Computational Environment For Many Electron Systems

by

RUNE ALBRIGTSEN

THESIS
for the degree of
MASTER OF SCIENCE

(Master of Science Thesis in Computational Physics)

Department of Physics
University of Oslo

March 2009

Faculty of Mathematics and Natural Sciences
University of Oslo

Contents

Chapter 1

Introduction

When designing numerical software it is common to specialize it to the current problem. This makes it easier to get started right away, without having to worry about how to generalize an algorithm and so forth. However, when the original problem changes slightly it usually involves changing small parts of program code scattered around in a random fashion. The aim of this thesis is to develop a general numerical software that can be used to study different quantum mechanical many electron systems with only small changes in the code. It is a design philosophy that is heavily inspired by Diffpack which is a library for solving partial differential equations.

We will first review the well known harmonic oscillator system and then generalize to quantum dots. Atomic systems are also briefly mentioned as they also can be studied with this software. We have chosen to limit our implementation to closed shell systems but it can easily be generalized to all types of electronic configurations.

The numerical method of choice is the Monte Carlo technique which is reviewed next. It is a very powerful tool for solving many dimensional systems with greater efficiency than standard grid methods. This is due to being able to choose which configurations that are important to the solution. In chapter 4 we present the implementation of the program and which optimization has been done. We then apply it to the many electron harmonic oscillator and compare it with known solutions to verify that the code works. In the last chapter we discuss what we have found and also possible extensions of the program.

Chapter 2

Electronic Systems

This chapter will be devoted to the theoretical treatment of some of the systems our program will be able to handle. The aim is to present single-particle wave functions which are good starting points for our many-body studies. Examples of such widely used single-particle wave functions are the the solutions to the harmonic oscillator problem or the solutions to hydrogen-like problems.

These single-particle wave functions are in turn used in constructing the so-called Slater determinant, which together with a correlation part, is a key part of our ansatz for the trial wave functions used in the Monte Carlo algorithm.

2.1 The Harmonic Oscillator

The harmonic oscillator describes systems where the force on a particle is proportional to the distance from an equilibrium position. The most famous example is of course that of a mass coupled to a spring which abides to Hooke's law. However, many other systems such as molecules, motions of an atom in a lattice and phonons can be described as either one or a collection of harmonic oscillators. In this section we will first review the one particle system in one dimension and then generalize to d dimensions. Since we confine ourselves to electronic systems, our particles are electrons. However, our formalism can easily be extended to other particle species, such as neutrons and protons.

2.1.1 One Dimension

The one-particle harmonic oscillator Hamiltonian in one dimension is

$$\begin{aligned} H &= \frac{1}{2m_e}p^2 + \frac{1}{2}m_e\omega^2x^2, && (2.1)\\ &= -\frac{\hbar^2}{2m_e}\frac{d^2}{dx^2} + \frac{1}{2}m_e\omega^2x^2, && (2.2)\end{aligned}$$

where ω is the oscillator frequency and m_e is the electron mass. The time independent Schrödinger equation $H\phi = E\phi$ becomes

$$-\frac{\hbar^2}{2m_e}\frac{d^2\phi}{dx^2} + \frac{1}{2}m_e\omega^2x^2\phi = E\phi. \tag{2.3}$$

Before we solve this it is convenient to introduce dimensionless variables

$$\begin{aligned} \underline{x} &= \frac{x}{\sqrt{\hbar/m_e\omega}}, && (2.4)\\ \epsilon &= \frac{2E}{\hbar\omega}. && (2.5)\end{aligned}$$

Dropping the underline in $\underline{x}$ and using the prime notation to indicate derivation we get

$$\phi'' + (\epsilon - x^2)\phi = 0. \tag{2.6}$$

This is a second-order linear homogeneous differential equation. To solve it we first look at the behaviour when $x \to \infty$. In this case we can neglect the ϵ term and get

$$\phi'' = x^2\phi, \quad x \to \infty. \tag{2.7}$$

We need a function that when derivated gives back the original function times x. The two functions $\phi(x) = e^{\pm\frac{x^2}{2}}$ do exactly this. Inserting them into eq. (2.7)

$$\phi'' = (x^2 + 1)\phi \approx x^2\phi, \quad x \to \infty. \tag{2.8}$$

Since a wave function must be normalizable, only the function $\phi(x) = e^{-\frac{x^2}{2}}$ belongs to the physical Hilbert space. The solution is therefore of the type $\phi(x) = f(x)e^{-\frac{x^2}{2}}$ for some unknown function f. Inserting this into eq. (2.6) we end up with

$$f'' - 2xf' + (\epsilon - 1)f = 0. \tag{2.9}$$

This type of equation is solved by the power series method where we assume that $f(x)$ is represented by a polynomial in x, namely $f(x) = \sum_{k=0}^{\infty} a_k x^k$. The derivatives are

$$f' = \sum_{k=0}^{\infty} k a_k x^{k-1}, \tag{2.10}$$

$$f'' = \sum_{k=0}^{\infty} k(k-1) a_k x^{k-2}, \tag{2.11}$$

which inserted into eq. (2.9) give

$$\sum_{k=0}^{\infty} k(k-1) a_k x^{k-2} + 2k a_k x^k + (\epsilon - 1) a_k x^k = 0. \tag{2.12}$$

To obtain a recursion relation we notice that the two first terms in f'' will be zero because of the $k(k-1)$ factor. Changing variables k with $k+2$ we get

$$f'' = \sum_{k=0}^{\infty} k(k-1) a_k x^{k-2} = \sum_{k=0}^{\infty} (k+2)(k+1) a_{k+2} x^k, \tag{2.13}$$

which results in

$$\sum_{k=0}^{\infty} \left[(k+2)(k+1) a_{k+2} - (2k+1-\epsilon) a_k\right] x^k = 0, \tag{2.14}$$

which is only fulfilled for all x when the terms inside the brackets vanish for all k, viz.,

$$a_{k+2} = \frac{2k+1-\epsilon}{(k+2)(k+1)} a_k, \quad k = 0, 1, 2, \ldots \tag{2.15}$$

and the first two coefficients, a_0 and a_1, can be chosen freely. The series will diverge for large x if there are no restrictions on ϵ. We show this by noting that for large x, the coefficients for large k will be dominate the series

$$\frac{a_{k+2}}{a_k} \simeq \frac{2}{k}. \tag{2.16}$$

Comparing this result with the recursion relation for the power series expansion of e^{x^2}

$$e^{x^2} = \sum_{k=0,2,4,\ldots} b_k x^k, \tag{2.17}$$

$$b_k = \frac{1}{(\frac{1}{2}k)!} \tag{2.18}$$

which gives the coefficient ratio

$$\frac{b_{k+2}}{b_k} = \frac{(\frac{1}{2}k)!}{(\frac{1}{2}(k+2))!} = \frac{1}{\frac{1}{2}k+1} \simeq \frac{2}{k}, \tag{2.19}$$

we see that the divergence is a fact. The only way to avoid this is by terminating the series so that $f(x)$ becomes a polynomial with a finite number of powers in x. This is possible by restricting

$$\epsilon = \epsilon_n = 2n+1, \quad n = 0, 1, 2, \ldots, \tag{2.20}$$

and choosing $a_1 = 0$ for even n or $a_0 = 0$ for odd n. We have in other words just showed that the energy for the harmonic oscillator is quantized. To find the polynomials $f_n(x)$ we insert eq. (2.20) into eq. (2.9) and get

$$f_n'' - 2xf_n' + 2nf_n, \quad n = 0, 1, 2, \ldots \tag{2.21}$$

It can be shown that the solution to these equations are the Hermite polynomials $H_n(x)$ which are defined as

$$H_n(x) = e^{x^2}\left(-\frac{d}{dx}\right)^n e^{-x^2}. \tag{2.22}$$

Using recursion relations

$$\begin{aligned} H_{n+1} &= 2xH_n - 2nH_{n-1}, && (2.23)\\ H_n' &= 2nH_{n-1}. && (2.24) \end{aligned}$$

the first five terms are

$$\begin{aligned} H_0 &= 1, && (2.25)\\ H_1 &= 2x, && (2.26)\\ H_2 &= 4x^2 - 2, && (2.27)\\ H_3 &= 8x^3 - 12x, && (2.28)\\ H_4 &= 16x^4 - 48x^2 + 12. && (2.29) \end{aligned}$$

The Hermite polynomials have also two other important properties

$$\begin{aligned} \int_{-\infty}^{\infty} H_n(x)H_m(x)e^{-x^2} &= 0, \quad n \neq m && (2.30)\\ \int_{-\infty}^{\infty} H_n^2(x)e^{-x^2} &= 2^n n! \int_{-\infty}^{\infty} e^{-x^2} = \sqrt{\pi}\, 2^n n!, && (2.31) \end{aligned}$$

which tell us that the wave functions are orthogonal and normalized. Switching back to normal coordinates (that it is inserting the correct dimensions) we get

$$\begin{aligned} \phi_n(x) &= \left(\frac{m_e\omega}{\pi\hbar}\right)^{\frac{1}{4}} \frac{1}{\sqrt{2^n n!}} e^{-m_e\omega x^2/2\hbar} H_n(x\sqrt{m_e\omega/\hbar}) && (2.32) \\ E_n &= (n+\frac{1}{2})\hbar\omega. && (2.33) \end{aligned}$$

We see that the ground state has a non-zero energy.

2.1.2 Harmonic oscillator in d dimensions

It is fortunately very easy to solve the general d-dimensional case once the one-dimensional case is obtained. The Hamiltonian is

$$H = \sum_{r=1}^{d} H_r = \sum_{r=1}^{d} \left(-\frac{\hbar^2}{2m_e} \frac{d^2}{d{x_r}^2} + \frac{1}{2} m\omega_r^2 x_r^2 \right), \tag{2.34}$$

which is a sum over d independent parts. This suggests a wave function on the form

$$\phi_N(x_1,\ldots,x_d) = \prod_{r=1}^{d} \phi_{n_r}(x_r), \quad N = n_1,\ldots,n_d. \tag{2.35}$$

If we identify $n_1 = n_x$, $n_2 = n_y$, $x_1 = x$, $x_2 = y$ and so forth, the Schrödinger equation can be written as

$$\phi_N^{-1} H \phi_N = E_N, \tag{2.36}$$

resulting in

$$\sum_{r=1}^{d} \left(-\frac{\hbar^2}{2m_e} \frac{\phi''_{n_r}}{\phi_{n_r}} + \frac{1}{2} m\omega_r^2 x_r^2 \right) = E_N. \tag{2.37}$$

This can only be fulfilled if each term in the sum is a constant that adds up to E_N

$$\begin{aligned} -\frac{\hbar^2}{2m_e} \frac{\phi''_{n_r}}{\phi_{n_r}} + \frac{1}{2} m\omega_r^2 x_r^2 &= E_r, \quad r = 1,2,\ldots,d && (2.38) \\ \sum_{r=1}^{d} E_r &= E_N. && (2.39) \end{aligned}$$

We thus end up with d one-dimensional equations that we already know the solution to

$$
\begin{aligned}
\phi_N(x_1,\ldots,x_d) &= \prod_{r=1}^{d}\Big(\frac{m_e\omega_r}{\pi\hbar}\Big)^{\frac{1}{4}}\frac{1}{\sqrt{2^{n_r}n_r!}}e^{-m_e\omega_r x_r^2/2\hbar} \\
&\quad \times H_{n_r}(x_r\sqrt{m_e\omega_r/\hbar}), && (2.40) \\
E_N &= \sum_{r=1}^{d}(n_r+\frac{1}{2})\hbar\omega_r. && (2.41)
\end{aligned}
$$

In the special case of an isotropic oscillator potential $\omega_r = \omega$, the energy is

$$
\begin{aligned}
E_N &= (N+\frac{d}{2})\hbar\omega, && (2.42) \\
N &= \sum_{r=1}^{d} n_r. && (2.43)
\end{aligned}
$$

Thus all the excited states are degenerate because different combinations of n_r will yield the same N. To calculate the spatial degeneracy $g_r(N)$ we note that $n_x = 0, 1, \ldots, N$ which are $N+1$ different values. In the two-dimensional case $n_y = N - n_x$ so the degeneracy is just $g_2(N) = N + 1$. If we include the spin degrees of freedom, the total degeneracy $G_2(N)$ becomes

$$G_2(N) = 2(N+1). \qquad (2.44)$$

As we add more and more electrons to the system (obeying the Pauli principle), they will be in the state which gives the lowest total energy for the system. If we measure how much energy is required to add or remove an electron it will depend on how many electrons N_e currently are in the system. The energy needed will have spikes at certain magic numbers $N_e = S_r(N)$ which corresponds to the case when all degenerate states up to a given single-particle level are occupied. This is given by the formula

$$S_r(N) = \sum_{i=1}^{N} G_r(i). \qquad (2.45)$$

The resulting shell structure for the two-dimensional case is displayed in table 2.1 The spatial degeneracy in three dimensions case is found by noting that $n_y = 0, 1, \ldots, N - n_x$ with are $N - n_x + 1$ different values for each n_x.

N	n_x	n_y	$G_2(N)$	$S_2(N)$
0	0	0	2	2
1	1/0	0/1	4	6
2	2/1/0	0/1/2	6	12
3	3/2/1/0	0/1/2/3	8	20

Table 2.1: Shell structure for the two-dimensional harmonic oscillator including various combinations of n_x and n_y

To get the total number of different values we have to sum over all n_x

$$\begin{aligned} g_3(N) &= \sum_{n_x=0}^{N}(N+1-n_x), && (2.46)\\ &= (N+1)^2 - \frac{N(N+1)}{2}, && (2.47)\\ &= (N+1)(N+1-\frac{N}{2}), && (2.48)\\ &= \frac{1}{2}(N+1)(N+2), && (2.49)\end{aligned}$$

which gives a total degeneracy

$$G_3(N) = (N+1)(N+2). \tag{2.50}$$

The shell structure is showed in table 2.2. The spatial degeneracy is related to the symmetry in the potential

N	$G_3(N)$	$S_3(N)$
0	2	2
1	6	8
2	10	20
3	20	40

Table 2.2: Shell structure for the three-dimensional harmonic oscillator

$$V(x_1,\ldots,x_d) = \sum_{r=1}^{d}\frac{1}{2}m_e\omega x_r^2 = \frac{1}{2}m_e\omega r^2 = V(r), \tag{2.51}$$

because when the harmonic oscillator is isotropic, each dimension contributes the same amount of energy.

2.2 Quantum Dots

Quantum dots are man made systems of trapped electrons that share some properties with atoms, hence the popular name "designer atoms". Their size ranges between 100 nm to 1 μm which is much larger than a regular atom. Atoms have typically spatial extensions that are of the size of 0.05-0.4 nm. The quantum dot is created in a semiconductor, typically gallium arsenide (GaAs), and confined either by a physical barrier, such as an insulator like aluminum gallium arsenide (AlGaAs), or an electric field. The confinement gives rise to a bowl shaped potential that can be approximated by the harmonic oscillator potential. One application of quantum dots is as qubits in a quantum computer by manipulating the electron states. Another is in biology for fluorescent labelling of both normal and cancer cells .

We will first consider the two-dimensional quantum dot and then extend our system to three dimensions. The purpose is to show that as long as the two-particle repulsive Coullomb interaction does not depend on spin, then the magnetic field will only result in an effective harmonic oscillator potential and a constant shift in the energy spectrum.

2.2.1 Two dimensions

We first consider the case with no electron-electron repulsion. This means that the Hamiltonian contains only onebody operators. The Hamiltonian reads

$$\hat{H} = \frac{1}{2m^*}\left(\boldsymbol{p} - \frac{e}{c}\boldsymbol{A}\right)^2 + \frac{1}{2}m^*\omega_0^2(x^2+y^2) + e\phi - \boldsymbol{\mu}_S \cdot \boldsymbol{B}, \qquad (2.52)$$

where $\boldsymbol{A}$ and ϕ are the vector and scalar potentials, respectively, associated with the external electromagnetic field. The last term in the Hamiltonian is the coupling of the electron spin magnetic moment $\boldsymbol{\mu}_S$, to the magnetic field and m^* is the effective electron mass. We will consider the special case of a constant and uniform magnetic field along the z-axis, $\boldsymbol{B} = (0, 0, B)$, and no electric field. The vector and scalar potentials are related to the electromagnetic field by the equations

$$\boldsymbol{E} = -\frac{1}{c}\frac{\partial \boldsymbol{A}}{\partial t} - \nabla\phi, \qquad (2.53)$$

$$\boldsymbol{B} = \nabla \times A. \qquad (2.54)$$

It is easy to see that we can only obtain a constant magnetic field if also $\boldsymbol{A}$ is constant in time. The scalar potential is now given by

$$\nabla\phi = 0, \qquad (2.55)$$

which has only the solution $\phi = k$, where k is a constant. We will however set $k = 0$ for the rest of this derivation. Before choosing $\boldsymbol{A}$, we expand the first term in the Hamiltonian

$$(\boldsymbol{p} - \frac{e}{c}\boldsymbol{A})^2 = p^2 - \frac{e}{c}(\boldsymbol{p} \cdot \boldsymbol{A} + \boldsymbol{A} \cdot \boldsymbol{p}) + \frac{e^2}{c^2}A^2, \tag{2.56}$$

and in general we have

$$\boldsymbol{A}(x, y, z) = (A_x(x, y, z), A_y(x, y, z), A_z(x, y, z)). \tag{2.57}$$

To fulfill our equations $\boldsymbol{A}$ and $\boldsymbol{p}$ will not commute unless we demand

$$\boldsymbol{A}(x, y, z) = (A_x(y, z), A_y(x, z), A_z(x, y)). \tag{2.58}$$

In this case we have $\nabla \cdot A = 0$, which means that we are working in the Coulomb gauge. One possibility for the vector potential is $\boldsymbol{A} = \frac{B}{2}(-y, x, 0)$ and by inserting this into eq. (2.56) we get

$$(\boldsymbol{p} - \frac{e}{c}\boldsymbol{A})^2 = p^2 - \frac{eB}{c}(xp_y - yp_x) + \frac{e^2B^2}{4c^2}(x^2 + y^2) \tag{2.59}$$

We note that $xp_y - yp_x = L_z$, where L_z is the angular momentum operator in the z direction. Inserting this into the Hamiltonian we get

$$\begin{aligned} H &= \frac{1}{2m^*}\left[p^2 - \frac{eB}{c}L_z + \frac{e^2B^2}{4c^2}(x^2 + y^2)\right] + \\ &\quad \frac{1}{2}m^*\omega_0^2(x^2 + y^2) + \frac{eg_s^*B}{2m^*c}S_z \end{aligned} \tag{2.60}$$

where g_s^* is the effective spin-gyromagnetic factor. It is easy to check that both L_z and S_z commute with the Hamiltonian. This means that the solution of the stationary Schrödinger equation will be an eigenfunction of L_z and S_z. The L_z operator has a simple representation in polar coordinates

$$L_z = -i\hbar\left(x\frac{\partial}{\partial y} - y\frac{\partial}{\partial x}\right) = -i\hbar\frac{\partial}{\partial\varphi} \tag{2.61}$$

with eigenfunction $e^{im\varphi}$ and eigenvalue $\hbar m$. Because the angles φ and $\varphi + 2\pi$ are required to give the same eigenfunction, m is restricted to the values $m = 0, \pm 1, \pm 2, \ldots$. The spin operator S_z is represented by the matrix

$$S_z = \frac{\hbar}{2}\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{2.62}$$

and its eigenfunction is the two component spinor

$$\boldsymbol{\chi} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \tag{2.63}$$

where $|c_1|^2$ and $|c_2|^2$ are the probabilities for a state with spin up and spin down, respectively. The eigenvalues are $\pm\hbar/2$ and tell us that the spin up state has a higher energy than the spin down state. More formally we write the total wave function as

$$\boldsymbol{\phi} = \phi\boldsymbol{\chi}, \tag{2.64}$$

and split the Hamiltonian in a spatial part and a spin part

$$H = H_\Omega + H_s, \tag{2.65}$$

where

$$H_s = \frac{eg_s^* B}{2m^* c} S_z. \tag{2.66}$$

The stationary Schrödinger equation becomes

$$\boldsymbol{\chi} H_\Omega \phi + \phi H_s \boldsymbol{\chi} = E\phi\boldsymbol{\chi} \tag{2.67}$$

which separates into the following system of coupled equations

$$H_\Omega \phi = E_\Omega \phi, \tag{2.68}$$

$$H_s \boldsymbol{\chi} = E_s \boldsymbol{\chi}, \tag{2.69}$$

where $E_\Omega + E_s = E$. Solving the last one is very easy since H_s is just a constant times S_z, giving

$$E_s = \frac{e\hbar g_s^* B}{2m^* c} s, \tag{2.70}$$

where $s = 1/2$ for spin up and $s = -1/2$ for spin down.

