Quantum Monte-Carlo Studies of Generalized Many-body Systems

by

Jørgen Høgberget

THESIS
for the degree of
MASTER OF SCIENCE

(Master in Computational Physics)

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

June 2013

Preface

blah blah

Contents

1

Introduction

blah blah

Part I

Theory

2

Quantum Monte-Carlo

Quantum Monte-Carlo (QMC) is a method of solving Schrödinger's equation using statistical *Markov Chain* (random walk) simulations, i.e. Monte-Carlo simulations. The statistical nature of Quantum Mechanics makes Monte-Carlo methods the perfect tool not only for accurately estimating observables, but also for extracting interesting quantities such as densities, i.e. probability distributions.

There are multiple ways to deduce the virtual[1] dynamics of QMC, some of which are more mathematically complex than others. In this chapter the focus will be on modeling the Schrödinger equation as a diffusion problem in complex (Wick rotated) time. Other more condensed mathematical approaches does not need the Schrödinger equation at all, however, for the purpose of transparency, this approach will be mentioned only briefly in Section 2.2.3.

In this chapter *Dirac Notation* will be used. See Appendix **??** for an introduction. The equations will be in atomic units, i.e. $\hbar = m_e = e = 1$.

2.1 Modeling Diffusion

Like any phenomena involving a probability distribution, Quantum Mechanics may be modeled by a diffusion process. In Quantum Mechanics, the distribution is given by $|\Phi(\mathbf{r}, t)|^2$, the Wave function squared. The diffusing elements of interest are the particles making up our system.

The idea is to have an ensemble of *Random Walkers* in which each walker is represented by a position in space and time. Once the walkers reach equilibrium, averaging values over the paths of the ensemble will yield average values corresponding to the probability distribution governing the movement of individual walkers. In other words: Counting every walker's contribution within a small volume $\mathrm{d}\mathbf{r}$ will correspond to $|\Phi(\mathbf{r}, t)|^2 \mathrm{d}\mathbf{r}$ in the limit of infinite walkers.

Such random movement of walkers are referred to as a *Brownian motion*, named after the British Botanist R. Brown, originating from his experiments on plant pollen dispersed in water. Markov chains are a subtype of Brownian motion, where a walkers next move is independent of previous moves. This is the stochastic process in which Quantum Monte Carlo is described.

The purpose of this section is to motivate the use of diffusion theory in Quantum Mechanics, and to

[1] As will be shown, the time parameter in QMC does not correspond to physical time, but rather an imaginary axis at a fixed point in time. Whether nature operates on a complex time plane or not is not testable in a laboratory, and the origin of the probabilistic nature of Quantum Mechanics will thus remain a philosophical problem.

derive the sampling rules needed in order to model Quantum Mechanical distributions by diffusion of random walkers correctly. I will be using natural units, that is $\hbar$, m_e, etc. are all set to unity, in order to simplify the expressions.

2.1.1 Stating the Schrödinger Equation as a Diffusion Problem

Consider the time-dependent Schrödinger equation for an arbitrary many-body wave function[2] $\Phi(\mathbf{r}, t)$ with an arbitrary energy shift E'

$$-\frac{\partial\Phi(\mathbf{r}, t)}{i\partial t} = (\widehat{\mathbf{H}} - E')\Phi(\mathbf{r}, t). \tag{2.1}$$

Given that the Hamiltonian is time-independent, the formal solution is found by separation of variables in $\Phi(\mathbf{r}, t)$ (see ref. [1] for more details)

$$\widehat{\mathbf{H}}\Phi(\mathbf{r}, t_0) = E\Phi(\mathbf{r}, t_0) \tag{2.2}$$

$$\Phi(\mathbf{r}, t) = \exp\left(-i(\widehat{\mathbf{H}} - E')(t - t_0)\right).\Phi(\mathbf{r}, t_0) \tag{2.3}$$

From Eq. (2.3) it is apparent that the time evolution operator is on the form

$$\widehat{\mathbf{U}}(t, t_0) = \exp\left(-i(\widehat{\mathbf{H}} - E')(t - t_0)\right). \tag{2.4}$$

The time-independent equation are solved for the ground state energy through methods such as *Full Configuration Interaction* or similar methods based on diagonalizing the Hamiltonian. The time-dependent equation is used by methods such as *Time-Dependent Multi-Configuration Hartree-Fock* in order to obtain solutions for the time-development of Quantum states. However, none of these originate from or resemble diffusion equations.

The original Schrödinger equation, however, does resemble a diffusion equation in complex time[3]. It can not be treated as a true diffusion equation, since the time evolved quantity, the wave function, is not a probability distribution unless it is squared. However, the equation involves a time derivative and a Laplacian, strongly indicating some sort of connection to a diffusion process.

Applying the complex time Wick rotation to the time evolution operator in Eq. (2.4) and choosing the energy shift E' equal to the true ground state energy E_0 of $\widehat{\mathbf{H}}$ yields the *projection operation* whose name will soon be apparent

$$t \to it$$

$$\widehat{\mathbf{U}}(t, 0) \to \exp\left(-(\widehat{\mathbf{H}} - E_0)\tau\right) \equiv \widehat{\mathbf{P}}(\tau).$$

Consider an arbitrary state $\Psi_T(\mathbf{r})$. Applying the new operator yields a new state $\Phi(\mathbf{r}, \tau)$ and expanding $\Psi_T(\mathbf{r})$ in the eigenstates $\Psi_i(\mathbf{r})$ of $\widehat{\mathbf{H}}$ yields

[2] See Section 2.6.1 for details regarding many-body wave functions.

[3] The physical time diffusion equation evolves the squared wave function, and can be deduced from the quantum continuity equation combined with Fick's laws of diffusion REFERER TIL TING :(

$$
\begin{aligned}
\Phi(\mathbf{r},\tau) &= \langle \mathbf{r}| \widehat{\mathbf{P}}(\tau) |\Psi_T\rangle \qquad (2.5)\\
&= \langle \mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_0)\tau\right) |\Psi_T\rangle \\
&= \sum_i C_i \langle r| \exp\left(-(\widehat{\mathbf{H}} - E_0)\tau\right) |\Psi_i\rangle \\
&= \sum_i C_i \Psi_i(\mathbf{r}) \exp\left(-(E_i - E_0)\tau\right) \\
&= C_0\Psi_0(x) + \sum_{i=1}^{\infty} C_i \Psi_k(x) e^{-\delta E_i \tau}, \qquad (2.6)
\end{aligned}
$$

where $C_i = \langle \Psi_i|\Psi_T\rangle$ and $\delta E_i = E_i - E_0 \geq 0$. In the limit where τ goes to infinity, the ground state is the sole survivor of the expression, hence the name projection operator.

$$
\begin{aligned}
\lim_{\tau\to\infty} \Phi(\mathbf{r},\tau) &= \lim_{\tau\to\infty} \langle \mathbf{r}| \widehat{\mathbf{P}}(\tau) |\Psi_T\rangle \\
&= C_0\Psi_0(x). \qquad (2.7)
\end{aligned}
$$

The projection operator can in other words be used to transform an arbitrary wave function $\Psi_T(\mathbf{r})$[4] into the true ground state, given that the overlap C_0 is non-zero.

In order to model the projection with Markov chains, the process needs to be split into subprocesses which in turn can be described as transitions in the Markov chain. Introducing a time-step $\delta\tau$, the projection operator can be rewritten as

$$
\widehat{\mathbf{P}}(\tau) = \prod_{k=1}^{n} \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right), \qquad (2.8)
$$

where $n = \tau/\delta\tau$. An important property to notice is that

$$
\widehat{\mathbf{P}}(\tau + \delta\tau) = \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) \widehat{\mathbf{P}}(\tau). \qquad (2.9)
$$

Using this relation in combination with Eq. (2.5), the effect of the projection operator in a single time-step is revealed

$$
\begin{aligned}
\Phi(\mathbf{r},\tau+\delta\tau) &= \langle \mathbf{r}| \widehat{\mathbf{P}}(\tau+\delta\tau) |\Psi_T\rangle \\
&= \langle \mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) \widehat{\mathbf{P}}(\tau) |\Psi_T\rangle \\
&= \langle \mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) |\Phi(\tau)\rangle \\
&= \int_{\mathbf{r}'}\int_{\mathbf{r}''} \langle \mathbf{r}|\mathbf{r}''\rangle \langle \mathbf{r}''| \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) |\mathbf{r}'\rangle \langle \mathbf{r}'|\Phi(\tau)\rangle \, \mathrm{d}\mathbf{r}'\mathbf{r}'' \\
&= \int_{\mathbf{r}'} \langle \mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) |\mathbf{r}'\rangle \, \Phi(\mathbf{r}',\tau)\mathrm{d}\mathbf{r}' \qquad (2.10)
\end{aligned}
$$

[4] This is called the *trial wave function* and will be covered in detail in Section 2.6.

For practical purposes, E_0 needs to be substituted with an approximation E_T to the ground state energy to avoid self consistency. From Eq (2.6) it is apparent that the projection will still converge as long as $E_T < E_1$, i.e. the energy of the first excitation.

$$\Phi(\mathbf{r}, \tau + \delta\tau) = \int_{\mathbf{r}'} \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_T)\delta\tau\right) | \mathbf{r}' \rangle \, \Phi(\mathbf{r}', \tau) \mathrm{d}\mathbf{r}' \tag{2.11}$$

$$\equiv \int_{\mathbf{r}'} G(\mathbf{r}, \mathbf{r}'; \delta\tau) \Phi(\mathbf{r}', \tau) \mathrm{d}\mathbf{r}' \tag{2.12}$$

The equations above are well suited for Markov Chain models, as an ensemble of walkers can be iterated by transitioning between configurations $|\mathbf{r}\rangle$ and $|\mathbf{r}'\rangle$ with probabilities given by the *Green's function*, $G(\mathbf{r}, \mathbf{r}'; \delta\tau)$.

The effect of the Green's function from Eq. (2.12) on individual walkers is not trivial. In order to relate the Green's function to well-known processes, the exponential is split into two parts, one containing only the kinetic energy operator $\widehat{\mathbf{T}} = -\frac{1}{2}\nabla^2$, and the second containing the potential $\widehat{\mathbf{V}}$ and the energy shift. This is known as the *short time approximation*[2]

$$G(\mathbf{r}, \mathbf{r}'; \delta\tau) = \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) | \mathbf{r}' \rangle \tag{2.13}$$

$$= \langle \mathbf{r} | \, e^{-\widehat{\mathbf{T}}\delta\tau} e^{-(\widehat{\mathbf{V}} - E_T)\delta\tau} \, | \mathbf{r}' \rangle + \frac{1}{2}[\widehat{\mathbf{V}}, \widehat{\mathbf{T}}]\delta\tau^2 + \mathcal{O}(\delta\tau^3). \tag{2.14}$$

The first exponential describes a transition of walkers governed by the Laplacian, which is a diffusion process. The second exponential is linear in position space and is thus a weighing function, distributing the correct weights to certain walkers. In other words:

$$G_{\text{Diff}} = e^{\frac{1}{2}\nabla^2\delta\tau} \tag{2.15}$$

$$G_{\text{B}} = e^{-(\widehat{\mathbf{V}} - E_T)\delta\tau}, \tag{2.16}$$

where B denotes *branching*. Modeling weights by branching will be covered in detail in Section 2.5.

The flow of QMC is then to use these Green's functions to propagate the ensemble of walkers into the next time step. The final distributions of walkers will correspond to that of the direct solution of the Schrödinger Equation, given that the time step is sufficiently small, and the number of cycles n are sufficiently large. These constraints will be covered in more detail later.

Incorporating only the effect of Eq.(2.15) results in a method called *Variational Monte Carlo*. Including the branching term as well results in *Diffusion Monte Carlo*. These methods will be discussed in sections 2.8 and 2.9. In either of these methods, diffusion is a key process.

2.2 Solving the Diffusion Problem

The diffusion problem introduced in the previous section uses a symmetric kinetic energy operator implying an *isotropic diffusion*, however, a more efficient kinetic energy operator can be introduced without violating the original equations, resulting in an *anisotropic diffusion* governed by the *Fokker-Planck equation*. These models will be the topic of this section.

For details regarding the transition from isotropic to anisotropic diffusion, see Section 2.2.3.

2.2.1 Isotropic Diffusion

Isotropic diffusion is a process in which diffusing particles sees all possible directions as an equally probable path. Eq. (2.17) is an example of this. This is the simplest form of a diffusion equation, that is, the case with a linear *diffusion constant*, D, and no drift terms.

$$\frac{\partial P(\mathbf{r},t)}{\partial t} = D\nabla^2 P(\mathbf{r},t). \tag{2.17}$$

From Eq. (2.15) it is clear that the value of the diffusion constant is $D = \frac{1}{2}$, originating from the term scaling the Laplacian in the Schrödinger Equation. A crucial point is that closed form expressions for the Green's function exists. This closed form expression in the isotropic case is a Gaussian distribution with variance $2D\delta t$ [2]

$$\boxed{G_{\text{Diff}}^{\text{ISO}}(i \to j) \propto e^{-|\mathbf{r}_i - \mathbf{r}_j|^2/4D\delta\tau}. \tag{2.18}}$$

These equations describe the diffusion process theoretically, however, in order to achieve specific sampling rules for the walkers, a connection between the time-dependence of the total distribution and the time-dependence of an individual walker's components in configuration space is needed. This connection is given in terms of a stochastic differential equation called *The Langevin Equation.*

The Langevin Equation for Isotropic Diffusion

The Langevin Equation is a stochastic differential equation used in physics to relate the time dependence of a distribution to the time-dependence of the degrees of freedom in a system. For isotropic diffusion, solving the Langevin equation using a Forward Euler approximation for the time derivative results in the following relation:

$$x_{i+1} = x_i + \xi, \qquad \begin{aligned} \text{Var}(\xi) &= 2D\delta t, \\ \langle \xi \rangle &= x_i, \end{aligned} \tag{2.19}$$

where ξ is a normal distributed number whose variance match that of the Green's function in Eq. (2.18). This relation is in agreement with the isotropy of Eq. (2.17) in the sense that the displacement is symmetric around the current position.

2.2.2 Anisotropic Diffusion: Fokker-Planck

Anisotropic diffusion, in contrast to isotropic diffusion, does not see all directions as equally probable. An example of this is diffusion according to the *Fokker-Planck Equation*, that is, diffusion with a drift term, $\mathbf{F}(\mathbf{r},t)$, responsible for pushing the walkers in the direction of configurations with higher probabilities, and thus closer to an equilibrium state.

$$\frac{\partial P(\mathbf{r},t)}{\partial t} = D\nabla \cdot \Big[\Big(\nabla - \mathbf{F}(\mathbf{r},t)\Big)P(\mathbf{r},t)\Big]. \tag{2.20}$$

As will be derived in details in Section 2.2.3, using the Fokker-Planck equation does not violate the original Schrödinger equation, but changes the representation of the ensemble of walkers to a mixed density. This means that QMC can be run with Fokker-Planck diffusion, leading to a more optimized way of sampling due to the drift term.

