Algorithms for Near-Term and Noisy Quantum Devices

Andrew Patterson

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Physics and Astronomy

University College London

April 21, 2023

I, Andrew Patterson, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

Quantum computing promises to revolutionise many fields, including chemical simulations and machine learning. At the present moment those promises have not been realised, due to the large resource requirements of fault tolerant quantum computers, not excepting the scientific and engineering challenges to building a fault tolerant quantum computer. Instead, we currently have access to quantum devices that are both limited in qubit number, and have noisy qubits.

This thesis deals with the challenges that these devices present, by investigating applications in quantum simulation for molecules and solid state systems, quantum machine learning, and by presenting a detailed simulation of a real ion trap device. We firstly build on a previous algorithm for state discrimination using a quantum machine learning model, and we show how to adapt the algorithm to work on a noisy device. This algorithm outperforms the analytical best POVM if ran on a noisy device. We then discuss how to build a quantum perceptron - the building block of a quantum neural network.

We also present an algorithm for simulating the Dynamical Mean Field Theory (DMFT) using a quantum device, for two sites. We also discuss some of the difficulties found in scaling up that system, and present an algorithm for building the DMFT ansatz using the quantum device. We also discuss modifications to the algorithm that make it more 'device-aware'.

Finally we present a pule-level simulation of the noise in an ion trap device, designed to match the specifications of a device at the National Physical Laboratory (NPL), which we can use to direct future experimental focus. Each of these sections is preceded by a review of the relevant literature.

Impact Statement

Quantum computing has the potential to have a revolutionary impact in some fields, via the exponential speedup of some quantum algorithms. Exponential speedups are usually reserved only for algorithms ran on fault tolerant devices, with error correction encoding and decoding. However, at the time of writing there is yet to be a large enough demonstration of a fault-tolerant quantum computer that can outperform classical computing.

The work in this thesis is concerned with the quantum devices that are available now, or will be in the short-term. These devices are limited in scale, and are noisy. This thesis works within both of these constraints to present methods for current quantum devices in the fields of chemistry and quantum machine learning.

Current quantum devices are being developed within and without academia, and this work could aid either group towards a demonstration of a quantum algorithm, ran on a near term device, that can complete a problem better that current classical computers. Within academia this would be a significant result, demonstrating the use of noisy quantum devices, outside of the initial problem. In industry, demonstration of a near-term quantum advantage would spur a lot more interest in quantum computing, especially in the current 'valley of uncertainty' we are in.

We expect the work presented here to be of interest to researchers in academia and industry who are working towards a demonstration of quantum advantage on a near-term device.

Acknowledgements

I would like to begin by thanking Ivan Rungger, my supervisor, for his support throughout the whole PhD, and for providing guidance and patience when I felt most stuck. I would also like to thank Dan Browne, who also helped supervise, and provided support and advice, with knowledge of so much of this area. Also, to the members of both Ivan and Dan's groups, who have challenged me and introduced so many new ideas. I cannot name everyone in each group, but I would like to mention Abishek Argawal, Asmae Benhemou, Adam Callison, Enrico Fontana, Oscar Higgot, Francois Jamet, Deep Lall, Alessandra Lingarolo, Avinash Mocheria, Artur Pesah, and James Seddon. You meet a lot of people if you attend two group meetings and stay for four years! Also, without all the members of the UCL Center for Doctoral Training cohort 4 I would not know half as much as I do know, and would have had double the stress, especially in the masters year. I would especially like to thank Lingling Lao, who has given so much support to me in the ADAPT project, even remotely from a completely different time zone!

I am also grateful for those who I worked with outside of the PhD, who took me on for internships and where I learned so much from them. This includes Ashley Montanaro, Steve Brierly and James Cruise, and Alastair Sinclair, and all the others I worked with.

I would like to personally thank my family and friends who helped me stay sane before and during the PhD. I would like to thank my mother, Maria Niland, for always believing in me and pushing me to go for the things I never thought I could get. I would also like to remember my grandfathers, who probably planted the seeds of my interest in physics in their own different ways. I would also like to remember

my father who I'm sure would have liked to see this thesis. I also need to mention David Southall and Eoin Bentick, who have always been interested in what I've been doing, even though is is so far from their interests.

Finally I need to give a million thanks to my fiancée, Juliet Eales, who has been a rock of support throughout my whole PhD. You have graciously endured so many complaints of 'the code isn't working', and provided encouragement and optimism anyway. You have also helped me in so many ways personally that I am a better person for today, and I thank you.

Contents

List of Figures

List of Tables

Part I

Introduction

Chapter 1

Introduction

Quantum computing has its roots in the algorithms developed in the 1990s and early 2000s, such as Shor's algorithm for period finding [1], Grover's search algorithm [2], and the Deutcsh-Joza algorithm [3]. These algorithms set out the high promise of quantum computing, that we can manipulate quantum states of matter to do computations exponentially faster than classical computers. One of the earliest discussions of the concept however is Feynman's statement that nature is not classical, and to model it we must use something not classical, e.g. a quantum computer [4] . Since the early days of quantum information science, when exponential speedups were found for many algorithms and it seemed that cryptography would soon be under threat, the field has made so many developments, but we still seem to be far from the world of exponential speedups and perfect simulations of nature that were first envisaged. This is due mainly to how *hard* it is to both manipulate a quantum state, and to keep it from interacting with its environment in such a way that the information in the state is preserved. In this thesis we will call any process which erases the information stored in a quantum state noise. Sources of noise can include unwanted interactions with molecules adsorbed to the surface of a device [5]–[8], electrical noise in the experimental apparatus [9], calibration drift [10]–[13] , actual loss of the qubits [14] , and even interactions with cosmic rays [15], [16]. However, we will for the time being treat all the noise universally, as decoherence.

It is noise in the computation, along with scaling challenges which prompted John Preskill to name the current era of quantum computing Noisy Intermediate

Scale Quantum (NISQ) [17]. In the NISQ era what we can achieve with quantum devices is limited by the low numbers of qubits, and the noise in the qubits. The noise in the qubits prohibits us from preforming Shor's factoring algorithm [1] as it requires the phase of qubits to interfere with high precision, such that noise destroys useful interference in these algorithms. Current devices have a modest qubit count, with IBM currently claiming to have 127 qubits in use, the highest count at time of writing[1] [18]. This qubit count limits the systems we can perform quantum simulations of, meaning there is yet to be a demonstration of a quantum simulation of a system that cannot be simulated approximately on a classical computer.

For reference, one of the 'holy grail' systems we wish to simulate is FeMoCo, a naturally occurring catalyst that helps to synthesise ammonia from nitrogen [27]. We wish to replicate the action of FeMoCo, as it can preform this task in the low energy environment of plants, whereas synthetic ammonia production requires the Haber process [28] involving temperatures around 500°C and pressures 100X atmospheric pressure. Currently nitrogen production requires around 3% of the world's total energy use [29] as it is required in fertilizer, simulation of the FeMoCo molecule as a step to recreating its action is a noble goal for quantum simulation. However, even ignoring the problem of noise, we would still require around 2024 qubits [30] for this simulation. When we include noise, we introduce the need for fault-tolerant algorithms and therefore error correction, we can estimate the total number of physical qubits needed following the estimation made in [31], of 1568 physical qubits per logical qubit, i.e. $1568 \times 2024 = 3.17 \times 10^6$ physical qubits.

[1] Qubit counts checked on 06/07/2022. There are metrics other than qubit count proposed, e.g. quantum volume [19], and algorithmic qubits [20] which also take into account the noise on the device. One issue with these is due to Goodhart's law, i.e. when a metric becomes a target, it ceases to be a good metric. For example a 10 qubit system currently holds the highest quantum volume record [21], which can easily be simulated by a classical device; whereas there are also devices which it is claimed cannot be simulated classically available [22], [23]. However, tensor network techniques [24] have improved the classical simulation 10,000 X, requiring approximately 14 more qubits to overcome, but this illustrates that whilst a 10 qubit device may hold the quantum volume record, there are existing devices which are much better in a different, also meaningful metric. Our final point in this increasingly long footnote is that connectivity is taken into account in the Quantum Volume metric, with the device in [21] claiming all-to-all qubit connectivity, due to the ability of the device to shuttle all the ions on the device. However, this is a different concept from the all-to-all connectivity of the IONQ ion trap device [25], as there is run-time required to perform the shuttling operation, and it is a noisy operation [14], [26], but it is treat the same way by the Quantum Volume metric.

We are clearly far from achieving a simulation of FeMoCo. Currently, one of the questions posed is if a current or near-term quantum device can demonstrate a useful algorithm that a classical device cannot. Quantum supremacy [32]–[38] was claimed to be achieved in 2019 [22], however new methods using tensor network simulations [24] can simulate these circuits with a 10,000X speed-up over the original simulation algorithm. A rough calculation, of $\lceil \log_2(10000) \rceil = 14$, suggests that a modest increase in the number of qubits of the Sycamore device could again claim quantum supremacy. However, there is little practical use for quantum supremacy experiments, outside of their own demonstration and as a verifiable source of random numbers [39]; we wish to find an algorithm for NISQ devices that is both of practical interest and out of reach of current classical computers. We will name this goal as quantum advantage, to distinguish it from quantum supremacy.

In this thesis we do not achieve quantum advantage, although we do make some progress towards this goal. We will see in Part II how NISQ devices can be used in machine learning problems, with results in Chapter 6 showing that a noisy quantum device can learn parameters which can overcome some of that noise. We will also discuss proposals for a quantum version of the perceptron [40], and how non-linearities can be introduced into Quantum Machine Learning (QML) algorithms. In Part III we will discuss a promising route towards quantum advantage, and Feynman's stated goal, simulating chemical systems. We will demonstrate an algorithm for simulating Dynamical Mean Field Theory (DMFT) in Chapter 11, and in Chapter 12 we will extend the size of the simulation and build on one of the most promising NISQ simulation algorithms, ADAPT [41], [42]. Finally in Part IV we will discuss some of the techniques used to algorithmically mitigate the noise in quantum devices, and we will implement a more in-depth simulation of a specific quantum device (Chapter 14), including some of the sources of noise mentioned at the beginning of this section. For the remainder of this introductory part, we will introduce some of the quantum computing concepts required later.

Chapter 2

Quantum Computing Primer

2.1 Gate Model of Quantum Computation

In this thesis we will be discussing the circuit (gate) model of quantum computation. A more in-depth introduction to the quantum computing topics here can be found in Ref. [43] Other models, e.g. adiabatic computation [44], measurement based computation [45], and analogue quantum simulation [46] are not discussed here. A qubit is the name for a quantum mechanical two-level system. Qubits can take many forms physically, e.g. states of an ion [47], direction of magnetic flux, or charge in a superconductor [48], the polarization of light [49], or the Fock state of photons [50]. We will describe all these types of qubits by a vector, representing the two measurement outcomes in the computational, Z, basis,

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{2.1}$$

Quantum mechanics is described by unitary operators, $U^\dagger U = \mathbb{1}$, and in the circuit model, we apply these operators, called gates, sequentially to the qubits. Some of the most common gates are named the Pauli gates, X, Y, Z, or denoted $\sigma^x, \sigma^y, \sigma^z$ which represent the flipping of the state vector around some axis on the Bloch sphere, Figure 2.1.

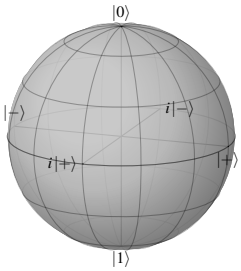


Figure 2.1: The Bloch sphere, which represents a single qubit. The state of a qubit is represented by a vector, and the poles represent the X,Y,Z axes. Pure states have the vector ending on the surface of the sphere, whereas mixed states the vector is closer to the origin. The maximally mixed state is a vector of 0 length centred at the origin.

2.2 Clifford Gates

In matrix form, the Pauli gates are

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.2}$$

We will also mention the identity operation here, denoted I or $\mathbb{1}$, which has no effect on the state, and for the $d = 2$ system is:

$$\mathbb{1}_2 = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{2.3}$$

Combining the Pauli gates with the CNOT operation,

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{2.4}$$

and the Hadamard gate,

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{2.5}$$

gives us the Clifford group [51]. This obeys group properties, i.e. closure, identity,

associativity, inverse, when the operation is matrix multiplication. The Pauli matrices themselves obey a cyclic commutativity,

$$\sigma^j \sigma^k = \delta_{jk} \mathbb{1} + i \varepsilon_{jkl} \sigma_l, \tag{2.6}$$

where δ_{jk} is the Kronecker delta function and ε_{jkl} is the Levi-Cita symbol[1].

We can represent these operations as gates in a circuit, with wires as qubits, and a measurement at the end, e.g. Figure 2.2 which uses only the gates discussed so far. Circuits containing only Clifford gates are not universal for quantum computation, in

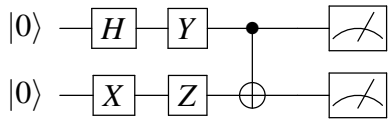


Figure 2.2: A simple quantum circuit to introduce the notation used here, so far only using Clifford group operations.

fact, Clifford circuits can be simulated on a classical computer [52], a theorem known as the Gottesman-Knill theorem. Arbitrary rotations around the axes of the Bloch sphere are not Clifford operations, and are used mostly in NISQ algorithms, when we move to fault tolerant computation, arbitrary rotations are estimated by T gates (below), and Toffoli (CCNOT) gates are used between qubits. The introduction of these gates makes the computation non-Clifford, and hard to simulate on a classical device. We have introduced the Clifford group via the Pauli matrices as they will be used later, however we note here that the generators of the Clifford group are the H, CNOT and

$$S = \sqrt{Z} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \tag{2.7}$$

gate. We can see this as $SS^\dagger = Z$ and Z gate, $HZH^\dagger = X$, allowing us to create all three Pauli gates. Also note that the gates of the Clifford group are easy to do in a fault-tolerant setting [51] and it is the implementation of the non-Clifford gates which is difficult in fault-tolerant quantum computation. The non-Clifford gate that

[1] $\varepsilon_{jkl} = +1$ if j,k are in canonical order, e.g. $j = x, k = y, l = z$ or $j = z, k = x, l = y$, $\varepsilon_{jkl} = -1$ if they are out of order, and $\varepsilon_{jkl} = 0$ if $j = k$.

is most used in fault-tolerant quantum computation is the

$$T = \sqrt{S} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix} \tag{2.8}$$

gate. We can see by $TXT^\dagger \notin$ Clifford that the T gate is not a Clifford gate. Adding the T gate to the Clifford group creates a set of gates which are universal for quantum computation.

However, due to the nature of NISQ technology we are easily able to apply parameterised Pauli rotation gates. In many of the implementations, notable ion traps and superconducting devices, we apply an electromagnetic pulse to the qubit to enact a gate. Applying the pulse for the total time t enacts the full Pauli gate, whereas applying it for a fractional time allows us to apply the rotation gate $\mathrm{R}_{\mathcal{P}}(\theta)$. The parametrisation of this gate allows us to enact many of the variational algorithms we will see later. Interestingly, applying these parameterised gates with high precision is harder for fault-tolerant devices, as the gates in use are completely discretized, meaning many applications of a partial rotation gate, such as the T gate may be required for some rotations.

2.3 Measurement

Measurement in quantum mechanics stands apart from the everything we have discussed so far, as a measurement operation is non-unitary, unlike quantum evolution. Given a measurement operator M_m and a measurement outcome m, and a general quantum state $|\psi\rangle$, the probability of measuring m is given by:

$$p(m) = \langle\psi| M_m^\dagger M_m |\psi\rangle , \tag{2.9}$$

the state of the system immediately after measurement is

$$\frac{M_m |\psi\rangle}{\sqrt{\langle\psi| M_m^\dagger M_m |\psi\rangle}}, \tag{2.10}$$

and the measurement operators obey the completeness relation, i.e. the probabilities sum to 1:

$$\sum_m M_m^\dagger M_m = I. \tag{2.11}$$

This formulation of measurement was first proposed by Max Born [53], and is sometimes called the Born rule. We will discuss an implementation of this rule in a quantum machine learning model in Chapter 5.

Measurements we will discuss here are projective, that is an observable M can be described by a spectral decomposition,

$$M = \sum_m m P_m, \tag{2.12}$$

where P_m is the projector onto the eigenspace of M with eigenvalue m. Projective measurement operators satisfy the completeness relation, like general measurement operators,

$$\sum_m P_m^\dagger P_m = I. \tag{2.13}$$

Additionally, projective measurements are Hermitian, $P_m^\dagger = P_m$, and orthogonal, $P_m P_{m'} = \delta_{m,m'} M_m$.

A Positive Operator-Valued Measure (POVM) describes a general set of operators, $\{E_m\}$ which are positive and satisfy $\sum_m E_m = I$. For a general POVM, we can define a set of measurement operators describing the POVM, $M_m = \sqrt{E_m}$ which satisfies the completeness relation, $\sum_m M_m^\dagger M_m = I$. We will use POVMs in Chapter 6, but we can briefly see how they can be useful. Given one of two states, $|\psi_1\rangle = |0\rangle, |\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, it is impossible to perfectly distinguish them, as they are not orthogonal. However, we can design a POVM that can not make an

error, but leave us with an indeterminate result. The POVM with elements

$$E_1 = \frac{\sqrt{2}}{2(1+\sqrt{2})}\left(|0\rangle - |1\rangle\right)\left(\langle 0| - \langle 1|\right), \tag{2.14}$$

$$E_2 = \frac{\sqrt{2}}{1+\sqrt{2}}|1\rangle\langle 1|, \tag{2.15}$$

$$E_3 = I - E_1 - E_2. \tag{2.16}$$

will never be incorrect with a measurement of $E_1 \implies |\psi_1\rangle, E_2 \implies |\psi_2\rangle$ and a measurement of E_3 is indeterminate. We have sacrificed some determinacy so that we never make an error.

2.3.1 Amplitude

We can describe the general single qubit state as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{2.17}$$

with $\sqrt{\alpha\alpha^*} + \sqrt{\beta\beta^*} = 1$, the absolute values of α, β printed this way to make explicit that they are complex numbers. Given the measurement operator $M_0 = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$, from Equation 2.9 that the probability of measuring $|0\rangle$ is given by $|\alpha|^2$. For this reason we describe α, β as probability amplitudes[2], as the absolute values sum to 1 (the state is normalised) and they describe the probability of measuring a component of a state.

2.3.2 Phase

A complex number can, via Euler's formula, be described by $z = re^{i\phi}$. For these purposes, we are restricted to the numbers where $\max|z|^2 = 1$, allowing us to set $r = 1$. We call ϕ the *phase* of z. Now consider the effect of $e^{i\phi}|\psi\rangle$ on all measurement operators M_m; as we have $p(m) = \langle\psi|e^{-i\phi}M_m^\dagger M_m e^{i\phi}|\psi\rangle = \langle\psi|M_m^\dagger M_m|\psi\rangle$, e.g. the phase has no effect on the measurement operators. We name this global phase invariance, meaning we can ignore any phases multiplied to the whole quantum

[2]Note that α, β themselves are not probabilities, as they can be negative or imaginary.

state. Conversely, when the phase of the components of a phase differ we see some effect. If we consider the states $|\psi_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, we may at first say that the probability of measuring $P(|0\rangle) = P(|1\rangle) = \frac{1}{2}$, so there is still no difference. However this misses that $|\psi_2\rangle$ interacts differently with quantum operations. Now consider applying a Hadamard gate to $|\psi_{1,2}\rangle$ and measuring, i.e. the circuit in Figure 2.3.

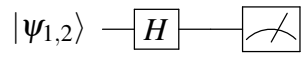


Figure 2.3: The measurement of $|+\rangle, |-\rangle$ is different after being passed through this circuit, illustrating the importance of relative phase.

The result is $|\psi_1\rangle \implies |0\rangle, |\psi_2\rangle \implies |1\rangle$, showing how the relative phase of a state can affect the outcome.

2.4 Quantum Fourier Transform

As we have just discussed the difference between phase and amplitude in a quantum state, we will take this opportunity to discuss in detail a quantum algorithm, the Quantum Fourier Transform [1], [54] (QFT). Similar to the classical discrete Fourier transform, the QFT takes a quantum state defined by $|x\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$ and outputs the state $|y\rangle = \sum_{k=0}^{N-1} y_k |k\rangle$ where y_k is the discrete Fourier transform of x_j. We can describe the QFT by its action on the basis state:

$$|j\rangle \to \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle . \tag{2.18}$$

Representing a number j in binary, $j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0$, we can describe the Fourier transform as:

$$\begin{aligned} |j_1, \ldots j_n\rangle = &\frac{1}{2^{n/2}} \big(\big(|0\rangle + e^{2\pi i 0.j_n} |1\rangle\big) \big(|0\rangle + e^{2\pi i 0.j_{n-1} j_n} |1\rangle\big) \\ &\ldots \big(|0\rangle + e^{2\pi i 0.j_1 j_2 \ldots j_n} |1\rangle\big) \big), \end{aligned} \tag{2.19}$$

where we represent the n digits of a decimal number as $0.j_1 j_2 \ldots j_n$. As QFT is a unitary operation, it can be performed on quantum computers, and offers an

exponential speedup, $\mathcal{O}(n^2)$ gates, over the classical Fast Fourier Transform, $\mathcal{O}(n2^n)$ gates.

2.5 Computational Complexity

Apart from the scientific curiosity of using quantum states for computation, much quantum computation research is an attempt to find a quantum speedup, i.e. a problem that can be solved on a quantum computer quicker than a classical computer. When we look for these problems, we want to classify the resource requirements of the quantum algorithm and compare it with the requirements of the best classical algorithm.

In the previous section, we discuss an exponential speedup, where the number of gates used to calculate the quantum Fourier transform is on the order of $\mathcal{O}(n^2)$, and the classical algorithm requires $\mathcal{O}(n2^n)$, where n is the number of qubits or bits representing the number. We consider the requirements asymptotically, i.e. when $n \to \infty$, and $\mathcal{O}(\cdot)$ represents the upper bound of the scaling. For some function $f(x)$ we say that $f(x) \in \mathcal{O}(g(x))$ if there exits some real number x_0 and positive real number M such that:

$$|f(x)| \leq Mg(x_0) \;\; \forall x \geq x_0. \tag{2.20}$$

This notation also focusses upon the algorithm, ignoring the factors arising from the implementation on a particular device such as SWAP gates required when qubits are disconnected.

One of the first algorithms to theoretically prove quantum speedups is Shor's algorithm for factoring [1]. The algorithm gives an exponential speedup in finding the prime factors of a prime number, a problem that one of the most widely used cryptographic algorithms relies upon being hard to work. It is also a good example of an NP problem, a complexity class of problems that are hard for classical computers, a solution is hard to find, but easy to verify it is correct. For example, given a large number $x = ab$, where a and b are prime it is hard to find a and b with no knowledge, but very easy to verify that $ab = x$.. However, the factoring problem is *not* NP-complete[3], meaning other problems in NP cannot be represented as an imple-

mentation of the factoring problem. footnotetextAssuming P $\neq$ NP. It is still hard for classical computers, the number field sieve, i.e. one of the best classical attempts to solve the factoring problem has complexity [55] $\mathcal{O}\left(\exp\left((\ln n)^{\frac{1}{3}}(\ln\ln n)^{\frac{2}{3}}\right)\right)$. It is this proven speedup that gives impetus for us to find other problems that can be solved more efficiently with a quantum computer.

Chapter 3

Noise Models

Noise in quantum circuits can be divided into two distinct categories: State Preparation And Measurement (SPAM) errors, and decoherence of the quantum state stored in the qubit [1]. There is much effort directed to reducing errors in experimental quantum computing, but in this project we will only consider theoretical or algorithmic error reduction / mitigation.

3.1 State Preparation and Measurement Errors

In many implementations SPAM errors are the most prevalent, e.g. in [22] they report an error rate of 0.93 % for two qubit gates performed simultaneously on all qubits averaged over a number of cycles (which includes state preparation errors), and a readout error of 3.8 % when measured on all qubits. Similarly but in a different technology, in an ion trap the SPAM error was three orders of magnitude higher than the two qubit gate error [58]. SPAM errors are grouped together as it is harder to distingush between the two, an incorrectly prepared state has the same effect as an erroneous measurement.

Strategies for mitigating SPAM errors first require that the errors are characterised, as the majority of implementations [59]–[63] first do some form of tomography on the SPAM errors, to get the transition matrix, T (which describes the act of SPAM errors on the identity circuit) and use that to reduce the effect of these errors

[1] There is also the phenomena of leakage in many qubit implementations, where the quantum state 'leaks' into a state that is not $|0\rangle$ or $|1\rangle$ (i.e. $|2\rangle$) [56] , but this can be modelled as decoherence, so will be treated as such here. It may be easy to treat it as decoherence in modelling, but becomes harder to correct than decoherence in error correcting codes [57].

on the final outcome, by applying $T^{\dagger}$. Characterising the errors is the most resource intensive step, as full tomography of the errors requires 2^n extra measurements [64] , this is why many of the efforts to reduce SPAM errors focus upon estimating T with high fidelity.

SPAM errors categorised this way can be split into two distinct sets: intrinsic and extrinsic. Intrinsic SPAM errors are those which are part of the state preparation or measurement process, e.g. thermal populations in the $|1\rangle$ state when preparing $|0\rangle$, and erroneous dark (light) counts when measuring $|1\rangle$ ($|0\rangle$). Extrinsic SPAM errors are those arising from the gates used to produce all the starting states required for process tomography; i.e. the $|01\rangle |10\rangle , |11\rangle$ states must be prepared for two-qubit tomography, which is achieved with an X gate. Extrinsic SPAM errors arise from the implementation of this gate. These errors present a problem for tomography, as the are not part of the errors we want to address. In Section 13.1.2 we will discuss Gate Set Tomography, which can overcome this limitation at the cost of increased quantum and classical resources.

In [60] they used machine learning to classify the output from a single qubit, getting an assignment fidelity of $\mathcal{F}_a = 0.9821$, and more recently a group has improved the fidelity of optical quantum measurements by 10 % using a deep neural network [65]. Other techniques for estimating T usually involve measuring transfer matrices for smaller sub-sets of qubits and building the whole system T from that, e.g. in Ref.s: [63], [66].

3.2 Experimental Metrics

When an experimentalist is discussing the quality of a qubit, two important metrics are the T_1 and T_2 times. The T_1 time refers to the relaxation time, and T_2 time to the dephasing time [67]. Assuming that $|0\rangle$ is the ground state of the system, the T_1 time is the decay constant for the probability that the qubit remains in the $|1\rangle$ state,

$$P(|1\rangle) = e^{t/T_1}. \tag{3.1}$$

The T_2 time is the same constant, but for the time for a relative phase to change, $|+\rangle \leftrightarrow |-\rangle$,

$$P(|-\rangle) = e^{t/T_2}. \tag{3.2}$$

Relaxation and dephasing can be combined into decoherence, i.e. the state vector in the Bloch sphere is pushed towards the origin, or the maximally mixed state [43].

3.3 Decoherence Errors

Decoherence errors can be described as incorrect implementations of the quantum circuit, either by a bad calibration, e.g. over (under) rotations, or by environmental noise causing the qubit to decohere. Although there are many different sources of this class of errors, the can all be modelled as a super operator, the operator-sum of the Kraus operators, [43]:

$$\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger, \tag{3.3}$$

and to preserve the trace of the state, ρ, the must obey the relation

$$\sum_k E_k^\dagger E_k = \mathbb{1}. \tag{3.4}$$

To describe the depolarising channel, which can model gate errors in a superconducting quantum device [67] , the Kraus operators, E_k are the single-qubit Pauli operators:

$$E_k = \left\{ \frac{p}{4}\sigma_j, \frac{3p}{4}\mathbb{1} \right\} j \in x, y, z \tag{3.5}$$

and p is the level of noise modelled.

The Generalised Amplitude Damping (GAD) channel describes energy relaxation, the tendency of the qubit to relax to the ground state via dissipation to the environment, without any action on the qubit (as opposed to the depolarising channel

above as the result of a gate). The Kraus operators for the GAD channel are [43] :

$$E_0 = \sqrt{p}\begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, E_1 = \sqrt{p}\begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix}, \\ E_2 = \sqrt{1-p}\begin{bmatrix} \sqrt{1-\gamma} & 0 \\ 0 & 1 \end{bmatrix}, \quad E_3 = \sqrt{1-p}\begin{bmatrix} 0 & 0 \\ \sqrt{\gamma} & 0 \end{bmatrix}. \tag{3.6}$$

In these operators, p is the overall level of noise, as above, and γ is a parameter that describes how much of the damping occurs in the amplitude or the phase of the state.

In quantum mechanics, the system can also lose information without dissipating any energy. Physically this can be caused by interaction with the environment such as from a stray charge. We can model it as the phase damping channel, which again can occur without any gate interaction.

$$E_0 = \sqrt{p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad E_1 = \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{3.7}$$

3.4 Systematic errors

Systematic errors occur during the application of a quantum gate and are due to an error or drift in the calibration of the device. They can be modelled on the Bloch sphere as an over or under rotation or an error in the axis of rotation. To model these errors, we usually include a random distribution in the parameter we wish to model. In real quantum devices however, we overcome these errors through randomisation, which will be discussed.

3.5 Markovian and Non-Markovian Noise

Finally, we must also define Markovian and non-Markovian noise, which describe some of the assumptions made in algorithms discussed in Chapter 13. Markovian noise [68] can be explained as memoryless. Memoryless here refers to the fact that the noise has no dependence on any previous states in the circuit, i.e. that the noise towards the end of a quantum circuit is the same as the noise at the beginning.

Non-Markovian noise therefore describes noise that has some dependence on the previous gates or state of a circuit. Many noise mitigation algorithms may make the assumption of Markovian noise, but this may not hold in reality [69], [70]. We can easily see this if we consider calibration drift, where the qubits will become uncalibrated over time.

Chapter 4

Error Correction

This body of work focusses upon NISQ devices, those which have low qubit numbers and noisy gates and are unsuitable for running error correction schemes. However, we will in this chapter discuss some of the features of Quantum Error Correction (QEC) that will be relevant to Chapter 15, and will be useful for discussion of the next steps needed to build a fault tolerant quantum computer. We will focus our discussion on the surface code [71]–[73] as it is one of the most-studied and promising techniques for preserving quantum states using error correction.

4.1 Error Correcting Codes

Isolating a quantum state from its environment is *hard*. A large amount of experimental work done on real devices is effort to isolate the qubits from the environment and other qubits during the computation [74]–[84], but this can only get so far, and information stored in quantum states will eventually decohere. Quantum Error Correcting codes, such as the surface code [85] store the quantum information in a protected subspace of the whole Hilbert space, which is achieved via stabiliser measurements.

4.1.1 Stabiliser Measurements

It is impossible to measure the eigenvalue of the $\hat{X}$ operator and $\hat{Z}$ operator of a single qubit simultaneously, i.e. the $\hat{Z}(\hat{X})$ measurement projects the qubit into either the $|0\rangle / |1\rangle$ ($|+\rangle / |-\rangle$) state. However, if we introduce a second qubit, we are able to measure the two qubit $\hat{Z}_a\hat{Z}_b(\hat{X}_a\hat{X}_b)$ operators, as these commute. These

measurements are knows as stabiliser measurements, as every measurement projects the state into one of the four eigenstates, 'stabilising' it. An $\hat{X}$ or $\hat{Z}$ error on any of these qubits will project the qubit state onto a different eigenstate than the no-error eigenstate, allowing the existence of an error to be detected. However, for this simple error detection code we are unable to specify which qubit an error occurred on, which makes it impossible to correct the error [1]. We increase the number of data qubits involved in a measurement and introduce a measure-qubit to give the surface code stabilisers: $\hat{Z}_a\hat{Z}_b\hat{Z}_c\hat{Z}_d$and $\hat{X}_a\hat{X}_b\hat{X}_c\hat{X}_d$. Due to the even number of non-commuting operators within these stabilisers it is easy to verify that these operators will commute, similar to the previous operators. However, if we arrange these qubits onto a 2D grid and include the measurement qubits, we see that an error on one of the data qubits affects at least two of the stabiliser measurements ($\hat{Z}_a(\hat{X}_a)$ errors affect the $X_{abcd}X_{aefg}(Z_{abcd}Z_{aefg})$ measurements, and a $\hat{Y}$ error is just a combination of $\hat{Z}\hat{X}$ errors), which allows us to isolate an error to a single qubit. This arrangement is depicted in Figure 4.1. The circuit to measure the $\hat{Z}_{abcd}(\hat{X}_{abcd})$ operators is shown in Figure 4.2.

4.1.2 Measurement Errors

So far, all of the errors considered have been either gate errors or decoherence on the data qubits. In current quantum devices the dominant source of error is measurement error [22], so we must build error correction codes which are robust to measurement noise also. This is done by repeating a single surface code cycle, of the stabiliser measurements discussed above, for a number of repeated rounds. Absent of gate errors, a measurement error will occur on a single measurement qubit, flipping the sign of a single stabiliser which will return to the original value in the next round. By including multiple rounds within a single logical step of the surface code, we can isolate measurement errors and correct for them, i.e. if the probability of a measurement error on a qubit in a single round is p, the probability of it persisting

[1] In the NISQ era, error detection without correction can still be implemented to achieve greater fidelity, just by throwing away runs where an error occurred. This technique is knows as quantum subspace expansion [86], but increasing qubit count and circuit length will increase the number of errors, eventually leading to all circuits being thrown away.

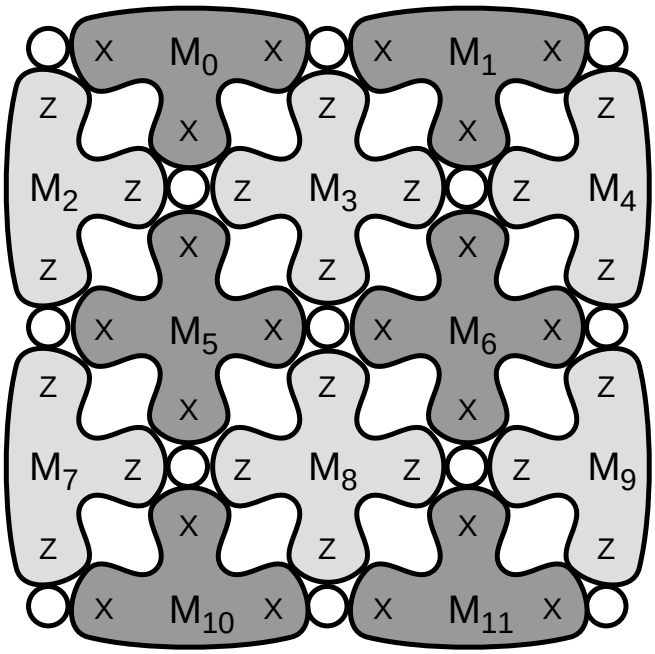


Figure 4.1: A small patch of the surface code showing the commuting measurement operators measured on the qubits at teh M_n locations. An error in either the Z or X eigenstate on a single qubit will affect two stabilisers, allowing us to pinpoint the error location. Figure reproduced from [87].

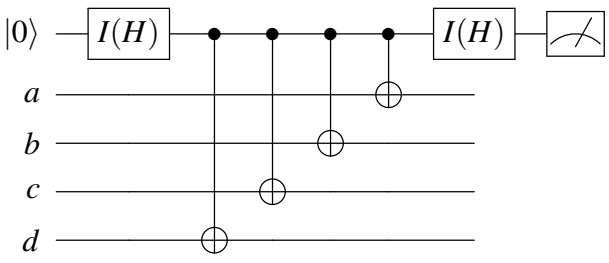


Figure 4.2: Circuit used to measure the $\hat{Z}(\hat{X})$ stabilisers, where the Hadamard gates are applied if the $\hat{X}$ stabiliser is measured.

for all of an n-round step is p^n, getting more unlikely the more rounds are in a logical step.

4.1.3 Decoding

This brief discussion of the surface code has not yet discussed how to correct errors that have been detected, even though it is an error correcting code. Due to the stabiliser measurements we are able to detect $\hat{Z}, \hat{X}, \hat{Y}$ errors occurring on the data qubits, and by measuring multiple rounds of the surface code, we are able to detect a measurement error on the measurement qubits. If a measurement error occurs isolated from any other measurements we can safely ignore it, and we are able to correct errors on data qubits by applying a quantum gate (as $P^2 = \mathbf{1}$ for each Pauli). However, the correction we apply may itself be noisy, meaning we are better off correcting these errors in our classical control software. We do this by observing that the stabiliser measurements project the qubits into an eigenstate of those measurements, but with the opposite sign of the correct measurement. For example, a $\hat{Z}_a$ will produce a sign flipped $\hat{X}_{abcd}\hat{X}_{aefg}$ measurement, which we can correct by noting the sign flipped measurement and flipping the sign of those two measurements each time they appear next (until another $\hat{Z}_a$ measurement occurs, correcting the original).

We can visualise the code in 3D space, with the x and y dimensions the plane of the surface code, and the z dimension representing time, errors in the data qubits are correlated spatially, and errors in the measurement are correlated in time. We need to extend our decoding of errors to the case where multiple errors occur in the same round, including measurement errors.

When an error occurs on a single physical qubit, it is flagged by a sign change in two stabilisers. When multiple errors occur, the output will be a change of the sign of the two stabilisers at the edges of a neighbouring qubit string. To decode these errors it is not necessary to match the exact errors, we only need to match two end points (so we can flip the signs of all the stabilisers in the chain), as the topological property of the surface code (see below, Section 4.1.4), means that closing any non-trivial loop ensures we correct the error. So the input to our decoder is a graph of nodes,

and the output should be the shortest path connecting two nodes.

This problem is actually #P-complete in general [88], so we can be pleased with minimum-weight perfect matching decoders [89], [90] which are able to give a solution to the matching problem efficiently. This decoder is also useful on other variants of quantum codes [91] , such as the colour code [92], the 3D surface code [93], [94], and the $XZZX$ code [95]; the latter is a more efficient code in the presence of biased noise [96] , which is a feature of some quantum devices [97].

4.1.4 Logical Operators

The surface code was first investigated using the topological order of the torus [73] where qubits were placed upon a torus. The surface code is the flattening of this torus onto the 2D plane, which introduces the boundary conditions necessary for utilising this code on a real (e.g. 2D planar superconducting) architecture. Consider a small patch of the surface code, which has boundaries that terminate on the X-stabilisers (smooth) and Z-stabilisers (rough). If we want to act with a logical operator on this patch, we need an operator that commutes with the stabilisers over the patch to ensure that the quantum state remains within the error-free subspace of the code. Acting with two simultaneous X operations on two data qubits within the patch will commute with the $\hat{Z}$ stabiliser acting on each of those qubits, but this stabiliser will then not commute with its neighbours. Therefore, to remain within the code space, we need to act with X operators across the whole width of this surface code patch, the same is true of acting with Z operators along the length. We now see the connection to the torus: only topologically distinct loops are able to affect the logical encoded state, i.e. the loop around the 'hole' of the torus, and the loop around the 'ring' of the torus. A string of errors which matches one of these operators causes a logical error that cannot be detected, so a larger surface code 'patch' produces less errors, allowing us to increase the code distance (length / width of a patch) to achieve a logical error rate, given an error rate of the physical qubits.

Part II

Quantum Machine Learning

Chapter 5

Quantum Machine Learning

Quantum Machine Learning (QML) can refer to a number of different problems, e.g. the processing of large amounts of classical data on a quantum device [98], or using a quantum algorithm to learn using a quantum state generated in experiment [99], or in the NISQ-era, learning the parameters of a quantum circuit that solve some task, e.g. the Max-Cut problem [100]. There are two main reasons for the use of quantum computers for machine learning, relating to the type of data to learn from. In the first case where the data originates from some quantum source, such as a sensor or experiment, we are able to probe the quantum correlations between states with much more effectiveness when we allow for quantum processing of the data, as shown in [99]. In the case where the data is classical in nature we must be more careful about any potential advantage. The ability to represent 2^n states in n qubits is where the advantage of QML on classical data is hoped to originate, and was the source of a claimed exponential speedup for matrix inversion problems [101]. However, more recently the loading of classical data into the quantum device has been considered in further detail, and when taking into account the circuit length for classical data loading, or by giving similar data structure assumptions to the classical device many QML algorithms have been shown to have only a polynomial speedup over classical counterparts, something we now call 'de-quantization' [102]–[104]

In this chapter we will briefly review the current state of quantum machine learning, in Chapter 6 we will present some work on quantum machine learning on a noisy device to perform the task of state discrimination, and in Chapter 7 we present

some work on the differing concepts of non-linearities within QML.

5.1 Classical Machine Learning

First, we will discuss some of the terminology from classical neural networks which will be useful here.

Classical artificial neural networks (ANNs) consist of a set of artificial neurons, also called nodes, connected via links [105], [106]. This is a conceptual description of ANNs, as the nodes are not physical but describe the operation done to some data as it reaches the node. The links connecting the nodes have an adjustable weight associated to them, which determines the strength of the influence of the output of one node on the node in the next layer. Within each node all the inputs with their given weights are added up, and an activation function is performed on this sum, which determines the output strength of the node. Typically the nodes are arranged in layers, where the output of all nodes in one layer is passed via the links to the inputs of all the nodes in the next layer.

The activation function is non-linear, e.g. a softmax, tanh, or ReLU function, and as a multilayer neural network with only linear activations can be replaced by a single layer, this non-linearity is believed to be a large source of the power of ANNs. We will see in Chapter 7 how non-linearities have been implemented in quantum neural networks. The power and flexibility of ANNs has been demonstrated recently with great interest, in such disparate tasks as generating novel images given a text prompt [107], generating novel text [108], and recognising objects within images [109][1].

In Chapter 6 we will demonstrate mid circuit measurements used to emulate dropout in ANNs. Dropout is a regularisation technique where nodes in the ANN are deleted at random. Regularisation is the set of techniques which try to ensure that a certain trained model does not overfit the training data, i.e. when a model fits perfectly to the training data, but not to test data, it has more likely memorised the function corresponding to the training data, than some underlying feature of the data.

[1] So fast is the pace of work in this field, this section is updated with new examples on every draft!

5.2 Quantum Machine Learning with Big (classical) Data

The success of classical machine learning with classical data spurred investigations into quantum algorithms that could improve upon the classical algorithms either by increasing accuracy or just decreasing the time taken to train a model. The Harrow Hassidim Lloyd (HHL) algorithm [101] is an early example of a proposal to decrease the time taken to train models. The HHL algorithm, and related work [110], [111] solve linear systems of equations by inverting a large matrix on a quantum device. One of the biggest applications of machine learning is in Principle Component Analysis [112] which amounts to solving an eigenvector problem. In Quantum Principle Component analysis [113] , we solve the eigenvector problem via matrix inversion with the HHL algorithm. Unfortunately, for many of the speedups to hold we require acces to the data in $\mathcal{O}(1)$ time. This required the data to be stored on a Quantum Random Access Memory (QRAM) [114], which is yet to be realised, but there has been some recent progress [115]. A QRAM is able to create a correlated superposition of the data provided,

$$\sum_{j=1}^{N} a_j |j\rangle_a |0\rangle_b \rightarrow^{\text{QRAM}} \sum_{j=1}^{N} a_j |j\rangle_a |D_j\rangle_b . \tag{5.1}$$

But one challenge of creating QRAM[2]is that to provide a speedup, many of the QML algorithms (including the HHL algorithm) require queries to be made in time $\log(N)$, but a QRAM requires $\mathcal{O}(N)$ resources to query a database of size N.

Another issue that quantum algorithms for classical data can come up against in de-quantization. This is where a novel quantum algorithm is proposed that has a speedup over the best classical algorithm, but the quantum algorithm inspires techniques that can be used on the classical device which also include the speedup. The most famous case of this is the de-quantization of a quantum algorithm for recommendation systems [116] which was de-quantized by Ewin Tang [102]. In

[2]Apart from the challenge of maintaining the superposition over long times, which is a challenge shared in quantum processing, see Chapter 4.

brief, an assumption made about the form of the input data that allowed it to be used in the QRAM efficiently is not used in the original algorithm; allowing the classical algorithm the same assumption (and working to show that the memory can be accessed efficiently) allowed Tang to show that the classical algorithm achieves the same speedup.

There is some scepticism that an efficient QRAM will ever be possible, for this reason, and that the topic of this thesis is noisy quantum devices, we will not discuss QML for big classical data further.

5.3 Quantum Machine Learning for Quantum Experiments

A more promising strategy for realising quantum advantage is to apply QML to the quantum states that are the output of some experiment. The challenge in this approach is realising practically a sensor which can store the quantum state and transfer it to the quantum processor. In [99] they demonstrate that quantum advantage can be achieved using quantum machine learning on the quantum states that are outputted from experiment. One demonstrated advantage applies to a classical probability distribution of states, showing that the quantum advantage from experiments is useful when applied to data outputted by some sensor.

One of the techniques to learn from states in [99] demonstrates that an exponential decrease in the number of experiments required to learn about a set of unentangled, but classically correlated states. The set of states is described by $\rho = 2^{-n}(I + \alpha P)$ with P an n-qubit Pauli operator. Without allowing quantum processing on the states, the best we can do to learn α and P is to use classical shadow tomography [117], which uses randomised Clifford measurements on each copy of ρ. The classical post processing required for this technique is exponential in n, and the randomised Clifford measurements require circuits that are too deep for today's devices, except for only small values of n. In contrast, if we allow quantum processing of the states, and a classical neural network to process the output of this computation, we require exponentially fewer experiments. In the quantum enhanced setting, we

store a copy of ρ in the quantum memory, and with the next copy of ρ generated we perform an entangling Bell measurement. The bitstring output of this is fed into a classical neural network, which has been trained on labelled states generated from smaller noiseless simulations. In [99], they demonstrate with a current quantum device, the Google Sycamore processor, that an exponential reduction in the number of experiments can be achieved.

The authors of [99] also show that a similar advantage can be achieved for quantum processes. In this demonstration, we are instead attempting to learn about the quantum channel, $\mathcal{E}$ that prepares the measured states, ρ. In this case, the states were generated by a random quantum circuit, which is either time symmetric or not. The task is to separate the measured states into these categories. The output of each setting was analysed by a classical neural network, which in this case was unsupervised, i.e. there was no labelled training data. In this case, the authors again find that the number of experiments required to separate the classes is exponentially lower with access to quantum processing, and that in experiment with the Sycamore chip, limited to 3000 experiments, the quantum enhanced networks could separate the data, whilst the purely classical networks could not.

In Chapter 6 we demonstrate a quantum neural network for a similar task, of state discrimination, and we show that a noisy circuit can discriminate the states better than the known POVM when subject to noise.

5.4 Quantum Machine Learning with NISQ devices

Here we will discuss some more QML results that do not rely upon QRAM or fault-tolerant devices. With NISQ devices in their infancy, many of the techniques here will rely upon optimisation and measurements which we will discuss in Chapter 10, therefore here we will restrict ourselves only to the QML specific parts of the algorithms.

5.4.1 Quantum Approximate Optimisation Algorithm

One of the earliest claimed demonstrations of a quantum device is the quantum annealer [118] , which uses novel hardware to do simulated annealing. Simulated

annealing is a classical algorithm [119], [120] where a particle in a landscape is slowly cooled to reach the global minimum. The quantum annealer adds the ability for the particle to tunnel through the potential barrier. The popularity of quantum annealing solutions to problems lead to the development of an analog of the quantum annealer in gate-based quantum computing architectures. This method is the Quantum Approximate Optimisation Algorithm [121] , or the Quantum Alternating Operator Ansatz [122] (QAOA).

The second abbreviation goes some way to describing what the ansatz is: a series of alternating parameterised operators that work analogously to simulated annealing. We apply a cost operator and a mixing operator alternately to the state for a number, p, of layers, giving the final state,

$$|\psi(\beta,\gamma)\rangle = \left\{ e^{-i\beta H_B} e^{-i\gamma H_p} \right\}^p |\psi_0\rangle \,. \tag{5.2}$$

Here H_B is the mixing Hamiltonian and H_p is the problem Hamiltonian, γ, β are free parameters to be optimised and $|\psi_0\rangle$ is a suitable starting state. The analogy with simulated annealing is that H_p imposes the problem of interest, like the simulated annealing landscape, and H_B mixes the state between potential solutions, with the strength of β determining how much mixing is done, like the temperature in simulated annealing. We can also think of the QAOA algorithm as a Trotterisation of the continuous time quantum annealing Hamiltonian.

By applying the mixing Hamiltonian,

$$H_B = \sum_{i=1}^{N} \sigma_i^x, \tag{5.3}$$

alternating with the constraint Hamiltonian, which depends on the problem implementation, to get the state $|\psi\rangle$, we measure $\langle\psi|C|\psi\rangle$ and minimise. A large part of the work in QAOA is done to design a Hamiltonian and cost function which well matches the problem.

5.4.2 Quantum Walk

In the classical random walk algorithm we have a discretized parameter space, e.g. a two dimensional grid, and the algorithm chooses which direction to move in randomly. In [123] the QAOA algorithm was augmented with a continuous time quantum walk as the mixing operator, to produce the Quantum Walk Optimisation Algorithm (QWOA).

Quantum interference and correlations are allowed in the quantum walk algorithm, as opposed to a classical random walk [124], [125]. The problem of traversing a decision tree tom determine if there is a node at the nth level is considered [124]. Classically, this can be solved if there is a high probability of reaching nodes at the nth level. However, if it is hard to distingush between trees with and without nodes at the nth level it is hard to solve classically.

With a quantum walk, the location of the 'walker' is encoded in a quantum state, and it is evolved with a quantum operator that picks the direction to move. If we use a Hadamard operation before evolving in the direction to travel, the state can traverse the graph in superposition, meaning that the standard deviation of the final state grows as T^2, where T is the time step, which only grows as T in the classical case.

In [123], where the QWOA algorithm was introduced, they replace Equation 5.3 with a continuous time quantum walk operator,

$$U_W(t) = U_\#^\dagger \mathcal{F}_M^\dagger e^{i\lambda t} \mathcal{F}_M U_\#. \tag{5.4}$$

Where $U_\#$ is the indexing unitary, which maps the M binary strings associated with valid combinations into the first M binary strings, which is undone by $U_\#^\dagger$. F performs the Quantum Fourier Transform, i.e. converting the data stored in the amplitudes into the phase of the state, and λ is a diagonal matrix that describes the connectivity of the objects.

5.4.3 Quantum Monte Carlo

Monte Carlo techniques, named for the casino, is named so as it repeatedly samples from a distribution, relying upon the law of large numbers to produce the expected value, by taking the empirical mean, μ, of the samples. Using Monte-Carlo on a classical computer has been very successful in many domains, including nuclear physics [126], materials science [127], statistical physics [128], and the application discussed here, finance [129]. This is despite Monte Carlo techniques requiring lots of computer resources as models become more complex, or the events modelled become rarer. We must also distinguish here between Quantum Monte Carlo, using Monte Carlo simulations of quantum systems [130], from the quantum speedup of Monte Carlo methods [131] which we will discuss.

In [131] they reduce the number of samples taken to estimate μ, to additive error ε, with standard deviation σ using $\tilde{\mathcal{O}}\left(\frac{\sigma}{\varepsilon}\right)$ samples, whereas the classical algorithm requires $\mathcal{O}\left(\frac{\sigma^2}{\varepsilon^2}\right)$ samples. This construction also allows for the sampled algorithm, $\mathcal{A}$ to be a quantum algorithm, or for it to be samples from a quantum experiment, as discussed in Section 5.3. The algorithm approximates the mean of non-uniform distributions using amplitude estimation.

Quantum Monte Carlo can estimate μ if it lies between $[0,1]$, which is used as a subroutine in the amplitude estimation algorithm, estimating the distribution of μ_l between $\left[2^{l-1}, 2^l\right]$ with $l = \{0,1,\ldots k\}$ for each l. This is done iteratively up to the final bound k to output the estimation $\tilde{\mu} = \sum_{l=0}^{k} 2^l \tilde{\mu}^l$. This is finally modified to produce the mean of functions which can also be negative.

The first algorithm that estimates $\mu \in [0,1]$ does all of the quantum heavy lifting, it is this algorithm which allows for the initial speedup. Firstly, we assume that the quantity we wish to measure can be encoded as a quantum algorithm, which we will call $\mathcal{A}$, and that it makes no mid-circuit measurements. The W operator that conditionally rotates an ancilla qubit by the value $\phi(x)$, which if the outcome of $\mathcal{A}$ given the input x. Here $x = \{0,1\}^k$ is the binary encoding of a real input, and the

operator W can be written as

$$W|x\rangle|0\rangle = |x\rangle\left(\sqrt{1-\phi(x)}|0\rangle + \sqrt{\phi(x)}|1\rangle\right). \tag{5.5}$$

By iterating the amplitude estimation algorithm on the state,

$$|\psi\rangle = (I\otimes W)(\mathcal{A}\otimes I)|0\rangle^{\otimes n+1}, \tag{5.6}$$

we can estimate μ, as a controlled W operator gives a phase kick-back of $\phi(\bar{x}) = \mu$ on the control qubit.

Amplitude estimation [132] decreases the number of samples required to estimate a value with a generalisation of the Grover's search algorithm [2]. The algorithm works by applying the diffusion operator,

$$\mathcal{D} = \begin{bmatrix} \frac{2}{N}-1 & \frac{2}{N} & \dots & \\ \frac{2}{N} & \frac{2}{N}-1 & \dots & \\ \vdots & \ddots & & \\ \frac{2}{N} & \frac{2}{N} & \dots & \frac{2}{N}-1 \end{bmatrix}, \tag{5.7}$$

which inverts the amplitudes of a state about their mean, e.g. reflects the vector. Combining this with the W operator which increases the amplitude of the output of $\mathcal{A}$, and iterating many times, means that measurement of $|x\rangle$ will give us the value of μ with high probability.

5.4.4 Quantum Boltzmann Machine

The Quantum Boltzmann Machine is another generalisation of an algorithm found in classical machine learning. A Hamiltonian is defined for the network of connected nodes, i.e. $H = \sum_{i<j} w_{ij}s_is_j + \sum_i \theta_i s_i$, with s_i the state of node i, w_{ij} the weight between nodes i, j and θ_i the bias of node i [3]. The Boltzmann Machine [133] is an early generative model which is so-called as it utilises the Boltzmann distribution,

$$P_\alpha = \frac{e^{-E_\alpha/k_BT}}{\sum_{\beta=1}^{M} e^{-E_\beta/k_BT}}, \tag{5.8}$$

to sample from the output nodes of the network, where P_α is the probability of state α, which has energy E_α, given by $H|\alpha\rangle$; k_B is the Boltzman constant, and T is the "Temperature" of the system.

The quantum implementation of a Boltzmann machine [134] we consider each site as a qubit, so the state of each site can be represented by a Pauli operator, which we will choose to be σ_i^x. The Hamiltonian is then given by

$$H = -\sum_{i<j} w_{ij}\sigma_i^z\sigma_j^z - \sum_i \theta_i\sigma_i^z + \sum_i \Gamma_i\sigma_i^x. \tag{5.9}$$

We then train the Quantum Boltzmann Machine (QBM) until the measured probability distribution of the measurement outcomes is close to the distribution of the input data. With a small enough systems, these can be implemented classically, but similar to the Ising model, larger systems may have to be sampled from using a quantum device.

[3]Notation here has been chosen to maintain consistency with machine learning literature, but is similar to the Ising model.

Chapter 6

Quantum State Discrimination with Quantum Neural Networks

This chapter is mostly based on the work in [135] which this author contributed to with the implementation and simulation of the whole system and the writing of the manuscript. This is building upon work done by the other authors of [135] in [136]. We present an extension to noisy devices of the approach for quantum state discrimination outlined in [136], a quantum analogue of a neural network used for state discrimination. In [136] simulations of shallow quantum circuits were trained to find the optimal Positive Operator Valued Measure (POVM), or measurement, to distinguish between two families of non-orthogonal quantum states. The families of states will be described below. Given an input state chosen randomly from one of the families, the output of the network should indicate which family the input was chosen from. To do this the network is trained on a set of labelled data, performing supervised learning [105]. The ideal POVM was learned via a classical optimiser using a gradient descent algorithm on the quantum parameters, which correspond to the rotation gates in the quantum circuit. This POVM is distinct from the error minimising POVMs, as it also attempts to minimise the occurrence of inconclusive results.

Quantum state discrimination is important in many emerging quantum technologies: quantum cryptography [137], entanglement concentration [138], quantum cloning [139], and quantum metrology and sensing [140], [141]. Quantum circuits

trained for classification could also be used in quantum machine learning problems as a classifier of quantum data.They could classify the output of other quantum circuits, e.g. the output of a quantum generative adversarial network (GAN) [142].

We represent the states as density matrices in the simulation, so that we can model noise in the quantum device. We also simulate calculation of the parameter gradients on the quantum device, which would also be subject to noise in a real machine. We find that with these extensions including the effect of noise the previous algorithm proposed for noiseless systems no longer performs optimally. To recover performance we reduce the number of trainable parameters through consideration of the circuit structure.

6.1 Defining States to be Discriminated

We wish to discriminate a two-qubit input state, $|\psi_{in}\rangle$, which in general can be represented as a normalised vector with 4 complex components. The state is chosen randomly from two sets of states, labelled a and b, and elements defined by:

$$|\psi_{in,a}\rangle = (\sqrt{1-a^2}, 0, a, 0), \tag{6.1}$$

$$|\psi_{in,b}\rangle = (0, \pm\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0), \tag{6.2}$$

Eqn.(6.1) defines the states in the first set, $|\psi_{in,a}\rangle$, where by setting the continuous real parameter $a \in (0,1]$ the specific state is chosen. The number of states within this first set is therefore infinite due to the fact that a can be varied continuously. Eqn. (6.2) defines the second set, $|\psi_{in,b}\rangle$, which has two elements, given by $|\psi_{in,b,+}\rangle = (0, 1/\sqrt{2}, 1/\sqrt{2}, 0)$ and $|\psi_{in,b,-}\rangle = (0, -1/\sqrt{2}, 1/\sqrt{2}, 0)$. The discrimination in our manuscript is performed between these two sets of states: one with an infinitely large continuous set of states, and one set with two states in it. State discrimination between general sets of states is discussed in Ref. [143]. We set the probability that a state from the a set appears as input state to $p_a = 1/3$, the probability that a b state with positive sign appears to $p_{p+} = 1/3$, and the probability that a b state with a negative sign appears to $p_{p-} = 1/3$. Note however that our results are expected to

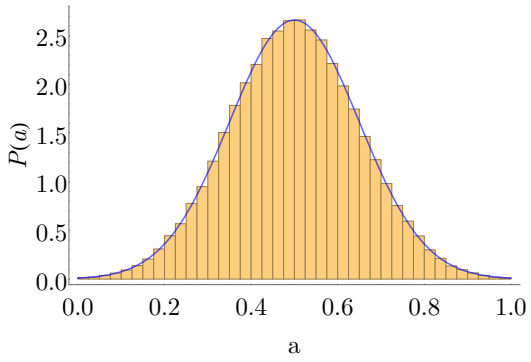


Figure 6.1: Distribution $P(a)$ of the parameter a for the input states in the first set (Eq.6.1), for an average value $\mu_a = 0.5$ and a standard deviation $\sigma_a = 0.15$. These values of μ_a and σ_a are used for the data in Figs. 6.4 - 6.8, and 6.11. Due to the rather large value of σ_a the most probable values of a are not confined only in a thin range around μ_a, but instead cover almost all the range between 0 and 1.

be applicable also to other sets of states, since the methods presented here are based on variational algorithms, which can be formulated for any target state.

We now define the probability distribution, $P(a)$, that describes how the parameter a of the first set of input states is statistically distributed in the range from 0 to 1. We use a normal distribution with an average value, μ_a, and a standard deviation around this average value, σ_a. As illustrative example in Fig.1 we show $P(a)$ for $\mu_a = 0.5$ and $\sigma_a = 0.15$. The set of states defined by this distribution has about 68% of the states in the range $a = 0.35$ to $a = 0.65$, 25% of the states within the range $a = 0.1$ to $a = 0.35$ and $a = 0.65$ to $a = 0.9$, and still a non-negligible probability of about 5% of finding a state in the range between $a = 0$ to $a = 0.1$ and $a = 0.9$ to $a = 1.0$.

Primarily our specific sets of states a and b are chosen to reproduce the work in [136] and in [144], where the state discrimination was performed in a laboratory. Secondly, these states are non-orthogonal and therefore cannot be distinguished perfectly without some probability of erroneous or inconclusive outcomes, making the problem harder for the algorithm. It is therefore an ideal case to verify the method, since the level of non-orthogonality can be tuned by choosing μ_a and σ_a in the distribution function of a in Eq.(6.1), with a value of a closer to 1 being more difficult to discriminate.

6.1.1 The Quantum Neural Network for State Discrimination

There are two non-orthogonal states to discriminate, so if we wish to have a network that can be trained to not commit any errors, we must allow for it to produce an inconclusive result [144]. This allows the network to give a 'don't know' result as opposed to an erroneous one. Therefore we have a minimum of three outputs, necessitating two measurement qubits.