We now move on to solving eq. (2.68). Due to the spatial symmetry in the Hamiltonian we switch to polar coordinates and define

$$\omega_B = \frac{eB}{2m^* c} \tag{2.71}$$

$$\omega^2 = \omega_0^2 + \omega_B^2. \tag{2.72}$$

The spatial part of the Hamiltonian is now

$$H_\Omega = H_{r\varphi} = -\frac{\hbar^2}{2m^*}\left(\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \varphi^2}\right) + i\hbar\omega_B \frac{\partial}{\partial \varphi} + \frac{1}{2} m^* \omega^2 r^2, \tag{2.73}$$

which is separable in r and φ. This means that the spatial part of the wave function is also separable

$$\phi_m(r,\varphi) = R(r)e^{im\varphi}. \tag{2.74}$$

Feeding this into the stationary Schrödinger equation results in the following linear ODE in $R(r)$

$$-\frac{\hbar^2}{2m^*}\left(\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} - \frac{m^2}{r^2}\right)R(r) + \frac{1}{2}m^*\omega^2 r^2 R(r) = \epsilon_m R(r), \tag{2.75}$$

where $\epsilon_m = E_\Omega - \hbar m\omega_B$. This equation is solved by using the same technique as for the one-dimensional harmonic oscillator. We omit therefore the derivation. The normalized solution is

$$\begin{aligned} R_{nm}(r) &= \sqrt{\frac{2n!}{(n+|m|)!}}\beta^{(|m|+1)/2}r^{|m|}e^{-\beta r^2/2}L_n^{|m|}(\beta r^2), & (2.76)\\ E_\Omega = E_{nm} &= (2n+|m|+1)\hbar\omega + \hbar m\omega_B, & (2.77) \end{aligned}$$

where $n = 0, 1, 2, \ldots,$ and

$$\beta = \frac{m^*\omega}{\hbar}, \tag{2.78}$$

and $L_n^{|m|}(\beta r^2)$ is the associated Laguerre polynomial which in the Rodriguez representation is defined as

$$L_n^m(r) = \frac{1}{n!}e^r r^{-m}\frac{\partial^n}{\partial r^n}\left(e^{-r}r^{n+m}\right). \tag{2.79}$$

The first three polynomials are

$$\begin{aligned} L_0^m(r) &= 1, & (2.80)\\ L_1^m(r) &= -r + m + 1, & (2.81)\\ L_2^m(r) &= \frac{1}{2}r^2 - (m+2)r + \frac{1}{2}(m+2)(m+1). & (2.82) \end{aligned}$$

In order to get $\phi_{nm}(r,\varphi)$ we just have to multiply eq. (2.76) with the normalized angular part. The result is

$$\phi_{nm}(r,\varphi) = \sqrt{\frac{n!}{\pi(n+|m|)!}}\beta^{(|m|+1)/2}r^{|m|}e^{-\beta r^2/2}L_n^{|m|}(\beta r^2)e^{im\varphi}. \tag{2.83}$$

The total energy of the system is

$$E = E_{nms} = E_{nm} + E_s = (2n+|m|+1)\hbar\omega + m\hbar\omega_B + g_s\hbar\omega_B s, \tag{2.84}$$

and to analyze the effect of the magnetic field we compare this energy to

$$E_{B=0} = E_{nm} = (2n + |m| + 1)\hbar\omega_0. \tag{2.85}$$

In this case we regain the energy spectrum of a two-dimensional harmonic oscillator as we should, but with different quantum numbers reflecting the change to polar coordinates. Clearly $N = 2n + |m|$, and the shell structure is shown in table 2.3. We see that the presence of the magnetic field makes the energy depend on the sign of m and s. The previous degenerate states will now separate more and more as the magnetic field increases. When there are no degeneracies left, the concept of shell structure may at first seem problematic. However for small magnetic fields the ionization energy will still have peaks at the same magic numbers as for the degenerate case. For strong magnetic fields this picture breaks down. We should also mention that for some special magnetic field strengths some of the non-degenerate energy levels will overlap and we can have so called accidental degeneracies. One example is when $\omega_0/\omega_B = \sqrt{(1+g_s^*)^2 - 1}$ which would make $E_{00\frac{1}{2}} = E_{0-1-\frac{1}{2}}$.

N	n	m
0	0	0
1	0	± 1
2	0/2	$\pm 2/0$
3	0/1	$\pm 3/\pm 1$

Table 2.3: Shell structure for the two-dimensional harmonic oscillator with polar coordinate quantum numbers

We briefly mention the relationship between the wave functions of the harmonic oscillator in polar and Cartesian coordinates by comparing the wave functions for $N = 1$. We use atomic units and omit the normalization factors in order to simplify the relations. In Cartesian coordinates the wave functions are given by

$$\phi_{10} = xe^{\frac{\omega}{2}(x^2+y^2)}, \tag{2.86}$$

$$\phi_{01} = ye^{\frac{\omega}{2}(x^2+y^2)}, \tag{2.87}$$

while in polar coordinates they are

$$\phi_{01} = re^{\frac{\omega}{2}r^2}e^{(i\varphi)}, \tag{2.88}$$

$$\phi_{0-1} = re^{\frac{\omega}{2}r^2}e^{(-i\varphi)}. \tag{2.89}$$

Using the following relations

$$\begin{aligned} e^{(\pm i\varphi)} &= \cos\varphi \pm i\sin\varphi, && (2.90)\\ x &= r\cos\varphi, && (2.91)\\ y &= r\sin\varphi, && (2.92) \end{aligned}$$

we can write the last two wave functions as

$$\phi_{0\pm1} = e^{\frac{\omega}{2}(x^2+y^2)}\left(x \pm iy\right). \tag{2.93}$$

They are thus related to each other by the normalized linear combination

$$\phi_{0\pm1} = \frac{1}{\sqrt{2}}\left(\phi_{10} \pm i\phi_{01}\right). \tag{2.94}$$

It also tells us that the two eigenfunctions of L_z are $x \pm iy$, with eigenvalues $\pm\hbar$.

2.2.2 Three dimensions

The only change in the Hamiltonian is the inclusion of the z-direction. Thus the spatial Hamiltonian reads

$$H_\Omega = \frac{1}{2m^*}\left[p^2 - \frac{eB}{c}L_z + \frac{e^2B^2}{4c^2}\left(x^2+y^2\right)\right] + \frac{1}{2}m^*\omega_0^2\left(x^2+y^2\right) + \frac{1}{2}m^*\omega_z^2 z^2 \tag{2.95}$$

where p^2 now includes p_z^2. The L_z operator commutes with H_Ω, however, an analytical solution is only attainable when $\omega_z = \omega$ since the magnetic field only shifts the part of the oscillator potential that is perpendicular to the magnetic field. Using this and changing to polar coordinates we get

$$\begin{aligned} H_{r\theta\varphi} &= -\frac{\hbar^2}{2m^*}\left[\frac{\partial^2}{\partial r^2} + \frac{2}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\left(\frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\varphi^2} + \cot\theta\frac{\partial}{\partial\theta} + \frac{\partial^2}{\partial\theta^2}\right)\right]\\ &\quad + i\hbar\omega_B\frac{\partial}{\partial\varphi} + \frac{1}{2}m^*\omega^2r^2. \end{aligned} \tag{2.96}$$

The part inside the parenthesis is equal to the operator $-\frac{1}{\hbar^2}L^2$ which also commutes with $H_{r\theta\varphi}$. Its eigenfunctions are the spherical harmonics Y_{lm} which in normalized form are

$$Y_{lm}(\theta,\varphi) = \delta_m\left[\frac{(2l+1)(l-|m|)!}{4\pi(l+|m|)!}\right]^{\frac{1}{2}} P_l^{|m|}(\cos\theta)e^{im\varphi}, \tag{2.97}$$

with the requirement $l = 0, 1, 2, \dots,$ and $m = 0, \pm 1, \pm 2, \dots, \pm l$ and where

$$\delta_m = \begin{cases} (-1)^m & m \geq 0, \\ 1 & m \leq 0. \end{cases} \tag{2.98}$$

The associated Legendre polynomials P_l^m are in the Rodriguez representation defined as

$$P_l^m(x) = \frac{(-1)^m}{2^l l!}(1-x^2)^{\frac{m}{2}} \left(\frac{d}{dx}\right)^{l+m} (x^2-1)^l, \quad |m| \leq l, \tag{2.99}$$

and the six lowest-order associated Legendre polynomials are shown in table 2.4. The eigenvalues of L^2 are $\hbar^2 l(l+1)$. The Hamiltonian is once again separable. An ansatz for the wave function is then

$\lvert m\rvert$	l	$P_l^m(x)$
0	0	1
0	1	x
0	2	$\frac{1}{2}(3x^2-1$
1	1	$1-\sqrt{1-x^2}$
1	2	$-3x\sqrt{1-x^2}$
2	2	$3(1-x^2)$

Table 2.4: Lowest order associated Legendre polynomials

$$\phi_{lm}(r, \theta, \varphi) = R(r) Y_{lm}(\theta, \varphi). \tag{2.100}$$

Inserting this into the stationary Schrödinger equation we end up with the differential equation

$$-\frac{\hbar^2}{2m^*}\left(\frac{\partial^2}{\partial r^2} + \frac{2}{r}\frac{\partial}{\partial r} - \frac{l(l+1)}{r^2}\right)R(r) + \frac{1}{2}m^*\omega^2 r^2 R(r) = \epsilon_m R(r), \tag{2.101}$$

which is quite similar to the two-dimensional one. The solution is

$$R_{nl}(r) = \left[\frac{2n!}{(n+l+1/2)!}\right]^{1/2} \beta^{(l+3/2)} r^l e^{-\beta r^2/2} L_n^{l+1/2}(\beta r^2), \tag{2.102}$$

$$E_\Omega = E_{nlm} = (2n+l+\frac{3}{2})\hbar\omega + \hbar m \omega_B \tag{2.103}$$

with n=0,1,2,.... As expected, there are no degenerate energy levels and if we set $B = 0$ we regain the energy spectrum of the harmonic oscillator by identifying $N = 2n + l$.

2.2.3 Two-particle system

We will now present how the problem of a two electron quantum dot can be solved analytically for certain values of ω. These results are taken from Ref. [1] and form an important basis for checking our numerical solution method and also to gain some physical insight on the role of correlations. The spatial Hamiltonian in two dimensions is

$$H_{r_\varphi} = \sum_{i=1}^{2} \left[\frac{1}{2m^*} \left(\boldsymbol{p}_i - \frac{e}{c} \boldsymbol{A}_i \right)^2 + \frac{1}{2} m^* \omega_0^2 r_i^2 \right] + \frac{e^2}{4\pi\epsilon_0 |\boldsymbol{r}_2 - \boldsymbol{r}_1|}. \tag{2.104}$$

By introducing the center-of-mass and relative coordinates we can split the Hamiltonian in two parts, one which depends on the relative coordinates only and one which depends on the center-of-mass coordinates. We can then use the separation of variables technique when solving the stationary Schrödinger equation. The coordinate transformation is given by

$$\boldsymbol{r} = \boldsymbol{r}_2 - \boldsymbol{r}_1, \tag{2.105}$$

$$\boldsymbol{R} = \frac{1}{2}(\boldsymbol{r}_1 + \boldsymbol{r}_2), \tag{2.106}$$

where $\boldsymbol{R}$ and $\boldsymbol{r}$ are the center-of-mass term and the relative term, respectively. The momenta will also be transformed according to

$$\boldsymbol{p} = \frac{1}{2}(\boldsymbol{p}_2 - \boldsymbol{p}_1), \tag{2.107}$$

$$\boldsymbol{P} = \boldsymbol{p}_1 + \boldsymbol{p}_2. \tag{2.108}$$

When $\boldsymbol{B}$ is a constant, Maxwell's equations implies that $\boldsymbol{A}$ must be linear

$$\boldsymbol{A}(r) = \boldsymbol{A}(r_2) - \boldsymbol{A}(r_1), \tag{2.109}$$

$$\boldsymbol{A}(R) = \frac{1}{2}(\boldsymbol{A}(r_1) + \boldsymbol{A}(r_2)). \tag{2.110}$$

The following relations will come in handy when transforming the Hamiltonian

$$p_1^2 + p_2^2 = \frac{1}{2}(P^2 + 4p^2), \tag{2.111}$$

$$r_1^2 + r_2^2 = \frac{1}{2}(4R^2 + r^2), \tag{2.112}$$

$$|\boldsymbol{r}_2 - \boldsymbol{r}_1| = r \tag{2.113}$$

$$\boldsymbol{p}_1 \cdot \boldsymbol{A}(r_1) + \boldsymbol{p}_2 \cdot \boldsymbol{A}(r_2) = \boldsymbol{p} \cdot \boldsymbol{A}(r) + \boldsymbol{P} \cdot \boldsymbol{A}(R) \tag{2.114}$$

$$A_{r_1}^2 + A_{r_2}^2 = \frac{1}{2} A(r)^2 + 2A(R)^2 \tag{2.115}$$

The result is

$$\begin{aligned} H_{r_\varphi} = \frac{1}{2}\left\{\frac{1}{2m^*}\left[P^2 - 4\frac{e}{c}\boldsymbol{P}\cdot\boldsymbol{A}(R) + 4\frac{e^2}{c^2}A(R)^2\right] + 2m^*\omega_0^2R^2\right\} \\ + 2\left\{\frac{1}{2m^*}\left[p^2 - \frac{e}{c}\boldsymbol{p}\cdot\boldsymbol{A}(r) + \frac{1}{4}\frac{e^2}{c^2}A(r)^2\right] + \frac{1}{8}m^*\omega_0^2r^2 \right. \\ \left. + \frac{e^2}{8\pi\epsilon_0 r}\right\}, \end{aligned} \tag{2.116}$$

and by introducing

$$\omega_R = 2\omega_0, \tag{2.117}$$

$$\omega_r = \frac{1}{2}\omega_0, \tag{2.118}$$

$$\boldsymbol{A}_R = 2\boldsymbol{A}(R), \tag{2.119}$$

$$\boldsymbol{A}_r = \frac{1}{2}\boldsymbol{A}(r), \tag{2.120}$$

we get

$$\begin{aligned} H_{r_\varphi} &= \frac{1}{2}\left\{\frac{1}{2m^*}\left[P - \frac{e}{c}\boldsymbol{A}_R\right]^2 + \frac{1}{2}m^*\omega_R^2R^2\right\} + \\ & \quad 2\left\{\frac{1}{2m^*}\left[p - \frac{e}{c}\boldsymbol{A}_r\right]^2 + \frac{1}{2}m^*\omega_r r^2 + \frac{e^2}{8\pi\epsilon_0 r}\right\} \qquad (2.121) \\ &\equiv \frac{1}{2}H_R + 2H_r. \qquad (2.122) \end{aligned}$$

The wave function can then be written in product form as

$$\Psi(\boldsymbol{r},\boldsymbol{R}) = \psi_r(\boldsymbol{r})\psi_R(\boldsymbol{R}) \tag{2.123}$$

and when inserted into the Schrödinger $H\Psi = E\Psi$ we end up solving the system of equations

$$H_r\psi_r = E_r\psi_r, \tag{2.124}$$

$$H_R\psi_R = E_R\psi_R, \tag{2.125}$$

with total energy

$$E = \frac{1}{2}E_R + 2E_r. \tag{2.126}$$

The solution to the center-of-mass problem, eq (2.125) is the same as for the one particle case. The center-of-mass energy is

$$E_R = E_{NM} = 2(N + |M| + 1)\hbar\omega + 2\hbar M\omega_B \tag{2.127}$$

where the extra factor of two comes from eq. (2.119) and eq. (2.117).

We now move on to solve eq. (2.124) which is the relative part. The presence of the $1/r$ term makes it impossible to get a general closed-form solution. Such solutions exist only for particular values of ω that can be found by solving for $a_n = 0$ where a_n is given by the following recurrence relation

$$\begin{aligned} a_0 &\neq 0 && (2.128)\\ a_1 &= \frac{1}{a_0'(2|m|+1)}\sqrt{\frac{2\hbar}{m^*\omega}}a_0, && (2.129)\\ a_n &= \frac{1}{n(n+2|m|)}\left\{\frac{1}{a_0'}\sqrt{\frac{2\hbar}{m^*\omega}}a_{n-1},\right.\\ &\quad \left. + \left[2\left(n+|m|-1\right)-\epsilon_{nm}\right]a_{n-2}\right\}, \quad n \geq 2, && (2.130)\end{aligned}$$

where

$$a_0' = \frac{8\pi\epsilon_0\hbar^2}{2m^*e^2}, \tag{2.131}$$

and

$$\epsilon_{nm} = 2(|m|+n), \quad n \geq 2. \tag{2.132}$$

The energy is given by

$$E_r = E_{nm} = \frac{1}{2}(|m|+n)\hbar\omega + \frac{1}{2}m\hbar\omega_B. \tag{2.133}$$

The procedure is to chose a value for n and insert ϵ_{nm} into the expression for a_n and then solve $a_n = 0$ with respect to ω. The ground state is given by $n = 2$ and $m = 0$ which gives $\omega = 1$ and energy $E_r = \hbar$. The ground state of the center-of-mass part is given by $N = M = 0$ which gives an energy $E_R = 2\hbar$. The total energy for the ground state of a two particle quantum dot is then $E = 3\hbar$. The energy for the non-interacting case is just the one particle energy multiplied with two (the spin part cancels) and will for the ground state be equal to the center of mass energy. We see that the interaction energy is 1/3 of the total energy.

In the three-dimensional case it can be shown that we only have to make the substitution $|m| = l + 1/2$ in the above expressions. The energy is

$$E_R = E_{NLM} = 2(2N + L + \frac{3}{2})\hbar\omega + 2\hbar M\omega_B \tag{2.134}$$

$$E_r = E_{nlm} = \frac{1}{2}(n + l + \frac{1}{2})\hbar\omega + \frac{1}{2}m\hbar\omega_B. \tag{2.135}$$

The ground state energy is given by $n = 2, l = m = 0$ and $N = L = M = 0$ for the relative and center of mass part, respectively. This gives $\omega = 1/2$ and total energy $E = 2\hbar$ from which the interaction part contributes $1/4$. We see that the interaction part is more important in the two-dimensional case. This is expected since lower dimensionalities give less degrees of freedom.

