error report
    ...
}

...
// Collect the MC results from all the nodes used in the simulation
// and initialize out file
...

int block_size, block_step_length;
double res[2];
double mean, sigma;

block_step_length = (max_block_size - min_block_size) /
    n_block_samples;

// We loop over all block sizes
for(i = 0; i < n_block_samples; i++) {
    block_size = min_block_size + i * block_step_length;
    blocking(mc_results, n, block_size, res);

    mean = res[0];
    sigma = res[1];

    ...
    // Output
    ...
}
...
// end program
...
}

```

The program start by reading from the command line the number of processes used when doing the MC simulations, then reads the minimum and maximum block size we want to use, and finally the number of block samples we want. Then the values from the MC simulation is collected, before the blocking procedure itself is done. The function `blocking(mc_results, n, block_size, res)` looks like:

```

// Finds mean value and variance of a block of a given size
void blocking(double *vals, int n_vals, int block_size, double *res)
{
    int n_blocks = n_vals/block_size;
    double* block_vals = new double[n_blocks];
    int i;

    for(i = 0; i < n_blocks; i++) {

```

```
    block_vals[i] = mean(vals + i * block_size, block_size);  
  }  
  meanvar(block_vals, n_blocks, res);  
  delete block_vals;  
}
```

As input it gets an array *vals* with all the MC values, an integer *n_vals* that tells the number of values in *vals*, the integer *block_size*, which tells us the desired block size for this particular calculation and finally the array *res*, which is where we return the results. The function *mean(vals + i * block_size, block_size)* calculates the mean of a block, and the function *meanvar(block_vals, n_blocks, res)* calculates the mean and variance of all the blocks.

8.6 Implementation of time-step extrapolation

To extrapolate the energy and error towards $\Delta t = 0$, we do VMC calculations on several different values of Δt , and blocking for each of those values. We then implement a numerical recipe from Section 15.4 in ref. [43], which gives us a linear fitting of the results, and an estimate of the energy and error at $\Delta t = 0$. The code we use for this is an old FORTRAN code supplied by Morten Hjorth-Jensen.

Chapter 9

Computational results

In this chapter we present the results from our simulations. We start out by validating that the code yields the correct results for the non-interacting system and the two-particle system. We then move on to minimizing the trial wave function for closed-shell systems of various sizes. We will use brute-force variation (varying the parameters manually), and the Davidon-Fletcher-Powell (DFP) algorithm to find the optimal parameters. The next step is to do blocking and time-step extrapolation with the optimal parameters, in order to obtain correct results for the energy and the error. We discuss also the dependency on the oscillator frequency ω . Finally, we will present and discuss the results obtained.

We will mainly consider systems with $N \in \{2, 6, 12, 20\}$, where N is the number of electrons in the dot. We will consider oscillator frequencies $\omega \in \{0.28, 0.5, 1\}$. At the very end of the chapter, we will present results for the 30- and 42-electron systems, then only considering $\omega = 1$.

We use atomic units (a.u.) in all calculations. This corresponds to our scaling of the Hamiltonian (see Section 5.4). Energies are then given in units of Hartrees (Ha). We only consider closed-shell systems as discussed in Section 5.1. For brevity we usually only say system, but it is implied that we mean closed-shell systems.

9.1 Validating the code

Before we do any large computation, it is important to validate that our code can give us the correct results for systems where we have analytical values for the energy. We start out by computing the ground-state energy for the non-interacting system. We then compute the energy for the two-particle system. If we get good results for these two systems, we are ready to do computations on more complicated systems.

9.1.1 The non-interacting system

From our discussion in Section 3.3.1, we know that the ground state of the non-interacting system can be exactly represented by a Slater determinant filled with the single-particle orbitals with lowest energy eigenvalues. The energy of a single-particle orbital is given in eq. (7.16), and the energy of the Slater determinant is given in eq. (3.37). In Table 9.1 we present the analytical energy of the non-interacting system for the first few closed-shell systems. We have used dimensionless units and the oscillator frequency is set to 1.

Table 9.1: The table shows the energy of a quantum dot with non-interacting particles. The energy is calculated for $\omega = 1$ and is given in units of Hartrees.

Shell number R	Number of particles	Energy (Ha)
1	2	2
2	6	10
3	12	28
4	20	60

Table 9.2: In this table we present our results from simulations of a closed-shell quantum dots with no correlation. The table shows the experimental results for the energy, and the variance. We also include how many MC cycles we have used in the simulations. The energy is given in units of Hartrees.

Number of particles	Energy (Ha)	Variance	#MC cycles
2	2.0000000	$\sigma^2 = 0.0000000$	10^6
6	10.0000000	$\sigma^2 = 0.0000000$	10^6
12	28.0000000	$\sigma^2 = 0.0000000$	10^6
20	60.0000000	$\sigma^2 = 0.0000000$	10^6

Our trial wave function is defined as in eq. (7.64), but with no Jastrow factor. This means that our trial wave function is on the correct theoretical form. Our single-particle orbitals are given as in eq. (7.19). The correct orbitals are given in eq. (7.15). This means that for $\alpha = 1$, we should obtain something very close to the exact result.

In our calculations we used an oscillator frequency of $\omega = 1$. We have included the results of the simulations in Table 9.2. Comparing Table 9.2 to Table 9.1 we see that we seem to get the exact results and a variance of zero. This is in agreement with our discussion in Section 6.5.1; the more closely our trial wave function resembles the exact wave function, the less strongly will E_L (the local energy) vary with the position. When $\alpha = 1$, our trial wave function is actually the exact wave function of the system. This means that the variance should be equal to zero (except for perhaps some small statistical fluctuations). In Figure 9.1 we have included a plot that shows how the energy varies with α . The figure shows a clear minimum for $\alpha = 1$.

We have also run the code for different values of ω , and reproduced the results predicted by eq. (3.37) and eq. (7.16). That our code is able to reproduce the exact results for the non-interacting system is a clear indication that the Slater determinant part of our program is correctly coded.

9.1.2 The two-particle case

In Section 5.2 we discussed the two-electron quantum dot, and we found the exact ground-state energy for a dot with an oscillator frequency $\omega = 1$. The energy should be $E_0 = 3E_H$ with our scaling of the system. This is a good benchmark to test if our Jastrow part is coded correct.

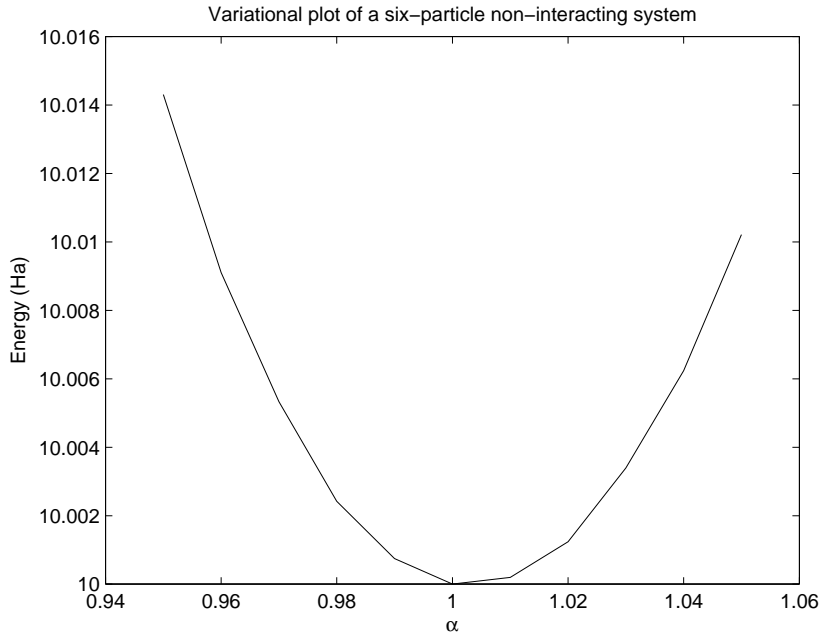


Figure 9.1: Shows the variational plot of the non-interacting system for 6 particles. We have used 1 000 000 Monte-Carlo (MC) cycles for each value of α , and varied α from 0.95 to 1.05 in steps of 0.01. The figure shows a clear minimum for $\alpha = 1$. Energies are in units of Hartrees.

In this case, we do not know the exact form of the wave function, and it is likely that our trial wave function does not have the exact form.

We first varied α from 0.5 to 1.4 in steps of 0.1, and β from 0.1 to 1 in steps of 0.1. We did 10^4 Monte-Carlo (MC) cycles for each iteration. We then closed in on the region where the energy was small. When we closed in on the good values of the variational parameters, we increased the number of MC cycles.

A final run was made with 100 million cycles. We did 100 variations in both α and β , meaning we had a total of 10 000 iterations. The resulting variational plot of the energy is shown in Figure 9.2. The optimal parameters are found by considering this plot.

As discussed in Section 6.5.1, a good way to find the ground state could also be to search for a minimum in the variance. We will not do this in our later discussion, but take the time in this section to verify that it is correct. In Figure 9.3 we have included a variational plot of the variance for the two-electron quantum dot.

We collect the results of this section in Table 9.3. We see that we get different parameters depending on whether we choose to minimize the variance or the energy. The differences are not very large, though, which leads us to believe that our wave-function *ansatz* works quite well for this system. The energy we get is quite close to the exact result from Section 5.2. The error is calculated according to eq. (A.32), which means we have assumed no correlations.

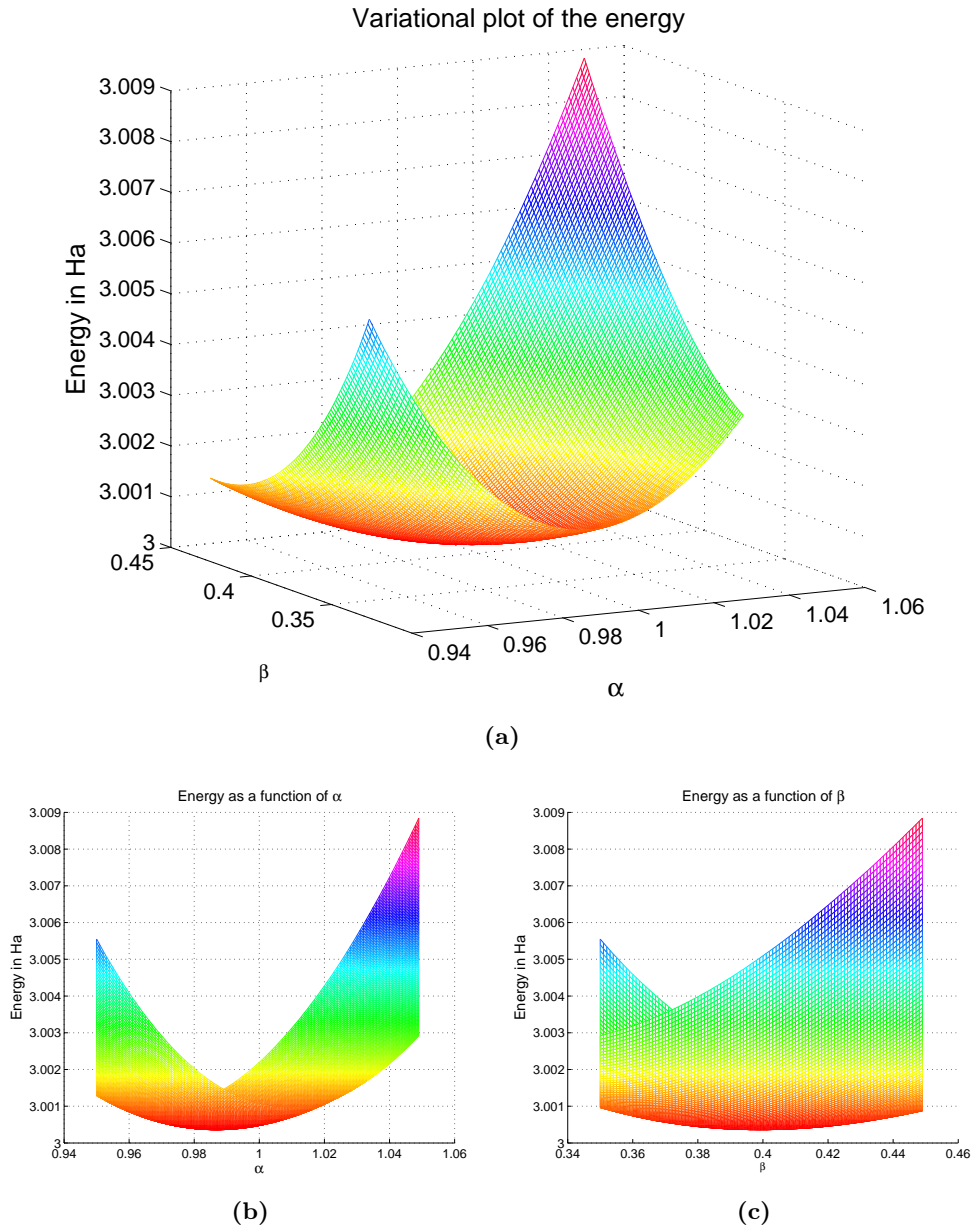


Figure 9.2: In (a) we have plotted the variational results for the energy for a two-electron quantum dot. The simulation was done with 10^8 MC samples, and 100 variations in both α and β . From (b) we read the optimal value of α , and from (c) we read the optimal value of β . We find the minimum at $\alpha \approx 0.987$ and $\beta \approx 0.398$. The energy is in units of Hartrees.

