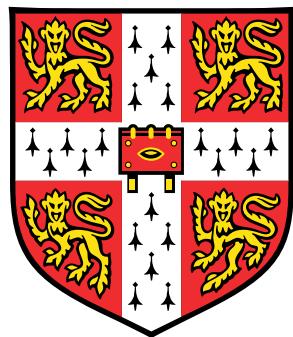


# **Optimal Importance Sampling in Quantum Monte Carlo for Lattice Models**



**Blaž Stojanovič**

Supervisor: Prof. A. Lamacraft

Department of Physics

This dissertation is submitted for the degree of  
*Master of Philosophy in Scientific Computing*

St. John's College

June 2021



I would like to dedicate this thesis to ...



## **Declaration**

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 15,000 words including appendices, figure legends, and tables.

Blaž Stojanović

June 2021



## **Acknowledgements**

And I would like to acknowledge ...

s



## **Abstract**

This is where you write your abstract ...



# Table of contents

|   |             |
|---|-------------|
| <b>List of figures</b>  | <b>xiii</b> |
| <b>List of tables</b>   | <b>xv</b>   |
| <b>Nomenclature</b>   | <b>xvii</b> |
| <b>1 Introduction</b>   | <b>1</b>    |
| 1.1 Thesis Contributions . . . . .  | 1           |
| 1.2 Thesis Structure . . . . .  | 1           |
| <b>2 Background</b>   | <b>3</b>    |
| 2.1 Lattice models . . . . .  | 3           |
| 2.1.1 Historical introduction . . . . .                                   | 3           |
| 2.1.2 The Schrödinger equation and the Feynman path integral . . . . .    | 3           |
| 2.1.3 Examples of lattice models . . . . .                                | 4           |
| 2.2 Solutions to the many-body problem and QMC . . . . .                  | 5           |
| 2.2.1 Overview . . . . .  | 5           |
| 2.3 The Feynman-Kac formula . . . . .                                     | 8           |
| 2.3.1 Stochastic processes . . . . .                                      | 8           |
| 2.3.2 Feynman-Kac formula . . . . .                                       | 8           |
| 2.3.3 Stoquastic Hamiltonians and Lattice-model representations . . . . . | 13          |
| 2.4 Quantum Mechanics, Control and loss functions . . . . .               | 14          |
| 2.4.1 Continuous space . . . . .  | 14          |
| 2.4.2 Discrete space . . . . .  | 14          |
| 2.5 Machine Learning . . . . .  | 15          |
| 2.5.1 Overview of ML approaches to the Quantum many-body problem          | 15          |
| 2.5.2 Neural Networks . . . . .   | 15          |

|   |           |
|---|-----------|
| <b>3 Methodology</b>                          | <b>17</b> |
| 3.1 Monte Carlo Importance Sampling . . . . . | 17        |
| 3.2 Metropolis-Hastings Algorithm . . . . .   | 18        |
| 3.3 Gradient based optimisation . . . . .     | 19        |
| 3.3.1 Gradient estimation . . . . .           | 19        |
| 3.3.2 Automatic differentiation . . . . .     | 21        |
| 3.4 <i>Optimal sampling</i> . . . . .         | 21        |
| <b>4 Results</b>                              | <b>23</b> |
| 4.1 Transverse Field Ising model . . . . .    | 23        |
| 4.2 Heisenberg model . . . . .                | 23        |
| 4.3 Bose-Hubbard model . . . . .              | 23        |
| <b>5 Conclusion</b>                           | <b>25</b> |
| 5.1 Direction for future work . . . . .       | 25        |
| 5.2 Remarks . . . . .                         | 25        |
| <b>References</b>                             | <b>27</b> |
| <b>Appendix A Jax ecosystem</b>               | <b>29</b> |
| <b>Appendix B Some additional results</b>     | <b>31</b> |
| <b>Appendix C Probability distributions</b>   | <b>33</b> |

# List of figures

|     |   |    |
|-----|---|----|
| 2.1 | Feynman-Kac for a 1D free particle . . . . .        | 10 |
| 2.2 | Feynman-Kac measure in a linear potential . . . . . | 11 |
| 3.1 | The reparametrization trick . . . . .               | 21 |



## **List of tables**



# Nomenclature

## Other Symbols

$\mathbb{E}$       Expectation

Var      Variance

## Acronyms / Abbreviations

CNN    Convolutional Neural Network

DL      Deep Learning

DMC    Diffusion Quantum Monte Carlo

e.g.    Exempli gratia ("for the sake of an example")

FP      Fokker-Planck

GAN    General Adversarial Network

i.e.    Id est ("it is")

i.i.d    Independent and identically distributed

ML      Machine Learning

NN      Neural Network

QMC    Quantum Monte Carlo

SDE    Stochastic Differential Equations

s.t.    Such that

VAE    Variational Autoencoder

VMC Variational Quantum Monte Carlo

w.r.t With respect to

# **Chapter 1**

## **Introduction**

### **1.1 Thesis Contributions**

### **1.2 Thesis Structure**



# Chapter 2

## Background

### 2.1 Lattice models

#### 2.1.1 Historical introduction

- I think starting from the magnetism point of view might be the best way to go, slowly lead into the field of condensed matter theory and lattice models.