2.3 Atomic systems

We want to show that our program is capable of solving multi-electron atoms by employing a Slater determinant defined by for example hydrogen-like wave functions. These wave functions are found by solving the stationary Schrödinger equation for one electron in an atom with Z protons. This is a $Z + 1$ body problem that is in general not analytically solvable. However, we can exploit that the nucleus is much heavier than the electron by many orders of magnitude. The effect will be that the nucleus is almost at rest compared to the electron. We will therefore approximate the problem by treating the nucleus as having no kinetic energy. Though one should note that this approximation will only be accurate if the momenta of the nucleus and the electron is of the same order of magnitude. This is easily shown by comparing their kinetic energy T given by

$$\begin{aligned} T_e &= \frac{P_e^2}{2m_e}, \quad (2.136) \\ T_Z &= \frac{P_Z^2}{2Zm_p}. \quad (2.137) \end{aligned}$$

We see that only if $P_e \approx P_Z$ will $T_e >> T_p$. This is one step in the Born-Oppenheimer (BO) approximation and is very important in molecular physics.

2.3.1 Hydrogen like wave functions

The Hamiltonian is

$$H = -\frac{\hbar^2}{2m_e}\nabla^2 - \frac{Ze^2}{4\pi\epsilon_0}\frac{1}{r}, \quad (2.138)$$

and is spherically symmetric. This means that, as for the three-dimensional quantum dot, the wave function is separable and we can use the ansatz

$$\phi_{lm}(r, \theta, \varphi) = R(r)Y_{lm}(\theta, \varphi) \quad (2.139)$$

where Y_{lm} are the spherical harmonics defined in eq. (2.97). The Schrödinger equation reads

$$\left\{ -\frac{\hbar^2}{2m_e}\left[\frac{\partial^2}{\partial r^2}+\frac{2}{r}\frac{\partial}{\partial r}+\frac{l(l+1)}{r^2}\right]-\frac{Ze^2}{4\pi\epsilon_0}\frac{1}{r}\right\} R(r)=ER(r). \tag{2.140}$$

The derivation of its solution is given in standard quantum mechanical textbooks, see for example Ref. [2]. The result is

$$E_n = -\frac{\hbar^2}{2m_e a_0^2}\frac{Z^2}{n^2}, \tag{2.141}$$

$$R_{nl}(r) = -\left[\frac{4(n-l-1)!}{a^3n^4[(n+l)!]^3}\right]^{\frac{1}{2}}(\rho_n r)^l e^{-\frac{1}{2}\rho_n r}L_{n+l}^{2l+1}(\rho_n r), \tag{2.142}$$

$$n = 1,2,\dots, \tag{2.143}$$

$$l = 0,1,\dots,n-1, \tag{2.144}$$

where $\rho_n = 2/na$ and $a = a_0/Z$. Comparing this energy with that of the three-dimensional harmonic oscillator we see that it only depends on one quantum number. To calculate the spatial degeneracy we see that for each n, l can take $n-1$ different values. We also know that for each l, m can take $g_l = (2l+1)$ different values and is called a subshell. By summing this we get

$$g_n = \sum_{l=0}^{n-1}(2l+1) = n^2, \tag{2.145}$$

and when including spin degrees of freedom the total degeneracy is $G_n = 2n^2$. The shell structure is shown in table 2.5.

nl	m	G_l	S_l	G_n	S_n
$1s$	0	2	2	2	2
$2s$	0	2	4		
$2p$	$-1,0,1$	6	10	8	10
$3s$	0	2	12		
$3p$	$-1,0,1$	6	18		
$3d$	$-2,-1,0,1,2$	10	28	18	28

Table 2.5: Shell structure in an atom using spectroscopic notation. The number of orbitals in each subshell is $G_l = 2g_l$ and the total number of orbitals in all subshells is $S_l = \sum G_l$.

2.4 Identical particle symmetry

When dealing with systems of more than one particle it is natural to label them with a number. This is valid in classical physics even if the particles are identical because [1] we can always separate the particles by observing their historical trajectories. However, this is not possible in the quantum mechanical case because observing the system would disturb it. The logical consequence of this is that interchanging a pair of particles in the wave function should not change any observable of the system. This is called particle symmetry and has measurable physical consequences. Let

$$\psi_{ij} = \psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N). \tag{2.146}$$

The symmetry requirement under exchange of two particles is equivalent to the equation

$$|\psi_{ij}|^2 = |\psi_{ji}|^2, \tag{2.147}$$

and has the general solution

$$\psi_{ji} = e^{i\beta}\psi_{ij}, \tag{2.148}$$

where β is a real number. To find β we define a permutation operator $\hat{P}_{ij}$

$$\hat{P}_{ij}\psi_{ij} = \psi_{ji}. \tag{2.149}$$

We need to know if it is Hermitian. First note that when it operates twice we must get the same wave function back

$$\hat{P}_{ij}^2 = I \tag{2.150}$$

which means that

$$\hat{P}_{ij} = \hat{P}_{ij}^{-1}. \tag{2.151}$$

It has also been shown (reference?) that for any operator O corresponding to some observable then

$$O = \hat{P}_{ij}^{\dagger} O \hat{P}_{ij}. \tag{2.152}$$

In particular for $O = I$ we get $\hat{P}_{ij}^{\dagger}\hat{P}_{ij} = I$ which together with eq. (2.151) shows that the permutation operator is Hermitian, $\hat{P}_{ij}^{\dagger} = \hat{P}_{ij}$. Using this and eq. (2.150) means that $\hat{P}_{ij}$ commutes with any observable O, the Hamiltonian

---

[1] Identical in the sense that all of their intrinsic properties like electric charge and spin are equal. In other words, there are no experiments that could detect any intrinsic differences between them.

as well. This means that any eigenfunction ψ_k of H is also an eigenfunction of $\hat{P}_{ij}$ with eigenvalue p_{ij}. Using eq. (2.150) again we get

$$\begin{aligned} \hat{P}_{ij}^2 \psi_k &= p_{ij}^2 \psi_k && (2.153) \\ &= \psi_k && (2.154) \end{aligned}$$

which means that $p_{ij} = \pm 1$. It can also be shown that the eigenvalue p_{ij} is independent of which particles are being permutated thus $p_{ij} = p$. It is an empirical fact that wave functions with $p = 1$ are symmetric ψ_S and describes particles with whole integer spin, namely bosons. For $p = -1$ the wave function is anti-symmetric ψ_A and describes particles with half integer spin which are fermions. This suggests that $\beta = 0$ describes a symmetric state while $\beta = \pi$ describes an anti-symmetric state. We can generalize this to include all possible two-particle permutations by the operator $\hat{P}$. This can be written as

$$\hat{P} = \prod_{i=1}^{N} \prod_{j=i+1}^{N} \hat{P}_{ij} \tag{2.155}$$

and has the properties

$$\begin{aligned} \hat{P}\psi_S &= +\psi_S && (2.156) \\ \hat{P}\psi_A &= (-1)^{n_p}\psi_A && (2.157) \end{aligned}$$

where n_p is the total number of two-particle permutations done by $\hat{P}$.

2.4.1 Systems of non-interacting fermions

The Hamiltonian of such a system with N particles is

$$\hat{H}_0 = \sum_{i=1}^{N} \hat{h}_i \tag{2.158}$$

where $\hat{h}_i$ is the one particle Hamiltonian. The eigenfunction is on the product state form

$$\psi_\mu(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \prod_{i=1}^{N} \phi_{\mu_i}(\boldsymbol{x}_i) \tag{2.159}$$

where ϕ_{μ_i} is an eigenfunction of $\hat{h}_i$ and μ_i is a set of quantum numbers describing the one particle state. While ψ_k is an eigenfunction of $\hat{H}_0$ it is not an eigenfunction of $\hat{P}$. However, we know that any linear combination of different product states is also an eigenfunction of H. It turns out that the following linear combination is an eigenstate of $\hat{P}$

$$\psi_\mu(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{\mu_1}(\boldsymbol{x}_1) & \phi_{\mu_1}(\boldsymbol{x}_2) & \ldots & \phi_{\mu_1}(\boldsymbol{x}_N) \\ \phi_{\mu_2}(\boldsymbol{x}_1) & \phi_{\mu_2}(\boldsymbol{x}_2) & \ldots & \phi_{\mu_2}(\boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{\mu_N}(\boldsymbol{x}_1) & \phi_{\mu_N}(\boldsymbol{x}_2) & \ldots & \phi_{\mu_N}(\boldsymbol{x}_N) \end{vmatrix} \tag{2.160}$$

which is called the Slater determinant. If we have the case $\mu_i = \mu_j$ for some i and j then two of the rows will be equal and render the determinant zero. This tells us that two fermions cannot occupy the same state and is a consequence of the well-known Pauli exclusion principle. Changing two particles is equal to changing two columns and gives a sign change in the determinant.

Chapter 3

Numerical Methods

This chapter is devoted to the treatment of some of the most common numerical methods in quantum mechanics. Any numerical method will have to use some kind of approximation to solve the problem. In quantum mechanics it is most common to approximate either the wave function or the interaction part of the Hamiltonian.

3.1 Monte Carlo

There is no precise definition of the large range of numerical methods that falls under the Monte Carlo (MC) category. However, they can be described as statistical simulation methods in the sense that they use a sequence of random numbers in the simulation. Monte Carlo methods are used in many different fields such as chemistry, biology, physics, biology, mathematics and computational finance.

The MC methods are especially useful when the degrees of freedom in the system are many, strongly coupled and hard to simplify. Examples are fluids, disordered materials, interacting baryons such as nucleons and strongly coupled solids. In mathematics it is used in the calculation of high-dimensional integrals, often with complicated boundary conditions.

The MC method is particularly suited for studying physical systems that are governed by a probability distribution function (PDF). The reason is that we can simulate the system directly and calculate any observable in case we have an analytical expression for. This is contrary to the standard deterministic approach which usually involves finding and solving a set of partial differential equations. The statistical interpretation of the quantum mechanical wave function makes the MC method well suited here as well.

3.1.1 The variatonal principle

The expectation value of the energy is

$$\left\langle \hat{H} \right\rangle = E[\Psi_{\boldsymbol{\alpha}}] = \frac{\langle \Psi_{\boldsymbol{\alpha}} | \hat{H} | \Psi_{\boldsymbol{\alpha}} \rangle}{\langle \Psi_{\boldsymbol{\alpha}} | \Psi_{\boldsymbol{\alpha}} \rangle} = \frac{\int d\mathbf{X} \Psi^*_{\boldsymbol{\alpha}}(\mathbf{X}) H(\mathbf{X}) \Psi_{\boldsymbol{\alpha}}(\mathbf{X})}{\int d\mathbf{X} \Psi^*_{\boldsymbol{\alpha}}(\mathbf{X}) \Psi_{\boldsymbol{\alpha}}(\mathbf{X})}, \tag{3.1}$$

where $\boldsymbol{X}$ is a shorthand for the set of position vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, and $\Psi_{\boldsymbol{\alpha}}$ is a trial wave function that is parameterized with the scalars $\boldsymbol{\alpha}$. We now expand the trial wave function in the energy eigenbasis (the exact eigenvavectors of a given Hamiltonian). These eigenvectors form a complete set of orthonormal eigenfunctions

$$\Psi_{\boldsymbol{\alpha}} = \sum_i a_i \psi_i. \tag{3.2}$$

Inserting this into the expression for the energy expectation value we get

$$E[\Psi_{\boldsymbol{\alpha}}] = \frac{\displaystyle\sum_{ij} (a_i^* a_j \int \psi_i^* \hat{H} \psi_j d\boldsymbol{X})}{\displaystyle\sum_{ij} (a_i^* a_j \int \psi_i^* \psi_j d\boldsymbol{X})}. \tag{3.3}$$

Using that $\hat{H}\psi_i = E_i \psi_i$ and the orthonormality condition $\int \psi_i^* \psi_j = \delta_{ij}$ gives

$$E[\Psi_{\boldsymbol{\alpha}}] = \frac{\displaystyle\sum_i |a_i|^2 E_i}{\displaystyle\sum_i |a_i|^2} \geq E_0, \tag{3.4}$$

where E_0 is the energy of the ground state. We have therefore proved that the variational energy is an upper bound to the exact ground state energy. This important property will be used to find a good wave function. The use of the uniform distribution to sample the integral for the expectation value of the energy would lead to a highly inefficient algorithm. The sampling method must generate points where the integrand is large. To achieve this we express eq. (3.1) in terms of the PDF

$$P(\boldsymbol{X}) = \frac{|\Psi_{\boldsymbol{\alpha}}(\mathbf{X})|^2}{\int d\mathbf{X}' |\Psi_{\boldsymbol{\alpha}}(\mathbf{X}')|^2} \tag{3.5}$$

by defining a local energy as

$$E_L(\boldsymbol{X}) \equiv \frac{1}{\Psi_{\boldsymbol{\alpha}}} \hat{H} \Psi_{\boldsymbol{\alpha}}. \tag{3.6}$$

The energy expectation value can now be written as a weighted average of the local energy

$$E[\Psi_{\boldsymbol{\alpha}}] = \langle E_L \rangle_P = \int P(\boldsymbol{X}) E_L(\boldsymbol{X}) d\boldsymbol{X} \approx \frac{1}{M} \sum_{i=1}^{M} P(\boldsymbol{X}_i) E_L(\boldsymbol{X}_i), \tag{3.7}$$

where M is the number of MC cycles. We will use the Metropolis algorithm to sample from P. That is a method based on a stochastic random walk and will introduce a statistical error ϵ in our computations of the local energy. The error is equal to the standard deviation of the distribution of $\langle E_L \rangle_P$ that we get by using different samples of P in each calculation of $\langle E_L \rangle_P$. The standard deviation is given by the square root of the variance $\sigma^2_{E_L}$ of the local energy, defined as

$$\begin{aligned}
\sigma^2_{E_L} &\equiv \left\langle (E_L - \langle E_L \rangle_P)^2 \right\rangle_P && (3.8)\\
&= \left\langle E_L^2 \right\rangle_P - 2 \left\langle E_L \right\rangle_P^2 + \left\langle E_L \right\rangle_P^2 , && (3.9)\\
&= \left\langle E_L^2 \right\rangle_P - \left\langle E_L \right\rangle_P^2 . && (3.10)
\end{aligned}$$

It is easy to check that if the trial wave function is an exact eigenfunction then the variance is zero. This property can be used to find the optimal parameters because we do not always know what the lowest energy is while the lowest variance is always zero. It is possible to show (see appendix A.1) that the error is given by

$$\epsilon = \sqrt{\frac{\tau}{M}} \sigma_{E_L}. \tag{3.11}$$

where τ is the autocorrelation time. It is equal to one if there are no correlations. This means that assuming no correlations gives a too optimistic estimate of the error. The standard way of computing τ is very time consuming so we will rather use the so-called blocking method which is an approximative method that gives reliable results for the standard error and standard deviation.

3.1.2 The Metropolis algorithm

Generating a set of points that are distributed according to some known PDF can be a rather non-trivial task. The starting point is the uniform distribution generated by a pseudo random number generator and we have to transform it into the desired PDF. One technique is the inversion method which can give us an analytic transformation function. Take X as a random variate whose PDF is the uniform distribution $u(x)$. Let Y be the random variate from our desired PDF $p(y)$. The objective is to find a function f so that $f(x) = y$. It can be shown that

$$\begin{aligned} p(y) = p(f(x)) &= u(x)\left|\frac{dx}{dy}\right|, & (3.12) \\ &= u(f^{-1}(y))\left|\frac{df^{-1}(y)}{dy}\right|. & (3.13) \end{aligned}$$

By using that $u(x) = 1$ since it is defined by the uniform distribution, we get

$$p(y)dy = df^{-1}(y). \tag{3.14}$$

Integrating both sides leads to

$$f^{-1}(y) = \int_{-\infty}^{y} p(y')dy' = P(y), \tag{3.15}$$

where $P(y)$ is the cumulative probability of $p(y)$. This means that

$$f(x) = P^{-1}(x). \tag{3.16}$$

The problem with this method is that the integral of $p(y)$ must be known and invertible. If not we could generate it numerically but that leads normally to a more inefficient algorithm. One important application of the inversion method is the so-called Box-Muller algorithm which generates a pair of Gaussian random numbers (y_1, y_2) with variance σ^2 and mean value μ, given a pair of uniformly distributed random numbers x_1, x_2 as input. It can be shown that [3]

$$\begin{aligned} y_1 &= \sigma^2\sqrt{-2\ln x_1}\cos(2\pi x_2) + \mu, & (3.17) \\ y_2 &= \sigma^2\sqrt{-2\ln x_1}\sin(2\pi x_2) + \mu. & (3.18) \end{aligned}$$

which will be used in the program

The Metropolis algorithm is a way of generating a distribution by constructing a Markov chain that has the desired distribution as its equilibrium distribution (the most likely state). The Markov chain will be created by a random walk in state space with the property that the next step of the walk only depends on the current state and some random number. All information about how the current state was reached is lost. The random walker is just a mathematical object that can represent any physical quantities. In this thesis it represents the state of the electrons which are governed by their positions. In this text we will often refer to a random walker or just walker(s) when we discuss the simulation process. The random walker(s) represents thus a collection of samples in our state space of possible events. These samples form the basis for computing various expectation values like the variance of the energy or the energy.

Let the set $\{S_1, \ldots, S_N\}$ be all the available states and let S_j represent the state at time i. Let

$$P_{kj} \equiv P(S_k \leftarrow S_j) \tag{3.19}$$

be the probability of going from state S_j to S_k in one time step, where S_k represents the state at time $i+1$. In our simulations time is used here in a loose way to label the different simulation steps (typically the distance between each sampling). It has nothing to do with the physical time of the system in consideration. Normalization and positivity of the probabilities demand that

$$0 \leq \; P_{kj}^{(i)} \leq 1, \quad k = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, N \tag{3.20}$$

$$\sum_{k=1}^{N} P_{kj} = 1, \quad j = 1, 2, \ldots, N, \tag{3.21}$$

Let $p_r^{(i)}$ be the probability that the system is in state S_r at time i. The probability distribution of the walkers can be represented by the vector

$$\boldsymbol{p}^{(i)} = \begin{bmatrix} p_1^{(i)} \\ p_2^{(i)} \\ \vdots \\ p_N^{(i)} \end{bmatrix}, \tag{3.22}$$

with the requirements

$$0 \leq \; p_k^{(i)} \leq 1, \quad k = 1, 2, \ldots, N \tag{3.23}$$

$$\sum_{k=1}^{N} p_k^{(i)} = 1. \tag{3.24}$$

The evolution of $\boldsymbol{p}$ is

$$p_k^{(i+1)} = \sum_j P_{kj} p_j^{(i)}, \tag{3.25}$$

which is equal to the matrix vector equation $\boldsymbol{p}^{(i+1)} = \boldsymbol{P}\boldsymbol{p}^{(i)}$. After m steps the state is

$$\boldsymbol{p}^{(m)} = \boldsymbol{P}^m \boldsymbol{p}^{(0)} \tag{3.26}$$

Assume that there exist an equilibrium distribution $\boldsymbol{p}^*$ given by

$$\boldsymbol{p}^* = \boldsymbol{P}\boldsymbol{p}^*. \tag{3.27}$$

Obviously $\boldsymbol{p}^*$ is an eigenvector of the transition matrix $\boldsymbol{P}$ with eigenvalue 1. We want to write this in a different way by subtracting $p_k^{(i)}$ and $\sum_j P_{jk}p_k^{(i)}$ from the left and right hand side (respectively) of eq. (3.25). This is possible since $\sum_j P_{jk} = 1$. The result is

$$p_k^{(i+1)} - p_k^{(i)} = \sum_j P_{kj}p_j^{(i)} - \sum_j P_{jk}p_k^{(i)}. \tag{3.28}$$

At equilibrium we must have $p_k^{(i+1)} = p_k^{(i)}$ which leads to

$$\sum_j P_{kj}p_j^{(i)} = \sum_j P_{jk}p_k^{(i)}. \tag{3.29}$$

The stronger condition

$$P_{kj}p_j = P_{jk}p_k \tag{3.30}$$

is called detailed balance and tells us that the individual flow between pairs of states are equal. Consider now splitting the move from S_j to S_k in two steps, first the move is suggested with probability ω_{kj}, then it is accepted with probabilty A_{kj}. The total probability for moving is the product

$$\omega_{kj}A_{kj} = P_{kj}. \tag{3.31}$$

The detailed balance equation now reads

$$\frac{A_{kj}}{A_{jk}} = \frac{\omega_{jk}p_k}{\omega_{kj}p_j}. \tag{3.32}$$

Many different choices of A will satisfy this equation but the choice in the Metropolis algorithm is

$$A_{kj} = \min\left[1, \frac{\omega_{jk}p_k}{\omega_{kj}p_j}\right] \tag{3.33}$$

$$A_{jk} = \min\left[1, \frac{\omega_{kj}p_j}{\omega_{jk}p_k}\right] \tag{3.34}$$

$$A_{jj} = 1 \tag{3.35}$$

This matrix has the advantage of being very easy to implement numerically and the probability distributions need not be normalized since the normalization factors cancel when computing the ratio between probabilities. The correspoding algorithm is to generate a uniform random number r and compare it with

$$v_{kj} = \frac{\omega_{jk} p_k}{\omega_{kj} p_j}. \tag{3.36}$$

If $v_{kj} > r$ then the move is accepted. In this thesis the distribution $\boldsymbol{p}$ corresponds to $|\Psi(\boldsymbol{X})|^2$. One example of a random walk is the algorithm

$$\boldsymbol{Y} = \boldsymbol{X} + (r - 0.5)l \tag{3.37}$$

where l is a step length. It is easy to see that increasing l would decrease sequential correlation, but unfortunately this also decreases the acceptance ratio because when $\boldsymbol{Y}$ is large $|\Psi(\boldsymbol{X})|^2$ is small. A small acceptance ratio means that the particle will get stuck in the same place and the resulting energy or other expectation values would most likely be strongly biased. Generally an acceptance ratio between 0.3 and 0.7 is a good starting point but it really has to be investigated for each experiment.

