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

During my stay at junior high school, I had no personal interest in mathematics. When the final year was finished, I sought an education within the first love of my life: Music. However, sadly, my application was turned down, forcing me to fall back on general topics. This is where I had my first encounter with physics, which should turn out not to be the last.

When high school ended, I applied for an education within structural engineering at the Norwegian University of Science and Technology, however, again I was turned down. Second on the list was the physics, meteorology and astronomy program at the University of Oslo. Never in my life have I been more grateful for being turned down, as starting at the University of Oslo introduced me to the second love of my life: Programming.

During the third year of my Bachelor I had a hard time figuring out which courses I should pick. Quite randomly, I attended a course on computational physics lectured by Morten Hjorth-Jensen. It immediately hit me that he had a genuine interest in the well-being of his students. It did not take long before I decided to apply for a Master's degree within the field of computational physics.

To summarize, me being here today writing this thesis is a result of extremely random events. However, I could not have landed in a better place, and I am forever grateful for my time here at the University of Oslo. So grateful in fact, that I am continuing my stay as a PhD student within the field of *multi-scale physics*.

I would like to thank my fellow Master students Sarah Reimann, Sigve Bøe Skattum, and Karl Leikanger for much appreciated help and good discussions. Sigve helped me with the minimization algorithm, that is, the darkest chapter of this thesis. A huge thanks to Karl for all those late nights where you gave me a ride home. Finally, thank you Sarah for withstanding my horrible German phrases all these years.

My supervisor, Morten Hjorth-Jensen, deserves a special thanks for believing in me, and filling me with confidence when it was much needed. This man is awesome.

Additionally I would like to thank the rest of the people at the computational physics research group, especially Mathilde N. Kamperud and Veronica K. B. Olsen whom with I shared my office, Svenn-Arne Dragly for helping me with numerous Linux-related issues, and "brochan" Milad H. Mobarhan for all the pleasant hours spent at Nydalen Kebab. You all contribute to making this place the best it can possibly be. Thank you!

<div align="right">Jørgen Høgberget</div>

Oslo, June 2013

Part I

Theory

1

Quantum Monte-Carlo

Quantum Monte-Carlo (QMC) is a method for solving Schrödinger's equation using statistical *Markov Chain* (random walk) 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 strategies which can be used in order 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 modelling 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 completeness, this approach will be mentioned only briefly in Section 1.2.3.

In this chapter, *Dirac Notation* will be used. See Appendix **??** for an introduction. The equations will be in atomic units, i.e. $\hbar = m_e = e = 4\pi\epsilon_0 = 1$, where m_e and ϵ_0 are the electron mass and the vacuum permittivity respectively.

1.1 Modelling Diffusion

Like any phenomena involving a probability distribution, Quantum Mechanics can be modelled 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 in the system at hand.

The basic idea is to introduce an ensemble of *random walkers*, in which each walker is represented by a position in space at a given 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 $d\mathbf{r}$ will correspond to $|\Phi(\mathbf{r}, t)|^2 d\mathbf{r}$ 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 QMC is described.

---

[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.

The purpose of this section is to motivate the use of diffusion theory in Quantum Mechanics, and to derive the sampling rules needed in order to model Quantum Mechanical distributions by diffusion of random walkers correctly.

1.1.1   Stating the Schrödinger Equation as a Diffusion Problem

Consider the time-dependent Schrödinger equation for an arbitrary wave function $\Phi(\mathbf{r}, t)$ using an arbitrary energy shift E'

$$-\frac{\partial \Phi(\mathbf{r}, t)}{i\partial t} = (\widehat{\mathbf{H}} - E')\Phi(\mathbf{r}, t). \tag{1.1}$$

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

$$\widehat{\mathbf{H}}\Phi(\mathbf{r}, t_0) = E\Phi(\mathbf{r}, t_0), \tag{1.2}$$

$$\Phi(\mathbf{r}, t) = \exp\left(-i(\widehat{\mathbf{H}} - E')(t - t_0)\right)\Phi(\mathbf{r}, t_0). \tag{1.3}$$

From Eq. (1.3) it is apparent that the time evolution operator is on the form

$$\widehat{\mathbf{U}}(t, t_0) = \exp\left(-i(\widehat{\mathbf{H}} - E')(t - t_0)\right). \tag{1.4}$$

The time-independent equation is solved for the ground state energy through methods such as *Full Configuration Interaction* [2] or similar methods based on diagonalizing the Hamiltonian. The time-dependent equation is used by methods such as *Time-Dependent Multi-Configuration Hartree-Fock* [3] in order to obtain the time-development of quantum states. However, neither of the equations originate from, or resemble, diffusion equations.

The original Schrödinger equation, however, does resemble a diffusion equation in complex time[2]. 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.

Substituting complex time with a parameter τ and choosing the energy shift E' equal to the true ground state energy of $\widehat{\mathbf{H}}$, E_0, the time evolution operator in Eq. (1.4) becomes the *projection operation* $\widehat{\mathbf{P}}(\tau)$, whose choice of name will soon be apparent. In other words:

$$t \to it \equiv \tau,$$

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

Consider an arbitrary wave function $\Psi_T(\mathbf{r})$. Applying the new operator yields a new wave function $\Phi(\mathbf{r}, \tau)$ in the following manner

---

[2]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 [4].

$$\Phi(\mathbf{r}, \tau) = \langle \mathbf{r} | \widehat{\mathbf{P}}(\tau) | \Psi_T \rangle \tag{1.5}$$
$$= \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\tau\right) | \Psi_T \rangle.$$

Expanding the arbitrary state in the eigenfunction of $\widehat{\mathbf{H}}$, $|\Psi_i\rangle$, yields

$$\Phi(\mathbf{r}, \tau) = \sum_i C_i \langle r | \exp\left(-(\widehat{\mathbf{H}} - E_0)\tau\right) | \Psi_i \rangle$$
$$= \sum_i C_i \Psi_i(\mathbf{r}) \exp\left(-(E_i - E_0)\tau\right)$$
$$= C_0 \Psi_0(x) + \sum_{i=1}^{\infty} C_i \Psi_k(x) e^{-\delta E_i \tau}, \tag{1.6}$$

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. In other words:

$$\lim_{\tau \to \infty} \Phi(\mathbf{r}, \tau) = \lim_{\tau \to \infty} \langle \mathbf{r} | \widehat{\mathbf{P}}(\tau) | \Psi_T \rangle$$
$$= C_0 \Psi_0(x). \tag{1.7}$$

The projection operator transforms an arbitrary wave function $\Psi_T(\mathbf{r})$, from here on referred to as the *trial wave function*, 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

$$\widehat{\mathbf{P}}(\tau) = \prod_{k=1}^{n} \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right), \tag{1.8}$$

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

$$\widehat{\mathbf{P}}(\tau + \delta\tau) = \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) \widehat{\mathbf{P}}(\tau). \tag{1.9}$$

Using this relation in combination with Eq. (1.5), the effect of the projection operator during a single time-step is revealed:

$$\Phi(\mathbf{r}, \tau + \delta\tau) = \langle \mathbf{r} | \widehat{\mathbf{P}}(\tau + \delta\tau) | \Psi_T \rangle$$
$$= \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) \widehat{\mathbf{P}}(\tau) | \Psi_T \rangle$$
$$= \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) | \Phi(\tau) \rangle$$
$$= \int_{\mathbf{r}'} \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) | \mathbf{r}' \rangle \langle \mathbf{r}' | \Phi(\tau) \rangle \, d\mathbf{r}'$$
$$= \int_{\mathbf{r}'} \langle \mathbf{r} | \exp\left(-(\widehat{\mathbf{H}} - E_0)\delta\tau\right) | \mathbf{r}' \rangle \Phi(\mathbf{r}', \tau) d\mathbf{r}', \tag{1.10}$$

where a complete set of position states were introduced.

For practical purposes, E_0 needs to be substituted with an approximation E_T to the ground state energy, commonly referred to as the *trial energy*, in order to avoid self consistency. From Eq. (1.6) it is apparent that the projection will still converge as long as $E_T < E_1$, that is, the trial energy is less than that of the first excitation. The resulting expression reads:

$$\Phi(\mathbf{r}, \tau + \delta\tau) = \int_{\mathbf{r}'} \langle\mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_T)\delta\tau\right) |\mathbf{r}'\rangle \, \Phi(\mathbf{r}', \tau)\mathrm{d}\mathbf{r}' \tag{1.11}$$

$$\equiv \int_{\mathbf{r}'} G(\mathbf{r}, \mathbf{r}'; \delta\tau)\Phi(\mathbf{r}', \tau)\mathrm{d}\mathbf{r}'. \tag{1.12}$$

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. (1.12) 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 energy operator $\widehat{\mathbf{V}}$ and the energy shift. This is known as the *short time approximation* [5]

$$G(\mathbf{r}, \mathbf{r}'; \delta\tau) = \langle\mathbf{r}| \exp\left(-(\widehat{\mathbf{H}} - E_T)\delta\tau\right) |\mathbf{r}'\rangle \tag{1.13}$$

$$= \langle\mathbf{r}| e^{-\widehat{\mathbf{T}}\delta\tau} e^{-(\widehat{\mathbf{V}} - E_T)\delta\tau} |\mathbf{r}'\rangle + \frac{1}{2}[\widehat{\mathbf{V}}, \widehat{\mathbf{T}}]\delta\tau^2 + \mathcal{O}(\delta\tau^3). \tag{1.14}$$

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 responsible for distributing the correct weights to the corresponding walkers. In other words:

$$G_{\mathrm{Diff}} = e^{\frac{1}{2}\nabla^2\delta\tau}, \tag{1.15}$$

$$G_{\mathrm{B}} = e^{-(\widehat{\mathbf{V}} - E_T)\delta\tau}, \tag{1.16}$$

where B denotes *branching*. The reasons for this name together with the complete process of modelling weights by branching will be covered in detail in Section 1.5.

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 distribution 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. (1.15) results in a method called *Variational Monte-Carlo* (VMC). Including the branching term as well results in *Diffusion Monte-Carlo* (DMC). These methods will be discussed in Sections 1.8 and 1.9 respectively. In either of these methods, diffusion is a key process.

1.2   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 1.2.3.

1.2.1 Isotropic Diffusion

Isotropic diffusion is a process in which diffusing particles sees all directions as an equally probable path. Eq. (1.17) is an example of this. The isotropic diffusion equation is

$$\frac{\partial P(\mathbf{r}, t)}{\partial t} = D \nabla^2 P(\mathbf{r}, t). \tag{1.17}$$

This is the simplest form of a diffusion equation, that is, the case with a linear *diffusion constant*, D, and no drift terms.

From Eq. (1.15) 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. An important 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 $2D\delta t$ [5]

$$G_{\text{Diff}}^{\text{ISO}}(i \rightarrow j) \propto e^{-|\mathbf{r}_i - \mathbf{r}_j|^2 / 4D\delta\tau}. \tag{1.18}$$

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 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:

$$x_{i+1} = x_i + \xi, \qquad \begin{aligned} \text{Var}(\xi) &= 2D\delta t, \\ \langle \xi \rangle &= x_i, \end{aligned} \tag{1.19}$$

where ξ is a normal distributed number whose variance matches that of the Green's function in Eq. (1.18). This relation is in agreement with the isotropy of Eq. (1.17) in the sense that the displacement is symmetric around the current position.

1.2.2 Anisotropic Diffusion and the Fokker-Planck equation

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. The Fokker-Planck equation reads:

$$\frac{\partial P(\mathbf{r},t)}{\partial t} = D\nabla \cdot \left[\left(\nabla - \mathbf{F}(\mathbf{r},t)\right)P(\mathbf{r},t)\right]. \tag{1.20}$$

As will be derived in detail in Section 1.2.3, 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.

As mentioned introductory, the goal of the Markov process 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. (1.20) is zero. This yields:

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

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

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

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 becomes

$$\begin{aligned}\mathbf{F}(\mathbf{r},t) &= \frac{1}{P(\mathbf{r},t)}\nabla P(\mathbf{r},t) \\ &= \frac{2}{|\psi(\mathbf{r},t)|}\nabla|\psi(\mathbf{r},t)|.\end{aligned} \tag{1.21}$$

In QMC, the drift term is commonly referred to as the *quantum force*. This is 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, its Green's function still has a closed form solution. This means that it can be evaluated efficiently. If this was not the case, the practical value would be reduced dramatically. The reason for this will become clear in Section 1.4. The closed form solution reads [5]

$$G_{\text{Diff}}^{\text{FP}}(i \to j) \propto e^{-(x_i-x_j-D\delta\tau F(x_i))^2/4D\delta\tau}. \tag{1.22}$$

As expected, the Green's function is no longer symmetric.