A necessity to the calculation is convergence to a stationary state. Using this criteria, the expression for the drift term can be found. A stationary state is obtained when the left hand side of Eq. (2.20) is zero:

$$\nabla^2 P(\mathbf{r},t) = P(\mathbf{r},t)\nabla\cdot\mathbf{F}(\mathbf{r},t) + \mathbf{F}(\mathbf{r},t)\cdot\nabla P(\mathbf{r},t).$$

In order to get cancellation in the remaining terms, the Laplacian term on the right hand side must cancel out the terms on the left. This implies that the drift term needs to be on the form $\mathbf{F}(\mathbf{r},t) = g(\mathbf{r},t)\nabla P(\mathbf{r},t)$. Inserting this yields

$$\nabla^2 P(\mathbf{r},t) = P(\mathbf{r},t)\frac{\partial g(\mathbf{r},t)}{\partial P(\mathbf{r},t)}\Big|\nabla P(\mathbf{r},t)\Big|^2 + P(\mathbf{r},t)g(\mathbf{r},t)\nabla^2 P(\mathbf{r},t) + g(\mathbf{r},t)\Big|\nabla P(\mathbf{r},t)\Big|^2.$$

Looking at the factors in front of the Laplacian suggests using $g(\mathbf{r},t) = 1/P(\mathbf{r},t)$. A quick check reveals that this also cancels the gradient terms. The resulting expression for the drift term become

$$\boxed{\begin{aligned}\mathbf{F}(\mathbf{r},t) &= \frac{1}{P(\mathbf{r},t)}\nabla P(\mathbf{r},t)\\ &= \frac{2}{|\psi(\mathbf{r},t)|}\nabla|\psi(\mathbf{r},t)|\end{aligned}} \tag{2.21}$$

In QMC, the drift term is commonly referred to as the *quantum force*, due to the fact that it is responsible for pushing the walkers into regions of higher probabilities, analogous to a force in Newtonian mechanics.

Another strength of the Fokker-Planck equation is that even though the equation itself is more complicated, it's Green's function still has a closed form solution. This means that we can evaluate it efficiently. If this was not the case, the practical value would be reduced dramatically. The reason for this will become clear in Section 2.4. As expected, it is no longer symmetric

$$\boxed{G_{\text{Diff}}^{\text{FP}}(i \to j) \propto e^{-(x_i - x_j - D\delta\tau F(x_i))^2/4D\delta\tau}.} \tag{2.22}$$

The Langevin Equation for the Fokker-Planck Equation

The Langevin equation in the case of a Fokker-Planck Equation has the following form

$$\frac{\partial x_i}{\partial t} = DF(\mathbf{r})_i + \eta, \tag{2.23}$$

where η is a so-called *noise term* from stochastic processes. Solving this equation using the same method as for the isotropic case yields the following sampling rules

$$x_{i+1} = x_i + \xi + DF(\mathbf{r})_i\delta t, \tag{2.24}$$

where ξ is the same as for the isotropic case. Observe that when the drift term goes to zero, the Fokker-Planck - and isotropic solutions are equal, just as required. For more details regarding the Fokker-Planck Equation and Langevin equations, see ref. [3], [4] and [5].

2.2.3 The connection between anisotropic- and isotropic diffusion models

To this point, it might seem far-fetched that switching the diffusion model to a Fokker-Planck diffusion does not violate the original equation, i.e. the complex time Schrödinger equation (the projection operator). Introducing the distribution function $f(\mathbf{r},t) = \Phi(\mathbf{r},t)\Psi_T(\mathbf{r})$, restating the imaginary time Schrödinger equation in terms of $f(\mathbf{r},t)$ yields

$$\begin{aligned}
-\frac{\partial}{\partial t} f(\mathbf{r},t) &= \Psi_T(\mathbf{r})\Big[-\frac{\partial}{\partial t}\Phi(\mathbf{r},t)\Big] &=& \Psi_T(\mathbf{r})\left(\widehat{\mathbf{H}} - E_T\right)\Phi(\mathbf{r},t) \\
&&=& \Psi_T(\mathbf{r})\left(\widehat{\mathbf{H}} - E_T\right)\Psi_T(\mathbf{r})^{-1} f(\mathbf{r},t) \qquad (2.25)\\
&&=& -\frac{1}{2}\Psi_T(\mathbf{r})\nabla^2\left(\Psi_T(\mathbf{r})^{-1} f(\mathbf{r},t)\right) + \widehat{\mathbf{V}} f(\mathbf{r},t) - E_T f(\mathbf{r},t).
\end{aligned}$$

Expanding the Laplacian term further reveals

$$\begin{aligned}
K(\mathbf{r},t) &\equiv -\frac{1}{2}\Psi_T(\mathbf{r})\nabla^2\left(\Psi_T(\mathbf{r})^{-1} f(\mathbf{r},t)\right) \\
&= -\frac{1}{2}\Psi_T(\mathbf{r})\nabla\cdot\left(\nabla\left[\Psi_T(\mathbf{r})^{-1} f(\mathbf{r},t)\right]\right) \qquad (2.26)\\
\nabla\left[\Psi_T(\mathbf{r})^{-1} f(\mathbf{r},t)\right] &= -\Psi_T(\mathbf{r})^{-2}\nabla\Psi_T(\mathbf{r}) f(\mathbf{r},t) + \Psi_T(\mathbf{r})^{-1}\nabla f(\mathbf{r},t), \qquad (2.27)
\end{aligned}$$

combining these two equations and using the product rule numerous time yields

$$\begin{aligned}
K(\mathbf{r},t) &= -\frac{1}{2}\Psi_T(\mathbf{r})\Big[\big(2\Psi_T(\mathbf{r})^{-3}\left|\nabla\Psi_T(\mathbf{r})\right|^2 f(\mathbf{r},t) \\
&\quad -\Psi_T(\mathbf{r})^{-2}\nabla^2\Psi_T(\mathbf{r}) f(\mathbf{r},t) \\
&\quad -\Psi_T(\mathbf{r})^{-2}\nabla\Psi_T(\mathbf{r})\cdot\nabla f(\mathbf{r},t)\big) \\
&\quad +\Psi_T(\mathbf{r})^{-1}\nabla^2 f(\mathbf{r},t) \\
&\quad -\Psi_T(\mathbf{r})^{-2}\nabla\Psi_T(\mathbf{r})\cdot\nabla f(\mathbf{r},t)\Big] \\
&= -\left|\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right|^2 f(\mathbf{r},t) \\
&\quad +\frac{1}{2}\Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r}) f(\mathbf{r},t) \\
&\quad +\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\cdot\nabla f(\mathbf{r},t) \\
&\quad -\frac{1}{2}\nabla^2 f(\mathbf{r},t).
\end{aligned}$$

Introducing the following identity helps clean up the messy calculations:

$$\begin{aligned}
\nabla\cdot\left(\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right) &= -\Psi_T(\mathbf{r})^{-2}\left|\nabla\Psi_T(\mathbf{r})\right|^2 + \Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r}) \\
\left|\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right|^2 &= -\nabla\cdot\left(\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right) + \Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r}),
\end{aligned}$$

which inserted into the expression for $K(\mathbf{r},t)$ reveals

$$\begin{aligned} K(\mathbf{r},t) &= \nabla\cdot\left(\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right)f(\mathbf{r},t) \\ &\quad +\left(\frac{1}{2}-1\right)\Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r})f(\mathbf{r},t) \\ &\quad +\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\cdot\nabla f(\mathbf{r},t) \\ &\quad -\frac{1}{2}\nabla^2 f(\mathbf{r},t). \end{aligned}$$

Inserting the expression for the quantum force $\vec{F}(\mathbf{r}) = 2\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})$ and the local kinetic energy $K_L(\mathbf{r}) = -\frac{1}{2}\Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r})$ simplifies the expression dramatically

$$\begin{aligned} K(\mathbf{r},t) &= -\frac{1}{2}\nabla^2 f(\mathbf{r},t)+\frac{1}{2}\underbrace{\left[\vec{F}(\mathbf{r})\cdot\nabla f(\mathbf{r},t)+f(\mathbf{r},t)\nabla\cdot\vec{F}(\mathbf{r})\right]}_{\nabla\cdot\left[\vec{F}f(\mathbf{r},t)\right]}+K_L(\mathbf{r})f(\mathbf{r},t) \\ &= \frac{1}{2}\nabla\cdot\left[\left(\nabla-\vec{F}(\mathbf{r})\right)f(\mathbf{r},t)\right]+K_L(\mathbf{r})f(\mathbf{r},t). \end{aligned}$$

Inserting everything back into Eq. (2.25) yields

$$\begin{aligned} -\frac{\partial}{\partial t}f(\mathbf{r},t) &= -\frac{1}{2}\nabla\cdot\left[\left(\nabla-\vec{F}(\mathbf{r})\right)f(\mathbf{r},t)\right]+K_L(\mathbf{r})f(\mathbf{r},t)+\widehat{\mathbf{V}}f(\mathbf{r},t)-E_Tf(\mathbf{r},t) \\ \frac{\partial}{\partial t}f(\mathbf{r},t) &= \frac{1}{2}\nabla\cdot\left[\left(\nabla-\vec{F}(\mathbf{r})\right)f(\mathbf{r},t)\right]-\left(E_L(\mathbf{r})-E_T\right)f(\mathbf{r},t), \qquad (2.28) \end{aligned}$$

which is a Fokker-Planck diffusion equation (Eq. (2.20)) with constant shift representing the branching Green's function in the case Fokker-Planck QMC (see Eq. (2.38).

Just as in traditional importance sampled Monte-Carlo integrals, optimized sampling is obtained by switching distributions into one which exploits known information about the problem at hand. In case of standard Monte-Carlo integration, the sampling distribution is substituted with one which are similar to the original integrand, resulting in a smoother sampled function, where as in Quantum Monte-Carlo, a distribution is constructed with the sole purpose of imitating the exact ground state in order to suggest moves more efficiently. It is therefore reasonable to call the use of Fokker-Planck diffusion *Importance sampled Quantum Monte-Carlo.*

The QMC energy estimated using the new distribution $f(\mathbf{r},t)$ will still yield the exact energy in the limit of convergence. Consider the following relation:

$$
\begin{aligned}
E_{\mathrm{DMC}} &= \frac{1}{N}\int f(\mathbf{r},\tau)\frac{1}{\Psi_T(\mathbf{r})}\widehat{\mathbf{H}}\Psi_T(\mathbf{r})\mathrm{d}\mathbf{r} \\
&= \frac{1}{N}\int \Phi(\mathbf{r},\tau)\widehat{\mathbf{H}}\Psi_T(\mathbf{r})\mathrm{d}\mathbf{r} \\
&= \frac{1}{N}\langle\Phi(\tau)|\,\widehat{\mathbf{H}}\,|\Psi_T\rangle \\
N &= \int f(\mathbf{r},\tau)\mathrm{d}\mathbf{r} \\
&= \int \Phi(\mathbf{r},\tau)\Psi_T(\mathbf{r})\mathrm{d}\mathbf{r} \\
&= \langle\Phi(\tau)|\Psi_T\rangle \\
E_{\mathrm{DMC}} &= \frac{\langle\Phi(\tau)|\,\widehat{\mathbf{H}}\,|\Psi_T\rangle}{\langle\Phi(\tau)|\Psi_T\rangle}.
\end{aligned}
$$

Assuming the DMC has converged to the exact ground state, i.e. $|\Phi(\tau)\rangle \rightarrow |\Phi_0\rangle$, letting the Hamiltonian work to the left yields

$$
\begin{aligned}
E_{\mathrm{DMC}} &= E_0\frac{\langle\Phi_0|\Psi_T\rangle}{\langle\Phi_0|\Psi_T\rangle} \\
&= E_0.
\end{aligned}
$$

2.3 Diffusive Equilibrium Constraints

Upon convergence of a Markov process, the ensemble of walkers on average span the systems most likely state. This is exactly the behavior of a real system of diffusing particles described by statistical mechanics: It will *thermalize*, that is, reach equilibrium.

Once thermalization is reached, expectation values may be sampled. However, simply spawning a Markov process and waiting for thermalization is an inefficient and unpractical scenario. This may take forever, and it may not; either way its not optimal. Introducing rules of acceptance and rejection on top of the suggested transitions given by the Langevin equation (Eq. (2.19) and Eq. (2.24)) will result in an optimized sampling. Special care must be taken not to break necessary properties of the Markov process. If any of the conditions discussed in this section break, there is no guarantee that the system will thermalize properly.

2.3.1 Detailed Balance

For Markov processes, detailed balance is achieved by demanding a *Reversible* Markov process. This boils down to a statistical requirement stating that

$$P_iW(i \rightarrow j) = P_jW(j \rightarrow i), \tag{2.29}$$

where P_i is the probability density in configuration i, and $W(i \rightarrow j)$ is the transition probability between states i and j.

2.3.2 Ergodicity

Another requirement is that the sampling must be ergodic, that is, the random walkers needs to be able to reach any configuration in the space spanned by the distribution function. It is tempting to define a brute force acceptance rule where only steps resulting in a higher overall probability is accepted, however, this limits the path of the walker, and will thus break the requirement of ergodicity.

2.4 The Metropolis Algorithm

The Metropolis Algorithm is a simple set of acceptance/rejection rules used in order to make the thermalization more effective. For a given probability distribution function P, the Metropolis algorithm will force sampled points to follow this distribution.

Starting from the criteria of detailed balance (see Eq. (2.29) and further introducing a model for the transition probability $W(i \to j)$ as consisting of two parts: The probability of selecting configuration j given configuration i, $g(i \to j)$, times a probability of accepting the selected move, $A(i \to j)$ yields

$$\begin{aligned} P_i W(i \to j) &= P_j W(j \to i) \\ P_i g(i \to j) A(i \to j) &= P_j g(j \to i) A(j \to i). \end{aligned} \tag{2.30}$$

Inserting the probability distribution as the wave function squared and the selection probability as the Green's function, the expression becomes

$$\begin{aligned} |\psi_i|^2 G(i \to j) A(i \to j) &= |\psi_j|^2 G(j \to i) A(j \to i) \\ \frac{A(j \to i)}{A(i \to j)} &= \frac{G(i \to j)}{G(j \to i)} \frac{|\psi_i|^2}{|\psi_j|^2} \equiv R_G(j \to i) R_\psi(j \to i)^2, \end{aligned} \tag{2.31}$$

where the defined ratios correspond to the Green's function - and wave function ratios respectively.