#### 2.1.2 The Schrödinger equation and the Feynman path integral

The wavefunction

$$\Psi(r_1, \dots, r_N) \quad (2.1)$$

the Schrödinger equation

$$i \frac{\partial \psi(\mathbf{r}, t)}{\partial t} = \hat{H} \psi(\mathbf{r}, t) \quad (2.2)$$

for a single particle in an external potential  $\hat{V}(\mathbf{r})$  the Hamiltonian is

$$\hat{H} \phi(\mathbf{r}) = -\frac{1}{2} \nabla^2 \phi(\mathbf{r}) + \hat{V}(\mathbf{r}) \phi(\mathbf{r}). \quad (2.3)$$

Alternatively to the Schrödinger equation one can use an integral Green's function representation to express the wavefunction  $\psi$  at some future time  $t_2$  given initial condition  $\psi(\mathbf{r}, t_1)$  as

$$\psi(\mathbf{r}_2, t_2) = \int \mathcal{K}(\mathbf{r}_2, t_2; \mathbf{r}_1, t_1) \psi(\mathbf{r}_1, t_1) d\mathbf{r}_1. \quad (2.4)$$

The solution to equation

$$\left( i \frac{\partial}{\partial t_2} - H_{\mathbf{r}_2} \right) \mathcal{K}(\mathbf{r}_2, t_2; \mathbf{r}_1, t_1) = i\delta(\mathbf{r}_1 - \mathbf{r}_2)\delta(t_1 - t_2) \quad (2.5)$$

and the *propagator*  $\mathcal{K}(\mathbf{r}_2, t_2; \mathbf{r}_1, t_1)$  is expressed using the Feynman path integral

$$\mathcal{K}(\mathbf{r}_2, t_2; \mathbf{r}_1, t_1) = \int_{\substack{\mathbf{r}(t_1)=\mathbf{r}_1 \\ \mathbf{r}(t_2)=\mathbf{r}_2}} \mathcal{D}\mathbf{r}(t) \exp \left( i \int_{t_1}^{t_2} \mathcal{L}(\mathbf{r}, \dot{\mathbf{r}}) dt \right), \quad (2.6)$$

where  $\mathcal{L}$  is the classical Lagrangian function of the system

$$\mathcal{L}(\mathbf{r}, \dot{\mathbf{r}}) = \frac{1}{2} \dot{\mathbf{r}}^2 - \hat{V}(\mathbf{r}), \quad (2.7)$$

and the integral is over all paths that satisfy the endpoint conditions.

### 2.1.3 Examples of lattice models

$$\hat{\sigma}_i^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}_i \quad \hat{\sigma}_i^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}_i \quad \hat{\sigma}_i^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}_i \quad (2.8)$$

#### Transverse-field Field Ising model

$$\hat{H}_{\text{Ising}} = -J \sum_{\langle i,j \rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_i \sigma_i^x \quad (2.9)$$

#### Heisenberg model

$$\hat{H}_{\text{Heisenberg}} = -\frac{1}{2} \sum_{j=1}^N [J_x \hat{\sigma}_j^x \hat{\sigma}_{j+1}^x + J_y \hat{\sigma}_j^y \hat{\sigma}_{j+1}^y + J_z \hat{\sigma}_j^z \hat{\sigma}_{j+1}^z + h \hat{\sigma}_j^z] \quad (2.10)$$

#### Bose-Hubbard model

$$\hat{H}_{\text{BH}} = -t \sum_{\langle i,j \rangle} \hat{b}_i^\dagger \hat{b}_j + \frac{U}{2} \sum_i \hat{n}_i (\hat{n}_i - 1) - \mu \sum_i \hat{n}_i \quad (2.11)$$

## 2.2 Solutions to the many-body problem and QMC

- Mention Complexity of QMC for fermions and bosons
- A more broad overview of methods that can be used for studying the many body problem (DRMG), with particular applications to lattice methods
- Partition QMC into nicer subsections

The Schrödinger equation underpins a large part of quantum chemistry and solid state physics. However, the quantum many-body problem, which amounts to solving the  $3N$ -dimensional Schrödinger equation is notoriously hard to solve. Ever since the postulation of the equation in 1925, great efforts have been made in solving the equation, both analytically and numerically. Perhaps most impactful was the development of various approximate methods to solve the many-body problem with the available computational resources. Hartree-Fock (HF) approaches solve an auxiliary system of independent electrons in a self-consistent field and assume that the wave function (for fermions) can be represented as a single Slater determinant. The HF method does not include electron correlation, which makes it a good approximation only in systems where correlation contributions are small. Post-HF methods, such as Coupled Cluster, Configuration interaction and Møller-Plesset theory include correlation by considering a linear combination of determinants. They can be extremely accurate but come at a high computational cost.

One of the most popular approaches used today is Density Functional Theory (DFT). It reformulates the many-body electron problem in terms of the 3-dimensional electron density  $n(\mathbf{r})$ , which is found by minimising the total energy functional  $E[n(\mathbf{r})]$  [13]. DFT provides an alternative line of thought to the truncated Hilbert space of single particle orbitals [20] and is used extensively for simulating large systems as linear scaling variants of DFT exist [28]. While DFT is theoretically exact the true energy functional  $E[n(\mathbf{r})]$  is not known and its parameterisations employ more accurate *ab initio* methods. One of which being Quantum Monte Carlo (QMC).

### 2.2.1 Overview

Quantum Monte Carlo is a class of methods that uses statistical sampling to directly deal with high-dimensional integration that arises from working with the many-body wave function. QMC methods are among the most accurate achieving chemical accuracy for smaller systems [10], and can achieve any degree of statistical precision sought. Quantum Monte Carlo is also very versatile and can be applied at both zero and finite temperatures [3]. The most

basic zero temperature QMC method is variational QMC (VMC). The method is composed of two parts, firstly it directly evaluates the variational energy  $E_V = \langle \Psi_T | \hat{H} | \Psi_T \rangle / \langle \Psi_T | \Psi_T \rangle$  of the system using Monte Carlo integration and a trial wave function  $\Psi_T$ . Secondly the parameters of the trial wave function are optimised such as to minimise the variational energy  $E_V$ , giving the method its name. The first application of VMC was to ground state  ${}^4\text{He}$  [21] and was later extended for studying many-body fermionic systems [6]. A way of obtaining excitation energies using VMC is to use a trial wave function that models an excited state of the system, if the trial wave function obeys a certain symmetry, the variational principle guarantees that this VMC energy calculation gives an upper bound on the lowest exact eigenstate of this symmetry. Furthermore, the method can be extended to study non-equilibrium properties of bosonic [5, 4], and fermionic [14] systems. The main advantage of VMC is its simplicity while the main drawback is that the accuracy is limited by the flexibility and form of the trial wave function [3]. As such VMC is usually employed as a first step in more advanced QMC simulations.