The optimal solution would be to take large steps to regions were $|\Psi(\boldsymbol{X})|^2$ is large. It turns out that there exists a possible procedure for doing this based on the so-called Fokker-Planck equation. It pushes the walkers according to the gradient of the distribution. The move mimics an isotropic diffusion process with a drift force F. The random walk is now given by

$$\boldsymbol{Y} = \boldsymbol{X} + D\boldsymbol{F}(\boldsymbol{X})\delta t + \chi \tag{3.38}$$

where D is the diffusion constant and χ is a Gaussian random variable with mean value equal zero and variance $2D\delta t$. The force term is given by

$$\boldsymbol{F} = \frac{1}{p}\nabla p \tag{3.39}$$

and with $p = |\Psi|^2$ we get

$$\boldsymbol{F} = \frac{2}{\Psi}\nabla\Psi. \tag{3.40}$$

Clearly the move is biased towards large $|\Psi|^2$ and is a form of importance sampling which means sampling the states that contribute the most to the physical quantity we wish to find. It can be shown that the probability for moving from $\boldsymbol{X}$ to $\boldsymbol{Y}$ is

$$\omega(\boldsymbol{Y}, \boldsymbol{X}) = e^{\frac{-(\boldsymbol{Y} - \boldsymbol{X} - D\delta t \boldsymbol{F}(\boldsymbol{x}))^2}{4D\delta t}}. \tag{3.41}$$

The acceptance matrix is then

$$A(\boldsymbol{Y}, \boldsymbol{X}) = \min\left[1, \frac{\omega(\boldsymbol{X}, \boldsymbol{Y})|\Psi(\boldsymbol{Y})|^2}{\omega(\boldsymbol{Y}, \boldsymbol{X})|\Psi(\boldsymbol{X})|^2}\right] \tag{3.42}$$

3.1.3 Trial wave function

The choice of wave function is very important in variational Monte Carlo calculations. They should include the necessary physical properties and also be computationally feasible. In most electronic systems the typical trial wave function consists of either one or a linear combination of Slater determinants multiplied with a correlation term that is only a function of the inter-electronic or inter-particle distances. We know that any wave function can be expanded in the Slater determinant basis

$$\Psi = \sum_{\mu} c_{\mu}\psi_{\mu}. \tag{3.43}$$

where μ runs over all electronic configurations. Obviously this expansion must be terminated somewhere. The fact that the computation of the Slater determinant is usually the most demanding part limits the number of terms that are practically possible to include. If we only want the ground state energy it turns out that only one term in the expansion gives remarkably good results. This term corresponds to the ground state configuration and is an exact solution to the non-interacting system. Because the term incorporates no correlations it makes the choice of correlation term all the more important. Which single particle basis to use when defining the Slater determinant must be based on the system at hand.

One important property that the wave function should have is to fulfill the so called *cusp* condition. We know that the local energy is finite everywhere which means that the divergence in the Coulomb energy when two charged particles come close together must be cancelled by a corresponding divergence in the kinetic energy. It leads to a discontinuity on the first derivative of the wave function, hence the name cusp. For atomic systems we have both the electron-nucleus cusp and the electron-electron cusp. In the Harmonic oscillator and quantum dot case only the electron-electron cusp is present. Because the Slater determinant ψ_{μ} does not depend on r_{ij} when i and j are electrons of opposite spin, the derivative with respect to r_{ij} must be zero and ψ_{μ} cannot satisfy the cusp condition. However, we know that the determinant goes to zero when two parallel spin electrons approach each other. By using $\boldsymbol{r_{ij}} = \boldsymbol{x}_j - \boldsymbol{x}_i$ we can write the determinant as $\psi_{SD}(\boldsymbol{x}_i, \boldsymbol{x}_j, \ldots) = \psi_{SD}(\boldsymbol{x}_i, \boldsymbol{x}_i + \boldsymbol{r_{ij}}, \ldots)$. By expanding it around $r_{ij} = 0$ we get

$$\psi_{SD}(\boldsymbol{x}_i, \boldsymbol{x}_i + \boldsymbol{r_{ij}}, \ldots) \approx \psi_{SD}(\boldsymbol{x}_i, \boldsymbol{x}_i + 0, \ldots) + r_{ij}\left.\frac{\partial \psi_{SD}}{\partial r_{ij}}\right|_{r_{ij}=0} + \ldots \tag{3.44}$$

The first term is zero while the derivative in the second term is in general not zero. It is a constant ρ_{ij} that does not depend on r_{ij} (we evaluated the derivative at $r_{ij} = 0$), but it does depend on i and j. In other words, for small r_{ij} values, we can write the determinant as $\psi_{SD} = r_{ij}\rho_{ij}$. It can be shown (see [4]) that the electronic cusp condition gives an equation of the form

$$\left.\frac{1}{\rho_{ij}}\frac{\partial \rho_{ij}}{\partial r_{ij}}\right|_{r_{ij}=0} = f(l) \tag{3.45}$$

where f depends on the Schrödinger equation and $l = 1$ applies to the case of particles with equal spin values and opposite spin values, respectively. This equation can never be fulfilled by the Slater determinant which means that the correlation part must do it. If we choose a wave function on the product form $\Psi = \psi_{SD}\psi_C$ it can be shown that the cusp condition is equal to

$$\left.\frac{1}{\psi_C}\frac{\partial \psi_C}{\partial r_{ij}}\right|_{r_{ij}=0} = \begin{cases} 1 & \text{opposite spin} \\ \frac{1}{3} & \text{paralell spin} \end{cases} \tag{3.46}$$

for the two-dimensional case and

$$\left.\frac{1}{\psi_C}\frac{\partial \psi_C}{\partial r_{ij}}\right|_{r_{ij}=0} = \begin{cases} \frac{1}{2} & \text{opposite spin} \\ \frac{1}{4} & \text{paralell spin} \end{cases} \tag{3.47}$$

in three dimensions. One popular type of correlation function is the Pade-Jastrow form

$$\psi_C = e^U \tag{3.48}$$

where

$$U = \sum_{i=1}^{N}\sum_{j=i+1}^{N} u_{ij} \tag{3.49}$$

and

$$u_{ij} = \frac{\sum_{k=1}^{n} \gamma_k r_{ij}^k}{1 + \sum_{k=1}^{n} \beta_k r_{ij}^k}. \tag{3.50}$$

In this case we have

$$\left.\frac{1}{\psi_C}\frac{\partial \psi_C}{\partial r_{ij}}\right|_{r_{ij}=0} = \gamma_1 \tag{3.51}$$

for the cusp condition. We will use the Pade-Jastrow form in this thesis.

3.1.4 Optimization techniques

The problem of finding a global minima in a multidimensional function is not easy. When we add statistical noise it becomes even harder. We have tried out a method introduced by A. Harju in [5] called the Stochastic Gradient Approximation (SGA) method. It uses the statistical noise to its advantage to avoid getting stuck in a local minima. The method bears some resemblance to the Simulated Annealing technique [3]. The SGA is an iterative scheme given by the equation

$$\boldsymbol{\alpha}_{i+1} = \boldsymbol{\alpha}_i - \ell_i \nabla_{\boldsymbol{\alpha}} \hat{O}(\boldsymbol{\alpha}), \tag{3.52}$$

where α is the parameter vector for the total wave function and $\hat{O}$ is some observable like the local energy or variance. The parameter ℓ_i is a step length that should satisfy the conditions

$$\sum_{i=1} \ell_i^2 < \sum_{i=1} \ell_i. \tag{3.53}$$

A simple choice is $\ell_i = 1/i$ but we will use the more complex scheme

$$\ell_i = \ell_0 \frac{1}{j_i^k}, \tag{3.54}$$

where $j_1 = 0$, $1/2 < k \leq 1$ and

$$j_{i+1} = \begin{cases} \frac{j_i}{2} & \text{if sign}(\frac{\partial \hat{O}}{\partial \alpha_j})_i = \text{sign}(\frac{\partial \hat{O}}{\partial \alpha_j})_{i-1} \\ j_i + 1 & \text{if sign}(\frac{\partial \hat{O}}{\partial \alpha_j})_i \neq \text{sign}(\frac{\partial \hat{O}}{\partial \alpha_j})_{i-1} \end{cases}, \tag{3.55}$$

The idea is that if there is no sign change in the derivative of the j'th component of $\boldsymbol{\alpha}$ then we have not yet reached the minimum and need to increase the step length to increase efficiency. If there has been a sign change, then we have passed the minimum and must decrease the step length. The derivative of the local energy can be shown to be [6]

$$\begin{aligned} \frac{\partial E(\boldsymbol{\alpha})}{\partial \alpha_j} &= \frac{\partial}{\partial \alpha_j} \frac{\int \Psi \hat{H} \Psi}{\int \Psi^2} & (3.56) \\ &= 2\left\langle E_L \frac{\Psi'}{\Psi} \right\rangle - 2\left\langle E_L \right\rangle \left\langle \frac{\Psi'}{\Psi} \right\rangle, & (3.57) \end{aligned}$$

where

$$\Psi' = \frac{\partial \Psi}{\partial \alpha_j}. \tag{3.58}$$

Similarly, it is shown in ref. [7] that the derivative of the variance is

$$\begin{aligned} \frac{\partial \sigma^2(\boldsymbol{\alpha})}{\partial \alpha_j} &= \frac{\partial}{\partial \alpha_j} \frac{\int \Psi^2 (E_L - \langle E_L \rangle)^2}{\int \Psi^2} \qquad (3.59) \\ &= 2\Bigg[\langle E_L' (E_L - \langle E_L \rangle) \rangle + \left\langle E_L^2 \frac{\Psi'}{\Psi} \right\rangle - \left\langle E_L^2 \right\rangle \left\langle \frac{\Psi'}{\Psi} \right\rangle \\ &\quad -2 \langle E_L \rangle \left\langle (E_L - \langle E_L \rangle) \frac{\Psi'}{\Psi} \right\rangle \Bigg], \qquad (3.60) \end{aligned}$$

where

$$E_L' = \frac{\partial E_L}{\partial \alpha_j}. \qquad (3.61)$$

Variance minimization is most frequently used because it is more efficient than straightforward energy minimization. This is because it is possible to lower the energy on the finite set of MC configurations while the true expectation value is actually raised [7]. The problem with variance optimization is that the parameter set for the lowest variance may not coincide with that for the lowest energy. The SGA algorithm allows for minimizing both energy and variance and use a wheighted mean of the two sets of parameters as the optimal one. The expressions for the derivative of the energy and variance involves computing the parameter gradient of the wave function and local energy which is hard to optimize and therefore computationally costly. This will be discussed in greater detail in the implementation chapter. The number of random walkers N_W used to compute the expectation values in eq. (3.57) and eq. (3.60) controls the amount of noise in the SGA algorithm.

Chapter 4

Implementation

In this chapter we will describe how the program is constructed and discuss the advantages and disadvantages of the implementation. The most important program requirements are listed below

- Numerical efficiency
- Flexibility, extensibility and readability
- Independent of the number of space dimensions

We want a flexible program meaning that can be applied to several electronic systems such as quantum dots and atoms, with little loss of numerical efficiency. This means we have to find commonalities in these systems and build a framework around them. The use of object oriented programming (OOP) techniques makes it much easier to accomplish this and we have therefore chosen to do so. It allows us to divide the program into separate modules that have their own data structures and functions that operate on them through an interface. Since the data structures in a module cannot be directly accessed by another module, special attention must be put into their design to limit unnecessary data flow. We have chosen C++ as programming language because it offers both OOP and fast number crunching. Historically FORTRAN (F77) has been the choice of the scientific community as it was developed for this purpose. It did not originally have support for OOP but some concepts were added in (F90/F95) while full support came with F2003. In contrast C++ got full OOP support from the early 90's and, as compilers have matured and the biggest performance pitfalls have been discovered, the language has become fast enough for scientific computing. The key words here are "fast enough" because it will in most cases still be slower compared to a tailor made F77 program. But keeping in mind that a project's total time is the sum of the development time and the program run time, one risks spending more time developing highly tailored programs than what is gained in decreased run time.

A programs numerical efficiency depends on a lot of factors of which the most important ones are listed below

- Algorithm optimization
- Data flow
- Memory management
- Function overhead

Optimizing algorithms so that the computation of a variable or function are not performed several times is very important. To measure a reduction in the number of operations, it is most common to represent it as a fraction (reduced by 1/8), or as a reduction in how the algorithm scales as a function of a variable, like the number of particles in the system. In our application, the most computationally demanding part is the Slater determinant and the correlation function. Therefor separate sub-chapters has been devoted to these topics.

4.1 The Structure of the Program

The program is divided into different building blocks, each doing a specialized task. Our philosophy has been to develop as general code as possible and implement the problem dependent part in a subclass. This way the program can be reused to other systems with little extra effort. All the parameters needed are read from different files and stored in the `Parameter` class. All the other classes will then access what they need there. The flow of the program is dictated by the general Monte Carlo algorithm and is implemented in the abstract `MC_Sample` class and it mainly interacts with the the `Wave_Function_MC` class. The electron's space and spin coordinates are stored in an array of `Particle` objects.

I have on purpose not specified how the trial position and the acceptance probability should be computed. This is because these quantities depend on the choosen sampling method and should therefore be specified in a subclass.

MC algorithm

```
Generate initial randomized electron configuration.
Thermalize
loop (MC steps)
   loop (all electrons)
      Compute trial position $\mathbf{X_T}$.
      Compute acceptance probability $\mathbf{A}$
      Accept or reject move according to the Metropolis test
      Sample the desired variables.
Compute the expectation value of the sampled variables
```

4.1.1 Implementation of the MC algorithm

The following code, from the function `MC_Sampling::run()`, implements the above algorithm

```
init();//initialize random positions
thermalize();

for(int current_cycle=1; current_cycle<= nr_cycles; current_cycle++){
  for (current_walker = 1 ; current_walker <= nr_walkers; current_walker++) {

    suggest_Move();//compute trial position

    accept_prob = calc_Accept_Prob();

    //metropolis test
    if(ran2(&seed) <= accept_prob){
    accept_Move();
    accepted_steps++;
    }
    else{
      reject_Move();
    }
  }//loop over walkers

  wf->calc_Sample(local_sample);

  //save all local energies for later statistical analysis
```

```
  local_samples(current_cycle) = local_sample(1)(1);

  for(int i=1; i<=nr_sample_vectors ; i++)
    average_sample(i) += local_sample(i);

}//loop over MC cycles

for(int i=1; i<=nr_sample_vectors ; i++)
  average_sample(i) /= nr_cycles;//compute expectation values
```

So the `MC_Sample` class provides an interface for the general algorithm and the subclass must implement the following virtual functions

- `suggest_Move()`
- `calc_Accept_Prob()`
- `accept_Move()`
- `reject_Move()`

according to the used sampling method. In our case importance sampling requires the computation of the quantum force vectors and is done in the subclass `MC_Importance_Sampling`. The data structure used to store the sampled values are of type `Object_Vector<Vector>`. This is simply a vector of vectors and is needed for storing alle the derivatives of the wavefunction with respect to the variational parameters used in the optmization algorithm.

4.2 Optimization of the Slater Determinant

An efficent evaluation of the Slater determinant is crucial to the overall speed of the program. We will therefore exploit the fact that we only need the ratios

$$\frac{\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{y})}, \quad \frac{\nabla_k\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}, \quad \frac{\nabla_k\psi_{SD}(\boldsymbol{y})}{\psi_{SD}(\boldsymbol{y})}, \quad \frac{\nabla_k^2\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}, \tag{4.1}$$

and not the wavefunction itself. Since we move one particle at the time, we can utilize some simple relations between the inverse of two matrices when they are equal except for one column. We define the Slater Matrix as

$$S_{ij}(\mathbf{x}) \equiv \phi_j(x_i).$$

The cofactor of S is another matrix $C(S)$ given by

$$C_{ij} = (-1)^{i+j} M_{ij}, \tag{4.2}$$

where M_{ij} is the determinant of the submatrix obtained by removing from S the i-th row and j-th column. The determinant $|S|$ can then be written as an expansion in its cofactors along any column j

$$|S| = S_{1j}C_{1j} + S_{2j}C_{2j}+, \dots, +S_{nj}C_{nj}, \tag{4.3}$$

or along any row i

$$|S| = S_{i1}C_{i1} + S_{2i}C_{2i}+, \dots, +S_{in}C_{in}. \tag{4.4}$$

The inverse of S can be written as

$$S_{ij}^{-1} = \frac{(C_{ij})^T}{|S|} = \frac{C_{ji}}{|S|}, \tag{4.5}$$

where C^T is the transposed cofactor matrix. By definition, the Slater matrix and its inverse must satisfy the relation

$$\sum_{k=1}^{N} S_{ik}(\mathbf{x})S_{kj}^{-1}(\mathbf{x}) = \delta_{ij}. \tag{4.6}$$

The inverse matrix may therefore be used to evaluate the determinant in the usual cofactor expansion. Consider the fraction of two such expansions

$$\frac{|S(\boldsymbol{y})|}{|S(\boldsymbol{x})|} = \frac{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{y})C_{ij}(\mathbf{y})}{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{x})C_{ij}(\mathbf{x})}, \tag{4.7}$$

with $\mathbf{y}$ and $\mathbf{x}$ beeing two electron configurations that are equal except for the coordinates of electron i. From eq. (4.2) we see that the cofactors $C_{ij}(\mathbf{y})$ and $C_{ij}(\mathbf{x})$ are independent of the coordinates of electron i. Combining this with eq. (4.5) gives

$$\frac{|S(\boldsymbol{y})|}{|S(\boldsymbol{x})|} = \frac{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{y})C_{ij}(\mathbf{x})}{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{x})C_{ij}(\mathbf{x})} = \frac{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{y})S_{ji}^{-1}(\mathbf{x})}{\sum\limits_{j=1}^{N} S_{ij}(\mathbf{x})S_{ji}^{-1}(\mathbf{x})}.$$

Using eq. (4.6) we get

$$R \equiv \frac{|S(\boldsymbol{y})|}{|S(\boldsymbol{x})|} = \sum_{j=1}^{N} S_{ij}(\mathbf{y}) S_{ji}^{-1}(\mathbf{x}), \tag{4.8}$$

for the ratio R. We can also compute the derivatives in a similar way, see for example ref. [4],

$$\frac{\nabla_k \psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})} = \sum_{j=1}^{N} \nabla \phi_j(x_i) S_{ji}^{-1}(\mathbf{x}), \tag{4.9}$$

and

$$\frac{\nabla_k \psi_{SD}(\boldsymbol{y})}{\psi_{SD}(\boldsymbol{y})} = \frac{1}{R} \sum_{j=1}^{N} \nabla \phi_j(y_i) S_{ji}^{-1}(\mathbf{x}), \tag{4.10}$$

and

$$\frac{\nabla_k^2 \psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})} = \sum_{j=1}^{N} \nabla^2 \phi_j(x_i) S_{ji}^{-1}(\mathbf{x}). \tag{4.11}$$

These expressions are quite fast to compute compared to a brute force method.