The Langevin Equation for the Fokker-Planck equation

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

$$\frac{\partial x_i}{\partial t} = DF(\mathbf{r})_i + \eta, \tag{1.23}$$

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

$$x_{i+1} = x_i + \xi + DF(\mathbf{r})_i \delta t, \tag{1.24}$$

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 Refs. [6–8].

1.2.3   Connecting 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

$$
\begin{aligned}
-\frac{\partial}{\partial t} f(\mathbf{r}, t) \;=\; \Psi_T(\mathbf{r})\left[-\frac{\partial}{\partial t}\Phi(\mathbf{r}, t)\right] \;&=\; \Psi_T(\mathbf{r})\left(\widehat{\mathbf{H}} - E_T\right)\Phi(\mathbf{r}, t) \\
&=\; \Psi_T(\mathbf{r})\left(\widehat{\mathbf{H}} - E_T\right)\Psi_T(\mathbf{r})^{-1} f(\mathbf{r}, t) \\
&=\; -\frac{1}{2}\Psi_T(\mathbf{r})\nabla^2\left(\Psi_T(\mathbf{r})^{-1} f(\mathbf{r}, t)\right) + \widehat{\mathbf{V}} f(\mathbf{r}, t) - E_T f(\mathbf{r}, t).
\end{aligned}
\tag{1.25}
$$

Expanding the Laplacian term further reveals

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

$$\tag{1.26}$$
$$\tag{1.27}$$

Combining these equations and applying the product rule numerous times yield

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

Introducing the following identity helps clean up the messy calculations:

$$-\left|\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right|^2 \quad = \quad \nabla\cdot\left(\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})\right) - \Psi_T(\mathbf{r})^{-1}\nabla^2\Psi_T(\mathbf{r}),$$

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

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

Inserting the expression for the quantum force $\mathbf{F}(\mathbf{r}) = 2\Psi_T(\mathbf{r})^{-1}\nabla\Psi_T(\mathbf{r})$ 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

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

Inserting everything back into Eq. (1.25) yields

$$
\begin{aligned}
-\frac{\partial}{\partial t}f(\mathbf{r},t) \quad = \quad & -\frac{1}{2}\nabla\cdot\left[(\nabla-\mathbf{F}(\mathbf{r}))f(\mathbf{r},t)\right] + K_L(\mathbf{r})f(\mathbf{r},t) + \widehat{\mathbf{V}}f(\mathbf{r},t) - E_T f(\mathbf{r},t) \\
\frac{\partial}{\partial t}f(\mathbf{r},t) \quad = \quad & \frac{1}{2}\nabla\cdot\left[(\nabla-\mathbf{F}(\mathbf{r}))f(\mathbf{r},t)\right] - (E_L(\mathbf{r})-E_T)f(\mathbf{r},t), \quad\quad\quad (1.28)
\end{aligned}
$$

which is the Fokker-Planck diffusion equation from Eq. (1.20) with a constant shift representing the branching Green's function in the case Fokker-Planck diffusion.

Just as in traditional importance sampled Monte-Carlo integrals, optimized sampling is obtained in QMC by switching distributions into one which exploits known information about the problem at hand. In the 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 QMC, 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* QMC.

The energy estimated using the new distribution $f(\mathbf{r},t)$ will still equal the exact energy in the limit of convergence. This is demonstrated in the following equations:

$$E_{\text{QMC}} = \frac{1}{N} \int f(\mathbf{r}, \tau) \frac{1}{\Psi_T(\mathbf{r})} \widehat{\mathbf{H}} \Psi_T(\mathbf{r}) d\mathbf{r}$$

$$= \frac{1}{N} \int \Phi(\mathbf{r}, \tau) \widehat{\mathbf{H}} \Psi_T(\mathbf{r}) d\mathbf{r}$$

$$= \frac{1}{N} \langle \Phi(\tau) | \widehat{\mathbf{H}} | \Psi_T \rangle,$$

where

$$N = \int f(\mathbf{r}, \tau) d\mathbf{r}$$

$$= \int \Phi(\mathbf{r}, \tau) \Psi_T(\mathbf{r}) d\mathbf{r}$$

$$= \langle \Phi(\tau) | \Psi_T \rangle,$$

which results in the following expression for the energy:

$$E_{\text{QMC}} = \frac{\langle \Phi(\tau) | \widehat{\mathbf{H}} | \Psi_T \rangle}{\langle \Phi(\tau) | \Psi_T \rangle}.$$

Assuming that the walkers have converged to the exact ground state, i.e. $|\Phi(\tau)\rangle = |\Phi_0\rangle$, letting the Hamiltonian work to the left yields

$$E_{\text{QMC}} = E_0 \frac{\langle \Phi_0 | \Psi_T \rangle}{\langle \Phi_0 | \Psi_T \rangle}$$

$$= E_0.$$

Estimating the energy in QMC will be discussed in detail in Sections 1.6.4 and 1.9.

1.3   Diffusive Equilibrium Constraints

Upon convergence of a Markov process, the ensemble of walkers will on average span the system's most likely state. This is exactly the behavior of a 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, or it may not; either way it is not optimal. Introducing rules of acceptance and rejection on top of the suggested transitions given by the Langevin equation in Eq. (1.19 or Eq. (1.24) will result in an optimized sampling. Special care must be taken not to violate 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.

1.3.1   Detailed Balance

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

$$P_i W(i \rightarrow j) = P_j W(j \rightarrow i), \tag{1.29}$$

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

1.3.2   Ergodicity

Another requirement is that the sampling must be *ergodic* [?], that is, the random walkers need 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.

1.4   The Metropolis Algorithm

The Metropolis Algorithm is a simple set of acceptance/rejection rules used in order to make the thermalization more efficient. 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 given in Eq. (1.29, and further introducing a model for the transition probability $W(i \rightarrow j)$ as consisting of two parts: The probability of selecting configuration j given configuration i, $g(i \rightarrow j)$, times a probability of accepting the selected move, $A(i \rightarrow j)$, yields

$$
\begin{aligned}
P_i W(i \rightarrow j) &= P_j W(j \rightarrow i), \\
P_i g(i \rightarrow j) A(i \rightarrow j) &= P_j g(j \rightarrow i) A(j \rightarrow i).
\end{aligned}
\tag{1.30}
$$

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

$$
\begin{aligned}
|\psi_i|^2 G(i \rightarrow j) A(i \rightarrow j) &= |\psi_j|^2 G(j \rightarrow i) A(j \rightarrow i), \\
\frac{A(j \rightarrow i)}{A(i \rightarrow j)} &= \frac{G(i \rightarrow j)}{G(j \rightarrow i)} \frac{|\psi_i|^2}{|\psi_j|^2} \equiv R_G(j \rightarrow i) R_\psi(j \rightarrow i)^2,
\end{aligned}
\tag{1.31}
$$

where the defined ratios correspond to the Green's function - and wave function ratio 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, $A(i \rightarrow j) = 1$. In other words, a more efficient 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. (1.31) for the case where $A(i \rightarrow j) = 1$, that is, the case where $P_i < P_j$. This yields

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

Combining both scenarios into one expression yield the following acceptance/rejection rules:

$$A(i \to j) = \begin{cases} R_G(i \to j)R_\psi(i \to j)^2 & R_G(i \to j)R_\psi(i \to j)^2 < 1 \\ 1 & \text{else} \end{cases}. \tag{1.32}$$

This equation can be simplified to

$$A(i \to j) = \min\{R_G(i \to j)R_\psi(i \to j)^2, 1\}. \tag{1.33}$$

In the isotropic diffusion case, the Green's function ratio cancels due to symmetry, i.e. $R_G(i \to j) = 1$, resulting in the standard Metropolis algorithm:

$$A(i \to j) = \min\{R_\psi(i \to j)^2, 1\}. \tag{1.34}$$

On the other hand, for Fokker-Planck diffusion, there will be no cancellation of the Green's functions. Inserting Eq. (1.22) into Eq. (1.32) results in the *Metropolis Hastings algorithm* [?]. The ratio of the Green's functions can be evaluated efficiently by simply subtracting the exponents of the exponentials. This is best demonstrated by calculating the logarithm

$$\begin{aligned} \log R_G^{\text{FP}}(i \to j) &= \log\left(G_{\text{Diff}}^{\text{FP}}(j \to i)/G_{\text{Diff}}^{\text{FP}}(i \to j)\right) \\ &= \frac{1}{2}\left(F(x_j) + F(x_i)\right)\left(\frac{1}{2}D\delta t(F(x_j) - F(x_i)) + x_i - x_i\right), \end{aligned} \tag{1.35}$$

$$A(i \to j) = \min\{\exp\left(\log R_G^{\text{FP}}(i \to j)\right) R_\psi(i \to j)^2, 1\}. \tag{1.36}$$

Derived from detailed balance, the Metroplis Algorithm is an essential part of any Markov Chain Monte-Carlo algorithm. Besides QMC, methods for solving problems such as the *Ising Model* greatly benefit from these rules [9].

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 the simulations is finite, and the trial positions of the walkers are random. A chart flow describing the implementation of the Metropolis algorithm and the diffusion process is given in Figure 1.1.
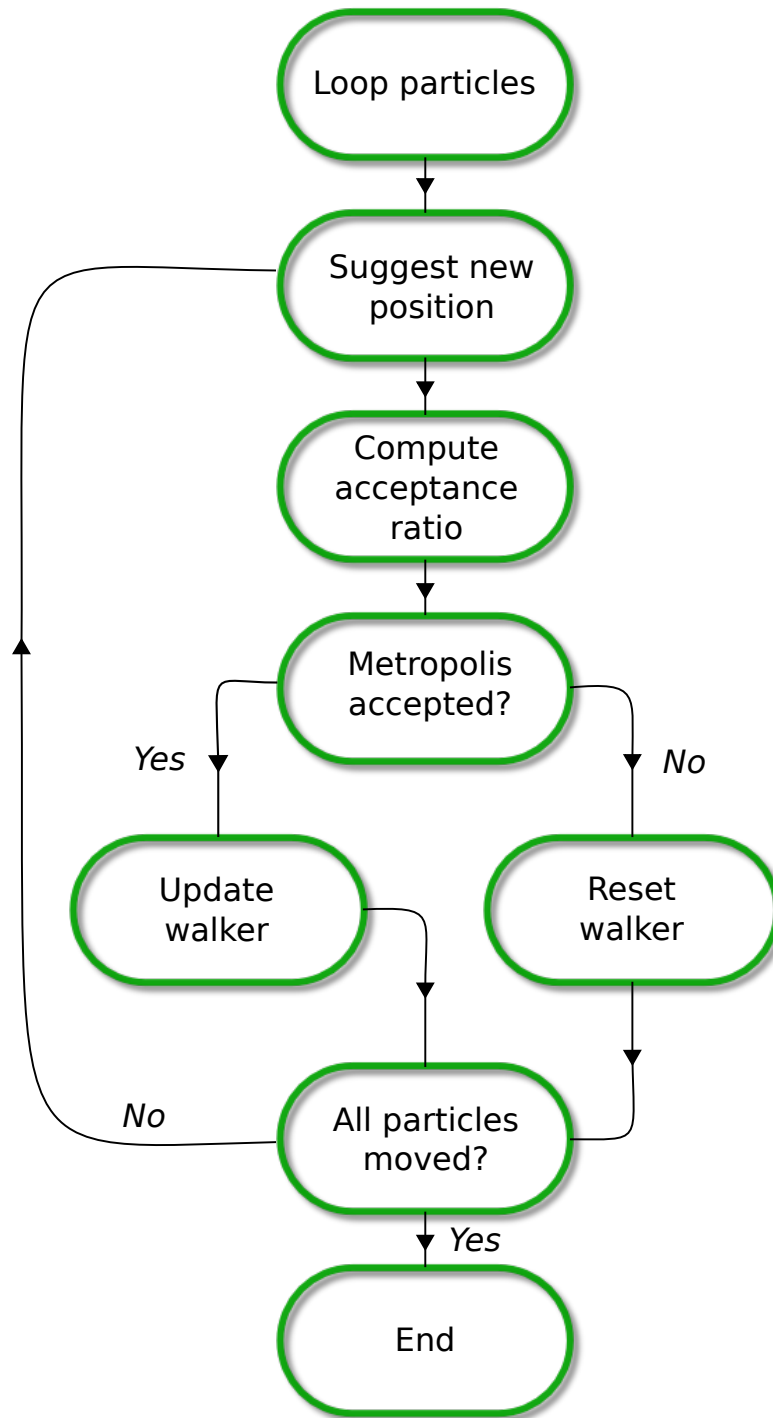
Figure 1.1: Flow chart describing the process of iterating a walker through a single time-step, that is, simulation the application of the Green's function from Eq. (1.15) using the Metropolis algorithm. New positions are suggested according to the chosen diffusion model.