Projector quantum Monte Carlo (PMC) is a class of QMC methods which are in essence nothing more than stochastic implementations of the power method to obtain the dominant eigenvector of a matrix or a kernel function [11]. Their distinct advantage over VMC is that they are not constrained by our parametrisation of the trial wave function, as they can describe arbitrary probability distributions. The projector  $\hat{P}$  has to be chosen in such a way, that the ground state of the system becomes the dominant eigenvector, i.e.  $|\Psi_0\rangle = \lim_{n \rightarrow \infty} \hat{P}^n |\Psi_T\rangle$ . Different ways of achieving this give rise to different flavours of PMC methods, e.g choice of space (real or orbital space) in which the walk is done and choosing either first or second quantisation. Using an exponential projector  $\hat{P} = e^{\tau(E_T \mathbb{1} - \hat{H})}$  can be interpreted as propagation in imaginary time  $\tau \rightarrow it$  in turn transforming the Schrödinger equation into a diffusion equation, which is a continuous limit of the random walk and lends itself to stochastic integration [26]. Directly sampling from the exact Green function is known as Projector Green Function Monte Carlo (GFMC) method [16, 17]. A convenient approximation to GFMC is its short-time approximation which leads to one of the most popular QMC methods, diffusion Monte Carlo (DMC) [10, 26]. In this regime one can exploit analytical solutions to diffusion and rate problems to write an explicit form of the Green's function. Additionally, by using the Trotter-Suzuki formula the time-step bias can be expressed and accounted for [3]. DMC is statistically implemented by using a population of walkers which either branch or die, over which the average is calculated. Reptation quantum Monte Carlo [26] (RMC) is an alternative formulation which only uses a single walker, and instead of branching and dying the MC moves mutate the path of that single walker. The use of a guiding wave function for importance sampling greatly

improves the statistical efficiency of PMC methods. The guiding wave function is usually obtained by means of VMC or some mean field calculation.

PMC methods suffer from the *sign problem*, which is present in Markov chain simulation of distributions that are not strictly positive, thus in fermionic and frustrated systems [11]. The problem refers to an exponential decrease in sampling efficiency with system size. The search for solutions of this problem is still an area of active research [10] but is in practice remedied by the *fixed-node* approximation [2]. It imposes a boundary condition into the projection, such that the projected state shares the same zero crossings (nodal surface) with a trial wave function, which is again usually obtained with VMC. The projected state is now only exact when the nodal surface is exact. Nevertheless this approximation is quite accurate [10]. Fixed node is widely used, one of its first applications being the electron gas [8], which is used in parameterisations of the exchange correlation functional in LSDA [29].

Quantum Monte Carlo methods have had a lot of success at finite temperatures. Auxiliary-field Monte Carlo, or Path Integral Monte Carlo [7], which leads to ring-polymer molecular dynamics, may be used for this purpose. Additionally QMC is not limited to continuum space applications and has been extensively used to study lattice models, notable examples being the cluster/loop algorithm and the worm algorithm [11, 25].

Quantum Monte Carlo methods are generally more computationally expensive than DFT approaches, but on the other hand QMC codes are, as a rule of thumb, simpler to implement. Furthermore, since the wave function does not need to be stored directly, QMC has reasonable storage requirements. The high computational cost of the QMC methods is remedied by the fact that they are intrinsically parallelisable, the core calculation involves generating (pseudo)-random numbers, performing a simple calculation and in the end averaging over the results. Therefore, implementations of QMC algorithms that have been applied to practical problems are optimised to run on massively parallel hardware with little overhead [23]. Finally, the repetitive nature of the Monte Carlo calculation lends itself to hardware acceleration using either graphical processing units (GPUs) or field-programmable gate arrays (FPGAs) [3].

## 2.3 Feynman-Kac: connecting Quantum Mechanics and Stochastic Processes

### 2.3.1 Stochastic processes

Introduce minimal necessary basics to understand the following section/discussions. This includes

- Random variables
- Markov processes, Markov process generator
- Wiener process
- 
- Master equation, Transition matrices and Fokker-Plank, Fokker-Planck generator?

#### Master equation

Describes the time evolution of a system which can be described as a probabilistic combination of states and the transitions between these states is encapsulated with the *transition rate matrix*.

$$\partial_t \mathbf{P} = A \mathbf{P}, \quad (2.12)$$

where  $A$  are the *connections* and  $\mathbf{P}$  are the probabilities.

### 2.3.2 Feynman-Kac formula

The Feynman path integral formulation (2.6) was extensively used by physicists for decades, even in the absence of a formal mathematical formulation which is hard to define because of the difficulties with defining an appropriate measure on the path space. Kac [15] provided a rigorous formulation of the *real-valued* case of the Feynman path integral, and the resulting *Feynman-Kac* formula provides a bridge between *parabolic* partial differential equations and stochastic processes.

To illustrate the Feynman-Kac formula let us consider a single particle with Hamiltonian

$$\hat{H} = -\frac{d^2}{dx^2} + V(x) \quad (2.13)$$

and the Schrödinger equation in *imaginary time*, which is of the elliptic type,

$$\partial_t |\psi_t\rangle = -\hat{H} |\psi_t\rangle. \quad (2.14)$$

Its formal solution, the time propagation of an initial wave function  $|\phi_0\rangle$  at  $t = 0$ , is written as

$$|\psi_t\rangle = e^{-\hat{H}t} |\phi_0\rangle. \quad (2.15)$$

From the spectral decomposition of the operator  $e^{-\hat{H}t}$  in terms of eigenstates  $|\phi_n\rangle$  and eigen-energies  $E_n$  of the Hamiltonian  $\hat{H}$

$$e^{-\hat{H}t} = \sum_n e^{-E_n t} |\phi_n\rangle \langle \phi_n|, \quad (2.16)$$

it follows that the term corresponding to the ground state of the system  $|\phi_0\rangle$  decays the slowest. Thus starting in some initial state and propagating for a long imaginary time  $it$  leads into the ground state with the decay rate giving the ground state energy as

$$\lim_{t \rightarrow \infty} |\psi_t\rangle \propto e^{-E_0 t} |\phi_0\rangle, \quad (2.17)$$