Table 9.3: In the first row in the table we present the parameters that minimize the energy, while in the second row we present the parameters that minimize the variance. All results are for the two-electron quantum dot.

α	β	Energy (Ha)	Variance	Error
0.987	0.398	$\langle E \rangle = 3.00036$	$\sigma^2 = 0.00185$	$4.3012E - 06$
0.980	0.420	$\langle E \rangle = 3.00046$	$\sigma^2 = 0.00156$	$3.9497E - 06$

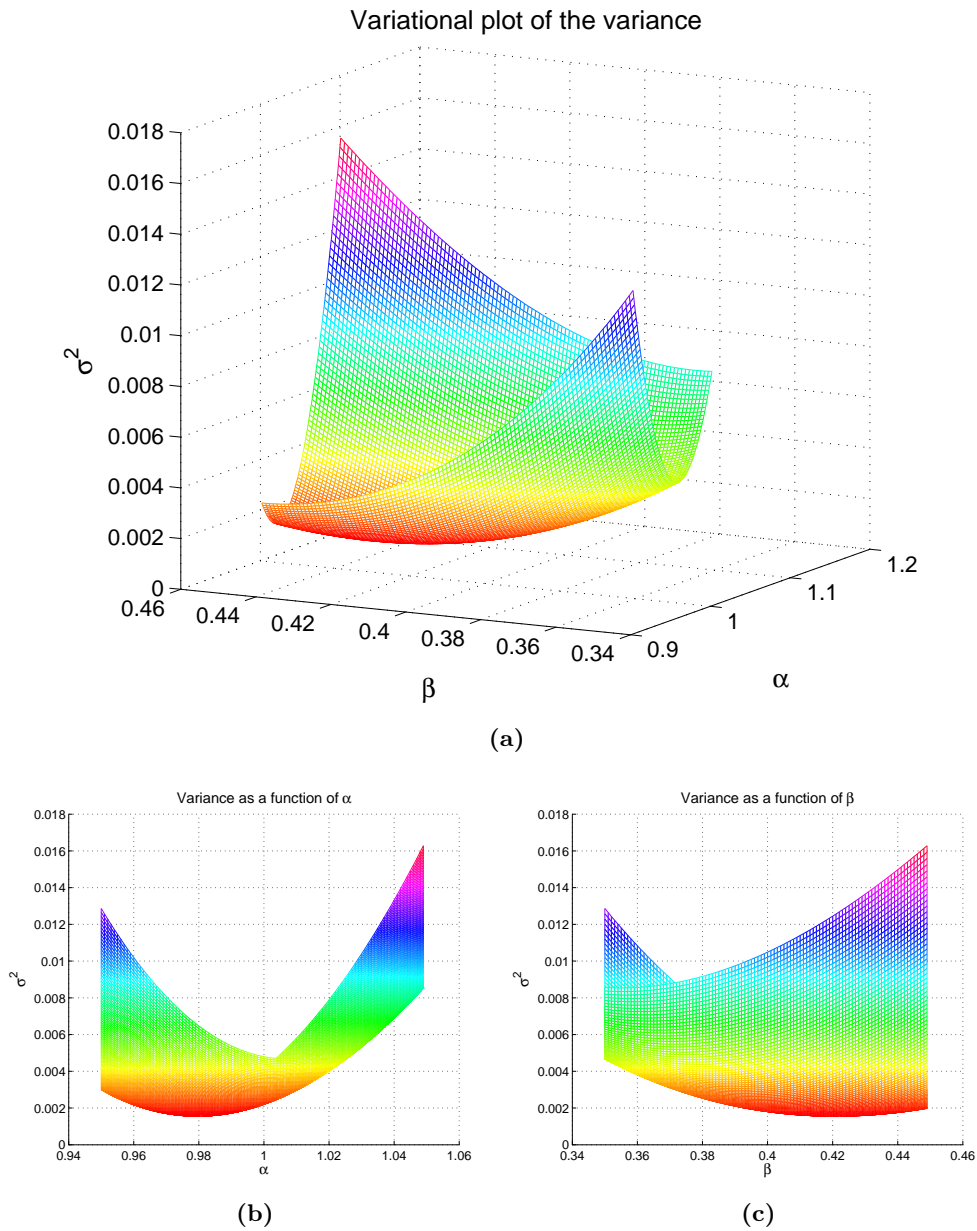


Figure 9.3: In (a) we have plotted the variational results for the variance for a two-electron quantum dot. The simulation was done with 10^8 MC samples. We have done 100 variations in both α and β . From (b) we read the optimal value of α , and from (c) we read the optimal value of β . We find the minimum at $\alpha \approx 0.98$ and $\beta \approx 0.42$.

Table 9.4: In this table we present the optimal parameters found for six-, twelve- and twenty-electron quantum dots, together with a first approximation of the energy for the systems. The results are calculated with $\omega = 1$.

N	α	β	Energy (Ha)
6	0.92	0.565	$\langle E \rangle = 20.19$
12	0.87	0.68	$\langle E \rangle = 65.79$
20	0.84	0.76	$\langle E \rangle = 156.06$

To get a more correct estimate for the energy and error, we should do blocking and time-step extrapolation. We do this in Section 9.3.1.

The code seems to be working well for both the non-interacting and the interacting system. We are then ready to do simulations of systems with more electrons.

9.2 Variational plots — Finding the optimal parameters

In this section we will present the result from QVMC simulations on quantum dots with six, twelve and twenty electrons. We show the variational plots, and find the optimal parameters. All the results presented in this section are calculated with the oscillator frequency $\omega = 1$. At the end, we will also present results from the DFP minimization of two-, six- and twelve-electron quantum dots.

9.2.1 Simulation of a six-electron quantum dot

To get the optimal parameters for the six-electron system, we first did small calculations to find the areas of α and β where the energy was small. We then did a large QVMC simulations with parameters

$$\alpha \in \{0.890, 0.895, 0.900, 0.905, 0.910, 0.915, 0.920, 0.925, 0.930, 0.935\},$$

$$\beta \in \{0.530, 0.538, 0.546, 0.554, 0.562, 0.570, 0.578, 0.586, 0.594, 0.602\},$$

and 100 000 000 MC cycles per iteration. The resulting variational plot of the energy is shown in Figure 9.4.

We have tabulated the results in Table 9.4, together with a first approximation of the energy of the system. We will do blocking and time-step extrapolation for the optimal parameters in Section 9.3.2 .

9.2.2 Simulation of a twelve-electron quantum dot

We repeat the procedure for the twelve-electron system. In this case it was hard to decide on good optimal parameters, because the function is quite flat in both the α and the β parameter around the minimum. We did a final QVMC simulation with the parameters

$$\alpha \in \{0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89\},$$

$$\beta \in \{0.620, 0.635, 0.650, 0.665, 0.680, 0.695, 0.710, 0.725\},$$

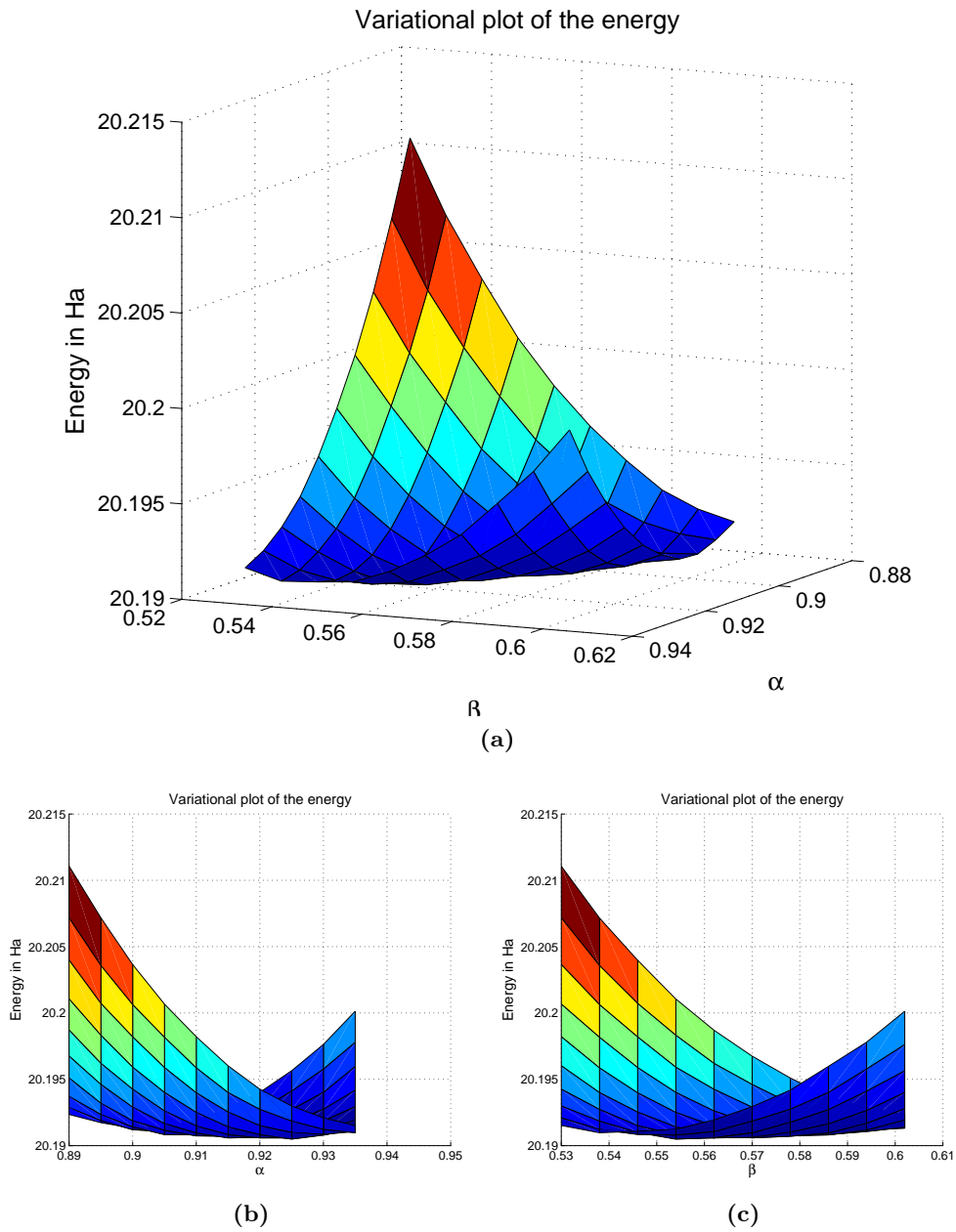


Figure 9.4: In (a) we have plotted the variational results for the energy for a six-electron quantum dot. The simulation was with 10^8 MC samples. We have done 10 variations in both α and β . From (b) we can read of the optimal value of α , and from (c) we read of the optimal value of β . We find the minimum at $\alpha \approx 0.920$ and $\beta \approx 0.565$.