Assume now that configuration i has a higher overall probability than configuration j. The essence of the Metropolis algorithm is that the step is automatically accepted, that is, $A(i \to j) = 1$. In other words: A more effective thermalization is obtained by accepting all these moves. What saves Metropolis from breaking the criteria of ergodicity is the fact that suggested moves to lower probability states are not automatically rejected. This is demonstrated by solving Eq. (2.31) for the case where $A(i \to j) = 1$, that is, the case where $P_i < P_j$. This yields

$$A(j \to i) = R_G(j \to i) R_\psi(j \to i)^2.$$

Concatenating both scenarios yields the following acceptance/rejection rules:

$$\boxed{A(i \to j) = \begin{cases} R_G(i \to j) R_\psi(i \to j)^2 & R_G(i \to j) R_\psi(i \to j)^2 < 1 \\ 1 & \text{else} \end{cases}} \tag{2.32}$$

Or more simplistic:

$$A(i \to j) = \min\{R_G(i \to j) R_\psi(i \to j)^2, 1\} \tag{2.33}$$

In the isotropic diffusion case, the Greens function ratio cancels due to symmetry, $R_G(i \rightarrow j) = 1$, leaving us with the standard Metropolis algorithm:

$$A(i \rightarrow j) = \min\{R_\psi(i \rightarrow j)^2, 1\} \tag{2.34}$$

On the other hand, for Fokker-Planck diffusion, there will be no cancellation of the Green's function. Inserting Eq. (2.22) into Eq. (2.32) results in the *Metropolis Hastings algorithm.* The ratio of Green's function can be evaluated efficiently by simply subtracting the exponents of the exponentials. This is demonstrated by calculating the logarithm

$$\begin{aligned} \log R_G^{\mathrm{FP}}(i \rightarrow j) &= \log\big(G_{\mathrm{Diff}}^{\mathrm{FP}}(j \rightarrow i)/G_{\mathrm{Diff}}^{\mathrm{FP}}(i \rightarrow j)\big) \\ &= \frac{1}{2}\big(F(x_j) + F(x_i)\big)\big(\frac{1}{2}D\delta t(F(x_j) - F(x_i)) + x_i - x_i\big) \end{aligned} \tag{2.35}$$

$$A(i \rightarrow j) = \min\{\exp\big(\log R_G^{\mathrm{FP}}(i \rightarrow j)\big)\, R_\psi(i \rightarrow j)^2, 1\} \tag{2.36}$$

Derived from detailed balance, the Metroplis Algorithm is as a must-have when it comes to Markov Chain Monte Carlo. Besides Quantum Monte Carlo, methods such as the *Ising Model* solvers greatly benefit from these rules [6].

In practice, without the Metropolis sampling, the ensemble of walkers will not span that of the trial wave function. This is due to the fact that the time-step used in simulations are finite, and the trial positions of walkers are random. A chart flow describing the implementation of the Metropolis algorithm and the diffusion process in general is given in Fig. 2.1.

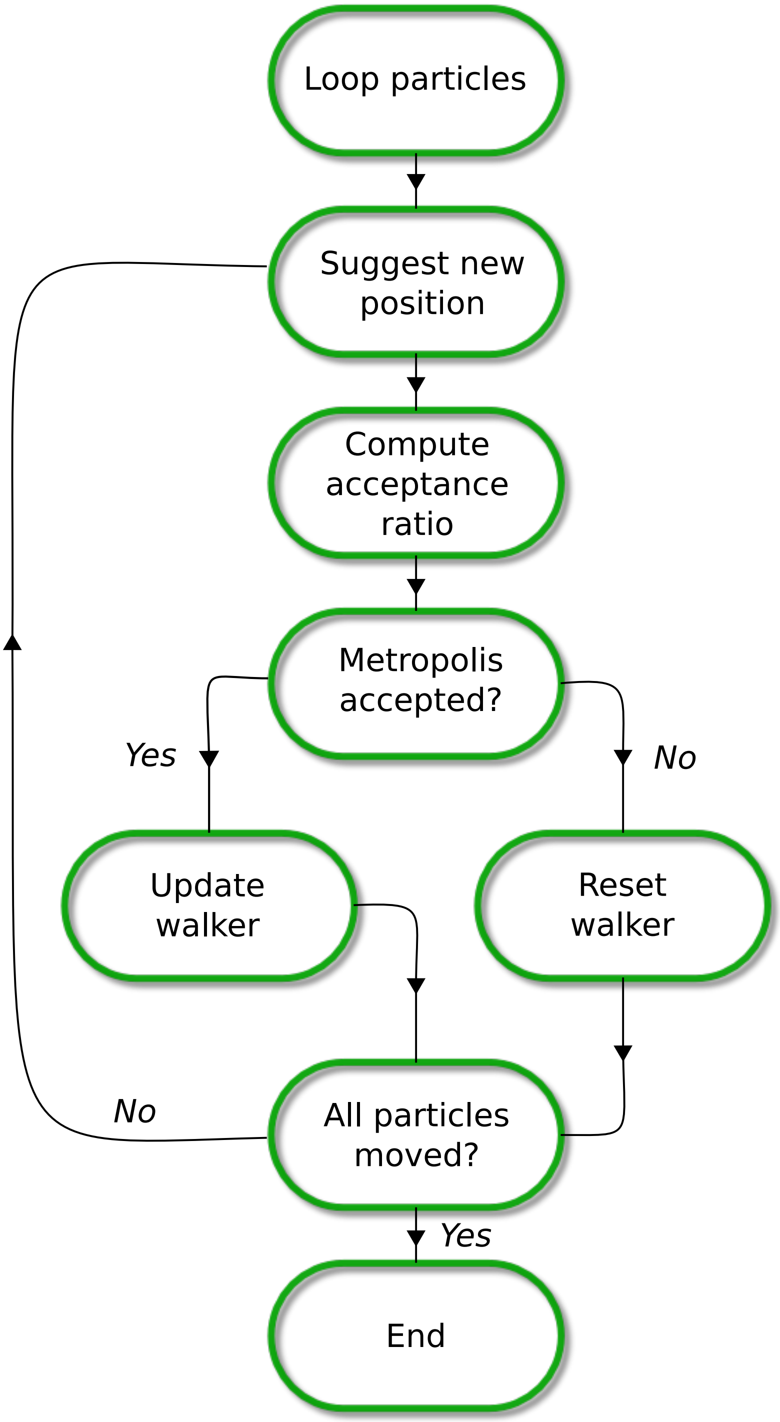

Figure 2.1: Flow chart for iterating a walker through a single time step, i.e. simulation the application of the Green's function from Eq. (2.15) using the Metropolis algorithm. New positions are suggested according to the chosen diffusion model.

2.5 The Process of Branching

In the previous section it became clear that the Metropolis test will guide the walkers to span the trial wave function. This implies that without further action, no changes to the distribution can be made, and the point of modeling the projection operator from Eq. (2.5) is rendered useless. The important fact to include is that the branching Green's function from Eq. (2.38) and (2.37) distribute weights to the walkers, effectively altering the spanned distribution.

The process of branching in Quantum Monte-Carlo is simulated by the creation and destruction of walkers with probability equal to that of the branching Green's function [2]. The explicit shapes in case of isotropic diffusion (ISO) and Anisotropic (FP) is

$$\begin{aligned} G_B^{\mathrm{ISO}}(i \to j) &= e^{-\left(\frac{1}{2}[V(x_i)+V(x_j)]-E_T\right)\delta\tau} \qquad (2.37) \\ G_B^{\mathrm{FP}}(i \to j) &= e^{-\left(\frac{1}{2}[E_L(x_i)+E_L(x_j)]-E_T\right)\delta\tau}, \qquad (2.38) \end{aligned}$$

where $E_L(x_i)$ is the energy evaluated in configuration x_i (see Section 2.6.3 for details). The three different scenarios which arise is

- $G_B = 1$: No branching, proceed main loop.
- $G_B = 0$: The current walker is to be removed from the current ensemble.
- $G_B > 1$: Make on average $G_B - 1$ replicas of the current walker.

Defining the following quantity allows for an efficient simulation of this behavior

$$\boxed{\overline{G}_B = \mathrm{floor}\left(G_B + a\right), \qquad (2.39)}$$

where a is a uniformly distributed number on $[0, 1)$. The chance that $\overline{G}_B = G_B + 1$ is then equal to $G_B - \mathrm{floor}(G_B)$. As an example, assume $G_B = 3.3$. The value of $\overline{G}_B$ is then either three or four, depending on whether $a < 0.7$ or not. The probability that $a < 0.7$ is obviously 70%, implying that there is a 30% chance that $\overline{G}_B$ is equal to four, and 70% chance that is is equal to three.

The process of branching is demonstrated in Fig. 2.2.

There are some programming challenges due to the fact that the number of walkers is not conserved, such as cleaning up inactive walkers and stabilizing the population across different computational nodes. For details regarding this, see the actual code reference at ref. [7]. Isotropic diffusion is in practice never used with branching due to the singularities in the Coulomb interaction (see Eq. (2.37)) This may lead to large fluctuations in the walker population, the exact opposite to the optimal behavior.

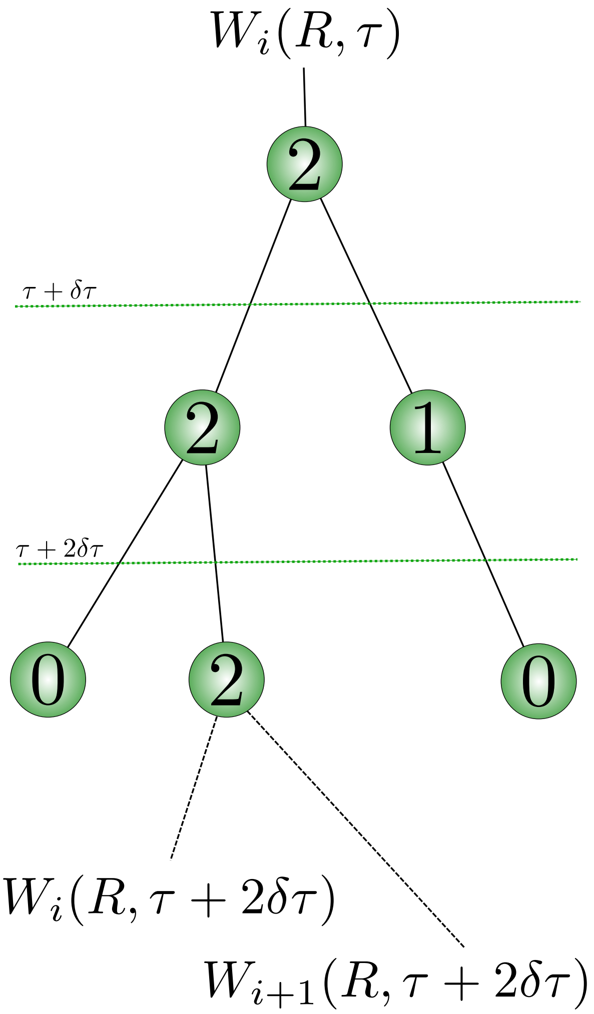

Figure 2.2: Branching illustrated. The initial walker $W_i(R,\tau)$ is branched according to the rules of Section 2.5. The numerical value inside the nodes represents $\overline{G}_B$ from Eq. (2.39). Each horizontal dashed line represent a diffusion step, i.e. a transition in time. Two lines exiting the same node represent identical walkers. After moving through the diffusion process, not two walkers should ever be equal (given that not all of the steps was rejected).

2.6 The Trial Wave Function

Recall Eq. (2.3)-(2.6). The initial condition, $\Phi(\vec{r}, t = 0)$, is in Quantum Monte-Carlo referred to as the trial wave function. Mathematically, we may choose any normalizable wave function, whose overlap with the exact ground state wave function, $\Psi_0(x)$, is non-zero. If the overlap is zero, that is, $C_0 = 0$ in Eq. (2.6), the entire diffusion formalism breaks down, and no final state of convergence can be reached. On the other hand, the opposite scenario implies the opposite behavior; the closer C_0 is to unity, the more rapidly $\Psi_0(x)$ will become the dominant contribution to our distribution.

Before getting into specifics, a few notes on many-body theory is needed. From this point on, all particles are assumed to be identical. For more information regarding basic Quantum Mechanics, I suggest reading ref. [1]. For mathematically rigid derivations of concepts, see ref. [8]. More details regarding many-body theory can be found in ref. [9].

2.6.1 Many-body Wave Functions

Many-body theory arise from the fact that we have *many-body interactions*, e.g. the Coulomb interaction between two particles. If this was not the case, i.e. for non-interacting particles, the full system would decouple into N single particle systems.

Finding the ground state is not surprisingly equivalent to solving the time-independent Schrödinger Equation

$$\widehat{\mathbf{H}}\Psi_0(x) = E_0\Psi_0(x), \tag{2.40}$$

where $x \equiv \{x_1, x_2, ..., x_N\}$. Exact solutions to realistic many-body systems rarely exist, however, like in Section 2.1.1, expanding the solution in a known basis is always legal, which reduces the problem into that of a *coefficient hunt*

$$\Psi_0(x) = \sum_{k=0}^{\infty} C'_k \Phi_k(x). \tag{2.41}$$

Different many-body methods, e.g. *Hartree Fock*[5] and genetic algorithms, give rise to different ways of hunting down these coefficients, however, certain concepts are necessarily common, for instance truncating the basis at some level, K:

$$\Psi_0(x) = \sum_{k=0}^{K} \tilde{C}'_k \Phi_k(x). \tag{2.42}$$

The many-body basis elements $\Phi_k(x)$ are constructed using N elements from a basis of single particle wave functions (or *orbitals* for short), $\phi_n(x_i)$, combined in different ways. The process of calculating basis elements often boils down to a combinatoric exercise involving combinations of orbitals.

Imagine electrons surrounding a nucleus, i.e an atom; a single electron occupying state n at a position x_i is then described by the orbital $\phi_n(x_i)$. Each unique[6] configuration of electrons will give rise to one

[5] Hartree-Fock is roughly a basis change from the non-interacting case into a basis which is orthogonal to one-particle excitations. The exact ground state wave function should be orthogonal to all excited states, so it's a fair approximation depending on the dominance of one-particle excitations in the given system.

[6] Two wave functions are considered equal if they differ by nothing but a phase factor.

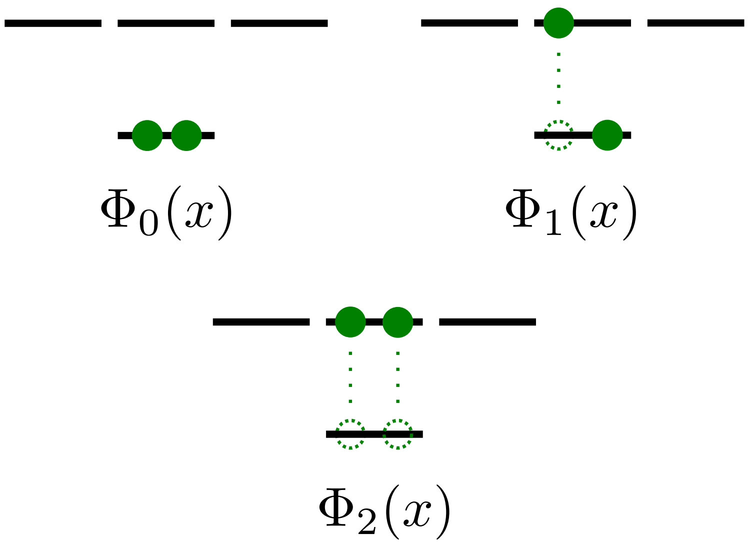

Figure 2.3: Three different electron configurations in an shell structure making up three different $\Phi_k(x)$, i.e. constituents of the many-body basis described in Eq. (2.42). An electron (solid dot) is represented by e.g. the orbital $\phi_{1s}(x_1)$.

unique $\Phi_k(x)$. In other words, the complete basis of $\Phi_k(x)$ is described by the collection of all possible excited states. $\Phi_0(x)$ is the ground state of the atom, $\Phi_1(x)$ has one electron exited to a higher shell, $\Phi_2(x)$ has another, and so on. See Fig. 2.3 for a demonstration of this.