1.5   The Process of Branching

In the previous section it became clear that the Metropolis test will guide the walkers to span a distribution representing the trial wave function. This implies that without further action, no changes to the distribution can be made, and the point of modelling the projection operator from Eq. (1.5) is rendered useless. The important fact to include is that the branching Green's function from Eq. (1.38) and Eq. (1.37) distribute weights to the walkers, effectively altering the spanned distribution.

The process of branching in QMC is simulated by the creation and destruction of walkers with probability equal to that of the branching Green's function [5]. The explicit shapes in case of isotropic - (ISO) and anisotropic diffusion (FP) are

$$
G_B^{\text{ISO}}(i \rightarrow j) \;=\; e^{-\left(\frac{1}{2}[V(x_i)+V(x_j)]-E_T\right)\delta\tau}, \tag{1.37}
$$

$$
G_B^{\text{FP}}(i \rightarrow j) \;=\; e^{-\left(\frac{1}{2}[E_L(x_i)+E_L(x_j)]-E_T\right)\delta\tau}, \tag{1.38}
$$

where $E_L(x_i)$ is the energy evaluated in configuration x_i (see Section 1.6.4 for details). The three different scenarios which arise is

- $G_B = 1$: No branching.

- $G_B = 0$: The current walker is to be removed from the current ensemble.

- $G_B > 1$: On average $G_B - 1$ replicas of the current walker are made.

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

$$
\overline{G}_B = \text{floor}\left(G_B + a\right), \tag{1.39}
$$

where a is a uniformly distributed number on $[0, 1)$. The probability that $\overline{G}_B = G_B + 1$ is then equal to $G_B - \text{floor}(G_B)$. 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.

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 code documentation at Ref. [10]. Isotropic diffusion is in practice never used with branching due to the singularities in the Coulomb interaction (see Eq. (1.37)). This singularity may cause large fluctuations in the walker population, which is far from an optimal behavior.

The process of branching is demonstrated in Figure 1.2.

$$W_i(R, \tau)$$

$$\tau + \delta\tau$$

$$\tau + 2\delta\tau$$

$$W_i(R, \tau + 2\delta\tau)$$

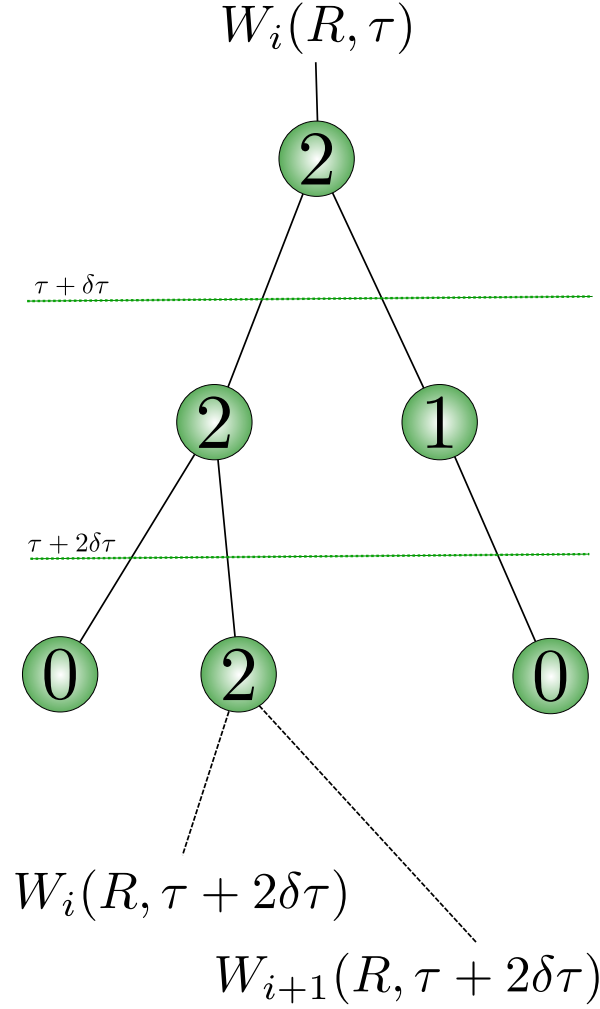$$W_{i+1}(R, \tau + 2\delta\tau)$$

Figure 1.2: The process of branching illustrated. The initial walker $W_i(R, \tau)$ is branched according to the rules of Section 1.5. The numerical value inside the nodes represents the value of $\overline{G}_B$ from Eq. (1.39). 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, no two walkers should ever be equal, given that not all of the steps was rejected by the Metropolis test.

1.6 The Trial Wave Function

The initial condition of the QMC calculations, that is, the trial wave function $\Psi_T(\mathbf{r})$, can in principle be chosen to be any normalizable wave function whose overlap with the exact ground state wave function, $\Psi_0(\mathbf{r})$, is non-zero.

If the overlap is zero, that is, if $C_0 = 0$ in Eq. (1.7), the 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 the exact ground state, $\Psi_0(\mathbf{r})$, will become the dominant contribution to the distribution.

In other words, the trial wave function should be chosen in such a way that the overlap is optimized, i.e. close to unity. Since the exact ground state is unknown, this overlap has to be optimized based on educated guesses and by forcing known properties of the exact ground state into the approximation. This will be the focus in this section.

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 Quantum Mechanical concepts and many-body theory, see for example Refs. [1, 2, 11].

1.6.1 Many-body Wave Functions

Many-body theory arise from the existence of *many-body interactions*, which in this thesis will be truncated at the level of the Coulomb interaction, that is, the two-body interaction. Nature operates using N-body interactions, however, it is overall safe to assume that the contributions beyond Coulomb decrease as the order of the interactions increase. If only one-body interactions were present, as is the case for non-interacting particles, the full system would decouple into N single particle systems, rendering many-body theory redundant.

Finding the ground state is, not surprisingly, equivalent to solving the time-independent Schrödinger Equation from Eq. (1.2) for the lowest energy eigenvalue, that is

$$\widehat{\mathbf{H}}\Psi_0(\mathbf{r}) = E_0\Psi_0(\mathbf{r}), \tag{1.40}$$

where $\mathbf{r} \equiv \{\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N\}$ represents the position of every particle. Exact solutions to realistic many-body systems rarely exist, however, like in Section 1.1.1, expanding the solution in a known basis $\Phi_k(\mathbf{r})$ is always legal, which reduces the problem into that of a *coefficient hunt*

$$\Psi_0(\mathbf{r}) = \sum_{k=0}^{\infty} C'_k \Phi_k(\mathbf{r}), \tag{1.41}$$

where the primed coefficients should not to be confused with the previous coefficients expanding an arbitrary state in the $\Psi_i(\mathbf{r})$ basis (see Eq. (1.6)). Different many-body methods give rise to different ways of estimating these coefficients, however, certain concepts are necessarily common, for instance truncating the basis at some level, K:

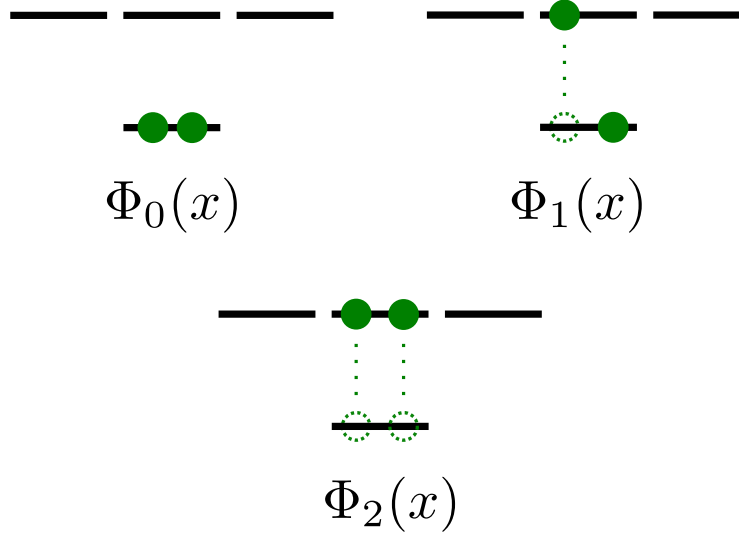$$\Psi_0(\mathbf{r}) = \sum_{k=0}^{K} \tilde{C}'_k \Phi_k(\mathbf{r}), \tag{1.42}$$

Figure 1.3: Three different electron configurations in an shell structure making up three different $\Phi_k(\mathbf{r})$, i.e. constituents of the many-body basis described in Eq. (1.42). An electron (solid dot) is represented by e.g. the orbital $\phi_{1s}(\mathbf{r}_1)$.

where $\tilde{C}'_k \neq C'_k$ unless K tends to infinity, however, it is overall safe to assume that the value of the coefficients decrease as k increase.

The many-body basis elements $\Phi_k(\mathbf{r})$ are constructed using N elements from a basis of single particle wave functions, or *orbitals* for short, where N denotes the total number of particles in the system. In other words, these orbitals, labeled $\phi_n(\mathbf{r}_i)$, combined in different ways give rise to the different many-body wave functions making up the total wave function. The process of calculating basis elements often boils down to an exercise in combinatorics involving combinations of orbitals.

To bring some clarity to the relation between the different wave functions, consider the following example: Imagine electrons surrounding a nucleus, i.e an atom. A single electron occupying a state with quantum number n at a position $\mathbf{r}_i$ is then described by the orbital $\phi_n(\mathbf{r}_i)$. Each unique[3] configuration of electrons (in terms of n) will give rise to one unique $\Phi_k(\mathbf{r})$. In other words, the complete basis of $\Phi_k(\mathbf{r})$ is described by the collection of all possible excited states and the ground state. $\Phi_0(\mathbf{r})$ is the ground state of the atom, $\Phi_1(\mathbf{r})$ has one electron exited to a higher shell, $\Phi_2(x)$ has another, and so on. See Figure 1.3 for a demonstration of this. The ordering of the terms in Eq. (1.42) are thus chosen to represent higher and higher excitations, i.e. the states has higher and higher energy eigenvalues.

To summarize, constructing an approximation to an unknown many-body ground state wave function involves three steps:

---

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

Step one	Choose a basis of orbitals $\phi_n(\mathbf{r}_i)$, e.g. hydrogen states.
Step two	Construct $\Phi_k(\mathbf{r})$ from $N \times \phi_n(\mathbf{r}_i)$.
Step three	Construct $\Psi_0(\mathbf{r})$ from $K \times \phi_k(\mathbf{r})$.

The last step is well described by Eq. (1.42), but is seldom necessary to perform explicitly; expressions involving the approximated ground state wave function is given in terms of the constituent $\Phi_k(\mathbf{r})$ elements and their coefficients.

Step one in detail

The Hamiltonian of an N-particle system is

$$\widehat{\mathbf{H}} = \widehat{\mathbf{H}}_0 + \widehat{\mathbf{H}}_{\mathrm{I}}, \tag{1.43}$$

where $\widehat{\mathbf{H}}_0$ and $\widehat{\mathbf{H}}_{\mathrm{I}}$ are respectively the one-body - and the many-body Hamiltonian. As mentioned in the introduction, the many-body interactions are truncated at the level of the two-body Coulomb interaction. The one-body term consist of the external potential $\widehat{\mathbf{u}}_{\mathrm{ext}}(\mathbf{r}_i)$ and the kinetic term $\widehat{\mathbf{t}}(\mathbf{r}_i)$ for all particles. In order words, the two Hamiltonians are written

$$
\begin{aligned}
\widehat{\mathbf{H}}_0 &= \sum_{i=1}^{N} \widehat{\mathbf{h}}_0(\mathbf{r}_i) \\
&= \sum_{i=1}^{N} \widehat{\mathbf{t}}(\mathbf{r}_i) + \widehat{\mathbf{u}}_{\mathrm{ext}}(\mathbf{r}_i),
\end{aligned}
\tag{1.44}
$$

and

$$
\begin{aligned}
\widehat{\mathbf{H}}_{\mathrm{I}} &\simeq \sum_{i<j=1}^{N} \widehat{\mathbf{v}}(r_{ij}) \\
&= \sum_{i<j=1}^{N} \frac{1}{r_{ij}},
\end{aligned}
\tag{1.45}
$$

where $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the distance between two particles.

In order to optimize the overlap C_0 with the exact wave function, the single particle orbitals are commonly chosen to be the eigenfunctions of the non-interacting single particle Hamiltonian, that is

$$\widehat{\mathbf{h}}_0(\mathbf{r}_i)\phi_n(\mathbf{r}_i) = \epsilon_n\phi_n(\mathbf{r}_i). \tag{1.46}$$

If no such choice can be made, choosing, free-particle solutions, Laguerre polynomials, or similar, is the general strategy. However, for these bases, the expansion truncation K from Eq. (1.42) needs to be higher in order to achieve a satisfying overlap.