The output of the network is determined by the measurement outcome. As we begin in a random configuration and are training the system, we can arbitrarily select which label a measurement outcome corresponds to:

$$\{|00\rangle : a, |01\rangle : b, |10\rangle : a, |11\rangle : \text{inconclusive}\}. \tag{6.3}$$

The choice of unbalanced labels may have an effect upon the outcome. For a random measurement outcome the probability to guess the right state is 1/2 for a and 1/4 for b. We partly mitigate this bias setting the probability of a and b states to appear as input to the values specified in the previous subsection, namely $p_a = 1/3$ and $p_b = p_{b+} + p_{b-} = 2/3$.

This results in the probability of correctly guessing the input state for a fully random measurement outcome to be $1/2p_a + 1/4p_b = 1/3$, and correspondingly the probability for an incorrect guess is equal to $2/3$. In general one might adapt the assignment of measurement outcomes to labels according to the considered specific task.

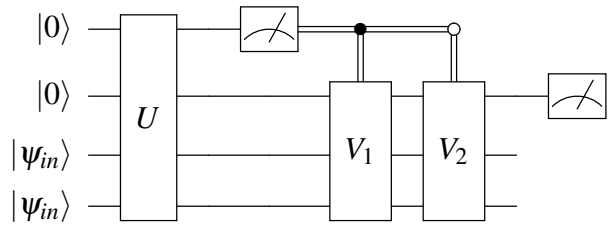
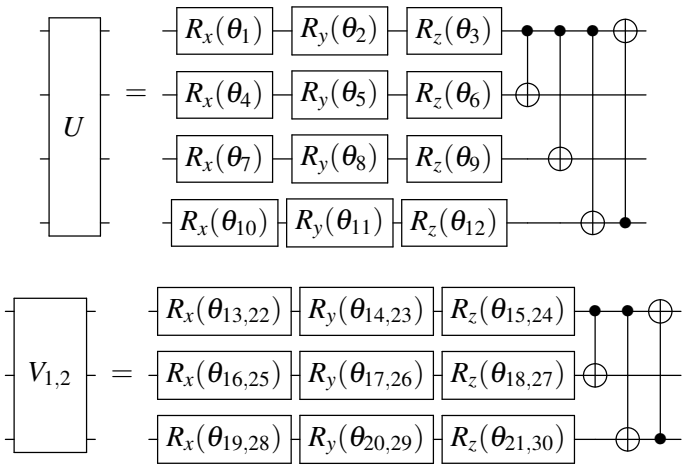


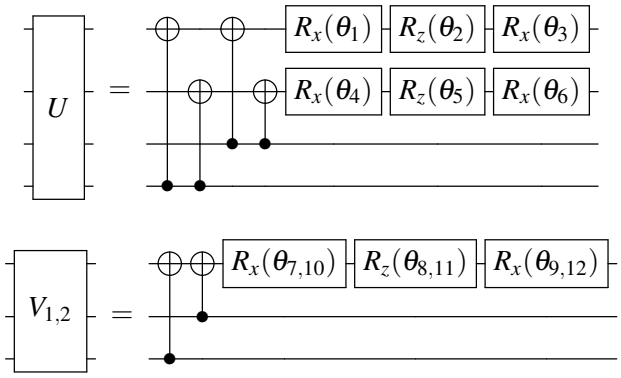
Figure 6.2: The general form of the quantum circuits used in this work.The input state is on the bottom two qubits, and measuring the first qubit introduces a non-linear dropout layer. The sub-circuits U, $V_{1,2}$ are shown in Figure 6.3.

The structure of the U and $V_{1,2}$ circuit blocks is given in Figure 6.3, where 6.3a shows the same circuits used in [136] and 6.3b shows the reduced circuits

introduced here, which we will discuss in more detail below. These circuits are small and have low-depth, so that they can be ran on a quantum computer which supports measurement as the circuit is running and classical feedback. This requires fast measurement and fast classical processing which is not possible in many current systems, but has been achieved in an ion-trap device [145], meaning this algorithm could run on a current device.



(a) The U and V circuit blocks originally used in [136].



(b) The form of the U and V blocks with a reduced number of parameters.

Figure 6.3: The circuits showing the trainable parameters, which are used in this work.Comparison of results obtained for the circuits 6.3b and 6.3a is made in Section 6.2.1.

The state discrimination task is then as follows: input states are drawn from two sets of states, and the classical optimiser must optimise the rotation angles $\theta_{1..n}$ of the quantum circuit to maximise the likelihood of a correct determination of the state.In our specific case it has to determine whether an input state from the a set of states or the b set of states. Note that only these states are allowed as input states

during both training and testing of the circuit. A correct determination is found when the measurement output of the quantum circuit is equal to the corresponding input state label as defined in Equation 6.3.

This task is the minimum error discrimination problem, which has been solved classically using semidefinite programming methods [146]–[148]. Here we solve the same problem using a noisy quantum circuit, and demonstrate the usefulness of using quantum measurement as a form of dropout. Note that each step of the semidefinite programming method requires $O(m^4)$ classical steps, where m is the length of the input vector, which scales exponentially with the number of qubits required to represent the states. The scaling of variational quantum circuits such as the ones used in our approach is not known currently, especially as the number of trainable parameters increases, and is a topic of ongoing research [149]–[152].

6.1.2 Optimisation

Since the input states are initially labelled, the task for the classical optimiser is a supervised learning task [105]. The optimiser used in this experiment is Adam [153], which has been found to work well in a number of quantum variational algorithms [66], [136], [154]–[156]. It has also been shown classically that Adam deals well with noisy gradients [157], which will be the output of our noisy quantum computer. We will fully discuss the choice of optimisers for quantum variational algorithms in Section 10.3. Other optimisers such as RotoSolve [158] have been proposed, and a comparison of performance can be made in future work.

Noisy gradients are a feature of the work here: as gradient calculation must be performed on the noisy quantum device, we expect that the output gradients will be noisy. We also expect that there will be non-optimal local minima in our loss landscape, as this is also a feature of the loss function in the noiseless case [136]. Finally we also expect that the loss landscape may feature ‘barren plateaus’, as these have been shown to be a feature of quantum optimisation problems [159], which is discussed in Section 10.4.

We define the function to minimise, the cost function, as

$$C = \alpha_{\text{err}} \text{P}_{\text{err}} + \alpha_{\text{inc}} \text{P}_{\text{inc}}, \tag{6.4}$$

where the positive real numbers $\alpha_{\text{err}}, \alpha_{\text{inc}}$ are the cost parameters used to bias the network towards minimising errors or inconclusive results ($\text{P}_{\text{err}}, \text{P}_{\text{inc}}$ are defined below). If for example we require the network to produce fewer errors, we can do this at the cost of recording more inconclusive results by increasing the value of α_{err} relative to the value of α_{inc}. The gradient of the cost function is calculated via the parameter-shift rule, Equation 10.30 discussed in Section 10.3.1. Such intermediate optimisation strategies based on changing the cost parameters, thereby allowing for a trade-off between minimum-error and unambiguous state discrimination, were studied in Refs.[160]–[163]. Here we use a quantum algorithm to solve this problem, and perform the analysis of how noise in a quantum device affects the results. We discuss the effect of changing the cost parameters in Section 6.2.1.

While the input states are always pure states, as they are processed in our quantum circuit the noise causes them to become mixed states. To simulate this effect of the noise in the quantum circuit we use density operators, ρ, to represent quantum states inside the quantum circuit. The measurement probabilities of a state, ρ, for a generalised measurement, $M = |\phi\rangle\langle\phi|$, are given by:

$$\langle\rho\rangle = \text{Tr}(|\phi\rangle\langle\phi|\rho), \tag{6.5}$$

and the quantum state after measurement is given by

$$\rho_{\text{measured}} = \frac{|\phi\rangle\langle\phi|\rho|\phi\rangle\langle\phi|}{\text{Tr}(|\phi\rangle\langle\phi|\rho)}. \tag{6.6}$$

Using this we can find the probability of an erroneous or inconclusive measurement

$$\text{P}_{\text{err}} = \sum_{\rho_i \in b} (\langle\rho_i\rangle_{00} + \langle\rho_i\rangle_{10}) + \sum_{\rho_i \in a} \langle\rho_i\rangle_{01}, \tag{6.7}$$

$$\mathrm{P_{inc}} = \sum_{\rho_i \in a,b} \langle \rho_i \rangle_{11}, \tag{6.8}$$

where ρ_i is the input state, and $\langle \rho_i \rangle_{jk}$ refers to the probability of obtaining a measurement of $|jk\rangle$ from the circuit.

Discrimination of these states, without the use of a variational algorithm, has been shown in the laboratory to reach the theoretical best success probability, $\mathrm{P_{suc}}$ of 0.833 for $\mu_a = 0.25, \sigma_a = 0.01$ [144]. This is a minimum loss, $L = 1 - \mathrm{P_{suc}} = \mathrm{P_{err}} + \mathrm{P_{inc}}$, of 0.166. For the equal probability case, $P(|00\rangle) = P(|01\rangle) = P(|10\rangle) = P(|11\rangle) = 0.25$, the success rate is 0.385, this translates into a loss of 0.635. This gives us lower and upper expected bounds to compare our results for the loss to.

6.1.3 Reduced circuit

For the probability distribution of a determined by $\mu_a = 0.25$ and $\sigma_a = 0.01$ the maximum theoretical success rate ($\mathrm{P_{suc}}$) is 0.8333 [144], which was obtained with the long circuit in Ref.[136]. However, after optimisation of circuit parameters for our larger circuit in Figure 6.3a we reach only 0.72, which is significantly smaller than the theoretical limit. We attribute this discrepancy to the different implementations of the optimisation procedure, and to the different calculation of the gradients. To overcome this sub-optimal result we designed the shorter circuits in Fig.6.3b. The choice of the reduced circuit is motivated by the consideration that for this task the rotations on the state qubits have a smaller effect on the measurement outcomes than rotations on the measurement qubits. This choice of structure is so that the input states are entangled with both output qubits, and then the measurement qubits are rotated. The choice of rotations about the x-axis, followed by the z-axis, and then again the x-axis allows for the initial state to be transformed to any other state on the surface of the Bloch sphere [43]. With this short circuit (Figure 6.3b) we obtain a success rate of 0.826, close to optimal performance. This is the circuit used for the results presented, except where we explicitly note that the longer circuit is used.

We note that as the shorter circuits do not explore the full Hilbert space of all the qubits, they may not be necessarily optimal for all discrimination tasks. Investigations into the capability of different variational quantum circuits have been

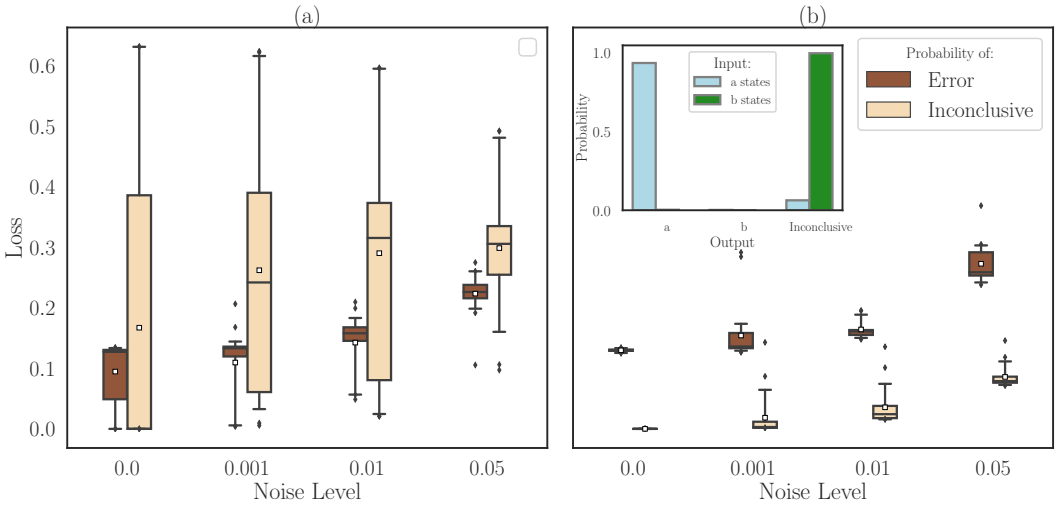


Figure 6.4: The distribution of P_{inc} and P_{err} from 25 repeats of **(a)** a network biased towards reducing errors $\alpha_{err} = 60$, $\alpha_{inc} = 10$.**(b)** One with a balanced cost function, $\alpha_{err} = \alpha_{inc} = 40$. Both with values $\mu_a = 0.5$, and $\sigma_a = 0.15$. An example undesirable output for a single minimising error run is in the inset, where no b states are measured correctly, but the network still converges (the x-axis shows the output label and the colour is the input state). The interquartile range is contained within the box, and the 5th and 95th percentiles are marked by the whiskers. Outliers of this range are marked by a diamond. The mean is marked with a white square, and the median is the line across the box.

made in [164]. Here we present evidence that when used on a noisy device, the smaller variational circuit converges to better results than the larger circuit. In general a trade-off needs to be made between this better resilience to noise and the ability of the circuit to distinguish very complex states.

6.1.4 Noise

As discussed in Section 3.3, we model the noise in the quantum computers with a superoperator, $\mathcal{E}(\rho)$, which is a completely positive, trace-preserving map on the state ρ [43]. For the single-qubit noise channel our operators are the single qubit Pauli operators, modified by the noise probability, p, to give the depolarising channel,

Equation 3.6, which we repeat here for ease of presentation:

$$E_0 = \sqrt{1-\frac{3p}{4}}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}, E_1 = \sqrt{\frac{p}{4}}\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix},$$
$$E_2 = \sqrt{\frac{p}{4}}\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}, \quad E_3 = \sqrt{\frac{p}{4}}\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}. \tag{6.9}$$

The probability of the single qubit noise channel is $p_{1q} = \frac{4}{5}p_{2q}$. This is the one-qubit marginal probability of error for the two-qubit gates [165], i.e. the probability of a single qubit error without condition of an error on the other qubit. This is a commonly used assumption in the quantum error correction literature [166], which assumes that the error process in single and two qubit gates is the same. In real devices the process can be quite different, but we nevertheless choose this method as it is an upper limit on the error probability of the single qubit gate. When quoting the noise level in this paper, we will always refer to p_{2q}. We set the highest noise level in our simulations to $p_{2q} = 0.1$, as this is an upper limit on two-qubit gate fidelities reported on current quantum hardware [18], [167], [168].

Note that here we have not considered asymmetric noise or different quality qubits. However, we believe that correcting for a systematic bias such as this is possible for a variational algorithm, as seen in [169]. Furthermore, in actual devices the single qubit noise probability reported is much lower than $4/5$ of the two qubit gate noise level. For example, the single qubit gate error rate reported in [168] is 1.4×10^{-3}, whereas the two qubit gate fidelity is 9.3×10^{-3}, and the ratio between these is approximately $3/20$, at least a factor of 5 lower. In our simulations the single qubit noise is set to the higher limit of 4/5, so that we are more demanding of the algorithm.

6.1.5 Simulation

Simulations of the quantum device were performed on a simulator built using the Tensorflow machine learning package [170], and verified with the Cirq [171] quantum simulation package. In our simulations we set the initial angles, which are our

parameters to be optimised, at random values. The labelled quantum state is an input to the circuit in Figure 6.2, that circuit is ran and the measurement probabilities calculated and with them the cost. The gradient of the cost with respect to each parameter is then calculated by the method described in section 10.3.1, and the parameters are updated according to the Adam optimiser to minimise the cost. This routine is repeated until the cost no longer significantly decreases.

During the training process of the optimal quantum circuit rotation angles, $\{\theta_i\}$, at each iteration we evaluate the cost function in Eq.(6.4) with a number of randomly chosen values of a within a given distribution $P(a)$. This number is 20 for each training step, of which there are typically 1000-3000 until convergence. Due to this rather large number of samples used in the training optimisation it is ensured that the whole distribution of a according to $P(a)$ is covered. Hence the circuit is trained to discriminate between the whole continuous set $\{|\psi_{in,a}\rangle\}$, with a distributed according to $P(a)$, and the two states in set $\{|\psi_{in,b}\rangle\}$. Once the circuit parameters are trained, for the testing step we use 250 samples of a values, distributed again following $P(a)$. Therefore the QNN is trained not only for one specific given value of a, but rather for the chosen distribution of a values.

Measurements here are calculated in the 'infinite-shot' regime, where the representation of the quantum state at the end of the circuit is used to extract exact measurement probabilities. The inclusion of statistical measurement noise can be expected to result in slower rate of convergence than obtained here. We note that in Ref. [155] it was demonstrated that convergence of variational algorithms is guaranteed even for single-shot measurements of the gradient. We indicate that convergence can also be achieved using this method in presence of measurement noise, although with a higher number of iterations.

6.2 Results

6.2.1 Effect of cost function choice and circuit depth

In Figure 6.4 we compare the obtained optimised P_{err} and P_{inc} for an error minimising cost function ($\alpha_{err} = 60, \alpha_{inc} = 10$) and a balanced cost function ($\alpha_{err} =$

$40, \alpha_{inc} = 40$). The error minimising cost function often results in a practically unusable network, because while it gives a low probability of error, the probability of inconclusive results is too high, as seen for an extreme case in the inset of Figure 6.4.

In comparison to the error minimising setting, the results for the balanced cost function are stable and generally give both small P_{err} and P_{inc}, with some P_{err} comparable to the error minimising setting. For the remaining analysis we therefore use the balanced cost function ($\alpha_{err} = \alpha_{incon} = 40$). We note that as the noise level is increased, P_{inc} and P_{err} progressively tend to larger values. The effect of noise will be analysed in detail in the next section.

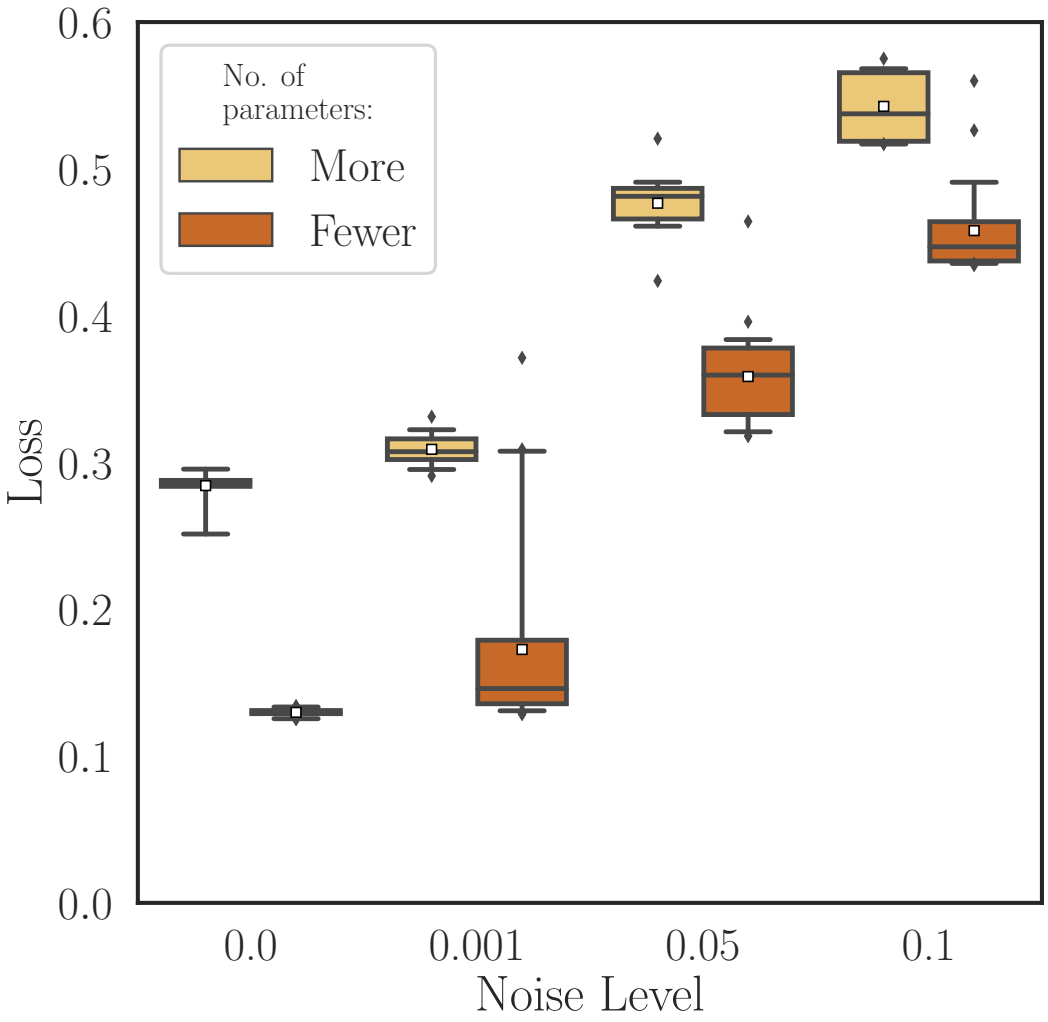


Figure 6.5: The distributions of loss ($P_{err} + P_{inc}$) at different noise levels for the two circuits shown in Figure 6.3. Both have other parameters fixed, $\mu_a = 0.5, \sigma_a = 0.15, \alpha_{err} = \alpha_{inc} = 40$. We observe that reducing the number of parameters is advantageous at all noise levels.

We next investigate the influence of the number of parameters in the quantum circuit on the loss. In Figure 6.5 we compare the distributions of loss between the circuit with more trainable parameters in Figure 6.3a to the circuit with fewer parameters in Figure 6.3b. It can be seen that the reduced circuits consistently perform better than the long circuits. It is more difficult to train circuits with a large

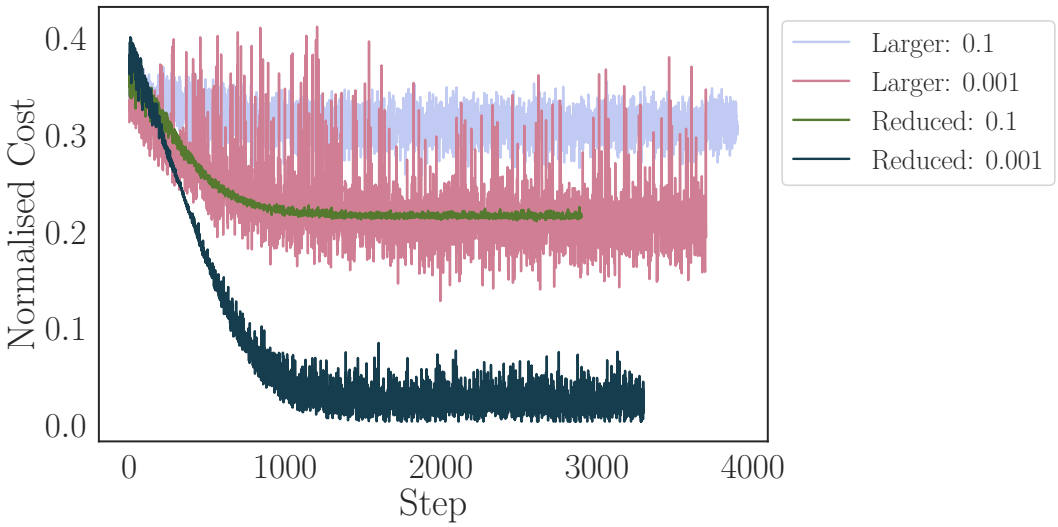


Figure 6.6: Evolution of the normalised cost functions for larger and reduced circuits for $\mu_a = 0.5$ and $\sigma_a = 0.15$, with noise levels of 0.001 and 0.1. Shown here is the number of steps taken to converge. Note that the time taken to complete a single step of the longer circuit is much greater than for the reduced circuit.

number of parameters both without and with noise, as seen in Figure 6.6. We see that the higher noise cases always converge to a higher loss, and that the reduced circuits perform better in both cases. These results show that in practice increasing the number of parameters used in a quantum circuit does not always have a beneficial effect. Importantly, even in the noiseless case the circuit with less parameters leads to better results. Furthermore the reduced number of parameters also significantly lowers the required run-time.

Even with very low noise, the output is worse for larger circuits. This suggests that with more parameters the algorithm struggles to optimise, when the gradient calculations are performed on the quantum device. Good performance of the short circuit in the presence of noise can be due to the noisy gradient regularising the training, thereby optimising performance [105]. Moreover, the Adam optimiser

has been designed to work well with noisy gradients [153]. The results seen here are indicative that a noise-resilient optimiser using gradients provided by a noisy quantum circuit can perform well.

6.2.2 Effect of noise: numerical analysis and model

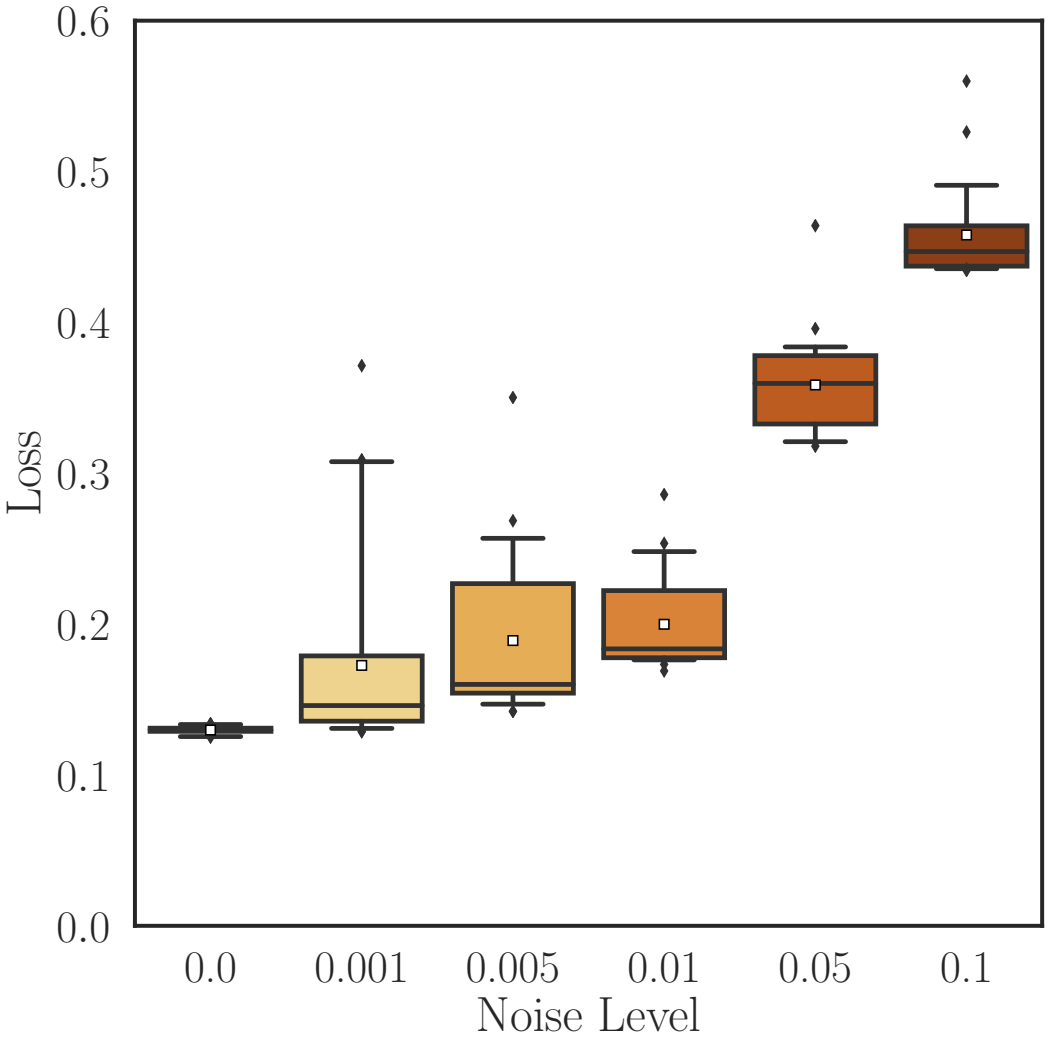


Figure 6.7: The distribution of loss for 25 repeats of training the network. The cost function is balanced, $\alpha_{\text{err}} = \alpha_{\text{inc}} = 40$, $\mu_a = 0.5$, and $\sigma_a = 0.15$. At levels of noise present in current devices, 0.01, the loss value is favourable, an average of 0.2.

In Figure 6.7 the noiseless case is compared to resulting optimised loss for increasing noise levels (note that in this section we always use the reduced circuit). It can be seen that using this algorithm with zero noise produces the lowest loss, as one expects intuitively. With increasing noise the average loss increases continuously.

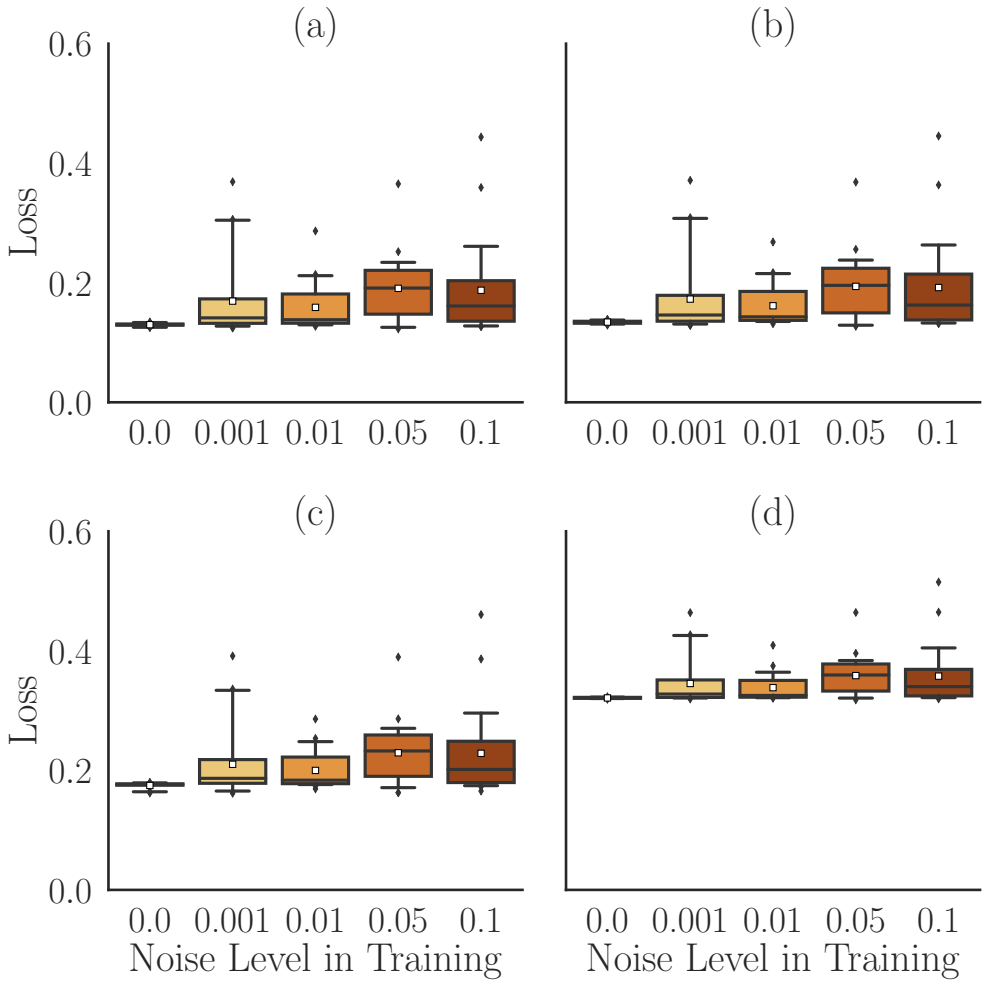


Figure 6.8: Distribution of loss ($P_{err} + P_{inc}$) against training noise for different noise levels in the validation circuit; **(a)** 0.0, **(b)** 0.001, **(c)** 0.01, **(d)** 0.05.

In presence of noise there are a few high-loss outliers, which we attribute to the optimiser becoming stuck in local minima of the cost function. As the noise is increased, performance deteriorates, but is no worse than the random output limit of $2/3 \approx 0.67$ (see Sec. 6.1.1).

Importantly, at noise levels comparable to current devices, $p_{2q} = 0.01$, the algorithm is still performing well, at an average loss of 0.2.

In general a high level of noise always leads to a higher loss. However, we find that when noise is applied only during the training of the parameters, the optimised parameters are rather resilient to this training noise. To show this in Figure 6.8 we

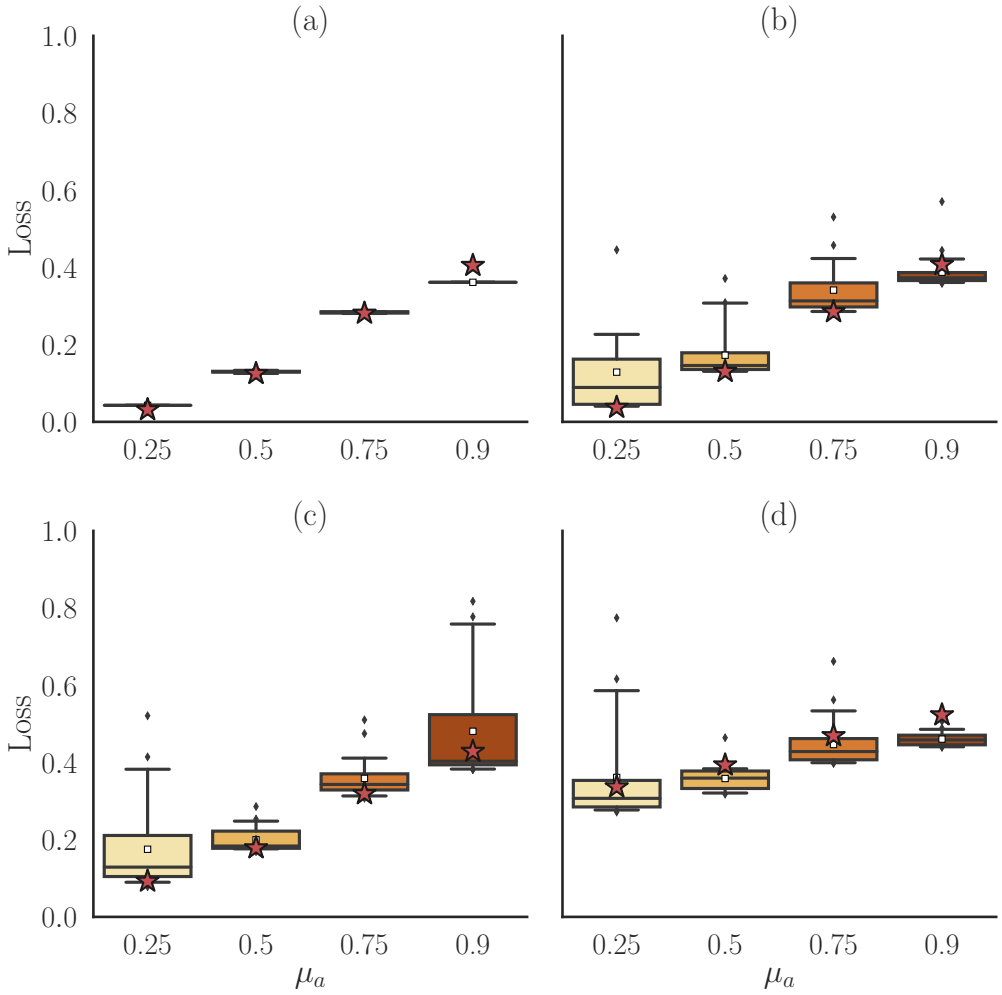


Figure 6.9: The distribution of loss ($P_{err} + P_{inc}$) and the effect of different values of μ_a. The cost function is balanced, $\alpha_{err} = \alpha_{inc} = 40$, and $\sigma_a = 0.15$. The noise level is **(a)** 0.0, **(b)** 0.001, **(c)** 0.01, **(d)** 0.05. We see that for lower values of μ_a, corresponding to smaller overlap between the states to be discriminated, the discrimination task is performed better. The red stars indicate the fidelity $F_{\tilde{a}\tilde{b}}$ between the two states after three applications of Kraus operators to each of the data qubits, as given by Eq. (6.11d).

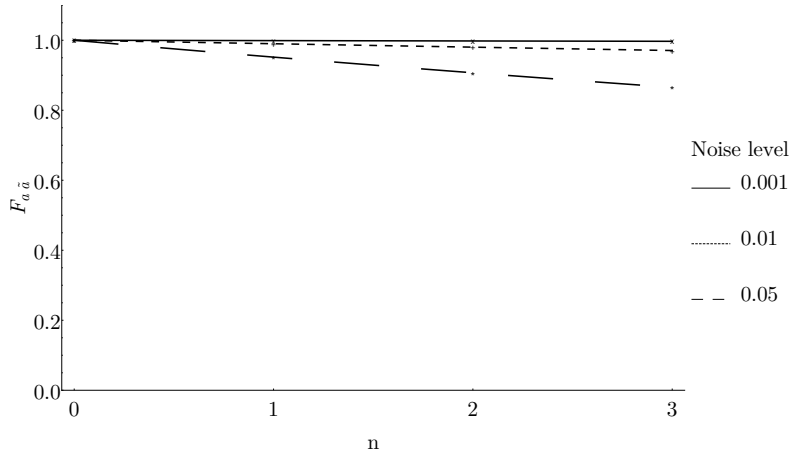
present the results when training the device at one noise level, and validating at another. We see that even with high levels of training noise the optimiser converges onto good parameters, as we find comparably low loss levels when validating those parameters trained at a high noise level with low noise in the validation step. This shows that for this task, the training steps are robust to a noisy training routine. Also here we find that when validating at noise levels seen in current devices, $p_{2q} = 0.01$,

the average loss does not increase above 0.25, which would be acceptable to use for state discrimination.

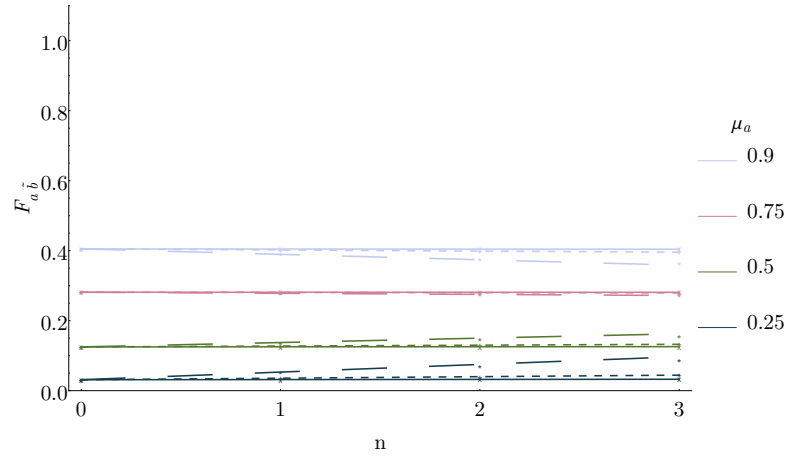
In order to provide an understanding of the numerically found changes of the loss with noise, in what follows we present a simple model that can describe the results. It is based on the notion that a larger overlap between the states to be discriminated generally makes discrimination more difficult. As outlined in Sec. 6.1, with our chosen set of states this overlap can be tuned by setting the value of a, and is equal to $a/\sqrt{2}$. We can therefore systematically evaluate the effect of noise on the discrimination for increasing overlap by increasing μ_a, and the results are shown in Figure 6.9. The loss increases for larger μ_a for all levels of noise. At high noise levels and high μ_a, some runs are performing even worse than the random output limit (0.67), but on average the loss remains well below that value. In general we conclude that the tolerable levels of noise depend on the overlap between the states, where small overlap allows the states to be discriminated even for higher noise in the quantum computer.

For large noise in the system the difference in loss between higher and lower values of μ_a is significantly reduced when compared to the low noise case. This seems to indicate that the noise on average reduces the difference between states as these pass the circuit, and hence effectively increases the averaged overlap. This effect can be illustrated for the ideal case of $a = 0$ and no noise, where discrimination can in principle be perfect since the states are orthogonal. However, in presence of noise there is a probability that a state is perturbed as the circuit is applied to it, and hence orthogonality between states is lost. This results in a certain probability of erroneous detection.

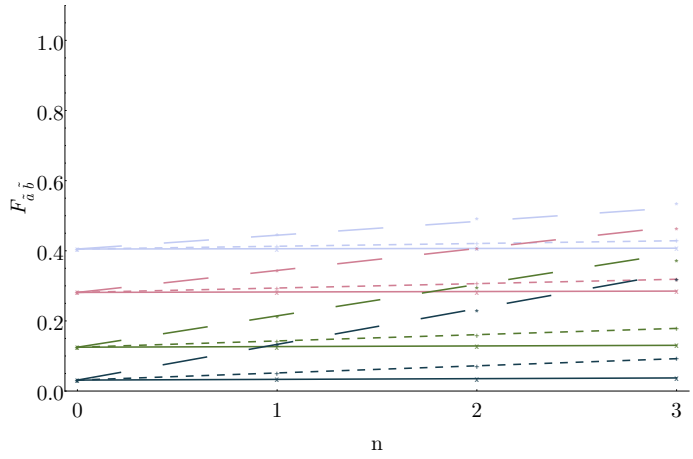
In order to estimate this effect on a semi-quantitative level for our used circuit, shown in Figures 6.2 and 6.3b, we note that in absence of noise the role of the data qubits is only to store the state $|\psi_{\text{in}}\rangle$, which then controls the state of the measurement qubits. In presence of noise the three noisy two-qubit gates applied to each of the two data qubits will perturb $|\psi_{\text{in}}\rangle$ during the processing of the circuit, which in turn will affect the measurement qubits via the control operation and hence



(a)



(b)



(c)

Figure 6.10: Fidelities as function of the number of applied noise channels, n, between (a) the same states with noise applied to one state ($F_{a\tilde{a}}$), (b) the two different states with noise applied to one state ($F_{a\tilde{b}} = F_{b\tilde{a}}$), and (c) with noise applied to both states ($F_{\tilde{a}\tilde{b}}$). Markers show the calculated numeric fidelities using Eq. (6.10), lines show the low order expansions given by Eq. (6.11). The low order expansion agrees well with the numerical results for all cases.

the outcome of the state discrimination. We therefore approximately model the effect of noise on the state discrimination by determining how much the repeated application of noisy channels at each gate affects a given $|\psi_{\text{in}}\rangle$, without considering the presence of the measurement qubits.

In what follows we quantify how the application of noise channels affects each state and its overlap with the state to be discriminated from. The quantum states with noise applied are represented by density matrices, so that the overlap between two states represented by the density matrices ρ and σ, respectively, is described by the fidelity, F, given by [172]:

$$F = \text{Tr}\left[\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}\right]^2. \tag{6.10}$$

Of particular interest here is the fidelity between a pure state $|\psi_a\rangle$ entering the circuit and its modified form due to the application of noise after three two-qubit gates. We denote this as $F_{a\tilde{a}}$, where the tilde on the second subscript indicates that the second state is the one where noise was applied. We use an analogous notation the other relevant quantities, which are $F_{b\tilde{b}}$, $F_{b\tilde{a}}$ and $F_{\tilde{b}\tilde{a}}$. We can compute these quantities numerically, but given the rather cumbersome form of Eqn. (6.10) it is difficult to relate the results to the fundamental parameters of the noisy discrimination process. We therefore provide a lowest order expansion of these terms in p, which we expect to be close to the exact results since we are only dealing with small p. We apply the noise model in Eqs. (3.3-6.9), to the states a and b in Eqs. (6.1) and (6.2), and obtain to first order in p.

$$F_{a\tilde{a}} = 1 - n\,p\,, \tag{6.11a}$$

$$F_{b\tilde{b}} = 1 - \frac{3n\,p}{2}, \tag{6.11b}$$

$$F_{a\tilde{b}} = F_{b\tilde{a}} = \frac{\mu_a^2}{2} + \frac{n\,p}{2}\left(1 - 2\mu_a^2\right), \tag{6.11c}$$

$$F_{\tilde{a}\tilde{b}} = \frac{\mu_a^2}{2} + n\,p\,\left(1 + \frac{\mu_a}{\sqrt{2}} - 2\mu_a^2 + \sqrt{1 - \frac{\mu_a^2}{2}}\right), \tag{6.11d}$$

where $n = 0\ldots3$ is the number of noisy channel applications, μ_a is the mean value

of a and p is the noise probability. Within this expansion order the fidelities are linear in both p and n, and the expansion coefficients are simple functions of μ_a.

For the noiseless case ($p = 0$), $F_{a\tilde{b}} = F_{\tilde{a}\tilde{b}} = \mu_a^2/2$, which corresponds to the absolute value squared of the overlap between the a and b states.

The results for the fidelities for increasing n are shown in Figure 6.10. It can be seen that the numerical results obtained directly with Eqn. (6.10) are captured rather well with the the analytical low order expansion in Eqs. (6.11a-6.11d). The value of $F_{a\tilde{a}}$ decreases with each application of a noisy channel, showing that the purity of the state degrades as the noise channels act on the state. The value of $F_{a\tilde{b}}$ increases with n for lower values of μ_a, where the noise acts to increase the fidelity between the states, while it decreases with n for higher values of μ_a, where the noise reduces the orthogonal component in $|\psi_a\rangle$. In Figure 6.10c the fidelity is plotted when the noise channel is applied to both states ($F_{\tilde{a}\tilde{b}}$), and it always increases with n. $F_{\tilde{a}\tilde{b}}$ is the relevant quantity the influence of noise on the loss: the application of the noisy circuit on the originally pure input states causes them to degrade into mixed states. The states to be discriminated are therefore not the input pure states any more, but these noisy states, and for each application of the noise channel they become harder to discriminate.

The minimal loss achievable in quantum state discrimination is generally a function of the fidelity between the two input states [173], [174]. The exact relation depends on the cost function that is minimised, and is only known analytically for a few special cases, such as minimal error discrimination or unambiguous discrimination [173], [174]. In this section we consider the case where the rate of inconclusive results and erroneous results is minimised simultaneously ($\alpha_{\text{err}} = \alpha_{\text{inc}} = 40$).

Before estimating the effect of noise we therefore need to determine the functional relation between the fidelity and the loss in the noiseless case for our circuit. In the top left panel of Fig. 6.9 we show the results for $F_{\tilde{a}\tilde{b}}(p = 0) = F_{ab}$ as red stars for each μ_a. One can see that the F_{ab} is approximately equal to the loss for all μ_a, so that to a good approximation for our circuit we can fit the relation as Loss $= F_{ab}$. In general the lower bound of the optimal theoretical loss is found in the minimum

error discrimination setting, where there is no inconclusive measurement. In our optimisation we include also the inconclusive measurement, the probability of which is minimised in the unambiguous setting, which gives an upper bound on the loss. Our minimised cost function is a combination of these settings, which simultaneously minimises errors and inconclusive results, and our relation for the loss in fact lies in between the optimum values in each of these boundary settings.

We can now verify the validity of our model for the effect of noise on the loss. To this aim we calculate $F_{\tilde{a}\tilde{b}}$ from Eqn. (6.11d) for $n = 3$, which corresponds to the number of entangling gates applied in our circuit to each data qubit. These results are presented in Figure 6.9 for the panels with $p \neq 0$.

One can see that $F_{\tilde{a}\tilde{b}}$ agrees rather well with the loss also in the presence of noise. This validates our model that the effect of noise on the state discrimination is mainly determined by the noise-induced increased overlap between the states as they are processed in the circuit. In particular for the highest noise the model captures well the fact that the effect of noise is large for small μ_a, while it is reduced for larger μ_a. Eqn. (6.11d) therefore allows us to estimate the minimal loss achievable with our circuit for a given p and μ_a.

Finally, we investigate the effect of the noise during training on the actual values of the optimised parameters in the circuit. In Figure 6.11 we present the distribution of θ_{10} for different values of noise. The values of θ are all taken modulo 2π, and at zero validation noise to remove the effect of validation errors on the loss. We see that the range of angles converged upon increases as the noise in the circuit increases. Some values become stuck at high loss, and there can be different values for the minimal loss parameters. The increase in noise seems to change not just the final loss, but the parameters found that minimise loss. We cannot rule out the correlation between different parameters as the noise level changes. Combined with what we see in Figure 6.8, that good parameters are still found at higher noise levels, we may conclude that noise in the circuit can push the optimiser out of local minima, so that it can find some other local minima at lower loss.

From the results presented here we see that this algorithm performs well in

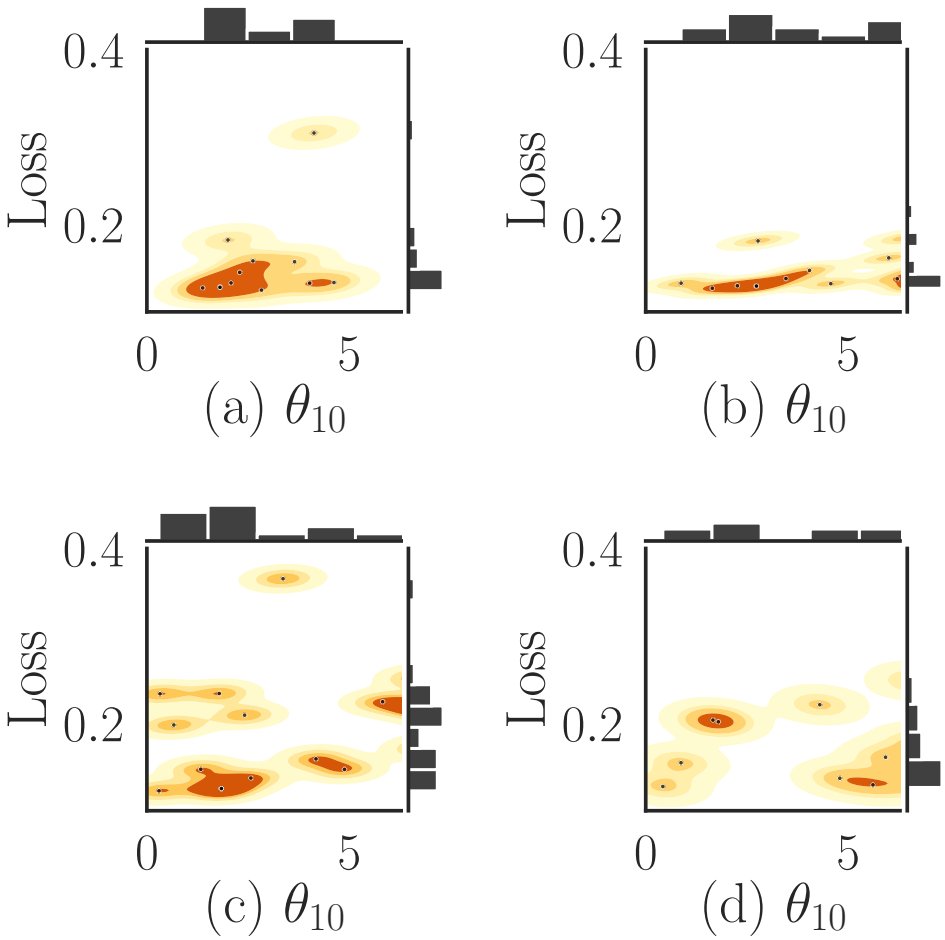


Figure 6.11: The distribution of loss and θ_{10} obtained at different noise levels. Here we can see the effect of noise on the values of θ_{10} that the optimiser converges to. We only show a single representative parameter θ_{10}, since we have a approximately similar behaviour for all other parameters. This is shown at different noise levels, **(a)** 0.0, **(b)** 0.001, **(c)** 0.01, **(d)** 0.05.

the presence of noise in the training and validation steps (Figure 6.7), and that parameters found on a noisy device work well when validated on a device with low noise (Figure 6.8). When calculating parameter gradients on a noisy quantum device, reducing the number of parameters has a positive effect, as shown in Figure 6.5 and Figure 6.6.

6.3 Discussion

In showing that a QNN for the task of state discrimination can be trained on a noisy device, we hope to add to the evidence that QML algorithms are best used on quantum data, especially in the near term. The QNN here has the advantage over a classical scheme, which requires measurement of the states, as corellation between the states can be learned in the trained circuit [99]. However, the introduction of noise does contradict some recent results in this field showing that noise leads to worse training outcomes [175]. In completing this work , we removed gates and parameters to beneficial effect (fewer gates mean fewer sources of infidelity); however, recent work in [176] demonstrates a critical overparameterisation is beneficial to training and convergence of QNNs. The tension between the two directions, of overparameterisation to improve training, and reducing gate count to overcome noise may only be overcome by improvements in hardware, to make the additional noise negligible.

Chapter 7

Non-Linearities in Quantum Neural Networks

7.1 Introduction: A QNN Primitive

Etymology of the term neural network reveals where the field initially took its largest inspiration from. The first proposals for algorithms that could learn were based on initial investigations of the human brain, e.g. the perceptron [40]. There was limited success with the very first neural network algorithms, and it seems that the later success of machine learning required the huge increase in scale of the amount of computation and data that could be used, including the invention of a novel type of hardware, the Graphical Processing Unit (GPU). In the sub-field of quantum computation that is Quantum Machine Learning and Quantum Neural Networks, we are yet to see a universal primitive, or building block, for these algorithms. The field is yet to settle upon an agreed quantum perceptron.

In this chapter we will discuss some of the proposals for quantum perceptrons, and we will see how researchers have tried to incorporate non-linearity into quantum neural networks. We have developed a package for simulating these circuit ansatze[1] [177] which we will use to compare the ability of these circuits to learn a simple non-linear function. But first we must begin with an admission of a mistake, a *mea culpa* for something this author stated in a previous paper.

[1] `https://github.com/DuckTigger/non-linearities`

7.2 A *mea culpa*

The work in this chapter was instigated thanks to a discussion [178] regarding a claim the author made in [135], that introduction of a mid-circuit measurement introduces a non-linearity into the quantum circuit. This claim is not true, and this can be verified by noting that a controlled gate commutes with measurement of the control qubit [43, p.188], therefore the controlled gate can be commuted through the measurement, and the circuit must implement a linear operation.

Classical neural networks have been used to solve some extremely difficult problems, such as image recognition [109], processing and generating natural language such as human speech [108], and generating novel images [107]. All of the classical neural networks mentioned previously use deep networks, including many layers of nodes in their structure. Non-linearities are essential to deep networks, as they are used in the activation function of a neuron and any extremely deep network without non-linear activation functions can be replaced by a single layer [106].

Quantum Mechanics is linear, operations in quantum mechanics are described by unitary transformations of quantum states, themselves described by vectors. If quantum mechanics admitted non-linear operations it would lose time-reversal symmetry, strong evidence that we cannot have a quantum mechanical non-linear operation. However, as we observe non-linear effects in the classical world it must be possible to simulate non-linear effects by embedding into some larger Hilbert space, which some of the techniques discussed in this chapter will do. It is the connection between the power of deep classical neural networks and non-linearities that has lead to investigation into introducing non-linearities into quantum neural networks. To this end, there have been many proposals for introducing non-linearities in quantum neural networks, some which do not, some that are only a trivial non-linearity, and some which can produce a non-linear function at the cost of more quantum resources. The work in this chapter will investigate some of these proposals, differentiate between them, and provide some numerics demonstrating the ability for the circuit proposals to ‘learn’ a linear and a non-linear function. The numerics for this work was completed by a module in Python which has been open-sourced [177], with the

ability to easily add new proposals for non-linearities.

7.3 What is a non-linearity?

Before continuing it is probably best to define what we mean by a non-linearity. A linear transformation maps between two vector spaces preserving vector addition and scalar multiplication. That is, for vector spaces V, W over the field K the function $f : V \to W$ is a linear map if for any two vectors $\mathbf{u}, \mathbf{v}$ and scalar $c \in K$ [179] :

$$f(\mathbf{u}+\mathbf{v}) = f(\mathbf{u}) + f(\mathbf{v}), \tag{7.1}$$

and

$$f(c\mathbf{u}) = cf(\mathbf{u}). \tag{7.2}$$

A non-linear transformation is one which does not preserve vector addition or scalar multiplication, for example, $f(x) = x^2$ is a non-linear transformation as $f((a+b)) = a^2 + b^2 + 2ab$ for $a, b \in \mathbb{R}$.

When we discuss non-linearities in quantum neural networks there is an additional distinction to be made, between non-linear transformations on the input data and non-linear transformations on the quantum state amplitudes.

7.3.1 Non-Linear Transformations on Input Data

A non-linear transformation on the input data to a Quantum Neural Network (QNN) is the same transformation that a classical neural network employs. For ease of exposition, we will restrict ourselves to integer numbers, although the extension to floating point numbers is the same as in a classical computer. If we encode the binary value of an integer in a set of qubits, including the sign qubit, we have a quantum state which represents the integer. To represent the integer -4, where the most significant qubit is the sign qubit, we prepare the state $|1100\rangle$.

The non-linear transformation we consider here is the Rectified Linear Unit

(ReLU) function, which performs the function:

$$\text{ReLU}(x) = \begin{cases} 0, & \text{if } x < 0 \\ X, & \text{otherwise.} \end{cases} \tag{7.3}$$

This can also be seen in Figure 7.1

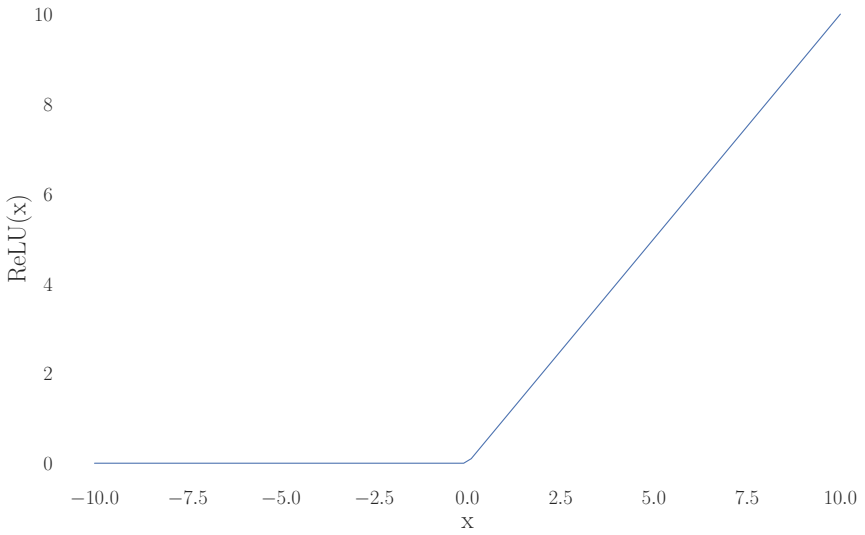


Figure 7.1: The ReLU non-linear activation function.

In some cases, we would like the quantum device to implement a non-linear function on the input data, encoded into qubits. For example, to implement the ReLU function on integers encoded in binary a function should output the state vectors as given in Table 7.1.

Table 7.1: ReLU function on encoded integers

x	$\|x\rangle$	ReLU(x)	ReLU ($\|x\rangle$)
-2	$\|110\rangle$	0	$\|000\rangle$
-1	$\|101\rangle$	0	$\|000\rangle$
0	$\|000\rangle$	0	$\|000\rangle$
1	$\|001\rangle$	1	$\|001\rangle$
2	$\|010\rangle$	2	$\|010\rangle$

It is simple to encode this function in a quantum circuit: a Toffoli gate, modified

to be controlled on 0 on the sign bit will produce exactly this output; given in Figure 7.2. Implementing the circuit to perform the non-linearity as an oracle is

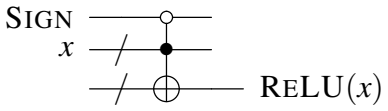


Figure 7.2: The single gate to implement the ReLU function on integers encoded in binary on qubits.

how the authors of [180] perform non-linear operations on the input to a QNN, which they extend by demonstrating an oracle can perform the transformation on amplitude-encoded data.

Given enough qubits and a deep enough circuit a quantum device can perform any transformation on encoded input data as a classical computer can, with polynomial overhead [43, p. 29]. We can show this in principle as the Toffoli gate is universal for reversible classical computation [181], i.e. classical computation with some garbage bits, and we can implement a Toffoli gate in a quantum computer.

7.4 Non-Linear Transformations on Quantum State Amplitudes

As discussed, evolution in quantum mechanics is done by unitary operations, so transformations on the state vector amplitudes must be linear. Nonetheless, in this section we will discuss two proposals [182], [183] for implementing non-linear transformations on the amplitudes of a quantum state. Neither of these proposals 'break' quantum mechanics or propose unphysical operations, instead they utilise a larger Hilbert space (by introducing ancilla qubits) and post-selection to implement what seems like a non-linear transformation, when the ancilla qubits and non-selected results are thrown away.

7.4.1 Quantum Singular Value Transform

The Quantum Singular Value Transform (QSVT) [183] is the first of the non-linear transformations on state amplitudes we will discuss here. In the QSVT we assume access to a state preparation oracle U and $U^\dagger$, and the controlled versions of both

operations:

$$U|0\rangle \rightarrow \sum_{i=1}^{N} c_k |k\rangle. \tag{7.4}$$

Here k is the index of the value c_k stored at that amplitude. This is in general a Quantum Random Access Memory (QRAM) operator, but it is possible to work in situations where U is efficient to prepare [184].