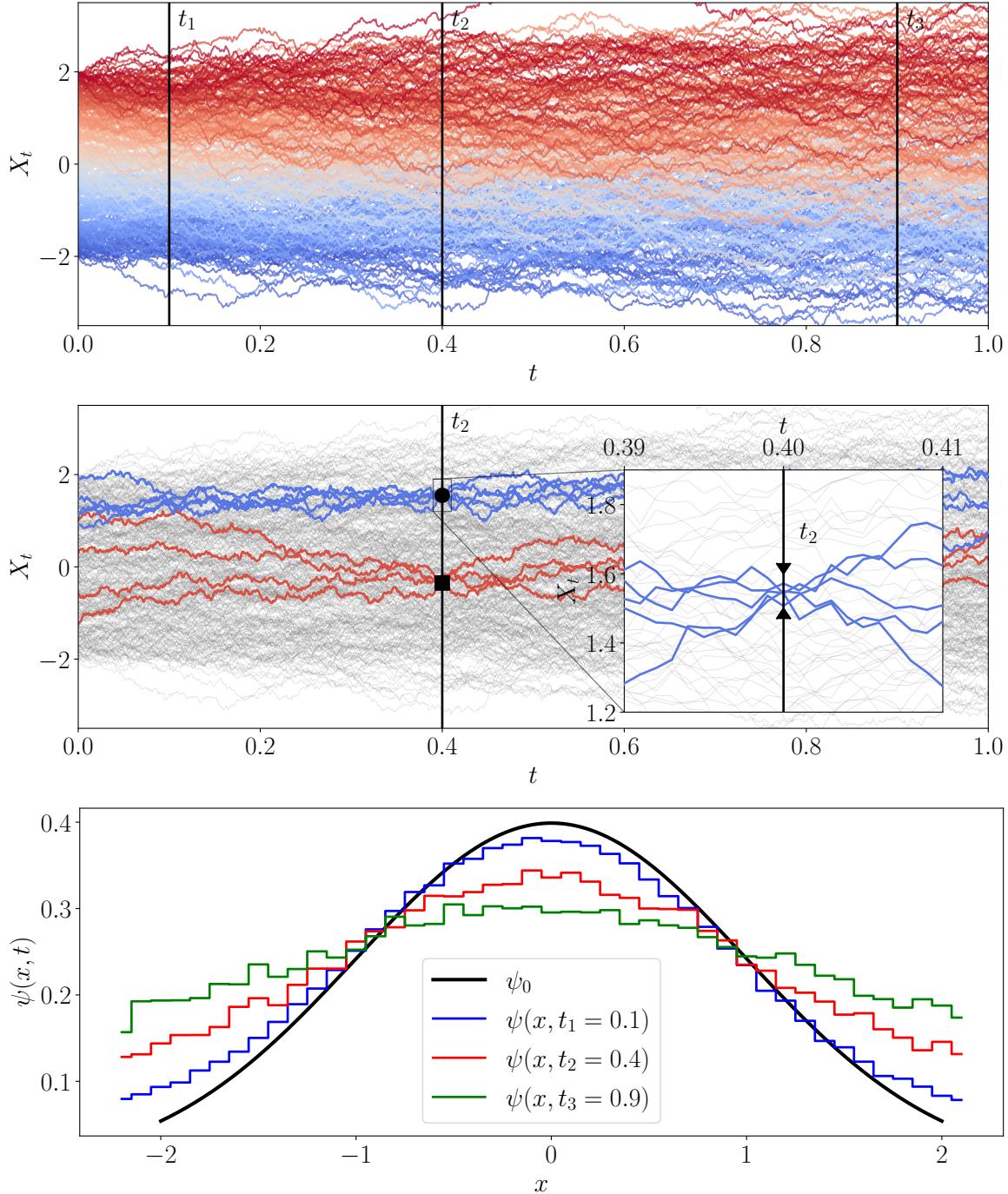
where  $E_0$  is the ground state energy and  $|\phi_0\rangle$  is the corresponding state. Kac noticed that the kinetic term of the Lagrangian in (2.6) could be interpreted as a measure on Brownian walks, and a solution to the imaginary time Schrödinger equation can be written as

$$\psi(x, t) = \mathbb{E}_{X \sim \text{Brownian with } X_t=x} \left[ \exp \left( - \int_0^t V(X_\tau, \tau) d\tau \right) \psi(X_0, 0) \right], \quad (2.18)$$

where only the **endpoint** at time  $t$  of the Brownian process fixed, whereas the starting point at time  $t = 0$  is not,  $\psi(x, 0)$  encodes the initial condition into this representation. When there is no external potential  $V(x) = 0$ , the Schrödinger equation in imaginary time is the diffusion equation and the Feynman-Kac solution is simply

$$\begin{aligned} \psi(x, t) &= \mathbb{E}_{X \sim \text{Brownian with } X_t=x} [\psi(X_0, 0)] \\ &= \frac{1}{\sqrt{2\pi t}} \int e^{-(x-x')^2/2t} \psi_0(x') dx' \end{aligned} \quad (2.19)$$

An illustration of the Feynman-Kac approach to the problem with no external potential  $V(x)$  in 1D is depicted in Fig. 2.1.