Step two in detail

In the case of *fermions*, that is, half-integer spin particles like electrons, protons, etc., $\Phi_k(\mathbf{r})$ is an anti-symmetric function[4] on the form of a determinant: The so-called *Slater determinant*. The shape of the

---

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

determinant is given in Eq. (1.47). 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, which in many ways are easier to deal with because of the lack of an exclusion principle. The bosonic many-body wave function is given in Eq. (1.48). In order to keep the terminology less abstract and confusing, the focus will be on systems of fermions from here on.

$$
\begin{aligned}
\Phi_0^{\mathrm{AS}}(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N) \quad & \propto \quad \sum_{\mathrm{P}} (-)^{\mathrm{P}} \widehat{\mathbf{P}} \phi_1(\mathbf{r}_1)\phi_2(\mathbf{r}_2)\ldots\phi_N(\mathbf{r}_N) \\[2mm]
& = \quad \begin{vmatrix} \phi_1(\mathbf{r}_1) & \phi_2(\mathbf{r}_1) & \cdots & \phi_N(\mathbf{r}_1) \\ \phi_1(\mathbf{r}_2) & \phi_2(\mathbf{r}_2) & \cdots & \phi_N(\mathbf{r}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(\mathbf{r}_N) & \phi_2(\mathbf{r}_N) & \cdots & \phi_N(\mathbf{r}_N) \end{vmatrix}, \qquad (1.47) \\[4mm]
\Phi_0^{\mathrm{S}}(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N) \quad & \propto \quad \sum_{\mathrm{P}} \widehat{\mathbf{P}} \phi_1(\mathbf{r}_1)\phi_2(\mathbf{r}_2)\ldots\phi_N(\mathbf{r}_N). \qquad\qquad (1.48)
\end{aligned}
$$

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. Any combination of N orbital elements $\phi_n(\mathbf{r}_i)$ can be used to produce different $\Phi_k(\mathbf{r})$. For illustrative purposes, and for the purpose of this thesis in general where a single determinant ansatz is used, only the ground state has been presented.

Dealing with correlations

The contributions to the ground state on the right-hand side in Eq. (1.41) for $k > 0$ are referred to as *correlation* terms. Given that the single particle wave functions are chosen by Eq. (1.46), the existence of the correlation terms, i.e. $C_k' \neq 0$ for $k > 0$, follows as a direct consequence of the electron-electron interaction, hence the name.

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 the calculations are performed using the exact wave function, the diverging terms will cancel out; the energy eigenvalue is independent of the position of the state.

In other words, a necessary property of the exact wave function is that the singularities in the many-body interactions are canceled. The basic idea is thus to make sure the trial wave function also has this property. By doing so, it is brought closer to the exact wave function.

These criteria are called *cusp conditions* [9], and serve as powerful guides when it comes to selecting an optimal trial wave function.

1.6.2   Choosing the Trial Wave Function

To recap, choosing the trial wave function boils down to optimizing the overlap $C_0 = \langle \Psi_0 | \Psi_T \rangle$ using a priori knowledge about the system at hand. As discussed previously, the optimal choice of single particle basis is the eigenfunctions of the non-interacting case (given that they exist). Starting from Eq. (1.42), from here on referred to as 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 ansatz for the trial wave function becomes

$$\Psi_T(\mathbf{r}_1, ..., \mathbf{r}_N) = \left[\sum_{k=0}^{K} C_k \Phi_k(\mathbf{r}_1, ..., \mathbf{r}_N) \right] \prod_{i<j}^{N} f(r_{ij}). \tag{1.49}$$

The idea is now to choose $f(r_{ij})$ in such a way that the cusp conditions are obeyed. This should, in light of previous discussions, reduce the amount of terms needed in the spatial wave function to obtain a satisfying overlap.

Explicit shapes

Several models for the correlation function exist, however, some are less practical than others. An example given in ref. [5] demonstrates this nicely: Hylleraas presented the following correlation function

$$f(r_{ij})_{\text{Hylleraas}} = e^{-\frac{1}{2}(r_i + r_j)} \sum_k d_k (r_{ij})^{a_k} (r_i + r_j)^{b_k} (r_i - r_j)^{e_k}, \tag{1.50}$$

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 1078 terms. For the purpose of QMC, including such vast amounts of parameters is out of the question. The strength of QMC is that very good results can be obtained using a very simple ansatz to the wave function.

A commonly used correlation function in studies involving few variational parameters is the *Padé Jastrow* function

$$\prod_{i<j}^{N} f(r_{ij}) = \exp(U),$$

$$U = \sum_{i<j}^{N} \left(\frac{\sum_k a_k r_{ij}^k}{1 + \sum_k \beta_k r_{ij}^k} \right) + \sum_{i}^{N} \left(\frac{\sum_k a_k' r_i^k}{1 + \sum_k \alpha_k r_i^k} \right).$$

For systems where the correlations are relatively well behaved, it is custom to drop the second sum all together, and keep only the $k = 1$ term from the first. The resulting function reads

$$f(r_{ij}; \beta) = \exp \left(\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}} \right), \tag{1.51}$$

where β is a variational parameter, and $a_{k=1} \equiv a_{ij}$ is a constant depending on the relative spin-orientation of particles i and j tuned in such a way that the cusp conditions are obeyed. For three dimensions, $a_{ij} = 1/4$ or $a_{ij} = 1/2$ depending on whether or not the spins of i and j are parallell or anti parallell respectively [5]. For two dimensions, the values are $a_{ij} = 1/3$ (parallell) or $a_{ij} = 1$ (anti-parallel) [12]. This is the correlation function used for all systems in this thesis.

Shifting the focus back to the spatial wave function, in the case of a fermionic system, the evaluation of an $N \times N$ Slater determinant severely limits the efficiency of many-particle simulations. However, assuming the Hamiltonian to be spin-independent, the eigenstates for different spin eigenvalues will be identical.

This fact results in the spatial wave function being split in two: One part for each spin eigenvalue. A detailed derivation of this is given in the appendix of Ref. [13]. The resulting wave function thus becomes

$$\Psi_T(\mathbf{r}; \beta) = \Big[\sum_{k=0}^{K} C_k' \tilde{\Phi}_k(\mathbf{r}_1, ..., \mathbf{r}_{\frac{N}{2}}) \tilde{\Phi}_k(\mathbf{r}_{\frac{N}{2}-1}, ..., \mathbf{r}_N) \Big] \prod_{i<j}^{N} f(r_{ij}; \beta). \tag{1.52}$$

Due to the identical nature of the particles, they may be arbitrarily ordered. For simplicity, the first half represents spin up, and the second half spin down. The spin up determinant will from here on be labeled $|\mathbf{S}^\uparrow|$, and the spin down one $|\mathbf{S}^\downarrow|$, where the $\mathbf{S}$ matrix will be referred to as the *Slater matrix*. Stitching everything together, the explicit shape of the trial wave function becomes

$$\Psi_T(\mathbf{r}_1, ..., \mathbf{r}_N; \beta) = \sum_{k=0}^{K} C_k' |\mathbf{S}^\uparrow|_k |\mathbf{S}^\downarrow|_k \prod_{i<j}^{N} f(r_{ij}; \beta). \tag{1.53}$$

This shape is referred to as a *multi-determinant* trial wave function unless $K = 1$, in which it will be referred to as a *single-determinant* trial wave function.

Limitations

Depending on the complexity of the system at hand, more complicated trial wave functions might be needed to obtain reasonable convergence. 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 alter 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 ($\tau = 0$) into a distribution closer to the exact wave function by Eq. (1.6).

Because of this fact, limitations due to the trial wave function in full[5] QMC is far less than what is the case of standard Monte-Carlo integration. A more complex trial wave function might convergence faster, but at the expense of being more CPU-intensive. This implies that CPU-time per walker can be traded for convergence time. For systems of many particles, the CPU-time per walker needs to be as low as possible in order to get the computation done in a reasonable amount of time. In other words, the choice of trial wave function needs to be done in light of the system at hand, and the specific aim of the computation.

On the other hand, when the number of particles in the system increase, it is safe to assume that the quality of the trial wave function will decrease. This is demonstrated in Ref. [14], where calculations for F_2 (18 particles) need an increase in the number of determinants to achieve a result with the same precision as calculations for O_2 (16 particles).

Single-determinant trial wave functions

In the case of well-behaving systems, a single determinant with a simple Jastrow factor serves as a reasonable trial wave function. This simplicity opens up the possibility of simulating large systems efficiently. More details regarding this will be given in Section **??**.

In order to further optimize the overlap with the exact wave function, a second variational parameter α is introduced in the spatial wave function

---

[5] "Full" in the sense that all Green's functions are applied. As will be revealed later, VMC corresponds to a standard importance sampled Monte-Carlo integration by omitting the branching process.

$$\Psi_T(\mathbf{r}_1, ..., \mathbf{r}_N; \alpha, \beta) = D^{\uparrow}(\alpha) D^{\downarrow}(\alpha) \prod_{i<j}^{N} f(r_{ij}; \beta). \tag{1.54}$$

Determining the optimal values of the variational parameters will be the topic of the next section. If the introduction of the variational parameter was redundant, optimizations would simply yield $\alpha = 1$.

1.6.3 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. (1.6). Consider the following relations

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

Just as with the projection operator, introducing $E_k = E_0 + \delta E_k$ where $\delta E_k \geq 0$ will 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 remarkable: No matter which trial wave function is used, the resulting energy will always be greater or equal to the exact ground state energy. This implies that the problem of choosing variational parameters comes down to a minimization problem in the parameters space

$$\frac{\partial \langle E \rangle}{\partial \alpha_i} = \frac{\partial}{\partial \alpha_i} \langle \Psi_T(\alpha_i) | \widehat{\mathbf{H}} | \Psi_T(\alpha_i) \rangle = 0 \tag{1.55}$$

In order to work with Eq. (1.55) in practice, it needs to be rewritten in terms of known values. Since the wave function depends on the variational parameter, the normalization factor needs to be included in the expression of the expectation value. Applying the product rule numerous times yields

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

The Hamiltonian does not depend on the variational parameters, hence both terms in the first expansion is equal. Cleaning up the expression yields

$$
\begin{aligned}
\frac{\partial \langle E \rangle}{\partial \alpha_i} &= 2 \left(\frac{\langle \Psi_T(\alpha_i) | \widehat{\mathbf{H}} \frac{\partial}{\partial \alpha_i} | \Psi_T(\alpha_i) \rangle}{\langle \Psi_T(\alpha_i) | \Psi_T(\alpha_i) \rangle} - \langle E \rangle \frac{\langle \Psi_T(\alpha_i) | \frac{\partial}{\partial \alpha_i} | \Psi_T(\alpha_i) \rangle}{\langle \Psi_T(\alpha_i) | \Psi_T(\alpha_i) \rangle} \right) \\
&= 2 \left(\left\langle E \frac{\partial \Psi_T}{\partial \alpha_i} \right\rangle - \langle E \rangle \left\langle \frac{\partial \Psi_T}{\partial \alpha_i} \right\rangle \right) .
\end{aligned}
\tag{1.56}
$$

In the case of $\Psi_T(\mathbf{r}; \alpha_i)$ being represented by a Slater determinant, the relationship between the variational derivative of the determinant and the variational derivative of the single particle orbitals $\phi_n(\mathbf{r}_i; \alpha_i)$ is

$$
\frac{\partial \Psi_T(\mathbf{r}; \alpha_i)}{\partial \alpha_i} = \sum_{p=1}^{N} \sum_{q=0}^{N/2} \phi_q(\mathbf{r}_p; \alpha_i) \left[\frac{\partial \phi_q(\mathbf{r}_p; \alpha_i)}{\partial \alpha_i} \right] \mathbf{S}_{qp}^{-1} ,
\tag{1.57}
$$

where $\mathbf{S}_{qi}^{-1}$ is the inverse of the Slater matrix, which will be discussed in more detail in Section **??**.

Using these expressions for the *variational energy gradient*, the derivatives can be calculated exactly the same way the energy. The gradient can then be used to move in the direction of the variational minimum in Eq. (1.55).

This strategy gives rise to numerous ways of finding the optimal parameters, such as using the well known Newton's method, conjugate gradient methods [15], 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. The gradient descent methods will be covered in Section 1.7.