The task of a performing non-linearity on quantum amplitudes is: given two non-linear operations P, Q, where linearity is as defined in Equations 7.1 and 7.2, and the input as prepared in Equation 7.4, a circuit successfully implementing non-linear operations on quantum amplitudes is able to prepare

$$\frac{1}{\sqrt{c}} \sum_{k=1}^{N} \left(P\left(x_k\right) + Q\left(y_k\right)\right) |k\rangle . \tag{7.5}$$

Here x_k, y_k are the real and complex components of c_k, and c is a normalisation factor.

The QSVT method is actually only part of this algorithm, introduced in [185] and is able to perform non-linear transformations on the singular values of a matrix. In [183] they combine this with a block-encoding strategy which encodes the real (complex) parts of the quantum state as the singular values of a matrix.

It is argued in [185] that the QSVT underlies many of the qauntum computing primitives, including Grover's search [2] and the Harrow, Hassidim, and Lloyd [101]; but not the quantum Fourier Transform (e.g. Shor's algorithm [1]).

A block encoding unitary is one which can prepare a matrix A, which may not be Hermitian, in a smaller sub-space of the Hilbert space:

$$A = \left(\langle 0|^{\otimes a} \otimes I\right) U \left(|0\rangle^{\otimes a} \otimes I\right), \tag{7.6}$$

$$U = \begin{pmatrix} A & . \\ . & . \end{pmatrix}. \tag{7.7}$$

meaning that if we act with U, measure the ancilla register a and post-select for the $|0\rangle^{\otimes a}$ outcome, we will have a state $\propto A|\psi\rangle$ in the other register.

In [183] they develop a method to block-encode the amplitudes of a quantum state, allowing for the QSVT deveoloped in [185] to apply the non-linear transformation to the amplitudes. Similar to Eqn. 7.6, we prepare matrices $\tilde{G}$, $\left(\tilde{G}'\right)$ with eigenvalues encoding the real (imaginary) part of the state amplitude:

$$\left(\langle 0|\otimes I_{2n+1}\right)\tilde{G}\left(|0\rangle\otimes I_{2n+1}\right)=\sum_{k=1}^{N}x_k|\phi_k\rangle\langle\phi_k|+\ldots, \tag{7.8}$$

$$\left(\langle 0|\otimes I_{2n+1}\right)\tilde{G}'\left(|0\rangle\otimes I_{2n+1}\right)=\sum_{k=1}^{N}y_k|\phi_k\rangle\langle\phi_k|+\ldots. \tag{7.9}$$

With the amplitudes encoded as eigenvalues, we can use the QSVT to perform a non-linear operation on them.

The non-linear transformations P and Q are assumed to be approximated by polynomial functions which obey:

$$\forall x\in[-1,1]:|P_{\Re}(x)|\leq\frac{1}{4}, \tag{7.10}$$

meaning we can use the QSVT [185] to compute a block encoding of $P(A/\alpha)$, where α is a multiplicative error. We use a classical computer to compute the description of the circuit in time $\mathcal{O}\left(\text{poly}\left(d,\log\left(\frac{1}{\delta}\right)\right)\right)$, where d is the degree of the polynomial, and δ is the approximation error of the polynomial. This scaling comes from the requirement of finding the roots of the degree-d polynomial in the classical algorithm to compute the circuit.

The algorithm to build this circuit used ideas from quantum signal processing [186], where we want to know what unitaries can be built from the gate sequence

$$e^{i\phi_0\sigma_z}W(x)e^{i\phi_1\sigma_z}W(x)\ldots, \tag{7.11}$$

with

$$W(x)=e^{i\arccos(x)\sigma_x}=\begin{bmatrix}x & i\sqrt{1-x^2}\\ i\sqrt{1-x^2} & x\end{bmatrix}. \tag{7.12}$$

Theorem 3 of [186] shows that there exist rotation angles ϕ_j such that

$$e^{i\phi_0\sigma_z}\prod_{j=1}^{k}\left(W(x)e^{i\phi_0\sigma_z}\right)=\begin{bmatrix}P(x) & iQ(x)\sqrt{1-x^2}\\ iQ^*(x)\sqrt{1-x^2} & P^*(x)\end{bmatrix} \tag{7.13}$$

if the polynomials P,Q obey

1. $\deg(P)\leq k, \deg(Q)\leq k-1$
2. Parity of P is $k$mod2, parity of Q is $k-1$mod2
3. $\forall x\in[-1,1]: |P(x)|^2+\left(1-x^2\right)|Q(x)|^2=1.$

The following two theorems in [185] show how to find the values of ϕ in $\mathcal{O}\left(\text{poly}\left(d,\log\left(\frac{1}{\delta}\right)\right)\right)$.

The circuit W is shown in Figure 7.3, where U is the state preparation oracle in Eqn. 7.4. The application of the non-linearity is performed by the circuit shown

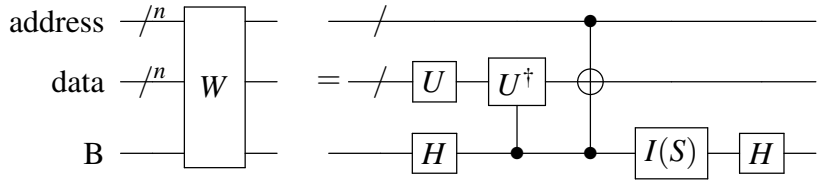


Figure 7.3: The W operator used to implement block encoding of the state amplitudes. After the Toffoli operation, the S gate is applied if we are operating on the imaginary part. On the real part, no operation is performed.

in Figure 7.4, with the $\mathcal{P},\mathcal{Q}$ unitaries fond via the algorithm given in [185]. This

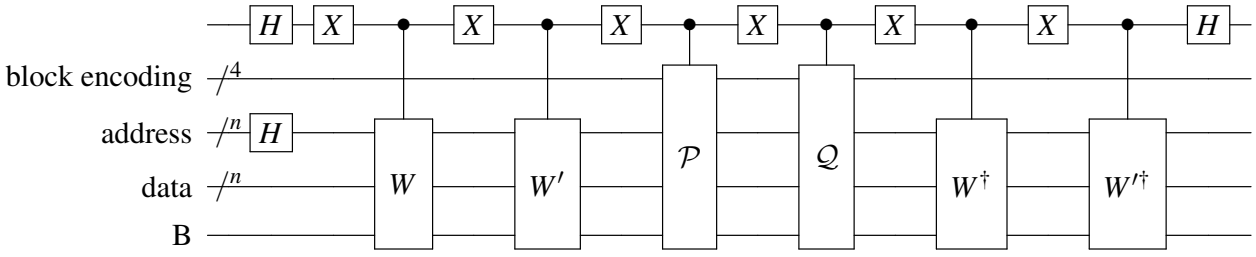


Figure 7.4: The full circuit to perform the non-linearity on state amplitudes, with a post-selected measurement on the ancilla and block-encoding qubits.

circuit produces the state

$$\frac{1}{8\gamma\sqrt{N}}\sum_{k=1}^{N}\left(P'(x_k)+Q'(y_k)\right)|0\rangle^{\otimes 5}|k\rangle_{\text{adrs}}|0\rangle_{\text{data, B}}+\ldots, \tag{7.14}$$

which will produce the non-linear operation on the address qubits given a post-selected measurement of $|0\rangle$ on the first five qubits, which occurs with probability $\frac{1}{64\gamma^2 N}\sum_k \left|P'(x_k)^2 + Q'(y_k)^2\right|$.

7.4.2 Weighted States

A different framework for implementing non-linear operations on quantum devices is the weighted states framework [182]. Whilst it does not use block-encoding or the QSVT, it still relies upon expanding the Hilbert space via ancilla qubits and post-selection of measurement outcomes. This method relies upon the fact that we do not have access to the amplitudes of a quantum state, only the measurement outcomes. Therefore, for a target (non-linear operation applied) output state τ we wish to calculate $\langle O\rangle_\tau = \mathrm{Tr}[\tau O]$ for any measurement operator O, this does not mean we need to construct τ on a single quantum device. In this framework we introduce the ancilla state σ and measurement operator M which are dependent on the function to be implemented. We define the input register I, the ancilla register A, and the input state ρ_I^{in}, we then evolve the input state and ancilla by a unitary operation, U, giving:

$$\rho_{\text{out}} = U\left(\sigma_A \otimes \rho_I^{\text{in}}\right)U^\dagger. \tag{7.15}$$

This output state is then re-grouped into three registers: system, S, environment, E, and garbage, G. We then design a measurement operator, M, acting on the environment register such that the desired output $\mathrm{Tr}[\tilde{\tau}O]$ onto the state ρ^{out}:

$$\mathrm{Tr}[\tilde{\tau}O] = \mathrm{Tr}_{SEG}\left[\rho^{\text{out}}(O \otimes M_E \otimes \mathbf{I}_G)\right] \forall O. \tag{7.16}$$

We can delay measurement of O, leaving $\tilde{\tau} = \mathrm{Tr}[\rho^{\text{out}}(\mathbf{I}_S \otimes M_E \otimes \mathbf{I}_G)]$ to be used as the input to another algorithm.

It is the design of σ_A, U, M_E that implements the non-linear transformation, in Figure 7.5 depicts the circuit for performing the Quantum State Polynomial (QSP). Before measurement, this circuit prepares the state

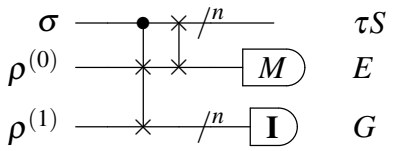


Figure 7.5: The circuit used to implement the quantum state polynomial in the weighted states framework. The second SWAP is inserted to maintain consistency with notation, but can easily be implemented in post-processing.

$$\rho_{=}^{\text{out}} \sum_{ii'jj'} \sigma_{00}\rho_{ij}^{(0)}\rho_{i'j'}^{(1)}|i0j\rangle\langle i'0j'| + \sigma_{01}\rho_{ij}^{(0)}\rho_{i'j'}^{(1)}|i0j\rangle\langle j'1i'| \tag{7.17}$$

$$+ \sigma_{10}\rho_{ij}^{(0)}\rho_{i'j'}^{(1)}|j1i\rangle\langle i'0j'| + \sigma_{11}\rho_{ij}^{(0)}\rho_{i'j'}^{(1)}|j1i\rangle\langle j'1i'|, \tag{7.18}$$

which when measured with the measurement operator M, produces the weighted state

$$\tilde{\tau} = \sigma_{00}M_{00}\text{Tr}\left[\rho^{(1)}\right]\rho^{(0)} + \sigma_{11}M_{11}\text{Tr}\left[\rho^{(0)}\right]\rho^{(1)} \tag{7.19}$$

$$+ \sigma_{01}M_{01}\rho^{(0)}\rho^{(1)} + \sigma_{10}M_{10}\rho^{(1)}\rho^{(0)}. \tag{7.20}$$

Through choice of σ and M we can produce states of the form $\alpha_{00}\rho^{(0)} + \alpha_{11}\rho^{(1)} + \alpha_{01}\rho^{(0)}\rho^{(1)} + \alpha_{10}\rho^{(1)}\rho^{(0)}$, which can clearly not be produced on a 'vanilla' quantum circuit.

7.5 Trainability of non-linear circuit proposals

We can investigate the Quantum Neural Network primitive circuit proposals here, with the simple test of replicating the ReLU function on integers encoded in the qubits. This test is proposed as it is simple, yet can be revealing for the effectiveness of the ansatz in training to a non-linear function which is possible in a quantum circuit, see Figure 7.2, yet non-trivial, as we shall see. We encode the integers in a binary representation, with a sign qubit, as in classical computing. To compare the usefulness of the ansatze given above, we have trained all circuits to replicate the ReLU function, in Figure 7.1. If we wish to use one of these circuits as a QNN primitive, it should be simple enough that duplicating the circuit many times does not require an onerous amount of training time, qubit number, or long coherence times. We also desire a circuit element that, when combined with other similar units,

is expressive enough to learn the required function. To test for these properties, we have attempted to train all of the ansatze given above to replicate the ReLU function, encoding the integer in the amplitude of the qubits, in a similar manner to integers encoded in classical bits, with a sign bit, as given in Table 7.1.

In the remainder of this section we will present the results of training the above QNN primitives to the ReLU function, with some commentary on the ease of training and the suitability of the proposed ansatz. We first attempt to train the ansatz to the linear function (i.e. replicate the identity operator) as a 'sanity check' to see if the ansatz is trainable at all. The python program to simulate these results is available at [177].

7.5.1 Toffoli Oracle

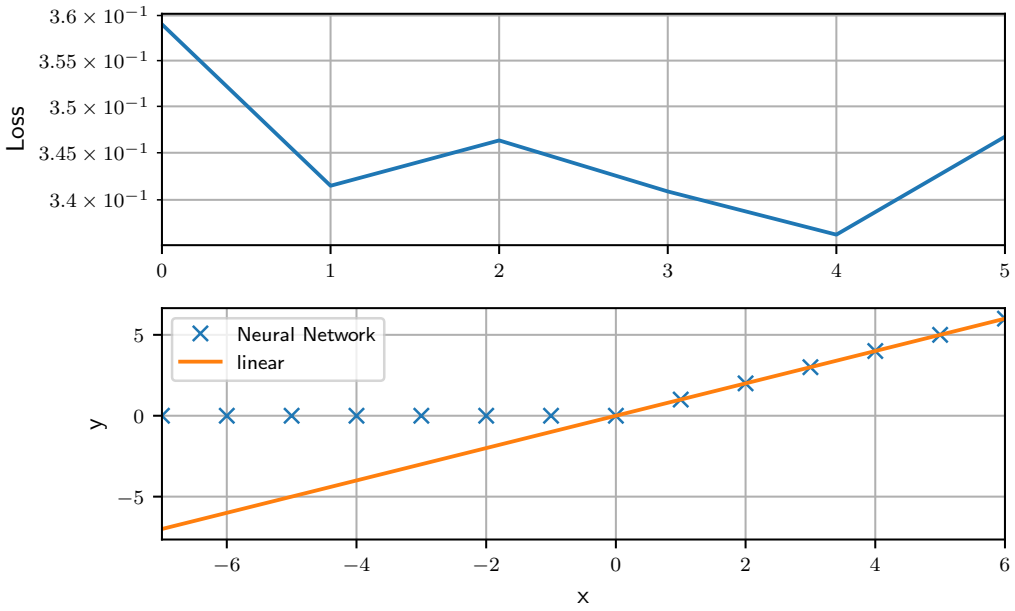
There is no training in this example, we present the Toffoli oracle in Figure 7.6 to demonstrate that a quantum circuit is able to represent the ReLU function. However, as there are no parameters to train we cannot reproduce the linear function. We can place any classical function as an oracle to the quantum device, but this will be costly, and may not bring about any advantages.

7.5.2 Dropout (mid-circuit measurements) Ansatz

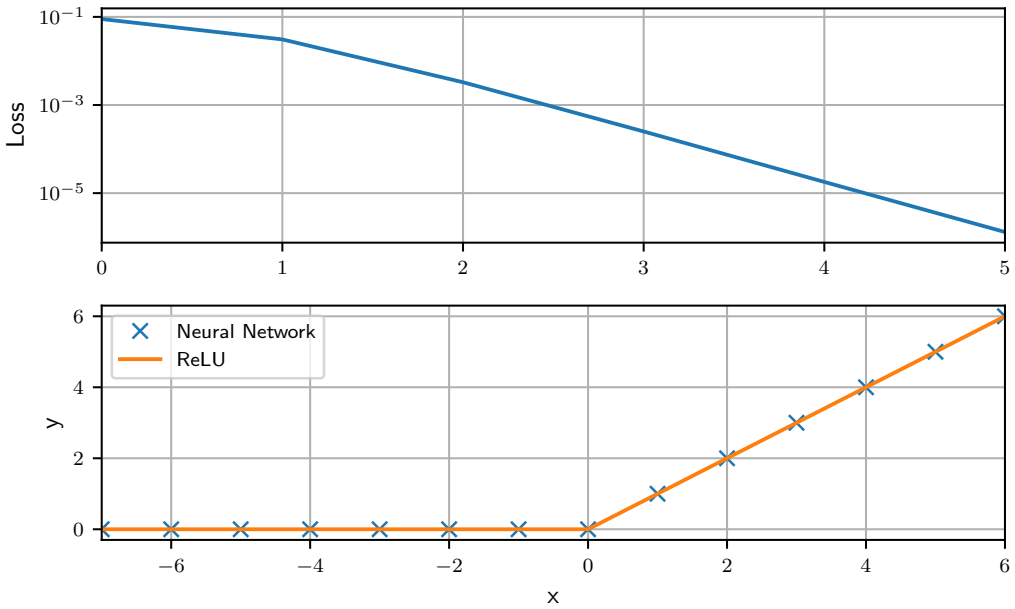
Here we see the results for training an ansatz with mid-circuit measurements, also known as a dropout ansatz from the analogy with dropout in classical neural networks. This is the ansatz we used in Chapter 6. We see in Figure 7.7 that we can learn the linear function quite well, but that reproducing the ReLU cannot be achieved.

7.5.3 Hypothetical Measurements

We expect that the hypothetical measurements will perform much the same as the dropout ansatz, due to the similarity between them. We observe this in Figure 7.8, the circuit fails to learn the ReLU function, but it does appear to be less noisy than the dropout ansatz.

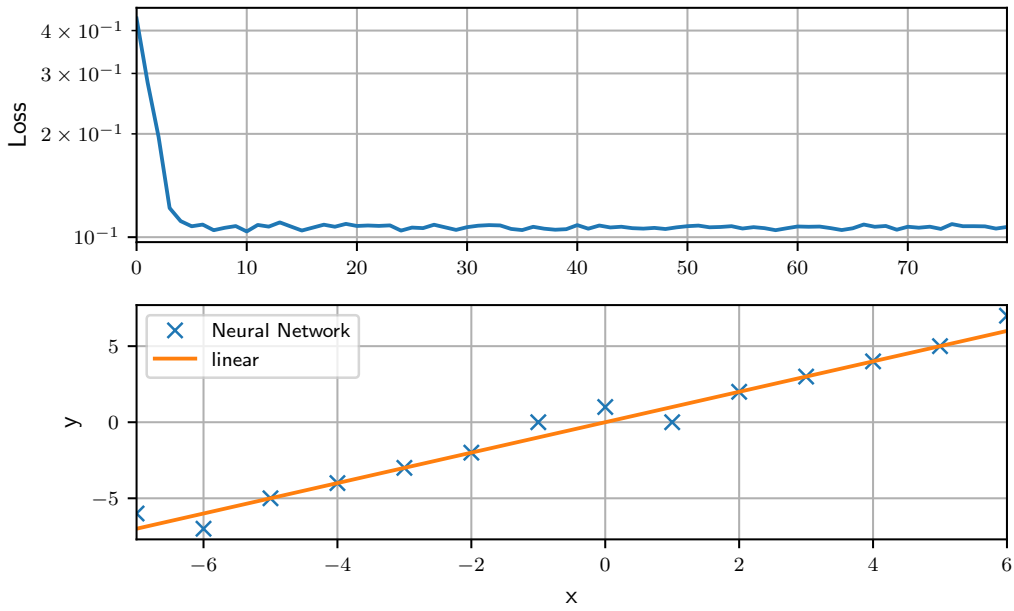


(a) Linear model

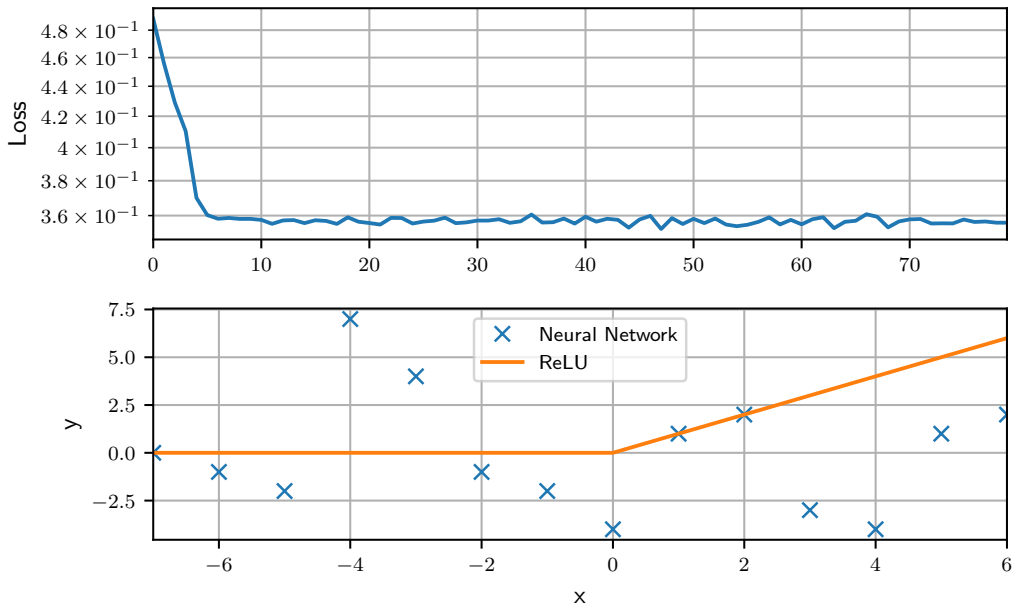


(b) ReLU model

Figure 7.6: Using a Toffoli gate as an oracle, e.g. the circuit in Figure 7.2 can reproduce the non-linear ReLU function, but it cannot be trained to output the linear function.

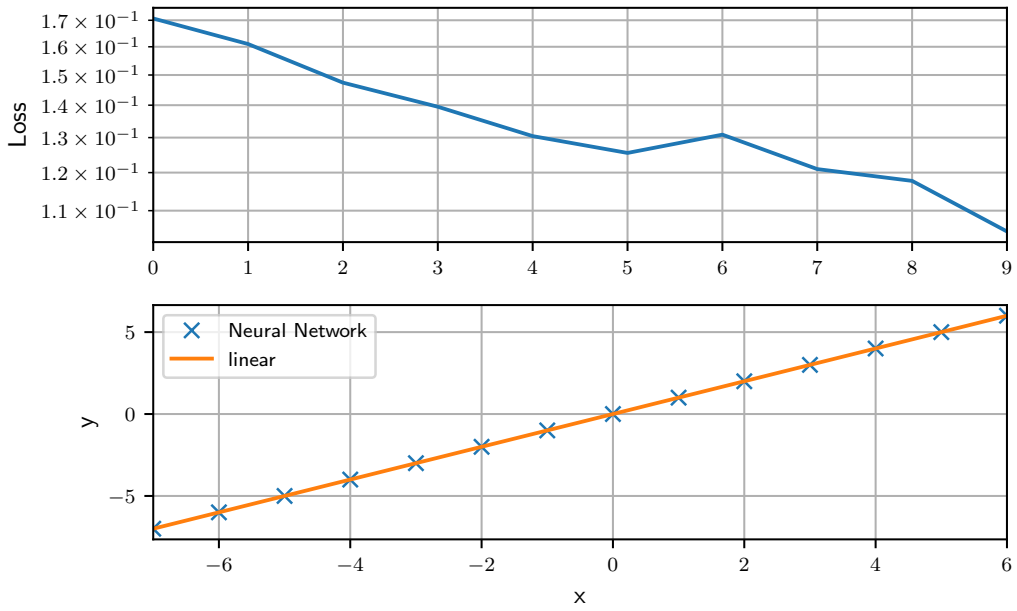


(a) Linear model

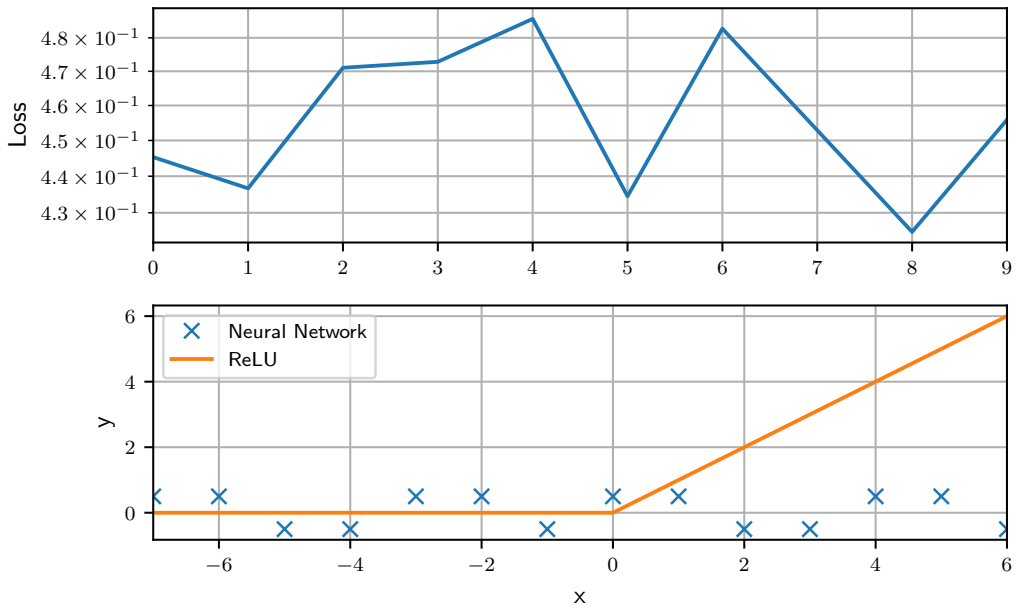


(b) ReLU model

Figure 7.7: The dropout ansatz being trained on a linear and non linear model. The model can capture the linear model, but fails with the non-linearity. In both models the loss function has converged quickly, which is good for the linear model, but suggests that this ansatz is not expressive enough to learn the non-linear model.



(a) Linear model



(b) ReLU model

Figure 7.8: The hypothetical measurements ansatz being trained on a linear and non linear model. The model can capture the linear model, but fails with the non-linearity. It learns the linear model well, with a smoothly decrealing loss function, but the loss function of the ReLU model is not well behaved.

7.5.4 Non-Linear Parameters

The proposal of adding the non-linearity at the parameters it studied here. In this proposal we apply a non-linear function to the gate parameter, after the gradient update has been given by the optimiser. In Figure 7.9 we see that the circuit can learn the linear model well, similar to a simple Pauli rotations model, but still fails to learn the ReLU function. In essence, this is because the unitary operations on the state are no different to those in the Pauli model.

7.5.5 Interleaved Ansatz

The interleaved ansatz, as proposed in [187] interleaves rotation gates encoding the input and rotation gates as parameters. This ansatz therefore encodes the input differently to the others considered in this chapter. An example in a single qubit is given in Figure 7.11.

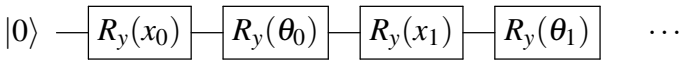


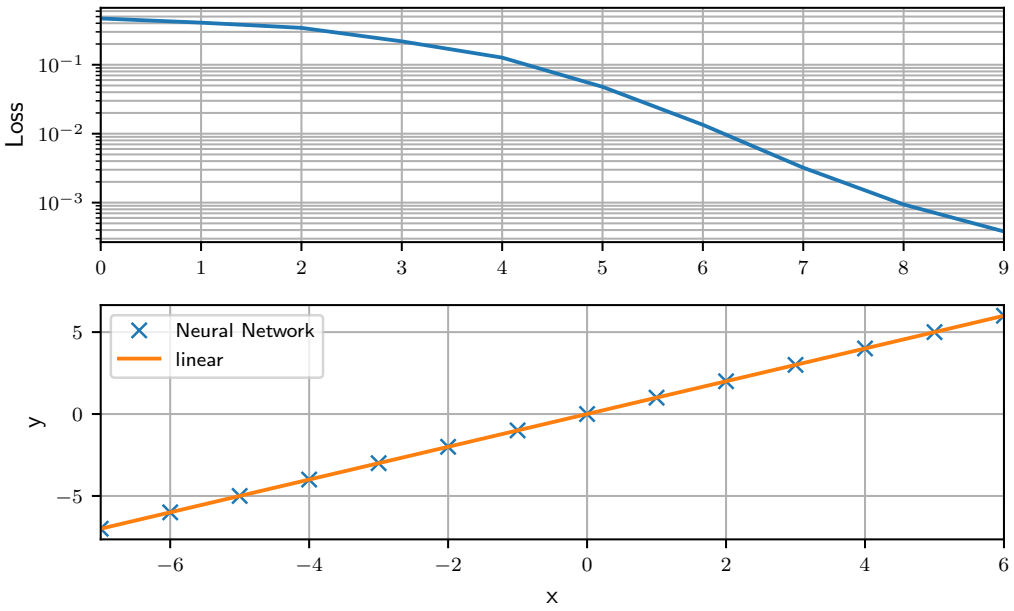
Figure 7.10: The circuit used in the interleaved ansatz, from [187], where x_i are elements of the input data and θ_i are the parameters. This can be extended to more qubits.

Considering the $0,0$ element of the final state, we see that the action of this ansatz is to apply a sinusoidal function to the input data,

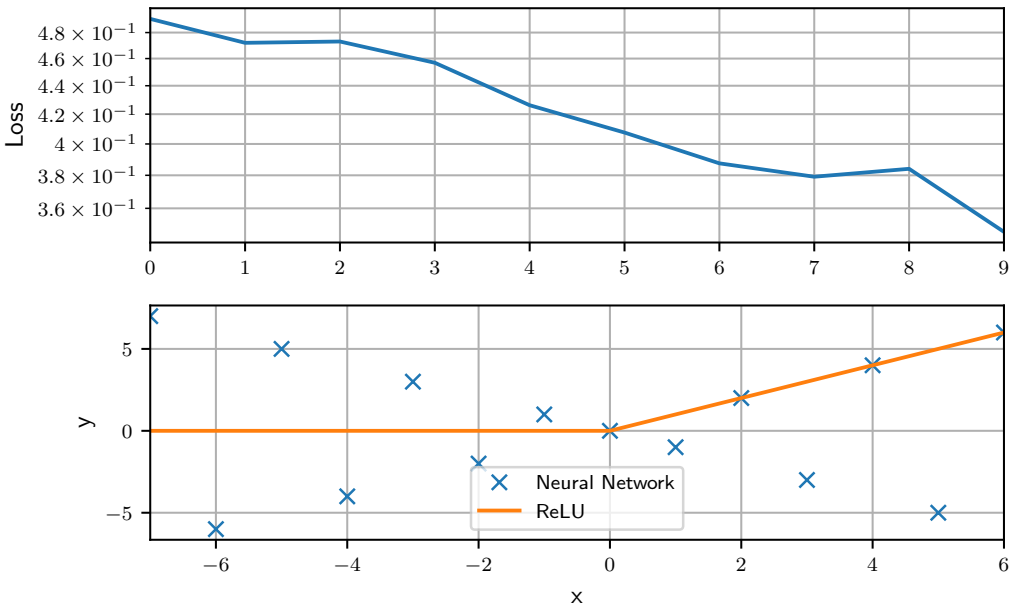
$$\rho_{00} = \cos(x_0)\sin(\theta_0) - \sin(x_0)\cos(\theta_0), \tag{7.21}$$

by chaining enough parameter gates and inputs together, we can apply sine functions of arbitrary frequency to the inputs. With enough gates, this ansatz is universal for quantum computing [187], but here we will restrict to a small circuit close in size to the other proposals investigated here.

Figure 7.11 shows the results for training the interleaved ansatz. Training a short interleaved ansatz reveals the sinusoidal feature of the operation, yet with the shorter circuits we are restricted to we cannot replicate either of the functions specified.

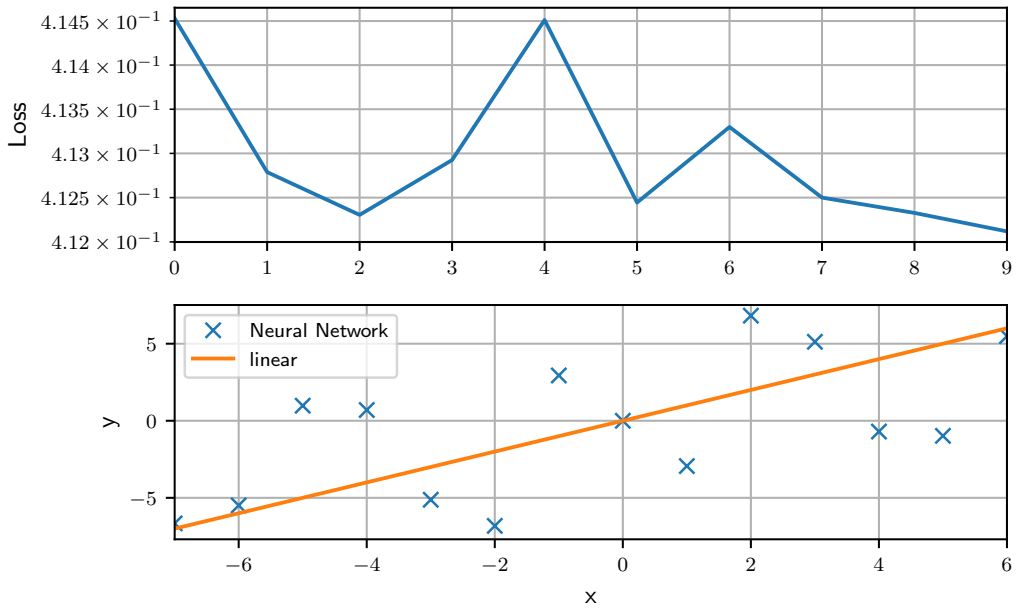


(a) Linear model

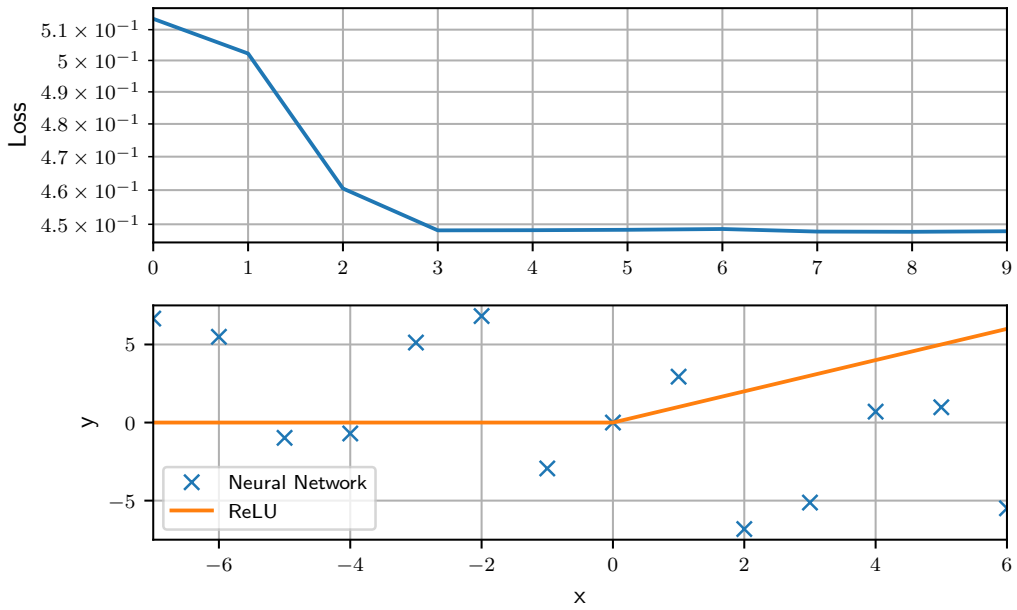


(b) ReLU model

Figure 7.9: The non-linear parameters ansatz being trained on a linear and non linear model. The model can capture the linear model, but fails with the non-linearity. We note that the loss function for the non-linear model has not fully converged, but we are limiting the available resources as we wish to investigate QNN primitives.



(a) Linear model

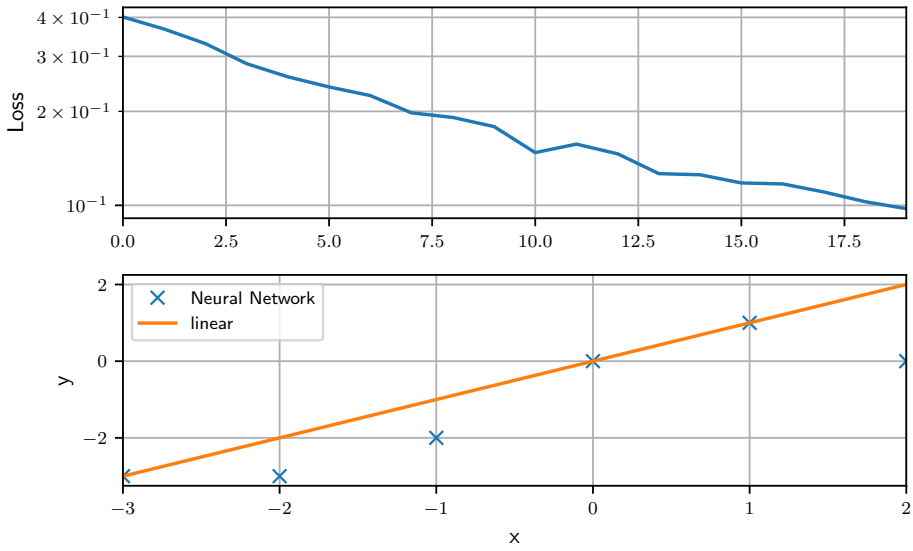


(b) ReLU model

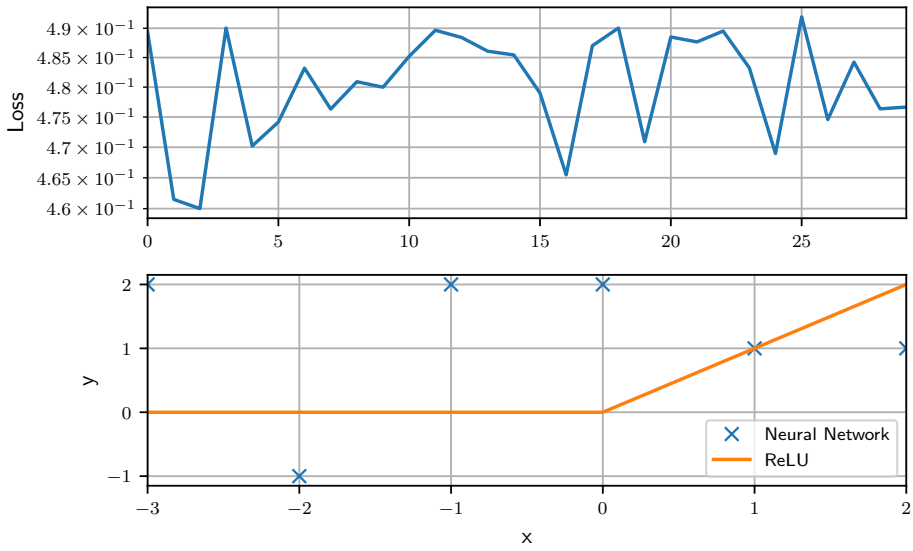
Figure 7.11: The interleaved circuits ansatz being trained on a linear and non linear model. With a short circuit, we are unable to easily reproduce either the non-linear or linear functions. The convergence of the loss function for the linear model suggests that we can do no better at this circuit length, but we may be able to train the linear model more.

7.5.6 Weighted States

The weighted states framework is takes a long time to train, due to the need to postselect on the measurement outcome. In Figure 7.12 we have approximated the linear function, but there are still issues with learning the ReLU function. This appears to show that the introduction of non-linearities on hte quantum state are not enough to learn an encoded non-linear function.



(a) Linear model



(b) ReLU model

Figure 7.12: The weighted states ansatz training to the linear and the ReLU function. Note that here we are using one fewer qubit than the other ansätze. This is due to the long training time required for this ansatz, which is a result of the post selection requirement of the ansatz. We see that a linear model can be trained, but the loss function of the ReLU model is not well behaved, suggesting that we can not train more for better results.

7.6 Discussion

In Figures 7.6- 7.12 we see that many of the QNN primitives struggle to learn a non-linear function with modest resources. This latter point is important as we propose to use these as building blocks for much larger algorithms, like how the signature of the perceptron can be seen in the individual nodes and weights of today's deep learning algorithms. All of the trainable primitives do manage to approximate the linear function though, with a decreasing loss function showing that there is some learning present. For this reason it may be salient to ask if ensuring that our QNN primitive implements a non-linear function, either in the state amplitudes or in the encoded data is the right direction to head in? The primitives reviewed here have all been presented in papers containing some numerical demonstration of their use in a QNN algorithm, apart from the only primitive to truly introduce a non-linearity to the quantum state, the weighted states [182], but this has demonstrated use in non-learning algorithms. It is for this reason that we suggest the holy grail of a QNN primitive that implements a non-linearity on a quantum state is overkill for the learning tasks we will require. We encourage readers to use the repository [177] to find their own QNN primitives.

Part III

Quantum Chemistry

Chapter 8

Classical Methods for Quantum Chemistry

Quantum chemistry is recognised as one of the potential applications of a quantum computer, a problem that classical computers struggle to solve, and one that quantum computers have an advantage. As we will see though, it may not be the case that this advantage will come in the NISQ era, due to the increasing resources required to run the Variational Quantum Eigensolver (VQE) at system sizes approaching quantum advantage. In this section we will review some classical computational techniques that are usually used as the starting point for quantum algorithms, followed by the workhorse of NISQ-era quantum computational chemistry, VQE, and we will conclude by reviewing one of the algorithms that will be used in the fault tolerant regime: quantum phase estimation, and a related algorithm, $\alpha-$VQE which attempts to interpolate between the regimes.

8.1 Classical Computational Quantum Chemistry

The goal of computational chemistry is to create a tractable model of some quantum system, e.g. a molecule or solid state system, which we can learn useful properties from. In this section we will discuss only molecules, as we will leave the sold state system covered in this thesis to Chapter 11. To get useful results, we usually focus our calculations on two aspects of the system: simulating time evolution via the

Shrödinger equation:

$$\frac{d}{dt}|\psi\rangle = -iH|\psi\rangle, \tag{8.1}$$

or finding the ground state, the lowest energy eigenvector of the Hamiltonian, H . Simulating time evolution is not covered in this thesis, but for a good review including time evolution, see Ref. [188]. Therefore in this section, we will begin with the general molecular Hamiltonian, and review the approximations and quantizations made until we get the second quantized Hamiltonian which can be approximated on a quantum device.

8.1.1 Molecular Hamiltonian

In later chapters we will present work on quantum algorithms for solving the ground states of molecular Hamiltonians, so we will review the formation of such Hamiltonians here, this review follows the treatment in [189]–[192].

To describe the Hamiltonian of a molecule we combine the kinetic terms for the nuclei and electrons, with the Coulomb repulsion between the electrons and nuclei, the nuclei and other nuclei and the electrons and other electrons. We do not take into account any relativistic effects here, and denote the number of electrons N and the number of nuclei K, the Hamiltonian is:

$$H = -\sum_i \frac{\hbar^2}{2m_e}\nabla_i^2 - \sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 - \sum_{i,I} \frac{e^2}{4\pi\varepsilon_0}\frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} \tag{8.2}$$

$$+ \frac{1}{2}\sum_{i\neq j} \frac{e^2}{4\pi\varepsilon_0}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \frac{1}{2}\sum_{I\neq J} \frac{e^2}{4\pi\varepsilon_0}\frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|}. \tag{8.3}$$

Here, $\hbar$ is Planck's constant, e, m_e is the electron charge and mass, and ε_0 is the vacuum permittivity. Capital letters relate to the nuclei, and $M_I, Z_I, \mathbf{R}_I$ is the mass, charge, and position of the Ith nuclei, and $\mathbf{r}_i$ is the position of the ith electron. ∇ is the Laplace operator, or the three dimensional partial derivative. The first two terms are the kinetic terms, and the final three are Coulomb terms. We can move to atomic units for ease, using the energy unit of a 1 Hartree $= e^2/4\pi\varepsilon_0 a_0, \hbar = 1$a.u., and $m_e = 1$a.u.. As our focus is quantum chemistry, e.g. the interactions of electrons in molecules, we can approximate the nuclei as point charges. This is the Born-

Oppenheimer approximation, which means we only need to solve the electronic Hamiltonian:

$$H_e = -\sum_i \frac{\nabla_i^2}{2} - \sum_{i,I} \frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} + \frac{1}{2}\sum_{i \neq j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}. \tag{8.4}$$

8.1.2 First Quantization

The first quantization we make is to approximate the true orbitals of electrons in the system with a set of basis wavefunction, $\{\phi_p(\mathbf{r}_i, \sigma_i)\}$, where $\mathbf{r}_i$ represents the spatial co-ordinates, and σ_i the spin of the ith electron. The spatial and spin indices are combined into a single index, $\mathbf{x}_i = (\mathbf{r}_i, \sigma_i)$. Two commonly used functions are Slater orbitals, which are exponential functions or Gaussian functions [193]. Gaussian functions are more tractable when calculating the integrals forming constants in the Hamiltonian, but Slater type orbitals better fit the form of the electron orbitals at greater distances. The trade-off usually made is to approximate the Slater orbitals with a linear combination of Gaussian orbitals.

One of the goals of making this discretisation is to enforce the antisymmetry of electrons, i.e. when an electron is swapped, the sign of the wavefunction is changed. Antisymmetry is also a product of the determinant of a matrix, so we can enforce this by writing the electron wavefunction as the determinant of a matrix formed from the M basis functions, known as a Slater determinant [194]:

$$\begin{vmatrix} \phi_0(\mathbf{x}_0) & \phi_1(\mathbf{x}_0) & \dots & \phi_{M-1}(\mathbf{x}_0) \\ \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \dots & \phi_{M-1}(\mathbf{x}_1) \\ . & . & \dots & . \\ \phi_0(\mathbf{x}_{N-1}) & \phi_1(\mathbf{x}_{N-1}) & \dots & \phi_{M-1}(\mathbf{x}_{N-1}) \end{vmatrix}. \tag{8.5}$$

The number of orbitals considered, M is larger than the number of electrons, but those electrons can only occupy N orbitals at a time. Algorithms for quantum devices using first quantised methods have been demonstrated [195]–[197], however in this thesis we focus upon the second quantized method.

8.1.3 Second Quantization

Similar to the first quantization, we write the Hamiltonian as M basis wavefunctions, and encode them in a vector indicating if an orbital is occupied or not:

$$\psi(\mathbf{x}_0 \dots \mathbf{x}_{N-1}) = |f_{M-1}, \dots, f_p, \dots, f_0\rangle, \tag{8.6}$$

where $f_p = 1$ if ϕ_p is occupied, and 0 otherwise. This is the occupation number vector, and all of these vectors form the Fock space. The ordering of the orbitals is significant, as we must respect the antisymmetry:

$$|\dots \phi_p, \dots, \phi_q, \dots\rangle = -|\dots, \phi_q, \dots, \phi_p, \dots\rangle. \tag{8.7}$$

In the second quantization method, we enforce the antisymmetry not by the determinant of a matrix, but through the actions of the operators.

Begin by defining the creation operator, $a_i^\dagger$, its action on an arbitrary Slater determinant $|\phi_p \dots, \phi_q\rangle$ is:

$$a_i^\dagger |\phi_i, \phi_p, \dots, \phi_q\rangle, \tag{8.8}$$

that is, it creates an electron in the spin-orbital ϕ_i. From the antisymmetric property of the wavefunction we have for an arbitrary occupation number vector, $|f\rangle$,

$$a_i^\dagger a_j^\dagger |f\rangle = |\phi_i, \phi_j, f\rangle, \tag{8.9}$$

$$a_j^\dagger a_i^\dagger |f\rangle = |\phi_j, \phi_i, f\rangle, \tag{8.10}$$

$$|\phi_i, \phi_j, f\rangle = -|\phi_j, \phi_i, f\rangle, \tag{8.11}$$

$$\left(a_i^\dagger a_j^\dagger + a_j^\dagger a_i^\dagger\right)|f\rangle = 0, \tag{8.12}$$

and we have derived the first commutation relation of fermionic operators:

$$\left\{a_i^\dagger, a_j^\dagger\right\} = 0. \tag{8.13}$$

From this, we see that changing the order of the operators changes the sign,

$$a_i^\dagger a_j^\dagger = -a_j^\dagger a_i^\dagger, \tag{8.14}$$

and if $i = j$, we have

$$a_i^\dagger a_i^\dagger = -a_i^\dagger a_i^\dagger = 0, \tag{8.15}$$

meaning we cannot create an electron in a n orbital that already exists, i.e. the Pauli exclusion principle:

$$a_i^\dagger a_i^\dagger |f\rangle = 0. \tag{8.16}$$

We define the annihilation operator, a_i as the adjoint of the creation operator, which annihilates an electron in spin-orbital i:

$$a_i |\phi_i, f\rangle = |f\rangle . \tag{8.17}$$

If the orbital i is not at the left hand side of the occupation number vector, $|f, \phi_i\rangle$ it must be switched with the orbital at the left hand side, which introduces a sign change:

$$a_i |\phi_j \phi_k \phi_i\rangle = -a_i |\phi_i \phi_k \phi_j\rangle = -|\phi_k \phi_j\rangle = |\phi_j \phi_k\rangle . \tag{8.18}$$

The annihilation operators also obey the same commutation relation:

$$\left\{a_i, a_j\right\} = 0 \tag{8.19}$$

$$a_i a_i |f\rangle = 0, \tag{8.20}$$

i.e. an electron cannot be annihilated twice.

The final commutation relation concerns both the annihilation and creation operators, firstly for $i \neq j$ which we know is zero if ϕ_j does not appear in $|f\rangle$ and ϕ_i does:

$$\left(a_i a_j^\dagger + a_j^\dagger a_i\right) |\phi_i f\rangle = -a_i |\phi_i \phi_j f\rangle - a_j^\dagger |f\rangle = |\phi_j f\rangle - |\phi_j f\rangle = 0. \tag{8.21}$$

Therefore,

$$\left\{a_i, a_j^\dagger\right\} = 0, i \neq j. \tag{8.22}$$

In the case where $i = j$, we create and destroy an electron in an unoccupied orbital, or destroy and create an electron in an occupied orbital (both $+0$), giving

$$\left\{a_i, a_i^\dagger\right\} = 1, \tag{8.23}$$

so our final commutation relation is:

$$\left\{a_i, a_j^\dagger\right\} = \delta_{ij}. \tag{8.24}$$

In the first quantized representation, the operators conserve the number of electrons, a property we must retain; therefore, creation and annihilation operators are used in pairs. We require a term to represent single electron operators, which can only involve one pair of spin-orbital operators: $a_p^\dagger a_q$, with the order required so that acting with this on the vacuum state produces zero. We also take into account two electron effects, such as electronic repulsion or the spin-orbit coupling. These are nonzero only if the Slater determinants contain at least two electrons, and differ in the occupation of two pairs of orbitals: $a_p^\dagger a_r^\dagger a_s a_q$, which are again ordered to give zero when acting on the vacuum or a single electron state. We can now build the second quantized representation of the electronic structure Hamiltonian:

$$H = \sum_{p,q} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s, \tag{8.25}$$

where the amplitudes h_{pq}, h_{pqrs} represent the probability of the single or double excitation between the spin-orbitals $pq, pqrs$, and are calculated by taking the average (integrating) over the spin-orbitals involved:

$$h_{pq} = \int \phi_p^* (\mathbf{x}) \left(-\frac{\nabla^2}{2} - \sum_I \frac{Z_I}{|\mathbf{r} - \mathbf{R}_I|} \right) \phi_q (\mathbf{x}) \, d\mathbf{x} \tag{8.26}$$

$$h_{pqrs} = \int \int \frac{\phi_p^* (\mathbf{x}_1) \, \phi_q^* (\mathbf{x}_2) \, \phi_r (\mathbf{x}_2) \, \phi_s (\mathbf{x}_1)}{|\mathbf{r}_1 - \mathbf{r}_2|} \, d\mathbf{x}_1 d\mathbf{x}_2. \tag{8.27}$$

The first, single electron term accounts for the kinetic energy of the electron, and their interaction with the nuclei, whilst the second accounts for the electron-electron Coulomb repulsion.

To fully solve this model we need to find all the eigenstates of the Hamiltonian. As eigenstates and eigenvalues describe all of the possible values of a physical operator, e.g. all of the observable properties of the system. To find all of the eigenstates we can form a complete basis from the Slater determinants, as seen in Equation 8.5, and the eigenstates can be expressed as a linear combination of Slater determinants

$$|\psi\rangle = \sum_f \alpha_f |f\rangle . \tag{8.28}$$

These are exact solutions, known as the Full Configuration Interaction (FCI), but only if all $\binom{M}{N}$ determinants are included, where N is the number of electrons, and M is the number of possible orbitals they can occupy, which can grow to be a huge number, meaning this does not scale well with the number of electrons. Therefore we usually make approximations to solve these systems on classical and on quantum devices. We will next review the classical methods Hartree-Fock and Coupled cluster, as these are relevant to the quantum algorithms we use.

8.2 Classical Approximations

8.2.1 Hartree-Fock

The Hartree-Fock approximation is one of the simplest approximations we make, assuming that the wavefunction as described above (taking into account the approximations already made) can be approximated by a single Slater determinant. In the Hartree-Fock approximation, we are not considering any correlation between electrons. We remove the Coulomb repulsion form the Hamiltonian, instead approximating it as the average charge distribution of all electrons. The Hartree-Fock method is an iterative procedure, by calculating the positions of each electron we can update the potential, which allows us to update the positions of the electrons; we iterate this procedure until the occupied orbitals converge.

The Hamiltonian we minimise in this approximation is the Fock operator:

$$\hat{f} = \sum_{pq} f_{pq} a_p^\dagger a_q, \tag{8.29}$$

where the scalar f is a combination of the single electron part, h_{pq}, from Equation 8.25 and a modified form of the two electron part:

$$f_{pq} = h_{pq} + V_{pq}, \tag{8.30}$$

$$V_{pq} = \sum_{I \in \text{occupied}} \left(h_{pqII} - h_{pIIq} \right) \tag{8.31}$$

where the index I is over all occupied spin-orbitals, p, q are over all unoccupied and occupied orbitals. In V_{pq} the first term represents the Coulomb interaction of an electron with the charge distributions of occupied orbitals, the second term is the exchange function and arises due to the antisymmetry of the wavefunction. The terms f_{pq} form the Fock matrix, which is diagonalised to give eigenvectors which are canonical spin orbitals, and eigenvalues which are orbital energies; but as the Fock matrix is defined in terms of its own eigenvectors, it must be reconstructed and re-diagonalised to solve. This is how we carry out the procedure discussed in the previous paragraph. As we are iterating to make the electric field consistent with itself, this procedure is knows as the self-consistent field (SCF) method [191].

Dynamic correlation can be added to the Hartree-Fock model to correct for the Coulomb repulsion between electrons, and the Hartree-Fock model is still a good approximation. However, there are systems where more than one Slater determinant is dominant in the wavefunction, and the Hartree-Fock method fails to be a good approximation. This is static correlation and needs to be modelled for systems such as transition metals and excited states, but we will usually begin with the Hartree-Fock state as a starting point.

8.2.2 Coupled Cluster

The coupled cluster method introduces corrections to the Hartree-Fock state via exponentiating excitation operators T_i:

$$|\psi_{\mathrm{CC}}\rangle = e^{\sum_i T_i} |\psi_{\mathrm{HF}}\rangle\,, \tag{8.32}$$

with

$$T_1 = \sum_{i\in\text{virtual},\alpha\in\text{occupied}} t_{i\alpha} a_i^\dagger a_\alpha \tag{8.33}$$

$$T_2 = \sum_{i,j\in\text{virt},\alpha,\beta\in\text{occ}} t_{ij\alpha\beta} a_i^\dagger a_j^\dagger a_\alpha a_\beta\,, \tag{8.34}$$

an so on. The T_i operators can include all excitation operators, but this does not scale well with system size, so we usually restrict to optimising over T_1, T_2, which is known as Coupled Cluster Singles and Doubles (CCSD). However, one drawback of the coupled cluster method is that it cannot be minimised via the variational principle, as the derivatives of the state become intractable functions.

8.2.3 Unitary Coupled Cluster

We will now introduce the Unitary Coupled Cluster (UCC) method, which has the advantage of minimisation via variation, and is better for treating static correlations. Unfortunately implementing UCC on a classical computer is very costly, but it can be implemented on a quantum device. The Unitary Coupled Cluster method is a simple extension of the original coupled cluster method, which uses the adjoint of the T_i operators to form a unitary operator:

$$|\psi_{UCC}\rangle = e^{\sum_i T_i - T_i^\dagger}\,. \tag{8.35}$$

We still usually truncate the ansatz at a certain excitation - the Unitary Coupled Cluster Singles and Doubles (UCCSD) is a common ansatz in quantum computing problems, which utilises T_1 and T_2. In quantum computing UCC is used as an ansatz for VQE, and we will review ansatze in Section 10.1.

Chapter 9

Quantum Methods for Quantum Chemistry

9.1 Mapping Fermionic Operators to Qubits

The operators $a_i^\dagger a_j$ introduced in Section 8.1.3 are not operators which we can naturally implement or measure on a quantum device. We will here discuss some of the proposed mappings between fermionic operators and qubit operators.

9.1.1 Jordan-Wigner Transformation

The Jordan-Wigner transformation [198] transforms between fermionic orbitals and Pauli matrices. It was proposed as a method for simulating fermions on quantum computers in 2002 [199]. Taking the canonical form of the fermionic excitation operators in the Z basis, and single site j:

$$a_j^\dagger = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, a_j = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}. \tag{9.1}$$

we can verify that a linear combination of σ_x and σ_y can reproduce these operators:

$$\frac{1}{2}\left(\sigma_j^x + i\sigma_j^y\right) = a_0^\dagger \tag{9.2}$$

$$\frac{1}{2}\left(\sigma_j^x - i\sigma_j^y\right) = a_0. \tag{9.3}$$

However, this does not apply when we consider more than one site, k with $j \neq k$, as the operators do not obey the commutation relations of Equation 8.22. To fix this, we introduce a string of $\sigma^z_{0\ldots j}$ operators which take into account the occupied orbitals from the beginning of the chain and site j.

$$\hat{a}_j \rightarrow \left(\prod_{l=1}^{j-1} -\sigma_l^z\right)\left(\sigma_j^x - i\sigma_j^y\right), \tag{9.4}$$

$$\hat{a}_j^\dagger \rightarrow \left(\prod_{l=1}^{j-1} -\sigma_l^z\right)\left(\sigma_j^x + i\sigma_j^y\right). \tag{9.5}$$

If we wish to enact a two - orbital operator, e.g. $a_j^\dagger a_k$, due to the Pauli relation $(\sigma_n^z)^2 = I$, we only need to use σ^z terms on the orbitals between j and k, not from $0\ldots j-1$.

This string of σ_z operators can be a drawback of Jordan-Wigner, consider a four qubit system, and let us write out the unitary cluster operator between orbitals 1 and 4, $e^{T_1^{1,4} - T_1^{1,4\dagger}}$:

$$T_1^{14} = t_4^1 \hat{a}_1^\dagger \hat{a}_4 \tag{9.6}$$

$$\rightarrow t_4^1 \sigma_+^1 \sigma_z^1 \sigma_z^2 \sigma_z^3 \sigma_-^4 \tag{9.7}$$

$$T_1^{14} - T_1^{14\dagger} = -2it_4^1 \left(\sigma_x^1 \sigma_z^2 \sigma_z^3 \sigma_y^4\right), \tag{9.8}$$

In the Jordan-Wigner transformation an operation that involves only the 1st and 4th orbitals must involve all of the qubits. This is shown in circuit form in Figure 9.1, to demonstrate that we now require two-qubit gates between all pairs of qubits.

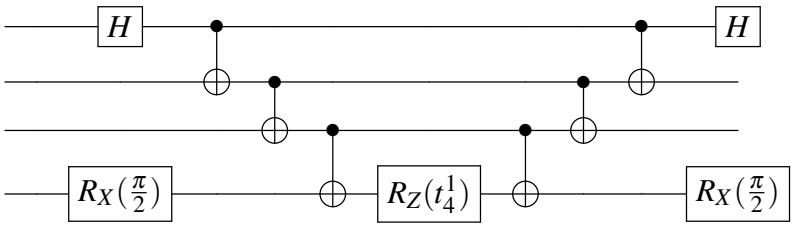


Figure 9.1: The Jordan-Wigner encoded circuit of the operator in Equation 9.8. Due to the Jordan-Wigner transformation, we now require two-qubit gates between all pairs of qubits.

It is for this reason that alternative fermionic encodings are proposed, which we will review here.

9.1.2 Parity Encoding

One solution to the problem of updating the parity of all the qubits between j and k is to store the parity information on qubit j. This encoding is the parity basis encoding [200], and encodes $p_j = \sum_{s=0}^{j} f_s$ on qubit j where f_s is the parity of qubit s. However, if we wish to operate with a creation or annihilation operator on orbital j, the correct operator depends on the state of qubit $j-1$, plus we must update the parity off all the qubits following j, e.g act with a string of $\sigma^x_{j+1\ldots N}$. We now see that using the parity basis offers no advantages over the Jordan-Wigner basis.