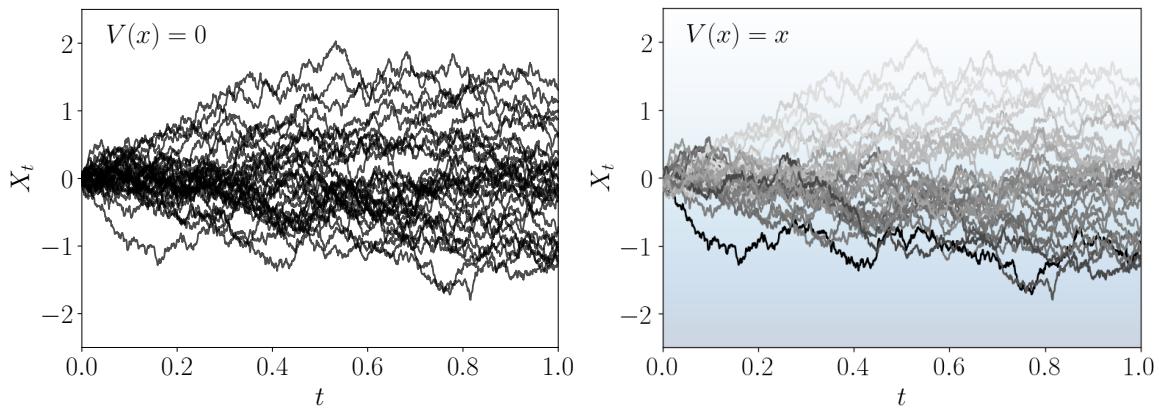


**Fig. 2.1 Feynman-Kac for a 1D free particle.** **top:**  $N = 400$  Brownian walks starting from different  $x_0$ , the color signifies initial position. In order to evaluate  $\psi$  between  $x - \frac{\delta x}{2}$  and  $x + \frac{\delta x}{2}$  at some time  $t$  we must first find Brownian paths that end there. **middle:** The paths that pass through at  $x \in (1.5, 1.6)$  (blue) and through  $x \in (-0.4, -0.3)$  (red) are colored, others are left in grey. **bottom:** Time evolution of the initial condition  $\psi_0 = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$ , by estimating  $\mathbb{E}[\psi(X_0, 0)]$  from the filtered paths at each timestep.

This simple case, does not involve the potential  $V(x)$ . The role of the potential in the Feynman-Kac formula is to weight the Brownian paths, in turn defining the Feynman-Kac *path measure*  $\mathbb{P}_{\text{FK}}$ . A path measure is simply a measure on the path space and the Feynman-Kac measure is related to the Brownian measure  $\mathbb{P}_0$  by the *Radon-Nykodym* derivative

$$\frac{d\mathbb{P}_{\text{FK}}}{d\mathbb{P}_0} = \mathcal{N} \exp\left(-\int V(X_t) dt\right), \quad (2.20)$$

where  $\mathcal{N}$  is a normalizing constant. Intuitively we can understand the measure as assigning more weight to Brownian paths that spend more time in the attractive region ( $V(x) < 0$ ) than in repulsive regions ( $V(x) > 0$ ), this is illustrated in Fig. 2.2.



**Fig. 2.2 Feynman-Kac measure in a linear potential. left:**  $N = 30$  Brownian paths. **right:** The paths colored by their likelihood under the Feynman-Kac measure with  $V(x) = x$ .

Moreover, this new stochastic process is Markovian. This is crucial for our approach because of the connection with SDEs. In the continuous case we have discussed so far, the mapping between the Fokker-Planck equation and the Schrödinger equation is straightforward in terms of a similarity transform. Starting from the FP equation with drift  $v(x) = -U'(x)$  given as a gradient of some potential function  $U(x)$  and its solution  $\rho(t, x)$

$$\frac{\partial \rho}{\partial t} = \mathcal{L}_{\text{FP}} \rho = \frac{\partial}{\partial x} \left[ \frac{\partial \rho}{\partial x} + U'(x) \rho \right], \quad (2.21)$$

we can define the function

$$\psi(x, t) = \frac{\rho(x, t)}{\sqrt{\rho_0(x)}}, \quad (2.22)$$

with  $\rho_0$  being the stationary distribution of the FP equation

$$\frac{\partial}{\partial x} \left[ \frac{\partial \rho}{\partial x} + U'(x)\rho \right] = 0 \quad \rightarrow \quad \rho_0(x) \propto \exp(-U(x)). \quad (2.23)$$

Function  $\psi(x, t)$  satisfies the imaginary time Schrödinger equation (2.14) with the Hamiltonian

$$H = -\frac{\partial^2}{\partial x^2} - \frac{U''}{2} + \frac{U'^2}{4}, \quad (2.24)$$

its ground state wavefunction is

$$\psi_0(x) = \sqrt{\rho_0(x)}. \quad (2.25)$$

In other words, the quantum ground state probability distribution  $|\psi_0|^2$  is the same as classical stationary distribution  $\rho_0$  of the stochastic process  $X_t$

$$dX_t = dW_t + v(X_t)dt, \quad (2.26)$$

in the literature referred to as the *Nelson's ground state process* [24, 1]. This connection serves as the backbone of our solution approach (revisit), as the ability to efficiently sample from the stochastic process is equivalent to sampling from the ground state of the quantum system. Even though the connection is simple, it comes with a caveat. Starting from the Schrödinger equation one needs to find the drift  $v(x)$  and while the connection with the potential of the Hamiltonian is clear-cut in 1D, this is not the case in many-body systems, i.e. the *inverse problem* of finding the stochastic process of a given Hamiltonian is difficult, and is one of the core problems approached in this thesis.

### 2.3.3 Stoquastic Hamiltonians and Lattice-model representations

A similar connection between stochastic processes and the ground state exists in the discrete space, the difference being that instead of FP we have the Master equation

For a Markov process over discrete states, we can write the master equation as

$$\frac{\partial P_j}{\partial t} = \sum_{k \neq j} [\Gamma_{k \rightarrow j} P_k - \Gamma_{j \rightarrow k} P_j]. \quad (2.27)$$

#### Single Particle on a Lattice

$$\partial_t \psi_j = \frac{1}{2} [\psi_{j+1} + \psi_{j-1} - 2\psi_j] + V_j \psi_j \quad (2.28)$$

$$\psi_j(t) = \mathbb{E}_{X \sim \text{SRW with } X_t=j} \left[ \exp \left( - \int_0^t V(X_\tau, \tau) d\tau \right) \psi_{X_0}(0) \right] \quad (2.29)$$

#### Transverse-field Ising model

#### Heisenberg model

Heisenberg ferromagnet

$$\hat{H}_F = -\frac{1}{2} \sum_j [\hat{\sigma}_j^x \hat{\sigma}_{j+1}^x + \hat{\sigma}_j^y \hat{\sigma}_{j+1}^y + \hat{\sigma}_j^z \hat{\sigma}_{j+1}^z] \quad (2.30)$$