The new inverse matrix is computed *only* if the move from $\mathbf{x}$ to $\mathbf{y}$ is accepted. If this move is rejected by the Monte Carlo algorithm, then no further computations are needed. The new inverse matrix is related to the old inverse matrix by, see ref. [4],

$$S_{kj}^{-1}(\mathbf{y}) = \begin{cases} S_{kj}^{-1}(\mathbf{x}) - \frac{1}{R} S_{ki}^{-1}(\mathbf{x}) \sum_{l=1}^{N} S_{il}(\mathbf{x}) S_{lj}^{-1}(\mathbf{x}) & \text{for } j \neq i, \\ \frac{1}{R} S_{ki}^{-1}(\mathbf{x}) & \text{for } j = i. \end{cases} \tag{4.12}$$

Because the hamiltonian is independent of spin, each pair of rows in the Slater matrix will be equal, resulting in $|S| = 0$. This is solved by approximating the slater determinant as

$$\psi_{SD} \approx \psi_{SD\uparrow} \cdot \psi_{SD\downarrow} \tag{4.13}$$

This product is not anti-symmetric under the exchange of electrons with opposite spin, but the authors of Ref. [8] have shown that gives it the same expextation value for the energy as the full Slater determinant as long as the interaction does not depend on spin. It also has the nice property of reducing the number of calculations by up to a factor of eight; for systems with equal numbers of spin-up and spin-down electrons, the two matrices will have dimension $\frac{N}{2} \times \frac{N}{2}$. The calculation of the inverse reduced matrix involves roughly $(N/2)^3 = N^3/8$ operations.. When moving one electron only one of these need to be updated, thereof the factor 1/8. We see that when we move only one particle the ratio R is either $\frac{\psi_{SD\uparrow}(\boldsymbol{x})}{\psi_{SD\uparrow}(\boldsymbol{y})}$ or $\frac{\psi_{SD\downarrow}(\boldsymbol{x})}{\psi_{SD\downarrow}(\boldsymbol{y})}$

4.2.1 Implementation

We have made a `Slater Matrix` class that computes the desired quantities in a brute force way. The optimizations presented above are implemented in the subclass `Slater_Matrix_MC`. They have a pointer to a `Single_Particle_Function` object that contains the basis functions ϕ_j. This way, the program can use any kind of basis functions and therefore be applied to different electronic structures. In addition to the Slater matrix itself, the class needs to store values for $\nabla\phi_j(x_i)$ and $\nabla^2\phi_j(x_i)$. While the latter is also stored in a matrix, the former is stored in a `Object_Matrix<Vector>`. In other words each matrix element is a vector object.

When splitting the full Slater matrix in the two spin component matrices, there is no longer a one-to-one mapping between the local column number and the particle number. The array `column_particle` contains this mapping while the array `particle_column` has the inverse mapping. This information is needed when getting the particle's coordinate. These arrays are made when spins of the particles are set up. The program flow of the class is shown in table 4.2.1. When the run is initialized, the variable `current_column` is set to 1. After the suggested particle move has been either accepted or rejected it is either incremented with 1 or set to 1 if this was the last particle in the cycle. This mechanism synchronizes the moving of particles between the Monte Carlo class and the Slater matrix class. The corresponding functions in the program are

- `calc_Trial_Column` computes $\phi(y_i)$ and $\nabla\phi(y_i)$
- `calc_Del_Ratio` computes $\frac{\nabla_k\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}$
- `calc_Trial_Del_Ratio` computes $\frac{\nabla_k\psi_{SD}(\boldsymbol{y})}{\psi_{SD}(\boldsymbol{y})}$
- `calc_Ratio` computes $\frac{\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{y})}$
- `calc_Del_Sq_Ratio` computes $\frac{\nabla_k^2\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}$

Slater matrix program flow

```
Compute $\phi$, $\nabla\phi$ and $\nabla^2\phi$ at initial configuration
loop (MC steps)
   loop (all electrons)
      Compute $\frac{\nabla_k\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}$.
      Compute $\phi(y_i)$ and $\nabla\phi(y_i)$
      Compute ratio $R = \frac{\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{y})}$
      Compute $\frac{\nabla_k\psi_{SD}(\boldsymbol{y})}{\psi_{SD}(\boldsymbol{y})}$
      If particle is accepted, compute $\nabla^2\phi(y_i)$ and $S^{-1}$
   Compute $\frac{\nabla_k\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}$ and $\frac{\nabla_k^2\psi_{SD}(\boldsymbol{x})}{\psi_{SD}(\boldsymbol{x})}$
```

They are declared as virtual functions and can therefore be redifined later to suit another optimization scheme if needed. Because we also have implemented the brute force way of computing these values, there is always the possibility of checking whether an optimization algorithm is correct. This allows for great flexibility for further program developments.

4.3 The Correlation Function

We will only consider correlation functions on the form

$$\psi_C = \prod_{i<j} g(r_{ij}) = \prod_{i<j} g_{ij}, \tag{4.14}$$

where r_{ij} is the inter-electronic distance and $\prod_{i<j}$ is a shorthand for $\prod_{i=1}^{N}\prod_{j=i+1}^{N}$.
We will focus on one class of such correlation functions which is called Padé-Jastrow, with the closed-for expression

$$g_{ij} = e^{f_{ij}}. \tag{4.15}$$

The gradient of such a function will usually need evaluation of the inter-electronic distance vector $\boldsymbol{r}_{ij}$. The variables r_{ij} and $\boldsymbol{r}_{ij}$ are given by

$$\begin{aligned}\boldsymbol{r}_{ij} &= \boldsymbol{x}_i - \boldsymbol{x}_j &(4.16)\\ r_{ij} &= |\boldsymbol{r}_{ij}| = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^2} &(4.17)\end{aligned}$$

They can be represented as matrices

$$r_{ij} = \begin{bmatrix} 0 & r_{12} & r_{13} & \dots & r_{1i} & \dots & \dots & r_{1N} \\ r_{12} & 0 & r_{23} & \dots & r_{2i} & \dots & \dots & r_{2N} \\ r_{13} & r_{12} & 0 & \ddots & \vdots & \dots & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 & r_{(i-1)i} & \dots & \dots & \vdots \\ r_{1i} & r_{1i} & \dots & r_{(i-1)i} & 0 & r_{i(i+1)} & \dots & r_{iN} \\ \vdots & \vdots & \vdots & \vdots & r_{i(i+1)} & 0 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0 & r_{(N-1)N} \\ r_{1N} & r_{1(N-1)} & \dots & \dots & \dots & \dots & r_{(N-1)N} & 0 \end{bmatrix}, \tag{4.18}$$

and

$$\boldsymbol{r}_{ij} = \begin{bmatrix} 0 & \boldsymbol{r}_{12} & \boldsymbol{r}_{13} & \dots & \boldsymbol{r}_{1i} & \dots & \dots & \boldsymbol{r}_{1N} \\ -\boldsymbol{r}_{12} & 0 & \boldsymbol{r}_{23} & \dots & \boldsymbol{r}_{2i} & \dots & \dots & \boldsymbol{r}_{2N} \\ -\boldsymbol{r}_{13} & -\boldsymbol{r}_{12} & 0 & \ddots & \vdots & \dots & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 & \boldsymbol{r}_{(i-1)i} & \dots & \dots & \vdots \\ -\boldsymbol{r}_{1i} & -\boldsymbol{r}_{1i} & \dots & -\boldsymbol{r}_{(i-1)i} & 0 & \boldsymbol{r}_{i(i+1)} & \dots & \boldsymbol{r}_{iN} \\ \vdots & \vdots & \vdots & \vdots & -\boldsymbol{r}_{i(i+1)} & 0 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0 & \boldsymbol{r}_{(N-1)N} \\ -\boldsymbol{r}_{1N} & -\boldsymbol{r}_{1(N-1)} & \dots & \dots & \dots & \dots & -\boldsymbol{r}_{(N-1)N} & 0 \end{bmatrix}. \tag{4.19}$$

It is easy to see that we only need $N-1$ updates in each matrix when a particle has been moved. Numerical computation of the correlation function is usually less expensive than for the Slater determinant, but on the other hand, there are less optimization possibilities. Again we exploit that we only need to compute the following ratios

$$R \equiv \frac{\psi_C^{\text{new}}}{\psi_C^{\text{curr}}},, \tag{4.20}$$

and

$$\frac{\nabla \psi_C}{\psi_C}, \quad \frac{\nabla^2 \psi_C}{\psi_C}. \tag{4.21}$$

First we consider the calculation of R. Moving one particle at a time requires $N-1$ updates of the g_{ij} matrix. When computing the ratio R, only the updated entries will be needed. More formally, when moving electron k ψ_C becomes

$$\psi_C = \prod_{i=1}^{N} \prod_{j=i+1}^{N} g_{ij} = \prod_{i=1}^{k-1} \prod_{j=i+1}^{N} g_{ij} \cdot \prod_{j=k+1}^{N} g_{kj} \cdot \prod_{i=k+1}^{N} \prod_{j=k+2}^{N} g_{ij}, \tag{4.22}$$

the last term does not include g_{ik} and will cancel in the ratio. However, the first term includes terms both with g_{ik} and without. We can isolate them by using the same trick as above

$$\prod_{i=1}^{k-1}\prod_{j=i+1}^{N} g_{ij} = \prod_{i=1}^{k-1}\prod_{j=i+1}^{k-1} g_{ij} \cdot \prod_{i=1}^{k-1} g_{ik} \cdot \prod_{i=1}^{k-1}\prod_{j=k+1}^{N} g_{ij}. \tag{4.23}$$

Only the middle terms of eq. (4.22) and eq. (4.23) will contribute and so the ratio becomes

$$R = \prod_{i=1}^{k-1} \frac{g_{ik}^{\text{new}}}{g_{ik}^{\text{curr}}} \prod_{i=k+1}^{N} \frac{g_{ki}^{\text{new}}}{g_{ki}^{\text{curr}}}. \tag{4.24}$$

The Padé-Jastrow ratio is

$$\begin{aligned} R &= \prod_{i=1}^{k-1} \frac{e^{f_{ik}^{\text{new}}}}{e^{f_{ik}^{\text{curr}}}} \prod_{i=k+1}^{N} \frac{e^{f_{ki}^{\text{new}}}}{e^{f_{ki}^{\text{curr}}}} & (4.25)\\ &= \exp\left[\sum_{i=1}^{k-1}(f_{ik}^{\text{new}} - f_{ik}^{\text{curr}}) + \sum_{i=k+1}^{N}(f_{ik}^{\text{new}} - f_{ik}^{\text{curr}})\right]. & (4.26) \end{aligned}$$

Calculating the gradient is a bit more tricky, but we start with writing ψ_C as

$$\psi_C = \prod_{i=1}^{N}\prod_{j=i+1}^{N} g_{ij} = G\prod_{i=1}^{k-1} g_{ik} \prod_{i=k+1}^{N} g_{ki}, \tag{4.27}$$

where G is the product of all the terms witout k. The derivative of this function is

$$\begin{aligned} \frac{\partial \psi_C}{\partial x_k} &= G\frac{\partial}{\partial x_k}\left(\prod_{i=1}^{k-1} g_{ik} \prod_{i=k+1}^{N} g_{ki}\right) & (4.28)\\ &= G\left(\prod_{i=1}^{k-1} g_{ik} \frac{\partial}{\partial x_k}\prod_{i=k+1}^{N} g_{ki} + \prod_{i=k+1}^{N} g_{ki} \frac{\partial}{\partial x_k}\prod_{i=1}^{k-1} g_{ik}\right). & (4.29) \end{aligned}$$

Furthermore

$$\begin{aligned}
\frac{\partial}{\partial x_k}\prod_{i=1}^{k-1} g_{ik} &= \frac{\partial}{\partial x_k}(g_{1k}\, g_{2k}\, \cdots\, g_{(k-1)k}) && (4.30)\\
&= \big(g_{2k}\, g_{3k}\, \cdots\, g_{(k-1)k}\big)\,\frac{\partial g_{1k}}{\partial x_k} + \\
&\quad \big(g_{1k}\, g_{3k}\, \cdots\, g_{(k-1)k}\big)\,\frac{\partial g_{2k}}{\partial x_k} + \cdots + \\
&\quad \big(g_{1k}\, g_{2k}\, \cdots\, g_{(i-1)k}\, g_{(i+1)k}\, \cdots\, g_{(k-1)k}\big)\,\frac{\partial g_{ik}}{\partial x_k} + \cdots + \\
&\quad \big(g_{1k}\, g_{2k}\, \cdots\, g_{(k-2)k}\big)\,\frac{\partial g_{(k-1)k}}{\partial x_k} && (4.31)\\
&= (g_{1k}\, g_{2k}\, \cdots\, g_{(k-1)k})\sum_{i=1}^{k-1}\frac{1}{g_{ik}}\frac{\partial g_{ik}}{\partial x_k} && (4.32)\\
&= \left(\prod_{i=1}^{k-1} g_{ik}\right)\sum_{i=1}^{k-1}\frac{1}{g_{ik}}\frac{\partial g_{ik}}{\partial x_k}, && (4.33)
\end{aligned}$$

where the step from eq. (4.31) to eq. (4.32) is done by multiplying each term with g_{ik}/g_{ik}. In the same way

$$\frac{\partial}{\partial x_k}\prod_{i=k+1}^{N} g_{ki} = \left(\prod_{i=k+1}^{N} g_{ki}\right)\sum_{i=k+1}^{N}\frac{1}{g_{ki}}\frac{\partial g_{ki}}{\partial x_k}. \tag{4.34}$$

Using eq. (4.33) and eq. (4.34) we can write eq. (4.29) as

$$\begin{aligned}
\frac{\partial \psi_C}{\partial x_k} &= G\left(\prod_{i=1}^{k-1} g_{ik}\prod_{i=k+1}^{N} g_{ki}\right)\left(\sum_{i=1}^{k-1}\frac{1}{g_{ik}}\frac{\partial g_{ik}}{\partial x_k} + \sum_{i=k+1}^{N}\frac{1}{g_{ki}}\frac{\partial g_{ki}}{\partial x_k}\right) && (4.35)\\
&= \psi_C\left(\sum_{i=1}^{k-1}\frac{1}{g_{ik}}\frac{\partial g_{ik}}{\partial x_k} + \sum_{i=k+1}^{N}\frac{1}{g_{ki}}\frac{\partial g_{ki}}{\partial x_k}\right). && (4.36)
\end{aligned}$$

The derivatives $\frac{\partial g_{ki}}{\partial x_k}$ can be computed numerically using the two point finite difference formula

$$\frac{\partial g_{ki}}{\partial x_k} = \frac{g(x_k + h) - g(x_k - h)}{2h}. \tag{4.37}$$

This requires direct access to the coordinates and a lot of programming. A trick is to use the chain rule

$$\frac{\partial g_{ij}}{\partial x_i} = \frac{\partial g_{ij}}{\partial r_{ij}}\frac{\partial r_{ij}}{\partial x_i}, \tag{4.38}$$

where

$$\begin{aligned}\frac{\partial r_{ij}}{\partial x_i} &= \frac{\partial}{\partial x_i}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + \dots} && (4.39)\\ &= \frac{x_i - x_j}{r_{ij}}. && (4.40)\end{aligned}$$

The gradient ratio will contain terms of the type

$$\sum_{i=k+1}^{N}\left(\frac{1}{g_{ki}}\frac{\partial g_{ki}}{\partial r_{ki}}\frac{(x_k - x_i)}{r_{ki}}\right)\hat{x} + \sum_{i=k+1}^{N}\left(\frac{1}{g_{ki}}\frac{\partial g_{ki}}{\partial r_{ki}}\frac{(y_k - y_i)}{r_{ki}}\right)\hat{y} + \dots \tag{4.41}$$

which is equal to the simpler expression

$$\sum_{i=k+1}^{N}\frac{\boldsymbol{r}_{ki}}{g_{ik}r_{ki}}\frac{\partial g_{ki}}{\partial r_{ki}}. \tag{4.42}$$

Thus the gradient ratio becomes

$$\frac{\nabla_k \psi_C}{\psi_C} = -\sum_{i=1}^{k-1}\frac{\boldsymbol{r}_{ik}}{g_{ik}r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}} + \sum_{i=k+1}^{N}\frac{\boldsymbol{r}_{ki}}{g_{ki}r_{ki}}\frac{\partial g_{ki}}{\partial r_{ki}} \tag{4.43}$$

where the minus sign in the first term stems from the relationship

$$\frac{\partial r_{ij}}{\partial x_j} = -\frac{\partial r_{ij}}{\partial x_i}. \tag{4.44}$$

Finding the Padé-Jastrow gradient ratio is now just a matter of a simple derivative

$$\begin{aligned}\frac{\partial g_{ij}}{\partial r_{ij}} &= \frac{\partial}{\partial r_{ij}}e^{f_{ij}} && (4.45)\\ &= g_{ij}\frac{\partial f_{ij}}{\partial r_{ij}} && (4.46)\end{aligned}$$

which inserted into eq. (4.43) gives

$$\frac{\nabla_k \psi_{PJ}}{\psi_{PJ}} = -\sum_{i=1}^{k-1}\frac{\boldsymbol{r}_{ik}}{r_{ik}}\frac{\partial f_{ik}}{\partial r_{ik}} + \sum_{i=k+1}^{N}\frac{\boldsymbol{r}_{ki}}{r_{ki}}\frac{\partial f_{ki}}{\partial r_{ki}} \tag{4.47}$$

The result for the Laplacian is more complicated, but we start with rewriting eq. (4.43) as

$$\nabla_k \psi_C = \psi_C\left(\sum_{i=1}^{k-1}\frac{1}{g_{ik}}\nabla_k g_{ik} + \sum_{i=k+1}^{N}\frac{1}{g_{ki}}\nabla_k g_{ki}\right). \tag{4.48}$$

By using the relation

$$\nabla \cdot (a\boldsymbol{b}) = \nabla a \cdot \boldsymbol{b} + a\nabla \cdot \boldsymbol{b}, \tag{4.49}$$

the Laplacian becomes

$$\begin{aligned}\nabla_k^2 \psi_C &= \nabla \psi_C \cdot \left(\sum_{i=1}^{k-1} \frac{1}{g_{ik}} \nabla_k g_{ik} + \sum_{i=k+1}^{N} \frac{1}{g_{ki}} \nabla_k g_{ki} \right) + \\ &\qquad \psi_C \nabla \cdot \left(\sum_{i=1}^{k-1} \frac{1}{g_{ik}} \nabla_k g_{ik} + \sum_{i=k+1}^{N} \frac{1}{g_{ki}} \nabla_k g_{ki} \right). \end{aligned} \tag{4.50}$$