However, there is an encoding which interpolates between the Jordan-Wigner and the parity encoding, offering advantages of each: the Bravyi-Kitaev encoding.

9.1.3 Bravyi-Kitaev Encoding

In the Jordan-Wigner encoding, we store information about the occupation of orbitals locally, and the parity information non-locally, whereas in the parity basis we store information about the parity locally, and occupation non-locally. In the Bravyi-Kitaev encoding [201] we balance the locality of the two sets of information. Each qubit stores information on the parity and occupation of orbitals, when j is even, qubit j stores the occupation of orbital j, when j is odd it also stores the occupation of a set of orbitals $\leq j$. The set of orbitals that a qubit stores is determined by a binary tree structure, which a review can be found in [200], [202]. A drawback of the Bravyi-Kitaev encoding in the NISQ era is that noise in the quantum circuit simulating the chemical system can lead to states which are unphysical. Therefore in the rest of this thesis we will work in the Jordan-Wigner transformation when applicable.

9.1.4 The Compact Fermion to Qubit Mapping

An encoding [203], [204] which maps fermions to qubits is unique due to its recency, can be applied directly to NISQ devices. Instead of a single qubit for each fermionic mode (or Slater orbital) , ancilla qubits are used to enforce the parity conditions,

and only certain operators are allowed on the edges of the interaction graph. This encoding is best suited to solid state systems, as the fermions lie on a grid, but other lattice geometries are possible [204]. Again this encoding will not be used in this work so the reader is referred to [203] for more information.

9.2 Simulating Time Evolution

One of the reasons to do quantum chemistry on a quantum device is so that we can simulate the time dynamics of a system. Given a Hamiltonian H we can simulate the effect of $e^{-iHt}|\psi\rangle$ if we have a quantum computer prepared in the state $|\psi\rangle$ and an algorithm for enacting H as quantum gates. We can use the Jordan-Wigner encoding to translate excitation operators in H into quantum gates, but we still need a method of turning a sum of terms in the Hamiltonian into a set of gates we can enact, which we use the Suzuki-Trotter formula for.

9.2.1 Suzuki-Trotter Formula

The Suzuki-Trotter formula [205]–[209] allows us to simulate the time evolution of a Hamiltonian with small error. The formula approximates the exponential of the sum of some matrices with their product:

$$e^{A+B} = \lim_{n\to\infty}\left(e^{A/n}e^{B/n}\right)^n, \tag{9.9}$$

We can express the Hamiltonian in this way allowing us to simulate it as a number of quantum gates applied to the starting state:

$$e^{-itH} = \Pi_{j=0}^{m} e^{-itH_j} \tag{9.10}$$

if the time step t is too large we can also break this up in the Suzuki-Trotter formula:

$$e^{-itH} = \left(\prod_{j=0}^{m} e^{-it/rH_j}\right)^r. \tag{9.11}$$

9.3 Phase Estimation

Here we will briefly discuss the quantum simulation algorithm that is used in fully fault-tolerant devices, with long coherence times and error correction, discussed in Chapter 4. For a more in-depth treatment, see [43, chap. 5]. In the general phase estimation algorithm the goal is to estimate the phase of the eigenvalue, λ, of some unitary, U. As the eigenvalue is between 0 and 1, it can be written as $e^{i2\pi\phi}$, where ϕ is the phase. To measure this phase, we can imprint it onto an ancilla qubit, so we require an operation that affects an ancilla state without becoming affected (so that we can increase precision by using more ancilla). To ensure that the state is not affected, we must first be in an eigenstate. We can see such an effect in the CNOT gate, with the $|+\rangle$ state as the control, transforming into in the $|0\rangle, |1\rangle$ basis, with $|-\rangle$ as the eigenstate of X, in Figure 9.2.

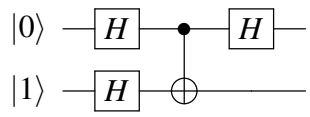


Figure 9.2: CNOT with the $|+\rangle$ state as control imprints a phase onto the control qubit.

The outcome of this circuit is

$$\begin{aligned} \mathrm{H}_0\mathrm{CNOT}\left[|+\rangle, |+\rangle\right] &\rightarrow \mathrm{H}_0\left[\frac{1}{\sqrt{4}}\left(|00\rangle + |11\rangle - |01\rangle - |10\rangle\right)\right] \\ &= \mathrm{H}_0\left[\frac{1}{\sqrt{2}}\left(|0\rangle|-\rangle - |1\rangle|-\rangle\right)\right] \\ &\rightarrow |1\rangle|-\rangle \end{aligned} \tag{9.12}$$

where we see that measuring the first qubit gives us the $|1\rangle$ measurement, which is the -1 eigenvalue of Z, giving us the correct value of -1 as the eigenvalue of the eigenstate $|-\rangle$ of X.

As the eigenstate was exactly represented by a qubit, we did not need to estimate the phase, by adding additional qubits. In the true phase estimation algorithm we may require more qubits to measure the phase, and if it requires more qubits to be written, the algorithm produces an estimation. To measure more bits of precision we will use a binary representation of the phase, given by $0.\phi_0\phi_1,\ldots\phi_n$, where each

ϕ_i is the measurement result of a qubit. We apply the target operation, U^{2^i} for each bit of precision, i, and use the inverse Quantum Fourier Transform (QFT) [54] to transform the data held in the phases into the measurement basis (the same act as the final Hadamard gate in Figure 9.2). Briefly, the inverse QFT, $\text{QFT}^\dagger$ transforms

$$\text{QFT}^\dagger \left[\frac{1}{2^{n/2}} \left(|0\rangle + e^{i2\pi 0.\phi_0} |1\rangle \otimes |0\rangle + e^{i2\pi 0.\phi_0\phi_1} |1\rangle \dots \right) \right] \rightarrow |\phi_0\rangle |\phi_1\rangle , \tag{9.13}$$

which allows us to write the circuit for the phase estimation algorithm in Figure 9.3.

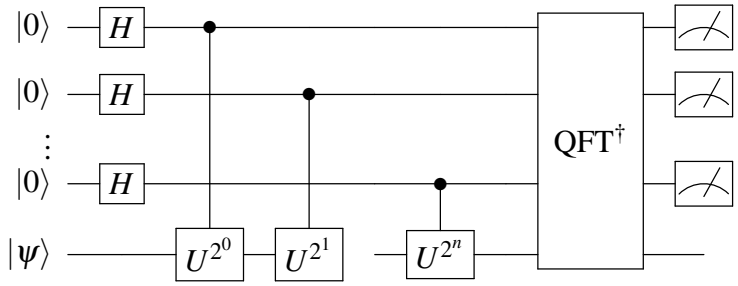


Figure 9.3: The general circuit for the phase estimation algorithm. Notice that we still require an eigenstate, $|\psi\rangle$ of U in the final register.

The action of this circuit is,

$$\begin{aligned} |0\rangle^{\otimes n} &\rightarrow |+\rangle^{\otimes n} \\ &\rightarrow \frac{1}{2^{n/2}} \left(|0\rangle + e^{i2\pi 0.\phi_0} |1\rangle \otimes |0\rangle + e^{i2\pi 0.\phi_0\phi_1} |1\rangle \cdots \otimes |0\rangle + e^{i2\pi 0.\phi_0\phi_1\dots\phi_n} |1\rangle \right) \\ &\rightarrow |\phi_0\phi_1\dots\phi_n\rangle . \end{aligned} \tag{9.14}$$

which allows us to measure the phase in the output register.

We note that so far we have assumed that the eigenstate $|\psi\rangle$ can be prepared in the second register. If we do not know how to prepare the eigenstate, we can still find the phase with high probability. As the eigenstates of U, $|\psi\rangle$ form a basis, we can expand whatever state prepared in that basis, $|\rho\rangle = \sum_\psi c_\psi |\psi\rangle$. The phase estimation algorithm will therefore prepare $\sum_\psi c_\psi |\tilde{\phi}_\psi\rangle_{\text{ancilla}} |\psi\rangle_{\text{data}}$, and we will measure the phase corresponding to $|\psi\rangle$ with probability $\left|c_\psi\right|^2$. This is the basis for the quantum principle component analysis (QPCA) discussed in Chapter 5. There has been some experimental progress towards phase estimation on near-term devices [210], but the

levels of noise, and low coherence times mean that it isstill slightly out of reach.

9.3.1 α-VQE

This algorithm [211] is discussed here as it interpolates between the VQE algorithm, to be discussed in Chapter 10 and phase estimation, it can be tuned to the coherence time of a noisy device, to get the most coherence possible. This reduces the stringent requirements of measurement samples $\mathcal{O}\left(\varepsilon^{-2}\right)$ in vanilla VQE. In α-VQE we prepare the eigenstate, $|\psi\rangle$ using VQE. The phase estimation algorithm presented above was first improved in [212] taking into account the ability of some quantum devices to measure and reset qubits [14] so that all the ancilla qubits are replaced by a single qubit which is reset. It also introduces a gate $R_z(-M\theta)$ to the first qubit, where M is the number of repetitions of U and θ is used to compare the phase to a known value. Next, Rejection Filtering Phase Estimation (RFPE) is introduced [213], [214] which essentially uses a Bayesian update model to change the values of M, θ to learn the phase. Finally, the innovation in [211] is to constrain M, θ to a parameter α which accounts for the length of coherence time available in the device.As the number of applications, M determines how long the circuit must run for, the Bayesian model in [213] can quickly require longer circuits than are possible. If we set the required precision ε, and set the maximum coherence time, $D_{\max} = \frac{1}{\varepsilon^{\alpha}}$, then α-VQE requires a minimum number of shots, $N_{\min}$ that scales as

$$N_{\min} = \begin{cases} \frac{2}{1-\left(\log(D_{\max})/\log(\frac{1}{\varepsilon})\right)}\left(\frac{1}{\varepsilon^2 D_{\max}^2} - 1\right) & \text{if } D_{\max} < \frac{1}{\varepsilon} \\ 4\log(\frac{1}{\varepsilon}) & \text{if } D_{\max} \geq \frac{1}{\varepsilon}. \end{cases} \tag{9.15}$$

This shows that even if $D_{\max} < \frac{1}{\varepsilon}$ we are able to reduce the number of iterations required. α-VQE chooses $(M, \theta) = \left(\frac{1}{\sigma^{\alpha}}, \mu - \sigma\right)$ where μ, σ are the mean and standard deviation of the posterior distribution calculated from the previous Bayesian update.

Chapter 10

The Variational Quantum Eigensolver

In this chapter we will present a review of the Variational Quantum Eigensolver, touching on many aspects of the algorithm, including encoding, optimisation, trainability, and the barren plateaus phenomena. This is an introduction which can be used in considering the work in Chapters 11 and 12. We will cover the specific topic of noise in VQE when we discuss noise in Chapter 13.

The Variational Quantum Eigensolver (VQE) [50], [169] is a method for finding the ground state of a Hamiltonian on NISQ quantum devices, whereas phase estimation is used when we have access to long coherence times and noiseless gates, covered in Section 9.3. The VQE makes use of the variational principle [190], i.e. that the energy calculated by the Hamiltonian cannot be lower than the ground state energy:

$$\langle \psi(\theta) | H | \psi(\theta) \rangle \geq \langle GS | H | GS \rangle \forall \psi(\theta). \tag{10.1}$$

We therefore prepare a parameterised state, with parameters θ, and vary θ until the energy $\langle \psi(\theta) | H | \psi(\theta) \rangle$ converges. It is hoped that the energy the algorithm converges to is the ground state energy, which will happen if the ansatz is expressive enough, and if the algorithm does not encounter a barren plateau.

The ansatz that a VQE algorithm uses is an approximation to the ground state of a Hamiltonian, and can be compared to similar classical approximations, e.g. the Unitary Coupled Cluster ansatz is closely related to the Coupled Cluster calculation in quantum chemistry using classical computers. A quantum device can represent n electron orbitals in n qubits, whereas the same number of orbitals would take 2^n bits

to represent classically. So it is from the ability to represent larger systems that it is hoped, but yet to be shown, that VQE will demonstrate advantage over classical algorithms.

10.1 Ansatz Design

To prepare the state $|\psi(\theta)\rangle$ we use a parameterised quantum circuit, called an ansatz. Designing an ansatz is one of the more important problems in a VQE experiment, where the ansatz must be expressive enough to contain the ground state, yet with too many parameters it will take a long time to train, and there is a connection between the expressibility of the ansatz and the incidence of barren plateaus [215], which we will discuss in Section 10.4.

10.1.1 Chemically Inspired Ansatz

In the chemically inspired ansatz we base the circuit upon some set of operators found in the problem. For example, as discussed in Section 8.2.3 we can approximate the ground state of the Hamiltonian with a set of operators such as the Unitary Coupled Cluster Singles and Doubles (UCCSD). A chemically inspired ansatz can be built from the encoded (e.g. Jordan-Wigner) form of these operators and parameterised by their multiplicative constant, which is introduced into the quantum circuit as a rotation gate, seen in Figure 9.1. The ansatz circuit can therefore be described by:

$$|\psi(\theta)\rangle = \Pi_{j=N}^{0} U_j(\theta_j), \tag{10.2}$$

where U_j are the UCCSD operators. Order of the operators is important, and choosing an ordering with double excitations before single excitations was found to be beneficial in [216].

10.1.2 Hardware Efficient Ansatz

By inspecting the form of a single encoded operator in Figure 9.1 we see that it requires many entangling gates. In the NISQ era it is common that these are the noisiest gates in the device [22]. The Hardware Efficient Ansatz (HEA) [217] is an attempt to reduce the overall number of gates in a circuit, and the incidence of the

noisiest types of gates. Instead of a parameterised block, $U(\theta)$, being some operator built from multiple gates, the HEA is built from gates which are native to the device. The ansatz was found to reduce the number of parameters required for a simple circuit, and reduce Trotterisation errors. The disadvantage of using the HEA is that we are no longer guaranteed to stay in some physical state when optimising, and we do not know if the ansatz chosen will encompass the ground state.

10.2 Measurement Grouping in VQE

A factor which greatly influences the run time of VQE experiments has not yet been discussed: shot noise. For simplicity, if we wish to measure a qubit prepared in $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$in the Z basis we can only obtain a single measurement at a time, a single measurement will give us $0(1)$ which does not give us the right impression of the answer. If we were to repeat this measurement 5 times we could get a sequence such as 00101, which is more accurate, but the proportions are not correct. This is why we must repeat the measurement many times, e.g. if we were to repeat this measurement 100 times we would more likely get the correct distribution. However, if we replace the state above with $\frac{1}{10}|0\rangle + \frac{\sqrt{99}}{10}|1\rangle$ it is clear that a strategy which repeats 100 times is very unlikely to capture the true distribution.

10.2.1 Variance of Measurement Operators

Following [169], we can formalise this description. We wish to measure the expectation value of $\langle\psi(\theta)|H|\psi(\theta)\rangle$ which we will relabel as $\langle H\rangle$ for convenience. The precision, ε to which we measure $\langle H\rangle$ is defined by measurements of $\langle H\rangle$ are normally distributed with a standard deviation of $\frac{\varepsilon}{2}$. When repeating single shots of a quantum device we are measuring the estimator, $\langle\hat{H}\rangle$, and to achieve a precision of ε we require a variance of ε^2 in $\langle\hat{H}\rangle$. Due to the linearity of the Hamiltonian, this cn be rewritten in terms of the Hamiltonian components $\alpha_j H_j$:

$$\langle\hat{H}\rangle = \sum_j \langle\alpha_j \hat{H}_j\rangle. \tag{10.3}$$

In the naïeve strategy, where we do no operator grouping, the covariance between individual measurements is 0,

$$\mathrm{Cov}\left[\langle\hat{H_A}\rangle, \langle\hat{H_B}\rangle\right] = 0 \forall A \neq B, \tag{10.4}$$

and the total variance is simply the sum of individual variances:

$$\mathrm{Var}\left[\langle\hat{H}\rangle\right] = \sum_j \mathrm{Var}\left[\alpha_j H_j\right]. \tag{10.5}$$

We can therefore estimate the total number of shots required in this experiment to be

$$n_{\mathbf{E}} = M \sum_j \frac{\mathrm{Var}\left[\langle\hat{\alpha_j H_j}\rangle\right]}{\varepsilon^2}, \tag{10.6}$$

where M is the number of sites that can be occupied.

However, we do not have *a priori* access to $\mathrm{Var}\left[\langle\hat{\alpha_j H_j}\rangle\right]$, so we must estimate it as the experiment progresses. To estimate the variance as we are conducting the experiment, we can use a Bayesian method. We begin by defining the likelihood of measuring some sequence of measurements X, with only two possible outcomes (e.g. $\{|0\rangle, |1\rangle\}$):

$$P(X|p) = \binom{N}{r} p^r (1-p)^{N-r}, \tag{10.7}$$

where the total number of measurements is N, and r is the number of specific measurements, say $|1\rangle$. X completely depends upon p, which is the probability of measuring $|1\rangle$. We will use the Beta distribution [218] which is defined by two variables, α and β, and we will update these variables in our Bayesian update.

$$P(p;\alpha,\beta) = \mathrm{Beta}(\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}, \tag{10.8}$$

where $\Gamma(z)$ is the gamma function:

$$\Gamma(z) = \int_0^\infty x^{z-1} e^{-x}\, dx. \tag{10.9}$$

The likelihood of sequence X of length N, containing r instances of $|1\rangle$ is therefore determined by the Beta function:

$$P(p|X) = \text{Beta}(\alpha + r, \beta + N - r) = \text{Beta}\left(\alpha', \beta'\right). \tag{10.10}$$

We can use these updated α', β' to get the expectation and variance of $p(|1\rangle)$:

$$\langle p \rangle = \frac{\alpha'}{\alpha' + \beta'}, \tag{10.11}$$

$$\text{Var}(p) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \tag{10.12}$$

One of the largest improvements that can be made in compute time of a VQE experiment is a result of grouping measurements better. In the naïeve case if the Hamiltonian is a sum of m terms, we must make m repeats of the circuit. To be explicit, if the Hamiltonian is $H = H_A + H_B$, to evaluate $\langle H \rangle$ we cannot in general do this in a single step, as we must measure $\langle H_A \rangle$ followed by $\langle H_B \rangle$. This is the strategy that was followed in the paper which first proposed and tested a VQE experiment [50]. This problem becomes much worse as the system size increases, for example, even a modest system such as H_2O requires 1086 terms. The number of terms for an n qubit system generally scales as n^4 [192], [219].

Fortunately, the number of distinct measurements required can be greatly reduced by measurement grouping strategies. The simplest of these is to measure operators that occur on separate qubits simultaneously, for example if H_A only had support on qubits 0 and 1, and H_B on qubits 2 and 3.

10.2.2 Grouping of Commuting Terms

In the next level of sophistication we can group the terms into commuting sets. If $[H_A, H_B] = 0$, the measurement of H_A does not affect the measurement of H_B, so these can be measured in sequence[1]. It would seem therefore that grouping measurements into a few commuting sets as possible will yield the quickest runtime, but this is not necessarily the case., in [169] they demonstrate a state and collection of commuting Paulis where a set of four operators requires fewer shots for a given precision than a

set of three.

When we group commuting operators to be measured in the same state preparation, Equation 10.6 no longer holds. The variance is defined as:

$$\mathrm{Var}[H] = \langle H^2 \rangle + \langle H \rangle^2, \tag{10.13}$$

when we measure terms in the same preparation we no longer have that $\sum_{i \neq j} \langle \alpha_i \alpha_j H_i H_j \rangle - \langle \alpha_i H_i \rangle \langle \alpha_j H_j \rangle = 0$. This is because we are no longer measuring the estimators sampled in separate state preparations as in Equation 10.4. Covariance between the operators is defined as:

$$\mathrm{Cov}[\langle H_A \rangle, \langle H_B \rangle] = \langle H_A H_B \rangle - \langle H_A \rangle \langle H_B \rangle. \tag{10.14}$$

This property can increase the variance of a group of commuting operators so that the most efficient grouping is not necessarily the smallest.

10.2.3 Distributing Measurement Shots

However, this is not the whole picture, as in what has come previously we have assumed that the total number of shots is distributed evenly over all of the operator groups. In [219] the theory of grouping measurement operators is improved to allow for a variable number of shots used on each operator grouping. If we redefine our Hamiltonian as a sum of groups, indexed by i of commuting operators, indexed by j,

$$H = \sum_{i=1}^{N} H_i = \sum_{i=1}^{N} \sum_{j=1}^{m_i} \alpha_{ij} P_{ij}, \tag{10.15}$$

where P_{ij} is Pauli operator j in group i, we can discuss the ratio between the minimum number of measurements needed for a given grouping of operators, M_g

[1] It may be possible in theory to measure qubits in the same state preparation, but in NISQ devices measurement usually disturbs the device to such a degree that subsequent measurements would be too noisy or not possible at all [14], [49], [220]–[222]. As the commuting operators can be diagonalised by a single matrix, we can construct a rotation to allow for simultaneous measurement, as discussed in [219].

and the ungrouped set, M_u:

$$R = \frac{M_u}{M_g} = \left(\frac{\sum_{i=1}^{N} \sum_{j=1}^{m_i} |\alpha_{ij}| \sqrt{\mathrm{Var}\left[P_{ij}\right]}}{\sum_{i=1}^{N} \sqrt{\mathrm{Var}\left[H_i\right]}} \right)^2 . \tag{10.16}$$

By comparing different grouping strategies we can maximise the value of R to obtain the most measurement savings for a particular grouping, M_g. The only drawback of the ratio R is that it cannot be evaluated without knowledge of the quantum state we wish to measure. Therefore in [219] they replace the variance with the expected value over the spherical distribution.

In the uniform spherical distribution the variance and covariance of Paulis become easier to treat, that is the expected variance is independent of P_i :

$$\begin{aligned} \mathbf{E}\left[\mathrm{Var}\left[P_i\right]\right] &= 1 - \mathbf{E}\left[\langle P_i \rangle^2\right] && (10.17)\\ &= 1 - \int \langle \psi | P_i | \psi \rangle^2 \, d\psi && (10.18)\\ &= 1 - \frac{1}{2^n + 1}, && (10.19) \end{aligned}$$

and the expected covariance is zero. Following [223], if we split the eigenspace of P_i, P_j into $(-1,-1),(-1,+1),(+1,-1),(+1,+1)$ we can write $|\psi\rangle$ as a general sum over these subspaces: $|\psi\rangle = a|\psi_{-1,-1}\rangle + b|\psi_{-1,1}\rangle + c|\psi_{1,-1}\rangle + d|\psi_{1,1}\rangle$.

$$\begin{aligned} \mathrm{Cov}\left[P_i, P_j\right]_{|\psi\rangle} &= \left(|a|^2 - |b|^2 - |c|^2 + |a|^2\right) && (10.20)\\ &\quad - \left(|a|^2 - |b|^2 + |c|^2 + |d|^2\right)\left(-|a|^2 + |b|^2 - |c|^2 + |d|^2\right), && (10.21) \end{aligned}$$

if we consider the matching state $|\psi'\rangle = b|\psi_{-1,-1}\rangle + a|\psi_{-1,1}\rangle + d|\psi_{1,-1}\rangle + c|\psi_{1,1}\rangle$, we can substitute to see that $\mathrm{Cov}\left[P_i, P_j\right]_{|\psi\rangle} = -\mathrm{Cov}\left[P_i, P_j\right]_{|\psi'\rangle}$. So each $|\psi\rangle$is cancelled by $|\psi'\rangle$, which means that over the uniform spherical distribution,

$$\mathbf{E}\left[\mathrm{Cov}\left[P_i, P_j\right]\right] = 0. \tag{10.22}$$

Using $\mathbf{E}\left[\mathrm{Var}\left[P_i\right]\right] = \text{const.}$ and $\mathbf{E}\left[\mathrm{Cov}\left[P_i, P_j\right]\right] = 0$, we can replace the terms in

Equation 10.16 with the constants in H:

$$\hat{R} = \left(\frac{\sum_{i=1}^{N} \sum_{j=1}^{N_i} |\alpha_{ij}|}{\sum_{i=1}^{N} \sqrt{\sum_{j=1}^{N_i} \alpha_{ij}^2}} \right). \tag{10.23}$$

This is a ratio we can calculate without needing access to the measurement outcomes as in [169], so we can formulate our measurement strategy before we run the experiment. If we were to brute force this calculation, we would have to consider all combinations of commuting operators different values of $\hat{R}$ which implies a high cost, given the total number of operators scales as $\mathcal{O}(N^4)$. In [224] a graph colouring algorithm is demonstrated, which requires the full commutation graph to be generated. Instead, the authors of [219] present an algorithm named SORTEDINSERTION which does not require generating the full commutation graph on average, only in the worst case. In summary, SORTEDINSERTION sorts the set of Pauli operators, P_j by the absolute value of their coefficients α_j. This is due to the dependence of the denominator of $\hat{R}$ on $\left|a_j^2\right|$. In that order, each P_j is checked with existing commuting collections of Pauli operators, if P_j commutes with all Paulis in that collection it is added to it; and we move on to P_{j+1}. As SORTEDINSERTION is a heuristic algorithm it does not require us to generate the full commuting graph, which has complexity $\Theta(nt^2)$, where t is the total number of terms; only in the worst case does SORTEDINSERTION require $\mathcal{O}(nt^2)$.

10.2.4 Measuring Commuting Operators Simultaneously

Finally, we must describe a way to actually do commuting measurements with a single state preparation. As noted in the footnotes to Section 10.2.2 it is not possible to measure a qubit in one basis followed by another with current hardware, instead, as all of the commuting Paulis in a group can be diagonalised by a single unitary [43, p.77] matrix, we must find a method of enacting this matrix in the circuit.

In [219], [223] this circuit is constructed via the stabiliser formalism [225], [226], which has its origins in error correction, see Chapter 4, where stabilisers are measurements which fix, or stabilise, the logical quantum state. To illustrate this, we will begin with the simplest possible example, the group $\{XX, YY, ZZ\}$, which

we can easily verify mutually commutes. The simultaneous eigenvectors (the basis which is orthonormal for all Paulis) is:

$$|\Phi^{\pm}\rangle = \frac{|00\rangle \pm |11\rangle}{\sqrt{2}}, \tag{10.24}$$

$$|\Psi^{\pm}\rangle = \frac{|01\rangle \pm |10\rangle}{\sqrt{2}}. \tag{10.25}$$

A Bell basis measurement can be prepared by the circuit in Figure 10.1. The operators

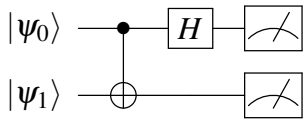


Figure 10.1: The Bell basis measurement, which measures in the basis formed by the vectors in Equation 10.24. The measurement outcome of qubit 0 corresponds to XX, and outcome on qubit 1 corresponds to ZZ. YY can be calculated from $-XX\,ZZ$.

we wish to measure are XX, ZZ, and those we can measure are ZI, IZ, and this circuit achieves the rotation that takes $XX \to ZI, ZZ \to IZ$ simultaneously. If we break this down in to transformations on the measurements, $UMU^{\dagger}$:

$$H_0 \text{CNOT}_{0,1} XX \text{CNOT}^{\dagger} H_0^{\dagger} \to H_0 X I H_0^{\dagger} \to ZI, \tag{10.26}$$

$$H_0 \text{CNOT}_{0,1} ZZ \text{CNOT}^{\dagger} H_0^{\dagger} \to H_0 I Z H_0^{\dagger} \to IZ. \tag{10.27}$$

This may be illuminating for a small example, but it is not a prescription for creating the measurement circuits.

We will use the stabiliser formalisation to proceed, where a group of m Pauli strings is represented by a $m \times 2n$ matrix, the first n rows contain a 1 if there is an Z on qubit j in the string, the second set of rows are the same for X, and Y is represented by a 1 in both sets. In some descriptions the matrices are transposed to be $2n \times n$, and a line can be drawn to separate the two sets of rows (columns). We can now do binary manipulations on the matrices to manipulate the Pauli strings. In the algorithms proposed in [219], [223] these manipulations represent quantum gates, with the end goal being a matrix representing $ZII\ldots, IZI\ldots, IIZ\ldots, \ldots$. For

the three qubit case with three Pauli operators, the desired matrix is:

$$\left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right]. \tag{10.28}$$

We can describe single and two-qubit gates as operations on the rows of the matrix, S, we have dropped the column index as it is the same for all gate operations:

- $\text{CNOT}_{i,j}$:
  - $S_i \rightarrow S_i + S_j$
  - $S_{j+n} \rightarrow S_{j+n} + S_{i+n}$
- $\text{CZ}_{i,j}$:
  - $S_i \rightarrow S_i + s_{j+n}$
  - $S_j \rightarrow S_j + S_{i_n}$
- H_i:
  - $S_i \leftrightarrow S_{i+n}$
- S_i:[2]
  - $S_i \rightarrow S_i + S_{i+n}$

[2]The S gate is called the phase, P gate in [219] and some other sources. It is also equal to $R_z(\pi/4)$. To be explicit, the matrix representation is:

$$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}. \tag{10.29}$$

We are also allowed to do classical post processing, e.g. computing $YY = -XX\,ZZ$, and relabelling qubits i, j in software[3]. We are also allowed to add a new column, i.e. a different stabiliser. In [219] they detail an algorithm for manipulating the matrix into the desired goal, e.g. Equation 10.28, which appends a maximum of $\mathcal{O}(mn/\log m)$ two qubit gates, where m is the number of Pauli strings.

10.3 Optimisation

Thanks to the variational principle, we are able to alter the parameters in our ansatz and save those with a lower energy, with the final energy we hope is the ground state. The scheme for altering the parameters is discussed in this section. In abstract, the VQE is no different from a classical parameterised objective function, which we are able to compute the gradients of, so we can utilise the wealth of research made into classical optimisers to minimise our VQE function. There is also a strong case to be made for using some unique properties of quantum computing to improve our optimisation, or to take into account the situation a quantum computer is ran under, such as the cloud model, considered in [227].

Many of the optimisers that we consider are gradient based, meaning they require us to calculate the gradient of the cost function (Hamiltonian) with respect to the gate parameters. In the following section we will discuss methods for calculating this gradient.

10.3.1 Gradient Calculations

In classical optimisation, we either have the analytical form of the function, so we can calculate the gradient, or we are able to use the finite differences method [136]. In NISQ quantum circuits, we are not able to use either of these methods; we do not have access to the wavefunction so cannot calculate the gradient analytically, and noise in the measurements is too large to make finite differences feasible. Fortunately we are able to exploit the form of the rotation gates to calculate the gradient.

As gradient-based optimisers are widely found in VQE experiments, calculation

[3]This can also be accomplished via the SWAP gate in the circuit, but in the NISQ era we want to move as much computation away from the quantum circuit as possible.

of the gradient has received much attention [155], [228]–[230]. The method we will review requires two circuit evaluations per parameter, multiplied by the number of shots required, although this can be reduced following [155]. In summary, the gradient is given by a shift of each parameter by $\frac{\pi}{2}$ around its current value:

$$\frac{\partial O(\theta)}{\partial \theta} = O\left(\theta + \frac{\pi}{2} - O\left(\theta - \frac{\pi}{2}\right)\right). \tag{10.30}$$

Here we will follow the treatment in [229], but any of [228]–[230] ar a good guide to this topic. We will begin by considering a gate $\mathcal{G}(\theta) = e^{-i\theta G}$ generated by a Hermitian operator G, e.g. the Pauli rotation gates. This gate is part of a longer circuit, with the expectation value given by:

$$\langle H \rangle = \langle 0 | U^\dagger(\theta) H U | \theta \rangle, \tag{10.31}$$

where θ denotes the vector of all parameters in the circuit, and we can re-write this as $\langle \psi | \mathcal{G}^\dagger Q \mathcal{G} | \psi \rangle$, where have used $U(\theta) = V\mathcal{G}(\theta)W$, and absorbed W into $|\psi\rangle = W|0\rangle$ and V into $Q = V^\dagger H V$. The partial derivative is then:

$$\partial_\theta \langle \psi | \mathcal{G}^\dagger Q \mathcal{G} | \psi \rangle = \langle \psi | \mathcal{G}^\dagger Q (\partial_\theta \mathcal{G}) | \psi \rangle + \langle \psi | (\partial_\theta)^\dagger \mathcal{G}^\dagger Q \mathcal{G} | \psi \rangle, \tag{10.32}$$

when the operators are Hermitian, this is

$$\frac{1}{2}\left(\langle \psi | (\mathcal{G} + \partial_\theta \mathcal{G})^\dagger Q (\mathcal{G} + \partial_\theta \mathcal{G}) | \psi \rangle - \langle \psi | (\mathcal{G} - \partial_\theta \mathcal{G})^\dagger Q (\mathcal{G} - \partial_\theta \mathcal{G}) | \psi \rangle \right). \tag{10.33}$$

Returning to the gate $\mathcal{G}(\theta) = e^{-i\theta G}$, the derivative is

$$\partial_\theta \mathcal{G} = -iGe^{-i\theta G}, \tag{10.34}$$

which we can substitute into Equation 10.33:

$$\partial_\theta \langle H \rangle = \langle \psi' | Q(-iG) | \psi' \rangle + \text{h.c.}, \tag{10.35}$$

and we have again absorbed $G|\psi\rangle = |\psi'\rangle$. If the gate G has two distinct eigenvalues, as is the case with the Pauli gates, we can shift the eigenvalues to the (unimportant) global phase, and parametrise the equation with r, which transforms Equation 10.33 to:

$$\partial_\theta \langle H \rangle = \frac{r}{2} \Big(\langle \psi' | \left(\mathbb{1} - ir^{-1}G\right)^\dagger Q \left(\mathbb{1} - ir^{-1}G\right) |\psi'\rangle \tag{10.36}$$

$$+ \langle \psi' | \left(\mathbb{1} + ir^{-1}G\right)^\dagger Q \left(\mathbb{1} + ir^{-1}G\right) |\psi'\rangle \Big). \tag{10.37}$$

If we can find a value θ such that $\mathcal{G}(\theta) = \frac{1}{\sqrt{2}}\left(\mathbb{1} \pm ir^{-1}G\right)$, we can calculate the gradient with two circuit evaluations.

When the gates we consider have eigenvalues $\pm r$, we can infer that $G^2 = r^2\mathbb{1}$, and by Taylor expanding $\mathcal{G}(\theta) = e^{-i\theta G}$, we get

$$e^{-i\theta G} = \sum_{k=0}^{\infty} \frac{(-i\theta)^k G^k}{k!} \tag{10.38}$$

$$= \sum_{k=0}^{\infty} \frac{(-i\theta)^{2k} G^{2k}}{(2k)!} + \sum_{k=0}^{\infty} \frac{(-i\theta)^{2k+1} G^{2k+1}}{(2k+1)!} \tag{10.39}$$

$$= \mathbb{1} \sum_{k=0}^{\infty} \frac{(-1)^k (r\theta)^{2k}}{(2k)!} - ir^{-1}G \sum_{k=0}^{\infty} \frac{(-1)^k (r\theta)^{2k+1}}{(2k+1)!} \tag{10.40}$$

$$= \mathbb{1}\cos(r\theta) - ir^{-1}G\sin(r\theta). \tag{10.41}$$

This is solved when $\theta = \frac{\pi}{4r}$. This completes the proof that we can calculate the analytocal gradient of the Hamiltonian (or cost function) via the parameter shift rule. Note that this is qualitatively similar to the finite differences method, but that here the difference between the two parameters, so it is not washed out by noise.

10.3.2 Gradient Based Optimisers

With a method for calculating the gradient of a parameter in a quantum circuit we can consider classical optimisers which require a gradient calculation. We will begin with the simplest scheme, Stochastic Gradient Descent (SGD).

10.3.2.1 Stochastic Gradient Descent

In all gradient descent optimisations we can imagine the problem as a multi dimensional landscape, and our optimum (or ground state) as the lowest point on the landscape. We change the parameters by a small amount in the direction and magnitude of the measured gradient. We additionally multiply the gradient by the learning rate, which is usually a number < 1 to minimise the effect of any large gradients encountered. SGD is the simplest form of gradient descent and applies parameter updates as the gradient multiplied by the learning rate, μ :

$$\theta_j^t = \theta_j^{t-1} - \mu \partial \theta_j^{t-1}, \tag{10.42}$$

where t indexes the step of the optimisation.

10.3.2.2 Momentum

The momentum optimiser does what it says on the tin: it adds momentum to the picture of a ball rolling down a hill we developed above. Momentum was developed in a classical optimisation perspective to allow the optimiser to 'roll' over flatter parts of the landscape that may exist[4]. The momentum optimiser adds a second update parameter, called the momentum, m_j which is stored between rounds, and is a modifier to the learning rate. There is another hyperparameter, here called β_0, which is usually set to 0.9 and is applied to the momentum update:

$$m_j^t = \beta_0 * m_j^{t-1} + \mu * \partial(\theta_j^{t-1}), \tag{10.43}$$

$$\theta_j^t = \theta_j^{t-1} + m_j^t. \tag{10.44}$$

10.3.2.3 AdaGrad

We can increase the specificity of these optimisers for each parameter, which the Adaptive Gradient algorithm (Adagrad) [231] does by storing a learning rate for each parameter. This is useful when a gradient vector is sparse, we would conclude that the update is very small, which is not useful for the weights that are in the sparse

[4]This does not help us to deal with barren plateaus, which is more unique to quantum optimisation settings, which we will discuss in Section 10.4.

vector. We also decrease this learning rate when a parameter receives more updates to prevent one parameter update dominating training. Introducing the decay vector, $\mathbf{v}$ which stores the squared gradient, the AdaGrad update rule is:

$$\mathbf{v}^t = \mathbf{v}^{t-1} + \left(\nabla\theta^t\right)^2, \tag{10.45}$$

$$\theta^{t+1} = \theta^t - \frac{\nu}{\sqrt{\mathbf{v}^t + \varepsilon}}\nabla\theta^t, \tag{10.46}$$

where we have introduced a small non-zero constant ε to prevent division by zero. Here we see that parameters with smaller gradient updates have a smaller value for v, which in turn increases the individual learning rate of that parameter.

10.3.2.4 AdaDelta

The Adadelta optimiser [232] is similar to the Adagrad optimiser, but instead of storing all of the previous gradients for each parameter in memory, something that can become very costly in deep learning scenarios, the rolling average of gradients is stored. This continues to update parameters after they receive many updates, which can decay towards the end of optimisations under AdaGrad.

10.3.2.5 Adam

The Adaptive moment estimation (Adam) optimiser [153] is most widely used in this thesis, it combines the ideas of individual parameter updates and momentum from the previous optimisers. One of the reasons for this is that it works well with noisy gradients. In the ball analogy the Adam optimiser is like a heavy ball with friction. Like AdaDelta, Adam stores a rolling average of past squared gradients for each parameter. Adam also stores the rolling average of past gradients:

$$m_j^t = \beta_1 m_j^{t-1} + (1-\beta_1)\,\partial\theta_j^{t-1}, \tag{10.47}$$

$$v_j^t = \beta_2 v_j^{t-1} + (1-\beta_2)\left(\partial\theta_j^{t-1}\right). \tag{10.48}$$

The authors of [153] found that as $\mathbf{v}, \mathbf{m}$ were initialised as zero vectors they were too heavily biased towards zero in initial steps. They correct for the bias by calculating

$$\hat{m}_j^t = \frac{m^t}{1-(\beta_1)^t}, \tag{10.49}$$

$$\hat{v}_j^t = \frac{v_j^t}{1-(\beta_2)^t}. \tag{10.50}$$

Here the parameters β_1, β_2 are the decay rates, and are hyperparameters that can be varied, but they are usually set to 0.9 and 0.999 respectively. The Adam update rule is then:

$$\theta_j^{t+1} = \theta_j^t - \frac{\nu}{\sqrt{\hat{v}^t} + \varepsilon}\hat{m}_j^t. \tag{10.51}$$

10.3.2.6 Model Gradient Descent

Model Gradient Descent (MGD) [227] is an optimisation algorithm developed for the most widely-used setting of quantum devices, that of a device in a remote lab accessed over the internet. In this setting the latency between receiving gradient values from the device, and calculating and sending parameter updates over the internet is significant, and can lead to a slow-down of overall computation. This becomes a larger problem when a user has paid for a block of time on a device, but spends a proportion of that time with the device idle whilst data is sent over the internet. Creating a model of the objective function, as MGD does, also allows us to utilise less shots in measurement of each point.

In MGD, we take the point we wish to evaluate and draw a radius around it of size δ, we then sample uniformly from this radius on the quantum device. As this does not require live updates to any parameters, we can send these circuits to the device as a single batch, reducing the idle time of the device. We save the results of this uniform sampling, so that we can use any points sampled that appear in the next radius. With the uniform sample of points, we use the classical device to create a least-squares fit model of the objective function within the radius δ, which we can calculate a gradient from. We then apply gradient descent to the parameters, giving us the new origin point to draw δ around. MGD also introduces a hyperparameter

which shrinks δ as the algorithm progresses, so that closer to the optimum we make a more precise estimation of the gradient. In [227], they show results of MGD preforming better in terms of a shorter wall clock time than other algorithms in the batched circuit setting, where we are only allowed to send a batch of circuits to the device at a time.

10.3.2.7 Rotosolve and Rotoselect

The Rotosolve and Rotoselect [158] algorithms are directly relevant to quantum problems. RotoSelect, and similar ideas published around the same time [233], [234] use the sinusoidal feature of the loss function for a single parameter (when all others are kept fixed) to find the minimum for that parameter without using gradient descent. This is given by:

$$\theta_d^* = \phi - \frac{\pi}{2} - \arctan2\left(2\langle M\rangle_\phi - \langle M\rangle_{\phi+\frac{\pi}{2}}, \langle M\rangle_{\phi+\frac{\pi}{2}} - \langle M\rangle_{\phi-\frac{\pi}{2}}\right) + 2\pi k, \quad (10.52)$$

where k is any integer and can be chosen to ensure $\theta_d^* \in (-\pi, \pi]$, and ϕ is real. Once all angles in the ansatz have been minimised this is repeated until some given convergence criteria are met. The RotoSelect algorithm improves on this by calculating $\theta_d^*(P)$ for each possible Pauli gate, P and changing the gate to that which minimises the energy; it then also repeats this for the whole circuit until convergence.

10.3.3 Hyperparameter Searches

If using the Adam optimiser, we now have at least three hyperparameters to choose[5], the learning rate and β_1, β_2. We can even consider our choice of optimiser as a hyperparameter. If we include ansatz specific choices, for example as we will see in Chapter 12 such as the choice of entangling gate or how many parameters to 'freeze' in an especially long circuit, we see that the number of different combinations of hyperparameters to try is growing rapidly. When optimising an ansatz that is not converging well it is difficult to know where to begin in choosing good parameters, especially as this is a young field with few examples of hyperparameters published. Therefore in the work leading to Chapter 12 we conducted a hyperparameter search,

[5]The parameter ε to prevent division by 0 is hardly ever changed from a default value of 10^{-8}.

where the space of all hyperparameters was reduced by searching for combinations that produced good convergence rates.

As it would be impossible to brute-force search all combinations of hyperparameters, a combination of randomised values is created. This has been shown independently to perform better than an ordered search [235]. In Chapter 12 we will present a combination of hyperparameters, differing from the default values, which may be of use in other VQE experiments.

10.4 Barren Plateaus

The barren plateaus problem was first presented in a Quantum Machine Learning context [159] yet it is also relevant to some VQE problems, and is suited to being presented here, within the section on optimisation. We are also reviewing barren plateaus here as it is relevant to the work presented in Chapter 12.

A barren plateau is a portion of the training landscape (the loss function in QML, or the problem Hamiltonian in VQE) where the gradient of the function vanishes for all relevant parameters. This clearly presents a problem for training as any gradient based optimiser is unable to minimise in a good direction. Non-gradient based optimisers, such as Nelder-Mead [236] also suffer, as a barren plateau exhibits little to no changes in the objective function, causing the optimiser to terminate early away from the global optimum.

Barren plateaus have been shown to occur in problems using the Hardware-Efficient ansatz[6], where the circuit can be thought of as a random, parameterised quantum circuit:

$$U(\theta) = \Pi_{l=1}^{L} U_l(\theta_l) W_l, \tag{10.53}$$

where $U_l(\theta_l)$ is the Hermitian operator generated by the parameterised gate V_l: $U_l(\theta_l) = e^{-i\theta_l V_l}$, and W_l is the unitary part of the circuit that has no dependence on θ_l, e.g. entangling gates. The gradient of a single parameter, θ_k, is defined in a similar manner to Equation 10.32, where we make the substitution $G = V$ to maintain

[6] As there is no chemistry to inspire a chemically-inspired ansatz in QML problems the barren plateaus problem is larger in QML. However, we still see the problem of vanishing gradients when using chemically-inspired ansatz, as we will see in Chapter 12.

consistency with the exposition in [159]:

$$\partial_k E = i \langle 0 | U_-^\dagger \left[V_k, U_+^\dagger H U_+ \right] U_- | 0 \rangle , \tag{10.54}$$

where U_- runs from $l = 0 \ldots k-1$, and U_+ runs from $l = k \ldots L$, i.e. the circuit before the parameterised gate k and the circuit including and after it. We then assume that the hardware efficient circuits making up U_-, U_+ are independent and that one or both match the Haar measure to the second moment. This is a reasonable assumption to make for most hardware efficient ansatze, and we will see how methods to overcome barren plateaus usually involve breaking these assumptions.

The Haar measure is a measure defined on the unitary group $U(N)$ for an N dimensional system and to sample from the Haar measure samples perfectly randomly from this group [43]. To say that a circuit samples from the Haar measure to the n-th moment means that the unitaries match the Haar measure up to the n-th statistical moment, i.e. for the second moment we mean that the unitaries sampled by $U_\pm$ match the Haar measure in their average and variance. This is the topic covered by unitary t-designs, which tries to find circuits that match the Haar measure up to the t-th moment [237]–[239].

With the assumption that $U_\pm$ are unitary 2-designs, we can use the properties of the Haar measure, $d\mu_{\text{Haar}}(U) = d\mu(U)$ on the unitary group, $U(N)$:

$$\int_{U(N)}^{d\mu(U) f(U)} = \int_{U(N)}^{d\mu(U) f(VU)} = \int_{U(N)}^{d\mu(U) f(UV)} . \tag{10.55}$$

We follow [159] in defining the average gradient, as this changes for each circuit implementation, so we define it using the probability distribution function of U, $p(U)$,

$$\langle \partial E \rangle = \int dU \, p(U) \partial_k \langle 0 | U^\dagger(\theta) H U(\theta) | 0 \rangle . \tag{10.56}$$

By assuming the independence of $U_\pm$ we can write $p(U)$ as

$$p(U) = \int dU_+ p(U_+) \int dU_- p(U_-) \delta \left(U_+ U_- - U \right) . \tag{10.57}$$

The Dirac delta function allows us to take the trace of the second part, and taking the trace allows us to re-order the terms in $\langle 0| U_-^\dagger \left[V_k, U_+^\dagger H U_+\right] |0\rangle$ and substitute $\rho_- = U_-|0\rangle\langle 0|U_-^\dagger$. This gives the definition of $\langle \partial_k E\rangle$,

$$\langle \partial_k E\rangle = i \int dU_- p(U_-) \mathrm{Tr}\left[\rho_- \int dU_+ p(U_+) \left[V, U_+^\dagger H U_+\right]\right]. \tag{10.58}$$

We can use the property of unitary 1-designs over the Haar measure:

$$\int d\mu(U) U O U^\dagger = \frac{\mathrm{Tr}[O]}{N} I. \tag{10.59}$$

Focusing firstly on the case where U_- is at least a 1-design, we substitute $O = |0\rangle\langle 0|$, which has trace of 1 to reduce the above to:

$$\langle \partial_k E\rangle = \frac{i}{N} \mathrm{Tr}\left[\left[V, \int dU_+ p(U_+) U_+^\dagger H U_+\right]\right] \tag{10.60}$$

$$= 0. \tag{10.61}$$

The final equality is obtained from the trace of a commutator of trace class operators[7]is zero.

Secondly, when U_+ is a 1-design, we utilise the property in Equation 10.59 within the Trace function:

$$\langle \partial_k E\rangle = i \int dU_- p(U_-) \mathrm{Tr}\left[\rho_- \int d\mu(U_+) \left[V, U_+^\dagger H U_+\right]\right] \tag{10.62}$$

$$= \frac{i\mathrm{Tr}[H]}{N} \int dU_- p(U_-) \mathrm{Tr}\left[\rho_- [V, I]\right], \tag{10.63}$$

$$= 0, \tag{10.64}$$

as the commutator of any operator and the identity is zero.

Whilst we have shown that $\langle \partial_k E\rangle = 0$ when U_+, U_- are independent and one is at least a 1-design, this does not demonstrate barren plateaus; i.e. the average gradient of a sine function as defined this way is zero. We must therefore show that

[7]A trace class operator is one for which a trace can be defined, and it is finite and independent of basis choice [43], which is true for our operator.

the variance of the gradient vanishes. Recall the variance in Equation 10.13, which in this case is $\text{Var}[\partial_k E] = \langle(\partial_k E)^2\rangle + \langle\partial_k E\rangle^2$, but we have just shown that $\langle\partial_k E\rangle = 0$ so we need only to calculate

$$\text{Var}[\partial_k E] = \langle(\partial_k E)^2\rangle. \tag{10.65}$$

The second moment of the Haar measure is given by the Weingarten functions [240], which for the second moment is:

$$\int d\mu(U) U_{i_1 j_1} U_{i_2 j_2} U^*_{i'_1 j'_1} U^*_{i'_2 j'_2} = \frac{\delta_{i_1 i'_1}\delta_{i_2 i'_2}\delta_{j_1 j'_1}\delta_{j_2 j'_2} + \delta_{i_1 i'_2}\delta_{i_2 i'_1}\delta_{j_1 j'_2}\delta_{j_2 j'_1}}{N^2-1} \tag{10.66}$$

$$- \frac{\delta_{i_1 i'_1}\delta_{i_2 i'_2}\delta_{j_1 j'_2}\delta_{j_2 j'_1} + \delta_{i_1 i'_2}\delta_{i_2 i'_1}\delta_{j_1 j'_1}\delta_{j_2 j'_2}}{N(N^2-1)}. \tag{10.67}$$

we can use diagrammatic notation to simplify this problem, which we will show in Appendix B.

Following the diagrammatic notation in the appendix, we are left with three equations for the variance, where U_+ is a 2-design, where U_- is a 2-design and where both are. We also here define $H_u = u^\dagger H u, \rho_u = u\rho u^\dagger$ and $\langle f(u)\rangle_x$ which is the average over u sampled from $p(U_x)$. When U_- is a two design:

$$\text{Var}[\partial_k E] = \frac{2\text{Tr}\left[\rho^2\right]}{N^2}\text{Tr}\left[\langle H_u^2 V^2 - (H_u V)^2\rangle_{U_+}\right] \tag{10.68}$$

$$= -\frac{\text{Tr}\left[\rho^2\right]}{2^{2n}}\text{Tr}\left[\langle [V, H_u]^2\rangle_{U_+}\right]. \tag{10.69}$$

When only U_+ is a 2-design:

$$\text{Var}[\partial_k E] = \frac{2\text{Tr}\left[H^2\right]}{N^2}\text{Tr}\left[\langle \rho_u^2 V^2 - (\rho_u V)^2\rangle_{U_-}\right] \tag{10.70}$$

$$= -\frac{\text{Tr}\left[H^2\right]}{2^{2n}}\text{Tr}\left[\langle [V, \rho_u]^2\rangle_{U_-}\right]. \tag{10.71}$$

Finally, in the case where both U_-, U_+ are 2-designs:

$$\text{Var}[\partial_k E] = 2\text{Tr}\left[H^2\right]\text{Tr}\left[\rho^2\right]\left(\frac{\text{Tr}\left[V^2\right]}{2^{3n}} - \frac{\text{Tr}[V]^2}{2^{4n}}\right). \tag{10.72}$$

Each of these cases is preceded by a factor of at least 2^{-2n}, showing that the gradient vanishes exponentially in the number of qubits.

10.4.1 Noise Induced Barren Plateaus

Building upon [159] noise induced barren plateaus [175] introduce noise in the quantum device to the discussion of barren plateaus. In [175] they study the magnitude of the gradient derivative $|\partial_k E|$ with respect to the number of qubits, n, the number of layers in an ansatz, L and the noise in the circuit, parameterised by a single parameter, $q = \max\{|q_X|, |q_Y|, |q_Z|\}$. Here, q_σ is the strength of the noise model, $\mathcal{N}_j$ on each qubit, j for each Pauli operator, the same noise model as we presented in Section 6.1.4

$$\mathcal{N}_j(\sigma) = q_\sigma \sigma. \tag{10.73}$$

They demonstrate that

$$|\partial_k E| \leq \sqrt{8\ln 2} N_O \, ||H_k||_\infty \, ||\omega||_\infty \, n^{1/2} q^{L+1}, \tag{10.74}$$

where $||.||_\infty$ is the infinity norm, N_O is the number of non-zero elements in the operator O, and ω is a vector of coefficients in the cost Hamiltonian H_k.

We can express any Hermitian operator Λ as

$$\Lambda = \lambda_0 \mathbb{1}^{\otimes n} + \lambda \dot{\sigma}_n, \tag{10.75}$$

with λ the vector of coefficients for the Pauli strings, represented in σ_n (excluding $\mathbb{1}$). The noise model $\mathcal{N}$ maps $\lambda_i \to \lambda_i' = q_X^{n_{x,i}} q_Y^{n_{y,i}} q_Z^{n_{z,i}}$, where $n_{\sigma,i}$ denotes the number of Pauli operators σ in the string i. With these definitions, we can show that $|\lambda'| \leq q\,|\lambda|$ holds, as $n_{X,i} + n_{Y,i} + n_{Z,1} \geq 1$. With a noise model of this form, which implements a set of noisy gates on each qubit after each layer of the circuit, i.e.

$$\mathcal{W} = \mathcal{N}\mathcal{U}_L \mathcal{N} \ldots \mathcal{U}_0. \tag{10.76}$$

With this form of circuit, $\left|\left|\mathcal{W}^\dagger(H)\right|\right|_\infty \leq q^L N_O \,||||_\infty$. This gives us the outline for the

proof in [175], in which they use this form of circuit and noise model, the strong data processing inequality, that symmetric depolarising noise pushes a quantum state closer to the maximally mixed state [241] to get the bound in Equation 10.74.

This shows us that the trainability of a variational quantum circuit, as well as the final accuracy can be affected by noise in the quantum device.

We will now review some methods for combatting the barren plateaus phenomena.

10.4.2 Local Cost Functions

In the work on local cost functions [242] we are considering the same problem as above, but we now look closer at the cost function, which generates E. So far, as we have mostly been discussing chemical ansatze we assume that the cost function is the Hamiltonian, H, and $\mathbf{E} = \langle\psi(\theta)|H|\psi(\theta)\rangle$. When discussing quantum machine learning problems, we have more freedom to choose the cost function, and we can define a general cost function as the expectation value of any operator

$$C = \mathrm{Tr}\left[OV(\theta)\rho V^{\dagger}(\theta)\right], \tag{10.77}$$

where the operator has the form

$$O = c_0\mathbb{1} + \sum_{i=1}^{N} c_i \hat{O}_{i1} \otimes \hat{O}_{i2} \otimes \cdots \otimes \hat{O}_{i\xi}. \tag{10.78}$$

We can therefore discuss two instances of the cost function: (i): where $N = 1$ and each $\hat{O}_{1k}$ is a non-trivial projector with rank r_k acting on subsystem S_k. One example of case (i) is the trace distance between the trained state $|\psi'\rangle = V(\theta)|0\rangle$, and a reference $|\psi_0\rangle$. The cost function in this case is $C_G = \mathrm{Tr}\left[O_G V(\theta)|\psi_0\rangle\langle\psi_0|V^{\dagger}(\theta)\right]$, with $O_G = \mathbb{1} - |\mathbf{0}\rangle\langle\mathbf{0}|$. In the second case, (ii): we have arbitrary N and $\hat{O}_{ik}$ is traceless, e.g. $\mathrm{Tr}\left[\hat{O}_{ik}\right] = 0$ and $\mathrm{Tr}\left[\hat{O}_{ik}^2\right] \leq 2^m$. A physical Hamiltonian is a good example of this class of cost function, as we can have $\hat{O}_{ik} = \bigotimes_{j=1}^{m} \sigma_j^{\mu}$ with σ^{μ} the Pauli operators. We can think of m as defining the locality of the operators, which will be relevant in what follows.

Following [242], they re-formulate the result in [159], Equation 10.72 in terms of the cost function as defined above. In case (i) they find

$$\mathrm{Var}\left[\partial C\right] \leq \mathcal{O}\left(2^{-(1-1/m\log_2 3)n}\right), \tag{10.79}$$

and in case (ii):

$$\mathrm{Var}\left[\partial C\right] \leq \mathcal{O}\left(\frac{1}{2^{(1-1/m)}n}\right), \tag{10.80}$$

both of which imply that the variance of the gradient vanishes when $m \geq 2$.

In [242] there is also a small example that demonstrates the usefulness of a local cost function when available: If we wish to do the Trace distance of some variational circuit, $V(\theta) = \bigotimes_{j=1}^{n} e^{-i\theta_j \sigma_x^j/2}$, with the same target and starting state, $|\mathbf{0}\rangle$, we can either employ a global cost function,

$$C_G = \mathrm{Tr}\left[O_G V(\theta)|\psi_0\rangle\langle\psi_0|V^\dagger(\theta)\right], \tag{10.81}$$

$$O_G = \mathbb{1} - |\mathbf{0}\rangle\langle\mathbf{0}|. \tag{10.82}$$

With such a simple circuit (just R_x rotations), we can analytically calculate the cost function as

$$C_G = 1 - \Pi_{j=1}^{n} \cos^2\left(\frac{\theta_j}{2}\right), \tag{10.83}$$

with the variance of the gradient is $\mathrm{Var}\left[\frac{\partial C_G}{\partial \theta_j}\right] = \frac{1}{8}\left(\frac{3}{8}\right)^{n-1}$.

If we now consider a local cost function,

$$C_L = \mathrm{Tr}\left[O_L V(\theta)|\psi_0\rangle\langle\psi_0|V^\dagger(\theta)\right], \tag{10.84}$$

$$O_L = \mathbb{1} - \frac{1}{n}\sum_{j=1}^{n} |0\rangle\langle 0|_j \otimes \mathbb{1}_{\bar{j}}, \tag{10.85}$$

with $\mathbb{1}_{\bar{j}}$ the identity on all qubits except j. This cost function evaluates to

$$C_L = 1 - \frac{1}{n}\sum_{j=1}^{n} \cos^2\left(\frac{\theta_j}{2}\right), \tag{10.86}$$

giving $\text{Var}\left[\frac{\partial C_G}{\partial \theta_j}\right] = \frac{1}{8n^2}$. We see that the global cost function vanishes exponentially with n, whilst the local cost function vanishes polynomially in n.

Ref. [242] also finds a lower bound on the variance in terms of circuit depth L, for m-local cost functions. For shallow circuits, $L \in \mathcal{O}(\log(n))$,

$$\Omega\left(\frac{1}{\text{poly}(n)}\right) \leq \text{Var}[\partial C], \tag{10.87}$$

and for deeper circuits, $L \in \mathcal{O}(\text{poly}(\log(n)))$ they find the gradient vanishes faster than polynomially, but slower than exponentially, i.e.

$$\Omega\left(\frac{1}{2^{\text{poly}(\log(n))}}\right) \leq \text{Var}[\partial C]. \tag{10.88}$$

We therefore have a guide to designing training schemes and cost functions in variational quantum algorithms. Unfortunately in the case of VQE on a molecular system, we are unable to choose our cost function with such freedom, as it is derived from the molecule. However, we will take into account the second part of this research in Chapter 12, as we will upper bound L by freezing the total number of trainable parameters.

A similar strategy for general variational algorithms (e.g. quantum machine learning) is proposed in [243]; which proposes an ansatz, splits it up into layers and appends and trains each layer in turn. This is similar to the ADAPT algorithm [41] , which we will build upon in Chapter 12, except that there is no strategy proposed for choosing the layers in the ansatz.

10.4.3 Initialisation Strategies

Instead of targeting the cost function to reduce the incidence of barren plateaus, in [244] the authors attack the assumptions the unitary 2-design property of the ansatz circuit. They propose an ansatz made of shallow blocks, which is parameterised by θ_j, each block has the form

$$U_m(\theta_m) = \Pi_{l=L}^{1} U_l(\theta_{l,1}^m) \Pi_{l=1}^{L} U_l(\theta_{l,2}^m), \tag{10.89}$$

where $\theta^m_{l,2}$ is chosen such that $U_l(\theta^m_{l,2}) = U_l(\theta^m_{l,1})^\dagger$ and each block, and therefore the whole circuit evaluates to the identity. We therefore increase the chances of a non-zero gradient for each parameter in the first optimisation step, with no guarantees on subsequent steps. In numerical trials presented in [244] they do show that the variance of the gradient remains higher in subsequent steps with this optimisation strategy, compared with a randomised optimisation strategy.