In other words, constructing a single solution of a many-body problem involves three steps:

Step one	Choose a set of orbitals $\phi_n(x_i)$.
Step two	Construct $\Phi_k(x)$ from $N\times$ $\phi_n(x_i)$.
Step three	Construct $\Psi_k(x)$ from $K\times$ $\phi_k(x)$.

The last step is well described by Eq. (2.42), but is seldom necessary to perform explicitly; expressions based on the calculated wave functions is given in terms of the constituents and their weights.

Step one in detail

The Hamiltonian for a N-particle system is

$$\widehat{\mathbf{H}} = \widehat{\mathbf{H}}_0 + \widehat{\mathbf{H}}_{\mathrm{I}}, \tag{2.43}$$

where $\widehat{\mathbf{H}}_0$ and $\widehat{\mathbf{H}}_{\mathrm{I}}$ are respectively the one-body and the many-body term. As an approximation, we truncate the many-body interactions at the Coulomb level. The one-body term consist of the external potential and the kinetic terms for all particles.

$$
\begin{aligned}
\widehat{\mathbf{H}}_0 &= \sum_{i=1}^{N} \widehat{\mathbf{h}}_0(x_i) && (2.44)\\
&= \sum_{i=1}^{N} \widehat{\mathbf{t}}(x_i) + \widehat{\mathbf{u}}_{\text{ext}}(x_i) \\
\widehat{\mathbf{H}}_{\text{I}} &\simeq \sum_{i<j=1}^{N} \widehat{\mathbf{v}}(r_{ij}) && (2.45)\\
&= \sum_{i<j=1}^{N} \frac{1}{r_{ij}}
\end{aligned}
$$

The single particle orbitals are chosen to be the solutions of the non-interacting case (given that they exist)

$$\widehat{\mathbf{h}}_0(x_i)\phi_n(x_i) = \epsilon_n \phi_n(x_i). \tag{2.46}$$

If no such choice can be made, choosing an generally suited basis, e.g. free-particle solutions, is the general strategy.

Step two in detail

In the case of *Fermions*, i.e. half-integer spin particles like electrons, protons, etc., $\Phi_k(x)$ is an anti-symmetric function[7] on the form of a determinant: The *Slater determinant.* The shape of the determinant is given in Eq. (2.47). The anti-symmetry is a direct consequence of the *Pauli Exclusion Principle*: At any given time, two fermions cannot occupy the same state.

Bosons on the other hand, have symmetric wave functions (see Eq. (2.48)), which in many ways are easier to deal with because of the lack of an exclusion principle. In order to keep the terminology less abstract and confusing, from here on, the focus will be on systems of fermions.

$$
\begin{aligned}
\Phi_0^{\text{AS}}(x_1, x_2, ..., x_N) &\propto \sum_{\text{P}} (-)^{\text{P}} \widehat{\mathbf{P}} \phi_1(x_1)\phi_2(x_2) \ldots \phi_N(x_N) \\
&= \begin{vmatrix} \phi_1(x_1) & \phi_2(x_1) & \cdots & \phi_N(x_1) \\ \phi_1(x_2) & \phi_2(x_2) & \cdots & \phi_N(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_N) & \phi_2(x_N) & \cdots & \phi_N(x_N) \end{vmatrix} && (2.47)\\
\Phi_0^{\text{S}}(x_1, x_2, ..., x_N) &\propto \sum_{\text{P}} \widehat{\mathbf{P}} \phi_1(x_1)\phi_2(x_2) \ldots \phi_N(x_N) && (2.48)
\end{aligned}
$$

Comment to Eq. (2.47) and Eq. (2.48): The permutation operator $\widehat{\mathbf{P}}$ is simply a way of writing *in any combination of particles and states*, hence the combinatoric exercise mentioned previously.

[7] Interchanging two particles in an anti-symmetric wave function will reproduce the state changing only the sign.

Dealing with correlations

The contributions to the sum on the right-hand side in Eq. (2.41) for $k > 0$ are often referred to as *correlation* terms. Given that the orbital wave functions are chosen by Eq. (2.46), the existence of the correlation terms, i.e. $C'_k \neq 0$ for $k > 0$, follows as a direct consequence of the many-body interaction, H_{I}.

As an example, imagine performing an energy calculation with two particles being infinitely close; the Coulomb singularity will cause the energy to blow up. However, if we perform the calculation using the exact wave function, the diverging terms will cancel out; the energy is position independent.

In other words, incorporating the correct correlated wave function will result in a cancellation in the diverging term as we approach singularities, a property which brings us closer to the exact wave function.

These criteria are called *Cusp Conditions*, and serve as a powerful guide when it comes to selecting a trial wave function.

2.6.2 Choice of Trial Wave function

Choosing the trial wave function boils down to optimizing the overlap described in the introduction using a priori knowledge about the system at hand. As discussed previously, the optimal choice of single particle basis is eigenfunctions of the non-interacting case (given that they exist). Starting from Eq. (2.42), the *spatial wave function*,the first step is to make sure the cusp conditions are obeyed.

Introducing the correlation functions $f(r_{ij})$, where r_{ij} is the relative distance between particle i and j, the general *anzats* for the trial wave function becomes

$$\Psi_T(x_1, ..., x_N) = \Big[\sum_{k=0}^{K} C_k \Phi_k(x_1, ..., x_N)\Big] \prod_{i<j}^{N} f(r_{ij}) \tag{2.49}$$

Explicit shapes

Several models for the correlation function exist, however, some are less practical than others. An example given in ref. [2] demonstrates this nicely: Hylleraas presented the following correlation function

$$f(r_{ij})_{\text{Hylleraas}} = e^{-\frac{1}{2}(r_i + r_j)} \sum_k d_k (r_{ij})^{a_k} (r_i + r_j)^{b_k} (r_i - r_j)^{e_k}, \tag{2.50}$$

where all k-subscripted parameters are free. Calculating the helium ground state energy using this correlation function with nine terms yields a four decimal precision. Eight digit precision is achieved by including almost 1078 terms. For the purpose of Quantum Monte-Carlo this is beyond overkill.

A more well suited correlation function is the *Padé Jastrow* function (skipping some redundant indices)

$$\begin{aligned} \prod_{i<j}^{N} f(r_{ij}) &= \exp(U) \\ U &= \sum_{i<j}^{N} \left(\frac{\sum_k a_k r_{ij}^k}{1 + \sum_k \beta_k r_{ij}^k} \right) + \sum_{i}^{N} \left(\frac{\sum_k a'_k r_i^k}{1 + \sum_k \alpha_k r_i^k} \right). \end{aligned}$$

For systems where the correlations are well behaving, it is custom to drop the second term, and keep only the $k = 1$ term, i.e.

$$f(r_{ij}; \beta) = \exp\left(\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}\right), \tag{2.51}$$

where β is a variational parameter, and a_{ij} is a spin-dependent constant tuned to obey the cusp condition.

Shifting the focus back to the spatial wave function, in the case of a fermionic system, the evaluation of a $N \times N$ Slater determinant is holding back the effectiveness of many-particle simulations. However, assuming we have a spin-independent Hamiltonian, we can split the spatial wave function is two; one part for each spin eigenvalue. A detailed derivation of this is given in the appendix of ref. [10]. Assuming spin-half particles we get

$$\Psi_T(x_1, ..., x_N; \beta) = \Big[\sum_{k=0}^{K} C_k \tilde{\Phi}_k(x_1, ..., x_{\frac{N}{2}}) \tilde{\Phi}_k(x_{\frac{N}{2}-1}, ..., x_N)\Big] \prod_{i<j}^{N} f(r_{ij}; \beta). \tag{2.52}$$

Since the particles are identical, we are free to say that the first half are spin up, and the second spin down, hence the splitting as above. For simplicity, the spin up determinant will from here on be labeled $D^{\uparrow}$, and the spin down one $D^{\downarrow}$. Stitching everything together yields the following explicit shape for a spin-independent Hamiltonian using a one-parameter Padé Jastrow function

$$\boxed{\Psi_T(x_1, ..., x_N; \beta) = \sum_{k=0}^{K} C_k D_k^{\uparrow} D_k^{\downarrow} \prod_{i<j}^{N} f(r_{ij}; \beta).} \tag{2.53}$$

This shape is referred to as a *Multi-determinant* trial wave function.

Limitations

Depending on the complexity of the system at hand, we need more complicated trial wave functions. However, it is important to distinguish between simply integrating a trial wave function, and performing the full diffusion calculation. As a reminder: Simple integration will not be able to tweak the distribution; what you have is what you get. Solving the diffusion problem, on the other hand, will alter the distribution from that of the trial wave function ($t = 0$) into a distribution closer to the exact wave function by Eq. (2.6).

Because of this fact, our limitations due to the trial wave function is far less than what is the case of standard Monte-Carlo integration. A heavier trial wave function might convergence faster, but at the expense of being more CPU-intensive. This means that we are in a position to trade CPU-time per walker for convergence time. For systems of many particles, CPU-time per walker needs to be as low as possible in order to get the computation done in a reasonable amount of time, i.e., the choice of trial wave function needs to be done in light of the system at hand, and the specific aim of the computation.

Single Determinant Trial Wave function

In the case of well-behaving systems, a single determinant converges rapidly enough. This simplicity opens up the possibility of simulating large systems efficiently. This will be referred to as a *single determinant* trial wave function, and serve as a very simple approximation. In order to optimize the overlap with the exact wave function, a variational parameter α is introduced in the spatial part (from Eq. (2.51), we already have β)

$$\boxed{\Psi_T(x_1, ..., x_N; \alpha, \beta) = D^{\uparrow}(\alpha) D^{\downarrow}(\alpha) \prod_{i<j}^{N} f(r_{ij}; \beta). \qquad (2.54)}$$

Determining optimal values for the variational parameters will be discussed in Section 2.6.5.

2.6.3 Calculating Expectation Values

The expectation value of an operator $\widehat{\mathbf{O}}$ is sampled through *local* values, $O_L(x)$

$$\begin{aligned} \langle \Psi_T | \widehat{\mathbf{O}} | \Psi_T \rangle &= \int \Psi_T(x)^* \widehat{\mathbf{O}} \Psi_T(x) \mathrm{d}x \\ &= \int |\Psi_T|^2 \left(\frac{1}{\Psi_T(x)} \widehat{\mathbf{O}} \Psi_T(x) \right) \mathrm{d}x \\ &= \int |\Psi_T|^2 O_L(x) \mathrm{d}x \qquad (2.55) \\ O_L(x) &= \frac{1}{\Psi_T(x)} \widehat{\mathbf{O}} \Psi_T(x) \qquad (2.56) \end{aligned}$$

Discretizing the integral (and thus introducing an error) yields

$$\langle \Psi_T | \widehat{\mathbf{O}} | \Psi_T \rangle \equiv \langle O \rangle \simeq \frac{1}{n} \sum_{i=1}^{n} O_L(x_i) \equiv \overline{O}, \qquad (2.57)$$

where x_i is a random variable following the distribution of the trial wave function. The *ensemble average*, $\langle O \rangle$ will, given ergodicity, equal the estimated average $\overline{O}$ in the limit $n \to \infty$, i.e.

$$\langle O \rangle = \lim_{n \to \infty} \overline{O} = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} O_L(x_i) \qquad (2.58)$$

In the case of the energy estimation, this means that once our walkers hit equilibrium, we can start sampling local values based on their configurations x_i. In the case of energies, we get

$$\langle E \rangle \simeq \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{\Psi_T(x_i)} \left(-\frac{1}{2} \nabla^2 \right) \Psi_T(x_i) + V(x_i) \right) \qquad (2.59)$$

2.6.4 Normalization

Every explicit calculation using the trial wave function in Quantum Monte-Carlo involves taking ratios. Ratios implies cancellation of the normalization factors. Eq. (2.32) from the Metropolis section, the Quantum Force in the Fokker-Planck equation, and the sampling of local values describes in the previous section demonstrates exactly this; everything involves ratios.

Not having to normalize our wave functions saves us a lot of CPU-time, but it also relieves us of complicated normalization factors in our single particle basis expressions; any constants multiplying $\phi_n(x_i)$ in Eq. (2.47) and Eq. (2.48), can be taken outside the sum over permutations, and will hence cancel when the ratio between two wave functions constituting of the same single particle orbitals are computed (note: Single determinant only).

In the case of multi-determinants trial wave functions, the normalization factors from the single particle basis elements will be absorbed by the respective determinant's coefficient C_k, and is hence obsolete in this case as well.