The first term is just

$$\psi_C \left(\frac{\nabla_k \psi_C}{\psi_C} \right)^2, \tag{4.51}$$

while the dot product in the second term is a bit more tricky. What we need to find is

$$\begin{aligned}\nabla_k \cdot \left(\frac{1}{g_{ik}} \nabla_k g_{ik} \right) &= \nabla_k \frac{1}{g_{ik}} \cdot \nabla_k g_{ik} + \\ &\qquad \frac{1}{g_{ik}} \nabla_k \cdot \nabla_k g_{ik}, \end{aligned} \tag{4.52}$$

where

$$\nabla_k \frac{1}{g_{ik}} = -\frac{1}{g_{ik}^2} \nabla_k g_{ik}, \tag{4.53}$$

and

$$\nabla_k \cdot \nabla_k g_{ik} = \nabla_k \cdot \left(-\frac{\boldsymbol{r}_{ik}}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} \right) \tag{4.54}$$

$$\begin{aligned} &= -\nabla_k \left(\frac{1}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} \right) \cdot \boldsymbol{r}_{ik} - \\ &\qquad \left(\frac{1}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} \right) \nabla_k \cdot \boldsymbol{r}_{ik}. \end{aligned} \tag{4.55}$$

Furthermore

$$\nabla_k \left(\frac{1}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} \right) = -\frac{\boldsymbol{r}_{ik}}{r_{ik}} \frac{\partial}{\partial r_{ik}} \left(\frac{1}{r_{ik}} \frac{\partial g_{ik}}{\partial r_{ik}} \right) \tag{4.56}$$

$$= -\frac{\boldsymbol{r}_{ik}}{r_{ik}} \left(-\frac{1}{r_{ik}^2} \frac{\partial g_{ik}}{\partial r_{ik}} + \frac{1}{r_{ik}} \frac{\partial^2 g_{ik}}{\partial r_{ik}^2} \right) \tag{4.57}$$

and it is easy to show that $\nabla_k \cdot \boldsymbol{r}_{ik} = -d$, where d is the number of space dimensions. Equation (4.55) now becomes

$$\begin{aligned}\nabla_k \cdot \nabla_k g_{ik} &= r_{ik}\left(-\frac{1}{r_{ik}^2}\frac{\partial g_{ik}}{\partial r_{ik}} + \frac{1}{r_{ik}}\frac{\partial^2 g_{ik}}{\partial r_{ik}^2}\right) + \frac{d}{r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}} && (4.58)\\ &= \frac{(d-1)}{r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}} + \frac{\partial^2 g_{ik}}{\partial r_{ik}^2}. && (4.59)\end{aligned}$$

Using this result, eq. (4.53) and

$$\begin{aligned}(\nabla_k g_{ik})^2 &= \left(-\frac{\boldsymbol{r}_{ik}}{r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}}\right)^2 && (4.60)\\ &= \left(\frac{\partial g_{ik}}{\partial r_{ik}}\right)^2 && (4.61)\end{aligned}$$

we can find the final expression for eq. (4.52)

$$\begin{aligned}\nabla_k \cdot \left(\frac{1}{g_{ik}}\nabla_k g_{ik}\right) &= -\frac{1}{g_{ik}^2}\left(\frac{\partial g_{ik}}{\partial r_{ik}}\right)^2 + \\ &\quad \frac{1}{g_{ik}}\left(\frac{(d-1)}{r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}} + \frac{\partial^2 g_{ik}}{\partial r_{ik}^2}\right). && (4.62)\end{aligned}$$

Inserting this result and eq. (4.51) into eq. (4.50) we get for the Laplacian ratio

$$\begin{aligned}\frac{\nabla_k^2 \psi_C}{\psi_C} &= \left(\frac{\nabla_k \psi_C}{\psi_C}\right)^2 + \sum_{i=1}^{k-1} -\frac{1}{g_{ik}^2}\left(\frac{\partial g_{ik}}{\partial r_{ik}}\right)^2 + \\ &\quad \frac{1}{g_{ik}}\left(\frac{(d-1)}{r_{ik}}\frac{\partial g_{ik}}{\partial r_{ik}} + \frac{\partial^2 g_{ik}}{\partial r_{ik}^2}\right) + \\ &\quad \sum_{i=k+1}^{N} -\frac{1}{g_{ki}^2}\left(\frac{\partial g_{ki}}{\partial r_{ki}}\right)^2 + \\ &\quad \frac{1}{g_{ki}}\left(\frac{(d-1)}{r_{ki}}\frac{\partial g_{ki}}{\partial r_{ki}} + \frac{\partial^2 g_{ki}}{\partial r_{ki}^2}\right). && (4.63)\end{aligned}$$

The Padé-Jastrow expression is a bit simpler

$$\begin{aligned}\frac{\nabla_k^2 \psi_{PJ}}{\psi_{PJ}} &= \left(\frac{\nabla_k \psi_{PJ}}{\psi_{PJ}}\right)^2 + \sum_{i=1}^{k-1}\frac{1}{f_{ik}}\left(\frac{(d-1)}{f_{ik}}\frac{\partial f_{ik}}{\partial r_{ik}} + \frac{\partial^2 f_{ik}}{\partial r_{ik}^2}\right) + \\ &\quad \sum_{i=k+1}^{N}\frac{1}{f_{ki}}\left(\frac{(d-1)}{r_{ki}}\frac{\partial f_{ki}}{\partial r_{ki}} + \frac{\partial^2 f_{ki}}{\partial r_{ki}^2}\right). && (4.64)\end{aligned}$$

The advantage of only using the derivatives in r_{ij} is that it is independent on the number of dimensions. The reason for not writing out eq. (4.51) is that when computing the local energy we will need both the gradient and the Laplacian. By computing the gradient first, we can just take the dot product with itself when computing the Laplacian afterwards.

Chapter 5

Results

In this chapter we will verify that the implementation is correct by comparing with known analytical cases. The ground state energy of systems with the one and two filled harmonic oscillator shells in both two and three dimensions are computed. With one filled shell (the lowest shell) this corresponds to a two-particle problem in two and three dimensions. With the two lowest shells filled, this corresponds to a six-electron problem in two dimensions and an eight-electron problem in three dimensions. The results are also applicable to quantum dots for weak magnetic fields because the energy splitting will cancel and the only difference in the energy is the shifted harmonic oscillator frequency. We will use atomic units [9] where $\hbar = e = 4\pi\epsilon = 1$ and drop the normalization factor since we only compute wave function ratios in the Metropolis algorithm. The one particle wave functions that are given by eq. (2.40) include now the variational parameters $\boldsymbol{\alpha}$ by letting $\omega_x \rightarrow \alpha_x\omega_x$ and so forth. The resulting single-particle functions are

$$\phi_{n_x,n_y}(x,y) = e^{-\frac{1}{2}(\alpha_x\omega_x x^2+\alpha_y\omega_y y^2)} H_{n_x}(\sqrt{\alpha_x\omega_x}x) H_{n_y}(\sqrt{\alpha_y\omega_y}y), \tag{5.1}$$

in two dimensions and

$$\begin{aligned}\phi_{n_x,n_y,n_z}(x,y,z) &= e^{-\frac{1}{2}(\alpha_x\omega_x x^2+\alpha_y\omega_y y^2+\alpha_z\omega_z z^2)} \\ &\quad \times H_{n_x}(\sqrt{\alpha_x\omega_x}x) H_{n_y}(\sqrt{\alpha_y\omega_y}y) H_{n_z}(\sqrt{\alpha_z\omega_z}z), \end{aligned} \tag{5.2}$$

in three dimensions. The two Hermite polynomials we need are listed below for easy reference

$$H_0(\sqrt{\alpha\omega}x) = 1, \tag{5.3}$$

$$H_1(\sqrt{\alpha\omega}x) = 2\sqrt{\alpha\omega}x. \tag{5.4}$$

$$\tag{5.5}$$

The total variational wave function is the product of the ground state configuration Slater determinant $\psi_{SD}(\boldsymbol{X}, \boldsymbol{\alpha})$ and the Pade-Jastrow correlation function $\psi_C(r_{ij}, \boldsymbol{\beta}, \boldsymbol{\gamma})$

$$\Psi(\boldsymbol{X}) = \psi_{SD}(\boldsymbol{X}, \boldsymbol{\alpha})\psi_C(r_{ij}, \boldsymbol{\beta}, \boldsymbol{\gamma}). \tag{5.6}$$

The linear Pade-Jastrow is used unless otherwise stated. This results in only one variational parameter β in the correlation part. The harmonic oscillator frequency is taken to be isotropic which means we only need one variational parameter α in the Slater determinant. The uncorrelated energies are for the case with filled harmonic oscillator shells This will be compared to the correlated energies below.

N	d	E
0	2	2ω
0	3	3ω
1	2	10ω
1	3	18ω

Table 5.1: Energy in atomic units for filled (N) harmonic oscillator shells or quantum dot in d dimensions

5.1 Two electrons

First we did a quick brute force calculation of the energy for the parameters

$$\alpha = 0.1, 0.15, \ldots, 1.6 \tag{5.7}$$

$$\beta = 0.0, 0.05, \ldots, 1.0 \tag{5.8}$$

with three random walkers and only 30000 MC cycles. The oscillator frequency is $\omega = 1$ and $\omega = 0.5$ in two and three dimensions respectively. The result is shown in fig. 5.1 and fig. 5.2. We see that the energy decreases as α increases to about 1. The optimal β is a bit harder to find as the energy does not seem to vary much in that area. By taking a two-dimensional slice of the same plot shown in fig. 5.3 and fig. 5.4 we see that in two dimensions there is clearly a minimum close to $\beta = 0.4$ while in three dimensions the energy is nearly constant for $\beta > 0.4$. For the optimization we wanted to see how effective the SGA algorithm is when the starting point was far from the minimum and decided to use $[\alpha_0, \beta_0] = [0.2, 1]$. The resulting parameter plots are shown in fig. 5.5 - 5.8. The optimal parameters we found are shown in table 5.2 together with the energy, uncorrelated variance and error. These energies are extremely close to the analytical result and verifies that our code is correct at least for two particles. The big test is moving on to more particles because that is when the whole Slater determinant machinery is really put to the test. The small variance tell us that the trial wave function is extremely close to the correct one which means the linear Pade-Jastrow function incorporates almost all the correlations that are lost when truncating the expansion of the wave function in the Slater determinant basis.

d	ω	α_{100}	β_{100}	E	σ^2	ϵ
2	1	0.99044	0.39994	3.00032	0.00183	$3.5e-05$
3	0.5	0.99425	0.20063	2.00009	0.00016	$1.5e-05$

Table 5.2: Energy for the two electron harmonic oscillator in 2 and 3 dimensions at the optimal variational parameters. The energy optimizations, which is shown in fig. 5.5 - 5.8, were obtained by using 3 random walkers and 10000 MC cycles at each iteration. The energy is then computed still using 3 random walkers but now increasing the number of MC cycles to 10 million. The error ϵ is found from the blocking plot shown in fig. 5.9 - 5.10. The energies are in perfect correspondence with the analytical result presented by M. Taut in [1] which is also discussed in 2.2.3

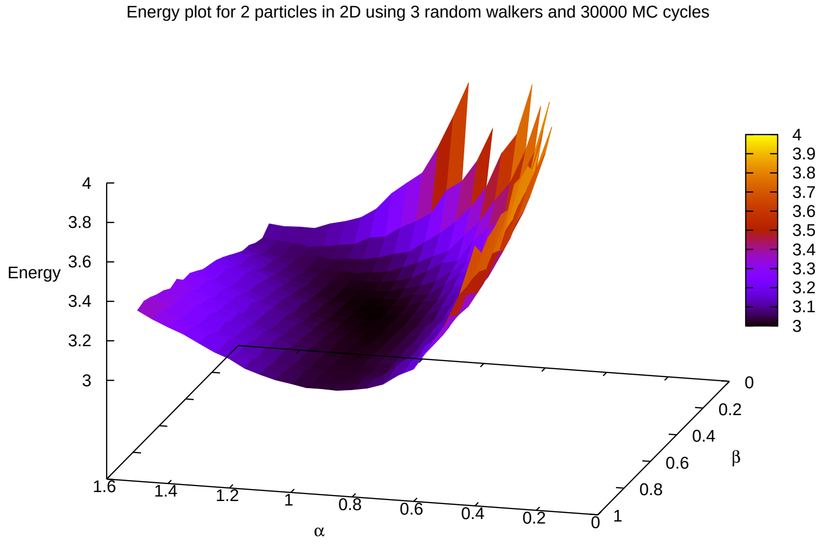

Figure 5.1: This is a plot of the energy in a two dimensional harmonic oscillator with $\omega = 1/2$, as function of the variational parameters α and β. The dark area suggests we should look for the energy minimum around $\alpha = 1$ while it is not so clear what the optimal β is. For $\alpha \approx 1$ the energy seem to have a low β dependence. This is confirmed later in fig. 5.3

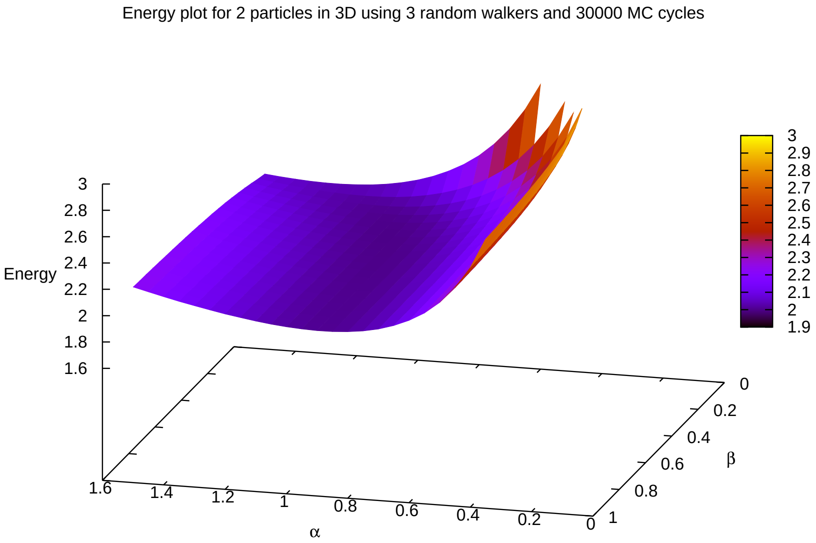

Figure 5.2: This is a plot of the energy in a three dimensional harmonic oscillator with $\omega = 1/2$ as a function of the variational parameters α and β. It resembles the shape of the two dimensional energy as it goes to infinity when $\alpha \rightarrow 0$ and has a minimum around $\alpha = 1$.

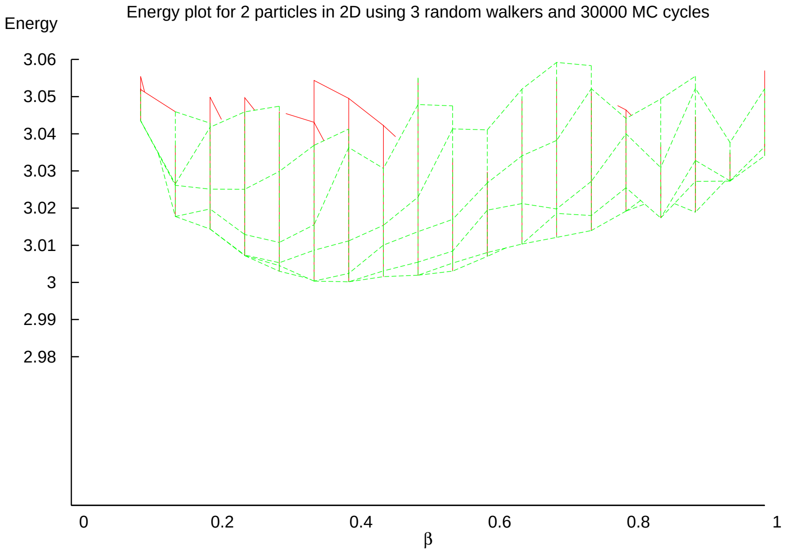

Figure 5.3: Two-dimensional slice of a three-dimensional plot of the energy as a function of the variational parameters from which we only see β. The green part increases towards the viewer while the red part increases in the opposite direction. The plot clearly shows an energy minimum around $\beta = 0.35$ which can be a good starting point for optimizing the energy.

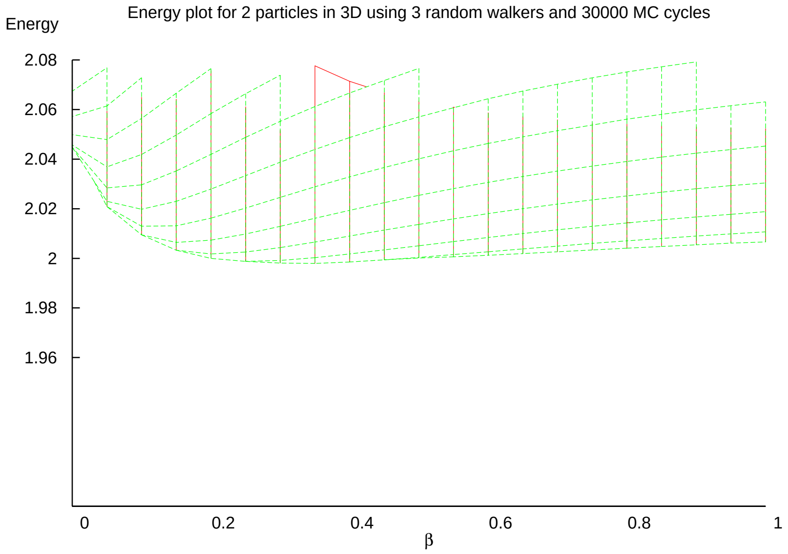

Figure 5.4: 2D slice of a 3D plot of the three dimensional energy as a function of the variational parameters rotated to only show β. Comparing with the two dimensional case in fig. 5.3 we see that the energy in three dimensions is less sensitive with respect to β which is the variational parameter for the correlation part of the wave function. This is both because ω in our two dimensional computation is twice as large and due to correlations being more important in lower dimensional systems.

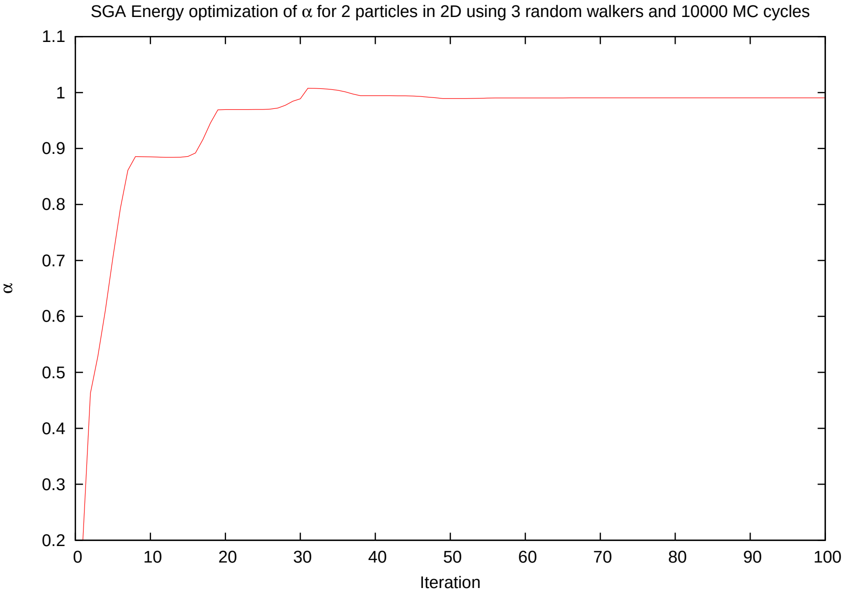

Figure 5.5: Here we see how α iterates towards the minimum value close to 1 as indicated in fig. 5.1. The method produces a jump like pattern which is due to the adaptive step algorithm were we have a variable that increases or decreases the step length according to the sign of the derivative of the wave function with respect to the variational parameters. This is described in greater detail in 3.1.4.