Table 9.5: In this table we present the optimal parameters and energy found for two-, six- and twelve-electron quantum dots using the DFP algorithm. We see that the results do not differ much from the results in table 9.4.

N	α	β	Energy (Ha)
2	0.987	0.4	$\langle E \rangle = 3.00044$
6	0.928	0.55	$\langle E \rangle = 20.1926$
12	0.877	0.678	$\langle E \rangle = 65.7953$

and 10 000 000 MC cycles per iteration. The resulting variational plot of the energy is shown in Figure 9.7. We have tabulated the results in Table 9.4, together with a first approximation of the energy of the system. We will do blocking and time-step extrapolation for the optimal parameters in Section 9.3.3.

9.2.3 Simulation of a twenty-electron quantum dot

We include also variational plots for the 20-electron system. The procedure is the same as before. The final QVMC simulation was made with the parameters

$$\begin{aligned} \alpha &\in \{0.78, 0.79, 0.80, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87\}, \\ \beta &\in \{0.700, 0.715, 0.730, 0.745, 0.760, 0.775, 0.790, 0.805, 0.820, 0.835\}. \end{aligned} \quad (9.1)$$

With a system of 20-electrons, the computations are starting to take a lot of time. Due to time limitations we are not able to do a very large simulation on this system. The final run was made with 1 000 000 MC cycles. The resulting plots are shown in Figure 9.8, and the optimal parameters are tabulated in 9.4, together with the first approximation of the energy.

9.2.4 Minimization with DFP

We want to use the minimization technique discussed in section 6.6. This is implemented as discussed in Section 8.4. We use this code to obtain the optimal parameters for the trial wave function. In Table 9.5 we have tabulated the results for quantum dots with two, six and twelve electrons. We see that the results are quite similar to the results in Tables 9.3 and 9.4.

This method works quite well for small systems. The advantage using this approach is that we do not have to manually vary the parameters. Instead the program searches for a minimum in a more intelligent manner. This means we do not have to use as many iterations as we did before. However, using this method, we need to calculate the gradient of Ψ_T with respects to α and β , and this is done numerically. We then need to evaluate the function Ψ_T four times in each MC loop. Evaluating this function takes a lot of computational power when N is large. This means that the DFP algorithm is not well suited for handling systems with a large number of particles, and several variational parameters.

It is possible to do a Hartree-Fock calculation of the system, and then parametrize the results in Gaussian type orbitals. We can then use these orbitals as our single-particle

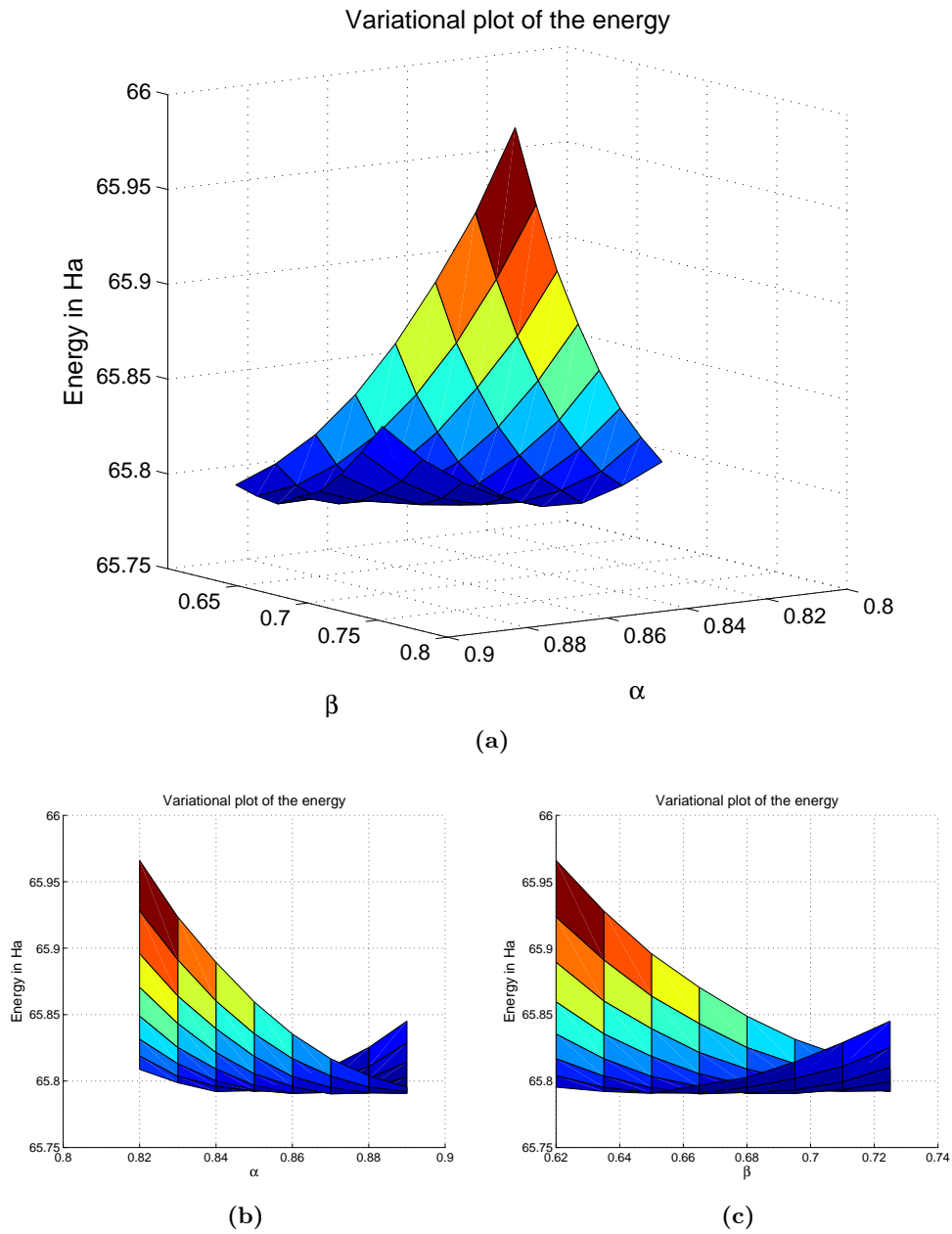


Figure 9.5: In (a) we have plotted the variational results for the energy for a twelve-electron quantum dot. The simulation was with 10^7 MC samples. We have done 10 variations in both α and β . From (b) we read the optimal value of α , and from (c) we read the optimal value of β . We find the minimum at $\alpha \approx 0.87$ and $\beta \approx 0.68$.

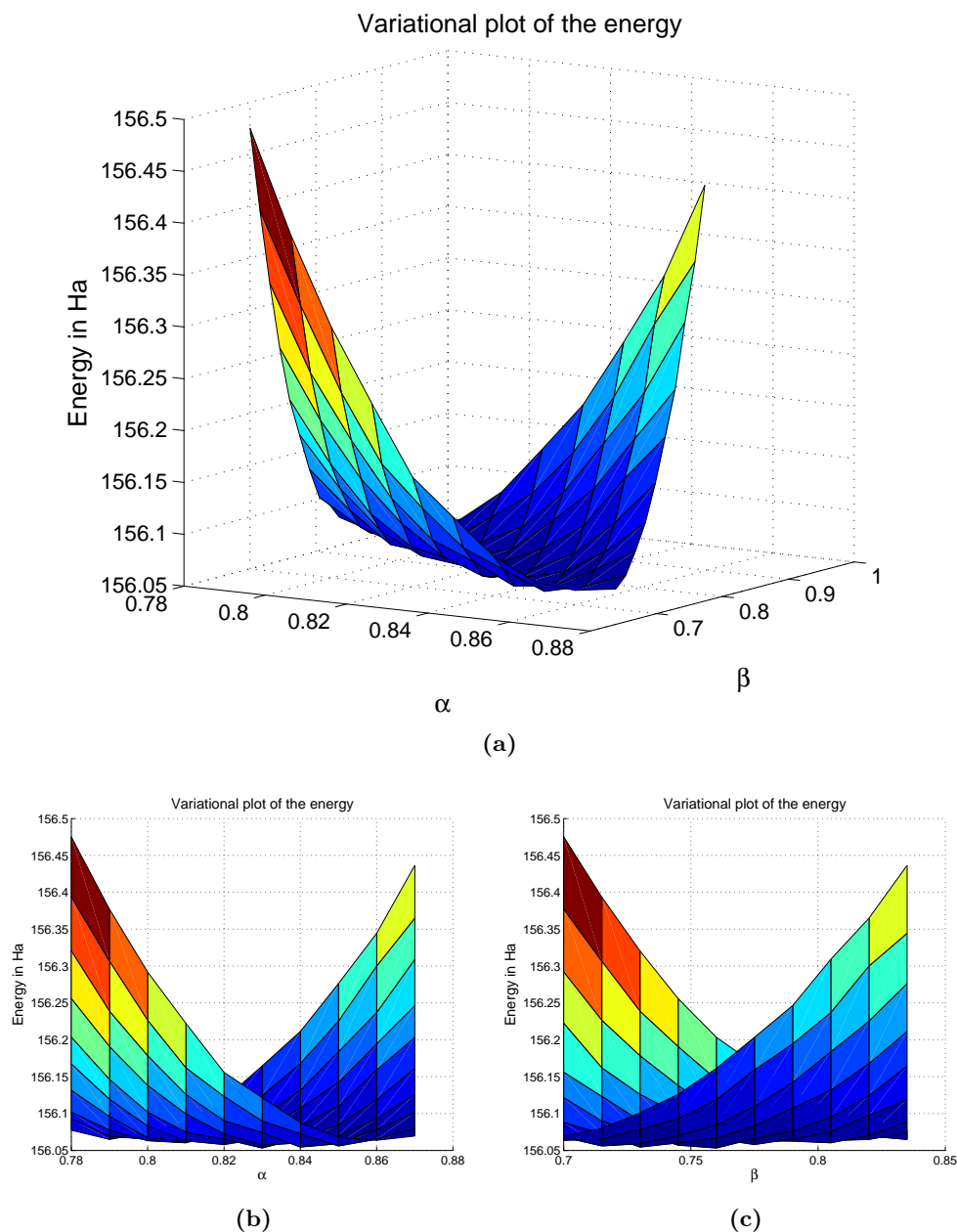


Figure 9.6: In (a) we have plotted the variational results for the energy for a twenty-electron quantum dot. The simulation was done with 10^6 MC samples. We have done 10 variations in both α and β . From (b) we read the optimal value of α , and from (c) we read the optimal value of β . We find the minimum at $\alpha \approx 0.84$ and $\beta \approx 0.76$.