2.6.5 Selecting Optimal Variational Parameters

All practical ways of determining the optimal values of the variational parameters originate from the same powerful principle: *The Variational Principle.* The easiest way of demonstrating the principle is to evaluate the expectation value of the energy, using an approach similar to what used in Eq. (2.6)

$$\begin{aligned} E_0 &= \langle\Psi_0|\,\widehat{\mathbf{H}}\,|\Psi_0\rangle \\ E &= \langle\Psi_T(\alpha,\beta)|\,\widehat{\mathbf{H}}\,|\Psi_T(\alpha,\beta)\rangle \\ &= \sum_{kl} C_k^* C_l \underbrace{\langle\Psi_k|\,\widehat{\mathbf{H}}\,|\Psi_l\rangle}_{E_k\delta_{kl}} \\ &= \sum_k |C_k|^2 E_k \end{aligned}$$

Using further that E_0 is the lowest energy eigenvalue, we can, just as done with the time propagator, introduce $E_k = E_0 + \delta E_k$ to simplify the arguments

$$\begin{aligned} E &= \sum_k |C_k^2|(E_0 + \delta E_k) \\ &= E_0 \underbrace{\sum_k |C_k^2|}_{1} + \underbrace{\sum_k |C_k|^2 \delta E_k}_{\geq 0} \\ &\geq E_0 \end{aligned}$$

The conclusion is stunning: No matter how we choose our trial wave function, we will never undershoot the exact ground state energy. This basically means that the problem of choosing variational parameters boils down to a minimization problem in the parameters space (we assume no maximum exist for finite parameter values)

$$\frac{\partial\langle E\rangle}{\partial\alpha_i} = \frac{\partial}{\partial\alpha_i}\langle\Psi_T(\alpha_i)|\,\widehat{\mathbf{H}}\,|\Psi_T(\alpha_i)\rangle = 0 \tag{2.60}$$

In order to work with Eq. (2.60) in practice, we need to rewrite it in terms of known values. Since our wave function is dependent on the variational parameter, we need to include the expression for the normalization factor before we can expand the expression

$$
\begin{aligned}
\frac{\partial\langle E\rangle}{\partial\alpha_i} &= \frac{\partial}{\partial\alpha_i}\frac{\langle\Psi_T(\alpha_i)|\,\widehat{\mathbf{H}}\,|\Psi_T(\alpha_i)\rangle}{\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle} \\
&= \frac{\left(\langle\Psi_T(\alpha_i)|\,\frac{\partial}{\partial\alpha_i}\widehat{\mathbf{H}}\,|\Psi_T(\alpha_i)\rangle + \langle\Psi_T(\alpha_i)|\,\widehat{\mathbf{H}}\frac{\partial}{\partial\alpha_i}\,|\Psi_T(\alpha_i)\rangle\right)}{\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle^2}\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle \\
&- \langle\Psi_T(\alpha_i)|\,\widehat{\mathbf{H}}\,|\Psi_T(\alpha_i)\rangle\frac{\left(\langle\Psi_T(\alpha_i)|\,\frac{\partial}{\partial\alpha_i}\right)|\Psi_T(\alpha_i)\rangle + \langle\Psi_T(\alpha_i)|\left(\frac{\partial}{\partial\alpha_i}\,|\Psi_T(\alpha_i)\rangle\right)}{\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle^2},
\end{aligned}
$$

where the product and division rules for derivatives have been applied. The Hamiltonian does not depend on the variational parameters, and hence both terms in the first expansion is equal. Cleaning up the expression yields

$$
\boxed{
\begin{aligned}
\frac{\partial\langle E\rangle}{\partial\alpha_i} &= 2\left(\frac{\langle\Psi_T(\alpha_i)|\,\widehat{\mathbf{H}}\frac{\partial}{\partial\alpha_i}\,|\Psi_T(\alpha_i)\rangle}{\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle} - \langle E\rangle\frac{\langle\Psi_T(\alpha_i)|\,\frac{\partial}{\partial\alpha_i}\,|\Psi_T(\alpha_i)\rangle}{\langle\Psi_T(\alpha_i)|\Psi_T(\alpha_i)\rangle}\right) \\
&= 2\left(\left\langle E\frac{\partial\Psi_T}{\partial\alpha_i}\right\rangle - \langle E\rangle\left\langle\frac{\partial\Psi_T}{\partial\alpha_i}\right\rangle\right) \qquad (2.61)
\end{aligned}
}
$$

Using this expression for the *variational gradient* means we can calculate the derivatives exactly the same way we calculate the energy, and use these derivatives to move in the direction of the variational minimum in Eq. (2.60).

This strategy give rise to numerous ways of finding the optimal parameters, such as using the well known Newton's method, conjugate gradient methods [11], steepest descent (similar to Newton's method), and many more. The method implemented for this thesis is called *Adaptive Stochastic Gradient Descent*, and is an efficient iterative algorithm for seeking the variational minimum.

2.7 Gradient Descent Methods

The direction of a gradient serves as a guide to extremal values. Gradient Descent, also called Steepest Descent, is a family of minimization methods using this property of gradients in order to back trace a local minimum in the vicinity of an initial guess.

2.7.1 General Gradient Descent

Seeking maxima or minima is simply a question of whether the positive or negative direction of the gradient is followed. Imagine a function, $f(x)$, with a minimum residing at $x = x_m$. The information at hand is then

$$
\begin{aligned}
\nabla f(x_m) &= 0 \qquad (2.62)\\
\nabla f(x_m - \mathrm{d}x) &< 0 \qquad (2.63)\\
\nabla f(x_m + \mathrm{d}x) &> 0 \qquad (2.64)
\end{aligned}
$$

where dx is a infinite decimal displacement.

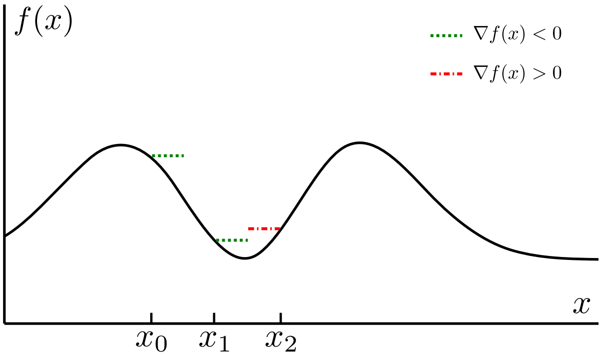

Figure 2.4: Two steps of a one dimensional Gradient Descent process. Steps are taken in the direction of the negative gradient (indicated by dotted lines).

Assume now that we start from an initial position x_0, and measure the direction of the gradient before we take a step in that direction. From Fig. 2.4 and the previous equations, we see that once we cross the minimum, the gradient changes sign. The brute force way of minimizing is to stop once the gradient changes sign, however, this would require extremely many extremely small steps in order to achieve good precision. The difference equation for the brute force case would be

$$x_{i+1} = x_i - \delta \frac{\nabla f(x_i)}{|\nabla f(x_i)|} \tag{2.65}$$

A better algorithm is to continue iterating even though the minimum is crossed. The brute force scheme breaks down in this case, oscillating between two points, e.g. x_1 and x_2 in Fig. 2.4, because of the constant step length δ. To counter this, a changing step length δ_i is introduced

$$\boxed{x_{i+1} = x_i - \delta_i \nabla f(x_i) \qquad (2.66)}$$

All Gradient/Steepest Descent methods is in principle described by Eq. (2.66). Some examples are

• Brute Force I	$\delta_i = \delta \frac{1}{|\nabla f(x_i)|}$
• Brute Force II	$\delta_i = \delta$
• Monotone Decreasing	$\delta_i = \delta / i^2$
• Newton's Method	$\delta_i = \frac{1}{\nabla^2 f(x_i)}$

Iterative gradient methods will only reveal one local extrema, depending on the choice of x_0 and δ. In order to find several extrema, multiple unique processes can be run. Calculating the local gradient is simply a finite difference calculation (assuming the analytic expression is not known).

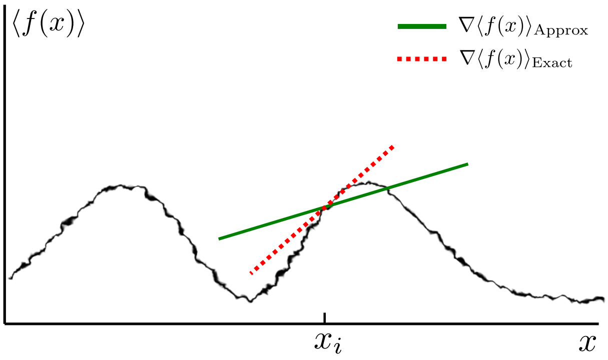
$\langle f(x)\rangle$
$\nabla\langle f(x)\rangle_{\text{Approx}}$
$\nabla\langle f(x)\rangle_{\text{Exact}}$
x_i
x

Figure 2.5: A one dimensional plot of an expectation valued function. Smeared lines are representing uncertainties due to rough sampling. The direction of the local gradient (solid green line) at a point x_i is not necessarily a good estimate of the actual analytic gradient (dashed red line).

2.7.2 Stochastic Gradient Descent

Minimizing stochastic quantities, such as the variance or expectation values, adds another layer of complications on top of the methods described in the previous section. Assuming a closed form expression for the stochastic quantity is unobtainable, the gradient needs to be calculated by using e.g. Monte-Carlo sampling. Eq. (2.61) is an example of such a process.

A full precise sampling of the stochastic quantities are expensive and unpractical. Stochastic Gradient methods use different techniques in order to make the sampling more effective, such as multiple walkers, thermalization, and more.

Using a finite difference scheme with stochastic quantities are dangerous, as uncertainties in the values will cause the gradient to become unstable when the variations are (expectedly) low close to the minimum. This is illustrated in Fig. 2.5.

2.7.3 Adaptive Stochastic Gradient Descent

Adaptive Stochastic Gradient Descent (ASGD) has it's roots in the mathematics of automated control theory [12]. The automated process, is that of choosing an optimal step length δ_i for the current transition $x_i \rightarrow x_{i+1}$. This process is based on the inner product of the old and the new gradient though a variable

$$X_i \equiv -\nabla_i \cdot \nabla_{i-1} \tag{2.67}$$

The step length from Eq. (2.66) is in ASGD modeled in the following manner

$$\begin{aligned}
\delta_i &= \gamma(t_i) && (2.68)\\
\gamma(t) &= a/(t+A) && (2.69)\\
t_{i+1} &= \max(t_i + f(X_i), 0) && (2.70)\\
f(x) &= f_{\text{min}} + \frac{f_{\text{max}} - f_{\text{min}}}{1 - (f_{\text{max}}/f_{\text{min}})e^{-x/\omega}} && (2.71)
\end{aligned}$$

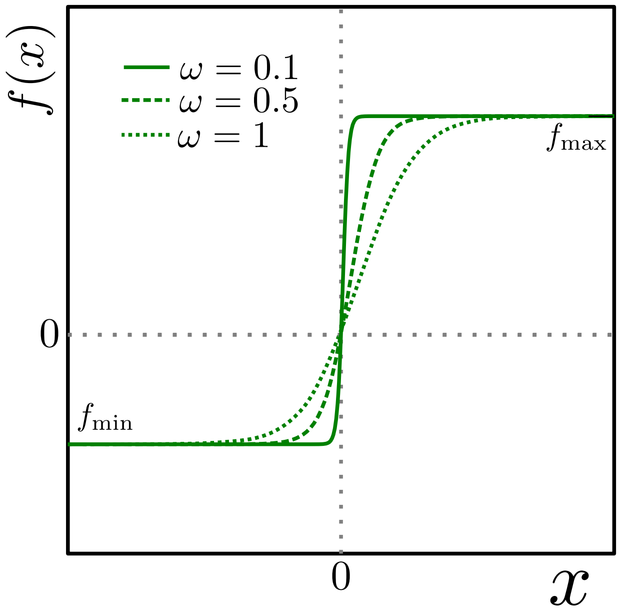

Figure 2.6: Examples of $f(X_i)$ as published in ref. [13]. As $\omega \to 0$, $f(x)$ approaches a step function.

with $f_{\text{max}} > 0$, $f_{\text{min}} < 0$, and $\omega > 0$. Free parameters are a, A, t_0, however, Ref. [13] suggests $A = 20$ and $t_0 = t_1 = A$ for universal usage.

Notice that the step length increase if t_i decrease and vice-versa. A smaller step length is sought for regions close to the minimum. The function of $f(x)$ is to alter the step length by changing the trend of t. If we are close to the minimum, a smaller step length is sought, and hence t must increase. We know from previous discussion, that if the sign of the gradient change, we have crossed the minimum. Crossing the minimum with ASGD has the following consequence

- Eq. (2.67): The value of X_i will be positive.
- Eq. (2.71): $f(X_i)$ will return a value in $[0, f_{\text{max}}]$ depending on the magnitude of X_i .
- Eq. (2.70): The value of t will increase, i.e. $t_{i+1} > t_i$.
- Eq. (2.69): The step length will decrease .

The second step regarding $f(X_i)$ can be visualized in Fig. 2.6.

Assumptions

- The statistical error in the sampled gradients are distributed with zero mean.

This is shown in ref. [13] to be true; they are normally distributed. The implication is that upon combining gradient estimates for N different processes, the accumulative error will tend to zero quickly.

- The step length $\gamma(t)$ is a positive monotone decreasing function defined on $[0, \infty)$ with maximum at $t = 0$.

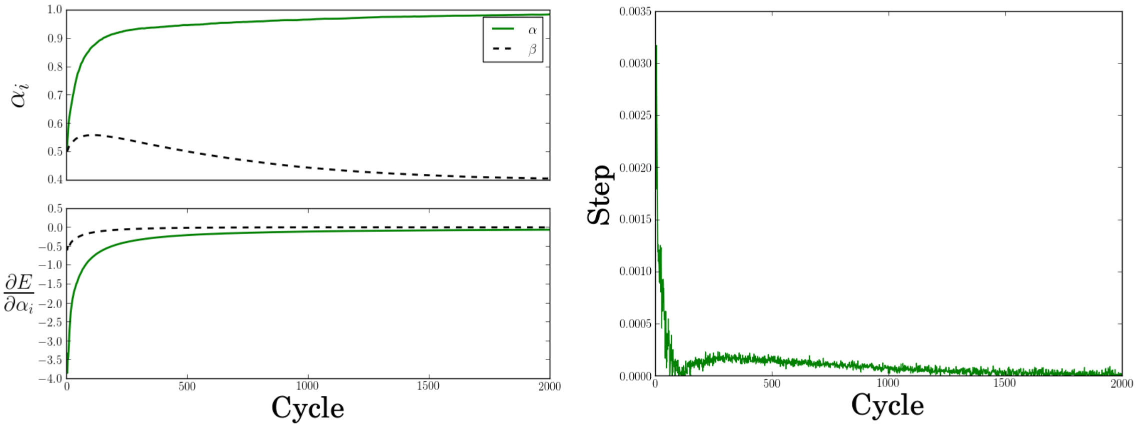

Figure 2.7: Results of Adaptive Stochastic Gradient Descent used on a two-particle Quantum Dot with unit oscillator frequency using 400 cycles pr. gradient sampling and 40 independent walkers. The right figure shows the evolution of the time step. The left figure shows the evolution of the variational parameters α and β introduced in Section 2.6 on top, and the evolution of the gradients on the bottom. The gradients are cycle averaged to reveal the pattern underlying the noise. We clearly see that they tend to zero, β somewhat before α. The step rushes to zero; we get a small rebound in the step after forcing it zero as it attempts to cross to negative values.

With $\gamma(t)$ being as in Eq (2.69), this is easily shown.

- The function $f(x)$ is continuous and monotone increasing with $f_{\text{min}} = \lim_{x\to\infty} f(x)$ and $f_{\text{max}} = \lim_{x\to-\infty} f(x)$.

This is exactly the behavior displayed in Fig. 2.6.

These assumptions and several more are described in more detail in ref. [13]. They underline that the shape and flow of the algorithm is in no way random; ASGD is optimizing minimization of stochastic quantities.

Implementation

A flow chart of the implementation is given in Fig. 2.8. For specific details regarding the implementation, i refer to the code [7]. An example of minimization using ASGD is given in Fig. 2.7.