1.6.4   Calculating Expectation Values

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

$$\langle \Psi_T | \widehat{\mathbf{O}} | \Psi_T \rangle \;=\; \int \Psi_T(x)^* \widehat{\mathbf{O}} \Psi_T(x) \mathrm{d}x$$

$$=\; \int |\Psi_T|^2 \left(\frac{1}{\Psi_T(x)} \widehat{\mathbf{O}} \Psi_T(x) \right) \mathrm{d}x$$

$$=\; \int |\Psi_T|^2 O_L(x) \mathrm{d}x. \tag{1.58}$$

$$O_L(x) \;=\; \frac{1}{\Psi_T(x)} \widehat{\mathbf{O}} \Psi_T(x). \tag{1.59}$$

Discretizing the integral yields

$$\langle \Psi_T | \widehat{\mathbf{O}} | \Psi_T \rangle \equiv \langle O \rangle \simeq \frac{1}{n} \sum_{i=1}^{n} O_L(x_i) \equiv \overline{O}, \tag{1.60}$$

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

$$\langle O \rangle = \lim_{n \to \infty} \overline{O} = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} O_L(x_i). \tag{1.61}$$

In the case of the energy estimation, this implies that once the walkers reach equilibrium, local values can be sampled based on their configurations $\mathbf{r}_i$ (remember that Metropolis ensures that the walkers follow $|\Psi_T(\mathbf{r})|^2$). In the case of energies, the explicit expression becomes

$$\langle E \rangle \simeq \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{\Psi_T(\mathbf{r}_i)} \left(-\frac{1}{2} \nabla^2 \right) \Psi_T(\mathbf{r}_i) + V(\mathbf{r}_i) \right). \tag{1.62}$$

Incorporating the branching Green's function G_B into the above equation is covered in the DMC section.

1.6.5   Normalization

Every explicit calculation using the trial wave function in QMC involves taking ratios. Calculating ratios implies a cancellation in the normalization factors. Eq. (1.32) from the Metropolis section, the quantum force in the Fokker-Planck equation, and the sampling of local values described in the previous section demonstrate exactly this; everything involves ratios.

Not having to normalize the wave functions does not only save a lot of CPU-time, but it also removes the need of including the normalization factors of the single particle wave functions; any constants multiplying $\phi_n(x_i)$ in Eq. (1.47) and Eq. (1.48) can be taken outside the sum over permutations, and will thus cancel when the ratio between two wave functions constituting of the same single particle orbitals are computed.

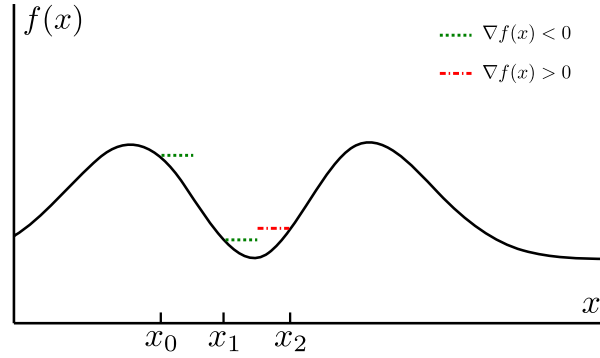Note, however, that this argument is valid for single determinant wave functions only.

Figure 1.4: Two steps of a one dimensional Gradient Descent process. Steps are taken in the direction of the negative gradient (indicated by dotted lines).

1.7 Gradient Descent Methods

The direction of a gradient serves as a guide to extremal values. Gradient descent, also called steepest descent[6], is a family of minimization methods using this property of gradients in order to backtrace a local minimum in the vicinity of an initial guess.

1.7.1 General Gradient Descent

Seeking maxima or minima is simply a question of whether the positive or the 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

$$\nabla f(x_m) \quad = \quad 0 \tag{1.63}$$
$$\nabla f(x_m - dx) \quad < \quad 0 \tag{1.64}$$
$$\nabla f(x_m + dx) \quad > \quad 0 \tag{1.65}$$

where dx is a infinite decimal displacement.

As an example, imagine starting from an initial guess x_0. The direction of the gradient is then calculated and followed a number of steps. From Figure 1.4 and the previous equations, it is clear that crossing the true minimum induces a sign change in the gradient. The brute force way of minimizing is to simply end the calculation at this point, however, this would require an extreme amount of very small steps in order to achieve good precision.

The difference equation describing the steps from the previous paragraph is

$$x_{i+1} = x_i - \delta \frac{\nabla f(x_i)}{|\nabla f(x_i)|}. \tag{1.66}$$

---

[6]In literature, steepest - and gradient descent are sometimes referred to as being different. However, for simplicity, these will not be differentiated.

An improved algorithm would be to continue iterating even though the minimum is crossed, however, this would cause the constant step-length algorithms to oscillate between two points, e.g. x_1 and x_2 in Figure 1.4. To counter this, a changing step-length δ_i is introduced

$$x_{i+1} = x_i - \delta_i \nabla f(x_i). \tag{1.67}$$

All gradient/steepest descent methods are in principle described by Eq. (1.67)[7]. Some examples are

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

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 sequentially or in parallel with different initial guesses.

1.7.2   Stochastic Gradient Descent

Minimizing stochastic quantities such as the variance and expectation values adds another layer of complications on top of the methods introduced 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. (1.56) is an example of such a process.

A precise sampling of the stochastic quantities is 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 is dangerous, as uncertainties in the values will cause the gradient to become unstable when the variations are low close to the minimum. This is illustrated in Figure 1.5.

1.7.3   Adaptive Stochastic Gradient Descent

Adaptive Stochastic Gradient Descent (ASGD) has its roots in the mathematics of automated control theory [16]. 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

$$X_i \equiv -\nabla_i \cdot \nabla_{i-1}. \tag{1.68}$$

The step-length from Eq. (1.67) is modelled in the following manner in ASGD:

---

[7]This fact sets the perfect scene for an object oriented implementation of gradient descent methods.
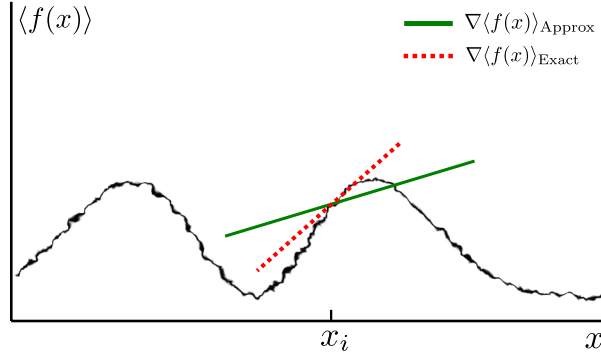
Figure 1.5: A one dimensional plot of an expectation value function. Smeared lines are representing uncertainties due to rough sampling. The direction of the local gradient (solid green line) at a point x_i is not necessarily a good estimate of the actual analytic gradient (dashed red line).

$$\delta_i = \gamma(t_i) \tag{1.69}$$

$$\gamma(t) = a/(t + A) \tag{1.70}$$

$$t_{i+1} = \max(t_i + f(X_i), 0) \tag{1.71}$$

$$f(x) = f_{\min} + \frac{f_{\max} - f_{\min}}{1 - (f_{\max}/f_{\min})e^{-x/\omega}} \tag{1.72}$$

with $f_{\max} > 0$, $f_{\min} < 0$, and $\omega > 0$. The free parameters are a, A and t_0, however, Ref. [17] 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 $f(x)$ is responsible of altering the step-length by changing the trend of t. Close to the minimum, a smaller step-length is sought, and hence t must increase. Being close to the minimum implies that the gradient changes sign frequently. Crossing the minimum with ASGD has the following consequence

- Eq. (1.68): The value of X_i will be positive.

- Eq. (1.72): $f(X_i)$ will return a value in $[0, f_{\max}]$ depending on the magnitude of X_i.

- Eq. (1.71): The value of t will increase, i.e. $t_{i+1} > t_i$.

- Eq. (1.70): The step-length will decrease.

The second step regarding $f(X_i)$ can be visualized in Figure 1.6.

Assumptions

These assumptions are selected direct citations from Ref. [17]. They are listed in order to give an impression that the shapes of the functions used in ASGD are not selected at random, but carefully chosen to work optimally in a stochastic space with averages estimated using very few samples.
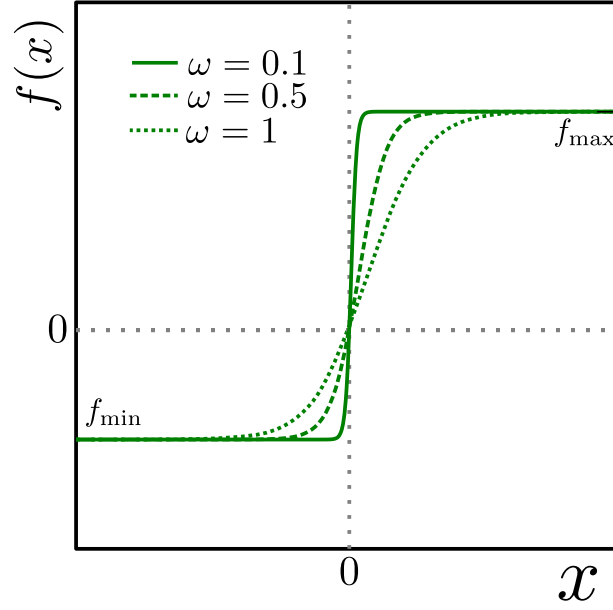
Figure 1.6: Examples of $f(X_i)$ as published in Ref. [17]. As $\omega \to 0$, $f(x)$ approaches a step function.

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

This is shown in Ref. [17]; 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 $t = 0$.

With $\gamma(t)$ being as in Eq (1.70), this is easily shown.

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

This is exactly the behavior displayed in Figure 1.6.

Implementation

A flow chart of the implementation is given in Figure 1.8. For specific details regarding the implementation, see the documentation of the code in Ref. [10]. An example of minimization using ASGD is given in Figure 1.7.
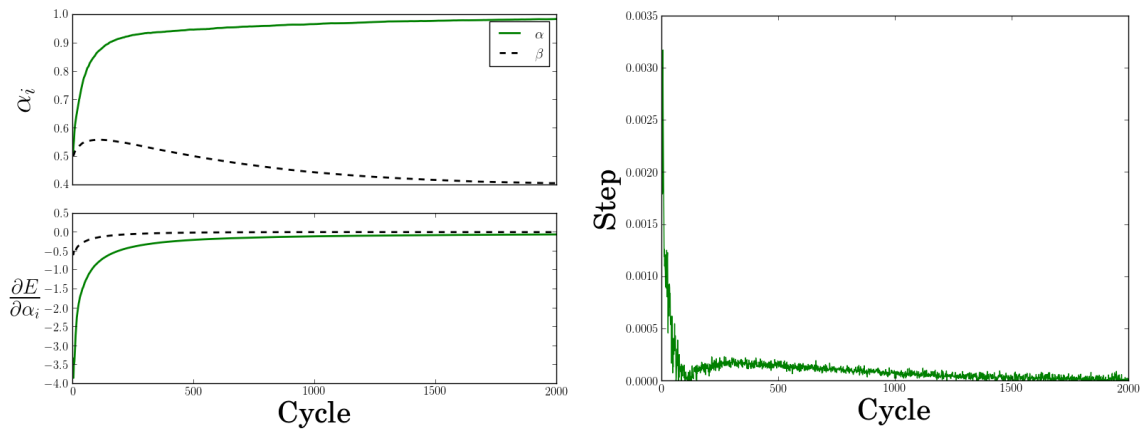
Figure 1.7: Results of Adaptive Stochastic Gradient Descent used on a two-particle quantum dot with unit oscillator frequency using 400 cycles pr. gradient sampling and 40 independent walkers. The right figure shows the evolution of the time-step. The left figure shows the evolution of the variational parameters α and β introduced in Section 1.6 on top, and the evolution of the gradients on the bottom. The gradients are averaged to reveal the pattern underlying the noise. Dispite this averaging, it is apparent that they tend to zero, β somewhat before α. The step rushes to zero with a small rebound as it attempts to cross to negative values.
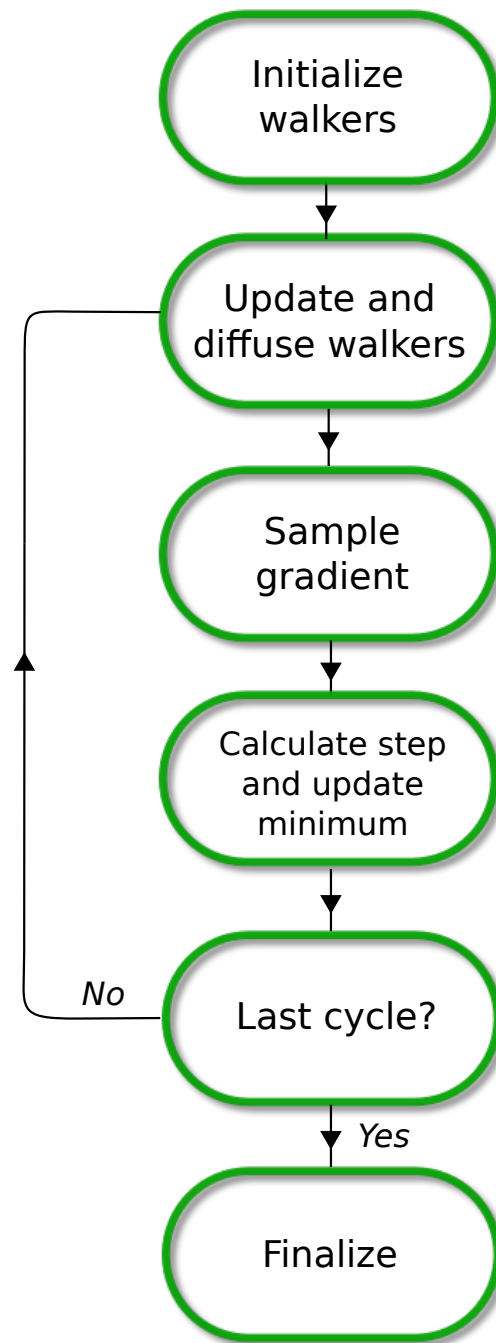
Figure 1.8: Chart flow of ASGD algorithm. Diffusing a walker is done as described in Figure 1.1. Updating the walkers involves recalculating any values affected by updating the minimum. The step is calculated by Eq. (1.70). In the case of QMC, the gradient is sampled by Eq. (1.56).