Table 9.6: In this table we give the energy and error calculated with blocking for three values of Δt . In the last row we give the extrapolated result. All values correspond to a quantum dots with two electrons.

Δt	Energy (Ha)
0.01	$\langle E \rangle = 3.00026 \pm 3 \times 10^{-5}$
0.025	$\langle E \rangle = 3.00036 \pm 2 \times 10^{-5}$
0.05	$\langle E \rangle = 3.00036 \pm 2 \times 10^{-5}$
Extrapolated:	$\langle E \rangle = 3.00029 \pm 3 \times 10^{-5}$

orbitals when we construct Slater determinants. These orbitals have already been optimized, making the need of the α -parameter superfluous. We then only need to concern ourselves with the parameter β . As discussed in Section 8.1, the Slater determinant is the most costly part of the wave function to calculate. If we only need to minimize the correlation function, it would greatly increase the efficiency of the DFP algorithm.

9.3 Time-step analysis with blocking

In this section we will use the optimized parameters found in the previous sections, and do time-step extrapolation and blocking to get more correct results for the energy and the error of quantum dots with two, six, twelve and twenty electrons. As described in Sections 6.7 and 6.8, blocking is used to get a correct estimate of the error of the sample mean, and time-step extrapolations is done in order to deal with the errors due to the finite step length in time used in eq. (6.72). All calculations in this section are done with $\omega = 1$.

9.3.1 Two electrons

We use the optimal parameters from the energy minimization in Table 9.3. We used three different Δt values when calculating the energy:

$$\Delta t \in \{0.01, 0.025, 0.05\}. \quad (9.2)$$

For each calculation, we use 10^8 MC cycles. This time we store the local energy calculated in *every single* loop of the MC calculation. These values are then loaded in the program discussed in Section 8.5. The results are plotted in Figure 9.7. We clearly see the plateau discussed in Section 6.7. At this plateau, the blocks are considered to be uncorrelated, and we read off the correct value of the error. In Table 9.6 we give the energy and error for the three Δt -s we have done calculations for. We then do a linear fitting as discussed in Section 8.6. The resulting extrapolated energy and error can also be found in table 9.6.

9.3.2 Six electrons

We use the optimal parameters from Table 9.4. We did three calculations of the energy with

$$\Delta t \in \{0.01, 0.025, 0.05\},$$

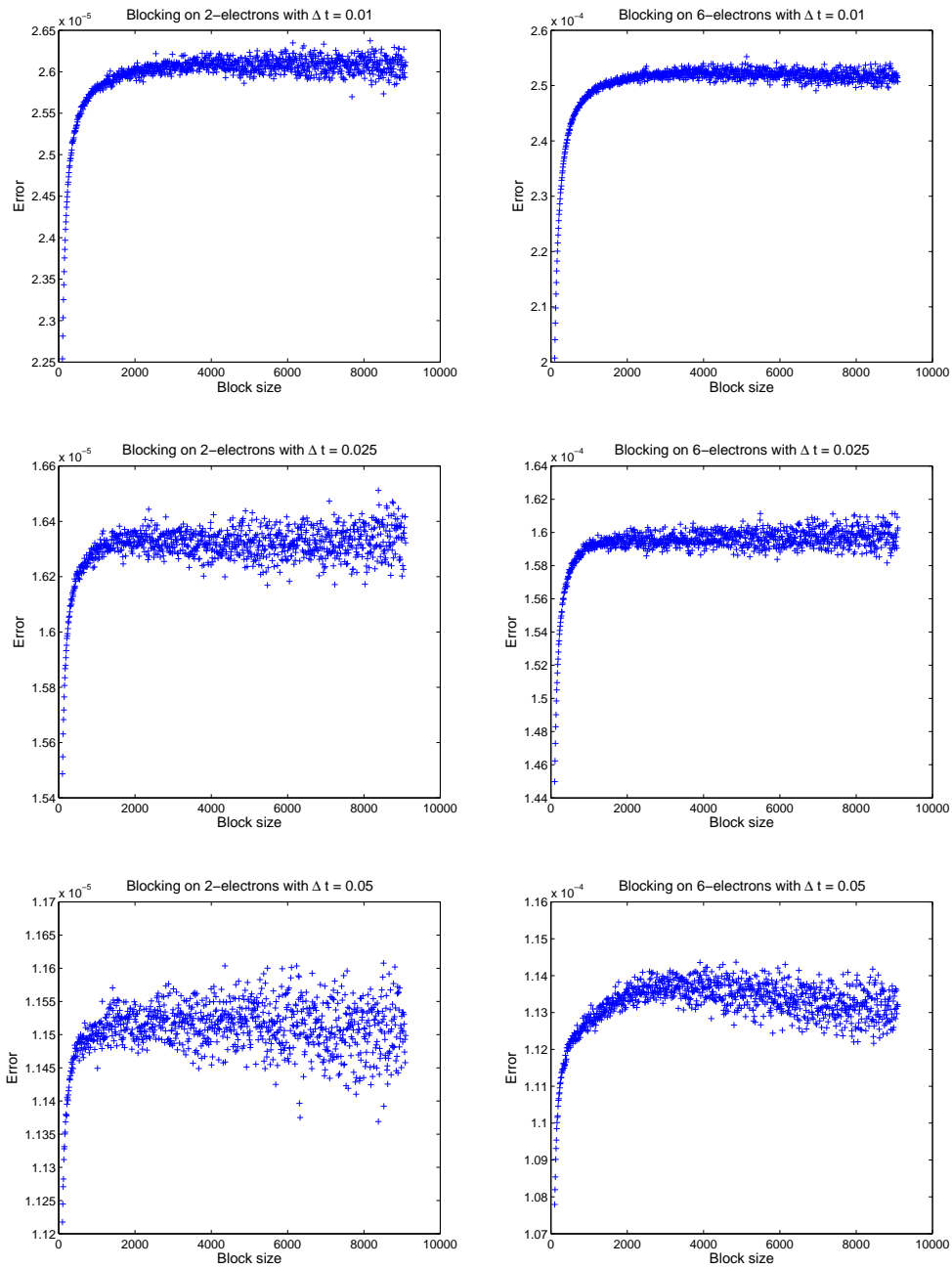


Figure 9.7: The plots show the error as a function of block size. To the left in the figure, we see the blocking plots for the two-electron quantum dot. To the right we see the plots for the six-electron quantum dot. The top two plots are for $\Delta t = 0.01$, the middle plots are for $\Delta t = 0.025$, and the bottom plots are for $\Delta t = 0.05$. The plateau discussed in section 6.7 is clearly visible. At this plateau we find the correct value for the error of the sample mean. If we study the plots, it may seem like the plot is smoother for small values of Δt . However, by studying the axes, we see that the resolution is finer for larger Δt , so the smoothness in the top plots is mostly due to a larger resolution in the y -axis.

Table 9.7: In this table we give the energy and error calculated with blocking for three values of Δt . In the last row we give the extrapolated result. All values correspond to a quantum dot with six electrons.

Δt	Energy (Ha)
0.01	$\langle E \rangle = 20.1900 \pm 3 \times 10^{-4}$
0.025	$\langle E \rangle = 20.1902 \pm 2 \times 10^{-4}$
0.05	$\langle E \rangle = 20.1904 \pm 1 \times 10^{-4}$
Extrapolated:	$\langle E \rangle = 20.1899 \pm 2 \times 10^{-4}$

and 10^8 MC cycles for each simulation. The resulting blocking plots are presented in Figure 9.7, and we present the energy and error for each Δt in Table 9.8. In Figure 9.8 we have included a plot of the energies and errors from Table 9.8 in order to give a visualisation of how the extrapolation is done.

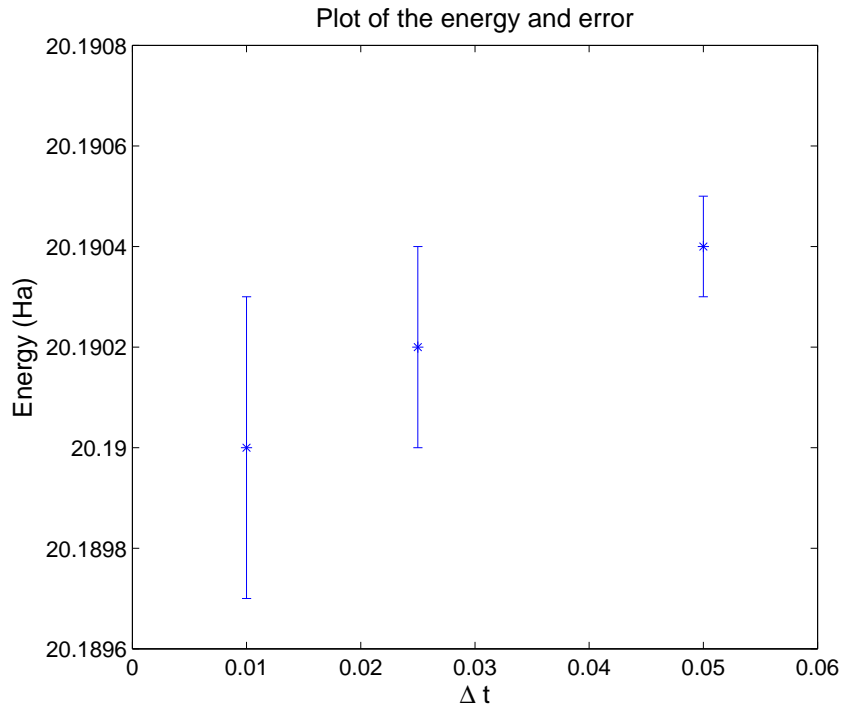


Figure 9.8: We have calculated the energy of a six-electron quantum dot using three different values of Δt . This plot shows the value of the energy for each value of Δt together with the corresponding error. The idea is then to make a linear fitting to these points, in order to extrapolate the value of the energy for $\Delta t = 0$.

9.3.3 Twelve electrons

The same procedure is repeated for the twelve-electron system. We use the optimal parameters from Table 9.6. We calculate the energy for Δt values:

$$\Delta t \in \{0.01, 0.025, 0.05\},$$

Table 9.8: In this table we give the energy and error calculated with blocking for three values of Δt . In the last row we give the extrapolated result. All values correspond to a quantum dot with twelve electrons.

Δt	Energy (Ha)
0.01	$\langle E \rangle = 65.7915 \pm 4 \times 10^{-04}$
0.025	$\langle E \rangle = 65.7909 \pm 3 \times 10^{-04}$
0.05	$\langle E \rangle = 65.7901 \pm 2 \times 10^{-04}$
Extrapolated:	$\langle E \rangle = 65.7918 \pm 4 \times 10^{-4}$

Table 9.9: In this table we give the energy and error calculated with blocking for three values of Δt . In the last row we give the extrapolated result. All values correspond to a quantum dot with twenty electrons.

Δt	Energy (Ha)
0.00	$\langle E \rangle = 156.0726 \pm 7 \times 10^{-04}$
0.025	$\langle E \rangle = 156.0695 \pm 4 \times 10^{-04}$
0.05	$\langle E \rangle = 156.0667 \pm 3 \times 10^{-04}$
Extrapolated:	$\langle E \rangle = 156.0732 \pm 6 \times 10^{-4}$

For each energy calculations, we use 10^8 MC cycles. The plots are presented in Figure 9.9. Once again we see that the error reaches a plateau. We present the calculated energy and error in Table 9.9, together with the extrapolated result.