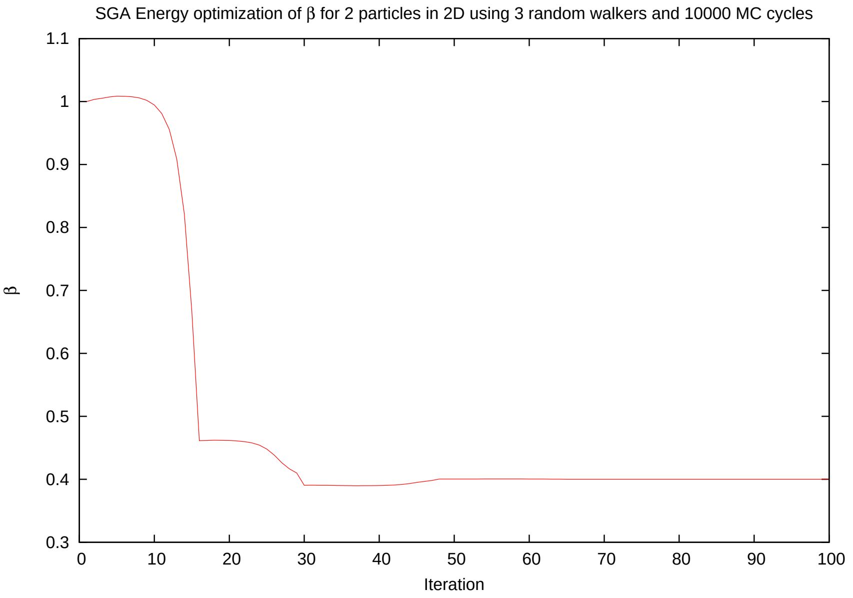

Figure 5.6: The optimization of the correlation parameter is pretty fast and converges after about 50 iterations which is quite fast considering we are only using 10000 MC cycles at each iteration. Also we choose a starting point which is quite far away and when also taking into account that the energy is almost constant in that direction, makes the SGA algorithm a worthy candidate for future considerations.

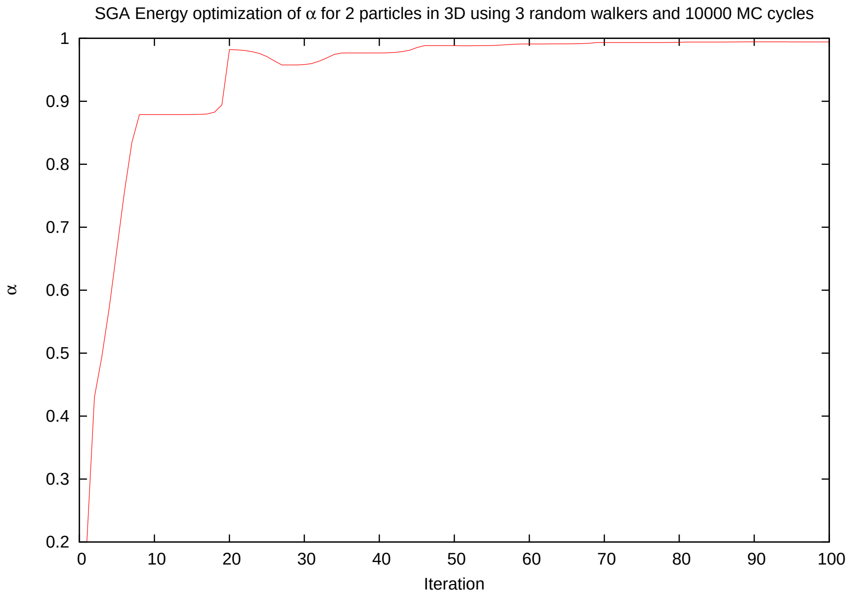

Figure 5.7: The Slater determinant variational parameter approaches unity.

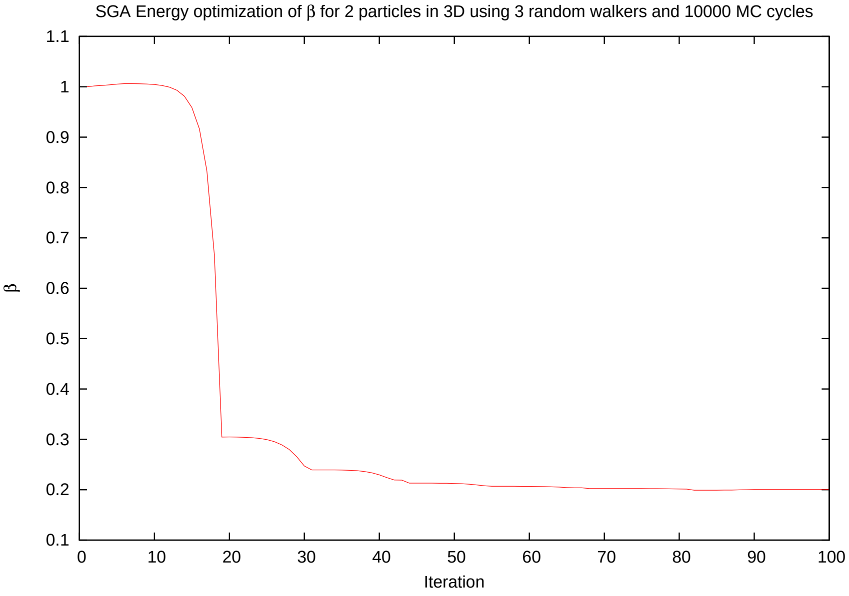

Figure 5.8: We see the same here as for the two particle case in which the optimal parameter is found quickly.

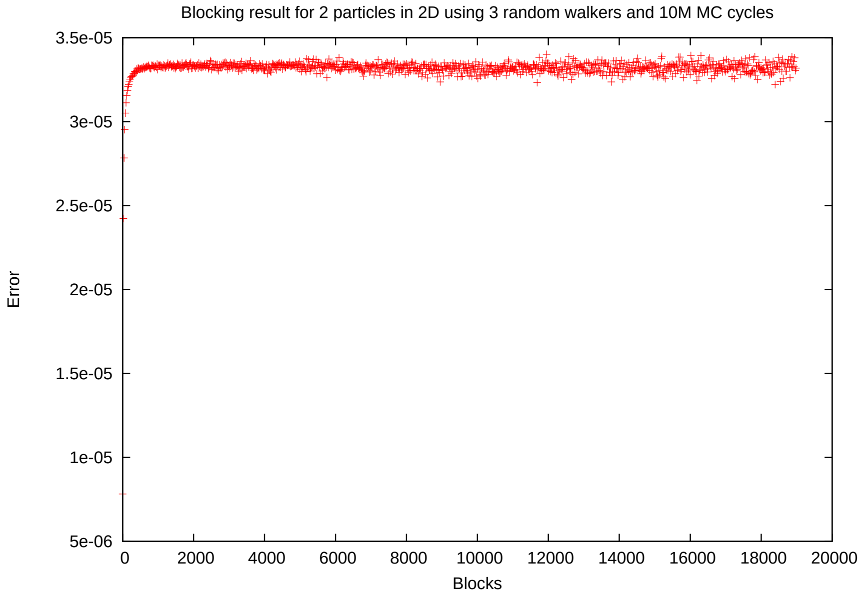

Figure 5.9: We see here the standard deviation of the distribution of the average of local energy as a function of the number of blocks the sample is divided into. When the number of blocks increases the distance between the i'th value in each block decreases which increases the sequentially correlation. This is an easy way of finding a good estimate for the error in a correlated data set. More details can be found in [10]

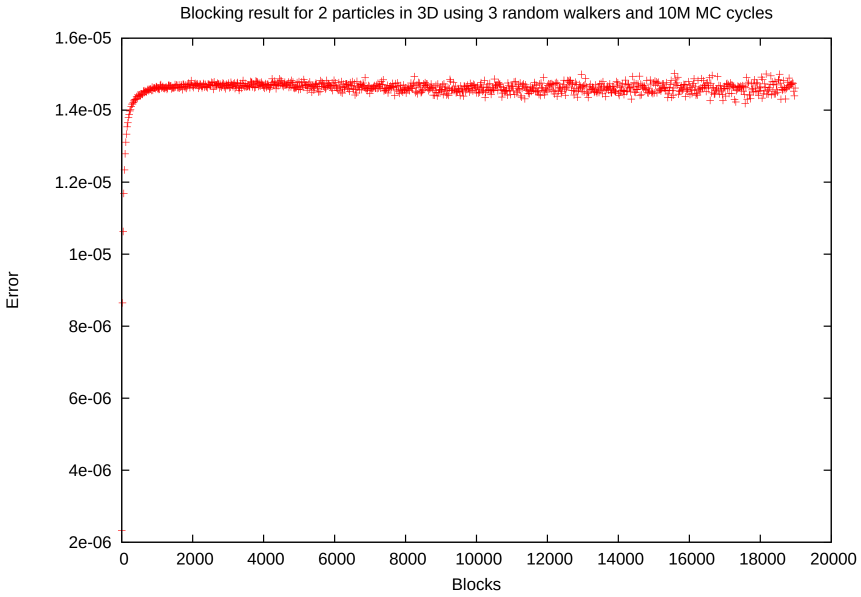

Figure 5.10: The error here is a bit smaller than in two dimensions because the correlations are less present in higher dimensional system. It represents the physical fact that the less space the electron can occupy the more often will it 'feel' the repulsion of the other electron hence they are more correlated.

5.2 Six and eight electrons

We now move to the next full shell system which in two and three dimensions have six and eight electrons, respectively. We have now used $\omega = 1$ in both dimensions since there are no analytical answers we can compare with.

We have first used a brute force calculation as in the two particle case to get a sense of how the energy behaves and this is shown in fig. 5.11 and fig. 5.12. The energy behaves in a similar way as the two particle case as it goes to infinity when the parameters are low. The optimal α seems a bit lower while β is again hard to pinpoint. Again we have rotated the previous plot so that it in effect becomes two dimensional with β as the only variational parameter, see fig. (5.13) and fig. (5.14). Apart from some rather large statistical fluctuations the β dependence of the energies β is weaker compared with the two particle case.

When searching for the energy minimum we now start at $[\alpha_0, \beta_0] = [1.0, 0.2]$. The optimal parameters we get are then used to compute the energy as in the two particle case. The results are displayed in table 5.3. We notice the 100% and 50% increase in energy compared to the non-interacting case.

d	ω	α_{100}	β_{100}	E	ϵ
2	1	0.926273	0.561221	20.1910	$3.5e-4$
3	1	0.961791	0.372215	32.6829	$2.4e-4$

Table 5.3: Energy for the six and eight electron harmonic oscillator in 2 and 3 dimensions, respectively. The increase in energy by including electron interactions is over 100% in two dimensions while in three dimensions it is over 50%. This is roughly the same increase as in the two particle case. Simen Kvaal [11] has used the CI method to solve this and obtained an energy of 20.1882 which is very close to our result and verifies that our code is correct.

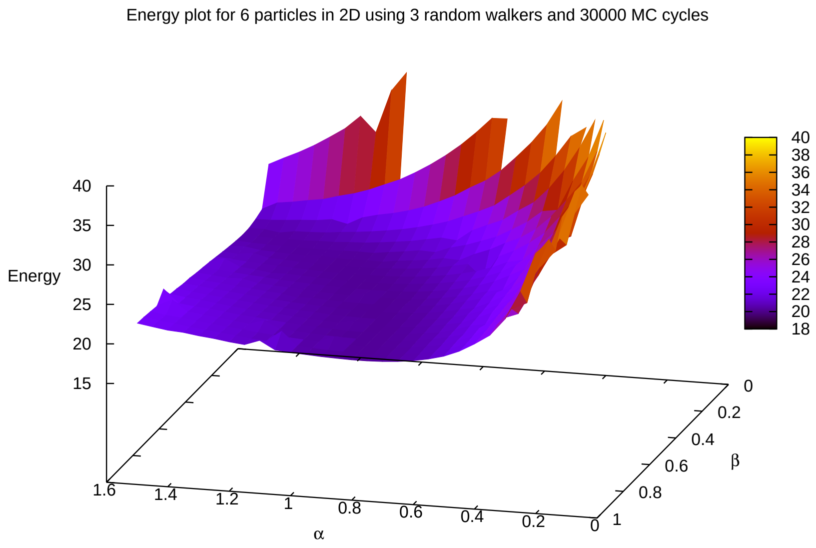

Figure 5.11: We recognize the energy shape from the two particle case as it is quite similar. Again there is this L shaped dark region where the energy is lowest. This makes it easier to choose a starting point for the SGA approximation.

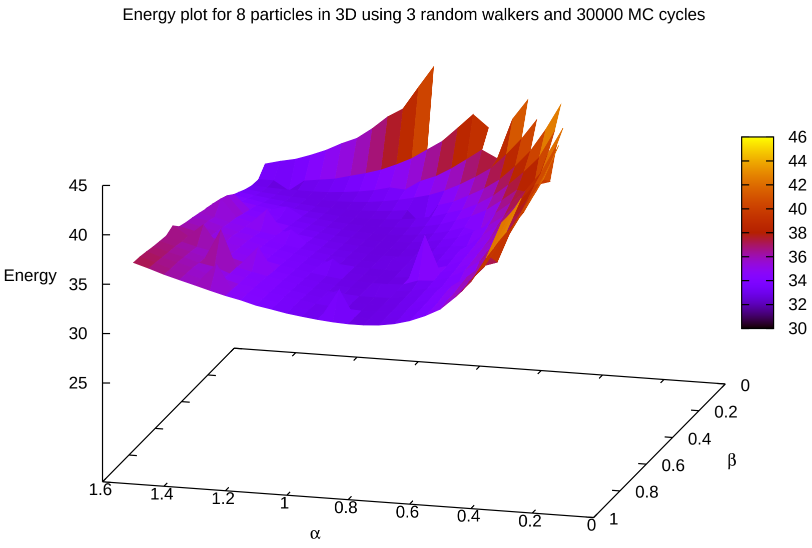

Figure 5.12: The energy seems now to increase on both sides of the α minimum while in two dimensions the energy is almost constant for $\alpha > 1$.

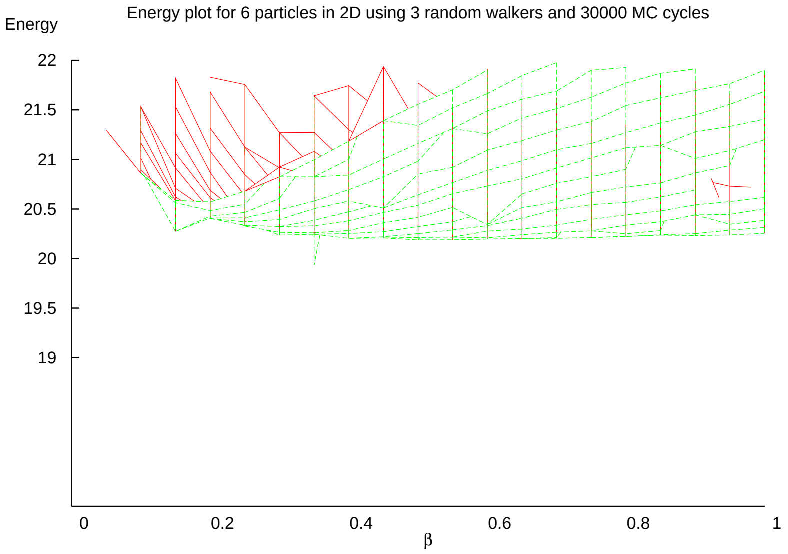

Figure 5.13: The energy is almost constant for $\beta > 0.4$. We use this information to start our parameter search at 0.2 to prevent the algorithm getting stuck in a local minima.

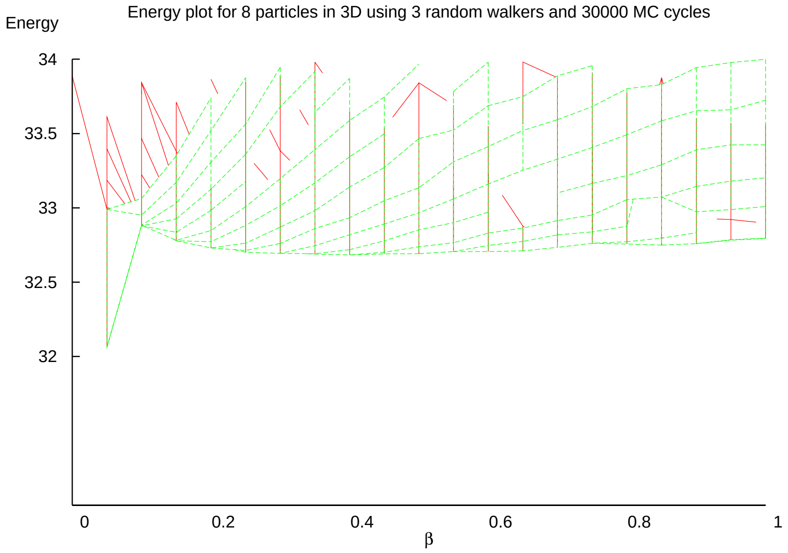

Figure 5.14: The same story here excepts for a rather large statistical spike in the energy. This is due to only using 30000 cycles which is not enough to get a good estimate, though it serves us well as a rough guide in deciding where to start the optimization

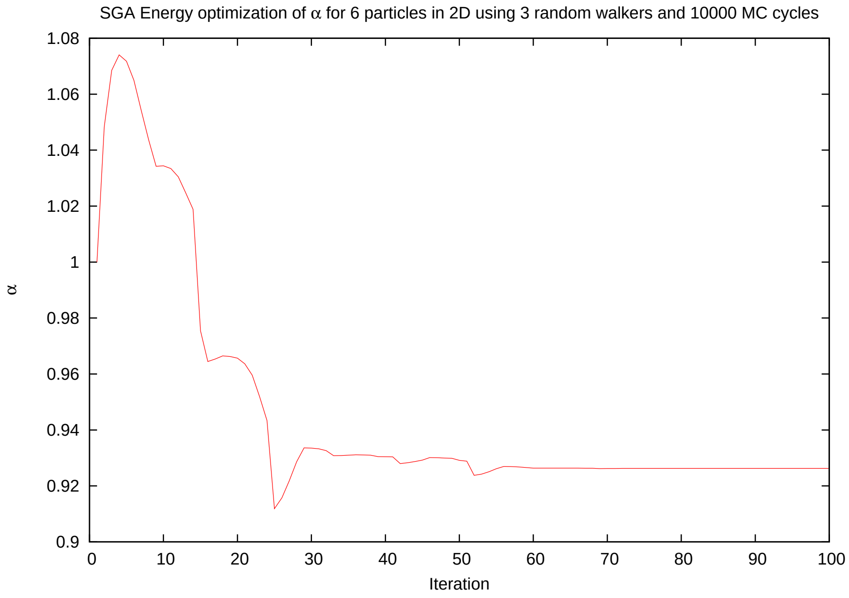

Figure 5.15: The increase in amount of particles increases the amount of zig-zags before converging. However a certain amount of noise is required because it increases the odds of not getting stuck in a local minima.