Again, we will use this result in Chapter 12, where we show that optimisation close to the identity for each new operator produces a better training optimisation.

10.4.4 Parameter Correlation

In [245] they consider the case of correlating parameters to address the barren plateaus phenomena. Similarly to [244] above, this method attempts to sever the link between the ansatz and the unitary 2-design. By reducing the total parameter search space, correlating parameters can make the optimisation easier in general; with the caveat that the global minimum may not be within the smaller parameter space, this is explored in Section 10.4.5 and [215].

The correlation of parameters can be made over space (all qubits in a certain layer have the same parameter), or time (a qubit has the same parameter over multiple gates in the circuit), or a mixture of both. This approach is also closely related to the QAOA ansatz [121], where the entangling layers all share a parameter, and the single qubit gates share a different parameter.

Considering the tractable problem described in Section 10.4.2 of quantum compiling the identity transformation, with a global cost function as Equation 10.81, the ansatz proposed is correlated over space; that is, the circuit on each qubit is the same, with the same parameters. The correlated ansatz can be written

$$\mathcal{M}(\theta) = \left(\Pi_{i=1}^{L} e^{-i\theta_i/2\sigma_y}\right)^{\otimes n}. \tag{10.90}$$

For pure input states, considering again θ_k we have

$$\langle(\partial_k C)^2\rangle = n^2 \int d\mu(\theta) \cos^{2(2n-1)}\left(\frac{1}{2}\sum_{i=1}^{L}\theta_i\right) \sin^2\left(\frac{1}{2}\sum_{i=1}^{L}\theta_i\right), \tag{10.91}$$

with $d\mu(\theta) = \frac{1}{(2\pi)^L}\Pi_{i=1}^{L} d\theta_i$. We can make the substitution $v_L = \sum_{i=1}^{L} \theta_i \text{mod } 2\pi$:

$$\langle(\partial_k C)^2\rangle = n^2 \int \frac{dv_L}{2\pi} \cos^{2(2n-1)}\left(\frac{v_L}{2}\right) \sin^2\left(\frac{v_l}{2}\right) \tag{10.92}$$

$$= \frac{2n^2}{4n-1} \int_{-\pi/2}^{\pi/2} \frac{du}{2\pi} \cos^{4n}(u) \tag{10.93}$$

$$= \frac{n^2}{2^{4n}(4n-1)} \left(\binom{4n}{2n}\right) \tag{10.94}$$

$$\approx \frac{n^{1/2}}{4\sqrt{2\pi}}, \tag{10.95}$$

with the second substitution $u = \frac{v_L}{2}$ and the approximation (true in the asymptotic limit) $\left(\binom{2l}{l}\right) \approx \frac{2^{2l}}{\sqrt{\pi l}}$. We see that this is polynomial in n, as opposed to the exponential in Equation 10.79 (as here $m = n \geq 2$). This toy example is only one of the examples explored in [245], where they elaborate on more general examples, and provide numerical evidence.

10.4.5 Barren Plateaus and Expressibility

In [215] they make an explicit connection between the trainability of an ansatz and the expressibility of an ansatz. This result is hinted at by the conclusions we draw from [245], that restricting the total space available to the parameters increases the magnitude and variance of the cost function gradient. As we can no longer say that ansatze with these restrictions represent 2-designs, the work in [215] relates the distance of an ansatz from a good 2-design to the variance of the gradient. The conclusions drawn are that a less expressive ansatz will have lager gradient variance and be easier to train; the challenge for the quantum scientist is therefore to find some ansatz which has reduced expressibility but nevertheless contains the solution to the problem in its parameter space.

We will review the proof given in [215]. We begin with the general form of the cost function (with no assumptions made on its locality) written again here to aid exposition:

$$C_{\rho,H}(\theta) = \text{Tr}\left[HU(\theta)\rho U(\theta)^\dagger\right], \tag{10.96}$$

which suffices for a VQE cost function, although in quantum machine learning

approaches we may be using multiple measurements or input states,

$$C = \sum_i \mathrm{Tr}\left[H_i U(\theta)\rho_i U(\theta)^\dagger\right]. \tag{10.97}$$

which we will will extend the results to.

The ensemble of unitaries expressed by our ansatz is denoted $\mathbb{U}$, this is the set of unitaries generated by the set of parameters $\left\{\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(y)}\right\}$, where each vector of parameters $\theta^{(y)}$ is different. We introduce a superoperator to quantify the expressibility of an ansatz, given a measurement operator, which is expressed in terms of distance from the Haar measure to the t-th moment.

$$\mathcal{A}_{\mathbb{U}}(t)(\cdot) := \tag{10.98}$$

$$\int_{\mathcal{U}(d)} d\mu(V) V^{\otimes t}(\cdot)V^{\dagger\otimes t} - \int_{\mathbb{U}} dU U^{\otimes t}(\cdot)U^{\dagger\otimes t}, \tag{10.99}$$

with $d\mu(V)$ the volume element of the Haar measure, $\mathcal{U}(d)$ the Haar distribution, and dU is the volume element of the uniform distribution over $\mathbb{U}$. If $\mathbb{U}$ forms a 2-design then $\mathcal{A}^{(2)}_{\mathbb{U}}(X) = 0 \forall X$, which is the situation covered in the original barren plateaus paper, [159]. We will herein consider only 2-designs, so $t = 2$ and can be omitted from notation.

Three measures of the expressibility are proposed, the first two dependent on ρ, H and therefore problem specific, and the third uses the diamond norm so can compare ansatze over different problems,

$$\varepsilon^{\rho}_{\mathbb{U}} = \left|\left|\mathcal{A}_{\mathbb{U}}\left(\rho^{\otimes 2}\right)\right|\right|_2 \tag{10.100}$$

$$\varepsilon^{H}_{\mathbb{U}} = \left|\left|\mathcal{A}_{\mathbb{U}}\left(H^{\otimes 2}\right)\right|\right|_2 \tag{10.101}$$

$$\varepsilon^{\diamond} = \left|\left|\mathcal{A}_{\mathbb{U}}\right|\right|_{\diamond}. \tag{10.102}$$

The diamond norm is computationally expensive, as it is the 1-norm of $\mathcal{A}_{\mathbb{U}}(\rho)$ maximised over all ρ.

We will begin by showing the separate cases for either the left, L or right, R of parameter k are close to a 2-design, and then move onto the case where both are

close to a 2-design.

For legibility, we will make the substitutions $\int_{\mathbb{U}_R} dU_R := \int_R$, $\int_{\mathbb{U}_L} dU_L := \int_L$, and for the Haar measure, $\int_{\mathcal{U}(d)} d\mu(V) = \int_{\mathcal{U}}$. The superoperator $\mathcal{A}$ is related to the variance when the left or right is a 2-design, $\mathrm{Var}_L[\partial_k C], \mathrm{Var}_R[\partial_k C]$:

$$\mathcal{A}(\rho^{\otimes 2}) = \int_{\mathcal{U}} V^{\otimes 2}\rho^{\otimes 2}V^{\dagger\otimes 2} - \int_R U_R^{\otimes 2}\rho^{\otimes 2}U_R^{\dagger\otimes 2} \tag{10.103}$$

$$= \mathrm{Var}_R[\partial_k C] - \int_R U_R^{\otimes 2}\rho^{\otimes 2}U_R^{\dagger\otimes 2} \tag{10.104}$$

$$\mathcal{A}(H^{\otimes 2}) = \int_{\mathcal{U}} V^{\otimes 2}H^{\otimes 2}V^{\dagger\otimes 2} - \int_L U_L^{\otimes 2}H^{\otimes 2}U_L^{\dagger\otimes 2} \tag{10.105}$$

$$= \mathrm{Var}_L[\partial_k C] - \int_L U_L^{\otimes 2}H^{\otimes 2}U_L^{\dagger\otimes 2}. \tag{10.106}$$

Recalling the partial derivative of the cost function:

$$\partial_k C = i\mathrm{Tr}\left[U_R\rho U_R^\dagger\left[V_k, U_L^\dagger H U_L\right]\right], \tag{10.107}$$

where the parameterised gate in question is generated by V_k, and a relation for the trace of operators in the same Hilbert space:

$$\mathrm{Tr}[A]\,\mathrm{Tr}[B] = \mathrm{Tr}[A\otimes B], \tag{10.108}$$

we can write the variance of the gradient as:

$$\mathrm{Var}[\partial_k C] = -\int_L\int_R \mathrm{Tr}\left[U_R^{\otimes 2}\rho^{\otimes 2}U_R^{\dagger\otimes 2}\left[V_k, U_L^{\dagger\otimes 2}H^{\otimes 2}U_L^{\otimes 2}\right]\right] \tag{10.109}$$

$$= -\int_L\int_R \mathrm{Tr}\left[\rho_R^{\otimes 2}\left[V_k, H_L^{\otimes 2}\right]\right], \tag{10.110}$$

here making the substitutions $\rho_R := U_R\rho U_R^\dagger, H_L := U_L^\dagger H U_L$. We will now show the derivation for the left and right parts of the circuit concurrently, and using the cyclic permutation of the trace for the right hand side, we begin by making a substitution.

$$X_{L,k} = [V_k, H_L] \qquad Y_{R,k} = [\rho_R, V_k]$$

Using the definitions of $\mathcal{A}$ above:

$$\mathrm{Var}\left[\partial_k C\right] = \mathrm{Var}_R\left[\partial_k C\right] + \int_L \mathrm{Tr}\left[\mathcal{A}_R(\rho^{\otimes 2}) X_{L,k}^{\otimes 2}\right] \qquad \mathrm{Var}\left[\partial_k C\right] = \mathrm{Var}_L\left[\partial_k C\right] + \int_R \mathrm{Tr}\left[\mathcal{A}_L(H^{\otimes 2}) Y_{R,k}^{\otimes 2}\right]$$

$$\left|\mathrm{Var}\left[\partial_k C\right] - \mathrm{Var}_R\left[\partial_k C\right]\right| \qquad \left|\mathrm{Var}\left[\partial_k C\right] - \mathrm{Var}_L\left[\partial_k C\right]\right|$$

$$\leq \left| \int_L \mathrm{Tr}\left[\mathcal{A}_R(\rho^{\otimes 2}) X_{L,k}^{\otimes 2}\right] \right| \qquad \leq \left| \int_R \mathrm{Tr}\left[\mathcal{A}_L(H^{\otimes 2}) Y_{R,k}^{\otimes 2}\right] \right|$$

Using the Triangle inequality:

$$\leq \int_L \left| \mathrm{Tr}\left[\mathcal{A}_R(\rho^{\otimes 2}) X_{L,k}^{\otimes 2}\right] \right| \qquad \leq \int_R \left| \mathrm{Tr}\left[\mathcal{A}_L(H^{\otimes 2}) Y_{R,k}^{\otimes 2}\right] \right|.$$

We will combine the two derivations, writing

$$\mathcal{A} = \begin{cases} \mathcal{A}_R\left(\rho^{\otimes 2}\right) & \text{LH column} \\ \mathcal{A}_L\left(H^{\otimes 2}\right) & \text{RH column.} \end{cases} \quad \chi = \begin{cases} X_{R,k} & \text{LH column} \\ Y_{L,k} & \text{RH column} \end{cases} \tag{10.111}$$

$$\left|\mathrm{Var} - \mathrm{Var}_{R(L)}\right| \leq \int_{\mathbb{U}_{R(L)}} \left|\left|\chi^{\otimes 2}\right|\right|_2 ||\mathcal{A}||_2 , \tag{10.112}$$

$$\left|\left|\chi^{\otimes 2}\right|\right|_2 = \sqrt{\mathrm{Tr}\left[\chi^{\otimes 2}\chi^{\otimes 2}\right]} = \sqrt{\mathrm{Tr}\left[\chi^2 \otimes \chi^2\right]} = \left|\mathrm{Tr}\left[\chi^2\right]\right| = \left|\mathrm{Tr}\left[[A,B]^2\right]\right|. \tag{10.113}$$

Consistent with out substitutions,

$$A = \begin{cases} V_k & \text{LH column} \\ \rho_R & \text{RH column,} \end{cases} \quad B = \begin{cases} H_L & \text{LH column} \\ V_k & \text{RH column} \end{cases} \tag{10.114}$$

$$\left|\mathrm{Tr}\left[[A,B]^2\right]\right| = \mathrm{Tr}\left[2ABAB - 2A^2B^2\right] \tag{10.115}$$

$$= 2\left|\mathrm{Tr}\left[ABAB\right] - \mathrm{Tr}\left[A^2B^2\right]\right| \tag{10.116}$$

$$\leq 2\left(\left|\mathrm{Tr}\left[ABAB\right]\right| + \left|\mathrm{Tr}\left[A^2B^2\right]\right|\right) \tag{10.117}$$

using first the cyclic permutation of the trace and then the triangle inequality. We must now return to split derivations as

$A^2 = \mathbb{1}$

$\left|\mathrm{Tr}\left[Z^2\right]\right| = 2\left(\left|\mathrm{Tr}\left[ABAB\right]\right| + \left|\mathrm{Tr}\left[B^2\right]\right|\right)$

Using the Cauchy-Schwarz inequality:

$\leq 2\left(\sqrt{\mathrm{Tr}\left[ABA\right]^2 \mathrm{Tr}\left[B^2\right]} + \left|\mathrm{Tr}\left[B^2\right]\right|\right)$

cyclic permutation of the trace:

$= 2\left(\sqrt{\mathrm{Tr}\left[AABBAAB\right] \mathrm{Tr}\left[B^2\right]} + \left|\mathrm{Tr}\left[B^2\right]\right|\right)$

$= 2\left(\sqrt{\mathrm{Tr}\left[B^2\right] \mathrm{Tr}\left[B^2\right]} + \left|\mathrm{Tr}\left[B^2\right]\right|\right)$

$= 4\left|\left|H\right|\right|_2^2.$

$B^2 = \mathbb{1}$

$\left|\mathrm{Tr}\left[Z^2\right]\right| = 2\left(\left|\mathrm{Tr}\left[ABAB\right]\right| + \left|\mathrm{Tr}\left[A^2\right]\right|\right)$

$\leq 2\left(\sqrt{\mathrm{Tr}\left[BAB\right]^2 \mathrm{Tr}\left[A^2\right]} + \left|\mathrm{Tr}\left[A^2\right]\right|\right)$

$= 2\left(\sqrt{\mathrm{Tr}\left[BBABBA\right] \mathrm{Tr}\left[A^2\right]} + \left|\mathrm{Tr}\left[A^2\right]\right|\right)$

$= 2\left(\sqrt{\mathrm{Tr}\left[A^2\right] \mathrm{Tr}\left[A^2\right]} + \left|\mathrm{Tr}\left[A^2\right]\right|\right)$

$= 4\left|\left|\rho\right|\right|_2^2.$

This gives us our result, that the gradient variance is bounded by:

$$\mathrm{Var}\left[\partial_k C\right] \leq \mathrm{Var}_R\left[\partial_k C\right] + 4\left|\left|\mathcal{A}_R\left(\rho\right)\right|\right|_2^2 \qquad \mathrm{Var}\left[\partial_k C\right] \leq \mathrm{Var}_L\left[\partial_k C\right] + 4\left|\left|\mathcal{A}_L\left(H\right)\right|\right|_2^2$$

The final relation we derive will find a bound on the variance when both L, R are close to 2-designs. We begin with Equation 10.107 and use the cyclic permutation of the trace to write

$$\begin{aligned}
\mathrm{Var}\left[\partial_k C\right] &= -\int_L \int_R \left(\mathrm{Tr}\left[\rho_R\left[V_k, H_L\right]\right]\right)^2 && (10.118)\\
&= \int_L \int_R \mathrm{Tr}\left[\rho_R^{\otimes 2}\left(V_k H_L V_k H_L + H_L V_k H_L V_k - 2 V_k H_L H_L V_k\right)\right] && (10.119)\\
&= \int_L \int_R \mathrm{Tr}\left[\rho_R^{\otimes 2}\left(V_k^{\otimes 2} H_L^{\otimes 2} + H_L^{\otimes 2} V_k^{\otimes 2} - 2\left(V_k \otimes \mathbb{1}\right) H_L^{\otimes 2}\left(\mathbb{1} \otimes V_k\right)\right)\right], && (10.120)\\
&= \int_{\mathcal{U}} \int_{\mathcal{U}} \mathrm{Tr}\Big[\\
&\left(\mathrm{Var}_R - \mathcal{A}_R(\rho_R)\right)\left(V_k^{\otimes 2}\left(\mathrm{Var}_L - \mathcal{A}_L(H)\right)\right. \\
&+ \left(\mathrm{Var}_L - \mathcal{A}_L(H)\right) V_k^{\otimes 2} \\
&\left. - 2\left(V_k \otimes \mathbb{1}\right)\left(\mathrm{Var}_L - \mathcal{A}_L(H)\right)\left(\mathbb{1} \otimes V_k\right)\right)\Big]. && (10.121)
\end{aligned}$$

In the final line we have substituted from Equations 10.103 - 10.105, the definition of $\mathcal{A}$ in terms of the variance given by a Haar random circuit. We can now collect

relevant terms into two integrals:

$$\begin{aligned}\mathrm{Var}\left[\partial_k C\right] = &\mathrm{Var}_{L,R}\left[\partial_k C\right] \\ &- \mathrm{Tr}\left[\mathcal{A}_R(\rho)\left(V_k^{\otimes 2}\mathcal{A}_L(H) + \mathcal{A}_L(H)V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\mathcal{A}_L(H)\left(\mathbb{1}\otimes V_k\right)\right)\right] \\ &+ \int_{\mathcal{U}} \mathrm{Tr}\left[\left(V_k^{\otimes 2}\mathcal{A}_L(H) + \mathcal{A}_L(H)V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\mathcal{A}_L(H)\left(\mathbb{1}\otimes V_k\right)\right)\tilde{\rho}^{\otimes 2}\right] \\ &+ \int_{\mathcal{U}} \mathrm{Tr}\left[\mathcal{A}_R(\rho)\left(V_k^{\otimes 2}\tilde{H}^{\otimes 2} + \tilde{H}^{\otimes 2}V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\tilde{H}^{\otimes 2}\left(\mathbb{1}\otimes V_k\right)\right)\right],\end{aligned} \tag{10.122}$$

with $\tilde{\rho} = U\rho U^\dagger$, $\tilde{H} = U^\dagger H U$, recalling that $\int_{\mathcal{U}} := \int_{\mathcal{U}(d)} d\mu(U)$.

For ease of notation, we can define

$$Z_{xk} := \left(V_k^{\otimes 2}\mathcal{A}_x(\omega_x) + \mathcal{A}_x(\omega_x)V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\mathcal{A}_x(\omega_x)\left(\mathbb{1}\otimes V_k\right)\right), \tag{10.123}$$

with $x = L(R), \omega_{L(R)} = \rho(H)$. As the integrals in Equation 10.122 are over the Haar measure, we can use the Weingarten formula [240], Equation 10.66 to solve them. We have two integrals to solve,

$$I_1 = \int_{\mathcal{U}} d\mu(U)\mathrm{Tr}\left[Z_{Lk}U_R^{\otimes 2}\rho^{\otimes 2}U_R^{\dagger\otimes 2}\right], \tag{10.124}$$

$$I_2 = \int_{\mathcal{U}} d\mu(U)\mathrm{Tr}\left[\mathcal{A}_R(\rho)\left(V_k^{\otimes 2}\tilde{H}^{\otimes 2} + \tilde{H}^{\otimes 2}V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\tilde{H}^{\otimes 2}\left(\mathbb{1}\otimes V_k\right)\right)\right]. \tag{10.125}$$

We can use the linearity of the Trace, and cyclic permutations to rearrange Equation 10.125,

$$\begin{aligned}&\mathrm{Tr}\left[\mathcal{A}_R(\rho)\left(V_k^{\otimes 2}\tilde{H}^{\otimes 2} + \tilde{H}^{\otimes 2}V_k^{\otimes 2} - 2\left(V_k\otimes\mathbb{1}\right)\tilde{H}^{\otimes 2}\left(\mathbb{1}\otimes V_k\right)\right)\right]. \\ =&\mathrm{Tr}\left[\mathcal{A}_R(\rho)V_K^{\otimes 2}U_L^{\dagger\otimes 2}H^{\otimes 2}U_L^{\otimes 2}\right] + \mathrm{Tr}\left[\mathcal{A}_R(\rho)U_L^{\dagger\otimes 2}H^{\otimes 2}U_L^{\otimes 2}V_K^{\otimes 2}\right] \\ &- 2\mathrm{Tr}\left[\mathcal{A}_R(\rho)\left(V_k\otimes\mathbb{1}\right)U_L^{\dagger\otimes 2}H^{\otimes 2}U_L^{\otimes 2}\left(\mathbb{1}\otimes V_k\right)\right], \\ =&\mathrm{Tr}\left[H^{\otimes 2}U_L^{\otimes 2}\mathcal{A}_R(\rho)V_K^{\otimes 2}U_L^{\dagger\otimes 2}\right] + \mathrm{Tr}\left[H^{\otimes 2}U_L^{\otimes 2}V_K^{\otimes 2}\mathcal{A}_R(\rho)U_L^{\dagger\otimes 2}\right] \\ &- 2\mathrm{Tr}\left[H^{\otimes 2}U_L^{\otimes 2}\left(\mathbb{1}\otimes V_k\right)\mathcal{A}_R(\rho)\left(V_k\otimes\mathbb{1}\right)U_L^{\dagger\otimes 2}\right],\end{aligned} \tag{10.126}$$

(10.127)

we can here use $\mathrm{Tr}\left[(\mathbb{1}\otimes V_k)\,\mathcal{A}_R(\rho)\,(V_k\otimes\mathbb{1})\right]=\mathrm{Tr}\left[(\mathbb{1}\otimes V_k)\,\mathcal{A}_R(\rho)\,(V_k\otimes\mathbb{1})\right]$ to simplify the final line to

$$\begin{aligned}&\mathrm{Tr}\left[H^{\otimes 2}U_L^{\otimes 2}\left(\mathcal{A}_R(\rho)V_K^{\otimes 2}+V_K^{\otimes 2}\mathcal{A}_R(\rho)-2\,(\mathbb{1}\otimes V_k)\,\mathcal{A}_R(\rho)\,(V_k\otimes\mathbb{1})\right)U_L^{\dagger\otimes 2}\right],\\&=\mathrm{Tr}\left[H^{\otimes 2}U_L^{\otimes 2}Z_{Rk}U_L^{\dagger\otimes 2}\right].\end{aligned}\tag{10.128}$$

In Appendix C we derive a relation relevant here using diagrammatic notation, and we will state the result, Equation C.2 here:

$$\begin{aligned}\int_{\mathcal{U}(d)}d\mu(U)\mathrm{Tr}\left[AU^{\otimes 2}BU^{\dagger\otimes 2}\right]&=\frac{1}{d^2-1}\left(\mathrm{Tr}\left[A\right]\mathrm{Tr}\left[B\right]+\mathrm{Tr}\left[AW\right]\mathrm{Tr}\left[BW\right]\right)\\&\quad-\frac{1}{d\,(d^2-1)}\left(\mathrm{Tr}\left[AW\right]\mathrm{Tr}\left[B\right]+\mathrm{Tr}\left[A\right]\mathrm{Tr}\left[BW\right]\right).\end{aligned}\tag{10.129}$$

We can easily substitute the rearranged I_1, I_2 here to give

$$\begin{aligned}I_1&=\frac{1}{d^2-1}\left(\mathrm{Tr}\left[Z_{Lk}\right]\mathrm{Tr}\left[\rho^{\otimes 2}\right]+\mathrm{Tr}\left[Z_{Lk}W\right]\mathrm{Tr}\left[\rho^{\otimes 2}W\right]\right)\\&\quad-\frac{1}{d\,(d^2-1)}\left(\mathrm{Tr}\left[Z_{Lk}W\right]\mathrm{Tr}\left[\rho^{\otimes 2}\right]+\mathrm{Tr}\left[Z_{Lk}\right]\mathrm{Tr}\left[\rho^{\otimes 2}W\right]\right),&(10.130)\\&=\frac{1}{d^2-1}\mathrm{Tr}\left[Z_{Lk}W\right]\mathrm{Tr}\left[\rho^2\right]-\frac{1}{d\,(d^2-1)}\mathrm{Tr}\left[Z_{Lk}W\right]\mathrm{Tr},&(10.131)\\I_2&=\frac{1}{d^2-1}\left(\mathrm{Tr}\left[H^{\otimes 2}\right]\mathrm{Tr}\left[Z_{Rk}\right]+\mathrm{Tr}\left[H^{\otimes 2}W\right]\mathrm{Tr}\left[Z_{Rk}W\right]\right)\\&\quad-\frac{1}{d\,(d^2-1)}\left(\mathrm{Tr}\left[H^{\otimes 2}W\right]\mathrm{Tr}\left[Z_{Rk}\right]+\mathrm{Tr}\left[H^{\otimes 2}\right]\mathrm{Tr}\left[Z_{Rk}W\right]\right),&(10.132)\\&=\frac{1}{d^2-1}\left(\mathrm{Tr}\left[H^2\right]\mathrm{Tr}\left[Z_{Rk}W\right]-\frac{1}{d\,(d^2-1)}\mathrm{Tr}\left[H\right]^2\mathrm{Tr}\left[Z_{Rk}W\right]\right).&(10.133)\end{aligned}$$

Where the second parts of the substitution use

$$\mathrm{Tr}\left[\rho^{\otimes 2}W\right] = \mathrm{Tr}\left[\rho^2\right], \tag{10.134}$$

$$\mathrm{Tr}\left[H^{\otimes 2}W\right] = \mathrm{Tr}\left[H^2\right], \tag{10.135}$$

$$\mathrm{Tr}\left[Z_{Lk}\right] = \mathrm{Tr}\left[Z_{Rk}\right] = 0, \tag{10.136}$$

$$\mathrm{Tr}\left[A^{\otimes 2}\right] = \mathrm{Tr}\left[A\right]^2, \tag{10.137}$$

$$\mathrm{Tr}\left[\rho\right] = 1. \tag{10.138}$$

Putting this back together with Equation 10.122 and rearranging using the triangle inequality, we get,

$$\begin{aligned}
&|\ \mathrm{Var}\left[\partial_k C - \mathrm{Var}_{L,R}\left[\partial_k C\right]\right]| \leq \\
&\frac{1}{d^2-1}\Bigg(|\mathrm{Tr}\left[Z_{Lk}W\right]| \qquad \left(\mathrm{Tr}\left[\rho^2\right] - \frac{1}{d}\right) + |\mathrm{Tr}\left[Z_{Rk}W\right]| \left(\mathrm{Tr}\left[H^2\right] - \frac{1}{d}\mathrm{Tr}\left[H\right]^2\right)\Bigg) \\
&+ \left|\mathrm{Tr}\left[\mathcal{A}_R(\rho^{\otimes 2})\right]\right|, \qquad (10.139) \\
&\qquad\qquad \leq \\
&\frac{d}{(d^2-1)}\Bigg(||Z_{Lk}||_2 \qquad \left(||\rho||_2^2 - \frac{1}{d}\right) + ||Z_{Rk}||_2^2 \left(||H||_2^2 - \mathrm{Tr}\left[H\right]^2\right)\Bigg) \\
&+ \left|\left|\mathcal{A}_R(\rho^{\otimes 2})\right|\right|_2 ||Z_{Lk}||_2, \qquad (10.140)
\end{aligned}$$

using the Cauchy-Schwarz inequality and $||W||_2 = d$ in the second line.

We can expand $||Z_{xk}||_2$ with the triangle inequality:

$$||Z_{xk}||_2 \leq \left|\left|V_k^{\otimes 2}\right|\right|_2 ||\mathcal{A}_x||_2 + ||\mathcal{A}_x||_2 \left|\left|V_k^{\otimes 2}\right|\right|_2 + 2\left|\left|(V_k \otimes \mathbb{1})\,\mathcal{A}_x\,(\mathbb{1} \otimes V_k)\,V_k^{\otimes 2}\right|\right|_2 \tag{10.141}$$

$$\leq 4\,||\mathcal{A}_x||_2\,. \tag{10.142}$$

This allows us to make the final substitution, relating the variance of the gradient

when the circuit to the left and right of the parameter is close to a 2-design:

$$
\begin{aligned}
&| \operatorname{Var}[\partial_k C - \operatorname{Var}_{L,R}[\partial_k C]]| \leq \\
&4 \left|\left|\mathcal{A}_R(\rho^{\otimes 2})\right|\right|_2 \left|\left|\mathcal{A}_L(H^{\otimes 2})\right|\right|_2 \\
&+ \frac{4d}{d^2-1}\left(\left|\left|\mathcal{A}_R(\rho^{\otimes 2})\right|\right|_2 \left(||H||_2^2 - \frac{1}{d}\operatorname{Tr}[H]^2 \right) + \left|\left|\mathcal{A}_L(H^{\otimes 2})\right|\right|_2 \left(||\rho||_2^2 - \frac{1}{d} \right) \right).
\end{aligned}
\tag{10.143}
$$

We now have three bounds, Equations 10.4.5, and 10.143, which bound the total variance in the gradient of the cost function by the expressibility of the ansatz. Each of the equations are valid at all points in the circuit, so the variance is always bounded by whichever is lowest, i.e. if towards the end of the circuit $\operatorname{Var}_L[\partial_k C]$ will be lowest, as $\mathbb{U}_L$ is closer to a 2-design, so the overall variance is bounded by Equation 10.4.5.

One important note is that the bounds on the variance formulated here only provide an upper bound, with no lower bound being possible; the conclusion being that a more expressive ansatz will result in a flatter cost landscape, but there is no guarantee that reducing the expressibility of an ansatz results in steeper gradients. In Chapter 12 we will develop some numerics to support this argument, and to investigate the variance in the environment of a changing ansatz.

Many of the algorithms discussed in this chapter deal with one aspect of NISQ computing, that of small qubit numbers and coherence times that do not allow us to use fault tolerant techniques, but we have not really discussed the other aspect, noise in the device. In Part IV and Section 13 we will discuss algorithmic techniques for reducing the impact of noise on a quantum computation, techniques that can usually be applied to both quantum chemistry and quantum machine learning problems, due to the similarities of training a parameterised circuit to minimise some objective function.

Chapter 11

Dynamical Mean Field Theory

In the previous chapters of this section, most of the quantum chemistry discussed involves finding the ground states of molecules. Here we will introduce a new system that NISQ computers can study, those modelled by Dynamical Mean Field Theory (DMFT). We will then show how a small DMFT instance was solved using an ion trap quantum device. Parts of this chapter are based upon work in [246], where this author contributed to the constructiuon of the Hamiltonian *in sillico* and simulation of VQE on quantum circuits. In Chapter 12 we will show how the ADAPT algorithm was used to solve bigger instances of DMFT systems.

11.1 Preliminaries and Algorithm Outline

We will present some history linking the Fermi-Hubbard model, an important model in solid state physics [247]–[252] and the Single-Impurity Anderson Model (SIAM). We will then show the outline of an algorithm for DMFT, but in this section we will only show the classical parts of this algorithm.

11.1.1 Fermi-Hubbard Model

The Fermi-Hubbard model [253] (also independently [254] by Ref.s [255], [256]) describes the interactions of fermions (hence the Fermi- prefix, hereafter we will only consider electrons) on a lattice, and is known as a tight binding model, as it assumes the electrons are distributed close to atomic cores. It was originally used to study the electrons in solid state systems transitioning between conducting and insulating systems, since then it is possible that high-temperature superconductivity

can be observed in the Hubbard model, [257], [258], but this is not considered by all [259], [260] and still an area of active research.

The Hubbard model describes the electrons on a lattice with two terms in a Hamiltonian, the hopping term describing how electrons hop between sites, and the on-site term which describes their repulsion whilst on the same site,

$$H = -\sum_{i,j}\sum_{\sigma\in\{\uparrow,\downarrow\}} t_{ij}a^{\dagger}_{i\sigma}a_{j\sigma} + U\sum_{i} a^{\dagger}_{i\uparrow}a^{\dagger}_{i\downarrow}a_{i\downarrow}a_{i\uparrow}. \tag{11.1}$$

We will restrict ourselves to considering a regular 2-dimensional lattice of sites, where only nearest-neighbour interactions are allowed $t_{ij} = t^{*}_{ji} = t\delta_{|i-j|,1}$. Despite being a simple model to write and describe, the Hubbard model is hard to solve, and can be used to describe a huge amount of different many-body systems.

The Fermi-Hubbard model has been considered as a candidate for quantum computer solutions, both in the NISQ era [249], [251], [261], and in fault-tolerant algorithms [262], [263], but in this thesis we will calculate using DMFT, which requires us to introduce the Single Impurity Anderson Model (SIAM).

11.1.2 Single Impurity Anderson Model

The Hubbard model in infinite dimensions can be mapped onto the single-impurity Anderson model [264], [265], so whilst the Anderson model is not a direct extension of the Hubbard model, we will now begin to discuss the Anderson model. The Anderson impurity model [266] was first formulated to describe magnetic impurities embedded in metals. In experiment [267] a resistivity minimum can be observed, where lowering temperatures no longer decreases resistivity but causes it to increase, this was known as the Kondo effect [268] and was not captured in the theoretical models in the first half of the 20^{th} century. It can be shown that the Anderson model replicates the features of Kondo's model, which in the simplest form utilises only a single impurity, the SIAM. In the SIAM we have a single impurity which is coupled to a bath of sites that can exchange electrons with the impurity, and we write the

Hamiltonian as a sum of three parts: the bath, the impurity and their interaction,

$$H = H_{\text{bath}} + H_{\text{imp}} + H_{\text{mix}}. \tag{11.2}$$

We will write the orbitals associated with the impurity with Greek subscripts, and the bath with Latin, we will also enumerate each spin orbital, as opposed to referring to the spin individually, i.e. $(1,\uparrow) \to \alpha = 1, (1,\downarrow) \to \alpha = 2$, until $(N_{\text{imp}},\downarrow) \to \alpha = 2N_{\text{imp}}$. The parts of Equation 11.2 are given by

$$H_{\text{imp}} = (\varepsilon_\alpha - \mu)\, a^\dagger_\alpha a_\alpha + \sum_{\alpha\beta\gamma\delta} U_{\alpha\beta\gamma\delta} a^\dagger_\alpha a^\dagger_\beta a_\gamma a_\delta, \tag{11.3}$$

$$H_{\text{mix}} = \sum_{\alpha i} \left(V_{\alpha i} a^\dagger_\alpha a_i + V^*_{\alpha i} a^\dagger_i a_\alpha \right), \tag{11.4}$$

$$H_{\text{bath}} = \sum_i \varepsilon_i a^\dagger_i a_i, \tag{11.5}$$

where $\varepsilon_{\alpha(i)}$ is the on-site energy of the impurity (bath), μ is the chemical potential, $U_{\alpha\beta\gamma\delta}$ are the electron interaction energies, and $V_{\alpha i}$ are the hopping matrix elements between the impurity and bath.

In what follows, we want to calculate the values for ε_i and $V_{\alpha i}$, as we know $\varepsilon_\alpha, \mu, U_{\alpha\beta\gamma\delta}$ from the underlying material.

11.1.3 Green's Function

The Green's function was developed in the 19$^{\text{th}}$ century by George Green in a study of electricity and magnetism [269]. It is a proposed solution to the ordinary differential equation, $L(x)G(s,x) = \delta(s-x)$ where $\delta(x)$ is the Dirac delta. If we know the solution to this equation at one point, and $L(x)$ is linear, we can superpose the solutions to find the general solution $u(x) = \int f(y)G(x,y)dy$ for the general form $L(x)u(x) = f(x)$.

In the context of the SIAM we define Green's functions for ω, the energy expressed as imaginary, or Matsubara [270] , frequencies [271] . We assume that the Green's function is a matrix, and we have a constant density of states. We also assume the band width is larger than the other energy scales in the problem, so that

the main physics is as a result of the interaction between U and the hybridisation function, $\Delta_\alpha(\omega)$, which is given by

$$\Delta_\alpha(\omega) = \sum_i \frac{|V_{\alpha i}|^2}{\omega + i\delta - \varepsilon_i}, \tag{11.6}$$

with δ here an infinitesimally small positive number.

11.1.4 DMFT loop

So in DMFT the algorithm is to achieve self-consistency between the retarded Green's functions for the original lattice model $G_{\text{lat}}(\omega)$ and the problem with a non-interacting impurity, $G_{0,\alpha}(\omega)$. We begin with an initial guess for the Hamiltonian parameters, calculate the difference $\left|G^{-1}_{\text{lat}}(\omega) - G^{-1}_{0,\alpha}(\omega)\right|^2$ and minimise it by updating $\varepsilon_\alpha, \varepsilon_i, V_{\alpha i}$. The minimisation can only be zero if an infinite number of bath sites is used, so we are content to just minimise the function [246].

Firstly, we present the Green's function for an interacting impurity at zero temperature,

$$G_\alpha(\omega) = \left(\omega + i\delta - \varepsilon_\alpha + \mu - \Delta_\alpha(\omega) - \Sigma_\alpha(\omega)\right)^{-1}, \tag{11.7}$$

where we have introduced the many-body self-energy, $\Sigma_\alpha(\omega)$ which accounts for all of the modifications to the non-interacting Green's function, $G_{0,\alpha}(\omega)$,

$$\Sigma_\alpha(\omega) = G^{-1}_{0,\alpha}(\omega) - G^{-1}_\alpha(\omega), \tag{11.8}$$

i.e. $G_{0,\alpha} = \left(\omega + i\delta - \varepsilon_\alpha + \mu - \Delta_\alpha(\omega)\right)^{-1}$. The many-body self-energy includes the corrections induced by $\sum_{\alpha\beta\gamma\delta} U_{\alpha\beta\gamma\delta} a^\dagger_\alpha a^\dagger_\beta a_\gamma a_\delta$, in H_{mix}. The final quantity to name is the non-interacting Green's function on the lattice,

$$G_{0,\text{lat}}(\omega) = \left(\omega + i\delta + \mu + \varepsilon_i - \Delta_{\text{lat}}(\omega)\right)^{-1}. \tag{11.9}$$

However, we do not yet have an independent function for $G_\alpha(\omega)$, so we are not yet ready to minimise. As the Green's function is a correlation function [272] we

can use its Lehman representation [273], [274],

$$G_\alpha(\omega) = \int_{-\infty}^{\infty} dx \frac{\rho_0(x)}{\omega + i\delta - \omega_n} \tag{11.10}$$

To simplify, we can assume that off-diagonal terms in $G_\alpha(\omega)$ can be neglected[1],

$$G_\alpha(\omega) = \sum_{n=1}^{M_{N_0-1}} \frac{\lambda_{\mathrm{h},\alpha,n}}{\omega + i\delta - \omega_{\mathrm{h},n}} + \sum_{n=1}^{M_{N_0+1}} \frac{\lambda_{\mathrm{p},\alpha,n}}{\omega + i\delta - \omega_{\mathrm{p},n}}. \tag{11.11}$$

The first sum runs over all states with one electron removed, the hole states; and the second sum is over all states with an electron added, the particle states. $\lambda_{\mathrm{p(h)},\alpha,n}$ are between zero and one, and satisfy

$$\Sigma_n \left(\lambda_{\mathrm{p},\alpha,n} + \lambda_{\mathrm{h},\alpha,n}\right) = 1. \tag{11.12}$$

The particle and hole frequencies are given by

$$\omega_{\mathrm{p},n} = E_{N_0+1,n} - E_{N_0,n} \tag{11.13}$$

$$\omega_{\mathrm{h},n} = E_{N_0,0} - E_{N_0-1,n} \tag{11.14}$$

and the matrix elements λ are calculated by

$$\lambda_{\mathrm{p},\alpha,n} = \left|\langle \psi_{N_0+1,n}| \sigma_\alpha^- |\psi_{N_0,0}\rangle\right|^2, \tag{11.15}$$

$$\lambda_{\mathrm{h},\alpha,n} = \left|\langle \psi_{N_0-1,n}| \sigma_\alpha^+ |\psi_{N_0,0}\rangle\right|^2. \tag{11.16}$$

Here N gives the number of electrons in the state, and $n = \{0 \ldots M_N\}$ indexes the eigenstates with N electrons. $|\psi_{N,0}\rangle$ gives the ground state for N electrons, $|\psi_{N,1}\rangle$ the first excited state, etc. N_0 gives the number of electrons in the overall ground state, so $|N_0, 0\rangle = |\psi_0\rangle$ is the overall ground state. The energies of states, $E_{N,n}$ are labelled similarly.

[1] We can still write the Green's functions as dense matrices, but we do not here for ease of exposition.

11.2 Quantum Algorithm

Up to this point, everything discussed can (and has [272]) be calculated on a classical computer. In this section we introduce the part of the algorithm that could benefit from calculations on a quantum device.

There are other proposals for calculating the Green's function using a quantum device, some which perform a Fourier transform on the real Green's function [275], [276]. As this requires very small time steps at large interaction strengths, the noise in NISQ devices it too high for accurate computation with this method. Another method which uses effectively averaged integrated quantities was proposed in [277]. Also note that at a similar time to the publication of [246] a method which also utilises VQE was demonstrated [278]. We propose a method utilising VQE as it is suitable for smaller NISQ devices, and has shown to be more noise-resilient [50], [279]–[281].

11.2.1 Quantities Calculated on the Quantum Device

As discussed above, we must calculate $\omega_{\mathrm{p(h)},n}$ and $\lambda_{\mathrm{p(h)},\alpha,n}$ to compute the impurity Green's function. We use the Jordan-Wigner transformation [198], Section 9.1.1 to represent the fermionic operators on the quantum device, with the impurity electrons appearing first in the index. For the impurity and bath ladder operators, we write

$$\sigma_\alpha^\pm = \left(\prod_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\frac{1}{2}\left(\sigma_\alpha^x \pm i\sigma_\alpha^y\right), \tag{11.17}$$

$$\sigma_i^\pm = \left(\Pi_{\beta=1}^{N_{\mathrm{imp}}}\sigma_\beta^z\right)\left(\Pi_{j=1}^{i-1}\right)\frac{1}{2}\left(\sigma_i^x \pm i\sigma_i^y\right). \tag{11.18}$$

With the Jordan-Wigner representation, for $\lambda_{\mathrm{p(h)},\alpha,n}$we begin with Equation 11.15, but as $\sigma_\alpha^{-(+)}$ creates (destroys) an electron in α we have

$$|\psi_{N',m}\rangle\,\sigma_\alpha^+\,\langle\psi_{N,n}| = \delta_{N',N-1}\,|\psi_{N',m}\rangle\,\sigma_\alpha^+\,\langle\psi_{N,n}| \tag{11.19}$$

$$|\psi_{N',m}\rangle\,\sigma_\alpha^-\,\langle\psi_{N,n}| = \delta_{N',N+1}\,|\psi_{N',m}\rangle\,\sigma_\alpha^-\,\langle\psi_{N,n}| \tag{11.20}$$

Rewriting the Jordan-Wigner transform gives us

$$\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x = \sigma_\alpha^+ + \sigma_\alpha^-. \tag{11.21}$$

We can substitute this into Equation 11.19 to get

$$\begin{aligned}\delta_{N',N-1}\langle\psi_{N-1,m}|\,\sigma_\alpha^+\,|\psi_{N,m}\rangle &= -\delta_{N',N+1}\langle\psi_{N+1,m}|\,\sigma_\alpha^-\,|\psi_{N,m}\rangle \\ &\quad + \langle\psi_{N',m}|\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x\,|\psi_{N,n}\rangle \qquad (11.22)\\ \langle\psi_{N-1,m}|\,\sigma_\alpha^+\,|\psi_{N,m}\rangle &= \langle\psi_{N-1,m}|\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x\,|\psi_{N,n}\rangle\,, \qquad (11.23)\\ \langle\psi_{N+1,m}|\,\sigma_\alpha^-\,|\psi_{N,m}\rangle &= \langle\psi_{N+1,m}|\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x\,|\psi_{N,n}\rangle\,. \qquad (11.24)\end{aligned}$$

This means that under the Jordan-Wigner transformation we do not have to increase the number of separate terms calculated, and the matrix terms $\lambda_{\mathrm{p(h)},\alpha,n}$ can be written

$$\lambda_{\mathrm{p},\alpha,n} = \left|\langle\psi_{N_0+1,n}|\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x\,|\psi_0\rangle\right|^2, \tag{11.25}$$

$$\lambda_{\mathrm{h},\alpha,n} = \left|\langle\psi_{N_0-1,n}|\left(\Pi_{\beta=1}^{\alpha-1}\sigma_\beta^z\right)\sigma_\alpha^x\,|\psi_0\rangle\right|^2. \tag{11.26}$$

11.2.2 Calculating the Ground State, excited states, and $N_0 \pm 1$ states

In what follows we will use $U_{N,n}$ to represent the unitary operator found by VQE that prepares $|\psi_{N,n}\rangle$ from the all $|0\rangle$ state; $|\psi_{N,n}\rangle = U_{N,n}\,|0\rangle$. The simplest operator to find is U_0, which prepares the ground state, and can be found the same as any other VQE ground state, discussed in Chapter 10. The next operator to find is that the one where $N = N_0 \pm 1$. For this we modify the original Hamiltonian to include a penalty term to enforce the target electron number, N_{targ},

$$\tilde{H} = H + \beta\left(\hat{N} - N_{\mathrm{targ}}\right)^2, \tag{11.27}$$

where $\hat{N} = \sum_\alpha \hat{n}_\alpha + \sum_i \hat{n}_i$ is the total number operator, and β is a large fixed number [282, ch.17]. The penalty term approach was introduced in [283], and is used

here due to its simplicity and robustness to noise. Here we follow the subspace expansion method [284] for calculating the excited states, but there are other possible methods [285], [286].

In the subspace expansion method, we rely upon the fact that the ground and excited states are mutually orthogonal, and are gapped at lower levels. To find the $k-th$ eigenstate we minimise the VQE problem, imposing an orthogonality condition between all of the ground and excited sates up to $k-1$. The subspace expansion method is:

1. Choose a set of input states $\left\{|\psi_j\rangle\right\}_{j=0}^{k}$ that are orthogonal, $\langle\psi_i||\psi_j\rangle = \delta_{ij}$. This imposes the orthogonality condition.

2. Construct the first, general ansatz circuit, $U(\theta)$.

3. Minimise the first loss function, $\mathcal{L}_1(\theta) = \sum_{j=0}^{k}\langle\psi_j|U^\dagger(\theta)HU((\theta))|\psi_j\rangle$. The optimal parameters found are denoted θ^*.

4. Construct a second ansatz circuit, $V((\phi))$ that only acts on the space spanned by $\left\{|\psi_j\rangle\right\}_{j=0}^{k}$.

5. Choose any index $s \in j$ and maximise the second cost function, $\mathcal{L}_2(\phi) = \langle\psi_s|V^\dagger(\phi)U^\dagger(\theta^*)HU(\theta)V(\phi)|\psi_s\rangle$.

In the subspace expansion method, the first minimization, over $U(\theta)$, finds the unitary which maps us from the space spanned by $\left\{|\psi_j\rangle\right\}_{j=0}^{k}$ to the one spanned by k eigenvectors of the Hamiltonian, $\left\{|E_j\rangle\right\}_{j=0}^{k}$. We note that the input states $\left\{\psi_j\right\}_{j=0}^{k}$ can easily be the computational basis states, as they are easy to prepare. Once $U(\theta)$ circuit has moved us into the subspace spanned by $\left\{E_j\right\}_{j=0}^{k}$, we use the restricted ansatz $V(\phi)$ to search the subspace. By maximising over ϕ we find the $k-$th excited state.

As a toy example, take a VQE problem over 4 qubits, which is the same as the later DMFT problem. For the $U(\theta)$ circuit we can propose any ansatz, for example a hardware efficient ansatz, which easily permits the computational basis states as orthogonal states. If we wish to find the second excited state, we can choose the first

three computational basis states as our input, $\left\{|\psi\rangle_j\right\} = \{|0000\rangle, |0001\rangle, |0010\rangle\}$. We can do the VQE minimisation over all three states simultaneously to get $U(\theta^*)$. Now, we must propose an ansatz $V(\phi)$ that spans over these three states only, this is where we have to do more careful ansatz design. A general Hardware Efficient Ansatz will not do, e.g. $R_y(\phi)$ gates on each qubit allows for the $|0011\rangle$ state. Instead for an odd number of basis states, we must find some way to restrict the output, e.g the circuit in Figure 11.1.

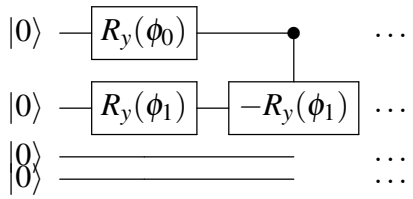


Figure 11.1: A circuit parameterised by ϕ that only explores the $|0000\rangle, |0001\rangle, |0010\rangle$ subspace.

Also presented in [284] are methods of finding excited states which require only a single optimisation procedure, but a more complicated cost function, and perhaps longer overall time. To find the $k-$th excited state, we again choose $\left\{|\psi\rangle_j\right\}_{j=0}^{k}$ orthogonal basis states, but we now minimise the cost function,

$$\mathcal{L}_w(\theta) = w\langle\psi_k|U^\dagger(\theta)HU(\theta)|\psi_k\rangle + \sum_{j=0}^{k-1}\langle\psi_j|U^\dagger(\theta)HU(\theta)|\psi_j\rangle. \tag{11.28}$$

The final quantity to calculate is $\lambda_{\mathrm{p(h)},\alpha,n}$, which we do by using the definition of $U_{N,n}$. For the particle case we have

$$\lambda_{\mathrm{p},\alpha,n} = \left|\langle 0|U^\dagger_{N_0+1,n}\left(\Pi_{\beta=1}^{\alpha-1}\sigma^z_\beta\right)\sigma^x_\alpha U_{N_0,0}|0\rangle\right|^2, \tag{11.29}$$

which can be similarly derived for the hole case. We already need to find the circuits for $U^\dagger_{N_0+1,n}$ and $U_{N_0,0}$ in the calculation of energy, so we only now need to evaluate $U^\dagger_{N_0+1,n}\left(\Pi_{\beta=1}^{\alpha-1}\sigma^z_\beta\right)\sigma^x_\alpha U_{N_0,0}$ and calculate the probability of measuring the all 0 state.

11.2.3 Regularisation

With everything covered so far, it is possible for us to perform the DMFT self-consistency loop. However, as we are discussing performing the calculations on a noisy device, there will be deviations from the exact values. In $\Sigma(\omega)$ these deviations lead to unphysical poles, as $\Sigma_\alpha(\omega) = G_{0,\alpha}^{-1}(\omega) - G_\alpha^{-1}(\omega)$, which cancels out the divergences around $G_\alpha(\omega) = 0$. When the divergences do not cancel , we are left with a DMFT loop that does not converge. $G_{\alpha,0}(\omega) = 0$ when there is a pole in $\Delta_\alpha(\omega) = \sum_i \frac{|V_{\alpha i}|^2}{\omega + i\delta - \varepsilon_i}$, i.e. ω is in the set $\{\varepsilon_i\}$. Cancellation of divergences therefore requires

$$G_\alpha(\varepsilon_i) = 0 \text{ and } \frac{dG_\alpha(\omega)}{d\omega} = \frac{dG_{0,\alpha}(\omega)}{d\omega}\bigg|_{\omega=\varepsilon_i}. \tag{11.30}$$

Combining this with Equation 11.11 gives us the rules

$$\sum_{n=0}^{M_{N_0-1}} \frac{\lambda_{\mathrm{h},\alpha,n}}{\varepsilon_i - \omega_{\mathrm{h},n}} + \sum_{n=0}^{M_{N_0+1}} \frac{\lambda_{\mathrm{p},\alpha,n}}{\varepsilon_i - \omega_{\mathrm{p},n}} = 0, \tag{11.31}$$

$$\sum_{n=0}^{M_{N_0-1}} \frac{\lambda_{\mathrm{h},\alpha,n}}{\left(\varepsilon_i - \omega_{\mathrm{h},n}\right)^2} + \sum_{n=0}^{M_{N_0+1}} \frac{\lambda_{\mathrm{p},\alpha,n}}{\left(\varepsilon_i - \omega_{\mathrm{p},n}\right)^2} = \frac{1}{V_i^2}. \tag{11.32}$$

We therefore perform a constrained optimisation procedure, where we calculate $\lambda_{\mathrm{p(h)},\alpha,n}, \omega_{\mathrm{p(h)},n}$ on the device and then change $\lambda_{\mathrm{p(h)},\alpha,n}$ as little as possible to satisfy the sum rules above. We only vary $\lambda_{\mathrm{p}(h),\alpha,n}$, keeping $\omega_{\mathrm{p(h)},n}$ fixed as the constrained minimisation procedure can be costly as the system size increases, depending on $\mathcal{O}(\omega^2)$, but only linearly on λ; we also expect that there is a lower error in $\omega_{\mathrm{p(h)},n}$ as it is calculated with a shorter circuit.

11.3 Two Site Example

So far the DMFT algorithm laid out can be applied to a general DMFT system of any size. We will now discuss the concrete example of a two-site DMFT system, with a single impurity and a single bath site. This example is used as it has been solved analytically [272], [275], which gives us good results to benchmark the quantum algorithm, and it is a number of qubits that could be used with an acceptable amount of noise on the hardware available at the time.

The Jordan-Wigner transformed Hamiltonian for the two-site DMFT problem is

$$\hat{H} = \frac{U}{4}\sigma_z^1\sigma_z^3 + \left(\frac{\mu}{2} + \frac{U}{4}\right)\left(\sigma_z^1 + \sigma_z^3\right) - \frac{\varepsilon}{2}\left(\sigma_z^2 + \sigma_z^4\right) + \frac{V}{2}\left(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2 + \sigma_x^4\sigma_x^3 + \sigma_y^3\sigma_y^4\right), \tag{11.33}$$

where we have re-mapped the indices to reduce the number of leading σ^z operators from the Jordan-Wigner transformation, we now order the orbitals based on spin first,

$$1_\uparrow \to q_1$$
$$2_\uparrow \to q_2$$
$$1_\downarrow \to q_3$$
$$2_\downarrow \to q_4$$

This allows us to define S_z, the total z component of the spin,

$$\hat{S}_z = \hat{n}_1 + \hat{n}_2 - \hat{n}_3 - \hat{n}_4, \tag{11.34}$$

$$S_z = \langle \hat{S}_z \rangle. \tag{11.35}$$

The derivations in [272] allow us to define self-consistency relations for the two site DMFT problem,

$$n_{\text{imp}} = n_{\text{lat}} \tag{11.36}$$
$$V_{12}^2 = z. \tag{11.37}$$

This is partly due to H being spin-independent, so $G_3 = G_1, \Sigma_3 = \Sigma_1$, leading to only two bath parameters to minimise, $V_{12} = V$ and ε_2. The occupation of the impurity, n_{imp} is

$$n_{\text{imp}} = \int_{-\infty}^{0} \text{DOS}_{\text{imp}}(\omega)\, d\omega, \tag{11.38}$$

with a similar quantity for the occupation of the bath site

$$n_{\text{lat}} = \int_{-\infty}^{0} \text{DOS}_{\text{lat}}(\omega)\, d\omega. \tag{11.39}$$

The density of states for the lattice and bath sites are given by

$$\text{DOS}_{\text{imp}}(\omega) = -\frac{2}{\pi}\text{Im}\left[G(\omega + i\delta)\right] \tag{11.40}$$

$$\text{DOS}_{\text{lat}}(\omega) = 2\rho_0\left[\omega + \mu - \Sigma(\omega)\right], \tag{11.41}$$

$$\rho_0(x) = \frac{1}{2\pi}\sqrt{4 - x^2}. \tag{11.42}$$

In the second consistency equation, z is the quasi-particle weight, or wavefunction re normalisation factor,

$$z = \left(1 - \frac{d\text{Re}\left[\Sigma(\omega)\right]}{d\omega}\bigg|_{\omega=0}\right)^{-1} = \left(1 - \frac{\text{Im}\left[\Sigma(i\delta)\right]}{\delta}\right)^{-1}. \tag{11.43}$$

11.3.1 Particle Hole Symmetry

We first consider the case where we have particle hole symmetry, i.e. $\omega_{\text{p},n} = -\omega_{\text{h},n}$ and $\lambda_{\text{p},\alpha,n} = \lambda_{\text{h},\alpha,n}$, leaving us free to consider only the particle contributions. We will here use the knowledge that at the ground state $N_0 = 2$ and $S_z = 0$, which requires experiments to deduce, but we require it to present which terms to calculate.

More reductions we can make due to particle hole symmetry are that, by definition, $E_{3,n} = E_{1,n}$; leaving us free to consider $E_{3,n}$. We also have spin symmetry in the Hamiltonian, Equation 11.33, giving

$$E_{3,0} = E_{3,1},\, S_z = -1 \text{ (1 for } N = 1\text{)} \tag{11.44}$$

$$E_{3,2} = E_{3,3},\, S_z = 1 \text{ (-1 for } N = 1\text{)} \tag{11.45}$$

$$\tag{11.46}$$

Considering now the $\lambda_{\text{p},\alpha,n}$ terms, we wish to calculate $\langle\psi_{3,n}|\,\sigma_\alpha^-\,|\psi_{2,0}\rangle$. Firstly,

we see that only for values of $\alpha = 1$ will we get a non-zero matrix element, as only then do the number of electrons match. Secondly, we can set forbidden transitions (between different spin numbers) to zero:

$$\lambda_{\mathrm{p},1,1} = \langle \psi_{3,1} | | \psi_{3,2} \rangle \rightarrow \uparrow\downarrow = 0 \tag{11.47}$$

$$\lambda_{\mathrm{p},1,3} = \langle \psi_{3,3} | | \psi_{3,2} \rangle \rightarrow \uparrow\downarrow = 0 \tag{11.48}$$

$$\lambda_{\mathrm{p},1,3} = \langle \psi_{3,3} | | \psi_{3,2} \rangle \rightarrow \downarrow\downarrow \neq 0 \tag{11.49}$$

$$\lambda_{\mathrm{p},1,2} = \langle \psi_{3,2} | | \psi_{3,2} \rangle \rightarrow \downarrow\downarrow \neq 0. \tag{11.50}$$

Finally, again using particle hole symmetry, and the constraint on $\lambda_{\mathrm{p(h)},\alpha,n}$ in Equation 11.12, $\sum_n \left(\lambda_{\mathrm{p},\alpha,n} + \lambda_{\mathrm{h},\alpha,n}\right) = 1$, we have the constraint

$$\lambda_{\mathrm{p},1,2} = \frac{1}{2} - \lambda_{\mathrm{p},1,0}, \tag{11.51}$$

which leaves us with a single independent parameter, $\lambda_{\mathrm{p},1,0} = \lambda$. Therefore, for the two-site DMFT at particle hole symmetry we have to calculate:

$$E_0$$
$$E_{3,0}$$
$$E_{3,2}$$
$$\lambda_{\mathrm{p},1,0}.$$

In the full DMFT loop we must also apply the regularisation of λ as given by Equation 11.32, which removes λ as a free parameter, simplifying to

$$\lambda = \frac{\omega_{\mathrm{p},0}^2 \left(V^2 - \omega_{\mathrm{p},2}\right)}{2V^2 \left(\omega_{\mathrm{p},0}^2 - \omega_{\mathrm{p},2}^2\right)}. \tag{11.52}$$

The advantage of choosing two-site DMFT at particle hole symmetry, along with a great reduction in the number of terms to calculate, is there is an analytic

solution available [272]. The solution is

$$V = \sqrt{z} = \begin{cases} \sqrt{1 - \left(\frac{U}{6}\right)^2} & \text{for } U < 6 \\ 0 & \text{for } U \geq 6. \end{cases} \tag{11.53}$$

We will evaluate the quantum part of the algorithm at $U = 4$ which gives the analytic value for $V = 0.745356$.

11.3.1.1 Penalty Term

The first approach we describe is the penalty term approach, where we calculate $E_0, E_{N_0+1,n}$ and $\lambda_{\mathrm{p},\alpha,n}$ via the methods described in Section 11.2, using a hardware efficient ansatz, given in Figure 11.2.

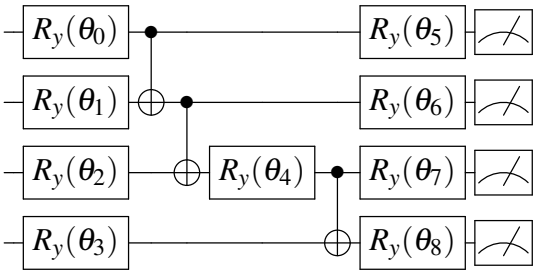


Figure 11.2: The four qubit hardware efficient ansatz used in the penalty term approach to the two-site DMFT problem. All gates are σ_y rotations as the final output is known to be real. When assuming particle hole symmetry we can set $\theta_4 = 0$.

11.3.1.2 Circuit Reduction

To reduce the resource requirements even further, we can design a separate Hamiltonian and circuit for each N and S_z, relying upon projections of the overall Hamiltonian onto a reduced, 2-qubit Hamiltonian. This projection is detailed in Appendix D.