9.3.4 Twenty electrons

We use the optimal parameters from Table 9.6, and calculate the energy for

$$\Delta t \in \{0.01, 0.025, 0.05\},$$

using 10^8 MC cycles for each calculation. The resulting blocking plots are presented in Figure 9.9. The plateau value of the error is found, and we present the results in Table 9.9.

9.4 Changing the oscillator frequency

We have also done some calculations for oscillator frequencies different from $\omega = 1$. The strategy is as before: We start by finding the optimal variational parameters, then do blocking and time-step extrapolation to find the energy and error. The results are presented in Table 9.10. Each calculation of the energy is done with a total of 10^8 MC cycles.

9.5 Splitting up the energy

We want to see how large the different contributions to the energies are. We can split the total energy in three parts:

$$\langle E_{tot} \rangle = \langle E_{kin} \rangle + \langle E_{osc} \rangle + \langle E_{ele} \rangle, \quad (9.3)$$

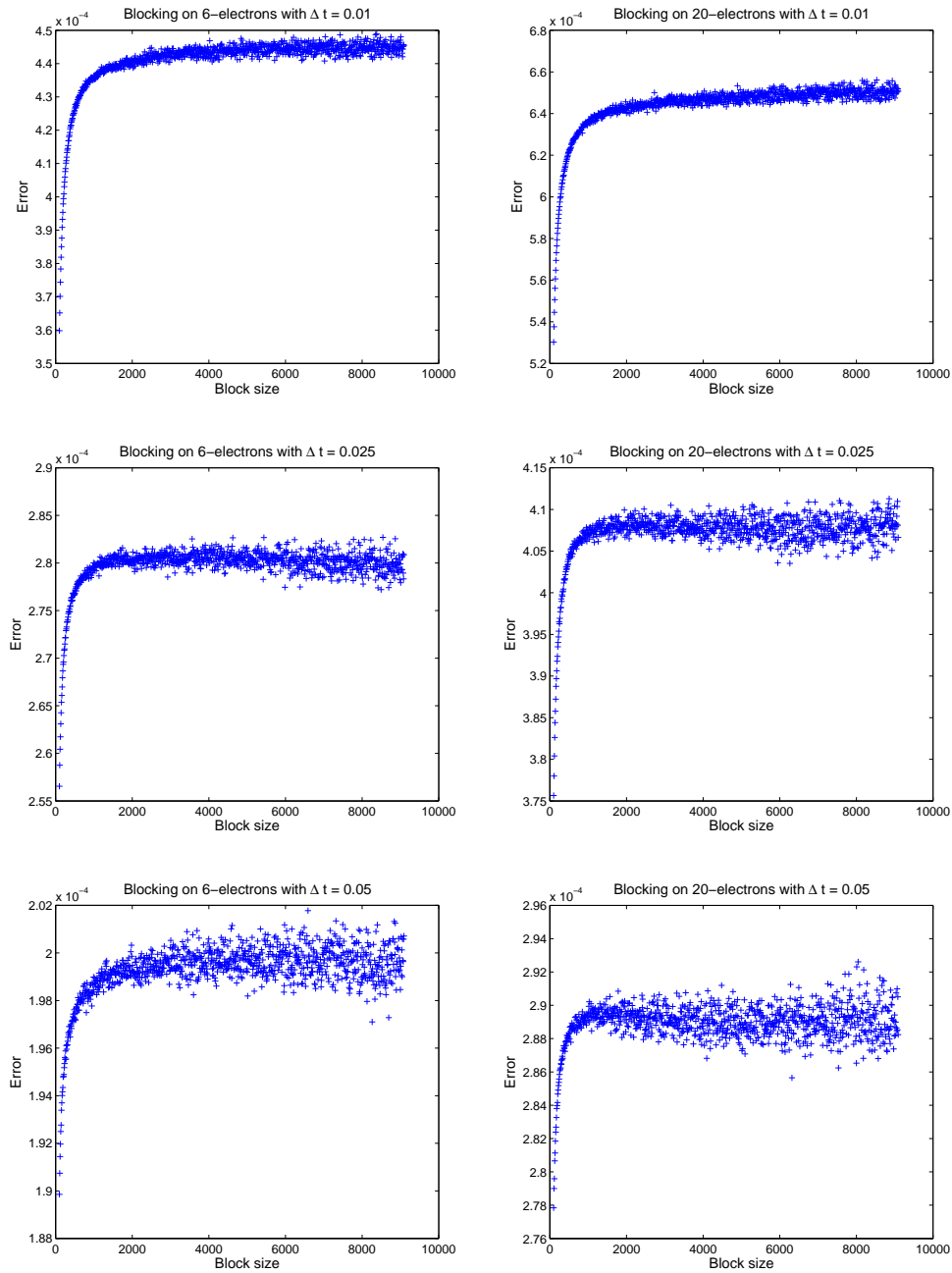


Figure 9.9: The plots show the error as a function of block size. To the left in the figure, we see the blocking plots for the twelve-electron quantum dot. To the right we see the plots for the twenty-electron quantum dot. The top two plots are for $\Delta t = 0.01$, the middle plots are for $\Delta t = 0.025$, and the bottom plots are for $\Delta t = 0.05$. The plateau discussed in section 6.7 is clearly visible. At this plateau we find the correct value for the error of the sample mean. The apparent differences in smoothness is only due to different resolutions on the y -axis.

Table 9.10: In this table we present the optimal parameters and energy found for two-, six-, twelve- and twenty-electron quantum dots calculated for two different values of the oscillator frequency.

N	ω	α	β	Energy (Ha)
2	0.28	0.975	0.25	$\langle E \rangle = 1.02212 \pm 3 \times 10^{-5}$
	0.5	0.98	0.31	$\langle E \rangle = 1.66027 \pm 3 \times 10^{-5}$
6	0.28	0.88	0.33	$\langle E \rangle = 7.6223 \pm 2 \times 10^{-4}$
	0.5	0.90	0.42	$\langle E \rangle = 11.8104 \pm 2 \times 10^{-4}$
12	0.28	0.805	0.38	$\langle E \rangle = 25.6989 \pm 4 \times 10^{-4}$
	0.5	0.84	0.48	$\langle E \rangle = 39.2359 \pm 4 \times 10^{-4}$
20	0.28	0.77	0.42	$\langle E \rangle = 62.0567 \pm 6 \times 10^{-4}$
	0.5	0.81	0.54	$\langle E \rangle = 94.0308 \pm 6 \times 10^{-4}$

where we have defined:

$$\langle E_{kin} \rangle = \left\langle \sum_i \frac{1}{2} \nabla_i^2 \right\rangle, \quad (9.4)$$

$$\langle E_{harm} \rangle = \left\langle \sum_i \frac{1}{2} \omega^2 (x_i^2 + y_i^2) \right\rangle, \quad (9.5)$$

$$\langle E_{ele} \rangle = \left\langle \sum_{i < j} \frac{1}{r_{ij}} \right\rangle. \quad (9.6)$$

We then have that $\langle E_{kin} \rangle$ is related to the kinetic energy, $\langle E_{harm} \rangle$ is the energy contribution from the confining potential (in our case a harmonic oscillator potential), and $\langle E_{ele} \rangle$ is the energy from the Coulomb repulsion between electrons. We want to calculate the expectation values of each of these energies separately. This is done for the same oscillator frequencies used before, and we use the optimal parameters found in Tables 9.3, 9.4 and 9.10 for the respective systems. The results are presented in Table 9.11.

9.6 Discussions of the results

We first consider the results in Table 9.11. Varying ω — the oscillator frequency — is equivalent to tuning the confining potential. This means the particles in the dot will become more confined when we increase ω , and less confined when we decrease ω . From this reasoning, it seems a bit strange that the role of the energy from the $1/r_{ij}$ term in the Hamiltonian increases relatively as ω decreases. We shall see why this is so. Obviously, the oscillator energy is proportional to ω^2 , so it is not strange at all that this energy decreases relatively with ω . From eq. (7.32), we know that the Laplacian of the single-particle orbitals is proportional to ω . This means that the kinetic energy of the determinantal part of Ψ_T will also be proportional to ω , and it follows that this will decrease with ω . Only the Coulomb energy from the electron-electron repulsion is not dependent on ω (it will, of course, be changed a bit by the fact that the inter-electron distances are changed, but there is no explicit dependency).

Table 9.11: In this table we present the results from calculating the expectation value of each part of the total energy separately. All energies are in units of Hartrees. We have used $\Delta t = 0.05$ in all calculations. In the parentheses behind the numbers we have included what percentage the relevant energy is of the total energy.

N	ω	E_{tot}	E_{kin}	E_{osc}	E_{ele}
2	0.28	1.0222	0.2531 (24.8%)	0.4235 (41.4%)	0.3456 (33.8%)
	0.5	1.6602	0.4466 (26.9%)	0.7008 (42.2%)	0.5130 (30.9%)
	1	3.0004	0.8908 (29.7%)	1.2955 (43.2%)	0.8141 (27.1%)
6	0.28	7.6218	0.9441 (12.4%)	2.8109 (36.9%)	3.8668 (50.7%)
	0.5	11.8102	1.7213 (14.6%)	4.4772 (37.9%)	5.6117 (47.5%)
	1	20.1902	3.5831 (17.8%)	7.9234 (39.2%)	8.6837 (43.0%)
12	0.28	25.7003	2.1780 (8.5%)	9.3232 (36.3%)	14.1991 (55.2%)
	0.5	39.2354	4.0757 (10.4%)	14.5713 (37.1%)	20.5884 (52.5%)
	1	65.7906	8.8318 (13.4%)	24.7721 (37.7%)	32.1867 (48.9%)
20	0.28	62.055	4.175 (6.7%)	21.804 (35.1%)	36.076 (58.1%)
	0.5	94.028	7.976 (8.5%)	33.519 (35.6%)	52.533 (55.9%)
	1	156.068	17.416 (11.2%)	56.995 (36.5%)	81.657 (52.3%)

When ω decreases, interactions get more and more important, and the kinetic energy diminishes. This means that $\langle p \rangle$ can be determined very sharply. The uncertainty relation tells us that $\langle r \rangle$ (the expectation value of the position) must then increase. This is something we expect as the particles get less and less confined when ω decreases.

We can also draw some conclusions from the optimal parameters of the trial wave functions of the respective systems. The parameter β is associated with the Jastrow factor in eq. (7.35). When this parameter is small, it means that the Jastrow factor becomes large, and it follows that the correlation part of the wave function is more important. Studying Table 9.4, we see that the parameter β increases as N increases. This means that the Jastrow part of our trial wave function becomes less important for systems with a higher number of particles, even though the energy from the electron-electron interaction increases because there are more particles which interact with each other. This is consistent with the findings in ref. [46] (see Fig. 2 in the article). We also see, from table 9.11, that the value of β gets smaller for a system when ω decreases. This supports our observation that correlations are more important when ω is small.

So far we have no idea if the results we have obtained make any sense, except for the results in Section 9.1, which seems to be quite good. In ref. [46] we find DMC (Diffusion Monte Carlo) and Coupled-Cluster calculations on the same systems we consider. The DMC method can in principle produce results that are as close to the exact as possible within the limits of computational resources (ref. [36, 39]). We will therefore treat the DMC results from ref. [46] as the exact ground state energies for the systems we consider. In Table 9.12 we present the results from ref. [46] together with our results. We see that our results are in very good agreement with the DMC results that are assumed to be exact. This is a clear indication that the Jastrow-Slater trial wave function in eq. (7.66) is a good trial wave function for a closed-shell system. Furthermore, we see that using only one Slater determinant in our Ψ_T gives us very good results. This indicates that particle-hole

Table 9.12: In this table we present our results for the energies for different N and ω , together with the DMC results from ref. [46]. The DMC results are considered to be very close to exact (for example, we see that the DMC results for $N = 2$, $\omega = 1$ is a reproduction of Tauts (ref. [34]) analytical results). Our calculations are in very good agreement with the DMC results — the different is in the third or fourth digit. All the energies in the table are in units of Hartrees.