1.8   Variational Monte-Carlo

As briefly mentioned in the Section 1.1.1, neglecting the branching term, i.e. setting $G_{\mathrm{B}} = 1$, corresponds to a method called Variational Monte-Carlo (VMC). The name comes from the fact that the method is is variational, that is, it supplies an upper bound to the exact ground state energy. The better the trial wave function is, the closer the VMC energy is to the exact ground state energy.

The converged state of the Markov chain in VMC is controlled by the Metropolis test, and will thus equal the trial wave function squared. From the flow chart of the VMC algorithm in Figure 1.9, it is clear that VMC corresponds to a standard Monte-Carlo integration of the local energy sampled using a distribution equal to the trial wave function squared.

1.8.1   Motivating the use of Diffusion Theory

An important question to answer is why e.g. the Langevin equation is so important if the result is simply an expectation value. Statistics states that *any* distribution may be used when calculating an expectation value. Why bother with a trial wave function, thermalization, and so on?

The reason is simple, yet not obvious. The quantity of interest, the local energy, *depends on the trial wave function*. This is demonstrated in the following expression:

$$
\begin{aligned}
E_{\mathrm{VMC}} &= \int P(\mathbf{r}) \frac{1}{\Psi_T(\mathbf{r})} \widehat{\mathbf{H}} \Psi_T(\mathbf{r}) \mathrm{d}\mathbf{r} \\
&\simeq \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\Psi_T(\mathbf{r}_i)} \widehat{\mathbf{H}} \Psi_T(\mathbf{r}_i),
\end{aligned}
\tag{1.73}
$$

where the points $\mathbf{r}_i$ are drawn from the distribution $P(\mathbf{r})$.

Equation (1.73) implies that the evaluation of the local energy in an arbitrary distribution $P(\mathbf{r})$ is *undefined* at the zeros of $\Psi_T(\mathbf{r})$. In other words, it is not guaranteed that $P(\mathbf{r})$ does not generate a point $\mathbf{r}_m$ in such a way that $\Psi_T(\mathbf{r}_m) = 0$.

However, if the distribution is chosen as $P(\mathbf{r}) = |\Psi_T(\mathbf{r})|^2$, the probability of sampling a point where the local energy is undefined equals zero. This comes as a consequence of the following relation

$$
\Psi_T(\mathbf{r}_m) = 0 \quad \implies \quad P(\mathbf{r}_m) = 0
\tag{1.74}
$$

In other words, the more undefined the energy is at a point, the less probable the point is. This hidden detail is what Quantum Monte-Carlo safely takes care of that standard Monte-Carlo does not.
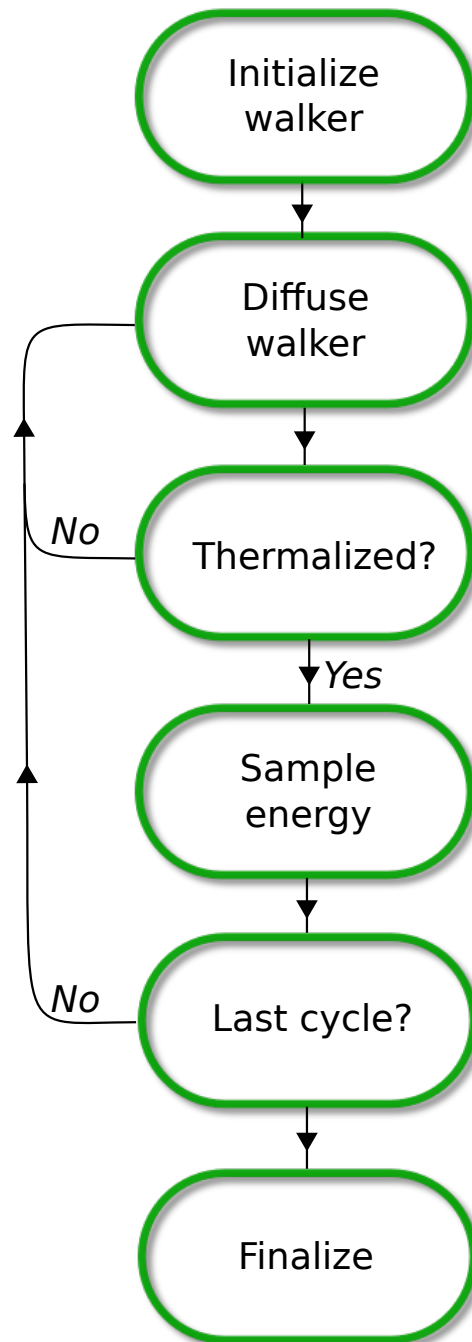
Figure 1.9: Chart flow of the Variational Monte-Carlo algorithm. The second step, *Diffuse Walker*, is the process described in Figure 1.1. Energies are sampled as described in Section 1.6.4. The thermalization is set to a fixed number of cycles.
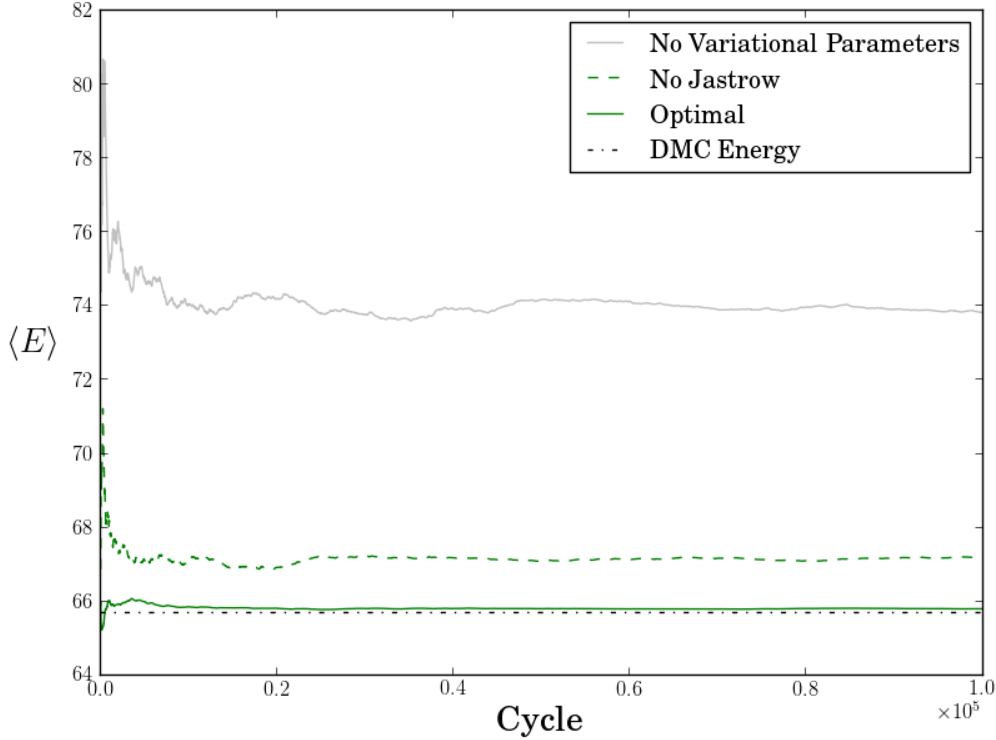
Figure 1.10: Comparison of the VMC energy using different trial wave functions. The DMC energy is believed to be very close to the exact ground state. It is apparent that adding a variational parameter to the trial wave function lowers the energy substantially, however, when adding the Jastrow factor (denoted *Optimal*), described in Section 1.6.2, the VMC energy gets very close to the "exact" answer. A lower energy means a better energy when dealing with variational methods. In this example, a 12-particle quantum dot with unit frequency is used.

1.8.2   Implementation

Beyond a given point, VMC does not benefit much from an increase of samples. It is much more important that the system is properly thermalized, than using several walkers or many cycles. It is therefore sufficient to use a single walker per VMC simulation.

The single walker is initialized in a normal distributed manner, and released to diffuse according to the process in Figure 1.1. After accumulating energies, the final energy is calculated as the mean of these. The process is described in Figure 1.9.

For more details regarding the specific implementation, see the code in Ref. [10].

1.8.3   Limitations

The only limitation in VMC is the choice of the trial wave function. This makes VMC extremely robust; it will *always* produce a result. As the overlap C_0 from Eq. (1.6) approach unity, the VMC energy approaches the exact ground state energy as a monotone decreasing function. Figure 1.10 demonstrates this effect. As more optimizations are added to the trial wave function, the lower the VMC energy gets.

1.9 <u>Diffusion Monte-Carlo</u>

Applying the full diffusion framework introduced in Sections 1.1.1 and 1.2 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 DMC is overall one of the most precise many-body methods available. However, just as any many-body method, DMC also has its limitations. These will be discussed in Section 1.9.3.

Where other many-body methods run into the *curse of dimensionality*, that is, the CPU-time scales very bad with increasing number of particles, DMC with its 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 of cycles, in order to reach convergence to the believed "exact" ground state in a satisfactory way.

1.9.1 Implementation

Branching is a major part of the DMC algorithm. Diffusion Monte-Carlo 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 initially distributed according to the trial wave function squared.

There are three layers of loops in the DMC method implemented in this thesis, two of which are obvious: The time-step - and walker loops. However, introducing a third *block loop* within the walker loop boosts the convergence dramatically. For each walker, this loop continues until the walker is either dead ($G_B = 0$), or has 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 Figure 1.11. Comparing it the the corresponding figure for VMC, it is apparent that DMC is by far more complicated.

1.9.2 Sampling the Energy

As mentioned in Section 1.1.1, the role of the branching Green's function is to distribute the correct weights to each walker. Each walker's contribution to the cumulative energy sampling should thus be weighed according to the value of the branching Green's function. Let E_k denote the cumulative energy for time-step $\tau = k\delta\tau$, n_w be the number of walkers in the system at time-step k, $\tilde{n}_{b,i}$ be the number of blocks walker i survives, and let $W_i(\mathbf{r}, \tau)$ represent walker i. The relation is then

$$E_k = \frac{1}{n_w} \sum_{i=1}^{n_w} \frac{1}{\tilde{n}_{b,i}} \sum_{l=1}^{\tilde{n}_{b,i}} G_{\mathrm{B}}\Big(W_i(\mathbf{r}, \tau_k + l\delta\tau)\Big) E_L\Big(W_i(\mathbf{r}, \tau_k + l\delta\tau)\Big). \tag{1.75}$$

As required, setting $G_B = n_w = n_b = 1$ reproduces the VMC expression.

The new trial energy from Eq. (1.13) is set equal to the previous cycle's average energy, that is

$$E_T = E_k. \tag{1.76}$$

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

---

[8]In literature, DMC is also known as *Projection Monte-Carlo*, for reasons described in Section 1.1.1.

$$E_{\mathrm{DMC}} = \overline{E_T} = \frac{1}{n} \sum_{k=1}^{n} E_k. \tag{1.77}$$

1.9.3   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 [18], which is far more automatic.

Time-step errors

The error introduced by the short time approximation in Eq. (1.14) goes as $\mathcal{O}(\delta\tau^2)$. In addition, 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 an effective reduction in the time-step, and is countered by scaling the time-step with the acceptance ratio upon calculating G_B [5]. However, DMC is rarely used without importance sampling, which, due to the quantum force, has an acceptance ratio very close to one. It is therefore common to ignore this problem, as its effect on the final result is minimal.

Selecting the time-step

Studying the branching Green's function in equation 1.38 in more detail reveals that it's magnitude increases exponentially with the spread of the energies ΔE, that is

$$G_B \propto \exp\left(\Delta E \delta\tau\right). \tag{1.78}$$

As will be shown in Section 1.11.1, the spread in energy samples are higher the worse of an approximation to the ground state the trial wave function is. In addition, the magnitude of the spread scales with the magnitude of the energy.