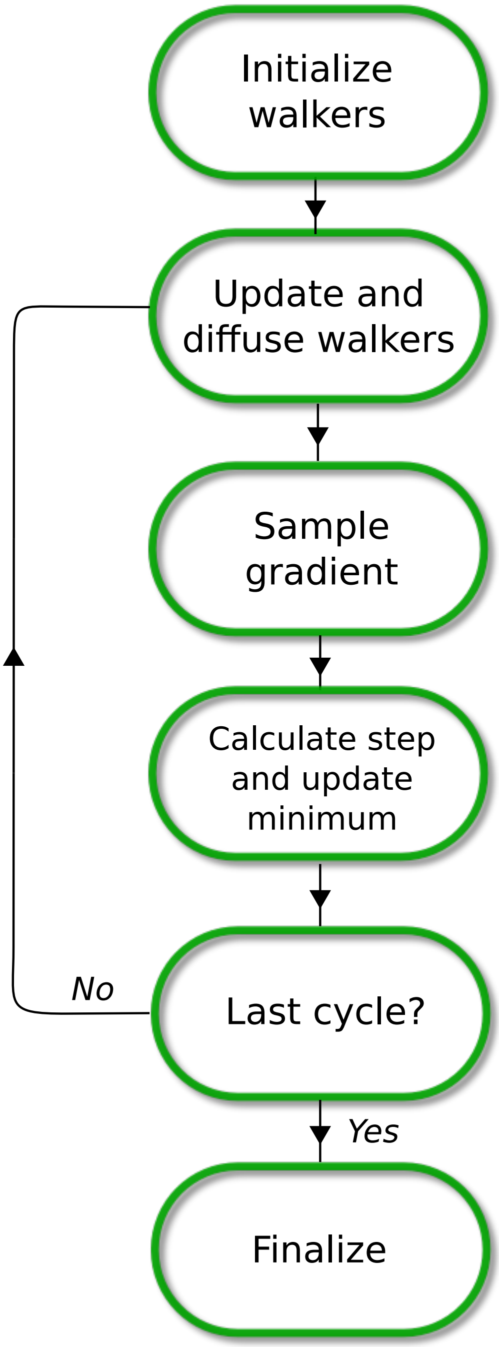

Figure 2.8: Chart flow of ASGD algorithm. Diffusing a walker is done as described in Fig. 2.1. Updating the walkers involves recalculating any values afflicted by updating the minimum. The step is calculated by Eq. (2.69). In case of Quantum Monte-Carlo, the gradient is sampled by Eq. (2.61).

2.8 Variational Monte-Carlo

As briefly mentioned in the derivation of the Quantum Monte-Carlo projection process, neglecting the branching term, i.e. force $G_{\mathrm{B}} = 1$, leaves us with a method called Variational Monte-Carlo (VMC). The naming is due to the fact that the method is variational; it supplies as upper bound for the exact ground state energy (see Section 2.6.5). The better the trial wave function, the better the answer.

Without the branching term, the optimal converged state of the Markov Chain is to span that of the trial wave function. From the flow chart of the VMC algorithm in Fig. 2.9, it is clear that VMC corresponds to nothing but a standard Monte-Carlo integration of the local energy, using the trial wave function for distributing the points.

2.8.1 Motivating the use of Diffusion Theory

The question becomes: Why bother with all the diffusion theory if the result is simply an expectation value? From statistics we know that we may use *any* distribution when calculating an expectation value. Why bother with a trial wave function, thermalization, etc.?

The reason is simple, yet not obvious. The quantity of interest, the local energy, is *dependent on a wave function* $\psi(x)$. It is not necessarily smooth, and even though the wave functions tend to zero at infinities, the local energy is *undefined*; a "0/0" expression (see Eq. (2.72). This is the case for all the roots of $\psi(x)$. Given an arbitrary distribution $P(x)$, we get

$$\begin{aligned} E_{\mathrm{VMC}} &= \int P(x)\frac{1}{\psi(x)}\widehat{\mathbf{H}}\psi(x)\mathrm{d}x \\ &\simeq \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\psi(x_i)}\widehat{\mathbf{H}}\psi(x_i), \end{aligned} \tag{2.72}$$

where the points x_i are drawn from the distribution $P(x)$.

We *need* to importance sample the integral with the distribution corresponding to that of the wave function used in the local energy in order to solve the integral in a satisfying manner, that is, avoid sampling close to the roots of $\psi(x)$ without "cheating". Introducing importance sampling is done by simply switching distributions (we do not need to scale the sampled function, we are computing an expectation value, not an arbitrary integral).

$$P(x) = |\psi(x)|^2$$

Suggesting new positions (diffusion) boils down to be analogous to calling a *random number generator* corresponding to our trial wave function squared distribution. The problem was the roots of $\psi(x)$, however, the distribution of points now share these roots, i.e. the probability of sampling a point where the local energy is undefined equals zero.

$$\psi(x_m) = 0 \quad \Longrightarrow \quad P(x_m) = 0 \tag{2.73}$$

In other words, the more undefined the energy is at a point, the less probable the point is. That is what we need, and it's what all the theory of Quantum Monte-Carlo safely delivers that standard Monte-Carlo can not.

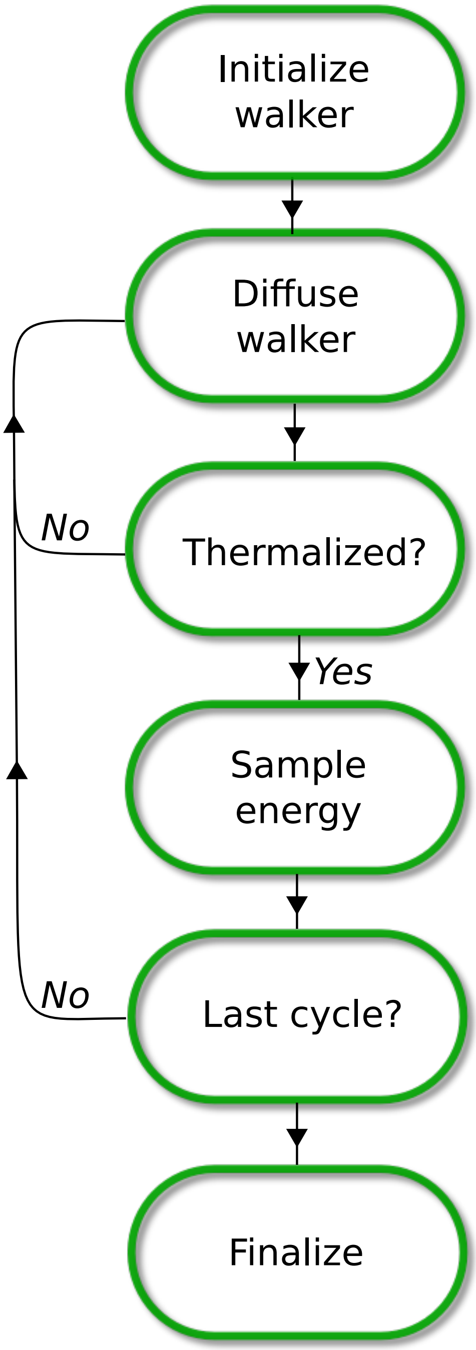

Figure 2.9: Chart flow of the Variational Monte-Carlo algorithm. The second step, *Diffuse Walker*, is the process described in Fig. 2.1. Energies are sampled as described in Section 2.6.3. Thermalization is usually set to a fixed number of cycles.

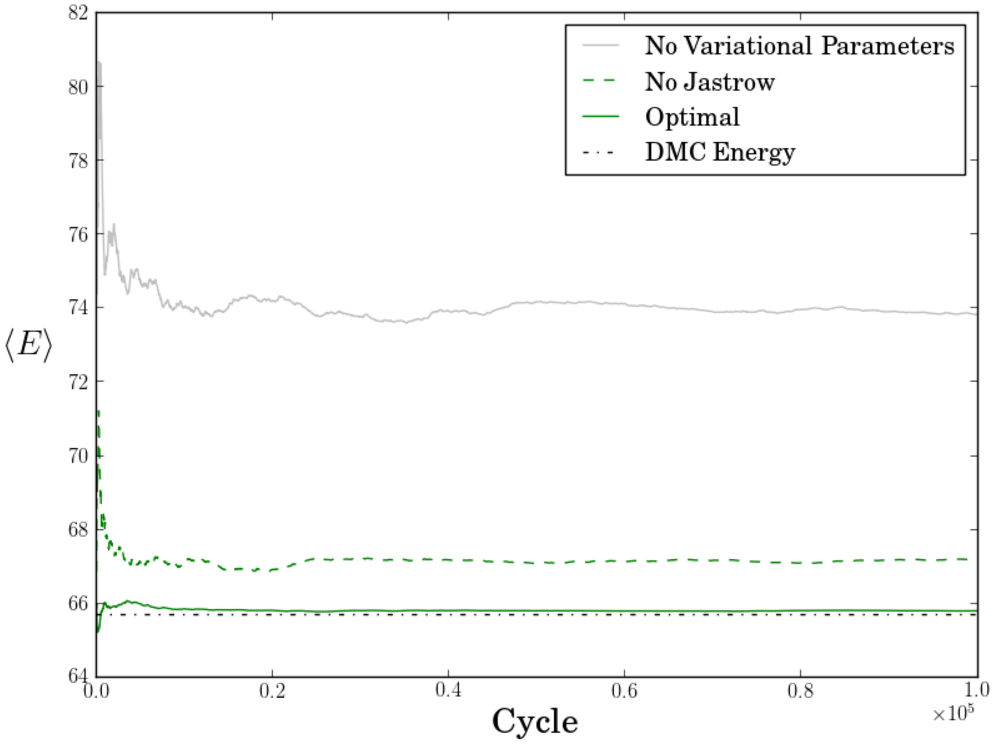

Figure 2.10: Comparison of the VMC energy using different trial wave functions. The DMC energy is believed to be very close to the exact ground state. We see that adding a variational parameter to the trial wave function lowers the energy somewhat, however, when adding the Jastrow factor described in Section 2.6.2 we get very close to the “exact” answer. Lower energy means a better energy when dealing with variational methods. In this example, a 12 particle Quantum Dot is used.

2.8.2 Implementation

Variational Monte-Carlo does not benefit much from an increase of samples (beyond a given point, that is). It is much more important that the system is thermalized, than that the number of walkers are high, or the final amount of samples are huge. It is therefore sufficient to use a single walker pr. VMC process.

The single walker is initialized in a normal distributed manner (with variance $2D\delta\tau$), and released to diffuse according to the process in Fig. 2.1. A flow chart of the VMC algorithm is given in Fig. 2.9. Finalizing the sampling involves scaling energies, calculating the variance, etc.

For details regarding specific implementations, I refer to the code in ref. [7].

2.8.3 Limitations

The only limitation in VMC is the choice of trial wave function. This makes VMC extremely robust; it will *always* produce a result. As the overlap C_0 in Eq. (2.6) approach unity, the VMC energy approaches the exact ground state energy as a monotone decreasing function. Fig. 2.10 described this effect.

2.9 Diffusion Monte-Carlo

Applying the full diffusion framework introduced in the previous sections results in a method known as Diffusion Monte-Carlo (DMC)[8]. Diffusion Monte-Carlo results are often referred to as *exact*, in the sense that it is overall one of the most precise many-body methods. Where other many-body methods run into the *curse of dimensionality* (CPU-time exponentially increasing with number of particles, precision, etc.), DMC with it's position basis Quantum Monte-Carlo formalism does not. With DMC, it is simply a matter of evaluating a more complicated trial wave function, or simulating for a longer period, in order to reach convergence to the "exact" ground state in a satisfactory way.

2.9.1 Implementation

DMC is a precise method; more statistics is like water in a desert - you can not get enough of it. In addition, branching is a major part of the algorithm. In other words: DMC uses a large ensemble of walkers to generate enormous amounts of statistics. These walkers are initialized using a VMC calculation, i.e. the walkers are distributed according to the trial wave function at $\tau = 0$.

There are three layers of loops in the DMC method implemented in this thesis. Normally one would only use two: The time-step and walker loops, however, introducing a third *block loop* within the walker loop will boost convergence dramatically. This loop continues until the current walker is either dead, or diffused n_b times. Using this method, "good" walkers will have multiple offspring pr cycle, while "bad" walkers will rarely survive the block loop. Perfect walkers will supply a ton of statistics as they surf through all the loops without branching ($G_B \sim 1$).

A flow chart of the DMC algorithm is given in Fig. 2.11.

2.9.2 Sampling the Energy

Unlike VMC, DMC does not weigh all walkers equally. It is therefore necessary to weigh each walker's contribution to a cumulative energy sampling accordingly, that is, with the branching Green's function. Let E_k denote the cumulative energy for time-step k, n_w be the number of walkers in our system at time-step k, $\tilde{n}_{b,i}$ be the number of blocks walker i survives, and let $W_i(\vec{r}, \tau)$ represent walker i. Then we have

$$E_k = \frac{1}{n_w}\sum_{i=1}^{n_w}\frac{1}{\tilde{n}_{b,i}}\sum_{l=1}^{\tilde{n}_{b,i}} G_{\mathrm{B}}\Big(W_i(\vec{r}, \tau_k + l\delta\tau)\Big)E_L\Big(W_i(\vec{r}, \tau_k + l\delta\tau)\Big) \tag{2.74}$$

As the formalism required, setting $G_B = n_w = n_b = 1$ reproduce the VMC algorithm.

The new trial energy (remember Eq. (**??**)) is set to be

$$E_T = E_k \tag{2.75}$$

The DMC energy is updated each cycle to be the trailing average of the trial energies

$$\boxed{E_{\mathrm{DMC}} = \overline{E_T} = \frac{1}{n}\sum_{k=1}^{n} E_k} \tag{2.76}$$

[8] In literature, DMC is also known as *Projection Monte-Carlo*, for reasons described in Section 2.1.1.