N	ω	QVMC Energy	DMC Energy (ref. [46])	Relative difference
2	0.5	$\langle E \rangle = 1.66027(3)$	1.65975(2)	0.03%
	1	$\langle E \rangle = 3.00029(3)$	3.00000(3)	0.01%
6	0.28	$\langle E \rangle = 7.6223(2)$	7.6001(1)	0.3%
	0.5	$\langle E \rangle = 11.8104(2)$	11.7888(2)	0.2%
	1	$\langle E \rangle = 20.1899(2)$	20.1597(2)	0.2%
12	0.28	$\langle E \rangle = 25.6989(4)$	25.6536(1)	0.2%
	0.5	$\langle E \rangle = 39.2359(4)$	39.159(1)	0.2%
	1	$\langle E \rangle = 65.7918(4)$	65.700(1)	0.1%
20	0.28	$\langle E \rangle = 62.0567(6)$	61.922(2)	0.2%
	0.5	$\langle E \rangle = 94.0308(6)$	93.867(3)	0.2%
	1	$\langle E \rangle = 156.0732(6)$	155.868(6)	0.2%

excitations in a closed shell system are not that important. The Pade-Jastrow correlation function is able to account for most of the correlations in the systems considered.

Even though it is hard to say for sure, it seems that the relative difference between our results and the DMC results get larger for smaller ω . This indicates that when the correlations get more important, our simple approximation of the trial wave function will not be ideal. When ω decreases, we should use a combination of Slater determinants, allowing the particles to be excited to higher orbitals. Furthermore, we could try implementing more variational parameters in the correlation function. This could be done by using more terms in the expansion in equation eq. (7.34). However, for the values of ω that we have done calculations for, it seems that our Ψ_T is a very good approximation, indeed. It would be interesting, though, to find out where our approximation breaks down, but we have not had time to do so in this work.

From our discussion in Section 5.1, we know that when the B -field increases and ω_0 is held constant, we split the degeneracies in our single-particle energies. This means that the shell-structure is broken. When ω_B gets large enough, our chosen single-particle orbitals are no longer viable; we need to use the orbitals that form the different Landau bands in order to create a good Slater determinant. We have not studied this in our thesis.

9.7 Quantum dots with more than twenty electrons

We want to show that the code we have developed in this thesis is able to simulate systems with more than twenty particles. One problem with making a code that is general enough to take on any closed-shell system, is that we do not have an algorithm to assign quantum numbers to each of the single-particle states. To solve this we have created a table in the orbital class that assigns quantum numbers to each state as the orbitals are

Table 9.13: We present the energy and optimal parameters found for 30- and 42-electron quantum dot. The simulation was done with very few MC cycles. Still, we see that our code is able to handle systems with many particles.

N	α	β	Energy (Ha)	#MC cycles
30	0.78	0.85	308.8	100 000
42	0.8	0.95	543.4	100 000

initiated in the code. This means that we need to fill in this table when we move to systems with higher numbers of particles.

Even for twenty particles, the computation takes a lot of resources and uses a lot of time. Unfortunately, we did not have time to do any large computation on systems larger than twenty particles, but we have done small calculations on 30- and 42-electron quantum dots, to show that we get good results for these systems. We did calculations only for $\omega = 1$. The results are tabulated in Table 9.13. We see the trend from before is still present. The value of β still increases as we increase N , which means the Jastrow part of the wave function is less important.

As far as we know, there has not been done any *ab initio* calculations on systems with more than 20 particles. We therefore do not have any grounds for comparison of the energy. Still, considering the accuracy of our calculations on smaller systems, we do not have any reason to believe that these results are very wrong.

Chapter 10

Conclusion

In this thesis we have done numerical studies of systems consisting of several interacting electrons in two dimensions. To be more specific, we have studied closed-shell quantum dots. The model we have used is the so-called parabolic (or circular) quantum dot. In this approximation, the confining potential is modelled by the harmonic oscillator potential, and the electron-electron interaction is given by the standard Coulomb interaction. The goal of the thesis was to develop a Quantum Variational Monte Carlo (QVMC) code in C++ that can approximate the ground state energy of such closed-shell quantum dots, using a Jastrow-Slater trial wave function. Furthermore we wanted to see how well the system can be described using only one Slater determinant.

The program developed can in principle easily be changed to handle other kinds of electronic system, such as atoms, molecules and other quantum-dot models.

We have tried two different minimization techniques. The first one was a brute-force approach, changing the variational parameters manually and calculating the energy for several different sets of variational parameters. The second technique was a Davidon-Fletcher-Powell (DFP) minimization which searches for a minimum in a more intelligent fashion. We found that the DFP approach works well for small systems, but is not well suited when N (the number of particles) increases to above $N = 12$.

Our studies of the parabolic quantum dots have been limited to closed-shell systems only. We have studied systems containing two, six, twelve and twenty electrons in detail, calculating the ground state energy for oscillator frequencies $\omega \in \{0.28, 0.5, 1\}$. We found that our trial wave function yields results very close to the (assumed) correct DMC results in ref. [46]. This indicates that a Jastrow-Slater wave function is a good approximation for our system. Furthermore, the approximation using only a single Slater determinant — the determinant representing the ground state of the non-interacting system — seems to be a good one. We have found that by decreasing the oscillator frequency, we effectively increase the importance of correlations in the system. Furthermore, we have found that correlations are not as important in large systems compared to smaller systems. Comparison of the relative differences between our results and the DMC results in Table 9.12 indicates that at some point, when ω gets small enough, we should include more Slater determinants in our trial wave function, or expand the Jastrow function. In this thesis, we have not studied this breaking point of our model. Furthermore, we have not studied the high-field limit, when the single-particle orbitals form Landau bands (as discussed

in Section 5.1). To study this limit, we need to change which single-particle orbitals we use to build our Slater determinant (the ground state of the non-interacting system will change). Probably, we would need to include several Slater determinants for such systems.

Our code is able to simulate even larger systems than twenty particles, though it is quite time consuming. We have obtained results of the ground state energy for 30- and 42-electron quantum dots. As far as we know, there has not been done any *ab initio* calculations on these systems. It is therefore not given that our results for these systems are any good, but because of our excellent results for smaller systems, we have no reason to believe that the results for these larger systems should be far off.

The sources of error in our calculation stems from the fact that our trial wave function is not the correct wave-function the systems under consideration. We could decrease the error by including more Slater determinants, and using more terms in the expansion in eq. (7.34). In general, though, we will never be able to represent the exact wave function in this way.

Future work

Our code has been constructed to only work for full shell systems. We could, however, introduce more Slater determinant to Ψ_T . Then we would also be able to study open-shell systems (systems where the outer shell is not full).

Another interesting addition would be to improve the single-particle basis. This could be done by first doing a Hartree-Fock (HF) calculation of the quantum dots and then parametrize the results in Gaussian-type orbitals. The approach cancels the need for a variational parameter in the Slater determinant — The determinant is already optimized through the HF calculations. Furthermore, this approach introduces correlation effects in the Slater determinant; the Gaussian type orbitals are typically linear combinations of orbitals from different shells, which means that even though we use only one Slater determinant, we have included the possibility of particle-hole excitations. This approach could make the DFP algorithm more efficient, since we no longer need to optimize the Slater determinant.

It would also be interesting to check where our Ψ_T breaks down, either by making ω very small, which means the correlations becomes very large, or by increasing the B -field so that the shell structure of our system is broken.

The code could in principle also be used for studying atoms and molecules, and it would be interesting to implement this. This should not be a problem, we only need to change the confining potential (for example, in atoms we need to include the Coulomb interaction between the electrons and the nucleus), and change the basis of the single-particle orbitals (in atoms, one uses normally hydrogen-like orbitals).

Finally, we would like to use a minimization technique that works well for a high number of particles. The DFP algorithm is not well suited for this, and the brute force approach by manually changing the parameters is clearly not optimal. The Stochastic-Gradient-Approximation (SGA) method (ref. [47]) is one possible approach. This method

utilizes the noise inherent in stochastic functions, and is very well suited for a stochastic method such as QVMC.

Appendix A

Statistics

VMC is an inherently statistical method, and it follows that we need some grasp of statistics in order to correctly analyze our results. In this appendix we will give a short introduction to many key aspects regarding the statistical terms used for calculating expectation values and its statistical error in the VMC process. As the nomenclature may differ from one textbook to another, it is convenient to establish a nomenclature suitable for our purpose.

A stochastic variable is for our purposes just the same as a random variable. It can be either continuous or discrete. We will denote stochastic variables by capital letters $X, Y, ..$

We define the *domain* of a stochastic variable as the set of all accessible values the variable can assume. In the discrete case an example of a domain is

$$\mathbb{D} = \{1, 2, 3, 4, 5\}. \quad (\text{A.1})$$

This domain specifies all the possible values of the stochastic variables X , so that $X \in \mathbb{D}$. In the continuous case we could for example have

$$\mathbb{D} = [a, b], \quad (\text{A.2})$$

and we must have that $X \in \mathbb{D}$.

The *probability density function* (PDF) is a function $\rho(x)$ on the domain that describes the possibility of all possible events. In the discrete case, the probability that a specific value of X occurs is given as

$$\text{Prob}(X = x) = \rho(x). \quad (\text{A.3})$$

In the continuous case, the PDF is a probability density. The probabilities are then related to $\rho(x)dx$. In other words, the probability for the stochastic variable X taking a value on the finite interval $[a, b]$ is defined by an integral

$$\text{Prob}(a \leq X \leq b) = \int_a^b \rho(x)dx \quad (\text{A.4})$$

There are certain properties a PDF must have. We have summed up the most important ones in Table A.1 (ref. [38]).

In our work, there are two important PDFs:.

Table A.1: Properties of PDFs.

	Discrete PDF	Continuous PDF
Domain	$\{x_1, x_2, \dots, x_N\}$	$[a, b]$
Probability	$\rho(x_i)$	$\rho(x)dx$
Cumulative	$P_i = \sum_{l=1}^i \rho(x_l)$	$P(x) = \int_a^x \rho(t)dt$
Positivity	$0 \leq \rho(x_i)$	$0 \leq \rho(x)$
Monotonic	$P_i \geq P_j$ if $x_i \geq x_j$	$P(x_i) \geq P(x_j)$ if $x_i \geq x_j$
Normalization	$P_N = 1$	$P(b) = 1$

1. The first PDF is the most basic one; the uniform distribution

$$\rho(x) = \frac{1}{b-a} H(x-a)H(b-x), \quad (\text{A.5})$$

where H is the Heaviside step function, defined as

$$H(x) = \begin{cases} 0, & x < 0, \\ 1, & x \geq 0. \end{cases} \quad (\text{A.6})$$

For our purposes, we use $a = 0$ and $b = 1$ in the uniform distribution.

2. The second PDF is the Gaussian distribution, also called the normal distribution. It is defined as

$$\rho(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (\text{A.7})$$

where μ and σ^2 is the mean and the variance of the PDF, respectively. These terms will be discussed presently.

Both of these PDFs are continuous, and in the following discussion we will assume continuous PDFs.