For $N = 1, N = 3$ (which has identical eigenvalues, as noted) we have the Hamiltonians

$$H = \frac{U}{4}\sigma_2^z + V\sigma_2^x \tag{11.54}$$

$$H = -\frac{U}{4}\sigma_2^z + V\sigma_2^x, \tag{11.55}$$

the first qubit does not appear in these Hamiltonians, as the eigenstates are degenerate,

with $S_z = 1$, and $S_z = -1$. The degenerate state is reached by applying a $R_y(\pi)$ gate to the first qubit. The ansatz for these Hamiltonians is given by Figure 11.3a.

For the case with $N = 2$ we have five configurations, one each of $S_z = \pm 2$ and four cases where $S_z = 0$. This is projected to the Hamiltonian

$$H = \frac{U}{4}\sigma_1^z\sigma_2^z + V\left(\sigma_1^x + \sigma_2^x\right), \tag{11.56}$$

with the ansatz for this Hamiltonian given in Figure 11.3b.

(a) The two-qubit circuit reduced ansatz for the $N = 1, 3$ cases, applied to the Hamiltonians in Equation 11.54.

(b) The two-qubit circuit reduced ansatz for the $N = 2$ case, applied to the Hamiltonian in Equation 11.56.

Figure 11.3: The ansatze used in the circuit reduction DMFT loop, where the original 4 qubit Hamiltonian has been projected onto 2 qubits.

11.4 VQE Results

VQE experiments were performed in simulation and on two quantum computing architectures, superconducting qubits via the IBM cloud, and the ion trap device at the University of Maryland. This author calculated the simulated results, using the cirq [171] python package, simulating the effect of statistical sampling using 5000 shots, and removing this component by accessing the relevant parts of the calculated wavefunction (which is impossible on a real quantum device). These results are presented in Table 11.1.

11.4.0.1 Error Mitigation

Noise on the real quantum devices can cause large deviations from the ideal values, leading to incorrect DMFT results. This is possible to see when as we are using a small proof of principle system, but will be impossible at larger system sizes. Error mitigation developed on the smaller system will be useful at larger system sizes.

Firstly, as VQE is a variational algorithm it can account for some calibration and systematic errors in gates (over and under rotations). It is possible (at this system

Table 11.1: Results for the required quantities in the simulated DMFT loop, compared to the exact results. ∞ shots removes statistical sampling error by accessing the components of the wavefunction directly, whilst 5000 shots simulates the effects of sampling noise.

		Four Qubit		Two Qubit	
	Exact	∞ shots	5000 shots	∞ shots	5000 shots
E_0	-1.795	-1.795	-1.804	-1.795	-1.811
$E_{3,0}$	-1.247	-1.232	-1.247	-1.247	-1.279
$E_{3,2}$	1.247	1.260	1.247	1.247	1.244
λ	0.262	0.262	0.288	0.262	0.273

size) to determine the optimal circuit parameters in simulation and input these into the quantum circuit, however it was found on the IBM device that this produced a more erroneous result than running the whole VQE experiment on the device, this is shown in Table 11.2. This suggests that there is some mis-calibration in the implemented rotation gates, which is overcome when running the whole experiment on the device.

Table 11.2: Results for the required quantities in the DMFT loop, as performed on the IBM superconducting device compared to the exact results. In the Optimal θ column the values are for the parameters found in simulation. The VQE column is where the final VQE experiment is performed on the quantum device, and the DMFT + VQE column is that where the whole DMFT loop has been ran in conjunction with the quantum device.

		Four Qubit	Two Qubit		
	Exact	Optimal θ	Optimal θ	VQE	DMFT + VQE
E_0	-1.795	-1.500	-1.700	-1.823	-1.809
$E_{3,0}$	-1.247	-1.111	-1.259	-1.248	-1.245
$E_{3,2}$	1.247	1.025	1.253	1.248	1.244
λ	0.262	0.113	0.210	0.275	0.271

11.4.0.2 Density of States Calculated Using Quantum Computer

One of the advantages of DMFT over DFT (non-interacting electrons) is major features of the density of states (DOS) cannot be captured in DFT. This can be seen in Figure 11.5b, the DOS in momentum space of the lattice site, where the non-interacting calculation shows a single peak, but taking into account electron interactions results in three distinct peaks in the DOS. The same is true of Figure 11.5 and the DOS on the impurity site, where the values of peaks are shifted, and there are

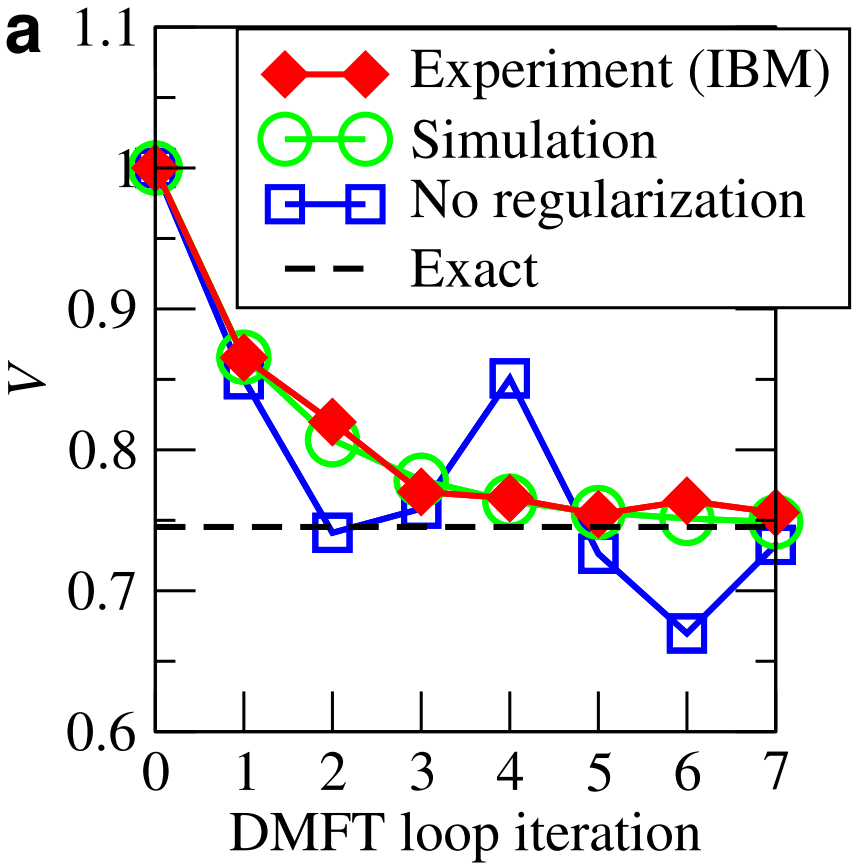


Figure 11.4: Comparison of the simulated VQE experiment with the results obtained from the IBM device with and without regularisation, and the exact results. This demonstrates the power of the regularisation sub-routine to mitigate errors in the noisy device.

more peaks. We also see that results of an experiment on the IBM hardware (using the two qubit circuit in Figure 11.3) agree with the exact results well. The value of $V = 0.755$ calculated in a DMFT loop on the quantum computer agrees well with the exact calculation, $V = 0.745\,356$.

11.4.1 Comparison with Literature

As discussed, a similar paper [278] was released around the same time as the work here was in preparation. The work there uses similar methods to also generate the density of states and obtain a good fit for the Kondo peaks. In [278] they describe a method to calculate the finite temperature Green's function. In the work presented here, we were able to reduce the Hamiltonian to a 2-qubit problem, which allowed us to make calculations on the small quantum devices available whilst this work was

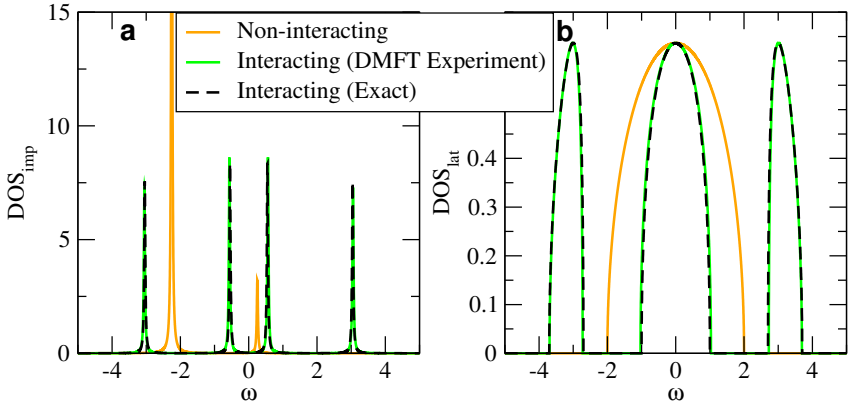


Figure 11.5: Density of States calculated for the (a) impurity site and (b) the lattice site for $U = 4$. Shown is the calculation as computed on the IBM superconducting device, which agrees well with the exact result (dashed). Also shown is the calculation made without interacting electrons (DFT), which does not replicate the features of DOS well.

in preparation, whereas [278] is restricted to numerical simulations only.

Chapter 12

Adaptive Variational Algorithm for Molecular Simulation

As discussed in Chapter 9 finding the ground state of physical systems is a promising area where near term quantum devices will be useful. To achieve quantum advantage the calculations performed on these devices must be of a reasonable size, e.g. on the order of 100 qubits [287]. At these numbers of qubits, and as calculations scale larger, designing the ansatz for a VQE circuit becomes hard. The Ansatz design challenge can be mitigated by the ADAPT [41], and its variant Qubit-ADAPT [42]. Due to the nature of the operators which each uses, we will denote the original ADAPT algorithm 'Fermionic ADAPT', and the variant employing hardware efficient operators 'Qubit ADAPT'. In this chapter we will review each of the ADAPT algorithms, and present the original work here, applying the ADAPT algorithm to the DMFT system discussed in Chapter 11, and our modification to ADAPT which can take into account properties of the device the algorithm is ran on.

12.1 ADAPT-VQE

In the VQE algorithm the experimenter must choose a parameterised ansatz for the quantum device to minimise, with two overall strategies to choose from: a chemically inspired ansatz, or a hardware-efficient ansatz. In the chemically inspired ansatz the circuit ran on the device represents a sub-set of operators that could occur in the system of interest. One choice of operators is the Unitary Coupled Cluster

Singles and Doubles (UCCSD), which have been reviewed in Chapter 10. In UCCSD the operators are the exponentiated, unitary $\left(e^{T-T^\dagger}\right)$ form of the fermionic single (excitation) and double (swap) operators. We can use this scheme to provide a pool of operators to design the ansatz with, but this will not give us an appropriate ordering, which has a non-trivial effect of the performance of the ansatz [41]. The ADAPT algorithm provides a method of designing the ansatz 'on-the-fly', where the ansatz is built up one operator at a time, and the next best operator to use is determined by a calculation on the quantum device. This has the advantage of reducing the total number of operators used over vanilla VQE, as the algorithm will stop when the energy has converged.

In outline, the ADAPT algorithm begins with a pool of operators to choose from, an initial state (e.g. the Hartree-Fock state) and measures the commutator of each operator with the current trial state. The operator with the largest commutator is chosen to be appended to the ansatz, unless all values are below some stopping condition, causing the algorithm to exit. With the new ansatz, a new round of VQE is performed to re-optimise the parameters. This algorithm is formalised in Algorithm 1.

12.1.1 Operator Pools

As has been alluded to, there are two distinct 'flavours' of ADAPT algorithm: Fermionic ADAPT and Qubit-ADAPT. The main difference in these arises from the choice of operators in the operator pool, $\mathcal{P}$ in Algorithm 1. In Fermionic ADAPT they are constructed from the single and double operators acting on the system, transformed into qubit operators via a chosen fermionic to qubit mapping (e.g. Jordan-Wigner).

Many of the fermionic to qubit mappings used incur a large overhead in two-qubit gates for a single operator, due to the requirement that parity is preserved. For example, an operator which acts on the first and last spins requires pairwise entangling operations between all qubits in the calculation. The Qubit-ADAPT algorithm takes inspiration from the hardware efficient ansatz [217], where the ansatz is instead designed from entangling gates native to the hardware device.

Algorithm 1 The original ADAPT algorithm for building the ansatz, A form the pool of operators, O. The difference between the Fermionic and Qubit ADAPT protocols is the form of the operators, in Fermionic they are chemically inspired operators, e.g. UCCSD. In Qubit, they are built from gates native to the device.

Require: H, the Hamiltonian, $O_1, O_2, \ldots O_m$, the operators, $|\psi_{\text{HF}}\rangle$, the Hartree-Fock state, n, the number of qubits on the device, VQE: VQE algorithm as a sub-routine.

E: Energy

$|\psi\rangle \leftarrow |0\rangle^{\otimes n}$ ▷ state of quantum device

$A \leftarrow \mathbb{1}$: The current ansatz

$H \leftarrow H_{\text{qubit}}$ ▷ Transform the Hamiltonian into a qubit representation (using Jordan-Wigner, Brayvi-Kitaev etc.)

$\mathcal{P} \leftarrow \{O_1, O_2, \ldots O_m\}$

if Fermionic ADAPT **then**

  $|\psi\rangle \leftarrow |\psi_{\text{HF}}\rangle$ ▷ Initialise the device into the Hartree-Fock State

end if

while E is not converged **do**

  $G \leftarrow 0$

  N: New operator

  for each $O \in \mathcal{P}$ **do**

    $g \leftarrow |\langle\psi|[O,H]|\psi\rangle|$

    if $g \geq G$ **then**

      $G \leftarrow g$

      $N \leftarrow O$

    end if

  end for

  $A \leftarrow AN$ ▷ Append the new operator to the ansatz

  $E \leftarrow \text{VQE}\left(\langle\psi|AH|\psi\rangle\right)$

end while

As in both the molecular and the DMFT case the Hamiltonians are real, we can first restrict the pool to those with an odd number of Y gates (or even $\times i$) in each operator, and any number of Z gates. We can also show that a set of $2n-1$ operators is complete for the system, meaning we can rotate any *real* state to any other *real* state[1].

Following [42] we begin with an arbitrary three qubit state, and prove we can move to n qubits from there. The arbitrary three qubit state $|\psi\rangle$ can be decomposed

[1] Of course, when the set of states may include imaginary states, the size of the operator pool must grow. This is the case in some quantum machine learning applications, such as in [288].

into

$$|\psi\rangle = |0\rangle_3 |\psi_0\rangle_{21} + |1\rangle_3 |\psi_1\rangle_{21} . \tag{12.1}$$

In general , the norm $\langle\psi_0\rangle\,\psi_0 \neq \langle\psi_1\rangle\,\psi_1$, but we can always use an arbitrary rotation on the third qubit, $e^{i\theta Y_3}$ to transform into

$$|\psi'\rangle = e^{i\theta Y_3} |\psi\rangle\, |0\rangle_3 |\psi'_0\rangle_{21} + |1\rangle_3 |\psi'_1\rangle_{21} , \tag{12.2}$$

with $|\psi'_0\rangle = \cos\theta\,|\psi_0\rangle + \sin|\psi_1\rangle$ and $|\psi'_1\rangle = \cos\theta\,|\psi_1\rangle - \sin\theta\,|\psi_0\rangle$. We can make the norms of these sates equal by choosing a θ such that

$$\tan 2\theta = \frac{\langle\psi_1\rangle\,\psi_1 - \langle\psi_0\rangle\,\psi_0}{2\,\langle\psi_0\rangle\,\psi_1}. \tag{12.3}$$

We can now show that with a state such as Equation 12.2 with $\langle\psi'_1\rangle\,\psi'_1 = \langle\psi'_0\rangle\,\psi'_0 = \frac{1}{2}$ can be transformed via single and two-qubit gates into a state with the third qubit factored out,

$$|\psi'\rangle \rightarrow (|0\rangle + |1\rangle)_3 |\chi\rangle_{21} . \tag{12.4}$$

First, expand $|\psi'\rangle$, collecting the possible terms on the first two qubits:

$$|\psi'\rangle = |00\rangle\,|\psi'_{0a}\rangle + |01\rangle\,|\psi'_{0b}\rangle + |10\rangle\,|\psi'_{1a}\rangle + |11\rangle\,|\psi'_{1b}\rangle . \tag{12.5}$$

Similarly to Equation 12.2, we can apply a Y rotation to each state with $|\psi\rangle_3 = 0(1)$ using the conditional rotations $e^{i\theta_0 \frac{1}{2}(1+Z_3)Y_2}, e^{i\theta_1 \frac{1}{2}(1-Z_3)Y_2}$ to give $\langle\psi'_{0a}\rangle\,\psi'_{0a} = \langle\psi'_{0a}\rangle\,\psi'_{0a} = \langle\psi'_{0b}\rangle\,\psi'_{0b} + \langle\psi'_{1a}\rangle\,\psi'_{1a} = \langle\psi'_{1b}\rangle\,\psi'_{1b} = \frac{1}{4}$.

As we are working with real states, we can think of $|\psi'_{0a}\rangle, |\psi'_{0b}\rangle$ as 2D vectors in the same plane. The rotation $e^{i\phi_1 Z_3 Z_2 Y_1}$ rotates these vectors in opposite directions in the plane, until they coincide at $|\chi_0\rangle$. This allows us to factor out $|\chi_0\rangle$:

$$e^{i\phi_1 Z_3 Z_2 Y_1} |\psi'\rangle = (|00\rangle + |01\rangle)\,|\chi_0\rangle + |10\rangle\,|\psi''_{1a} + |11\rangle\,|\psi''_{1b}\rangle\rangle . \tag{12.6}$$

We now apply a conditional rotation on the second qubit to bring it to the state $|0\rangle$:

$$e^{i\phi_2 \frac{1}{2}(1+Z_3)Y_2} e^{i\phi_1 Z_3 Z_2 Y_1} |\psi'\rangle = \sqrt{2}|00\rangle |\chi_0\rangle + |10\rangle |\psi''_{1a}\rangle + |11\rangle |\psi''_{1b}\rangle . \tag{12.7}$$

By a similar argument the gate $e^{i\phi_3 Z_3 Z_2 Z_1}$ can rotate $|\psi''_{1a}\rangle$ and $|\psi''_{1b}\rangle$ into each other, which does have an effect of $|\chi_0\rangle \to |\chi'_0\rangle$. Again, we use the conditional rotation $e^{i\phi_4 \frac{1}{2}(1-Z_3)Y_1}$ to move the terms in qubit 2 into the $|0\rangle$ state,

$$e^{i\phi_4 \frac{1}{2}(1-Z_3)Y_1} e^{i\phi_3 Z_3 Z_2 Z_1} e^{i\phi_2 \frac{1}{2}(1+Z_3)Y_2} e^{i\phi_1 Z_3 Z_2 Y_1} |\psi'\rangle = \sqrt{2}\left(|00\rangle |\chi'_0\rangle + |10\rangle |\chi_1\rangle\right). \tag{12.8}$$

Again we see that a rotation, $e^{i\phi_5 Z_3 Z_2 Y_1}$ on qubit 1 will rotate the $|\chi'_0\rangle, |\chi_1\rangle$ vectors into the same state, $|\chi\rangle$, giving $\sqrt{2}(|0\rangle + |1\rangle)_3 |0\chi\rangle$. This has all been done with gates from the set $\{iZ_3Z_2Y_1, iZ_3Y_2, iY_2\}$. We call this set $\left\{V^{\text{red}}\right\}_3$, the reduced gate set. Finally, we introduce the single rotation on the third qubit, Y_3 and the conditional operation on qubit 1, $e^{i\phi_6 Z_3 Z_2 Y_1}$, giving us the result we desire on three qubits,

$$e^{i\phi_6 Z_3 Z_2 Y_1} e^{i\frac{\pi}{4}Y_3} e^{i\phi_5 Z_3 Z_2 Y_1} e^{i\phi_4 \frac{1}{2}(1-Z_3)Y_1} e^{i\phi_3 Z_3 Z_2 Z_1} e^{i\phi_2 \frac{1}{2}(1+Z_3)Y_2} e^{i\phi_1 Z_3 Z_2 Y_1} |\psi'\rangle = |000\rangle . \tag{12.9}$$

Therefore, we can map any *real* three qubit state to any other *real* three-qubit state, using the gates $\{iZ_3Z_2Y_1, iZ_3Y_2, iY_2, iY_3\}$, which we will denote $\{V\}_3$

Considering the n-qubit case, if we say now that 3 was the $n+1$ case, we can replace the conditional gates $Z_3Z_2Y_1$ with the reduced set multiplied by Z_{n+1}, $Z_{n+1}\left\{V^{\text{red}}\right\}_n$; and we replace the rotation $\frac{1}{2}(1+Z_3)Y_2$ with $\frac{1}{2}(1+Z_{n+1})Y_n$. Doing these substitutions, we can make the same argument as above, but with $|\psi_{0a}\rangle, |\psi_{1a}\rangle, |\psi_{0b}\rangle, |\psi_{1b}\rangle, |\chi_0\rangle, |\chi_1\rangle$ now $n-1$ qubit states. We only need to replace the last two operations with

$$e^{i\frac{\pi}{4}\frac{1}{2}(1-Z_{n+1})Y_n} e^{i\phi Z_{n+1} Z_n Y_{n-1}} e^{i\frac{\pi}{4}\frac{1}{2}(1-Z_{n+1})Y_n}, \tag{12.10}$$

which rotates us into the state where the n and $n+1$ qubits have the same parity, performs $\phi Z_{n+1} Z_n Y_{n-1}$ and rotates back out.

Following this argument, the set $\{V\}_n$ can transform any *real n* qubit state into any other *real n* qubit state. This set is of size $2n-1$ scaling linearly with the number of qubits, meaning that the calculation of energy gradients in qubit ADAPT does not blow up with the number of qubits[2].

12.2 Device Aware ADAPT

In this work we present a further modification to the ADAPT scheme which takes into account the properties of the device the algorithm is being ran on, which we call Device-Aware ADAPT. As will be clear by now, in the NISQ era we are working with devices which have limited numbers of qubits, noisy qubits, and in some cases limited qubit connectivity [3]. If we take the 27 qubit IBM device shown in Figure 12.1 an operation between qubits 0 and 9 requires many SWAP operations to complete. Even in cases where we have all-to-all connectivity it may be the case that some pairs of qubits are badly linked, i.e. the fidelity of two qubit gates between those is significantly lower than the average.

We may want to sacrifice some ansatz expressibility to avoid gates requiring many SWAPS or badly linked qubits on the actual hardware, which is what we do when introducing the device-aware ADAPT algorithm.

Device-Aware ADAPT is a simple modification to the ADAPT algorithm, where we take some characterisation of the device that the algorithm is running on, including connectivity and noisy linkages, and apply a penalty to each operator in the pool according to how disconnected the qubits are, or if there are any bad links between them. This penalty term is applied to the gradient calculation in line 13 of Algorithm 1, with the goal being that another operator has a smaller initial commutator but fewer bad connections in, it will be used instead.

One question to ask is if we are so concerned with bad linkages, why not remove

[2] A naïeve implementation may place all combinations of even numbered Y Pauli strings on all qubits as the gate set, which increases massively with qubit number.

[3] Limited connectivity is true for superconducting devices [18], [22], and whilst some ion trap devices have true all to all connectivity [289], some devices implement all-to-all connectivity via physical swapping of ions [290]. This may be a higher fidelity operation than the SWAP gate implemented in superconducting technologies, but it does not aid in total execution time, and does not scale well with the number of qubits.

all operators from the pool which contain them? The reason we apply this strategy is that we can avoid bad links where possible, but we are still allowing the algorithm to choose these operators if they have a large enough commutator. The next reasonable concern is, with the availability and power of modern quantum compilers [291]–[293], will they not be able to mitigate these effects? This is not a question we can answer easily analytically, due to the adaptive nature of the algorithm, but we will present numerical results concerning this in Section 12.3.

As discussed in Section 10.4.5 and [215] , the expressibility of an ansatz is linked to the trainability of that ansatz, if it is likely to encounter barren plateaus in training. By introducing a device aware modification to the ADAPT algorithm we are reducing the expressibility of the ansatz by making some operators less likely to appear, so it is natural to ask if this modification has an effect upon the trainability of the ansatz. Barren plateaus in parameterised loss function, $\mathcal{L}_{\rho,H}$ are defined as the average of the partial derivative of all parameters vanishing:

$$\langle \partial_k \mathcal{L} \rangle := \langle \frac{\partial \mathcal{L}_{\rho,H}(\theta)}{\partial \theta_k} \rangle \to 0 \forall k. \tag{12.11}$$

However, if this average contains large fluctuations away from zero then the ansatz may still be trainable, so we use the Chebyshev inequality,

$$\Pr(|\partial_k \mathcal{L}| \geq \delta) \leq \frac{\mathrm{Var}[\partial_k \mathcal{L}]}{\delta^2}, \tag{12.12}$$

to bound the value of the partial derivative with respect to the variance. In the results, we calculate the variance,

$$\mathrm{Var}[\partial_k \mathcal{L}] = \langle (\partial_k \mathcal{L})^2 \rangle - \langle \partial_k \mathcal{L} \rangle^2 \tag{12.13}$$

over all parameters. In the ADAPT scheme, we also have the additional complication that the ansatz changes over time, or with each additional operator. Therefore, we calculate the variance of parameters in an operator at each epoch and show how this changes with the additional of more operators. Here an epoch is defined as beginning

with the addition of a new operator, or when the algorithm returns to line 19. Clearly the parameters in the first operator will have a variance defined for all epochs, but parameters added in later epochs will not have a variance defined at earlier epochs.

12.3 ADAPT Results

The results here are divided into two sections, where the algorithm was performed on molecular systems to allow for comparison of device aware ADAPT with the original ADAPT protocol, and where we have used both the original and device aware algorithms on the DMFT system presented in Chapter 11, due to issues found in scaling up the system to larger qubit numbers.

12.3.1 Molecular Systems

The two systems considered here are the H_4 hydrogen chain and LiH. Molecular data and UCCSD operators were generated using the Openfermion [294] python package. Here we show the device aware and non-device aware versions of ADAPT using the Fermionic, e.g. UCCSD, pool of operators. To compare the device aware and non-device aware strategies, each final circuit was compiled using the $t\,|\text{ket}\rangle$ [295] compiler. The compiler is allowed to move qubits around to choose the strategy with the fewest entangling gates, but must compile any SWAPS into three CNOT gates; this is the same for each version of the circuit. As a first demonstration of the device aware strategy, we do not provide any additional information than the qubit connectivities, applying a penalty of between $0.9 - 0.999$ to each entangling gate between disconnected qubits. The device we model is a sub-section of the IBM [18] 27 qubit system, we are allowed at most 12 qubits from the device. A schematic of these qubits and connections is given in Figure 12.1.

12.3.1.1 Hydrogen Chain

In Figure 12.2 we see the outcomes of both the device aware and non-device aware algorithms in simulating the ground state of the H_4 hydrogen chain for different bond distances. Unfortunately this does not give us a good idea of how the device aware and non device aware algorithms compare, so in Figure 12.3 we compare the average error for each algorithm. As we can see, there is a small accuracy penalty

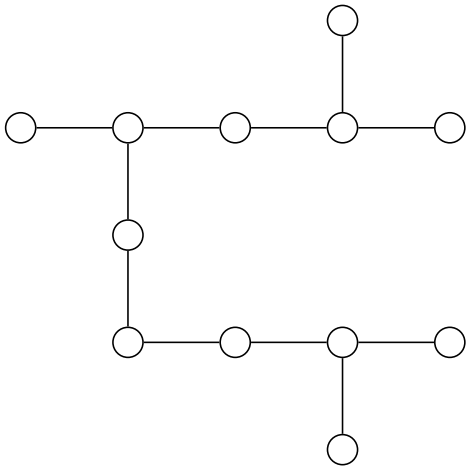


Figure 12.1: The restricted set of qubits that the device aware algorithm targets, and the qubits the t |ket⟩ compiler compiles to.

for the device aware algorithm. In Table 12.1 we see the gains that we can make in choosing the device aware strategy, with a reduction in the average number of two qubit gates (in the compiled circuit).

We also give the values of hyperparameters for the results displayed Table 12.1, The parameters relating to the VQE subroutine are:

1. Optimiser: Which optimisation algorithm is used, see Section 10.3.

2. Learning Rate: The learning rate of the optimiser.

3. For optimisers with additional hyperparameters, such as Adam, the default values are used unless explicitly stated.

The ADAPT specific parameters are:

1. Operator Precision: The value which, if the gradient of all new operators is below this, the algorithm stops.

2. No. Parameters Optimised: In longer circuits, we freeze the first $M - m$ parameters at values found in previous VQE routines, M being the total number of parameters and m this value. This means that the time taken to optimise does not need to increase with circuit length.

Parameter	Value
Op. Precision	5×10^{-5}
Learning Rate	0.01
Optimiser	Adam
No. Parameters Optimised	10
Init Close To I	True
Reinitialise	False
DA Weight	0.9

	Original ADAPT	Device Aware ADAPT
Total Runs	18	21
Avg. Error	1.53×10^{-4}	2.51×10^{-4}
Avg. Circuit Length	148	141
Avg. Overall 2 qubit gates	165	177

Table 12.1: The average accuracy, overall circuit length, and number of compiled two qubit gates comparing the original and device aware ADAPT algorithms, for the hydrogen chain system.

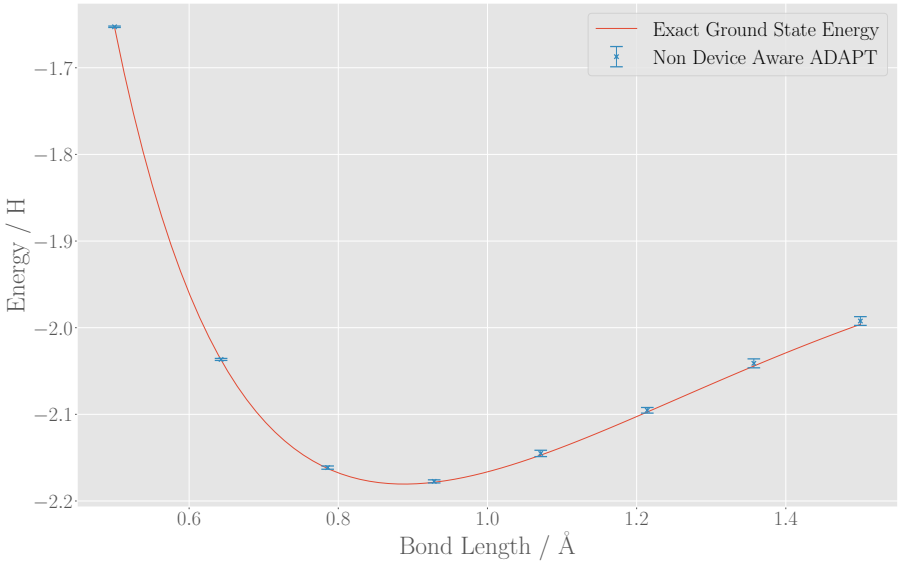
3. Initialise close to identity: If True, when a new operator is added to the circuit we initialise it with a very small parameter. When True, we take advantage of the previous optimisations, when False we have a longer time to optimise, but less chance of becoming stuck in a local minima.

4. Reinitialise: Similar to the previous hyperparameter, where if True we start all unfrozen parameters from some random initial value and perform VQE, if False all parameters retain the values found in the previous epoch. Combining Reinitialise = True, Initialise Close to I = True doesn't make sense as a hyperparameter choice though.

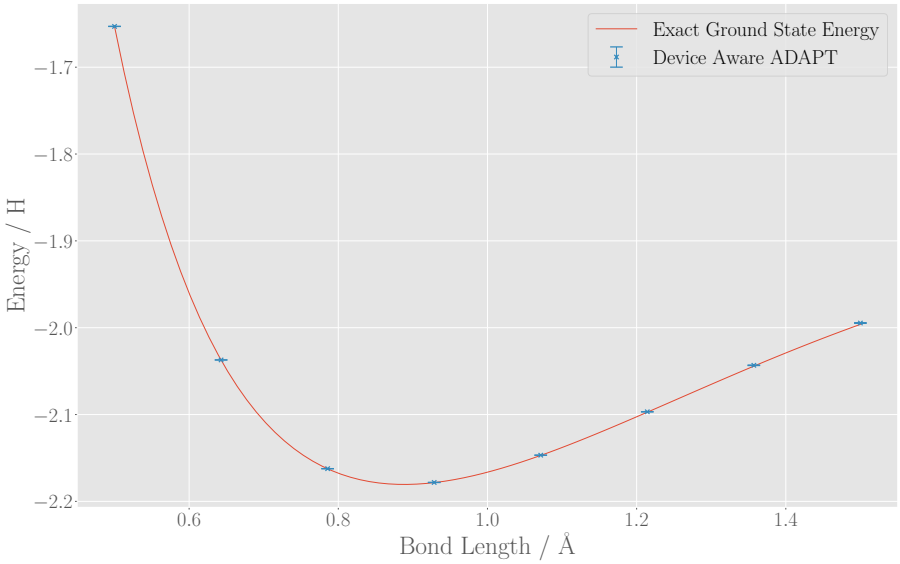
5. DA weight: In Device Aware, each operator gradient is multiplied by this value for each entangling gate between two disconnected qubits in the hardware. There is a method for passing a more detailed noise map into the algorithm, with custom weights for each qubit pair.

12.3.1.2 Lithium Hydride

We now present the results of the same experiments ran on the larger LiH molecule. In Figure 12.4 and in Figure 12.5 we see that the accuracy of the algorithm has



(a)



(b)

Figure 12.2: The full configuration interaction ground state energy of the H_4 chain and the energy found by the *a*) Original ADAPT algorithm, *b*) the Device Aware ADAPT algorithm.

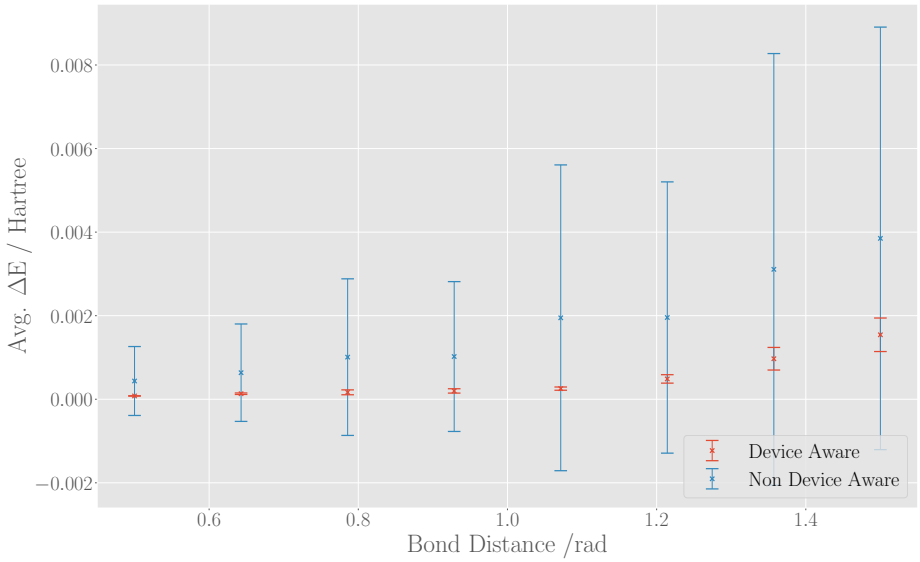
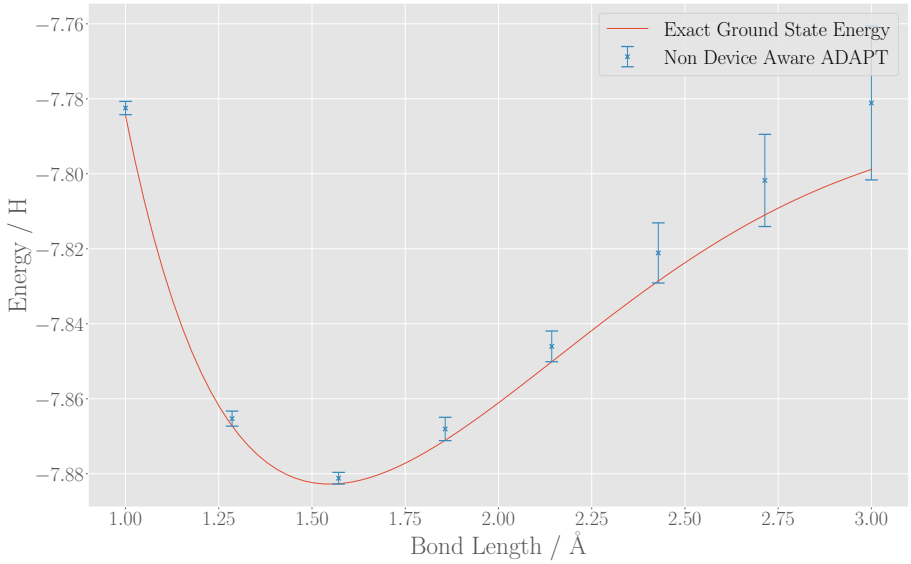


Figure 12.3: The average error from the full configuration interaction energy of the H_4 chain calculated using the device aware and original versions of the ADAPT algorithm.

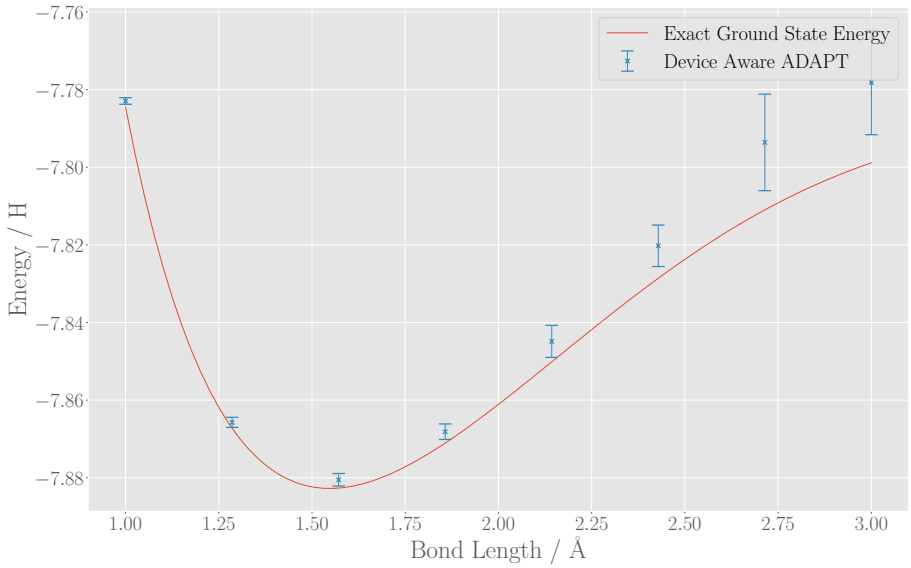
decreased with the system size, although we note that the operator gradient precision remains the same, and we are returned shorter circuits than in the H_4 case. This suggests that the operator gradient absolute values decrease with the number of qubits along with the parameter gradient absolute values, as we saw in Section 10.4. Again we present the average number of two qubit gates in the final compiled circuits in Table 12.2, and we see that there is a reduction in the overall number of two qubit gates.

12.3.2 ADAPT for DMFT

We will now discuss applying the ADAPT algorithm to a new system, the quantum sub-routine of the DMFT algorithm presented in Chapter 11. This system was the motivation for investigating the ADAPT algorithm in this project, as attempts to investigate larger systems than the single impurity, single bath site model had faltered. To increase the accuracy of the SIAM model, we usually add more bath sites. In the quantum algorithm this means increasing the number of qubits by 2 (to store each orbital of the new site). In the work presented in Appendix D we reduced the



(a)



(b)

Figure 12.4: The full configuration interaction ground state energy of the LiH molecule and the energy found by the a) Original ADAPT algorithm, b) Device Aware ADAPT algorithm.

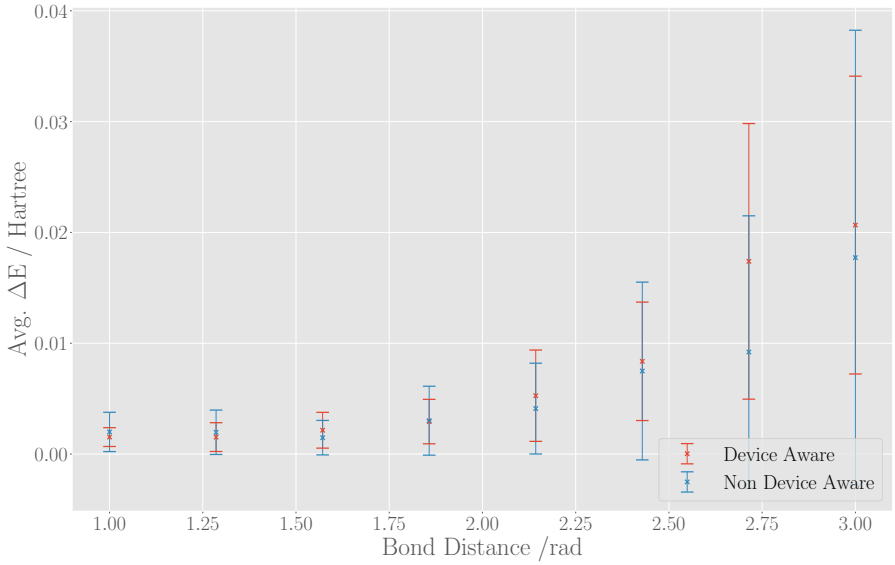


Figure 12.5: The average error from the full configuration interaction energy for the LiH system calculated using the device aware and original versions of the ADAPT algorithm.

Parameter	Value
Op. Precision	5×10^{-5}
Learning Rate	0.01
Optimiser	Adam
No. Parameters Optimised	10
Init Close To I	True
Reinitialise	False
DA Weight	0.9

	Original ADAPT	Device Aware ADAPT
Total Runs	61	55
Avg. Error	5.87×10^{-3}	7.48×10^{-3}
Avg. Circuit Length	65	71
Avg. Overall 2 qubit gates	85	77

Table 12.2: The average accuracy, overall circuit length, and number of compiled two qubit gates comparing the original and device aware ADAPT algorithms, for Lithium Hydride.

number of qubits by projecting the Hamiltonian onto a subspace and using separate Hamiltonians in which we used the assumption of particle-hole symmetry. This is clearly a significant amount of work which becomes unfeasible at larger system sizes and away from particle-hole symmetry. When running VQE on larger system sizes we found that the ansatz we chose became stuck in local minima, this motivated our search for an algorithm that builds its ansatz programmatically. Here we present the results of running the ADAPT algorithm on the DMFT system. As we have discussed the outer DMFT loop in Chapter 11, we will here only concern ourselves with reaching the ground state of the DMFT system, using values for $\varepsilon_i V_{\alpha i}$ from the converged point, where accuracy is most important. The results presented below are for ADAPT running with hardware native operators, Qubit-ADAPT [42].

We will here repeat the Jordan-Wigner transformed Hamiltonian for many sites,

$$H = \sum_{\alpha} U_\alpha - \sigma_\alpha^z \sigma_{\alpha+k}^z \left(\frac{\mu}{2} \frac{-U}{4}\right) \sigma_\alpha^z + \sigma_{\alpha+k}^z + \sum_i \frac{\varepsilon_i}{2} \left(\sigma_i^z + \sigma_{i+N}^z\right) + \sum_{\alpha,i} \sum_{\chi \in \{x,y\}} \left(\sigma_\alpha^\chi \prod_{j=\alpha+1}^{i-1} \sigma_j^z \sigma_i^\chi + \sigma_{\alpha+k}^\chi \prod_{j=\alpha+k+1}^{i+N-1} \sigma_j^z \sigma_{i+N}\right), \tag{12.14}$$

where α is the index over the impurity, i is the index over the bath sites, N represents the number of sites and we recall that we have re-ordered the orbitals from the canonical setting such that $1 \to 1\uparrow, 2 \to 2\uparrow \ldots N \to N\uparrow, N+1 \to 1\downarrow \ldots 2N \to N\downarrow$ to reduce the number of σ^z gates required. The values for constants used at the Green's function self consistency point can be found online in the repository [296].

12.3.2.1 Issues in using ADAPT in the DMFT system

We will briefly discuss here some of the issues encountered in applying the ADAPT algorithm to the DMFT system, removed from issues in using ADAPT for molecular systems, as in the original proposal. Firstly, we have not observed guaranteed convergence with the ADAPT algorithm, unlike in the molecular case. We have observed the algorithm not reaching the gradient stop parameter and continuing to add gates, this causes the energy to diverge and become noisy. Due to this, we introduce a second stopping condition, early stopping [297] where we terminate the algorithm if a VQE

	Proportion Converged	
	Original ADAPT	Device Aware ADAPT
$\{V\}_n$	0	0
$\{G\}_n$	0.81	0.75
$\{V\}_n + \{G\}_n$	1.0	0.9

Table 12.3: Proportion converged of the device aware and original ADAPT algorithms on the 6 qubit DMFT system for different operator pools.

subroutine does not return a lower energy than the previous epoch five times in a row. Secondly, in the Qubit-ADAPT paper [42] , and above in Section 12.1.1 we show that the pool built recursively from $\{V\}_n = \left\{Z_n \{V\}_{n-1}, iY_n, iY_{n-1}\right\}$ can map all real states to all other real states, so is a good candidate for an operator pool. In Ref. [42] they also propose the operator pool $\{G\}_n = \{iZ_0Y_1, iZ_1Y_2 \ldots iZ_{n-1}Y_n, iY_0 \ldots iY_n\}$ which they show can construct $\{V\}_n$ through commutators of $\{G\}_n$, which contains fewer entangling gates in each operation, so should be better implemented in real hardware. Unfortunately, we find that using $\{G\}_n$ as an operator pool for the DMFT system never converges close to the true minimum, and gets stuck in local minima[4]. We have tested other optimisation algorithms, but we find vanilla Stochastic Gradient Descent (SGD) is more reliable, this is probably due to the learning rate decay in Adam getting stuck in local minima. Thankfully, we are able to more reliably reach the global minimum using the $\{V\}_n$ operator pool, and we have even better performance combining both pools. The proportion of experiments that converge, i.e. do not become stuck in a local minima is given in Table 12.3 This gives us a total pool size of $4n-4$, which increases the amount of time measuring the operator gradient at each step, but allows us to achieve convergence in all the original ADAPT experiments.

Finally, we also find that initialising the system in the Hartree-Fock state also leads to the optimiser becoming stuck in a local minima, and we instead initialise the circuit with a set of parameterised R_y rotations (in DMFT there will always be an even number of qubits, so we are not concerned with the factor of i).

[4]This negative result appears no matter the different hyperparameter we search over, there have been many trials of this operator pool on the DMFT system.

Parameter	Value
Op. Precision	1×10^{-6}
Learning Rate	0.01
Optimiser	SGD
No. Parameters Optimised	20
Init Close To I	True
Reinitialise	False
DA Weight	0.75 - 0.9

	Original ADAPT	Device Aware ADAPT
Total Runs	40	80
Avg. Error	1.58×10^{-7}	3.25×10^{-7}
Avg. Circuit Length	43	31
Avg. Overall 2 qubit gates	22	22

Table 12.4: The average accuracy, overall circuit length, and number of compiled two qubit gates comparing the original and device aware ADAPT algorithms, for the 6 qubit DMFT system.

12.3.2.2 Two Site DMFT

For completeness, we first replicate the results given in Chapter 11, for the DMFT system on four qubits, given in Table 12.4. We find good agreement with the ground state found previously.

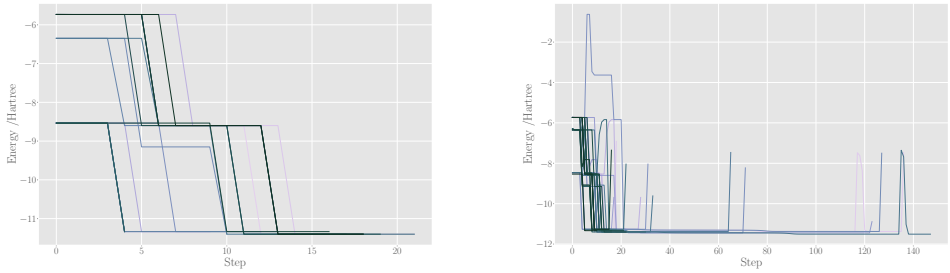
12.3.2.3 Three Site DMFT

In three site DMFT we have a single impurity and two bath sites, which is mapped onto a six qubit Hamiltonian, Equation 12.14.

In Figure 12.6 we have aggregated the histories of many runs, combined with the operator gradient at that point. The operator gradient found descends smoothly in the results that converge, but on those that we must early stop, the value can increase. Table 12.5 compares the average error and compiled two qubit gate counts for the device aware and non-device aware versions of the algorithm.

12.3.2.4 Four Site DMFT

In four site DMFT we have a single impurity and three bath sites. In Figure 12.7 we have aggregated the histories of many runs, combined with the operator gradient at that point. The operator gradient found descends smoothly in the results that converge, but on those that we must early stop, the value can increase. Table 12.6



(a) **(b)**

Figure 12.6: The energy history and operator gradient for the ADAPT algorithm on the six qubit DMFT system. a): Original algorithm, b): Device aware algorithm.

Parameter	Value
Op. Precision	1×10^{-6}
Learning Rate	0.01
Optimiser	SGD
No. Parameters Optimised	20
Init Close To I	True
Reinitialise	False
DA Weight	0.75 - 0.9

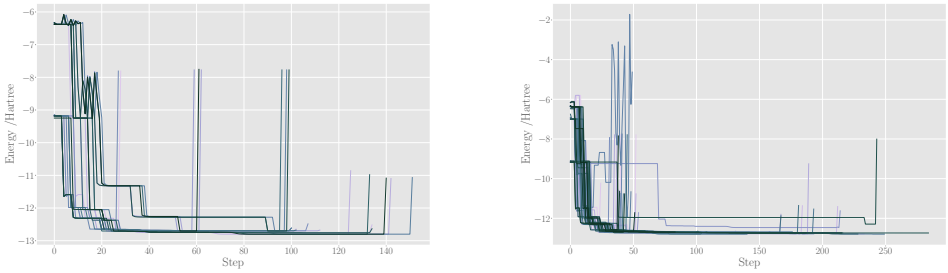
	Original ADAPT	Device Aware ADAPT
Total Runs	14	25
Avg. Error	0.13	0.19
Avg. Circuit Length	25	29
Avg. Overall 2 qubit gates	12	10

Table 12.5: The average accuracy, overall circuit length, and number of compiled two qubit gates comparing the original and device aware ADAPT algorithms, for the 6 qubit DMFT system.

compares the average error and compiled two qubit gate counts for the device aware and non-device aware versions of the algorithm.

12.3.3 Barren Plateaus and Parameter Derivatives

In Section 10.4 we discussed the results from [215], and the link between the expressibility of an ansatz and the incidence of barren plateaus, that a more expressive ansatz has a lower upper bound on the average variance of the gradient for each parameter. We note that this is not necessarily a two-way relationship, that reducing expressibility does not always equate to removing barren plateaus. However, with the introduction of the device aware version of ADAPT we have a natural comparison



(a) **(b)**

Figure 12.7: The energy history and operator gradient for the ADAPT algorithm on the eight qubit DMFT system. a): Original algorithm, b): Device aware algorithm.

Parameter	Value
Op. Precision	1×10^{-6}
Learning Rate	0.01
Optimiser	SGD
No. Parameters Optimised	20
Init Close To I	True
Reinitialise	False
DA Weight	0.75 - 0.9

	Original ADAPT	Device Aware ADAPT
Total Runs	16	29
Avg. Error	0.13	0.14
Avg. Circuit Length	101	93
Avg. Overall 2 qubit gates	46	44

Table 12.6: The average accuracy, overall circuit length, and number of compiled two qubit gates comparing the original and device aware ADAPT algorithms, for the 8 qubit DMFT system.

between a more expressive ansatz given by the original algorithm, and the restricted expressibility given by device aware ADAPT. By de-weighting certain operators in the pool we make it less likely that they show up, we can imagine an adversarial scenario where an operator is never chosen due to the weight of the entangling gates it involves, this makes the final ansatz less expressive by not choosing this operator.

Measuring the variance of a parameter in an ADAPT algorithm presents us with a choice of where to aggregate the data; we could present an overall average for the two algorithms, but we need to make the choice of whether this is in the final circuit, or as the ADAPT outer loop proceeds. We can also aggregate on a parameter by

parameter basis, to learn more about the variance as the algorithm progresses, but we are left with the difficulty of accounting for parameters that only occur in later layers.

The method we have chosen is to record the variance for the parameters in each layer of the circuit, averaged over all the circuits ran. This means that parameters in later layers of the circuits do not have a value for all epochs (where an epoch begins with the addition of a new operator). The results given in this section record the epoch on the x-axis, and the average variance on the y-axis, with a new graph given for each parameter. The scale of the y-axis for each system is shared between the original and device aware versions of the algorithm. Therefore, the graph for the 3rd layer parameters will not contain a value at epochs $0, 1, 2$, but will show the variance in all subsequent layers. We will here give a representative sample of the graphs, choosing more interesting results, and we will give the graphs for all parameters in Appendix E.

12.3.3.1 Hydrogen Chain

In Figure 12.8 we present the variance results for the first, fourth and final layers, where we see that the variance of the cost function is quite low for most parameters, but increases in the middle of the circuit[5]. We also see that there is a marked decrease in the variance of parameters in the device aware algorithm, showing that this reduction in expressibility is *not* accompanied by a decrease in the incidence of barren plateaus.

12.3.3.2 Lithium Hydride

In Figure 12.9 we see similar phenomena as in the Hydrogen chain, although the peak in the layer variance comes toward the end, in the penultimate layer, 7. We also observe that the absolute value of the variance is lower here.

12.3.3.3 Three Site DMFT

Observing the variance of the DMFT systems results, we see that the circuits found in the DMFT system are much longer. Again we show the first, last, and peak of

[5]This average is not affected by the freezing of parameters, once a parameter is frozen we no longer count its contributions to the variance.

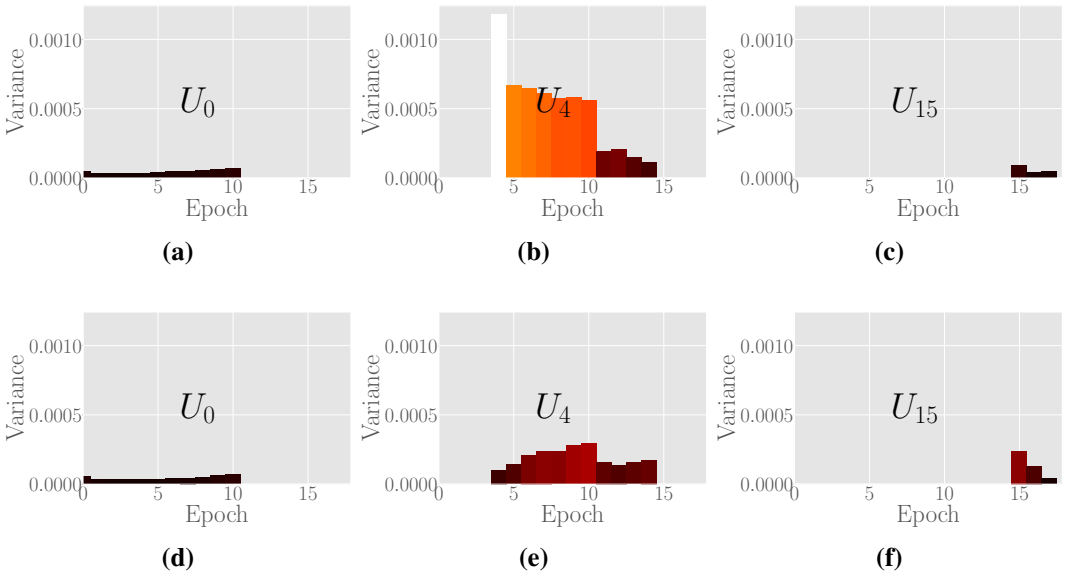


Figure 12.8: H_4 chain: Comparison between the average variance of the gradient of the parameters in layers 0, 4, 15 of the: a, b, c)original and d, e, f) device aware ADAPT algorithms.

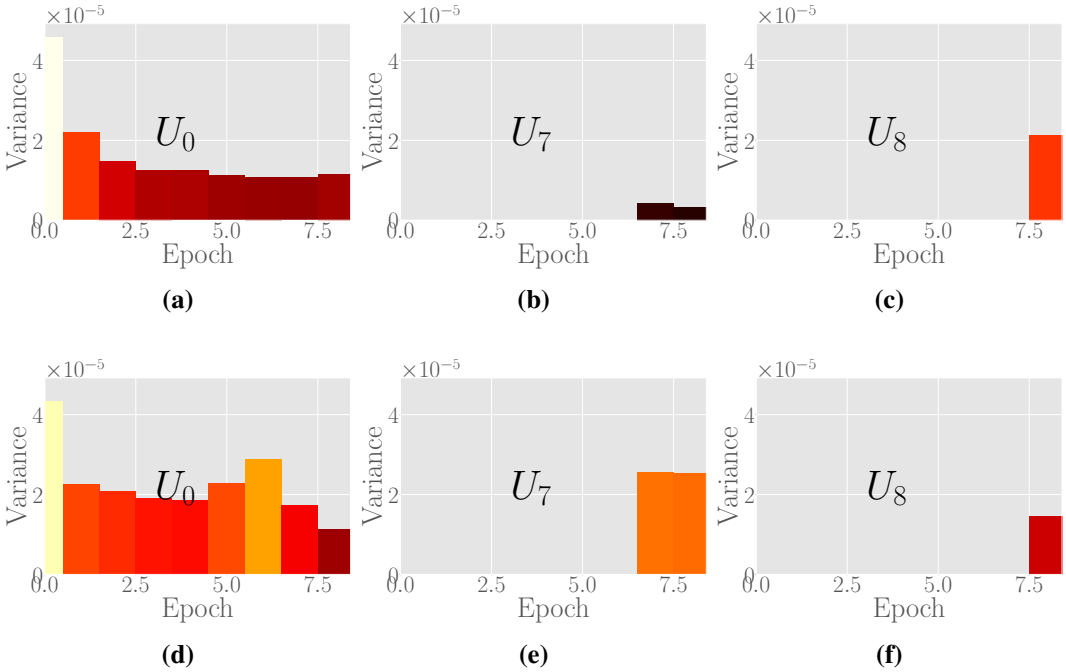


Figure 12.9: LiH: Comparison between the average variance of the gradient of the parameters in layers 0, 4, 15 of the: a, b, c)original and d, e, f) device aware ADAPT algorithms.

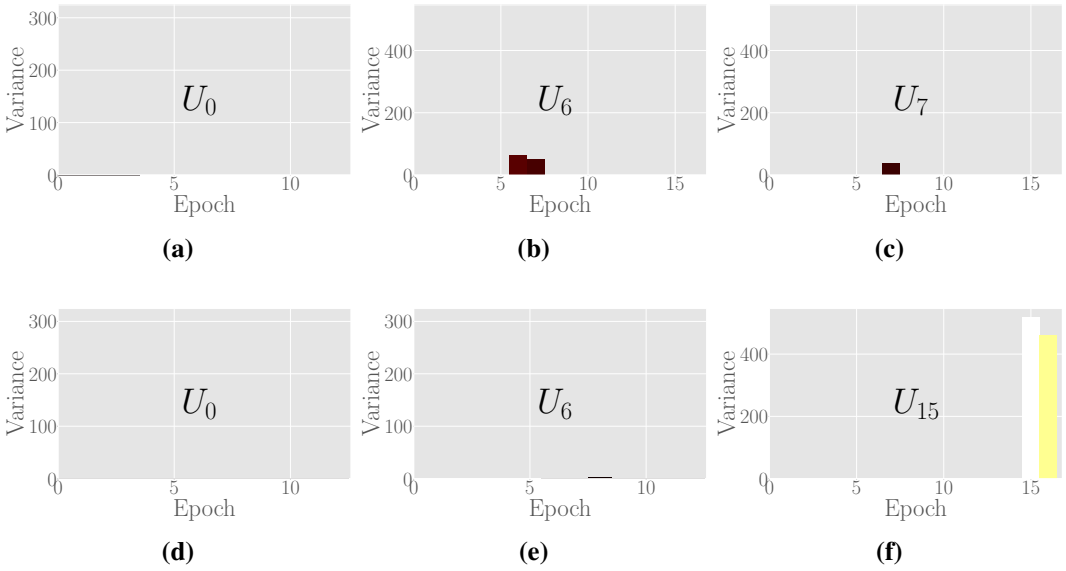


Figure 12.10: 3 site DMFT: Comparison between the average variance of the gradient of the parameters in layers 0, 7, and the final layer of the: a, b, c)original and d, e, f) device aware ADAPT algorithms.

variance. We see the same effect in the DMFT system, that there is a decrease in the variance in the device aware algorithm.

12.3.3.4 Four Site DMFT

In the eight qubit DMFT system, we again observe a decrease in the average parameter gradient, and no increase from restricting the expressibility. In Appendix E we give the variance graphs for all layers of each system, for completeness.