The XY model.

$$\begin{aligned} \hat{H}_{XY} &= - \sum_j [\hat{\sigma}_j^x \hat{\sigma}_{j+1}^x + \hat{\sigma}_j^y \hat{\sigma}_{j+1}^y] = H_F + \frac{1}{2} \sum_j \hat{\sigma}_j^z \hat{\sigma}_{j+1}^z \\ &= -\mathcal{W} + \sum_j [n_j (1 - n_{j+1}) + n_{j+1} (1 - n_j)] \end{aligned} \quad (2.31)$$

$$\psi_{s_{1:N}}(t) = \mathbb{E}_{\Sigma_{[0,t]} \sim \text{SEP with } \Sigma_t = s_{1:N}} \left[ \exp \left( - \int_0^t dt' \sum_j [n_j (1 - n_{j+1}) + n_{j+1} (1 - n_j)] \right) \psi_{\Sigma_0}(0) \right] \quad (2.32)$$

#### Bose-Hubbard model

## 2.4 Quantum Mechanics, Control and loss functions

### 2.4.1 Continuous space

### 2.4.2 Discrete space

## 2.5 Machine Learning

### 2.5.1 Overview of ML approaches to the Quantum many-body problem

#### Neural Network Ansatzes

- 
- 
- 
- 

#### 2.5.2 Neural Networks

##### Convolutional Neural Networks



# Chapter 3

## Methodology

### 3.1 Monte Carlo Importance Sampling

The most common application of Monte Carlo methods is evaluation of integrals in high dimensional space. There MC has a distinct advantage over quadrature methods, as the statistical error decreases with the square root of samples irregardless of the dimensionality of the problem. Integrals of a function  $g(\mathbf{R})$

$$I = \int g(\mathbf{R})d\mathbf{R}, \quad (3.1)$$

where  $\mathbf{R}$  is the *configuration* of the system or simply a *walker*, can be integrated by use of an *importance function*  $P(\mathbf{R})$ , where  $\int d\mathbf{R}P(\mathbf{R}) = 1$  and  $P(\mathbf{R}) \geq 0$ . The integral can be rewritten in the form

$$\int g(\mathbf{R})d\mathbf{R} = \int \frac{g(\mathbf{R})}{P(\mathbf{R})}P(\mathbf{R})d\mathbf{R} = \int f(\mathbf{R})P(\mathbf{R})d\mathbf{R}, \quad (3.2)$$

where  $f(\mathbf{R}) = g(\mathbf{R})/P(\mathbf{R})$ . The importance function  $P(\mathbf{R})$  can be interpreted as a probability density. If we generate an infinite number of random uncorrelated configurations  $\mathbf{R}_m$  from the distribution  $P(\mathbf{R})$ , the sample average is a good estimator of the integral  $I$

$$I = \lim_{M \rightarrow \infty} \left\{ \frac{1}{M} \sum_{m=1}^M f(\mathbf{R}_m) \right\}, \quad (3.3)$$

and for an approximation with a finite number of samples

$$I \approx \frac{1}{M} \sum_{m=1}^M f(\mathbf{R}_m). \quad (3.4)$$

Under conditions where the central limit theorem holds [10], the estimator is normally distributed with variance  $\sigma_f^2/M$ , which can also be estimated from the samples as

$$\frac{\sigma_f^2}{M} \approx \frac{1}{M(M-1)} \sum_{m=1}^M \left[ f(\mathbf{R}_m) - \frac{1}{M} \sum_{n=1}^M f(\mathbf{R}_n) \right]^2. \quad (3.5)$$

## 3.2 Metropolis-Hastings Algorithm

The integration technique from the previous section relies on our ability to obtain samples from a probability distribution  $P(\mathbf{R})$ . In the case of QMC these distributions are high-dimensional and cannot be directly sampled from. Moreover their normalisations are usually not known. The Metropolis-Hastings algorithm [12], see Algorithm 1, avoids direct sampling from the distribution  $P(\mathbf{R})$  and is insensitive to its normalisation. It uses a Markov process whose stationary distribution  $\pi(\mathbf{R})$  is the same as  $P(\mathbf{R})$  to generate a sequence of configurations  $\{\mathbf{R}_n\}_P$  that are drawn from  $P(\mathbf{R})$ . A Markov process is completely defined with its transition probability  $P(\mathbf{R} \rightarrow \mathbf{R}')$ , which is the probability of transitioning from state  $\mathbf{R}$  to state  $\mathbf{R}'$ . For the process to have a unique stationary distribution two conditions must be met, the process must be *ergodic* and it must obey *detailed balance*

$$P(\mathbf{R})P(\mathbf{R} \rightarrow \mathbf{R}') = P(\mathbf{R}')P(\mathbf{R}' \rightarrow \mathbf{R}), \quad (3.6)$$

rewritten as

$$\frac{P(\mathbf{R})}{P(\mathbf{R}')} = \frac{P(\mathbf{R}' \rightarrow \mathbf{R})}{P(\mathbf{R} \rightarrow \mathbf{R}')}. \quad (3.7)$$

The right transition probability  $P(\mathbf{R} \rightarrow \mathbf{R}')$  is not known, but we can express it with a trial move transition probability  $T(\mathbf{R} \rightarrow \mathbf{R}')$  which we sample and acceptance probability  $A(\mathbf{R} \rightarrow \mathbf{R}')$  as

$$P(\mathbf{R} \rightarrow \mathbf{R}') = T(\mathbf{R} \rightarrow \mathbf{R}')A(\mathbf{R} \rightarrow \mathbf{R}'). \quad (3.8)$$

For equation (3.7) to hold, the acceptance probability must be

$$A(\mathbf{R} \rightarrow \mathbf{R}') = \min \left( 1, \frac{T(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}')}{T(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R})} \right). \quad (3.9)$$

Thus to sample from any probability distribution we need only have the ability to calculate probabilities  $P(\mathbf{R})$  and to sample from a trial transition probability  $T(\mathbf{R} \rightarrow \mathbf{R}')$ . The efficiency of the algorithm depends on the amount of trial moves that we reject. All trial moves would

be accepted if  $T(R \rightarrow R') = P(R')$ , which would just mean sampling from  $P$  directly and is the very problem we are trying to solve with Metropolis-Hastings.