Let $f(x)$ be an arbitrary function on the domain of the stochastic variable X whose PDF is $\rho(x)$. The *expectation value* of f with respects to ρ is defined as

$$\langle f \rangle_X \equiv \int_{\mathbb{D}} f(x)\rho(x)dx, \quad (\text{A.8})$$

where $\int_{\mathbb{D}}$ means integrating over the whole domain of X . Whenever the PDF is known implicitly, we suppress the subscript X . An important special case of expectation values is what we call *moments*. The n -th moment is defined as

$$\langle x^n \rangle \equiv \int_{\mathbb{D}} x^n \rho(x)dx. \quad (\text{A.9})$$

The zero-th moment is just the normalization requirement of ρ . The first moment, $\langle x \rangle$, is called the mean of ρ . It is often denoted by μ :

$$\langle x \rangle = \mu_X = \int_{\mathbb{D}} x\rho(x)dx. \quad (\text{A.10})$$

Qualitatively, the mean represents the average value of the PDF on the domain. It is therefore usually called the expectation value of ρ .

A special version of the moments is the set of *central moments*. The n -th central moment is defined as

$$\langle (x - \langle x \rangle)^n \rangle \equiv \int_{\mathbb{D}} (x - \langle x \rangle)^n \rho(x) dx. \quad (\text{A.11})$$

The zero-th and first central moments are both trivial. They are equal to 1 and 0, respectively. The second central moment, though, is of great importance. It is known as the *variance* of ρ . For the stochastic variable X , the variance is usually denoted as σ_X^2 or $\text{Var}(X)$:

$$\begin{aligned} \sigma_X^2 = \text{Var}(X) &= \int_{\mathbb{D}} (x - \langle x \rangle)^2 \rho(x) dx \\ &= \int_{\mathbb{D}} (x^2 - 2x \langle x \rangle + \langle x \rangle^2) \rho(x) dx \\ &= \langle x^2 \rangle - 2 \langle x \rangle \langle x \rangle + \langle x \rangle^2 \\ &= \langle x^2 \rangle - \langle x \rangle^2. \end{aligned} \quad (\text{A.12})$$

The square root of the variance is called the *standard deviation* of ρ , denoted by σ . It is qualitatively interpreted as the spread of ρ around its mean.

The PDFs discussed above are called *univariate*, meaning they are functions of only one stochastic variable. A general PDF may consist of any number of variables, in which case it is called *multivariate*. Let $P(x_1, \dots, x_n)$ be the multivariate PDF for the set $\{X_i\}$ of n stochastic variables, and let $F(x_1, \dots, x_n)$ be an arbitrary function over the joint domain of all X_i . The expectation value of F with respects to P is then defined as

$$\langle F \rangle_{X_1, \dots, X_n} = \int \dots \int F(x_1, \dots, x_n) P(x_1, \dots, x_n) dx_1 \dots dx_n. \quad (\text{A.13})$$

If we want to find the expectation value of an arbitrary function $f(x_i)$ on the domain of just one of the stochastic variables X_i , we still need to integrate over the total domain:

$$\langle f \rangle_{X_i} = \int \dots \int f(x_i) P(x_1, \dots, x_n) dx_1 \dots dx_n. \quad (\text{A.14})$$

The different stochastic variables may depend upon each other, in which case they are called *correlated*. We define that the variables are *uncorrelated* or independent if the PDF P can be factorized as

$$P(x_1, \dots, x_n) = \prod_{i=1}^n \rho(x_i). \quad (\text{A.15})$$

where $\rho(x_i)$ is the univariate PDF of X_i . We observe that if all X_i are uncorrelated, eq. (A.14) reduces to the simple form of eq. (A.8). The concept of correlation is very important for our MC calculations as.

A variant of the above variance is what we call the *covariance*. Consider a set of n stochastic variables, $\{X_i\}$, with the multivariate PDF $P(x_1, \dots, x_n)$. The covariance of the two stochastic variables X_i and X_j is defined as

$$\begin{aligned} \text{Cov}(X_i, X_j) &= \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle \\ &= \langle x_i x_j - x_i \langle x_j \rangle - \langle x_i \rangle x_j + \langle x_i \rangle \langle x_j \rangle \rangle \\ &= \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle - \langle x_i \rangle \langle x_j \rangle + \langle x_i \rangle \langle x_j \rangle \\ &= \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle. \end{aligned} \quad (\text{A.16})$$

Its easy to see that if $i = j$, the covariance reduces to the variance. If the variables X_i and X_j are uncorrelated, the PDFs factorize as in eq. (A.15), and we will have that $\langle x_i x_j \rangle = \langle x_i \rangle \langle x_j \rangle$ (assuming $i \neq j$), which means that $\text{Cov}(X_i, X_j) = 0$.

We will now consider the covariance of linear combinations of stochastic variables. Let $\{X_i\}$ and $\{Y_j\}$ be two sets of stochastic variables, and let $\{a_i\}$ and $\{b_j\}$ be two sets of scalars. We consider the linear combinations

$$U = \sum_i a_i X_i, \quad V = \sum_j b_j X_j. \quad (\text{A.17})$$

The covariance between these two linear combinations will be (ref. [38])

$$\text{Cov}(U, V) = \sum_{i,j} a_i b_j \text{Cov}(X_i, Y_j), \quad (\text{A.18})$$

and

$$\text{Var}(U) = \text{Cov}(U, U) = \sum_{i,j} a_i a_j \text{Cov}(X_i, X_j). \quad (\text{A.19})$$

In the special case where the stochastic variables are uncorrelated, the off-diagonal (we may consider The covariance as a matrix C_{ij}) terms are zero, which results in

$$\text{Var}(U) = \sum_i a_i^2 \text{Cov}(X_i, X_i) = \sum_i a_i^2 \text{Var}(X_i). \quad (\text{A.20})$$

Let us now try to apply this on empirical observations. Consider a *stochastic process* that produces sequentially a chain of values $\{x_1, x_2, \dots, x_i, \dots\}$. These values are created by pseudo-random generators (discussed in Section 6.1). We will call each value a *measurement*, and the entire set is called our measured *sample*. The action of measuring all the elements of a sample is called a stochastic *experiment* (since they are often associated with results of some observations; in our case the measurement of the energy of a quantum mechanical system). We assume that our measurements are distributed according to some PDF $\rho_X(x)$, where X is just the formal symbol for the stochastic variables whose PDF is $\rho_X(x)$. Instead of searching for this PDF, we are usually only looking for the mean μ_X and variance σ_X^2 .

In practical situations a sample is always of finite size. Let that size be n . The mean of the sample is then defined as

$$\bar{x}_n \equiv \frac{1}{n} \sum_{i=1}^n x_i, \quad (\text{A.21})$$

and the sample variance is

$$\text{Var}(x) \equiv \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_n)^2, \quad (\text{A.22})$$

and the sample covariance

$$\text{Cov}(x) \equiv \frac{1}{n} \sum_{i,j} (x_i - \bar{x}_n)(x_j - \bar{x}_n). \quad (\text{A.23})$$

This covariance is a measure of the correlation between succeeding measurements of a sample. The above quantities for mean, variance and covariance must *not* be confused

with the similarly named quantities for stochastic variables!

It can be shown that in the limit $n \rightarrow \infty$, the sample mean approaches the true mean μ_X of the chosen PDF (ref. [38])

$$\lim_{n \rightarrow \infty} \bar{x}_n = \mu_X, \quad (\text{A.24})$$

and \bar{x}_n can be seen as an estimate of μ_X . The question is then: How good is this approximation? To answer this, we need the error estimate of our mean; this is a quantity that tells us how well we can reproduce the mean if we did another experiment. We are therefore interested in the PDF of the sample mean itself. Its standard deviation will be a measure of the spread of sample means, and we will simply call it the *error* of the sample mean, or simply sample error, denoted err_X . In practice, we will only be able to produce an estimate of err_X , since the exact value would require knowledge of the true PDF, which we usually do not have.

In a sample, each of the measurements x_i can be associated with its own stochastic variable X_i . The sample mean is then also associated with a stochastic variable

$$\bar{X}_n = \frac{1}{n} \sum_i X_i. \quad (\text{A.25})$$

This is similar to the earlier discussed linear combinations of stochastic variables, now with $1/n$ as a common coefficients. The PDF of \bar{X}_n , denoted $\rho_{\bar{X}_n}(x)$, is the desired PDF of the sample means. It is generally not possible to express this PDF in closed form, but in the limit $n \rightarrow \infty$, it is possible to make an approximation. This result is called the *central limit theorem* (ref. [38]), which tells us that as n goes to infinity, $\rho_{\bar{X}_n}(x)$ approaches a Gaussian distribution:

$$\lim_{n \rightarrow \infty} \rho_{\bar{X}_n}(x) = \sqrt{\frac{1}{2\pi(\text{Var}(x)/n)}} e^{-\frac{n(x-\mu_X)^2}{2(\text{Var}(x)/n)}}. \quad (\text{A.26})$$

For a proof, we refer to ref. [38]. This PDF will have a mean and variance that equals the true mean and variance μ_X and σ_X^2 .

The error is now given by

$$\text{err}_X^2 = \text{Var}(\bar{X}_n) = \frac{1}{n^2} \sum_{i,j}^n \text{Cov}(X_i, X_j). \quad (\text{A.27})$$

To find the *exact* error, we need the true mean. We would need the exact PDFs of the variables X_i to obtain this, however, we have only the measurements in one sample. The best we can do is to let the sample itself be an estimate of the PDF of each of the X_i . Our estimate of μ_{X_i} is then the sample mean \bar{x}_n itself. Instead, we estimate the true mean by the sample mean:

$$\mu_{X_i} = \langle x_i \rangle \approx \frac{1}{n} \sum_{k=1}^n x_k = \bar{x}_n. \quad (\text{A.28})$$

We can then give an estimate of the covariance in eq. (A.27):

$$\begin{aligned}
 \text{Cov}(X_i, X_j) &= \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle \approx \langle (x_i - \bar{x}_n)(x_j - \bar{x}_n) \rangle \\
 &\approx \frac{1}{n} \sum_l \left(\frac{1}{n} \sum_k (x_k - \bar{x}_n)(x_l - \bar{x}_n) \right) \approx \frac{1}{n} \frac{1}{n} \sum_{k,l} (x_k - \bar{x}_n)(x_l - \bar{x}_n) \\
 &= \frac{1}{n} \text{Cov}(x)
 \end{aligned} \tag{A.29}$$

By the same procedure, we can estimate the variance of any of the stochastic variables X_i as

$$\begin{aligned}
 \text{Var}(X_i) &= \langle x_i - \langle x_i \rangle \rangle \approx \langle x_i - \bar{x}_n \rangle \\
 &\approx \frac{1}{n} \sum_{k=1}^n (x_k - \bar{x}_n) \\
 &= \text{Var}(x)
 \end{aligned} \tag{A.30}$$

We now rewrite eq. (A.27) as

$$\begin{aligned}
 \text{err}_X^2 &\approx \frac{1}{n^2} \sum_{i,j} \frac{1}{n} \text{Cov}(x) \\
 &= \frac{1}{n^2} n^2 \frac{1}{n} \text{Cov}(x) \\
 &= \frac{1}{n} \text{Cov}(x).