Setting an upper bound to the branching function might seem like a good idea, however, each time a walker is denied its replicas, an uncontrollable error is introduced in the distribution.

The solution is to balance out the increase in ΔE by lowering the time-step accordingly. That is

$$\delta\tau \propto \frac{1}{\Delta E}. \tag{1.79}$$

However, having too low a time-step will hinder DMC from evolving walkers efficiently, especially if the positional span of the distribution is large. In other words, a balance must be found. An indication of whether the time-step was chosen too low or not is obtained by looking at the resulting DMC density. If the density is spiky and disorganized, the time-step was too low.

Another source of error is due to the *fixed node approximation*. This approximation will be covered in the next section.
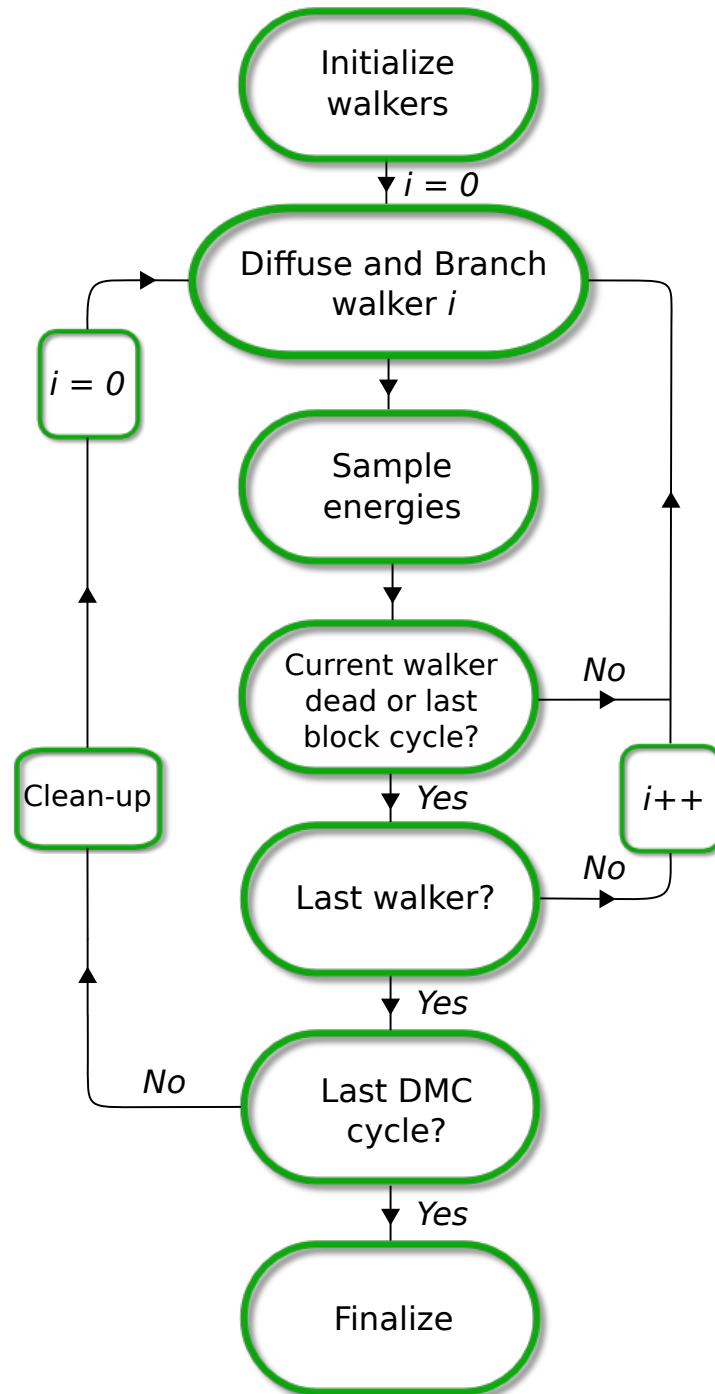
Figure 1.11: Chart flow of the Diffusion Monte-Carlo algorithm. The variable i represents the currently moved walker. The second step, *Diffuse and Branch Walker*, is the process described in Figure 1.1 in combination with the branching from Figure 1.2. Energies are sampled as in Eq. (1.75). Thermalization is set to a fixed number of cycles.

1.9.4   Fixed node approximation

Looking at Eq. (1.48), it is apparent that by choosing positive phases for the single particle wave functions, the bosonic many-body wave function is exclusively positive. For fermions however, the sign change upon interchanging two particles introduces the possibility that the wave function will have both negative and positive regions, independent of the choice of phases in the single particle wave functions.

As DMC iterates, the density of walkers at a given time, $P(\mathbf{r}, \tau)$, represents the projected wave function from Eq. (1.6) multiplied by the trial wave function.

$$P(\mathbf{r}, \tau) = \Phi(\mathbf{r}, \tau)\Psi_T(\mathbf{r}). \tag{1.80}$$

This relationship is described in high detail in Section 1.2.2.

Applying the projector operator to the distribution yields

$$\lim_{\tau \to \infty} P(\mathbf{r}, \tau) = \langle \Phi_0 | \Phi_T \rangle \, \Phi_0(\mathbf{r})\Psi_T(\mathbf{r}), \tag{1.81}$$

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 of the exact ground state and the trial wave function will generally be different.

To avoid this anomaly in the density, $\Phi(\mathbf{r}, \tau)$ and $\Psi_T(\mathbf{r})$ have to change sign simultaneously[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:

$$\frac{\Psi_T(\mathbf{r}_i)}{\Psi_T(\mathbf{r}_j)} < 0 \quad \Longrightarrow \quad A(i \to j) = 0, \tag{1.82}$$

where $A(i \to j)$ is the probability of accepting the move, as described in Section 1.4. An illustrative example is given in Figure 1.12.

1.10   Estimating One-body Densities

The one-body density is defined as

$$\rho(\mathbf{r}_1) = \underset{\mathbf{r}_2\mathbf{r}_3}{\iint} ... \underset{\mathbf{r}_N}{\int} |\Phi(\mathbf{r}_1\mathbf{r}_2...\mathbf{r}_N)|^2 \, \mathrm{d}\mathbf{r}_2...\mathrm{d}\mathbf{r}_N. \tag{1.83}$$

Unlike the distribution $|\Phi(\mathbf{r})|^2$, which describes the distribution of any of the particles in the system, the one-body density $\rho(\mathbf{r}_1)$ describes the simultaneous distribution of every particle in the system, that is, $\rho(\mathbf{r}_1)\mathrm{d}\mathbf{r}_1$ represents the probability of finding *any* of the system's N particles within the volume element $\mathrm{d}\mathbf{r}_1$. Due to the indistinguishable nature of the particles, the fact that the first coordinate is chosen is purely conventional; any of the N coordinates contain information about all the particles. For the same reason, the one-body density should be normalized to the number of particles, and not to unity.

---

[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 regions, however, due to the position space being infinite, this requires an enormous amount of walkers to succeed.
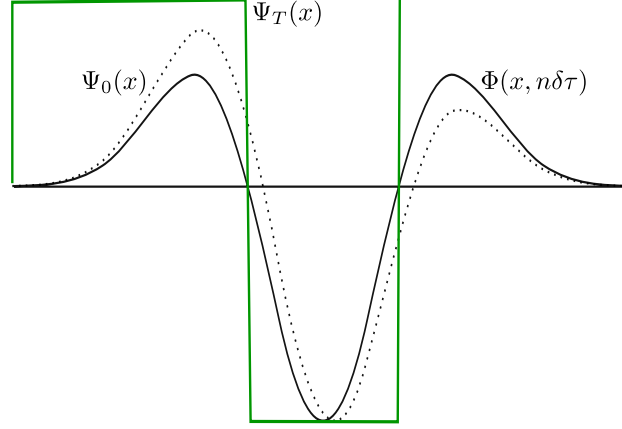
Figure 1.12: A one-dimensional illustration of the fixed node approximation. The dotted line represents the exact ground state $\Psi_0(x)$. The distribution of walkers after n Monte-Carlo cycles is represented by $\Phi(x, n\delta\tau)$. The trial wave function $\Psi_T(x)$ shares nodes with $\Phi(x, n\delta\tau)$, making it impossible for $\Phi(x, n\delta\tau)$ to match $\Psi_0(x)$.

In a Monte-Carlo simulation, estimating the one-body density is done by collecting snapshots of the walkers' positions. These snapshots serve as samples to a histogram where each set of Cartesian coordinates give rise to one histogram-count.

1.10.1   Estimating the Exact Ground State Density

It was mentioned in the section on VMC that the final state of convergence was to have to walkers span the trial wave function. This implies that the one-body density of the trial wave function, called the *variational density*, can be calculated using positional data generated from VMC.

The challenge lies in estimating the one-body density of the exact wave function given a set of DMC data. As described in Section 1.2.2, the distribution of walkers span the *mixed density* $f(\mathbf{r}, \tau) = \Phi(\mathbf{r}, \tau)\Psi_T(\mathbf{r})$, which does not correspond to the ground state distribution unless the trial wave function is indeed the exact ground state. Hence a histogram of the DMC data does not suffice when it comes to presenting the exact wave function.

This implies the need of a method for transforming the one-body density of $f(\mathbf{r}, \tau)$ into the *pure density* $|\Phi(\mathbf{r}, \tau)|^2$. To achieve this, the following relation is needed [5]

$$\langle A \rangle_0 = \frac{\langle \Phi_0 | \widehat{\mathbf{A}} | \Phi_0 \rangle}{\langle \Phi_0 | \Phi_0 \rangle} \simeq 2 \frac{\langle \Phi_0 | \widehat{\mathbf{A}} | \Psi_T \rangle}{\langle \Phi_0 | \Psi_T \rangle} - \frac{\langle \Psi_T | \widehat{\mathbf{A}} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle} + \mathcal{O}(\Delta^2)$$
$$= 2\langle A \rangle_{\text{DMC}} - \langle A \rangle_{\text{VMC}} + \mathcal{O}(\Delta^2), \tag{1.84}$$

where $\Delta \equiv \Psi_T(\mathbf{r}) - \Phi(\mathbf{r}, \tau)$.

Expressed in terms of *density operators*, the expectation values for the different methods are [19]

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

where tr denotes the *trace*, i.e. the sum of all eigenvalues. Inserting these equations into Eq. (1.84) yield

$$\mathrm{tr}(\widehat{\rho}_0 \widehat{\mathbf{A}}) \simeq 2\mathrm{tr}(\widehat{\rho}_{\mathrm{DMC}} \widehat{\mathbf{A}}) - \mathrm{tr}(\widehat{\rho}_{\mathrm{VMC}} \widehat{\mathbf{A}}) + \mathcal{O}(\Delta^2)$$
$$\simeq \mathrm{tr}\left(\left(2\widehat{\rho}_{\mathrm{DMC}} - \widehat{\rho}_{\mathrm{VMC}}\right) \widehat{\mathbf{A}} \right) + \mathcal{O}(\Delta^2), \tag{1.85}$$

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

$$\widehat{\rho}_0 \simeq 2\widehat{\rho}_{\mathrm{DMC}} - \widehat{\rho}_{\mathrm{VMC}}. \tag{1.86}$$

It is clear that if this relation holds for regular densities, it will hold for one-body densities as well. In other words, the resulting densities from DMC can be combined with the corresponding one from VMC to produce the pure density.