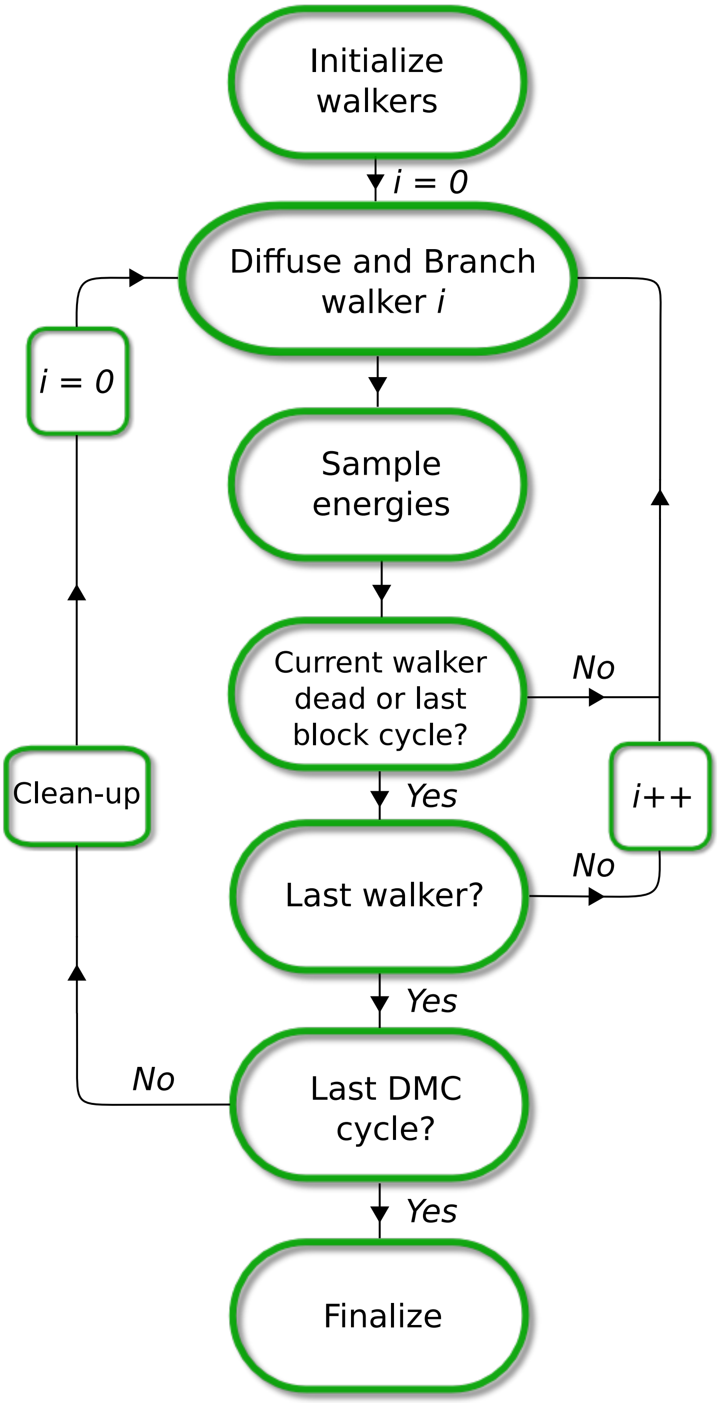

Figure 2.11: Chart flow of the Diffusion Monte-Carlo algorithm. The count variable i is the index of the walker loop. The second step, *Diffuse and Branch Walker*, is the process described in Fig. 2.1 in combination with the branching from Fig. 2.2. Energies are sampled as described in Section 2.9. Thermalization is not done in the same way as VMC (see Fig. 2.9), but rather includes the entire flow with a reduced number of DMC - and block cycles.

2.9.3 Limitations

By introducing the branching term, DMC is a far less robust method compared to VMC. Action must be taken in order to stabilize the iterations through tuning of parameters such as population size, time-step, block size, etc. This is the disadvantage of DMC compared to other many-body methods such as Coupled Cluster, which is far more *Plug and Play.*

Time Step Errors

The error introduced by the short time approximation goes as $\mathcal{O}(\delta\tau^2)$ (see Eq. (2.14)). There is a second error related to the time-step, arising from the fact that not all steps are accepted by the Metropolis algorithm. This introduces a effective reduction in the time step, and is countered by scaling the time step with the acceptance ratio upon calculating G_B. However, DMC is rarely used without importance sampling (Fokker-Planck), which, due to the Quantum Force, has an acceptance ratio ~ 1. It is therefore common to ignore this problem, and use a sufficiently low time step.

Selecting the Time-step

Studying the branching Green's function in more detail reveals that it's magnitude increase exponentially with the spread of the energies

$$G_B \propto \exp\left(\Delta E \delta\tau\right) \tag{2.77}$$

As will be shown in Section 2.11.1, the spread in energy samples are higher the worse of an approximation to the ground state our trial wave function becomes. In addition, the magnitude of the spread scales with the magnitude of the energy. Due to finite computer memory, we have N slots dedicated for storing walkers on every node. Too large branching factors may cause the system to max out the memory on one node before the walkers can be redistributed across all the nodes.

The solution is to balance out the increase in ΔE by lowering the time-step accordingly. However, too low a time-step will hinder DMC to evolve walkers. Every time we perform a time-step without having the trial energy close to the exact energy, we introduce an error to the Quantum Monte-Carlo equations (remember the transition from Eq. (2.6) to Eq. (**??**)). It is therefore necessary to have rapid convergence down to a sufficiently low energy level, where the magnitude of the exact energy error matches those of the other systematic errors.

Another source of error is due to the *fixed node approximation.* This approximation will be covered in the next section.

2.9.4 Fixed node approximation

Looking at Eq. (2.48), we notice that by choosing positive phases for our single particle wave functions, the bosonic many body wave function is exclusively positive. For fermions however, the sign change upon permuting a particle pair introduces the possibility that the wave function will have both negative and positive regions, independent of our choice of phases in the single particle wave functions.

As we simulate importance sampled DMC in time, the density of walkers at a given time, $P(x, \tau)$, is given by Eq. (2.6) multiplied by the trial wave function (see the end of Section 2.2.2 for details)

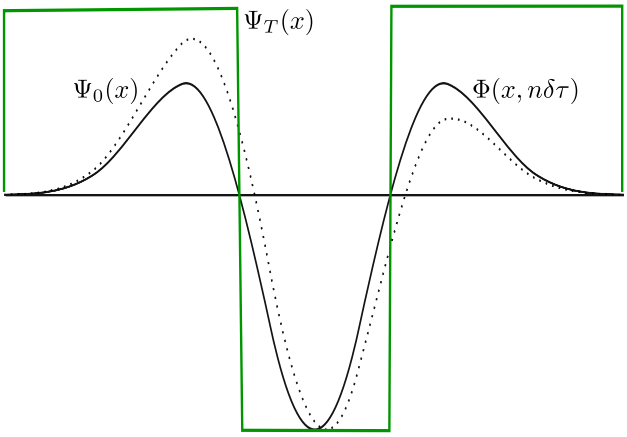

Figure 2.12: An illustration of the fixed node approximation. The dotted line is the exact ground state $\Psi_0(x)$. The distribution of walkers at cycle n, $\Phi(x, n\delta\tau)$, similar in shape with $\Psi_0(x)$, however, sharing nodes with the trial wave function $\Psi_T(x)$ (box-like function for illustration purposes), and thus making it impossible to match the true ground state exactly.

$$P(x,\tau) = \Phi(x,\tau)\Psi_T(x), \tag{2.78}$$

where

$$\lim_{\tau\to\infty} P(x,\tau) = \Phi_0(x)\Psi_T(x), \tag{2.79}$$

which, if interpreted as a density, should always be greater than zero. In the case of Fermions, this is not guaranteed, as the node structure, i.e. the roots, of the exact ground state and the trial wave function will generally be different.

To avoid this anomaly in the density, $\Phi(x,\tau)$ and $\Psi_T(x)$ have to change sign together[9]. The brute force way of solving this problem is to *fix* the nodes by rejecting a walker's step if the trial wave function changes sign:

$$\frac{\Psi_T(x_i)}{\Psi_T(x_j)} < 0 \quad \Longrightarrow \quad A(i\ \rightarrow\ j) = 0 \tag{2.80}$$

where $A(i\ \rightarrow\ j)$ is the probability of accepting the move, as described in Section 2.4. An illustrative example is attempted in Fig. 2.12.

2.10 Estimating the One-body Density

The one-body density is defined as

[9] It should be mentioned that more sophisticated methods exist for dealing with the sign problem, some of which splits the distribution of walkers into a negative and a positive region.

$$\rho_\Phi(\mathbf{r_1}) = \iint\limits_{\mathbf{r_2}\mathbf{r_3}} ... \int\limits_{\mathbf{r_N}} |\mathbf{\Phi}(\mathbf{r_1}\mathbf{r_2}...\mathbf{r_N})|^\mathbf{2}\, \mathrm{d}\mathbf{r_2}...\mathrm{d}\mathbf{r_N}. \tag{2.81}$$

Due to the indistinguishable nature of the particles, the integral above corresponds to projecting every degree of freedom into a single configuration. In a Monte-Carlo simulation, estimating this quantity is done by collecting snapshots of the position data. These snapshots serve as samples to a histogram where each set of Cartesian coordinates (independent of particle number) give rise to one count on the grid.

2.10.1 Estimating the Exact Ground State Density

The one-body density of the trial wave functions is in other words trivial; create a histogram of sampled VMC positions. The challenge is estimating the one-body density of the exact wave function, given a set of DMC data. As described in Section 2.2.2, the distribution of walkers are designed to span $f(\mathbf{r},\tau) = \mathbf{\Phi}(\mathbf{r},\tau)\mathbf{\Psi_T}(\mathbf{r})$, which does not correspond to the ground state distribution unless the trial wave function is indeed the exact ground state.

In other words: We need a method to transform a given estimate from the *mixed estimate* (using $f(\mathbf{r},\tau)$) into the *pure estimate* using the ground state distribution $|\Phi_0(\mathbf{r})|^\mathbf{2}$. This is done by using a third estimate obtained from VMC, the *variational estimate*, which is the estimate sampled from $|\Psi_T(\mathbf{r})|^\mathbf{2}$. It is shown in ref. [2] that the relation is

$$\begin{aligned}\langle A\rangle_0 = \frac{\langle\Phi_0|\,\widehat{\mathbf{A}}\,|\Phi_0\rangle}{\langle\Phi_0|\Phi_0\rangle} &\simeq 2\frac{\langle\Phi_0|\,\widehat{\mathbf{A}}\,|\Psi_T\rangle}{\langle\Phi_0|\Psi_T\rangle} - \frac{\langle\Psi_T|\,\widehat{\mathbf{A}}\,|\Psi_T\rangle}{\langle\Psi_T|\Psi_T\rangle} + \mathcal{O}(\Delta^2) \\ &= 2\langle A\rangle_{\mathrm{DMC}} - \langle A\rangle_{\mathrm{VMC}} + \mathcal{O}(\Delta^2)\end{aligned} \tag{2.82}$$

where $\Delta \equiv \Phi_0(\mathbf{r}) - \mathbf{\Psi_T}(\mathbf{r})$. Expressed in terms of *density operators*, the expectation values becomes

$$\begin{aligned}\langle A\rangle_0 &= \mathrm{tr}(\widehat{\rho}_0\widehat{\mathbf{A}}) \\ \langle A\rangle_{\mathrm{VMC}} &= \mathrm{tr}(\widehat{\rho}_{\mathrm{VMC}}\widehat{\mathbf{A}}) \\ \langle A\rangle_{\mathrm{DMC}} &= \mathrm{tr}(\widehat{\rho}_{\mathrm{DMC}}\widehat{\mathbf{A}})\end{aligned}$$

where tr denotes the *trace* (sum of eigenvalues. Read more in ref. [14]). Inserting these equation into Eq. (2.82) yields

$$\begin{aligned}\mathrm{tr}(\widehat{\rho}_0\widehat{\mathbf{A}}) &\simeq 2\mathrm{tr}(\widehat{\rho}_{\mathrm{DMC}}\widehat{\mathbf{A}}) - \mathrm{tr}(\widehat{\rho}_{\mathrm{VMC}}\widehat{\mathbf{A}}) + \mathcal{O}(\Delta^2) \\ &\simeq \mathrm{tr}\left((2\widehat{\rho}_{\mathrm{DMC}} - \widehat{\rho}_{\mathrm{VMC}})\,\widehat{\mathbf{A}}\right) + \mathcal{O}(\Delta^2)\end{aligned} \tag{2.83}$$

which leads to the conclusion that the mixed density can be transformed in the following manner

$$\widehat{\rho}_0 \simeq 2\widehat{\rho}_{\mathrm{DMC}} - \widehat{\rho}_{\mathrm{VMC}} \tag{2.84}$$

In other words, combining the one-body densities from DMC and VMC according to Eq. (2.84) will give a good view of the exact one-body density.

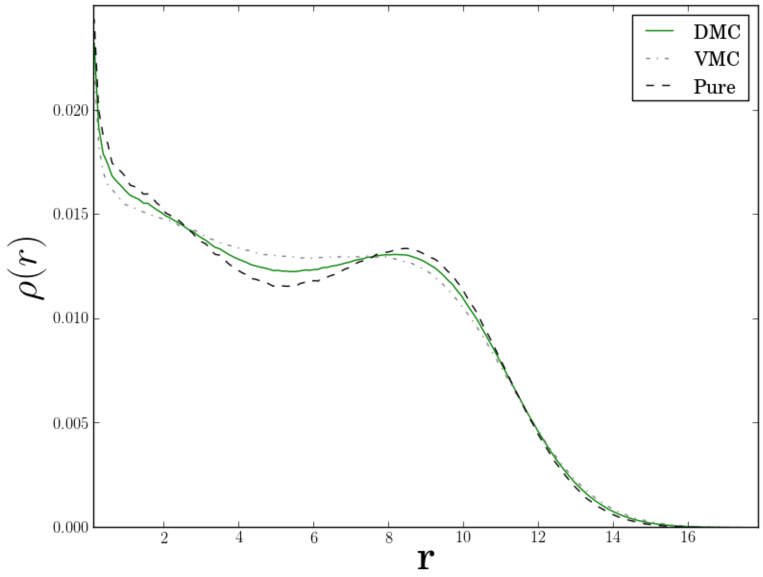

Figure 2.13: An example radial one-body density of a 12-particle Quantum Dot with $\omega = 0.1$. The one-body density is flat in for VMC in comparison to DMC. The pure density calculated from Eq. (2.84) transforms the mixed density $f(\mathbf{r},\tau) = \mathbf{\Phi}(\mathbf{r},\tau)\mathbf{\Psi_T}(\mathbf{r})$ from DMC into $|\Phi(\mathbf{r},\tau)|^\mathbf{2}$ with very small errors. The density diverges close to zero due to a "$\frac{0}{0}$" expression (see Eq. (2.87)).

2.10.2 Radial Densities

In the case of symmetric wave functions, like in the case for Helium and Quantum Dots, the radial one-body density is obtained by integrating out the angular dependencies of the one-body density. This corresponds to weighing all angular directions equally in a brute for average and should only be applied to algular symmetric systems.

Calculating the integral in two and three dimensions yields

$$\begin{aligned} I_{3\mathrm{D}} &= \iint \rho(r_1,\theta_1,\phi_1) r_1^2 \sin\theta_1 \mathrm{d}\theta_1 \mathrm{d}\phi_1 \\ &= r_1^2 \rho(r_1) \qquad (2.85) \\ I_{2\mathrm{D}} &= \int \rho(r_1,\phi_1) r_1 \mathrm{d}\phi_1 \\ &= r_1 \rho(r_1) \qquad (2.86) \end{aligned}$$

In practice, this integral is calculated by creating a histogram $H(r)$ of sampled radii. Transforming this histogram into the radial one-body density $\rho(r_1)$ is according to Eq. (2.85) and Eq. (2.86) done in the following manner

$$\rho(r_1) = \frac{H(r_1)}{r_1^{(d-1)}} \qquad (2.87)$$

where d denotes the number of dimensions. An example radial one-body density is given in Fig. 2.13

2.11 Estimating the Statistical Error

As with any statistical result, appropriate statistical errors needs to be included for it to be taken seriously. Systematic errors, that is, errors introduced due to limitations in the model, will be discussed in each method's respective section. Statistical errors, that is, the deviation from the true ensemble average due to the fact that we can never fulfill the equality in Eq. (2.58), can be estimated using several methods, some of which are *naive* in the sense that they assume the dataset to be completely *uncorrelated*.