12.4 Discussion

Results presented in Tables 12.1- 12.6 seem to suggest that adding the device aware portion of the algorithm brings modest benefits in the reduction of two qubit gate counts for a reduction in overall accuracy. We also see in Figures 12.8- 12.11 that we do not get an increase in trainability by arbitrarily reducing expressibility. On this point we are sanguine, as it was not expected that we would get an increase in trainability *and* a reduction in the circuit noise[6]. We are more optimistic about the device aware algorithm regarding the first point, as it seems that the reduction in two-qubit gate count improves with the size of the system, and we are allowing the t

[6]No such thing as a free lunch.

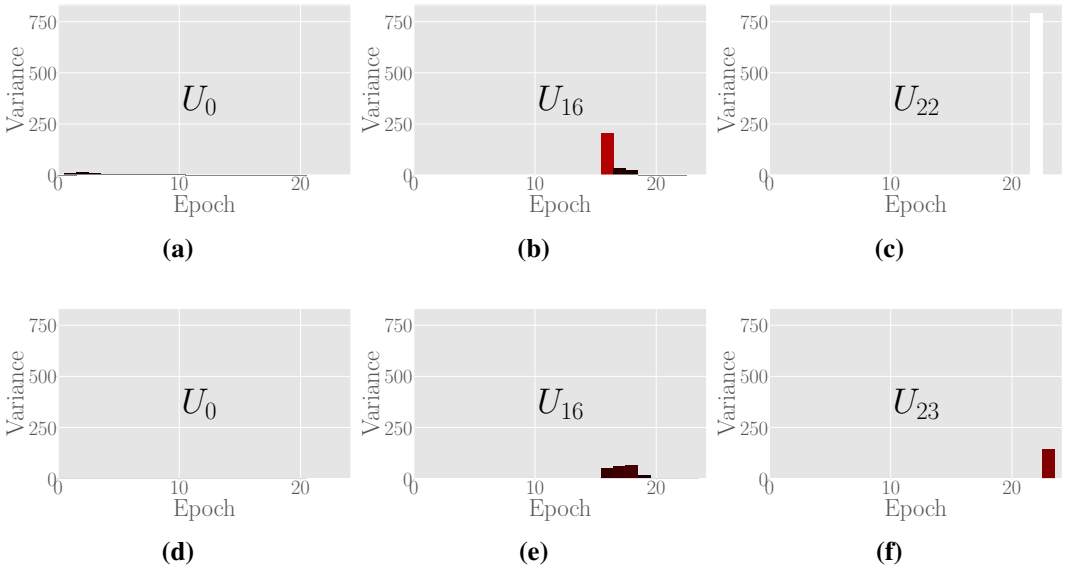


Figure 12.11: 4 site DMFT: Comparison between the average variance of the gradient of the parameters in layers 0, 7, and the final layer of the: a, b, c)original and d, e, f) device aware ADAPT algorithms.

$|\text{ket}\rangle$ compiler to move sites at will - if we consider a noise model that includes a bad qubit, we could avoid this qubit algorithmically if do not allow the compiler to move qubits. We do not believe that t$|\text{ket}\rangle$ allows for the 'avoidance' of bad qubits yet. We believe the significance of the result here is a demonstration of the hypothesis proposed in [215], that a reduction in expressibility is not an easy route to restoring trainability to an ansatz.

We were also able to demonstrate an instance of the ADAPT algorithm on the DMFT system for the first time, and we have noted some of the issues encountered. We note that whilst the sparse operator pool $\{G\}_n$ proposed in [42] works well to find the ground state of molecular systems, it fails with this system. Future work could be done to improve the operator pool available for the algorithms based on the Fermi-Hubbard model such as DMFT. Another proposal which we do not have time to implement is a form of the Adam optimiser tailored to the ADAPT algorithm, e.g. one which does not reset at each epoch but instead adjusts its per-parameter learning rate dependent on the late VQE iteration in ADAPT.

Part IV

Noisy Quantum Devices

Chapter 13

Techniques for Mitigating Noise in Quantum Devices

In Chapter 10 we discussed algorithms for simulating chemical systems on small quantum devices, and in Chapter 11 we saw how using knowledge of the underlying system allowed us to decrease the width of the circuit and vastly improve results. We also saw in Chapter 6 that a variational algorithm for a quantum machine learning task can, with some circuit reductions still perform well in the presence of noise, optimising in some cases better than the analytical best POVM in a noisy circuit. However, none of these discussions has engaged in techniques to reduce the noise in the final calculation. In this chapter we will discuss some of the algorithmic techniques for reducing the effect of noise in quantum calculations, and in the following chapter we present a pulse level simulation of noise in a specific quantum device, with the aim that the ability to model and understand the noise better allows us to target experimental techniques on certain noise sources.

There are some techniques for noise mitigation, such as the Haah spin-echo [298] and dynamical decoupling [299] which are implemented at the hardware level. Each uses non-circuit gate pulses that rotate the qubit state, bringing a state undergoing decoherence back into coherence. These techniques can increase the coherence times of a single qubit by almost 40 times [298] but as they are usually implemented at the device level, we will not discuss them further in this thesis.

In Chapter 3 we saw of the forms that noise in quantum devices can take, and

equations to modelling noise as a quantum channel. In this thesis, all noise simulated will be of the form presented in Chapter 3, except in Chapter 14 where we investigate a noise model with more grounding in experimental reality.

13.1 Randomised Techniques

Randomisation is a powerful tool in noise mitigation, and has been used in several different algorithms to measure and reduce the overall noise [300]–[306]. The power of randomisation is that by randomising systematic sources of noise they can be turned into stochastic noise sources. Stochastic sources of noise can be mitigated easier, either through algorithmic techniques or in fully error corrected quantum devices.

Using randomisation in noise measurement and mitigation relies upon two features of quantum circuits. In measuring the noise of a quantum circuit, randomised benchmarking relies upon the fact that the Clifford group, $\{X, Z, H, CNOT\}$ can be simulated classically, [52] and has a group structure.

The power of randomisation in mitigating noise is related to the group structure of the Pauli gates, e.g.

$$\sigma_j \sigma_k = \delta_{jk} \mathbb{1} + \varepsilon_{jkl} \sigma_l, \tag{13.1}$$

where ε_{jkl} is the Levi-Cita symbol. Essentially, as we can express a Pauli gate as a combination of two other Paulis we can randomise the actual gates performed in a circuit for each repetition of that circuit.

In this section we will first discuss how randomisation is used to measure the noise in a device, and then we will show how it has also been used to reduce the noise in quantum computations.

13.1.1 Randomised Benchmarking

As the Clifford group is classically simulable, we can propose a very long Clifford circuit and use a classical computer to output a shallow operation which can undo the circuit (exploiting the group structure). Randomised benchmarking [300]–[303] uses these properties to provide a single number that can measure the typical noise in a device when running a complex and deep quantum circuit.

The steps in randomised benchmarking are as follows:

1. For each m in the sequence $\{1,2,\ldots,m\ldots M\}$:

2. Generate a random sequence of m gates from the Clifford group on the n qubits in the device.

3. Use the classical device to generate the final Clifford operation such that the overall effect of the circuit is the identity. The sequence of gates is denoted i_m and we write the operator of circuit performed with noise as $\mathcal{S}_{i_m}$

4. For each sequence use the quantum device to measure $\mathrm{Tr}\left[E_\psi \mathcal{S}_{i_m}(\rho_\psi)\right]$, where ρ_ψ and E_ψ are the initial state and POVM measurement taking into account any errors.

5. Average over many randomisations for the sequence length m, the averaged sequence fidelity is given by

$$F_{\mathrm{seq}}(m,\psi) = \mathrm{Tr}\left[E_\psi \mathcal{S}_m\left(\rho_\psi\right)\right], \tag{13.2}$$

with

$$\mathcal{S}_m = \frac{1}{|\{i_m\}|}\sum_{|\{i_m\}|}^{i_m} \mathcal{S}_{i_m} \tag{13.3}$$

the average over all $\mathcal{S}_{i_m}$.

6. Fit these results to the model

$$F_{\mathrm{seq}}^{(1)}(m,\psi) = A_1 p^m + B_1 + C_1 (m-1)\left(q-p^2\right)p^{m-2}, \tag{13.4}$$

where A_1, B_1, C_1 account for the state preparation and measurement errors, and the error from the final operation. p parametrises the noise for the circuit, which determines r in the next step, and $q = \sum_{j=2}^{m} \frac{q_j}{(m-1)}$ is the depolarising parameter of the gates, meaning $(q-p^2)$ is a measure of the gate dependence of the errors.

7. The average error rate, r, for this instance of m is given by

$$r = 1 - p - \frac{1}{d}(1-p). \tag{13.5}$$

with $d = 2^n$, the dimension of this circuit.

8. Do this for the next m up to M to find how the error per Clifford (EPC) changes with circuit length.

First we consider the simpler case where there is no gate or time dependence to the errors. This is not a realistic model, as gate dependence is a feature of many implementations [307], [308][1], and errors usually increase over time with calibration drift [22], [309], but it we will use it as a starting point. We denote the ideal operation as $\mathcal{C}_{i_j}$ and the noise associated with it as $\Lambda_{i_j,j}$ giving the operation implemented, $\mathcal{S}_{I_m}$ as:

$$\mathcal{S}_{i_m} = \prod_{j=1}^{m+1} \left(\Lambda_{i_j,j}\mathcal{C}_{i_j}\right). \tag{13.6}$$

In the gate and time independent simplification $\Lambda_{i_j,j} = \Lambda$.

Implementing the identity operation $\mathcal{C}_{i_j}C_{i_j}^{\dagger}$ creates the overall operation

$$\mathcal{S}_{i_m} = \Lambda\Pi_{j=1}^{m}\left(\Pi_{i_j}^{i_1}\mathcal{C}_{i_j}^{\dagger}\Lambda\Pi_{i_j}^{i_1}\mathcal{C}_{i_j}\right) \tag{13.7}$$

averaging over this circuit gives the average sequence fidelity

$$F_{\text{seq}}(m,\psi) = \text{Tr}\left[E_\psi\Lambda\Lambda_{\text{twirl}}^{m}(\rho_\psi)\right], \tag{13.8}$$

where $\Lambda_{\text{twirl}} = \sum_{i_j}\frac{\Pi_{i_j}^{i_j}\mathcal{C}_{i_j}^{\dagger}\Lambda\mathcal{C}_{i_j}}{K}$., with K the size of the Clifford group on n qubits. We can here define $\mathcal{D}_{i_j} = \Pi_j^1\mathcal{C}_{i_j}$ and $\tilde{\Lambda} = \mathcal{D}_{i_j}^{\dagger}\Lambda\mathcal{D}_{i_j}$ This reduces to the gate independent fidelity model

$$F_{\text{seq}}^{0}(m,\psi) = A_0p^m + B_0, \tag{13.9}$$

with $A_0 = \text{Tr}\left[E_\psi\Lambda\left(\rho_\psi - \frac{1}{d}\mathbb{1}\right)\right]$, and $B_0 = \text{Tr}\left[E_\psi\lambda\frac{\mathbb{1}}{d}\right]$. However, this is not realistic,

[1]This can be an advantage, as error correcting codes can have better performance under biased noise. [310], [311].

and Λ depends upon the gate and the time[2] $\Lambda \to \Lambda_{i_j,j}$ so we will make a peterbative expansion of $\Lambda_{i_j,j}$ around the mean value of $\Lambda_{i_j,j} \to \bar{\Lambda}$ over all i_j, j. We can also define the deviation from the mean for each $\Lambda_{i_j,j}$ as $\delta\Lambda_{i_j,i} = \Lambda_{i_j,j} - \bar{\Lambda}$. Considering only the first order expansion of the perturbation, $\mathcal{S}_{i_m}^{(1)}$, there are three different cases, firstly the edge cases, where the expansion acts upon the first and the last gates, $\mathcal{S}_{i_m}^{(1a)} + \mathcal{S}_{i_m}^{(1c)}$, and secondly the cases in the bulk of the circuit, $\mathcal{S}_{i_j,j}^{(1b)}$. This expansion gives

$$\mathcal{S}_{i_m}^{(1a)} = \bar{\Lambda}\Pi_{j=m}^{2}\tilde{\Lambda}_{i_j}\left(\mathcal{D}_{i_1}^{\dagger}\delta\Lambda_{i_1,1}\mathcal{D}_{i_1}\right) \tag{13.10}$$

$$\mathcal{S}_{i_m}^{(1b)} = \bar{\Lambda}\tilde{\Lambda}_{i_m}\ldots\left(\mathcal{D}_{i_1}^{\dagger}\delta\Lambda_{i_j,j}\mathcal{D}_{i_j}\right)\ldots\tilde{\Lambda}_{i_1} \tag{13.11}$$

$$\mathcal{S}_{i_m}^{(1c)} = \delta\Lambda_{i_{m+1},m+1}\tilde{\Lambda}_{i_m}\ldots\tilde{\Lambda}_{i_1}. \tag{13.12}$$

Averaging these three terms over each i_m gives

$$\mathcal{S}_{m}^{(1a)} = \Lambda\Lambda_{\text{dep}}^{m-1}\left(\mathcal{Q}_1 - \Lambda_{\text{dep}}\right), \tag{13.13}$$

$$\mathcal{S}_{m}^{(1b)} = \sum_{j=2}^{m}\Lambda\left(\left(\mathcal{Q}_j\Lambda\right)_{\text{dep}} - \Lambda_{\text{dep}^2}\right)\Lambda_{\text{dep}}^{m-2}, \tag{13.14}$$

$$\mathcal{S}_{i_m}^{(1c)} = \left(\mathcal{R}_{m+1} - \Lambda\Lambda_{\text{dep}}\right)\Lambda_{\text{dep}}^{m-1}. \tag{13.15}$$

where we have defined $\mathcal{Q}_j = \frac{1}{K}\sum_{i_j}\mathcal{D}_{i_j}^{\dagger}\Lambda_{i_j,j}\mathcal{D}_{i_j}$, $\mathcal{R}_{m+1} = \frac{1}{K}\sum_{i_m}\Lambda_{i'_m,m+1}\left(\mathcal{C}_{i_m}^{\dagger}\Lambda\mathcal{C}_{i_m}\right)$. Finally we get expressions for the constants in Equation 13.4:

$$A_1 = \text{Tr}\left[E_\psi\Lambda\left(\frac{\mathcal{Q}_1(\rho_\psi)}{p} - \rho_\psi + \frac{(p-1)\mathbb{1}}{pd}\right)\right] + \text{Tr}\left[E_\psi\mathcal{R}_{m+1}\left(\frac{\rho_\psi}{p} - \frac{1}{pd}\mathbb{1}\right)\right], \tag{13.16}$$

$$B_1 = \text{Tr}\left[E_\psi\mathcal{R}_{m+1}\left(\frac{1}{d}\mathbb{1}\right)\right], \tag{13.17}$$

$$C_1 = \text{Tr}\left[E_\psi\Lambda\left(\rho_\psi 1\frac{1}{d}\mathbb{1}\right)\right]. \tag{13.18}$$

In Chapter 14 we will use the randomised benchmarking algorithm provided in

[2]The assumption here is that the timescale for this is significantly shorter than the timescale of the gate operation.

Qiskit [292] to investigate a simulated model of an ion trap.

13.1.2 Pauli Frame Randomisation and Gate Set Tomography

Gate Set Tomography (GST) [312]–[315] addresses a similar problem to randomised benchmarking - the need to characterise the noise in a device, but here we do not wish to include any State Preparation And Measurement (SPAM) errors. The disadvantage of Gate Set Tomography is that there is an increase in the number of circuits we need to run, and an increase in the complexity of the classical post-processing. The post-processing requires solving a non-linear optimisation problem, given in Eqn. 13.21 for which a semi-definite programming problem can be used.

The GST protocol is:

1. Write the set of experimental gates as:

$$G^{(\text{exp})} = \left\{\mathcal{R}_{\mathcal{E}_1}, \mathcal{R}_1, \mathcal{R}_{\mathcal{E}_2}, \mathcal{R}_2, \ldots \mathcal{R}_{\mathcal{E}_N}, \mathcal{R}_N\right\}, \tag{13.19}$$

   the faulty version of the ideal gate set.

2. Perform a series of N^3 experiments of the form:

$$m_{ijk} = \langle\langle M_0 | \mathcal{R}_{\mathcal{E}_k} \mathcal{R}_k \mathcal{R}_{\mathcal{E}_j} \mathcal{R}_j \mathcal{R}_{\mathcal{E}_i} \mathcal{R}_i | \rho_0 \rangle\rangle. \tag{13.20}$$

3. This contains the tomography data necessary to reconstruct each element of $G^{(exp)}$

4. If we assume that the errors, $\mathcal{R}_{\mathcal{E}_n}$ are physical and small, $||\mathcal{E}_n|| \ll 1$, we can describe the problem as a least-squares estimation:

$$\begin{aligned} &min_{\tilde{\mathcal{R}}_{\mathcal{E}_1} \ldots} \sum_{ijk} \Big| m_{ijk} \\ &+ \langle |M_0\rangle 2\mathcal{R}_k \mathcal{R}_j \mathcal{R}_i - \tilde{\mathcal{R}}_{\mathcal{E}_k} \mathcal{R}_k \mathcal{R}_j \mathcal{R}_i - \mathcal{R}_k \tilde{\mathcal{R}}_{\mathcal{E}_j} \mathcal{R}_j \mathcal{R}_i - \mathcal{R}_k \mathcal{R}_j \tilde{\mathcal{R}}_{\mathcal{E}_i} \mathcal{R}_i | \rho_0 \rangle\rangle \Big|^2, \end{aligned} \tag{13.21}$$

   where the $\tilde{\cdot}$ represent the free parameters of the estimation.

5. Solving the least-squares estimation problem can be classically costly, but produces a good estimation of the error maps, $\tilde{\mathcal{R}}_{\mathcal{E}_i}$ of the gate set.

In [305], they use GST to show that randomised compiling, discussed below, turns coherent and systematic errors into a Pauli error channel.

13.1.3 Randomised Compiling

The features of Clifford gates which make the above randomised benchmarking scheme scalable can also be used in a standard quantum computation. Randomised compiling [304] tailors the noise inherent in the device, which could have arbitrary coherence or spatial correlations (systematic noise) into stochastic Pauli noise. One motivation for converting the noise this way is that the error thresholds for error correcting codes are much higher for stochastic Pauli noise (10^{-3} [316]), where the noise channel can be described by a weighted sequence of Pauli operators, than for generic noise (10^{-6} [317]), where the noise must be treated by a full Lindblad master equation. We can define the average error rate $r(\mathcal{E})$ for a noise map $\mathcal{E}$ as the gate fidelity over all input states,

$$r(\mathcal{E}) = 1 - \int d\psi \, \langle\psi| \mathcal{E}\left(|\psi\rangle\langle\psi|\right) |\psi\rangle \,, \tag{13.22}$$

and the worst case error-rate, ε as the diamond norm of the distance from identity, the diamond norm, discussed in Section 10.4.5, is the 1-norm maximised over all states,

$$\varepsilon(\mathcal{E}) = \frac{1}{2} \left|\left| \mathcal{E} - \mathbb{1} \right|\right|_{\diamond} . \tag{13.23}$$

Stochastic Pauli noise also has a lower worst-case error rate than coherent errors, given the same average error rate.

The randomised compiling protocol splits the available hardware gates into a set of ‘easy’ and ‘hard’ gates, which are determined by the hardware and the error rates on each gate. Generally speaking, the hard gates consist of the two qubit gates (CNOT or CZ), and the harder single qubit gates (H and $\mathrm{T} = |0\rangle\langle 0| + e^{\frac{i\pi}{8}} |1\rangle\langle 1|$). The easy gates are the single qubit Pauli gates and the phase gate (T^2).

The circuits can now be separated into clock cycles which consist of a round of

easy gates followed by a round of hard gates. To perform randomised compiling, we replace each easy gate with a 'dressed' easy gate:

$$C_k \to \tilde{C}_k = T_k C_k T_{k-1}^c, \tag{13.24}$$

where the gates T_k are randomly chosen from the subset of easy gates (the twirling set) and T_k^c is the correction gate: $T_k^c = G_k T_k^\dagger G_k^\dagger$, i.e. it undoes the randomisation on the previous step, with the previous hard gate, G_k taken into account. The compiler now recompiles the three twirled easy gates into a single easy gate, ensuring the final circuit contains the same number of gates and is logically identical to the original circuit. Ideally, this randomisation is repeated for each single-shot repeat of the circuit, meaning that a single shot of the circuit will not have the noise tailored to the Pauli channel, but over the whole experiment the noise can be described as:

$$\mathcal{E}(\rho) = \sum_{P \in \mathbf{P}^{\otimes \mathbf{n}}} c_P P \rho P^\dagger, \tag{13.25}$$

where $\mathbf{P}^{\otimes n}$ is the set of Pauli operators. This is the stochastic Pauli noise channel, which has the good properties for error correcting codes described above.

Chapter 14

Simulating A Noisy Ion Trap at the Pulse Level

Work in this chapter was undertaken in collaboration with the Ion microtrap team at NPL, and discussions of the ion trap device will mostly match their device.

In previous chapters, when noise in a device has been considered it has been a simple noise model, e.g. symmetric depolarising noise in Chapter 6. As a general noise model, and one which is most stringent on any simulated algorithms, this is a good model to use in numerical simulations, it also matches well the noise seen in some real devices such as the Google Bristlecone chip[1]. However, when working with a single device, and targeting specific sources of noise we require a more detailed noise model.

In this chapter we will explore the process of building such a noise model for an ion trap device in use at the National Physical Laboratory (NPL) [77], [79], [84]. The noise model relies upon pulse level simulations of quantum operations on the ions, and includes an Application Programming Interface (API) to integrate the simulator with Qiskit [292], to take advantage of the randomised benchmarking module. Simulated randomised benchmarking experiments were then ran for real and simulated levels of noise sources, which can aid in prioritising work on reducing certain sources of noise.

[1] Private communication, Ryan Babbush, 2019

14.1 Modelling Ion Traps

We will begin by following the treatment of trapped ion devices in [318], introducing the Hamiltonians used to model trapped ion quantum systems. We shall then continue by introducing the Lindblad operators used to model the noise in the device at NPL. This noise is parameterised, which means that the simulation of the device can be modified to simulate similar devices, with enough knowledge of local conditions.

We describe the ion as a two-level quantum system, driven by a classical force corresponding to the trapping potential. The Hamiltonian of the two-level system is given by:

$$H_{\text{sys}} = \frac{1}{2}\hbar\omega_0\sigma_z, \tag{14.1}$$

where ω_0 is the energy difference between the two states, σ_z is the Pauli-z operator, and$\hbar$ is Planck's constant. We consider only the motion of the ion along the axis of the trapping field, and describe this with the standard harmonic oscillator Hamiltonian:

$$H_{\text{trap}} = \frac{\hat{\mathbf{p}}^2}{2m} + \frac{1}{2}\omega_z^2\hat{\mathbf{z}}^2 \tag{14.2}$$

where $\hat{\mathbf{z}}$ is the position along the trap axis, ω_z is the trapping field potential, and $\hat{\mathbf{p}} = -\hbar\frac{\partial}{\partial z}$, the momentum operator. We can introduce the standard ladder operators:

$$\hat{a} = \sqrt{\frac{m\omega_z}{2\hbar}}(\hat{\mathbf{z}} + \frac{i}{m\omega_z}\hat{\mathbf{p}}), \tag{14.3}$$

$$\hat{a}^\dagger = \sqrt{\frac{m\omega_z}{2\hbar}}(\hat{\mathbf{z}} - \frac{i}{m\omega_z}\hat{\mathbf{p}}), \tag{14.4}$$

to rewrite the trapping Hamiltonian as:

$$H_{\text{trap}} = \hbar\omega_z(\frac{1}{2} + \hat{a}^\dagger\hat{a}). \tag{14.5}$$

Finally, we must consider the interaction of the ion and the laser light driving gates, in this we will not consider the readout light field or cooling field as they do not materially affect the operation of quantum gates. We will first consider a single ion interacting with a light field given by an effective frequency ω, effective phase,

ϕ, and effective wavevector $k = |\mathbf{k}|\cos\theta$, where $\mathbf{k}$ is the effective wavevector of the light and θ is the angle between the laser and the trapping z-axis. The effective frequency and wavevector is the difference of those quantities of the two light fields illuminating the ion. These are associated with a certain on-resonance Rabi frequency, Ω for the transition described. This interaction is then described by the Hamiltonian:

$$H_{\text{int}} = \frac{\hbar}{2}\Omega(\hat{\sigma}_+ + \hat{\sigma}_-)(e^{i(k\hat{z}-\omega t+\phi)} + e^{i(k\hat{z}-\omega t+\phi)}). \tag{14.6}$$

We then combine these Hamiltonians to get a full description of the system, given by:

$$H = H_{\text{free}} + H_{\text{int}} \tag{14.7}$$

Where H_{free} is the free Hamiltonian given by:

$$H_{\text{free}} = H_{\text{sys}} + H_{\text{trap}} = \hbar(\omega_0\sigma_z + \omega_z(\frac{1}{2} + \hat{a}^\dagger\hat{a})). \tag{14.8}$$

By splitting the Hamiltonian this way, we are easily able to see how to move to the interaction picture [318], with the interaction operator, $\hat{V} = H_{\text{int}}$ and the

transformation operator $\hat{U}_0 = \exp[-(i/\hbar)H_{\text{free}}t]$

$$H_I = \hat{U}_0^\dagger H_{\text{int}} \hat{U}_0 \tag{14.9}$$

$$= \frac{\hbar}{2}\Omega \exp\left[\frac{i}{2}\omega_0\sigma_z t\right](\hat{\sigma}_+ + \hat{\sigma}_-)\exp\left[\frac{-i}{2}\omega_0\sigma_z t\right]$$
$$\times \exp\left[(i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right](e^{i(k\hat{z}-\omega t+\phi)} + e^{i(k\hat{z}-\omega t+\phi)})\exp\left[(-i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right] \tag{14.10}$$

$$= \frac{\hbar}{2}\Omega(\hat{\sigma}_+ e^{i\omega_0 t} + e^{-i\omega_0 t}\hat{\sigma}_-)$$
$$\times \exp\left[(i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right](e^{i(\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))-\omega t+\phi)} + e^{i(\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))-\omega t+\phi)})$$
$$\times \exp\left[(-i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right] \tag{14.11}$$

$$= \frac{\hbar}{2}\Omega(\hat{\sigma}_+ e^{i\omega_0 t} + e^{-i\omega_0 t}\hat{\sigma}_-)$$
$$\times \exp\left[(i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right](e^{i(\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))-\omega t+\phi)} + e^{i(\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))-\omega t+\phi)})$$
$$\times \exp\left[(-i\omega_z(\frac{1}{2}+\hat{a}^\dagger\hat{a})t\right] \tag{14.12}$$

The first substitution in the second line is covered in Appendix F, and the second substitution can be made by reversing the identities in equations 14.3 & 14.4, considering we have moved into the Heisenberg picture, and have picked up factors of $u(t)$ and $u^*(t)$, this derivation is covered in [318].

$$\hat{z}(t) = \sqrt{\frac{2m\omega_z}{\hbar}}(\hat{a}u^*(t) + \hat{a}^\dagger u(t)), \tag{14.13}$$

making the substitution $\eta = kz_0$, with $z_0 = \sqrt{\frac{2m\omega_z}{\hbar}}$ and η is the Lamb-Dicke parameter.

In the second half of the equation, dealing with the motional states, the Hamiltonian H_{trap} will be cancelled, so we can expand the whole Hamiltonian as:

$$H_I = \frac{\hbar}{2}\Omega\Big(\sigma_+ e^{i(\omega_0-\omega)t+\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))+\phi} + \sigma_- e^{-i(\omega_0-\omega)t+\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))+\phi}$$
$$\sigma_- e^{i(\omega_0+\omega)t-\eta(\hat{a}u^*(t)+\hat{a}^\dagger u(t))-\phi} + \sigma_+ e^{-i(\omega_0+\omega)t-\eta(\hat{a}u^*(t)-\hat{a}^\dagger u(t))-\phi}\Big). \tag{14.14}$$

The terms on the second line oscillate with frequency $\omega_0 + \omega$, as this is a fast oscillation which has little effect on the evolution, these terms are neglected; this is called the Rotating Wave Approximation (RWA). We also make the substitution $\delta = \omega_0 - \omega$. Finally, we will consider the expression $\hat{a}u^*(t) + \hat{a}^\dagger u(t)$, where the $u(t)$ is only slowly varying when the detuning of the laser δ is equal to an integer multiple of the trapping frequency, ω_z, called the motional sidebands, and $u(t)$ simplifies to $e^{i\omega_z t}$. Here we substitute $\omega_z \approx \nu$ to maintain consistent notation with the literature, a full derivation for this can again be found in [318]. The final form of the Hamiltonian is then:

$$H = \frac{\hbar}{2}\Omega\sigma_+ \exp\Big\{i\eta(\hat{a}e^{-i\nu t} + \hat{a}^\dagger e^{i\nu t})\Big\}e^{-i(\delta t+\phi)} + \text{H.c.} \tag{14.15}$$

14.2 Two-Qubit Gate

From equation 14.15 we see that the vibrational mode of the ion can be entangled with the internal state of the ion. The Mølmer-Sørensen gate [319] uses this principle to enact an entangling gate on a string of ions. Here we will only consider the two-qubit gate for simplicity. The ions are illuminated with light tuned to two different frequencies, to the red and blue motional sidebands; i.e. $\omega_\pm = \omega_0 \pm (\omega_z + \Delta)$, where Δ is a small detuning that can be optimised. Adding the interactions from the two light fields for two ions gives a Mølmer-Sørensen gate Hamiltonian:

$$H_{\text{MS}} = \frac{\hbar}{2}\Omega(\sigma_{+,1} + \sigma_{+,2}) \exp\Big\{i(\eta(\hat{a}e^{-i\nu t} + \hat{a}^\dagger e^{i\nu t}))\Big\}e^{-i(\delta t+\phi)} + \text{H.c.}, \tag{14.16}$$

where the $\sigma_{+,i}$ is the operator acting on ion i. Here, we can make the Lamb-Dicke approximation, which is to assume that $\eta << 1$, and so we can expand $\exp\big\{i(\eta(\hat{a}e^{-i\nu t} + \hat{a}^\dagger e^{i\nu t}))\big\}$ to first order, and disregard terms of $\mathcal{O}(\eta^2)$ and above. This leaves us with the Hamiltonian,

$$H_{\text{MS}} = \frac{\hbar}{2}\Omega\eta e^{-i\phi}(\sigma_{+,1} + \sigma_{+,2})((\hat{a}e^{-i\varepsilon t} + \hat{a}^\dagger e^{i\varepsilon t})) + \text{H.c.}, \tag{14.17}$$

where we have made the substitution $\varepsilon = \nu - \delta$.

14.3 Sources of Noise

To numerically model the Ion Trap, we will use QuTip [320], which numerically simulates the ion trap using the Lindblad master equation:

$$\dot{\rho} = -\frac{i}{\hbar}[H,\rho] + \sum_i \Gamma_i \left(\mathcal{L}_i \rho \mathcal{L}_i^\dagger - \frac{1}{2}\left\{ \mathcal{L}_i^\dagger \mathcal{L}_i, \rho \right\} \right). \tag{14.18}$$

All evolution in quantum mechanics is unitary, so to model noisy dynamics, i.e. non-unitary evolution, Lindblad operators, $\mathcal{L}_i$ are introduced. These are constrained to $\sum_i \mathcal{L}_i^\dagger \mathcal{L}_i = \mathbb{1}$ to ensure that the overall evolution is completely positive and trace-preserving [43].

In this section, we will list the noise sources modelled in this project, the derivation of their Lindblad operators and the associated rates, Γ_i. We will begin with the derivation of the noise master equation for single qubit gates to aid explanation, and then we will focus upon the Mølmer-Sørensen gate.

14.3.1 Driving Field Amplitude Noise

14.3.1.1 Single Qubit Noise

To model noise on the single qubit gates in the system, we will use the Hamiltonian for Rabi flopping, making adjustments to the driving Hamiltonian for the appropriate noise source. Noise on the amplitude of the driving laser directly changes the Rabi frequency of the transition:

$$H = \hbar\delta\sigma_z + \frac{1}{2}\hbar(\Omega_0 + \Delta\Omega(t))\sigma_x, \tag{14.19}$$

where $\Delta\Omega(t)$ is the time dependent noise in the Rabi frequency, and δ is the beam detuning from ω_0. As the Rabi frequency is proportional to the amplitude of the driving beam, we can assume the same for the noisy term, $\Omega(t) \propto V(t)$. Utilising the relation between autocorrelation and Power Spectral Density (PSD) as given in Appendix G, we can find a relation between this noise term and the PSD of the

amplitude white noise:

$$G(\tau) = \langle V(t_1)V(t_1+\tau)\rangle, \tag{14.20}$$

$$= \lim_{T\to 0}\frac{1}{T}\int_0^T V(t+\tau)V(t)dt. \tag{14.21}$$

the final relation is only true in the limit of white noise, $N_0(f)$ is the amplitude noise PSD and P_0 is the power in the beam.

In Appendix H we derive the form of the master equation:

$$\hbar\frac{d\langle\rho\rangle}{dt} = [H_1,\langle\rho\rangle] - \frac{i}{2\hbar}\Gamma[H_2,[H_2,\langle\rho\rangle]], \tag{14.22}$$

in what follows we will drop the expectation value of ρ, as it will always be assumed this is the case. This master equation can be converted into the Lindblad form, Eqn. 14.18

By substituting $H_1 = \frac{1}{2}(\delta\sigma_z + \Omega_0\sigma_x)$, $H_2 = \sigma_x$, $\Gamma = \Omega_0\sqrt{\frac{N_0}{8P_0}}$, we get the master equation including terms for white noise on the driving field amplitude:

$$\frac{d\rho}{dt} = \frac{1}{2}i\delta[\sigma_z,\rho] - \frac{1}{2}i\Omega_0[\sigma_x,\rho] - \frac{N_0}{8P_0}\Omega_0^2[\sigma_x,[\sigma_x,\rho]]. \tag{14.23}$$

Converted into the Lindbald form, the Lindblad operator is $\mathcal{L} = \sqrt{\Gamma}\sigma_x$.

14.3.1.2 Two-qubit gates

Considering the same noise term when applied to the Mølmer-Sørensen gate, we can write the Hamiltonian as:

$$H = (\Omega + \Delta\Omega(t))\,H_{MS}, \tag{14.24}$$

where:

$$H_{\mathrm{MS}} = \frac{\hbar}{2}\eta e^{-i\phi}(\sigma_{+,1}+\sigma_{+,2})\left(\left(\hat{a}e^{-i\varepsilon t} + \hat{a}^\dagger e^{i\varepsilon t}\right)\right) + \mathrm{H.c.}. \tag{14.25}$$

Applying an analogous derivation as in the single-qubit noise term, we see that in this case, $H_1 = H_2 = H_{MS}$, and the same derivation for Γ can be followed as in 14.20, so $\Gamma = \Omega_0 \sqrt{\frac{N_0}{8P_0}}$.

14.3.2 Driving Field Frequency Noise

14.3.2.1 Single Qubit Noise

Noise on the frequency of the driving laser produces a phase shift on the raising and lowering components of the driving Hamiltonian:

$$\frac{\hbar\Omega}{2}(e^{-i(\delta t+\phi(t))}\sigma_+ + e^{i(\delta t+\phi(t))}\sigma_-). \tag{14.26}$$

To analyse the effect of this, we must make another interaction picture substitution, with the transformation operator: $U = e^{-i\phi(t)\sigma_+\sigma_-}$, so we have:

$$\tilde{\rho} := U^\dagger \rho U \tilde{H} := U^\dagger H U = \frac{\hbar\Omega}{2}(e^{-i\delta t}\sigma_+ + e^{i\delta t}\sigma_-). \tag{14.27}$$

to find the differential $d\tilde{\rho}$, we use the chain rule, noting that $\left(dU^\dagger\right) = i\sigma_+\sigma_- U^\dagger d\phi$:

$$d\tilde{\rho} = i\sigma_+\sigma_- U^\dagger \rho U d\phi - iU^\dagger \rho U \sigma_+\sigma_- d\phi - \frac{i}{\hbar}U^\dagger [H,\rho] U dt \tag{14.28}$$

$$= -\frac{i}{\hbar}\left[\tilde{H},\tilde{\rho}\right] dt + i[\sigma_+\sigma_-,\tilde{\rho}]\, d\phi \tag{14.29}$$

$$= -\frac{i}{\hbar}\left[\tilde{H},\tilde{\rho}\right] dt + i\delta f(t)\,[\sigma_+\sigma_-,\tilde{\rho}]\, dt, \tag{14.30}$$

where the final substitution is for $\delta f(t) = \dot{\phi}(t)$, which represents the fluctuations in frequency, which we can take in the white noise limit:

$$\langle \delta f(t+\tau) f(t) \rangle = \Gamma \delta(t). \tag{14.31}$$

Here, Γ is related to the frequency white noise PSD by:

$$\Gamma = \frac{N_0(f_0)}{P_0} f_0^2 m \tag{14.32}$$

which has a $\frac{1}{f^2}$ spectrum, so that [321] :

$$N_0(f) = N_0(f_0)\left(\frac{f_0}{f}\right)^2. \tag{14.33}$$

We can now use 14.28 and 14.22 to get the master equation:

$$\dot{\rho} = -\frac{1}{2}i\Omega\left[e^{-i\delta t}\sigma_+ + e^{i\delta t}\sigma_-, \rho\right] - \frac{1}{2}\Gamma\left[\sigma_+\sigma_-, [\sigma_+\sigma_-, \rho]\right]. \tag{14.34}$$

14.3.2.2 Two Qubit Noise

For noise in the Mølmer-Sørensen gate, we first assume that the frequency fluctuations will affect each ion the same, to simplify the treatment, and we will proceed in an analogous way to the single qubit treatment. Again, the frequency fluctuations will add a random phase to the raising and lowering operators of the qubit states:

$$H = \frac{\hbar}{2}\eta e^{-i\phi+\delta\phi(t)}\left(\sigma_{+,1}+\sigma_{+,2}\right)\left(\left(\hat{a}e^{-i\varepsilon t} + \hat{a}^\dagger e^{i\varepsilon t}\right)\right) + \text{H.c.}. \tag{14.35}$$

Now, we move to an interaction picture with the transformation

$$U = e^{-i\phi(t)\sigma_{+,1}\sigma_{+,2}\sigma_{-,1}\sigma_{-,2}}. \tag{14.36}$$

So, applying to the state and the Hamiltonian:

$$\tilde{\rho} := U^\dagger \rho U \tag{14.37}$$

$$\tilde{H} := U^\dagger H U = \frac{\hbar}{2}\eta e^{-i\phi}\left(\sigma_{+,1}+\sigma_{+,2}\right)\left(\left(\hat{a}e^{-i\varepsilon t} + \hat{a}^\dagger e^{i\varepsilon t}\right)\right) + \text{H.c.}. \tag{14.38}$$

and similarly to Equation 14.28 we arrive at:

$$d\tilde{\rho} = -\frac{i}{\hbar}\left[\tilde{H}, \tilde{\rho}\right]dt + i\delta f(t)\left[\sigma_{+,1}\sigma_{+,2}\sigma_{-,1}\sigma_{-,2}, \tilde{\rho}\right]dt, \tag{14.39}$$

where Γ has the same form as in Equation 14.32.

14.3.2.3 Frequency noise at resonance

When considering frequency noise for the single qubit gate, we are operating on resonance with the qubit transition frequency. However, a Mølmer-Sørensen gate is operated at the red and blue sidebands - the qubit transition frequency $\pm$ a single vibrational mode frequency. In this case, we will have some residual laser frequency noise resonant with the qubit transition frequency, which has some probability of causing a Rabi flop (or σ_x gate). To model this contribution to the error in our device, we will assume that the noise at the carrier frequency takes the form of a Dirac delta function, and that it only contributes to adding some probability of a σ_x; that is that it contributes the Lindblad operator: $\mathcal{L} = \sqrt{\Gamma}\sigma_x$, where Γ is as defined in Equation 14.32, for the frequency $f_0 = \omega$.

14.3.3 Driving Field Phase Noise

The final contribution to the noise from the qubit laser is phase noise. Phase noise can arise from the quality of the laser phase-locking and from master clock instability, but both amount to the same effect [75]. In all experiments, the experimental frame is chosen to be one that is co-rotating with the Lamour frequency (ω_0), the master clock ensures that the experimental apparatus is rotating with the same frequency. If there is phase noise in the laser or instability in the master clock, the overall effect add some error to the driving frequency. This has the effect of adding a σ_z term to the Hamiltonian.

14.3.3.1 Single Qubit Noise

For the single qubit Rabi flopping, the Hamiltonian is:

$$H = \frac{1}{2}\left(\delta\omega_0(t) - \delta\dot{\phi_N}(t)\right)\sigma_z + \frac{\hbar\Omega}{2}(e^{-i(\delta t)}\sigma_+ + e^{i(\delta t)}\sigma_-). \tag{14.40}$$

where $\dot{\phi_N}$ accounts for the phase fluctuations of the local oscillator or phase noise of the driving laser, and $\delta\omega_0(t)$ accounts for the change in transition frequency of the qubit to due external factors, such as noise in the applied magnetic field.

We can use the same technique as detailed in Appendix H with this Hamiltonian

to arrive at the same master equation with the Lindblad operators:

$$\mathcal{L} = \frac{\sqrt{\Gamma}}{\hbar}\sigma_z \tag{14.41}$$

where here Γ is the white noise PSD of phase fluctuations in the considered term (i.e. master clock fluctuations, driving field fluctuations, or noise fluctuations changing the qubit transition frequency.).

14.3.3.2 Two Qubit Noise

We will assume that the phase noise / transition frequency noise affects each qubit the same for simplicity of treatment, although it is easy to add individual terms. As the noise in this case acts in a similar manner as in the single qubit case, we have the same Lindblad operators as in Equation 14.41.

14.3.4 Noise in the Motional Modes

The discussion of noise in this section and the following relates only to the motion of the ion in the trap. As the Mølmer-Sørensen gate entangles the motional modes of the ions with the internal degrees of freedom, noise in the vibration of the ion will affect this gate, it will not affect the single qubit gates significantly. The ion will vibrate within the trapping field with some frequency, ω_z, as described in Section 14.1.

14.3.4.1 Motional Frequency Fluctuations

First, we will discuss some white noise fluctuations in the oscillation frequency, ω_z of the ion, such that the overall frequency is given by: $\omega(t) = \omega_z + \delta\omega_z(t)$, the Hamiltonian for the motion can then be given by the hamiltonian of a quantum harmonic oscillator:

$$H = \hbar\omega_z\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right) + \hbar\delta\omega_z(t)\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right). \tag{14.42}$$

If we move into the interaction picture with the operator $U = e^{i\omega_o t}$ the remaining Hamiltonian is only the noise component:

$$H_I = \hbar\delta\omega_z(t)\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right). \tag{14.43}$$

If we now assume that the frequency fluctuations are white, i.e.:

$$\langle \delta\omega_z(t+\tau)\delta\omega(t)\rangle = \Gamma\delta(\tau) \tag{14.44}$$

we can follow the treatment in Appendix H to arrive at the master equation:

$$\dot{\rho} = -\frac{1}{2}\Gamma\left[\hat{a}^\dagger\hat{a},\left[\hat{a}^\dagger\hat{a},\rho\right]\right], \tag{14.45}$$

where we can clearly see that the Lindblad operator in this case is:

$$\mathcal{L} = \sqrt{\frac{\Gamma}{2}}\hat{a}^\dagger\hat{a}. \tag{14.46}$$

Following the derivation in [321], we find the rate for this noise to be:

$$\Gamma = \omega_z^2\frac{N_0^{AM}}{P_c}. \tag{14.47}$$

14.3.4.2 Noisy Electric Field

As the ion is a charged particle, stray electric fields from the device can couple to the motion of the ion. In this case we can describe the systems as a quantum harmonic oscillator driven by some fluctuating classical force, $F(t)$:

$$H = \hbar\omega_z\left(\hat{a}^\dagger\hat{a}+\frac{1}{2}\right) - q_0F(t)\left(\hat{a}+\hat{a}^\dagger\right), \tag{14.48}$$

where q_0 is the charge of the ion.

Moving again into the interaction picture, we find:

$$H = -q_0F(t)\left(\hat{a}e^{-i\omega t}+\hat{a}^\dagger e^{i\omega t}\right), \tag{14.49}$$

which gives us the master equation:

$$\dot{\rho} = -\frac{\Gamma}{4\hbar m\omega_z}\left[\hat{a}e^{-i\omega t}+\hat{a}^\dagger e^{i\omega t},\left[\hat{a}e^{-i\omega t}+\hat{a}^\dagger e^{i\omega t},\rho\right]\right] \quad (14.50)$$

$$= -\frac{\Gamma}{4\hbar m\omega_z}\left[e^{-2i\omega t}\left(\hat{a}^2\rho-2\hat{a}\rho\hat{a}+\rho\hat{a}^2\right)+e^{2i\omega t}\left(\hat{a}^{\dagger 2}\rho-2\hat{a}^\dagger\rho\hat{a}^\dagger+\rho\hat{a}^{\dagger 2}\right)\right. \quad (14.51)$$

$$\left.+\hat{a}\hat{a}^\dagger\rho+\hat{a}^\dagger\hat{a}\rho-2\hat{a}^\dagger\rho\hat{a}-2\hat{a}\rho\hat{a}^\dagger+\rho\hat{a}^\dagger\hat{a}+\rho\hat{a}\hat{a}^\dagger\right], \quad (14.52)$$

and dropping the terms which are rotating faster and slower (those prefaced by $e^{\pm 2i\omega t}$):

$$\dot{\rho} = -\frac{\Gamma}{4\hbar m\omega_z}\left[\hat{a}\hat{a}^\dagger\rho+\hat{a}^\dagger\hat{a}\rho-2\hat{a}^\dagger\rho\hat{a}-2\hat{a}\rho\hat{a}^\dagger+\rho\hat{a}^\dagger\hat{a}+\rho\hat{a}\hat{a}^\dagger\right]. \quad (14.53)$$

To calculate the rate Γ, note that the force is a product of the charge on the ion and fluctuating electric fields:

$$F(t) = qE(t), \quad (14.54)$$

we then assume the E field fluctuations are white, and define the PSD of the fluctuations by:

$$\langle E(t+\tau)E(t)\rangle = \frac{1}{2}S_E\delta(\tau) \quad (14.55)$$

We see that this is equivalent to the Lindblad operators:

$$\mathcal{L} = \sqrt{\frac{q^2S_E}{8\hbar m\omega_z}}. \quad (14.56)$$

14.4 Pulse Simulations of Gates

Armed with the master equation and Lindblad operators describing our noise sources, we can use the master equation solver in QuTip [320] to simulate the application of gates on the quantum device by stepping through time with gate Hamiltonians turned on. By defining Hamiltonians and times which correspond to certain gates we can build an API that allows us to control the pulse level simulations of gates on the ion trap device via the Qiskit [292] set of tools. As we have defined X, Z and a two qubit gate, we have a simulator of a universal quantum device, which the

qiskit compiler can compile any circuit to. Although in this work we will use the randomised benchmarking module of qiskit to evaluate the noise in our simulated device.

Randomised benchmarking, as discussed in Chapter 13 has been chosen as it mimics the quantum circuits ran on NISQ devices, and is widely used to evaluate other devices, allowing for easier comparison. We can control individual sources of noise in our simulation, and will asses which source has the largest relative effect on the randomised benchmarking score.

14.5 Simulated Randomised Benchmarking Results

These simulations were made in order to determine which sources of noise have the largest effect on a quantum computation, and so which systems should be prioritised in error reduction work. In each of the results shown here all noise sources apart from the one studied are reduced to 0, whilst the source of interest is sweeped over a range of plausible values. The Error Per Clifford (EPC) is calculated at each of the noise rates, which is plotted to show how focus upon this source can improve the quantum computation.

14.5.1 Heating Rate

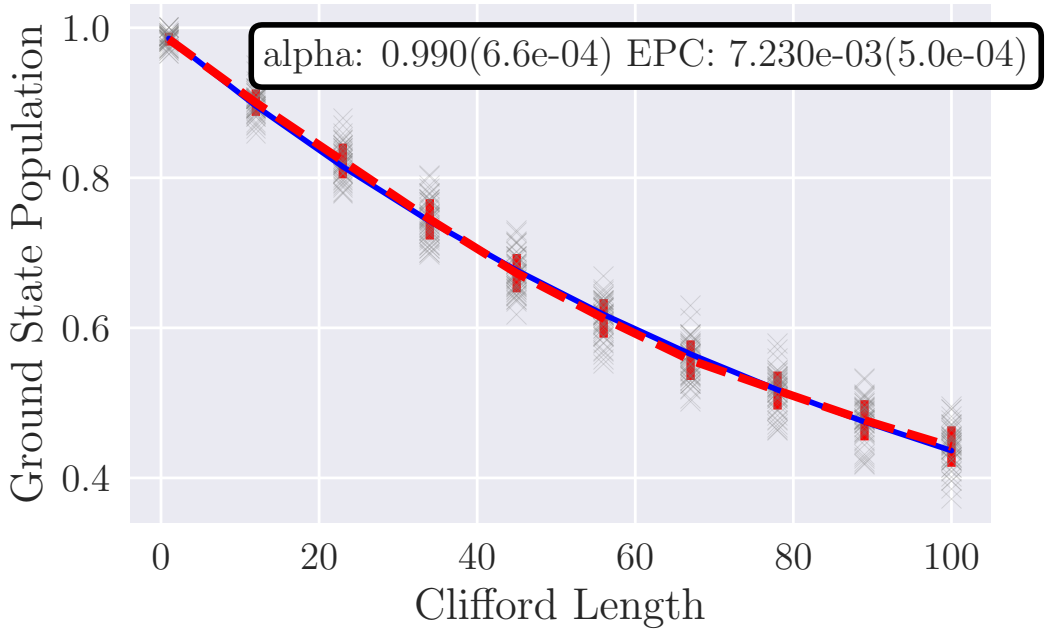
The heating rate of the ion is induced by the coupling of the ion to a noisy electric trapping field, and is related to the noise in the electric field by [322]:

$$\dot{\bar{n}} = \frac{q^2 S_E}{4\hbar m\omega} \tag{14.57}$$

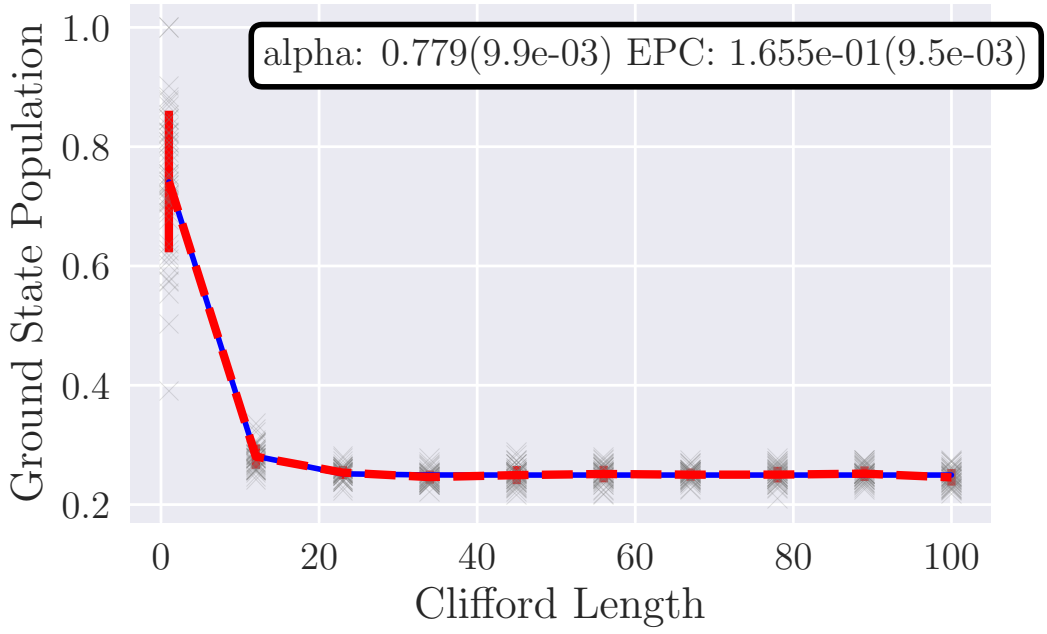
A typical randomised benchmarking experiment, for $\dot{\bar{n}}$ is given in Figure 14.1.

14.5.2 Impact of Noise levels on computation

Here we plot the average Error per Clifford (EPC), taken from randomised benchmarking experiments, against the noise level.



(a) $\dot{\bar{n}} = 0.1$



(b) $\dot{\bar{n}} = 5$

Figure 14.1: The randomised benchmarking results for the two extremes of the range simulated.

14.5.2.1 White Noise

In Figure 14.2 we plot the EPC against the Power Spectral Density (PSD) of noise in the Rabi beam. This noise feeds into noise on both the amplitude and phase of the Rabi beam, which increases the average EPC in a linear fashion. Figure 14.3 shows

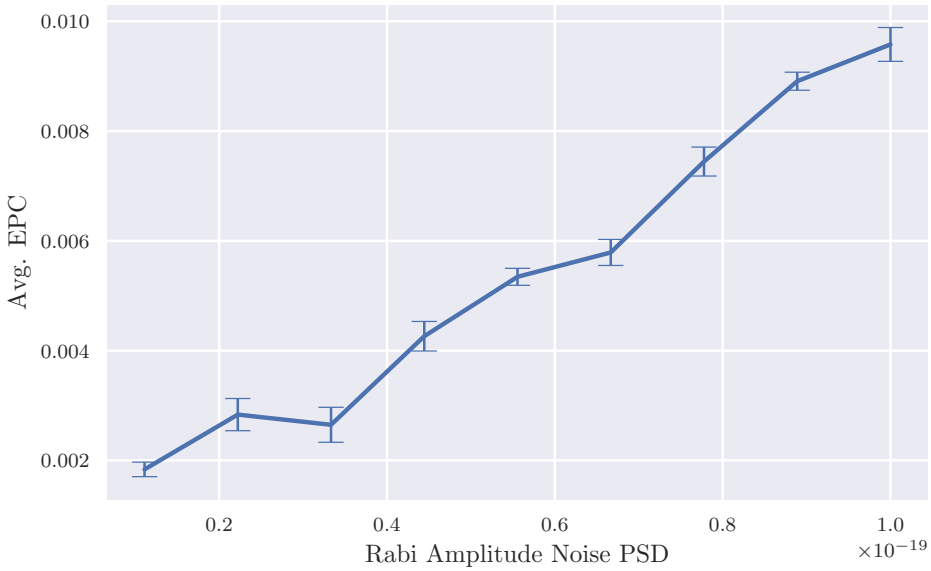


Figure 14.2: The noise PSD of the Rabi beam, and the linearly increasing average EPC this causes.

the EPC for a noisy electric field, which is the same as the individual experiments shown in Figure 14.1. Figure 14.4 shows a related phenomenon: the performance of the trap related to the average number of quanta in the trap at the beginning of the experiment, $\bar{n}$.

14.5.2.2 Jitter

Jitter is a separate class of error to the white noise errors studied previously. The term refers to the instability in the central value of a certain set parameter. In these simulations, jitter is simulated by a random perturbation from the central value at every timestep. The level of jitter describes how large the random perturbation can be.

Figure 14.5 shows how even a small amount of jitter in the setting of the sideband frequency can adversely effect the EPC, whereas Figure 14.6 shows that

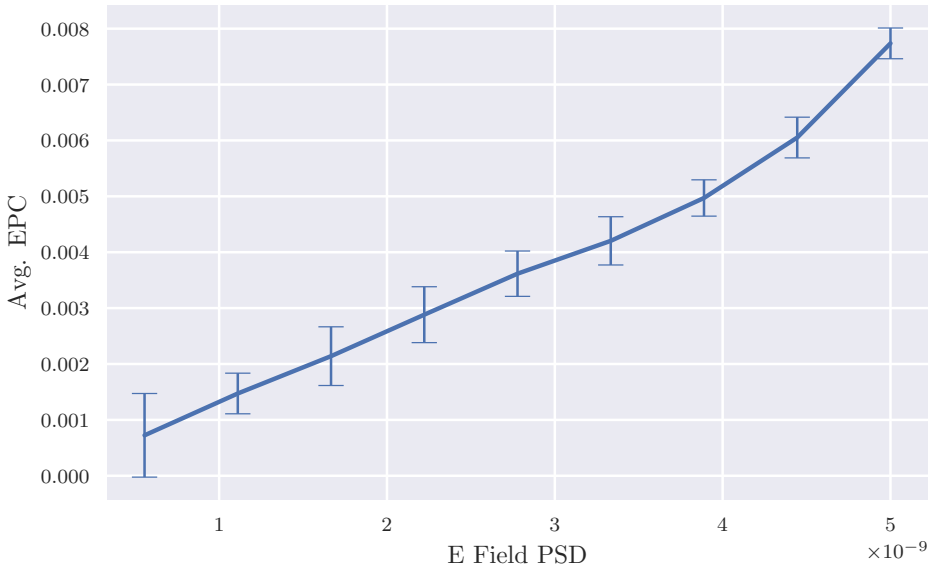


Figure 14.3: The average EPC for an increase in the noise PSD of an electric field, which affect the heating rate of the qubits as in Equation 14.57.

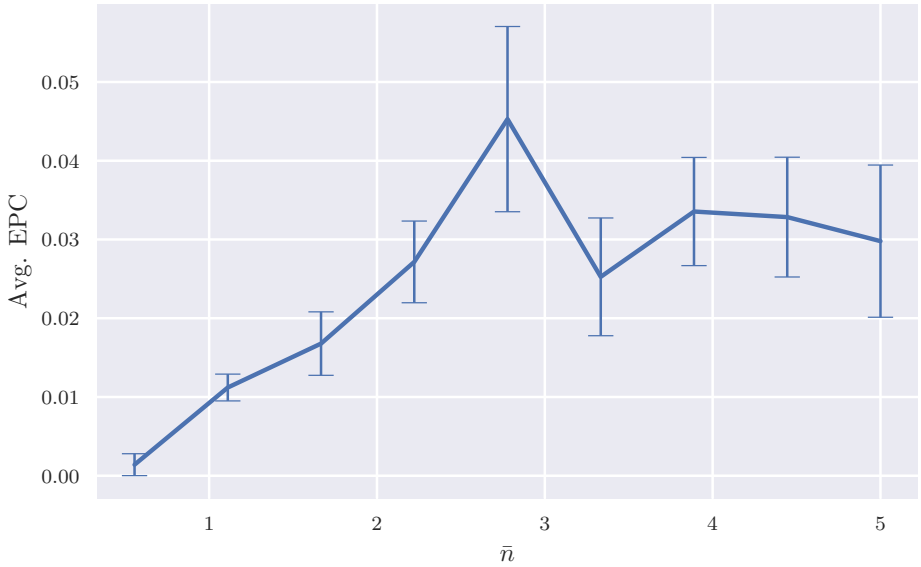


Figure 14.4: The average EPC against the average number of phonon quanta in the trap at the beginning of the experiment, $\bar{n}$

jitter on the value of the Rabi frequency has a less pronounced and more random effect.

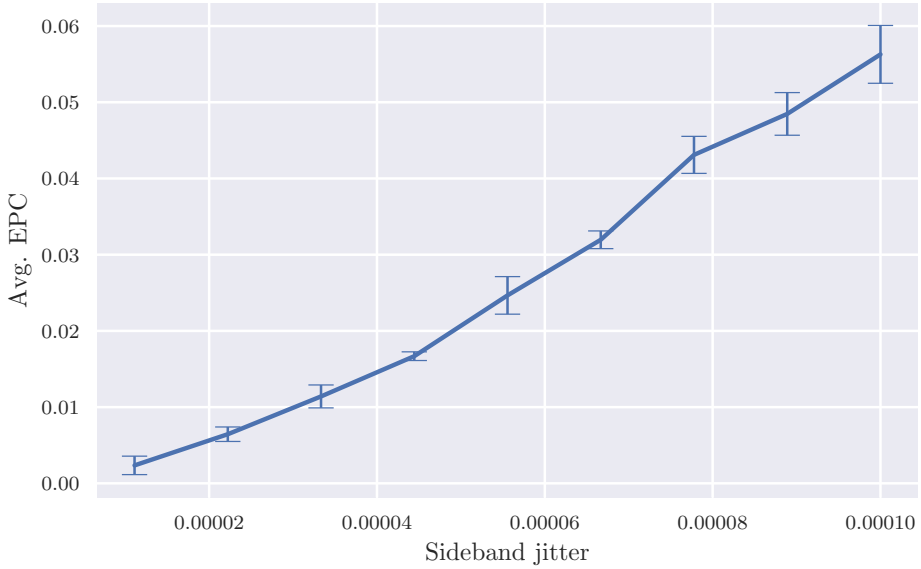


Figure 14.5: Adding a random jitter to the frequency of the sideband lasers can have a large negative effect on performance.