 \end{aligned} \tag{A.31}$$

Or, in the special case where the measurements in the sample are uncorrelated, we get

$$\begin{aligned}
 \text{err}_X^2 &= \frac{1}{n^2} \sum_{i,j} \text{Cov}(X_i, X_j) \\
 &= \frac{1}{n^2} \sum_i \text{Var}(X_i) = \frac{1}{n^2} \sum_i \text{Var}(x) \\
 &= \frac{1}{n^2} n \text{Var}(x) = \frac{1}{n} \text{Var}(x)
 \end{aligned} \tag{A.32}$$

The error of the sample is then just its standard deviation divided by the square root of the number of measurements in the sample. This is a useful and easy algorithm that works well as a first approximation of the error, but in most experiments, we cannot completely ignore the correlations.

It is convenient to rewrite eq. (A.31) as

$$\begin{aligned}
 \text{err}_X^2 &\approx \frac{1}{n} \text{Cov}(x) = \frac{1}{n} \text{Var}(x) + \frac{1}{n} (\text{Cov}(x) - \text{Var}(x)) \\
 &= \frac{1}{n^2} \sum_{k=1}^n (x_k - \bar{x}_n)^2 + \frac{2}{n^2} \sum_{k < l} (x_k - \bar{x}_n)(x_l - \bar{x}_n).
 \end{aligned} \tag{A.33}$$

The first term is the uncorrelated error. This means the second term accounts for the error correction due to correlations in the measurements. For uncorrelated measurements, this term is zero.

Computationally, the first term in eq. (A.33) is very easy to obtain. It is given as

$$\text{Var}(x) = \langle x^2 \rangle - \langle x \rangle^2, \quad (\text{A.34})$$

which means we only need the expectation values $\langle x \rangle$ and $\langle x^2 \rangle$. The second term is a bit more demanding because we need to store each individual measurements in order to calculate the cross terms. We analyze the problem by splitting the correlation term into partial sums

$$f_d = \frac{1}{n} \sum_{k=1}^{n-d} (x_k - \bar{x}_n)(x_{k+d} - \bar{x}_n), \quad (\text{A.35})$$

which means the correlation term can be written as

$$\frac{2}{n^2} (x_k - \bar{x}_n)(x_l - \bar{x}_n) = \frac{2}{n} \sum_{d=1}^{n-1} f_d. \quad (\text{A.36})$$

The value f_d is a measure of how correlated values separated by the distance d are. Notice that when $d = 0$, we have that $f_0 = \text{Var}(x)$. We define the *autocorrelation function* as

$$\kappa_d = \frac{f_d}{\text{Var}(x)} \quad (\text{A.37})$$

which gives us a useful measure of the correlation at distance d , starting with $\kappa_0 = 1$. We now rewrite the sample error as

$$\begin{aligned} \text{err}_X^2 &= \frac{1}{n} \text{Var}(x) + \frac{2}{n} \text{Var}(x) \\ &= \left(1 + 2 \sum_{d=1}^{n-1} \kappa_d \right) \frac{1}{n} \text{Var}(x) \\ &= \frac{\tau}{n} \text{Var}(x), \end{aligned} \quad (\text{A.38})$$

where we have defined the *autocorrelation time*

$$\tau = 1 + 2 \sum_{d=1}^{n-1} \kappa_d. \quad (\text{A.39})$$

We see that the sample error can be expressed in terms of the uncorrelated sample variance times the autocorrelation time. This autocorrelation time can be used to define the effective number of measurements, $n_{eff} = n/\tau$, for us to simply approximate the error by the sample variance. From this point of view, we interpret sequential correlations as an effective reduction of measurements by a factor τ . Neglecting this factor will always cause us to underestimate the true sample error.

Bibliography

- [1] D. Griffiths, *Introduction to Quantum Mechanics - International Edition*. Pearson Education International, 2005.
- [2] P. C. Hemmer, *Kvantemekanikk*. Tapir Akademisk Forlag, 2005.
- [3] D. C. Lay, *Linear Algebra*. Pearson Education International, 2006.
- [4] J. M. Leinaas, “Non-relativistic quantum mechanics, lecture notes - fys 4110.” <http://www.uio.no/studier/emner/matnat/fys/FYS4110/h09/>.
- [5] M. L. Boas, *Mathematical Methods in the Physical Sciences*. Wiley, 2006.
- [6] R. Cassanas and H. Siedentop, “The ground-state energy of heavy atoms according to brown and ravenhall: absence of relativistic effects in leading order,” *J. Phys. A: Math. and Gen.*, vol. 39, no. 33, pp. 10405–10414, 2006.
- [7] E. K. U. Gross, E. Runge, and O. Heinonen, *Many-Particle Theory*. IOP Publishing, 1991.
- [8] S. M. Reimann and M. Manninen, “Electronic structure of quantum dots,” *Rev. Mod. Phys.*, vol. 74, no. 4, pp. 1283–1342, 2002.
- [9] D. A. Nissenbaum, *The stochastic gradient approximation: an application to li nanoclusters*. PhD thesis, Northeastern University, Boston, Mass., 2008.
- [10] S. Tarucha, D. G. Austing, T. Honda, R. J. van der Hage, and L. P. Kouwenhoven, “Shell filling and spin effects in a few electron quantum dot,” *Phys. Rev. Lett.*, vol. 77, no. 17, pp. 3613–3616, 1996.
- [11] S. Weiss, M. Thorwart, and R. Egger, “Charge qubit entanglement in double quantum dots,” *Europhys. Lett.*, vol. 76, no. 5, pp. 905–911, 2006.
- [12] W. Lu, Z. Ji, L. Pfeiffer, K. W. West, and A. J. Rimberg, “Real-time detection of electron tunnelling in a quantum dot,” *Nature*, vol. 423, no. 6938, pp. 422–425, 3004.
- [13] G. Burkard, D. Loss, and D. P. DiVincenzo, “Coupled quantum dots as quantum gates,” *Phys. Rev. B*, vol. 59, no. 3, pp. 2070–2078, 1999.
- [14] M. A. Reed, “Quantum dots,” *Scientific American*, vol. 268, no. 41, pp. 118–123, 1993.
- [15] L. P. Kouwenhoven and C. M. Marcus, “Quantum dots,” *Physics World*, vol. 11, no. 6, pp. 35–39, 1998.

- [16] L. P. Kouwenhoven, C. M. Marcus, P. L. McEuen, S. Tarucha, R. M. Westervelt, and N. S. Wingreen, "Electron transport in quantum dots." <http://marcuslab.harvard.edu/papers/KouwenhovenReview.pdf>.
- [17] J. O. Winter, *Development and Optimization of Quantum Dot-Neuron Interfaces*. PhD thesis, The University of Texas at Austin, 2004.
- [18] C. Kittel, *Introduction to Solid State Physics*. John Wiley & Sons, Inc, 2005.
- [19] D. J. Norris, *Measurement and Assignment of the Size-Dependent Optical Spectrum in Cadmium Selenide (CdSe) Quantum Dots*. PhD thesis, Massachusetts Institute of Technology, 1995.
- [20] R. Rossetti, S. M. Beck, and L. E. Brus, "Direct observation of charge-transfer reactions across semiconductor: aqueous solution interfaces using transient raman spectroscopy," *J. of the Am. Chem. Soc.*, vol. 106, no. 4, 1984.
- [21] D. F. Salisbury, "Quantum dots that produce white light could be the light bulb's successor." <http://www.physorg.com/news7421.html>.
- [22] A. Luque and A. Martí, "Increasing the efficiency of ideal solar cells by photon induced transitions at intermediate levels," *Phys. Rev. Lett.*, vol. 78, no. 26, pp. 5014–5017, 1997.
- [23] M. Y. Levy, C. Honsberg, A. Martí, and A. Luque, "Quantum dot intermediate band solar cell material systems with negligible valence band offsets," 2005.
- [24] P. V. Kamat, "Quantum dot solar cells. semiconductor nanocrystals as light harvesters," *J. Phys. Chem. C*, vol. 112, no. 48, pp. 18737–18753, 2008.
- [25] T. Jamieson, R. Bakhshi, D. Petrova, R. Pockock, M. Imani, and A. M. Seifalian, "Biological applications of quantum dots," *Science Direct*, vol. 28, no. 31, pp. 4717–4732, 2007.
- [26] P. Merlot, "Many-body approaches to quantum dots," Master's thesis, University of Oslo, 2009.
- [27] A. Kumar, S. E. Laux, and F. Stern, "Electron states in a gas quantum dot in a magnetic field," *Phys. Rev. B*, vol. 42, no. 8, pp. 5166–5175, 1990.
- [28] M. Macucci, K. Hess, and G. J. Iafrate, "Electronic energy spectrum and the concept of capacitance in quantum dots," *Phys. Rev. B*, vol. 48, no. 23, pp. 17354–17363, 1993.
- [29] D. Heitmann and J. P. Kotthaus, "The spectroscopy of quantum dot arrays," *Physics Today*, vol. 46, no. 6, pp. 56–63, 1993.
- [30] D. Heitmann, "Far infrared spectroscopy of quantum-dots and antidot arrays," *Physica B: Condensed Matter*, vol. 212, no. 3, pp. 201–206, 1995.
- [31] J. M. Leinaas, "Classical mechanics and electrodynamics, lecture notes - fys 3120." <http://www.uio.no/studier/emner/matnat/fys/FYS3120/v09/>.
- [32] E. Flage-Larsen, "Exact diagonalization of quantum dots," Master's thesis, University of Oslo, 2006.

-
- [33] M. P. Lohne, “Coupled-cluster studies of quantum dots,” Master’s thesis, University of Oslo, 2010.
- [34] M. Taut, “Two electrons in a homogeneous magnetic field: particular analytical solutions,” *Journal of Physics A: Mathematical and General*, vol. 27, no. 3, pp. 1045–1055, 1994.
- [35] M. B. Tavernier, E. Anisimovas, F. M. Peeters, B. Szafran, J. Adamowski, and S. Bednarek, “Four-electron quantum dot in a magnetic field,” *Phys. Rev. B*, vol. 68, no. 20, pp. 205305–105313, 2003.
- [36] J. M. Thijssen, *Computational Physics*. Cambridge University Press, 2007.
- [37] I. Shavitt and R. J. Bartlett, *Many-Body methods in Chemistry and Physics*. Cambridge University Press, 2009.
- [38] M. Hjorth-Jensen, “Computational physics.”
<http://www.uio.no/studier/emner/matnat/fys/FYS3150/h10/undervisningsmateriale/>.
- [39] B. Hammond, W. Lester, and P. Reynolds, *Monte Carlo Methods in Ab Initio Quantum Chemistry*. World Scientific, 1994.
- [40] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state calculations by fast computing machines,” *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [41] W. K. Hastings, “Monte carlo sampling methods using markov chains and their applications,” *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.
- [42] C. J. Umrigar and C. Filippi, “Energy and variance optimization of many-body wave functions,” *Phys. Rev. Lett.*, vol. 94, no. 15, pp. 150201–150204, 2005.
- [43] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C++*. Cambridge University Press, 2002.
- [44] H. G. Flyvbjerg, H. and Petersen, “Error estimates on averages of correlated data,” *J. Chem. Phys.*, vol. 91, no. 1, pp. 461–466, 1989.
- [45] S. Siljamäki, *Wave function methods for quantum dots in magnetic field*. PhD thesis, Helsinki University of Technology, 2003.
- [46] M. P. Lohne, M. Hagen, G. and Hjorth-Jensen, and F. Kvaal, S. and Pederiva, “Ab initio computation of circular quantum dots,” *arXiv:1009.4833v1*, 2010.
- [47] A. Harju, B. Barbiellini, S. Siljamäki, R. M. Nieminen, and G. Ortiz, “Stochastic gradient approximation: An efficient method to optimize many-body wave functions,” *Phys. Rev. Lett.*, vol. 79, no. 7, pp. 1173–1177, 1997.