2.11.1 The Variance and Standard Deviates

Given a set of samplings, e.g. local energies, the variance is a measure of their spread from the true mean value

$$\begin{aligned}
\mathrm{Var}(E) &= \left\langle (E - \langle E \rangle)^2 \right\rangle \\
&= \left\langle E^2 \right\rangle - 2\underbrace{\langle E \langle E \rangle \rangle}_{\langle E \rangle \langle E \rangle} + \langle E \rangle^2 \\
&= \left\langle E^2 \right\rangle - \langle E \rangle^2 \qquad (2.88) \\
&\simeq \overline{E^2} - \overline{E}^2 \qquad (2.89)
\end{aligned}$$

In the case of having the exact wave function, i.e $|\Psi_T\rangle = |\Psi_0\rangle$, the variance becomes zero:

$$\begin{aligned}
\mathrm{Var}(E)_{\mathrm{Exact}} &= \langle \Psi_0 | \widehat{\mathbf{H}}^2 | \Psi_0 \rangle - \langle \Psi_0 | \widehat{\mathbf{H}} | \Psi_0 \rangle^2 \\
&= E_0^2 - (E_0)^2 \\
&= 0
\end{aligned}$$

The variance is in other words an excellent measure of how good a fit different trial wave functions are to the system. Note however, a common misconception is to use the numerical value of the variance to compare *different* systems; for instance, if system A has variance equal to half of system B's, one could easily conclude that system A has the best fit. This is not true. The variance has units (in the case of local energies) energy squared, and will thus scale with the magnitude of the energy. One can only safely use the variance as a direct measure locally in each specific system, e.g. Beryllium simulations.

Another misconception is that the variance is a direct numerical measure of the error. This can in no way be true given that the units mismatch. The *standard deviation*, σ, is the square root of the variance,

$$\sigma^2(x) = \mathrm{Var}(x), \qquad (2.90)$$

and has hence a unit equal to that of the measured value. It is therefore related to the *spread* in the sampled value; zero deviation implies perfect samples, while increasing deviation means increasing spread and statistical uncertainty. The standard deviation is in other words a useful quantity when it comes to calculating the error, i.e. the expected deviation from the exact mean $\langle E \rangle$.

2.11.2 The Covariance and correlated samples

It was briefly mentioned in the introduction that certain error estimation techniques was too naive in case of correlated samples. Two samples, x, y, are said to be correlated if their *covariance*, $\mathrm{Cov}(x,y)$, is non-zero

$$\begin{aligned}\mathrm{Cov}(x,y) &\equiv \langle (x-\langle x\rangle)(y-\langle y\rangle)\rangle \\ &= \langle xy - x\langle y\rangle - \langle x\rangle y + \langle x\rangle\langle y\rangle\rangle \\ &= \langle xy\rangle - \langle x\langle y\rangle\rangle \underbrace{-\langle y\langle x\rangle\rangle + \langle\langle x\rangle\langle y\rangle\rangle}_{0} \\ &= \langle xy\rangle - \langle x\rangle\langle y\rangle\,. \end{aligned} \tag{2.91}$$

Notice that $\mathrm{Cov}(x,x) = \mathrm{Var}(x)$. Using this definition, whether of not we have correlated samples boils down to whether or not $\langle xy\rangle = \langle x\rangle\langle y\rangle$.

The consequence of ignoring correlations is an error estimate which is generally less than the true error; correlated samplings are more clustered, i.e. less spread, due to previous samplings' influence on the value of the current sample. Denoting the true standard deviation as σ_c, the above discussion can be distilled to

$$\sigma_c(x) \geq \sigma(x), \tag{2.92}$$

where σ is the deviation from Eq. (2.90).

2.11.3 The Deviate from the Exact Mean

There is an important difference between the deviate from the exact mean, and the deviate of a single sample from it's combined mean, i.e.

$$\sigma(\overline{x}) \neq \sigma(x). \tag{2.93}$$

Imagine doing a number of simulations, each resulting in a unique $\overline{x}$, the quantity of interest is not the deviation within a single simulation, but the deviation between all the simulations.

$$m \equiv \overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.94}$$

$$\sigma^2(m) = \left\langle m^2\right\rangle - \langle m\rangle^2 \tag{2.95}$$

Combining the above equations yields

$$
\begin{aligned}
\sigma^2(m) &= \left\langle \frac{1}{n^2}\left[\sum_{i=1}^{n} x_i\right]^2 \right\rangle - \left\langle \frac{1}{n}\sum_{i=1}^{n} x_i \right\rangle^2 \\
&= \frac{1}{n^2}\left(\left\langle \sum_{i=1}^{n} x_i \sum_{j=1}^{n} x_j \right\rangle - \left\langle \sum_{i=1}^{n} x_i \right\rangle \left\langle \sum_{j=1}^{n} x_j \right\rangle\right) \\
&= \frac{1}{n^2}\sum_{i,j=1}^{n} \langle x_i x_j\rangle - \langle x_i\rangle\langle x_j\rangle \\
&= \frac{1}{n^2}\sum_{i,j=1}^{n} \mathrm{Cov}(x_i, x_j) \qquad (2.96)
\end{aligned}
$$

This result is important; the true error is given in terms of the covariance, and is, as discussed previously, only equal to the sample variance if our samples are uncorrelated. Going back to the definition of covariance in Eq. (2.91), we see that in order to calculate the covariance as in Eq. (2.96), we need to know the true mean $\langle x_i\rangle$. Using $m = \overline{x}$ as an approximation to the true mean yields

$$
\begin{aligned}
\mathrm{Cov}(x_i, x_j) &\equiv \langle (x_i - \langle x_i\rangle)(x_j - \langle x_j\rangle)\rangle \\
&\simeq \langle (x_i - m)(x_j - m)\rangle \\
&\simeq \frac{1}{n^2}\sum_{k,l=1}^{n} (x_k - m)(x_l - m) \qquad (2.97) \\
&\equiv \frac{1}{n}\mathrm{Cov}(x) \qquad (2.98)
\end{aligned}
$$

Inserting this relation into Eq. (2.96) yields

$$
\boxed{
\begin{aligned}
\sigma^2(m) &= \frac{1}{n^2}\sum_{i,j=1}^{n} \mathrm{Cov}(x_i, x_j) \\
&\simeq \frac{1}{n^2}\sum_{i,j=1}^{n} \frac{1}{n}\mathrm{Cov}(x) \\
&= \frac{1}{n^3}\mathrm{Cov}(x)\underbrace{\sum_{i,j=1}^{n}}_{n^2} \\
&= \frac{1}{n}\mathrm{Cov}(x), \qquad (2.99)
\end{aligned}
}
$$

which serves as an estimate of the full error including correlations.

Explicitly computing the covariance is rarely done in Monte-Carlo simulations; if the sample size is large, it is extremely expensive. A variety of alternative methods to counter the correlations are available, the simplest of which is to define a *correlation length*, τ[10], which defines an interval at which points from the sampling sets are used for actual averaging. In other words, only the points $x_0, x_\tau, ..., x_{n\tau}$ are used in the calculation of $\overline{x}$

[10] This parameter is often referred to as the *auto-correlation time* in the literature.

$$\overline{x} = \frac{1}{n}\sum_{k=0}^{n} x_{k\cdot\tau} \tag{2.100}$$

This basically means that we need $n\tau$ samples in order to get the same magnitude of samples to our average as in Eq. (2.94); the *effective sample size* becomes $n_{\text{eff}} = n_{\text{tot}}/\tau$. $\tau = 1$ defines the uncorrelated case. For details regarding the derivations of τ based on the covariance, see ref. [15] and ref. [6].

2.11.4 Blocking

Introducing correlation lengths in the system solver is not an efficient option. Neither is calculating the covariance of billions of data points. However, the error is not a value vital to the simulation process, i.e. you do not need to know the error at any stage during the sampling. This means that we can post process the error calculation (given that we store the data sets).

An efficient algorithm for calculating the error of correlated data is *blocking*. This method is described in high detail in ref. [15], however, details aside, the idea itself is quite intuitive: Given a data set of N samples from a single Monte-Carlo simulation, imagine dividing the dataset into *blocks* of n samples, that is, blocks of size $n_b = N/n$. The error σ_n in each block will increase as n decrease, (see Eq. (2.99))

$$\sigma_n \propto \frac{1}{\sqrt{n}} \tag{2.101}$$

However, treating each block as an individual simulation, we have n_b averages m_n that we can use to calculate the total error in Eq. (2.95), that is, estimate the covariance

$$\overline{m_n^r} \equiv \frac{1}{n_b}\sum_{k=1}^{n_b} m_n^r \tag{2.102}$$

$$\begin{aligned}\sigma^2(m) &= \left\langle m^2\right\rangle - \langle m\rangle^2 \\ &\simeq \overline{m_n^2} - (\overline{m_n})^2\end{aligned} \tag{2.103}$$

The approximation should hold for a range of different block sizes, however, just as there is no a priori way of telling the correlation length, there is no a priori way of telling how many blocks is needed. What we do know however, is that if the system is correlated, there should be a range of different block sizes which fulfill Eq. (2.103) to reasonable precision. The result of a blocking analysis is therefore a series of (n, $\sigma(m)$) pairs. Plotting these pairs should in light of previous arguments result in a increasing curve which stabilizes over a certain span of block sizes. This plateau will then serve as a reasonable approximation to the covariance. See Fig. 2.14 for a demonstration of the blocking technique.

2.11.5 Variance Estimators

The standard intuitive variance estimator given by

$$\sigma^2(x) \simeq \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^2\right) - \overline{x}^2 \tag{2.104}$$

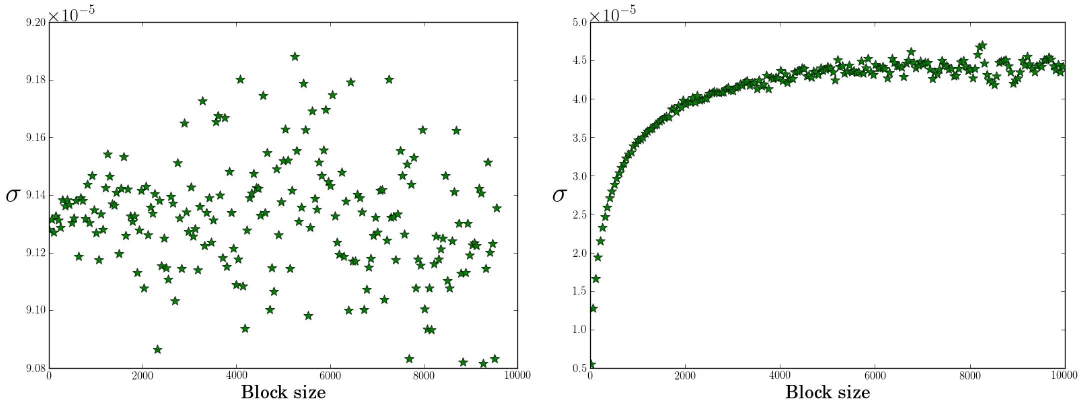

Figure 2.14: Left hand side: Blocking result of (approximately) uncorrelated data generated from a uniform Monte-Carlo integration of $\int_1^2 2x\mathrm{d}x$ resulting in 3.00003 (exact is 3.0). This is in excellent agreement with the magnitude of the error $\sim 9 \cdot 10^{-5}$. On the right hand side we have the blocking result (DMC) for 6-particle $\omega = 0.1$ Quantum Dot, used in Table **??**. We clearly see the plateau discussed in the blocking section, resulting in a total error in the range of $4.5 \cdot 10^{-5}$ to $5 \cdot 10^{-5}$.

is just an example of a variance estimator. A more precise estimator is

$$\sigma^2(x) \simeq \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 = \left(\frac{1}{n-1}\sum_{i=1}^{n} x_i^2\right) - \frac{n}{n-1}\overline{x}^2 \tag{2.105}$$

which is only noticeably different from Eq. (2.104) when the sample size gets small, as in blocking analysis. It is therefore standard to use Eq. (2.105) for blocking errors.

Part II

Results

Appendices

Bibliography

[1] D. Griffiths, *Introduction to Quantum Mechanics*, 2nd ed. Pearson, 2005.

[2] B. Hammond, J. W. A. Lester, and P. J. Reynolds, *Monte Carlo Methods in Ab Initio Quantum Chemistry*, S. Lin, Ed. World Scientific Publishing Co., 1994.

[3] C. W. Gardiner, *Handbook of stochastic methods for physics, chemistry, and the natural sciences*, 3rd ed. Berlin: Springer-Verlag, 2004. [Online]. Available: http://www.loc.gov/catdir/enhancements/fy0818/2004043676-d.html

[4] H. Risken and H. Haken, *The Fokker-Planck Equation: Methods of Solution and Applications Second Edition.* Springer, 1989.

[5] W. T. Coffey, Y. P. Kalmykov, and J. T. Waldron, *The Langevin Equation: With Applications to Stochastic Problems in Physics, Chemistry, and Electrical Engineering.* . World Scientific, Singapore, 2004.

[6] M. Hjorth-Jensen, "Computational Physics," 2010.

[7] J. Høgberget, *Git Repository: LibBorealis*, 2013. [Online]. Available: http://www.github.com/jorgehog/QMC2

[8] J. J. Sakurai, *Modern Quantum Mechanics*, Revised ed ed. New York: Addison-Wesley, 1994.

[9] I. Shavitt and R. J. Bartlett, *Many-Body Methods in Chemistry and Physics.* Cambridge: Cambridge University Press, 2009.

[10] D. A. Nissenbaum, "The stochastic gradient approximation: an application to li nanoclusters," Ph.D. dissertation, Northeastern University, 2008. [Online]. Available: http://hdl.handle.net/2047/d10016466

[11] G. Golub and C. Van Loan, *Matrix computations.* Johns Hopkins Univ Press, 1996, vol. 3.

[12] A. Harju, B. Barbiellini, S. Siljamäki, R. M. Nieminen, and G. Ortiz, "Stochastic Gradient Approximation: An Efficient Method to Optimize Many-Body Wave Functions," *Physical Review Letters*, vol. 79, pp. 1173–1177, Aug. 1997.

[13] S. Klein, J. P. W. Pluim, M. Staring, and M. A. Viergever, "Adaptive Stochastic Gradient Descent Optimisation for Image Registration." *International Journal of Computer Vision*, vol. 81, no. 3, pp. 227–239, 2009. [Online]. Available: http://dx.doi.org/10.1007/s11263-008-0168-y

[14] J. M. Leinaas, “Non-Relativistic Quantum Mechanics,” lecture notes FYS4110.

[15] H. Flyvbjerg and H. G. Petersen, “Error estimates on averages of correlated data,” *The Journal of Chemical Physics*, vol. 91, no. 1, pp. 461–466, 1989. [Online]. Available: http://link.aip.org/link/?JCP/91/461/1