---

**Algorithm 1:** Metropolis-Hastings

---

**Result:** A set of configurations  $\{R_n\}_P$  sampled from  $P$

Initialize walker at random configuration  $R$ ;

**while** no. samples less than  $N$  **do**

- Generate new configuration  $R'$  with transition probability  $T(R \rightarrow R')$ ;
- Accept the move  $(R \rightarrow R')$  with probability
- $A(R \rightarrow R') = \min\left(1, \frac{T(R' \rightarrow R)P(R)}{T(R \rightarrow R')P(R')}\right)$ ;
- Append  $R$  to the set of configuration;

**end**

---

## 3.3 Gradient based optimisation

### 3.3.1 Gradient estimation

- **TODO:** Rewrite with a more general tone, do not only talk about the ELBO. Still use Mohammeds review! It is very good.

In order to perform gradient descent on the ELBO objective, we need to be able to evaluate its gradients with respect to parameters  $\theta$  and  $\phi$ . Taking the gradient w.r.t generative parameters  $\theta$  is straightforward, because we can change the order of the expectation operator and the gradient, leaving us with

$$\begin{aligned} \nabla_{\theta} \mathcal{L}_{\theta, \phi}(x) &= \nabla_{\theta} \mathbb{E}_{q_{\phi}(z|x)} [\log p_{\theta}(x, z) - \log q_{\phi}(z|x)] \\ &\simeq \nabla_{\theta} (\log p_{\theta}(x, z) - \log q_{\phi}(z|x)) \\ &= \nabla_{\theta} (\log p_{\theta}(x, z)), \end{aligned} \tag{3.10}$$

where  $\simeq$  denotes an unbiased estimator. This reversing of the order of operations is not possible when taking gradients w.r.t variational parameters  $\phi$  because the expectation  $\mathbb{E}_{q_{\phi}(z|x)}$  is performed w.r.t the approximate posterior  $q_{\phi}(z|x)$ . The gradient could be estimated with a vanilla Monte Carlo estimator, but it has very high variance and is not practical [19].

The problem of stochastic gradient estimation of an expectation of a function is a well studied problem that transcends machine learning and has a variety of applications [9, 27]. Different estimators differ in from and their properties, variance being one of the most important. In their review [22] Mohamed et al. categorise MC gradient estimators into three categories

### Score-function estimator

*Score-function estimator:* The score function is a logarithm of a probability distribution w.r.t to distributional parameters. It can be used as a gradient estimator

$$\begin{aligned}\nabla_{\theta} \mathbb{E}_{p_{\theta}(\mathbf{x})}[f(\mathbf{x})] &= \nabla_{\theta} \int p_{\theta}(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} \\ &= \mathbb{E}_{p_{\theta}(\mathbf{x})}[f(\mathbf{x}) \nabla_{\theta} \log p_{\theta}(\mathbf{x})].\end{aligned}\tag{3.11}$$

The score-function estimator is compatible with any cost function, it requires that the measure  $p_{\theta}(\mathbf{x})$  is differentiable and easy to sample. Very importantly it is applicable to both discrete and continuous distribution, but has a drawback of having high variance.

### Pathwise estimator

Continuous distributions can be sampled either directly by generating samples from the distribution  $p_{\theta}(\mathbf{x})$  or indirectly, by sampling from a simpler base distribution  $p(\epsilon)$  and transforming the variate through a deterministic path  $g_{\theta}(\epsilon)$ . Using this, it is possible to move the source of randomness in such a way that the objective is differentiable. In essence this approach pushes the parameters of the measure into the cost function which is then differentiated. The estimator is

$$\begin{aligned}\nabla_{\theta} \mathbb{E}_{p_{\theta}(\mathbf{x})}[f(\mathbf{x})] &= \nabla_{\theta} \int p_{\theta}(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} \\ &= \nabla_{\theta} \int p(\epsilon) f(g_{\theta}(\epsilon)) d\epsilon \\ &= \mathbb{E}_{p(\epsilon)}[\nabla_{\theta} f(g_{\theta}(\epsilon))].\end{aligned}\tag{3.12}$$

This was the gradient estimator originally used in the VAE implementation [19] there named as the *reparametrization trick*, see also Figure 3.1. In many cases the transformation paths are so simple they can be implemented in one line of code, referred to as *one-liners*. The pathwise-estimator can only be used on differentiable cost functions, but is easy to implement and crucially has lower variance than the score-function estimator.

### Measure-valued gradient estimator

Which exploits the properties of signed-measures, is beyond the scope of this report.