1.10.2   Radial Densities

Integrating out every degree of freedom except one radial coordinate from the density results in the radial one-body density times the radius (squared for three dimensions). In other words

$$I_{\mathrm{3D}} = \iint \rho(\mathbf{r}_1, \theta_1, \phi_1) r_1^2 \sin\theta_1 \mathrm{d}\theta_1 \mathrm{d}\phi_1$$
$$\propto r_1^2 \rho(\mathbf{r}_1), \tag{1.87}$$
$$I_{\mathrm{2D}} = \int \rho(\mathbf{r}_1, \phi_1) r_1 \mathrm{d}\phi_1$$
$$\propto r_1 \rho(\mathbf{r}_1). \tag{1.88}$$

In practice, these integrals are calculated by creating histograms $H(r)$ of all sampled radii. Transforming the histograms into radial one-body densities $\rho(r_1)$ are according to Eq. (1.87) and Eq. (1.88) done in the following manner

$$\rho(\mathbf{r}_1) = \frac{H(\mathbf{r}_1)}{r_1^{(d-1)}}, \tag{1.89}$$

where d denotes the number of dimensions. An example presenting two radial one-body densities is given in Figure 1.13. Notice, however, that the radial density for certain systems such as atoms needs to be scaled with r or r^2 in order to reveal the interesting shapes.
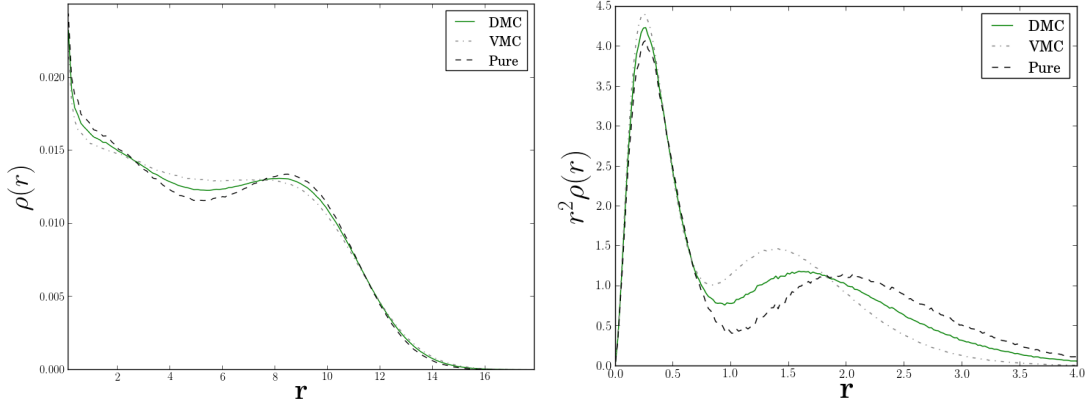
Figure 1.13: Two examples of radial one-body densities for VMC, DMC, and the pure density from Eq. (1.86). On the left: A 12-particle two-dimensional quantum dot with frequency $\omega = 0.1$. The density diverges close to zero due to a "$\frac{0}{0}$" expression (see Eq. (1.89)). On the right: Unscaled radial density for the beryllium atom, i.e. $r_1^2 \rho(\mathbf{r}_1)$. The densities will be discussed in the result section.

1.11 Estimating the Statistical Error

As with any statistical result, the statistical errors needs to be supplied in order for it to be taken seriously. Systematic errors, that is, errors introduced due to limitations in the model, is discussed in each method's respective section, and will not be related to the statistical error.

Statistical errors, that is, the deviation from the true ensemble average due to the fact that the equality in Eq. (1.61) can never be fulfilled, can be estimated using several methods, some of which are *naive* in the sense that they assume the dataset to be completely *uncorrelated*, i.e. the samples are independent of each other.

1.11.1 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

$$
\begin{aligned}
\mathrm{Var}(E) &= \left\langle (E - \langle E \rangle)^2 \right\rangle \\
&= \left\langle E^2 \right\rangle - 2 \underbrace{\left\langle E \left\langle E \right\rangle \right\rangle}_{\langle E \rangle \langle E \rangle} + \left\langle E \right\rangle^2 \\
&= \left\langle E^2 \right\rangle - \left\langle E \right\rangle^2 & (1.90) \\
&\simeq \overline{E^2} - \overline{E}^2 & (1.91)
\end{aligned}
$$

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 of a fit different trial wave functions are to the system. A common misconception is that the numerical value of the variance can be used to compare properties of *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 fitting trial wave function. However, this is not true. The variance has unit energy squared (in the case of local energies), 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,

$$\sigma^2(x) = \mathrm{Var}(x), \tag{1.92}$$

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 $\langle E \rangle$.

1.11.2   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

$$\begin{aligned}
\mathrm{Cov}(x, y) &\equiv \langle (x - \langle x \rangle)(y - \langle y \rangle) \rangle \\
&= \langle xy - x\langle y \rangle - \langle x \rangle y + \langle x \rangle \langle y \rangle \rangle \\
&= \langle xy \rangle - \langle x \langle y \rangle \rangle \underbrace{- \langle y \langle x \rangle \rangle + \langle \langle x \rangle \langle y \rangle \rangle}_{0} \\
&= \langle xy \rangle - \langle x \rangle \langle y \rangle. \tag{1.93}
\end{aligned}$$

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

The consequence of ignoring the correlations is a resulting 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[10]. Denoting the true standard deviation as σ_c, the above discussion can be distilled to

$$\sigma_c(x) \geq \sigma(x), \tag{1.94}$$

where $\sigma(x)$ is the deviation from Eq. (1.92).

---

[10]Samples in QMC is obviously correlated due to the nature of the Langevin equation (difference equation).

1.11.3 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 its combined mean, in other words:

$$\sigma(\overline{x}) \neq \sigma(x). \tag{1.95}$$

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 the results of all the simulations.

$$m \equiv \overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{1.96}$$

$$\sigma^2(m) = \langle m^2 \rangle - \langle m \rangle^2 \tag{1.97}$$

Combining the above equations yields

$$
\begin{aligned}
\sigma^2(m) &= \left\langle \frac{1}{n^2} \left[\sum_{i=1}^{n} x_i \right]^2 \right\rangle - \left\langle \frac{1}{n} \sum_{i=1}^{n} x_i \right\rangle^2 \\
&= \frac{1}{n^2} \left(\left\langle \sum_{i=1}^{n} x_i \sum_{j=1}^{n} x_j \right\rangle - \left\langle \sum_{i=1}^{n} x_i \right\rangle \left\langle \sum_{j=1}^{n} x_j \right\rangle \right) \\
&= \frac{1}{n^2} \sum_{i,j=1}^{n} \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle \\
&= \frac{1}{n^2} \sum_{i,j=1}^{n} \mathrm{Cov}(x_i, x_j)
\end{aligned} \tag{1.98}
$$

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 the samples are uncorrelated. Going back to the definition of covariance in Eq. (1.93), it is apparent that in order to calculate the covariance as in Eq. (1.98), the true mean $\langle x_i \rangle$ needs to be known. Using $m = \overline{x}$ as a necessary approximation to the true mean yields

$$
\begin{aligned}
\mathrm{Cov}(x_i, x_j) &\equiv \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle \\
&\simeq \langle (x_i - m)(x_j - m) \rangle \\
&\simeq \frac{1}{n^2} \sum_{k,l=1}^{n} (x_k - m)(x_l - m) \tag{1.99} \\
&\equiv \frac{1}{n} \mathrm{Cov}(x) \tag{1.100}
\end{aligned}
$$

Inserting this relation into Eq. (1.98) yields

$$
\begin{aligned}
\sigma^2(m) &= \frac{1}{n^2} \sum_{i,j=1}^{n} \mathrm{Cov}(x_i, x_j) \\
&\simeq \frac{1}{n^2} \sum_{i,j=1}^{n} \frac{1}{n} \mathrm{Cov}(x) \\
&= \frac{1}{n^3} \mathrm{Cov}(x) \underbrace{\sum_{i,j=1}^{n}}_{n^2} \\
&= \frac{1}{n} \mathrm{Cov}(x),
\end{aligned}
\tag{1.101}
$$

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*[11], τ, 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 $\overline{x}$

$$
\overline{x} = \frac{1}{n} \sum_{k=0}^{n} x_{k \cdot \tau}
\tag{1.102}
$$

This implies that $n\tau$ samples are needed in order to get the same magnitude of samples to the average as in Eq. (1.96); the *effective sample size* becomes $n_{\mathrm{eff}} = n_{\mathrm{tot}}/\tau$. In the cases where $\tau = 1$, the sample set is uncorrelated. For details regarding the derivations of τ based on the covariance, see Refs. [20] and [9].

1.11.4   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. there is no need to know the error at any stage during the sampling. This means that the error estimation can be done post process (given that the sample set is stored).

An efficient algorithm for calculating the error of correlated data is *blocking*. This method is described in high detail in ref. [20], 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 naturally increase as n decrease, (see Eq. (1.101))

$$
\sigma_n \propto \frac{1}{\sqrt{n}}
\tag{1.103}
$$

However, treating each block as an individual simulation, n_b averages m_n can be used to calculate the total error from Eq. (1.97), that is, estimate the covariance

---

[11]In literature, this parameter is often referred to as the *auto-correlation time*.
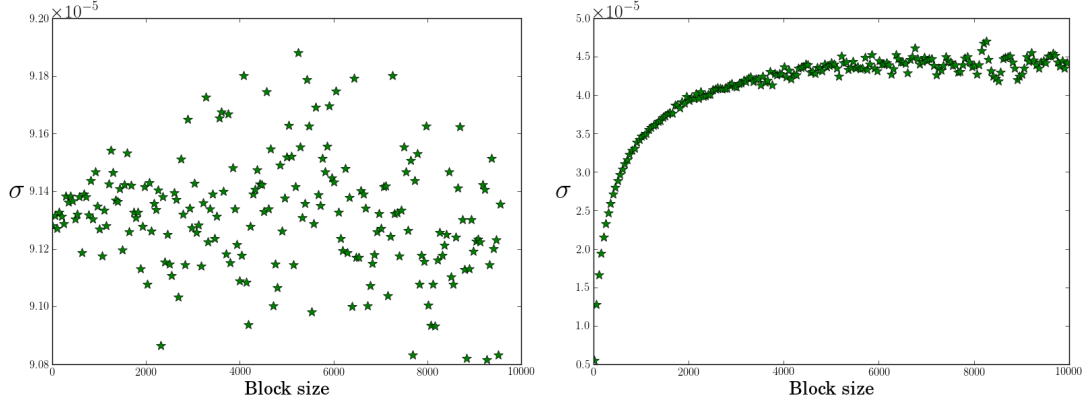
Figure 1.14: Left hand side: Blocking result of (approximately) uncorrelated data generated from a uniform Monte-Carlo integration of $\int_1^2 2x \mathrm{d}x$ resulting in 3.00003 (exact is 3.0). This is in excellent agreement with the magnitude of the error $\sim 9 \cdot 10^{-5}$. There is no sign of a plateau, which implies fairly uncorrelated data (the span of the spread is small and apparently random). Right hand side: Blocking result of a DMC simulation of a 6-particle $\omega = 0.1$ quantum dot. The plateau is strongly present, implying correlated data. The resulting total error is $\sim 4.5 \cdot 10^{-5}$.

$$\overline{m_n^r} \equiv \frac{1}{n_b} \sum_{k=1}^{n_b} m_k^r \tag{1.104}$$

$$\begin{aligned} \sigma^2(m) &= \langle m^2 \rangle - \langle m \rangle^2 \\ &\simeq \overline{m_n^2} - (\overline{m_n})^2 \end{aligned} \tag{1.105}$$

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. However, what is known, is that if the system is correlated, there should be a range of different block sizes which fulfills Eq. (1.105) to reasonable precision.

The result of a blocking analysis is therefore a series of $(n, \sigma(m))$ pairs which can be plotted. The plot should in light of previous arguments result in a increasing curve which stabilizes over a certain span of block sizes (where Eq. (1.105) is fulfilled). This plateau will then serve as a reasonable approximation to the covariance, that is, the true average. See Figure 1.14 for a demonstration of resulting blocking plots.

1.11.5 Variance Estimators

The standard intuitive variance estimator

$$\sigma^2(x) \simeq \frac{1}{n} \sum_{i=1}^n (x_i - \overline{x})^2 = \left(\frac{1}{n} \sum_{i=1}^n x_i^2 \right) - \overline{x}^2, \tag{1.106}$$

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

$$\sigma^2(x) \simeq \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2 = \left(\frac{1}{n-1} \sum_{i=1}^{n} x_i^2 \right) - \frac{n}{n-1} \overline{x}^2, \tag{1.107}$$

which is only noticeably different from Eq. (1.106) when the sample size gets small, as it does in blocking analysis. It is therefore standard to use Eq. (1.107) for blocking errors.

Part II

Results

Appendices

Bibliography

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

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

[3] S. B. Skattum, "Time Evolution of Quantum Dots," Master's thesis, University of Oslo, 2013.

[4] J. J. Mares *et al.*, "Periodic reactions and quantum diffusion." [Online]. Available: http://www.fzu.cz/~sestak/yyx/periodic%20reactions.pdf

[5] 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.

[6] 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

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

[8] 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.

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

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

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

[12] L. E. Lervåg, "VMC CALCULATIONS OF TWO-DIMENSIONAL QUANTUM DOTS," Master's thesis, University of Oslo, 2010.

[13] 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

[14] C. Filippi and C. J. Umrigar, "Multiconfiguration wave function for quantum Monte Carlo calculations of first-row diatomic molecules," *The Journal of Chemical Physics*, vol. 105, Jul. 1996.

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

[16] 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.

[17] 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

[18] C. Hirth, "Studies of quantum dots: Ab initio coupled-cluster analysis using OpenCL and GPU programming," Master's thesis, University of Oslo, 2012.

[19] J. M. Leinaas, "Non-Relativistic Quantum Mechanics," lecture notes FYS4110.

[20] 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