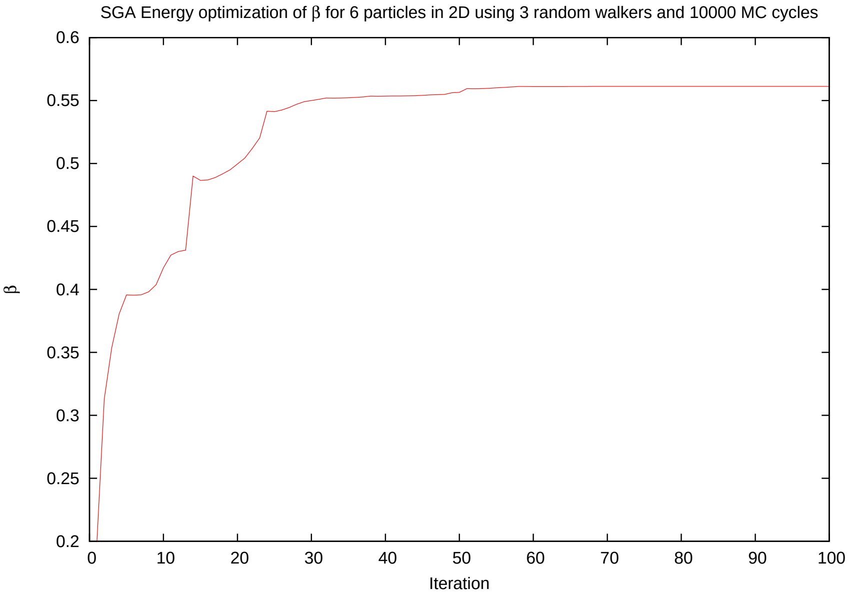

Figure 5.16: Here the optimal β is found rather easy.

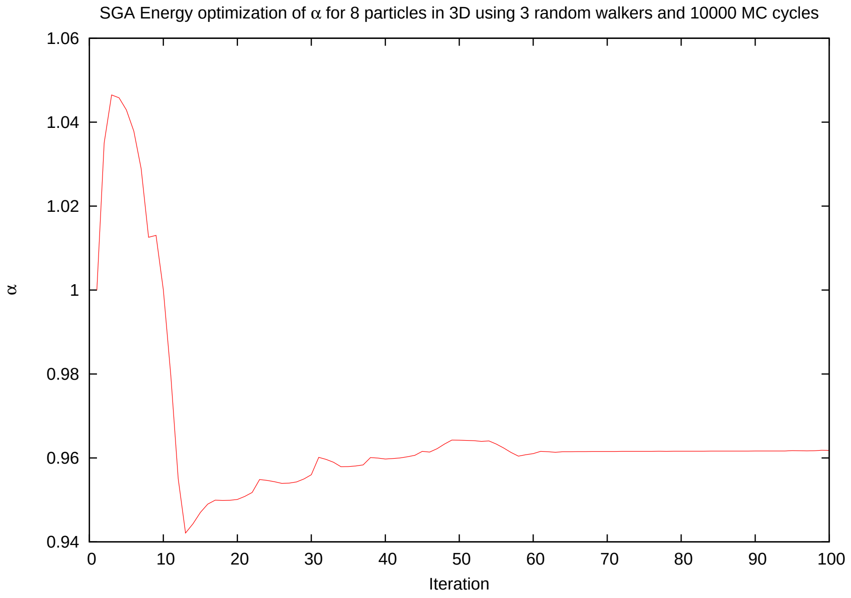

Figure 5.17: We see a couple of large oscillations in the wrong direction before convergence is achieved. This may suggest decreasing the base step length ℓ_0 in the algorithm to increase efficiency.

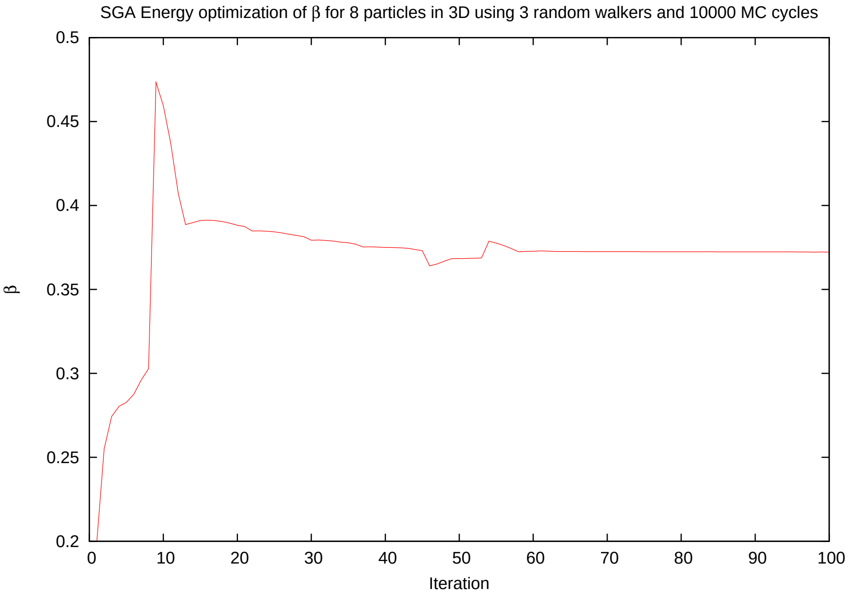

Figure 5.18: The same basic behaviour as for the other cases.

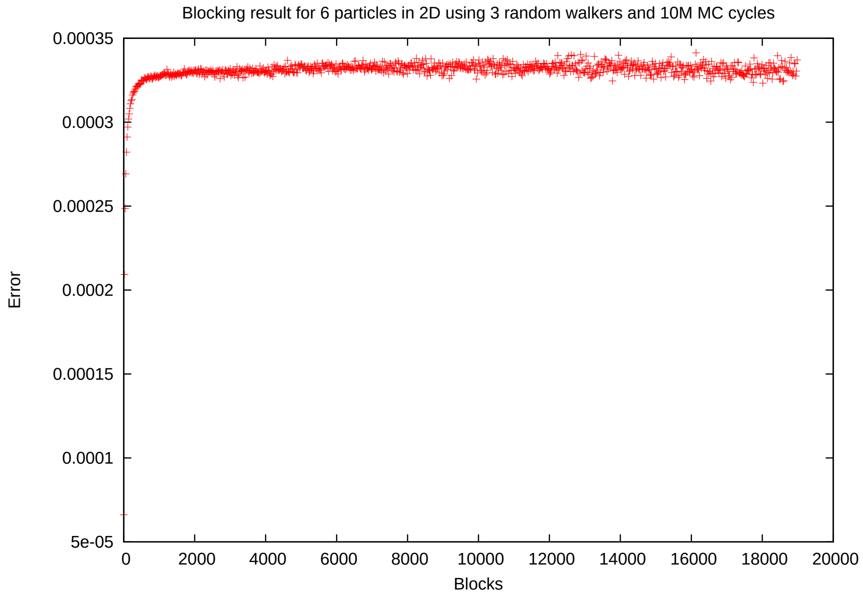

Figure 5.19: We see that the correlations are much larger than in the two particle case. This is to be expected.

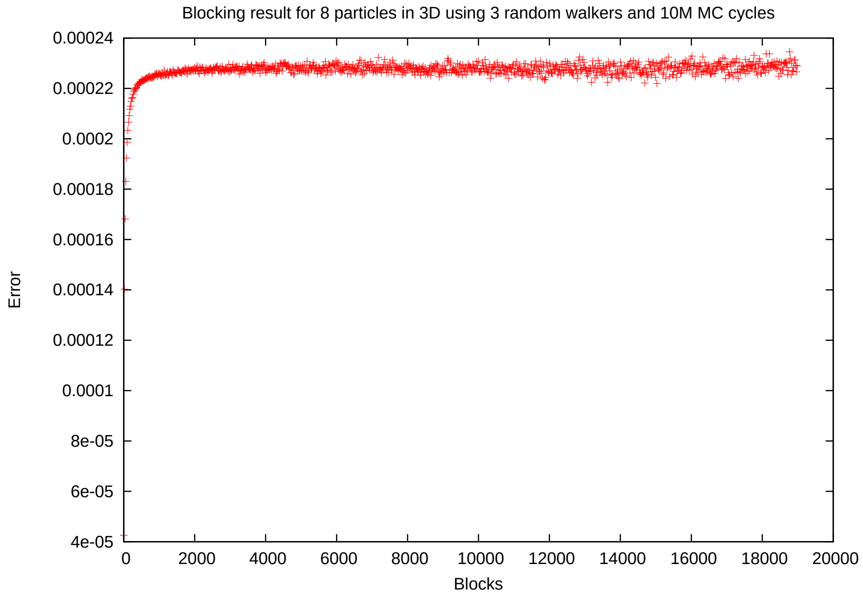

Figure 5.20: The correlations are smaller than in two dimensions even though there are two more particles.

Chapter 6

Conclusion

We have created a flexible and easily extensible program and verified that it works by applying it to the harmonic oscillator system and compared with both analytical results and other software. We have only studied closed shell systems because in that case a single Slater determinant is often enough to yield good results. To study other systems the program must be modified to include linear combinations of all possible Slater determinants but that should be fairly easy. It can also be modified to also handle nuclear systems by modifying the local energy class to include Hamiltonian's used in nuclear systems. The choice of basis functions is also separate from the main program. One can use a Hartree fock basis or any other basis by just adding the one particle functions and the rest of the program does not need to be touched. We have thus made a very general software as intended.

Appendix A

A.1 Statistical analysis

The material is mainly borrowed from [12] and [4] and is devoted to finding the error of the mean of a stochastic sample. This is needed to estimate the error of the energy we compute using the Monte Carlo method. When dealing with random numbers and their probability distributions we assign them a stochastic variable X that is defined in some domain which contains all the possible values of X. The domain can be discrete or continuous and the probability distribution function (PDF) $p(x)$ gives us the probability for one of the values in the domain to occur. A common example is the throw of a dice that has a discrete domain and contains the values $\{1, 2, 3, 4, 5, 6\}$. The corresponding PDF is just $p(x) = 1/6$ for all x in the domain because all values are equally probable. If we had two dice the domain would double and the PDF would no longer be equal for all x.

In the continuous case we can no longer talk about the probability for one value when the domain contains infinitely many. It would just be zero. We can only talk about the the probability for getting a value in some interval within the domain. We define that the probability for getting a value in the interval $[x, x + dx]$ is $p(x)dx$ and the PDF is actually a probability density function. More generally

$$\text{Prob(a} \leq \text{X} \leq \text{b)} = \int_a^b p(x)dx \tag{A.1}$$

and it has to fulfill two properties

$$0 \leq p(x) \leq 1, \tag{A.2}$$

$$\int_{-\infty}^{\infty} p(x) = 1, \tag{A.3}$$

which states that all the probabilities must be positive and add up to unity. The expectation value of X is

$$\mu_X = \langle x \rangle = \int x p(x) dx \tag{A.4}$$

and is also called the mean value. The variance of X, denoted $\mathrm{Var}(X)$ or σ_X^2, is defined as

$$\begin{aligned}
\sigma_X^2 &\equiv \left\langle (x - \langle x \rangle)^2 \right\rangle = \int (x - \langle x \rangle)^2 p(x) dx, && \text{(A.5)}\\
&= \int x^2 - 2x \langle x \rangle + \langle x \rangle^2, && \text{(A.6)}\\
&= \left\langle x^2 \right\rangle - 2 \langle x \rangle \langle x \rangle + \langle x \rangle^2, && \text{(A.7)}\\
&= \left\langle x^2 \right\rangle - \langle x \rangle^2. && \text{(A.8)}
\end{aligned}$$

and tells us how spread out the distribution is. There are also functions of a stochastic variable $f(X)$ which itself is considered a stochastic variable Y with its own PDF $p_Y(y)$. The mean value is

$$\begin{aligned}
\mu_Y = \langle y \rangle &= \int y p_Y(y) dy && \text{(A.9)}\\
&= \int f(x) p_X(x) dx && \text{(A.10)}\\
&\equiv \langle f \rangle_X && \text{(A.11)}
\end{aligned}$$

which means that the variance is simply

$$\sigma_f^2 = \left\langle f^2 \right\rangle_X - \langle f \rangle_X^2 \tag{A.12}$$

A multivariate PDF $P(x_1, \ldots, x_n) = P(\boldsymbol{x})$ is the function for the set of corresponding stochastic variables $\{X_1, \ldots, X_n\} = \boldsymbol{X}$. Its mean value is now a multidimensional integral

$$\begin{aligned}
\mu_{\boldsymbol{X}} = \langle \boldsymbol{x} \rangle &= \int \boldsymbol{x} P(\boldsymbol{x}) d\boldsymbol{x} && \text{(A.13)}\\
&= \int \cdots \int x_1 \cdots x_n P(x_1, \ldots, x_n) dx_1 \cdots dx_n, && \text{(A.14)}
\end{aligned}$$

and likewise for the mean value of a the multivariate function $F(\boldsymbol{X})$. The stochastic variables are uncorrelated if the PDF can be written as the product form

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} p_i(x_i) \tag{A.15}$$

where $p_i(x_i)$ is the PDF corresponding to X_i. In the multivariate case we can define the so called covariance of two stochastic variables as

$$\begin{aligned}
\mathrm{Cov}(X_i, X_j) &\equiv \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle \,, && \text{(A.16)}\\
&= \langle x_i x_j \rangle - \langle x_i \langle x_j \rangle \rangle - \langle x_j \langle x_i \rangle \rangle + \langle \langle x_i \rangle \langle x_j \rangle \rangle \,, && \text{(A.17)}\\
&= \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle \,. && \text{(A.18)}
\end{aligned}$$

If X_i and X_j are uncorrelated, then by using eq. (A.15), we get that $\langle x_i x_j \rangle = \langle x_i \rangle \langle x_j \rangle$ and the covariance is zero. In other words, the covariance is a measure of the correlation between stochastic variables. However, zero covariance does not necessarily imply no correlations. In the special case $X_i = X_j$ then the covariance reduces to the variance. If we put all the covariances in a matrix then it would be symmetric and the diagonal elements equal to the variance. In the special case that a multivariate function $F(\boldsymbol{X})$ is linear

$$F(\boldsymbol{X}) = \sum_i a_i X_i \tag{A.19}$$

then the mean is

$$\langle F \rangle_{\boldsymbol{X}} = \sum_i a_i \langle X_i \rangle \tag{A.20}$$

and the variance can be shown to be

$$\sigma_F^2 = \sum_i a_i^2 \sigma_{X_i}^2 + 2 \sum_i \sum_{j=i+1} a_i a_j \mathrm{Cov}(X_i, X_j). \tag{A.21}$$

We can now move on to consider a stochastic experiment which consists of obtaining a sequence of numbers $\{x_i\}$ that we assume are distributed according to an unknown PDF. We call a certain sequence of numbers for a sample of the PDF and we can of course obtain several different samples by doing more experiments. When we use the word experiment it sounds like something physical like measuring radioactive decay but it also includes numerical experiments. Consider the expectation value of some function f which is given by eq. (A.10). If we know the PDF and we want to compute the mean value numerically, it involves a numerical integration which is always approximated as a discrete sum. The approximation would depend on the discretization method and the number of integration points. The Monte Carlo integration technique could sample the PDF with the Metropolis algorithm which uses Markov chains. A Markov chain is a stochastic random walk and the integral must be considered a stochastic experiment where the sequence of numbers corresponds to the value of f at each point in the random walk. In this thesis the function of interest is the local energy and we need to find how good the MC approximation is. Because the samples are finite, the mean value and variance of the corresponding PDF must be approximated. First we define the mean of a sample of size n to be

$$\bar{x} \equiv \frac{1}{n}\sum_{i=1}^{n} x_i. \tag{A.22}$$

The sample variance is

$$\sigma_x \equiv \frac{1}{n}\sum_{i=1}^{n} (x_i - \bar{x}), \tag{A.23}$$

while the sample covariance is

$$\mathrm{Cov}(x) \equiv \frac{1}{n}\sum_{i=1}^{n} (x_i - \bar{x})(x_j - \bar{x}). \tag{A.24}$$

The sample covariance measures the sequential correlation between succeeding measurements in a sample. As the sample size n goes to infinity then the sample mean approaches the true mean. But the big question here is how good of an approximation is the sample mean of the true mean. A good measure of the error ϵ in the approximation is the variance of the sample mean $\sigma_{\bar{x}}$, which must not be confused with the sample variance σ_x. The sample mean is also a stochastic variable with its own mean value and variance. The straightforward way of computing $\sigma_{\bar{x}}$ is to do many experiments and use eq. (A.10). If we only have one experiment, then we can divide the sample into m subsamples and treat each subsample as an experiment. This is called blocking and will be discussed in greater detail a bit later. There is another approach which also gives analytic information about the correlation effects.

We can associate each measurement x_i in the sample with its own stochastic variable X_i. The stochastic variable for the sample mean is then

$$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{A.25}$$

The PDF of $\overline{X}_n$ is

$$p_{\overline{X}_n}(x) = \int p_X(x_1)\cdots\int p_X(x_n)\delta\left(x - \frac{1}{n}\sum_{i=1}^{n} x_i\right) dx_1\cdots dx_n \tag{A.26}$$

The central limit theorem (referanse?) gives

$$\lim_{n\to\infty} p_{\overline{X}_n}(x) = \left(\frac{n}{2\pi\sigma_X^2}\right)^{\frac{1}{2}} e^{\frac{n(x-\mu_X)^2}{2\sigma_X^2}} \tag{A.27}$$

The variance can be shown to be

$$\sigma_{\overline{X}_n}^2 \approx \frac{1}{n}\mathrm{Cov}(x). \tag{A.28}$$

If the sample is uncorrelated the variance is

$$\sigma_{\overline{X}_n}^2 \approx \frac{1}{n}\sigma_x. \tag{A.29}$$

A good measure of the correlation is the autocorrelation time

$$\tau = 1 + 2\sum_{d=1}^{n-1}\kappa_d \tag{A.30}$$

where κ_d is the autocorrelation function given by

$$\kappa_d = \frac{\sigma_x}{n}\sum_{k=1}^{n-d}(x_k - \bar{x})(x_{k+d} - \bar{x}) \tag{A.31}$$

This is a costly function to evaluate and quite often it decreases exponentially with time. We could exploit this fact and to find a d_0 so that $k_d \approx 0$ for $d > d_0$. This would reduce the computation time a considerable amount but empirical studies have shown that k_d does not die out completely and tends to oscillate around zero.

We now shortly explain the blocking method which is an easy an efficient method. It consits of dividing our dataset into b_s blocks and compute the variance of the sample mean. By increasing the blocksize the variance should increase exponentially until it reaches a plateau where the blocks are no longer correlated. This blocksize and variance will be our estimate for the correlation lentgh d_0 for the true correlated variance.

Bibliography

[1] M. Taut, J. Phys A **27**, 1045 (1994).

[2] P. Hemmer, *Kvantemekanikk 5. utgave* (Tapir Akademisk Forlag, 2005).

[3] S. A. T. B. P. F. William H. Press, William T. Vetterling, *Numerical Recipies in C++ 2. Edition* (Cambridge University Press, 2002).

[4] B. L. Hammond, W. A. Lester, and P. Reynolds, *Monte Carlo Methods in ab Initio Quantum Chemistry* (World Scientific, 1994).

[5] S. R. A. Harju, B.Barbiellini, Phys Rev Letters **79**, 1173 (1997).

[6] A. R. X. Lin, H. Zhang, J. Chem. Phys. **112**, 2650 (2000).

[7] C. Umrigar, Phys Rev Letters **94**, 150201 (2005).

[8] Moskowitz and Kalos, Int. J. Quantum Chem. **20**, 1107 (1981).

[9] R. Shankar, *Principles of Quantum Mechanics* (Springer, 1994).

[10] H. H. Petersen, J. Chem. Phys. **91**, 461 (1989).

[11] S. Kvaal, *Analysis of many-body methods for quantum dots*, PhD thesis, University of Oslo, 2009.

[12] M. Røstad, Efficient quantum monte carlo calculations of many-body body systems, Master's thesis, University of Oslo, 2004.