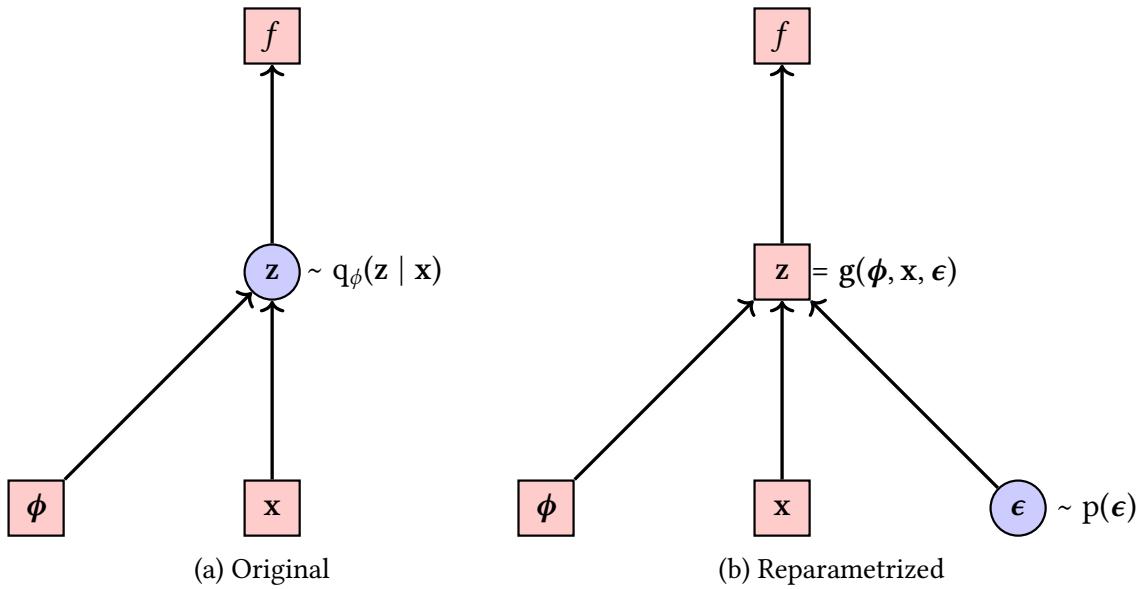


Fig. 3.1 The reparametrization trick, adapted from [18]. The stochasticity of the  $z$  node is pushed out into a separate input to the same node, resulting in deterministic gradients w.r.t  $\phi$  through the node.

### 3.3.2 Automatic differentiation

## 3.4 Optimal sampling: optimal sampling in lattice models



# **Chapter 4**

## **Results**

**4.1 Transverse Field Ising model**

**4.2 Heisenberg model**

**4.3 Bose-Hubbard model**



# **Chapter 5**

## **Conclusion**

### **5.1 Direction for future work**

### **5.2 Remarks**



# References

- [1] Albeverio, S., Ho/egh-Krohn, R., and Streit, L. (1977). Energy forms, hamiltonians, and distorted brownian paths. *Journal of Mathematical Physics*, 18(5):907–917.
- [2] Anderson, J. B. (1975). A random-walk simulation of the schrödinger equation: H+ 3. *The Journal of Chemical Physics*, 63(4):1499–1503.
- [3] Austin, B. M., Zubarev, D. Y., and Lester Jr, W. A. (2012). Quantum monte carlo and related approaches. *Chemical reviews*, 112(1):263–288.
- [4] Carleo, G., Becca, F., Sanchez-Palencia, L., Sorella, S., and Fabrizio, M. (2014). Light-cone effect and supersonic correlations in one-and two-dimensional bosonic superfluids. *Physical Review A*, 89(3):031602.
- [5] Carleo, G., Becca, F., Schiró, M., and Fabrizio, M. (2012). Localization and glassy dynamics of many-body quantum systems. *Scientific reports*, 2(1):1–6.
- [6] Ceperley, D., Chester, G. V., and Kalos, M. H. (1977). Monte carlo simulation of a many-fermion study. *Physical Review B*, 16(7):3081.
- [7] Ceperley, D. M. (1995). Path integrals in the theory of condensed helium. *Reviews of Modern Physics*, 67(2):279.
- [8] Ceperley, D. M. and Alder, B. J. (1980). Ground state of the electron gas by a stochastic method. *Physical review letters*, 45(7):566.
- [9] Chriss, N. A. and Chriss, N. (1997). *Black Scholes and beyond: option pricing models*. McGraw-Hill.
- [10] Foulkes, W., Mitas, L., Needs, R., and Rajagopal, G. (2001). Quantum monte carlo simulations of solids. *Reviews of Modern Physics*, 73(1):33.
- [11] Gubernatis, J., Kawashima, N., and Werner, P. (2016). *Quantum Monte Carlo Methods: Algorithms for Lattice Models*. Cambridge University Press.
- [12] Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*.
- [13] Hohenberg, P. and Kohn, W. (1964). Inhomogeneous electron gas. *Physical review*, 136(3B):B864.

- [14] Ido, K., Ohgoe, T., and Imada, M. (2015). Time-dependent many-variable variational monte carlo method for nonequilibrium strongly correlated electron systems. *Physical Review B*, 92(24):245106.
- [15] Kac, M. (1949). On distributions of certain wiener functionals. *Transactions of the American Mathematical Society*, 65(1):1–13.
- [16] Kalos, M. (1962). Monte carlo calculations of the ground state of three-and four-body nuclei. *Physical Review*, 128(4):1791.
- [17] Kalos, M. (1966). Stochastic wave function for atomic helium. *Journal of Computational Physics*, 1(2):257–276.
- [18] Kingma, D. P. (2017). Variational inference & deep learning: A new synthesis.
- [19] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- [20] Kohn, W. (1999). Nobel lecture: Electronic structure of matter—wave functions and density functionals. *Reviews of Modern Physics*, 71(5):1253.
- [21] McMillan, W. L. (1965). Ground state of liquid he 4. *Physical Review*, 138(2A):A442.
- [22] Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. (2020). Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62.
- [23] Needs, R., Towler, M., Drummond, N., Lopez Rios, P., and Trail, J. (2020). Variational and diffusion quantum monte carlo calculations with the casino code. *The Journal of chemical physics*, 152(15):154106.
- [24] Nelson, E. (1967). Dynamical theories of brownian motion, princeton univ. *Press, Princeton, NJ*.
- [25] Prokof'Ev, N., Svistunov, B., and Tupitsyn, I. (1998). Exact, complete, and universal continuous-time worldline monte carlo approach to the statistics of discrete quantum systems. *Journal of Experimental and Theoretical Physics*, 87(2):310–321.
- [26] Reynolds, P. J., Tobochnik, J., and Gould, H. (1990). Diffusion quantum monte carlo. *Computers in Physics*, 4(6):662–668.
- [27] Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., et al. (2020). Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609.
- [28] Skylaris, C.-K., Haynes, P. D., Mostofi, A. A., and Payne, M. C. (2005). Introducing onetep: Linear-scaling density functional simulations on parallel computers. *The Journal of chemical physics*, 122(8):084119.
- [29] Vosko, S. H., Wilk, L., and Nusair, M. (1980). Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis. *Canadian Journal of physics*, 58(8):1200–1211.

## **Appendix A**

### **Jax ecosystem**



## **Appendix B**

### **Some additional results**



## **Appendix C**

### **Probability distributions**