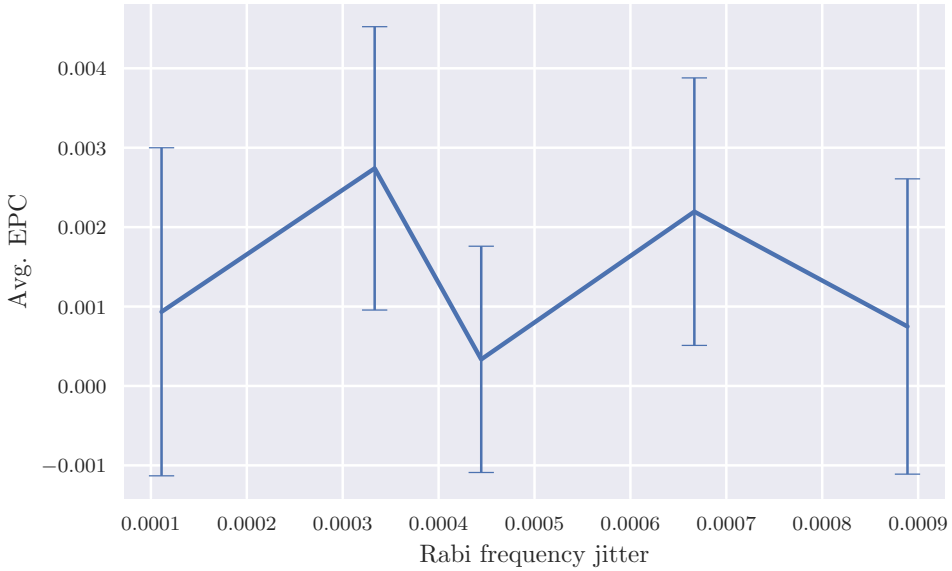


Figure 14.6: Whereas jitter on the set Rabi frequency (controlled via the amplitude of the Rabi beam) has less pronounced, and more random effect.

14.5.2.3 Carrier Frequency Resonance

As described in Section 14.3.2.3, we model a Dirac delta peak resonant with the qubit transition frequency during a Mølmer-Sørensen gate. The level of this noise is described as the height of the on-resonant peak, superimposed on the white noise of the driving laser field. Figure 14.7 shows how the height of this peak affects the average EPC.

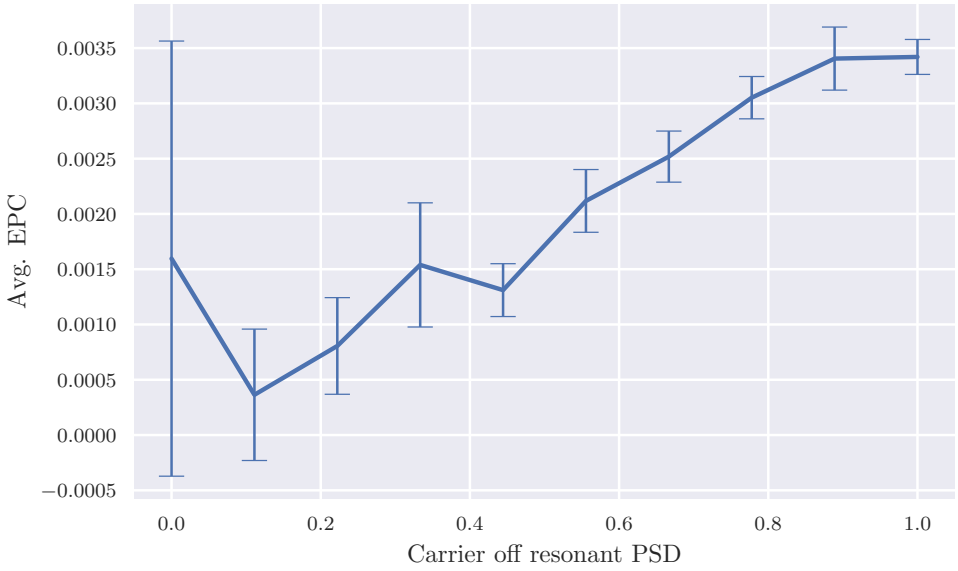


Figure 14.7: Increasing the spectral density of the laser resonant with the qubit transition increases the EPC of a set of randomised benchmarking experiments.

14.6 Discussion

Through theoretical modelling of the individual sources of noise in the ion trap at NPL, described in Section 14.1, we are able to numerically simulate the trap. Unfortunatley, the experimental device was unavailable at the time of writing, so we were unable to comapre to the target device. This builds upon work in [318], [321], [322] by incorporating Hamiltonians for ion trap devices into a single simulation, which was not available to NPL experimentalists before, and allows us to test how the reduction of common noise sources affects the overall computation.

One advantage of this numerical simulation is our ability to now identify the

sources of noise that have the largest effect on the metric that we actually care about, the error per Clifford given by a randomised benchmarking experiment. We only care about this as we will use some of the randomisation techniques described in Chapter 13 to ensure systematic errors are averaged into decoherence. We see from the scales of each of the simulations presented in Figures 14.2 - 14.7 that reducing sideband jitter, $\bar{n}$ will have the best effect on overall noise.

Chapter 15

Conclusions

At the conclusion of this thesis, we can reflect upon the state of quantum computing, and its prospects for the future. In the five years since this author enrolled in the UCL Quantum Technologies CDT we have seen quantum devices break out of universities and into private companies, the quantum supremacy result has been announced [22], and soon disputed [24], the NISQ-era has been defined [17] and we have developed our understanding of some of the problems with NISQ computation [159]. We have also seen developments in fault-tolerant quantum computation [94], [323] such that the surface code is no longer the default assumption. There have been so many developments that we cannot list them all here. That said, we still have not had a convincing demonstration of quantum advantage, we are yet to see a quantum computer outperform a classical computer in some relevant task.

This thesis has been split into parts to reflect three different focuses here. Firstly, in Part II we reviewed the state of quantum machine learning on NISQ devices, including demonstrations from financial applications. In Chapter 6 we introduced an algorithm for quantum state discrimination, and showed what modifications are needed when considering a noisy device. We simulated that noisy device, and showed that the learning algorithm can, in some cases, do better than calculating the ideal POVM and using a noisy device. In Chapter 7 we discuss proposals for the building block of quantum neural networks, a quantum version of the perceptron. We also try to clarify some of the discussion around non-linearities in QNNs, distinguishing between non-linearities that act on data encoded in a quantum state, and non-linearities

on the amplitude of a state. We then implement some of these algorithms numerically, showing that a non-linearity on the state amplitude requires a lot of resource for little training benefit. This implementation has been open-sourced [177] so that others can test proposals in the same framework.

Secondly, in Part III we discuss quantum algorithms for quantum chemistry. In Chapter 11 we demonstrate a quantum algorithm for the DMFT system, which has received much less attention than molecular systems, whilst still being an important algorithm for solid state physics, with implications for battery technology, and implementing the algorithm on real devices. In Chapter 12 we investigate increased system sizes for the DMFT algorithm, and how ansatze for these systems can be built with the ADAPT algorithm [41], [42]. We add a device aware component to the ADAPT algorithm which may have implications for systems larger than those studied here, but also gives evidence that a naive reduction in expressibility will not reduce the incidence of barren plateaus.

Finally, in Part IV we discuss in more detail how noise affects current quantum devices. In Chapter 13 we review some of the algorithmic techniques that are used to reduce noise in NISQ algorithms, and note that many techniques use randomisation to turn biased noise into depolarisation. In Chapter 14 we investigate in more detail some of the sources of noise in an ion-trap device, describing how the processes affect the qubits and gates in the device. We then use this to build a simulation of the device, allowing us to isolate the noise sources, suggesting directions for experimental optimisation.

Throughout this thesis we have seen how current quantum devices are limited by their scale, and the amount of noise in the device. There is still more work to do if we wish to achieve quantum advantage with a non-fault tolerant quantum computer, with one of the largest bottlenecks being the number of repetitions required to measure a Hamiltonian. If we consider a molecular Hamiltonian, the number of terms in the Hamiltonian scales as $\mathcal{O}(n^4)$, requiring a circuit repeat to measure each term, or a small group of terms. Techniques to reduce the number of terms, such as [219] can be implemented, and we can augment our quantum device with a classical neural

network [324] to reduce the number of measurements. Measuring all the terms in the Hamiltonian once is also not enough and this process must be repeated many times $\mathcal{O}\left(10^4\right)$ times due to statistical uncertainty, and even more repeats are required to account for the noise in the device. Leaving aside hardware improvements to be made, it seems like attacking these problems is where the most gains can be made for NISQ-era computers. As hardware improves, we must ensure that we are making the most of the hardware, e.g. the α-VQE algorithm that uses as much of the coherence time as possible to reduce the number of circuit repetitions. With all that said, we can approach the problem from the other direction, by improving the noise threshold, qubit count, and T-gate count of error corrected algorithms. This may take us longer to demonstrate quantum advantage, but the techniques developed will be used on ever larger quantum devices.

Quantum computing has developed so much since the 1990s, when only single qubits were developed experimentally. There is much further to go, but the benefits that we may reap, such as faster drug discovery, quantum-enhanced learning from experiments are so great that we must continue to build.

Appendices

We begin the appendices with Appendix A, which defines some of the mathematical concepts used throughout. Appendices B and C use diagrammatic notation to prove relations used in the section covering barren plateaus, Section 10.4. Appendix D shows how to project the two site, four qubit DMFT Hamiltonian onto only two qubits. Appendices F, G, H are used to derive relations used in simulating the noise in Ion Traps, Chapter 14.

Appendix A

Mathematical Definitions

In this appendix we will present some definitions which are used in derivations.

A.1 Norms

The p-norm of a vector $\mathbf{x}$ is given by:

$$||\mathbf{x}||_p := \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} \tag{A.1}$$

The 2-norm is the Eucledian norm, most commonly used, and is assumed to be the norm if not otherwise stated.

The infinity norm is the p-norm as $p \to \infty$, and is the maximum of the absolute value of all elements of $\mathbf{x}$.

$$||\mathbf{x}||_\infty := \max_i |x_i| \tag{A.2}$$

We can define a similar class of norms for quantum operators, the Schatten p-norms. The Schatten p-norms applies the p-norm to the singular values of a matrix.

$$||A||_p = \left(\sum_{i=1}^{\min\{m,n\}} \sigma_i^p(A) \right)^{\frac{1}{p}}, \tag{A.3}$$

where $\sigma_i(A)$ are the singular values of the matrix A. The singular values of a matrix are given by the diagonal entries of the diagonalisation of A:

$$A = U\Sigma V^T \tag{A.4}$$

where the diagonal entries $\sigma_i = \Sigma_{i,i}$ of Σ are the singular values. The $p = 1$ Shcatten norm is also known as the trace norm, the $p = 2$ norm can be called the Frobenius norm, and the $p = \infty$ is known as the spectral norm. The Frobenius norm can also be calculated by

$$||A||_F = \sqrt{\mathrm{Tr}\left[A^T \cdot A\right]} = \sqrt{\sum_{i,j=1}^{n} |a_i j|^2}, \tag{A.5}$$

and this is assumed to be the norm of any operators or matrices, unless otherwise stated. The Frobenius norm satisfies some properties, for a unitary operator, U,

$$||A||_F = ||AU||_F = ||UA||_F \,, \tag{A.6}$$

and

$$||A^* A||_F = ||AA^*||_F \leq \left(||A||_F\right)^2 \tag{A.7}$$

and

$$||A+B||_F^2 = ||A||_F^2 + ||B||_F^2 + 2\langle A, B\rangle_F, \tag{A.8}$$

which are inequalities we will review next.

The trace norm can also be given by

$$||A||_1 = \mathrm{Tr}\left[\sqrt{A^* A}\right]. \tag{A.9}$$

The diamond norm is the trace norm for an operator, maximised over all possible inputs which have trace norm ≤ 1.

$$||\phi||_\diamond := \max_{X;||X||_1 \leq 1} ||(\phi \otimes \mathbb{1}_n) X||_1 \tag{A.10}$$

The diamond norm is used in the definition of the diamond distance, which quantifies how close two completely positive, trace non-increasing maps, given by $\mathcal{E}, \mathcal{F}$ are:

$$d_\diamond(\mathcal{E}, \mathcal{F}) := ||\mathcal{E} - \mathcal{F}||_\diamond - \max_\rho ||(\mathcal{E} \otimes \mathbb{1}_n)\rho - (\mathcal{F} \otimes \mathbb{1}_n)\rho||_1 \tag{A.11}$$

and is taken over all density matrices of dimension n^2.

The max norm of an operator is the maximum absolute value of all elements:

$$||A||_{\max} = \max_{ij} \left|a_{i,j}\right|. \tag{A.12}$$

A.2 Inequalities

The triangle inequality, which we can use when proving lower bounds, states that the norm of the sum of two vectors is less than or equal to the sum of their norms:

$$||\mathbf{x}+\mathbf{y}|| \leq ||\mathbf{x}|| + ||\mathbf{y}||. \tag{A.13}$$

The Triangle inequality can be seen as a consequence of geometry (i.e. a triangle), but it is in fact a consequence of the more general Cauchy-Schwarz inequality. The Cauchy-Schwarz inequality, which is true for all vectors, $\mathbf{u}, \mathbf{v}$ in an inner product space,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle. \tag{A.14}$$

Here $\langle \cdot, \cdot \rangle$ is the inner product. In quantum mechanics, we usually replace this definition of the inner product by bra-ket notation, $\langle \psi, \phi \rangle \rightarrow \langle \psi \rangle \phi$. Every inner product defines a norm,

$$||\psi|| = \sqrt{\langle \psi \rangle \psi}, \tag{A.15}$$

and if the vector space, H on which this is defined is complete with respect to this norm we call it a Hilbert space. A Cauchy sequence is a sequence, of e.g. vectors, where the elements of the sequence become closer together as the sequence progresses. The vector space is complete if the Cauchy sequence of vectors converges to a limit within the vector space.

Finally, an inequality related probability rather than geometry is Chebyshev's inequality. Chebyshev's inequality, defined for many probability distributions, guarantees that a portion of the total values lie within a distance, defined by the standard

deviation, close to the mean,

$$P\left(|X-\mu| \geq k\sigma\right) \leq \frac{1}{k^2}. \tag{A.16}$$

Here, X is the random variable we are measuring, it has an expected value, μ, and non-zero variance, σ^2. Defining $k > 0$, a real number, we can choose this to describe our distribution, e.g. $k = \sqrt{2}$ guarantees the probability that a value lies outside the range $\left(\mu - \sqrt{2}, \mu + \sqrt{2}\right)$ is no more than $\frac{1}{2}$.

Appendix B

Diagrammatic notation for linear algebra

In Section 10.4 regarding barren plateaus, we are presented with the Weingarten function [240] for integrating over the second moment of the Haar distribution:

$$\begin{aligned}\int d\mu(U) U_{i_1 j_1} U_{i_2 j_2} U^*_{i'_1 j'_1} U^*_{i'_2 j'_2} = \\ \frac{\delta_{i_1 i'_1}\delta_{i_2 i'_2}\delta_{j_1 j'_1}\delta_{j_2 j'_2} + \delta_{i_1 i'_2}\delta_{i_2 i'_1}\delta_{j_1 j'_2}\delta_{j_2 j'_1}}{N^2-1} \\ - \frac{\delta_{i_1 i'_1}\delta_{i_2 i'_2}\delta_{j_1 j'_2}\delta_{j_2 j'_1} + \delta_{i_1 i'_2}\delta_{i_2 i'_1}\delta_{j_1 j'_1}\delta_{j_2 j'_2}}{N(N^2-1)}.\end{aligned} \tag{B.1}$$

Here we will use diagrammatic notation to show how this reduces to Equation 10.72.

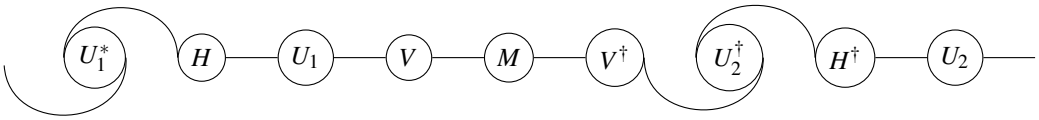
Diagrammatic notation [325]–[327] is a method of representing matrix manipulations as diagrams, in a way that can be much easier to parse, it is also closely related to the representation of tensor networks [325]. In diagrammatic notation the indices of a matrix are represented as the edges of a graph, and the matrices as nodes. The delta functions in Equation B.1 tell us to remove the nodes for U, U^* and which indices to connect. The equation for the variance, averaged over the Haar measure,

is given by:

$$\langle(\partial E)\rangle = \int dU\, p(U)\partial_k \left(\langle 0|U^\dagger(\theta)HU(\theta)|0\rangle\right)^2, \tag{B.2}$$

$$= \int dU\, p(U)U^\dagger HUVMV^\dagger U^\dagger H^\dagger U + \int dU\, p(U)VU^\dagger HUMU^\dagger H^\dagger UV$$
$$- \int dU\, p(U)VU^\dagger HUMA^\dagger U^\dagger H^\dagger U - \int dU\, p(U)U^\dagger HUVMU^\dagger H^\dagger UV^\dagger, \tag{B.3}$$

with $M = |0\rangle\langle 0|$. We can represent the first term in this equation as a diagram:

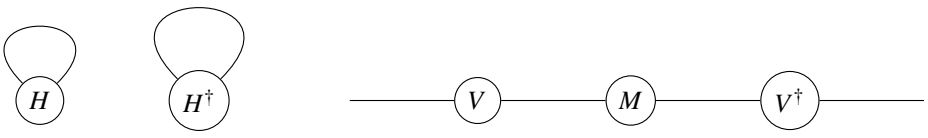


The delta functions in Equation B.1 translate to removing the U_1, U_1^*, U_2, U_2^* nodes and connecting:

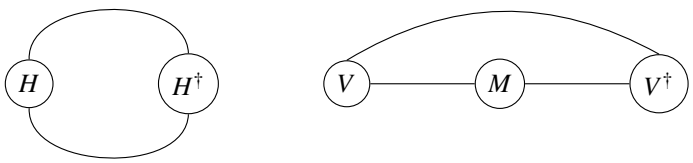
Table B.1: What each combination of delta function means in diagrammatic notation.

$\delta_{i_1 i_1'}\delta_{i_2 i_2'}\delta_{j_1 j_1'}\delta_{j_2 j_2'}$	Connect L $\leftrightarrow$ L and R $\leftrightarrow$ R of $(U_1, U_1^*), (U_2, U_2^*)$
$\delta_{i_1 i_2'}\delta_{i_2 i_1'}\delta_{j_1 j_2'}\delta_{j_2 j_1'}$	Connect L $\leftrightarrow$ L and R $\leftrightarrow$ R of $(U_1, U_2^*), (U_2, U_1^*)$
$\delta_{i_1 i_1'}\delta_{i_2 i_2'}\delta_{j_1 j_2'}\delta_{j_2 j_1'}$	Connect L $\leftrightarrow$ L of $(U_1, U_1^*), (U_2, U_2^*)$ and R $\leftrightarrow$ R of $(U_1, U_2^*)(U_2, U_1^*)$
$\delta_{i_1 i_2'}\delta_{i_2 i_1'}\delta_{j_1 j_1'}\delta_{j_2 j_2'}$	Connect R $\leftrightarrow$ R of $(U_1, U_1^*)(U_2, U_2^*)$ and L $\leftrightarrow$ L of $(U_1, U_2^*), (U_2, U_1^*)$

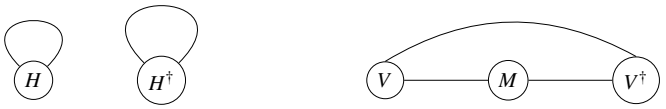
In our example this results in:



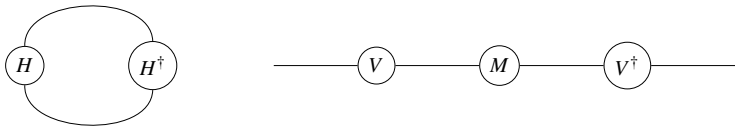
+



for the first term (multiplied by $\frac{1}{d^2-1}$), and



+



for the second term (multiplied by $\frac{1}{N\left(N^2-1\right)}$). We can now re-write these in the matrix form to get:

$$\int dU\, p(U) U^\dagger H U V M V^\dagger U^\dagger H^\dagger U = \tag{B.4}$$

$$= \frac{1}{N^2-1}\left(\mathrm{Tr}\left[HH^\dagger\right]\mathrm{Tr}\left[VMV^\dagger\right] + |\mathrm{Tr}\left[H\right]|^2 VMV^\dagger\right) \tag{B.5}$$

$$- \frac{1}{N\left(N^2-1\right)}\left(|\mathrm{Tr}\left[H\right]|^2 \mathrm{Tr}\left[VMV^\dagger\right] + \mathrm{Tr}\left[HH^\dagger\right] VMV^\dagger\right). \tag{B.6}$$

repeating this procedure for the other terms in Equation B.2 gives us the form of the variance we require:

$$\mathrm{Var}\left[\partial_k E\right] = \frac{1}{N^2-1}\langle\left(VV^\dagger - |\mathrm{Tr}\left[V\right]|^2\right)\left(\mathrm{Tr}\left[HH^\dagger\right] - \frac{1}{d}|\mathrm{Tr}\left[H\right]|^2\right)\rangle_{U_\pm} \tag{B.7}$$

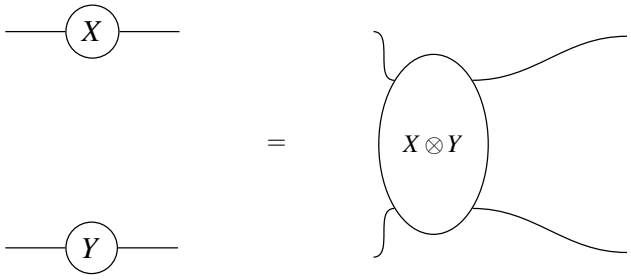
taking into account the expectation of $U_\pm$ over the Haar measure, we get the result in Equations 10.70 and 10.68.

Appendix C

Diagrammatic Notation for Expressibility

A problem in Section 10.4.5 can simirary be solved by diagrammatic notation, which we will do here. This is again related to integrating over the Haar measure, so the Weingarten function, Equation B.1 is again relevant, and the diagrammatic notation can be solved using the method outlined in Table B.1. As this is done for two terms with the same structure, we will solve in the general case, using matrices A and B of dimension d^2 and U which is the Haar volume element, and of dimension d.

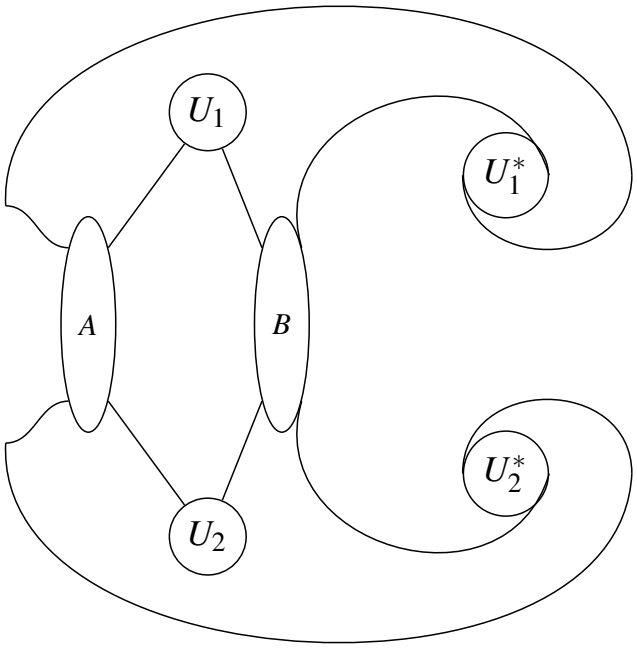
First we will show how a tensor product is done in diagrammatic notation:



We can now turn to the equation to solve,

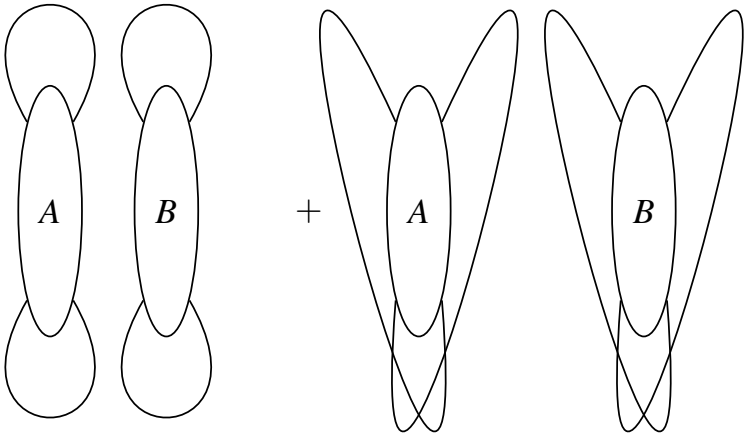
$$\int_{\mathcal{U}} d\mu(U) \mathrm{Tr}\left[AU^{\otimes 2} BU^{\dagger \otimes 2}\right], \tag{C.1}$$

writing in diagrammatic notation we have

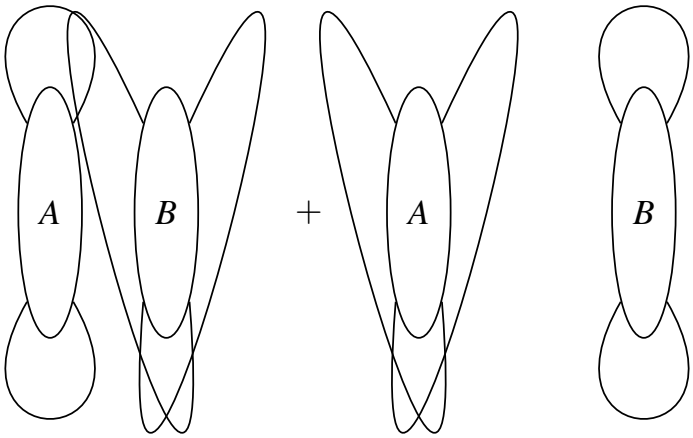


we now apply the contraction rules in Table B.1,

$$\frac{1}{(d^2-1)} \times (\text{L-L, R-R of } (U_1, U_1^*),\ (U_2, U_2^*)$$
$$+\text{L-L, R-R of } (U_1, U_2^*), (U_2, U_1^*)\ ) :$$



$$\frac{1}{d\,(d^2-1)}\,(\text{L-L of } (U_1, U_1^*), (U_2, U_2^*),\ \text{R-R of } (U_1, U_2^*), (U_2, U_1^*)$$
$$+\text{R-R of } (U_1, U_1^*), (U_2, U_2^*)\ \text{L-L of } (U_1, U_2^*), (U_2, U_1^*)\ )$$



To write the tensors where the indices have been swapped in matrix form, we introduce the swap operator, W that acts as $W|i\rangle|j\rangle = |j\rangle|i\rangle$, and has the Hilbert-Schmidt norm $||W||_2 = d$. We can now re-write these tensors in equation form, as used in Section 10.4.5,

$$\int_{\mathcal{U}(d)} d\mu(U) \mathrm{Tr}\left[AU^{\otimes 2}BU^{\dagger\otimes 2}\right] = \frac{1}{d^2-1}\left(\mathrm{Tr}[A]\,\mathrm{Tr}[B] + \mathrm{Tr}[AW]\,\mathrm{Tr}[BW]\right) - \frac{1}{d\,(d^2-1)}\left(\mathrm{Tr}[AW]\,\mathrm{Tr}[B] + \mathrm{Tr}[A]\,\mathrm{Tr}[BW]\right). \tag{C.2}$$

Appendix D

DMFT Circuit Reduction

In Chapter 11 we are considering the DMFT system with two sites and four qubits. Additionally, we restrict to the particle hole symmetric case, which allows us to make reductions in the number of quantities to be calculated. In this Appendix we will see how we can project the full four qubit Hamiltonian onto only two qubits, with separate Hamiltonians for each value of N.

The number operator is given by

$$\hat{N} = \hat{n}_1 + \hat{n}_2 + \hat{n}_3 + \hat{n}_4, \tag{D.1}$$

and the total z spin component,

$$\hat{S}_z = \hat{n}_1 + \hat{n}_1 - \hat{n}_3 - \hat{n}_4, \tag{D.2}$$

with $\hat{n}_\alpha = \hat{\sigma}_-^\alpha \hat{\sigma}_+^\alpha$, we can restrict the final wavefunction to a fixed number of electrons, and total z spin component. We can rewrite

$$\hat{n}_\alpha = \hat{\sigma}_-^\alpha \hat{\sigma}_+^\alpha = \frac{1}{2}(1 - \hat{\sigma}_z^\alpha), \tag{D.3}$$

substituting into $\hat{N}$ and $\hat{S}_z$:

$$\hat{N} = 2 - \frac{1}{2}\left(\hat{\sigma}_z^1 + \hat{\sigma}_z^2 + \hat{\sigma}_z^3 + \hat{\sigma}_z^4\right), \tag{D.4}$$

$$\hat{S}_z = \frac{1}{2}\left(-\hat{\sigma}_z^1 - \hat{\sigma}_z^2 + \hat{\sigma}_z^3 + \hat{\sigma}_z^4\right). \tag{D.5}$$

We can verify that the Hamiltonian in Equation 11.33 commutes with $\hat{N}$ and $\hat{S}_z$, $\left[\hat{H},\hat{N}\right] = \left[\hat{H},\hat{S}_z\right] = 0$. Therefore we can evaluate the quantities needed separately for given values of N and S_z, and use degeneracies to reduce the number of qubits required.

Starting with $N = 0$ we will obtain new operators which can fully characterise the system for each particle number (maximum for two sites is $N = 4$). The expectation value of an operator will be denoted by the removal of the hat, $\sigma_z^\alpha = \langle\hat{\sigma_z^\alpha}\rangle$ and we will use these values to form a basis: $|\sigma_z^1\sigma_z^2,\sigma_z^3,\sigma_z^4\rangle$.

When $N = 0$, from Eqn. D.4, $\sigma_z^1 + \sigma_z^2 + \sigma_z^3 + \sigma_z^4 = 4$, implying $\sigma_z^1 = \sigma_z^2 = \sigma_z^3 = \sigma_z^4 = 1$, and $S_z = 0$. Therefore there is only one possible state, $|1,1,1,1\rangle$ with the energy

$$E_{N=0,S_z=0} = \langle 1,1,1,1|\hat{H}|1,1,1,1\rangle = \mu - \frac{U}{4} - \varepsilon_2. \tag{D.6}$$

Similarly, when $N = 4$ we have $\sigma_z^1 + \sigma_z^2 + \sigma_z^3 + \sigma_z^4 = -4$, implying $\sigma_z^1 = \sigma_z^2 = \sigma_z^3 = \sigma_z^4 = -1$, and $S_z = 0$. The energy in this case is

$$E_{N=4,S_z=0} = \langle -1,-1,-1,-1|\hat{H}|-1,-1,-1,-1\rangle = -\mu + \frac{3U}{4} + \varepsilon_2. \tag{D.7}$$

For $N = 1$ we obtain $\sigma_z^3 + \sigma_z^4 = 2 - \sigma_z^1 - \sigma_z^2$, and $S_z = 1 - \sigma_z^1 - \sigma_z^2$. As $\left|\sigma_z^\alpha\right| = 1$, we have that $\sigma_z^3 + \sigma_z \leq 2$ and $0 \leq \sigma_z^1 + \sigma_z^2 \leq 2$, giving only two possible values for $S_z \in \{-1,1\}$. So $\hat{S}_z$ determines the spin imbalance of the system, and can by determined with the operator $\hat{\tilde{\sigma}}_z^1 = -\hat{S}_z$,

$$\hat{\tilde{\sigma}}_z^1 = \frac{1}{2}(\hat{\sigma}_z^1 + \hat{\sigma}_z^2 - \hat{\sigma}_z^3 - \hat{\sigma}_z^4). \tag{D.8}$$

Imbalance of the electron number over the two sites is

$$\hat{\tilde{\sigma}}_z^2 = \frac{1}{2}(\hat{\sigma}_z^1 - \hat{\sigma}_z^2 + \hat{\sigma}_z^3 - \hat{\sigma}_z^4). \tag{D.9}$$

The ladder raising operator of these can be thought of as a spin preserving hop between sites (swapping orbitals $3 \to 1, 4 \to 2$) for $\tilde{1}$, and a spin flip on-site (swapping

orbitals $2 \to 1, 4 \to 3$) for the $\tilde{2}$. These are the ladder operators

$$\hat{\tilde{\sigma}}^1_+ = \hat{\sigma}^3_- \hat{\sigma}^1_+ + \hat{\sigma}^4_- \hat{\sigma}^2_+, \tag{D.10}$$

$$\hat{\tilde{\sigma}}^2_+ = \hat{\sigma}^2_- \hat{\sigma}^1_+ + \hat{\sigma}^4_- \hat{\sigma}^3_+, \tag{D.11}$$

this can be solved for the $\hat{\tilde{\sigma}}_{x,y}$ operators, with the expansion of

$$\hat{\tilde{\sigma}}^2_x = \frac{1}{2}\left(\sigma^1_x \sigma^2_x + \sigma^1_y \sigma^2_y + \sigma^4_x \sigma^3_x + \sigma^3_y \sigma^4_y\right) \tag{D.12}$$

matching the coefficient of V in the Hamiltonian, Equation 11.33. These operators obey the canonical commutation relations when $N = 1$. Thinking of a system with $N = 1$ and a single electron on site 1 with spin up $\hat{\sigma}^1_z = -1$, we have that $S_z = 1 = -\hat{\tilde{\sigma}}^1_z$ and $\hat{\tilde{\sigma}}^2_z = -1$, giving the inverse projection

$$\hat{\sigma}^1_z = -\frac{1}{2}(1 - \hat{\tilde{\sigma}}^1_z)(1 - \hat{\tilde{\sigma}}^2_z) + 1, \tag{D.13}$$

which can be done for the other operators:

$$\hat{\sigma}^2_z = -\frac{1}{2}(1 - \hat{\tilde{\sigma}}^1_z)(1 + \hat{\tilde{\sigma}}^2_z) + 1, \tag{D.14}$$

$$\hat{\sigma}^3_z = -\frac{1}{2}(1 + \hat{\tilde{\sigma}}^1_z)(1 - \hat{\tilde{\sigma}}^2_z) + 1, \tag{D.15}$$

$$\hat{\sigma}^4_z = -\frac{1}{2}(1 + \hat{\tilde{\sigma}}^1_z)(1 + \hat{\tilde{\sigma}}^2_z) + 1. \tag{D.16}$$

These can be substituted into Equation 11.33 to obtain an expression which requires only two qubits

$$\hat{H} = \left(\frac{\mu}{2} + \frac{\varepsilon_2}{2}\right)\hat{\tilde{\sigma}}^2_z + \hat{V\sigma}^2_x + \left(\frac{\mu}{2} - \frac{U}{4} - \frac{\varepsilon_2}{2}\right). \tag{D.17}$$

$N = 3$ When $N = 3$, we can follow a similar mapping to the $\hat{\tilde{\sigma}}$ operators, as S_z can again only take the values $\{-1, 1\}$. However, for the inverse mapping we must re-think the conditions. When $N = 3$, with no electron on site 1 with spin up ($\hat{\sigma}^1_z = 1$) we have $S_z = -1 = -\hat{\tilde{\sigma}}^1_z$, and $\hat{\tilde{\sigma}}^2_z = 1$ allowing us to construct the inverse

projection

$$\hat{\sigma}_z^1 = \frac{1}{2}(1+\hat{\tilde{\sigma}}_z^1)(1+\hat{\tilde{\sigma}}_z^2) - 1, \tag{D.18}$$

$$\hat{\sigma}_z^2 = \frac{1}{2}(1+\hat{\tilde{\sigma}}_z^1)(1-\hat{\tilde{\sigma}}_z^2) - 1, \tag{D.19}$$

$$\hat{\sigma}_z^3 = -\frac{1}{2}(1-\hat{\tilde{\sigma}}_z^1)(1+\hat{\tilde{\sigma}}_z^2) - 1, \tag{D.20}$$

$$\hat{\sigma}_z^4 = -\frac{1}{2}(1-\hat{\tilde{\sigma}}_z^1)(1-\hat{\tilde{\sigma}}_z^2) - 1. \tag{D.21}$$

This can be substituted into Eqn. 11.33 for the two-qubit Hamiltonian

$$\hat{H} = \left(\frac{\mu}{2} + \frac{\varepsilon_2}{2} - \frac{U}{2}\right)\hat{\tilde{\sigma}}_z^2 + V\hat{\sigma}_x^2 - \left(\frac{\mu}{2} - \frac{U}{4} - \frac{\varepsilon_2}{2}\right). \tag{D.22}$$

Finally, for $N = 2$, from Eqn. D.1 we require $\sigma_z^3 + \sigma_z^4 = -\sigma_z^1 - \sigma_z^2$, and that $S_z = -\sigma_z^1 - \sigma_z^2$, giving $S_z \in \{-2, 0, 2\}$. For $S_z = -2$ there is only one possible state, $\sigma_z^1 = \sigma_z^2 = -1$, with $E_{N=2,S_z=2} = -\frac{U}{4}$. Similarly, for $S_z = 2$, $\sigma_z^1 = \sigma_z^2 = 1$ and $E_{N=2,S_z=-2} = -\frac{U}{4}$.

When $S_z = 0$, we have $\sigma_z^2 = \sigma_z^1$, which implies $\sigma_z^4 = -\sigma_z^3$. This restriction in the degrees of freedom allows us to represent this system with two qubits also. We set the z components of the new qubits 1 and 2 to that of the original qubits 1 and 3, and the negation of this is the z component of the original qubits 2 and 4. The x operators of the new qubits need to ensure that flipping the values of qubits 1 and 3

simultaneously flips the values of qubits 2 and 4. The new operators are:

$$\hat{\tilde{\sigma}}_z^1 = \hat{\sigma}_z^1, \tag{D.23}$$

$$\hat{\tilde{\sigma}}_z^2 = \hat{\sigma}_z^3, \tag{D.24}$$

$$\hat{\tilde{\sigma}}_z^3 = -\hat{\sigma}_z^2, \tag{D.25}$$

$$\hat{\tilde{\sigma}}_z^4 = -\hat{\sigma}_z^4, \tag{D.26}$$

$$\hat{\tilde{\sigma}}_x^1 = \hat{\sigma}_x^1 \hat{\sigma}_x^2, \tag{D.27}$$

$$\hat{\tilde{\sigma}}_x^2 = \hat{\sigma}_x^3 \hat{\sigma}_x^4, \tag{D.28}$$

$$\hat{\tilde{\sigma}}_y^1 = \hat{\sigma}_y^1 \hat{\sigma}_x^2, \tag{D.29}$$

$$\hat{\tilde{\sigma}}_y^2 = \hat{\sigma}_y^3 \hat{\sigma}_x^4. \tag{D.30}$$

Substituting these operators into Eqn. 11.33 gives us the Hamiltonian

$$\hat{H} = \frac{U}{4}\hat{\sigma}_z^1 \hat{\sigma}_z^2 + \left(\frac{\mu}{2} - \frac{U}{4} + \frac{\varepsilon_2}{2}\right)(\hat{\sigma}_z^1 + \hat{\sigma}_z^2) + V(\hat{\sigma}_x^1 + \hat{\sigma}_x^2). \tag{D.31}$$

We must complete a similar procedure for the values of $\lambda_{\mathrm{p}(h),\alpha,n}$, which for 2 site DMFT $\alpha = 1$. In the full, four qubit case

$$\lambda_{\mathrm{h},1,n} = \left| \langle 0 | \hat{U}_{2,0}^\dagger \hat{\sigma}_-^1 \hat{U}_{1,n} | 0 \rangle \right|^2. \tag{D.32}$$

Using the substitutions derived above, we have for the one electron case, applying $\hat{\tilde{\sigma}}_-$ gives us the two electron case with $S_z = 0$. This can be evaluated on a two qubit quantum circuit with:

$$\lambda_{\mathrm{h},1,n} = \left| \langle \tilde{0} | \hat{\tilde{U}}_{2,0}^\dagger \hat{\tilde{\sigma}}_-^1 \hat{\tilde{U}}_{1,n} | \tilde{0} \rangle \right|^2, \tag{D.33}$$

where the tilde indicates a two-qubit operator. As we have $S_z = 0$, for non-zero $\lambda_{\mathrm{h},1,n}$ we require S_z for the single electron case. This implies $\tilde{\sigma}_z^1 = 1$, and that for $\hat{U}_{1,n}|0\rangle$ with $\tilde{\sigma}_z^1 = -1$, $\lambda_{\mathrm{h},1,n} = 0$. This means we can safely ignore the $\hat{\tilde{\sigma}}_y^1$ component of $\hat{\tilde{\sigma}}_-^1$, so that

$$\lambda_{\mathrm{h},1,n} = \left| \langle \tilde{0} | \hat{\tilde{U}}_{2,0}^\dagger \hat{\tilde{\sigma}}_x^1 \hat{\tilde{U}}_{1,n} | \tilde{0} \rangle \right|^2, \tag{D.34}$$

For the particle elements can also be obtained this way, given by

$$\lambda_{\mathrm{p},1,n} = \left| \langle \tilde{0} | \hat{\tilde{U}}^{\dagger}_{2,0} \hat{\tilde{\sigma}}^{1}_{+} \hat{\tilde{U}}_{1,n} | \tilde{0} \rangle \right|^2, \tag{D.35}$$

When removing an electron from the three electron state, we go to the two electron ground sate with $S_z = 0$, for non-zero matrix elements we require $S_z = 1$ in the three electron state. This implies $\tilde{\sigma}^1_z = -1$ in the three electron state. Therefore, states $\hat{U}_{3,n} |\tilde{0}\rangle$ where $\tilde{\sigma}^1_z = 1$ have $\lambda_{\mathrm{p},1,n} = 0$, and we can again ignore the $\hat{\tilde{\sigma}}^1_y$ component, giving

$$\lambda_{\mathrm{p},1,n} = \left| \langle \tilde{0} | \hat{\tilde{U}}^{\dagger}_{2,0} \hat{\tilde{\sigma}}^{1}_{x} \hat{\tilde{U}}_{3,n} | \tilde{0} \rangle \right|^2. \tag{D.36}$$

Using the projections here, we are able to calculate the two-site DMFT system on only two qubits, using the ansatze given in Figure 11.3.

Appendix E

Additional Variance results for DMFT

In Chapter 11 we discussed some selected results from measuring the variance of the parameter gradients as the ADAPT algorithm proceeded. The systems and figures are:

System	Original ADAPT	Device Aware ADAPT
H_4 chain	Figure E.2	Figure E.1
LiH	Figure E.4	Figure E.3
3-site DMFT	Figure E.6	Figure E.5
4-site DMFT	Figures E.9, E.10	Figures E.7, E.8

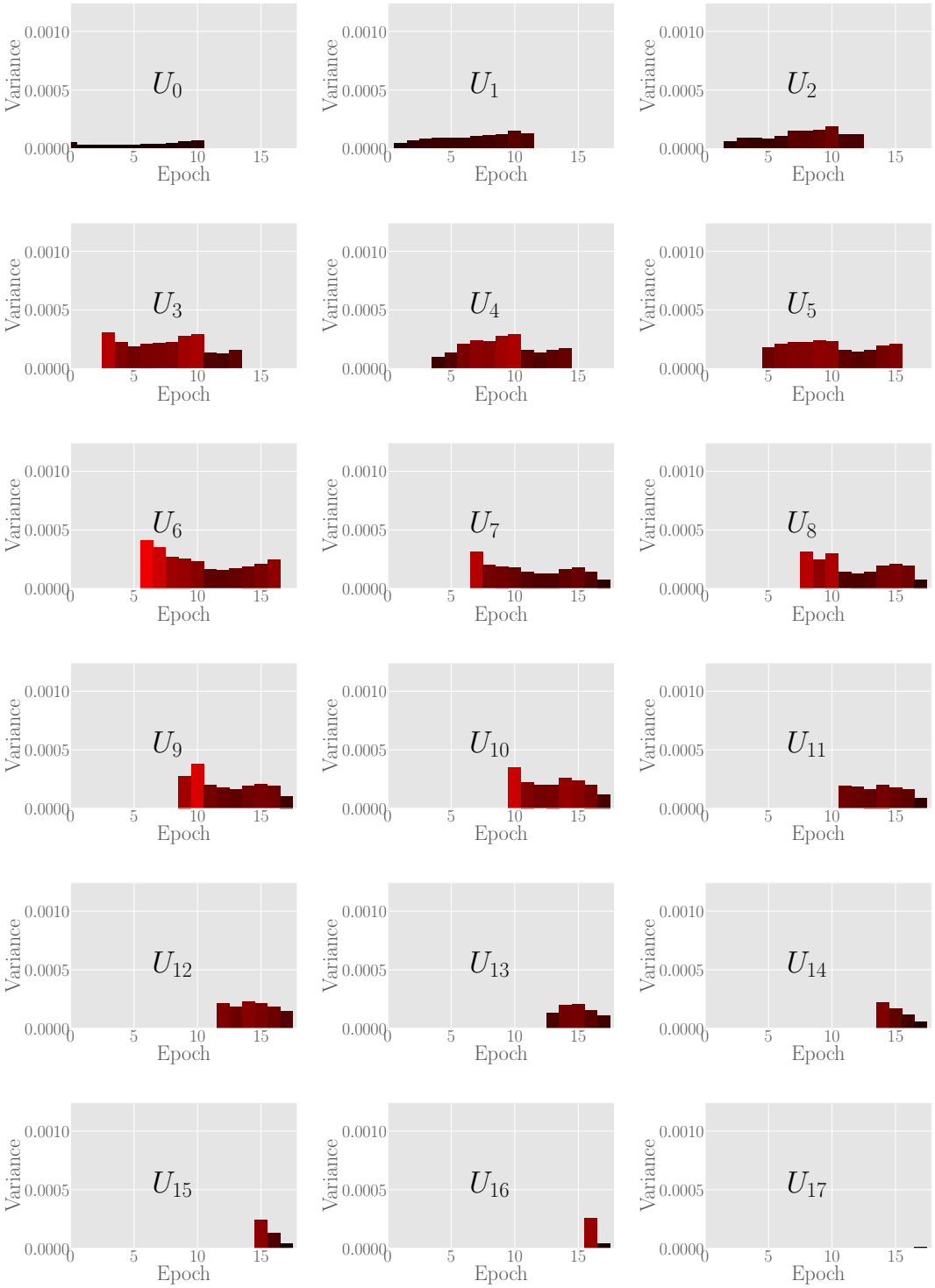


Figure E.1: H_4 system, device aware ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. As the y axis is shared between this and the other variant of the algorithm, we see that this variant of the algorithm did not contain the largest variance. In contrast to the results of [215] we find that the larges variances appear in the middle of the circuit. This could be due to the fermionic nature of the operators, meaning the circuits are no longer 2-designs.

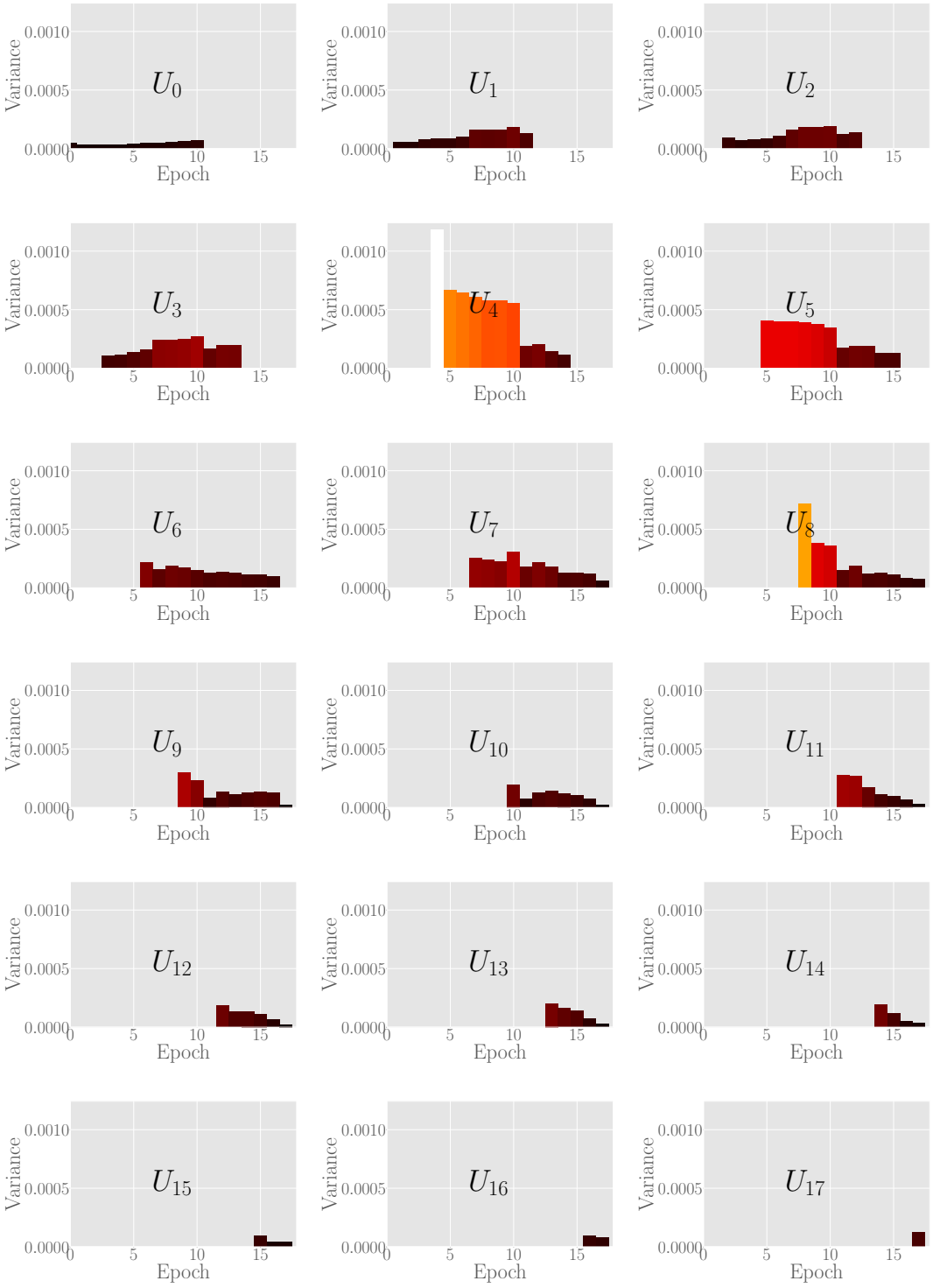


Figure E.2: H_4 system, original ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. This version of the algorithm has the highest variances by far, and is consistently high in the midddle of the circuit, contrasting [215]. This could be due to the fermionic nature of the operators, meaning the circuits are no longer 2-designs.

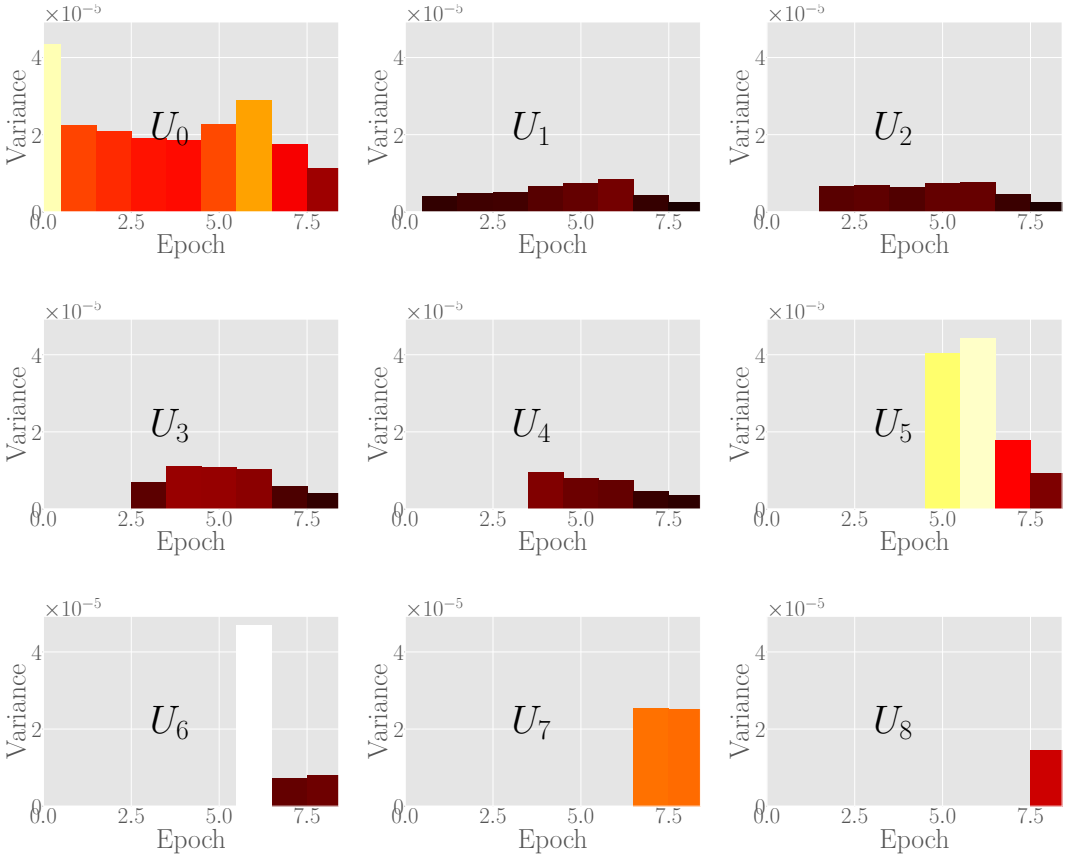


Figure E.3: LiH system, device aware ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. This version of the algorithm has the highest variances consistently, including the middle of the circuits, contrasting [215]. This could be due to the fermionic nature of the operators, meaning the circuits are no longer 2-designs.

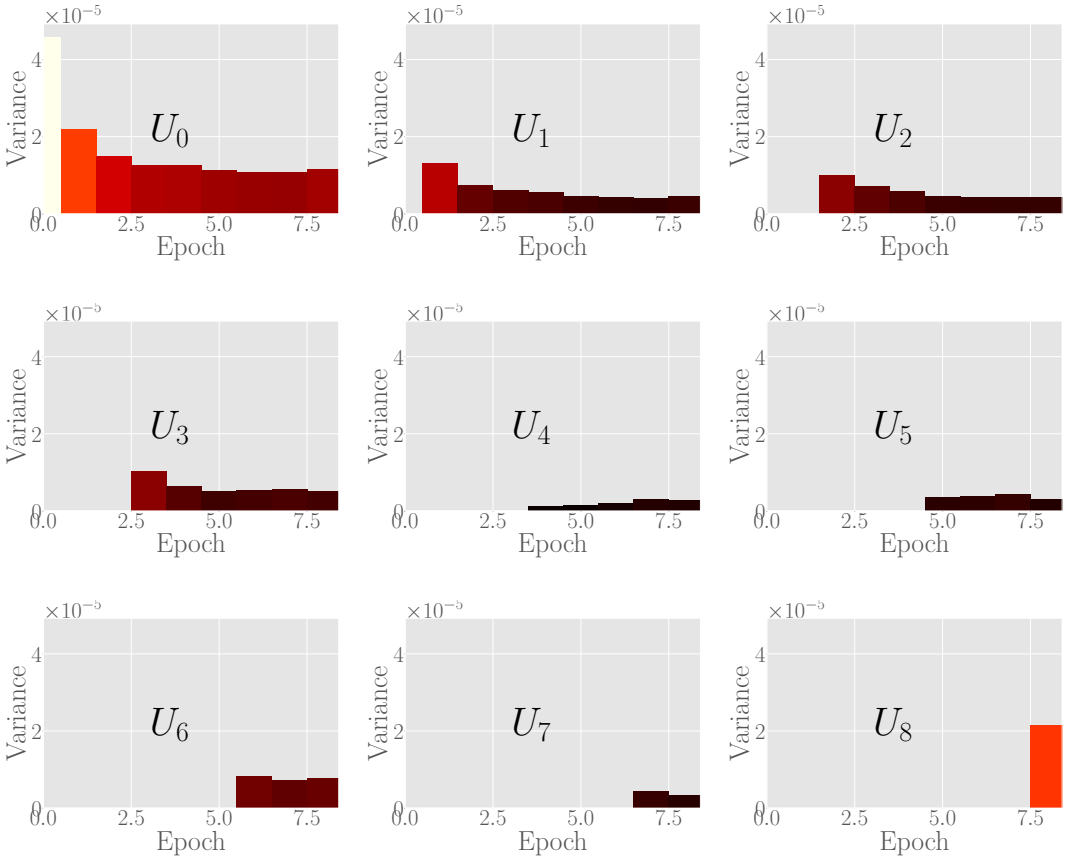


Figure E.4: LiH system, original ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. This version of the algorithm has the highest variance at the beginning of the circuit, but is usually lower than the device aware version. Unlike the other molecular systems here, we see that the lowest variances are in the middle of the circuit, consistent with the predictions in [215].

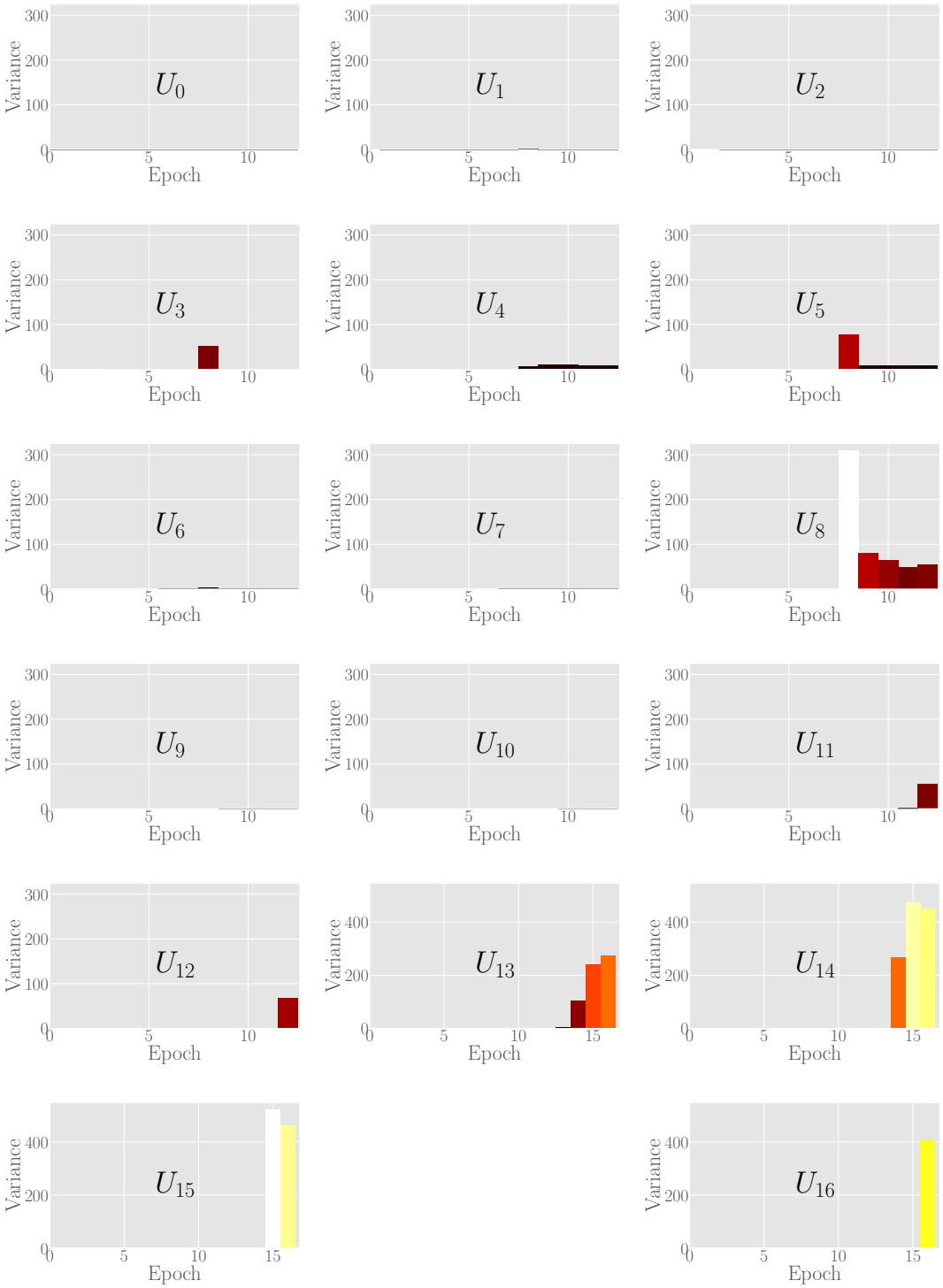


Figure E.5: Three site DMFT system, device aware ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. We see by the white bar that the largest variance between this and the original ADAPT algorithm came in the device aware variant. We also see that the variance is much lower in the middle of the circuit, which is consistent with the findings in [215].

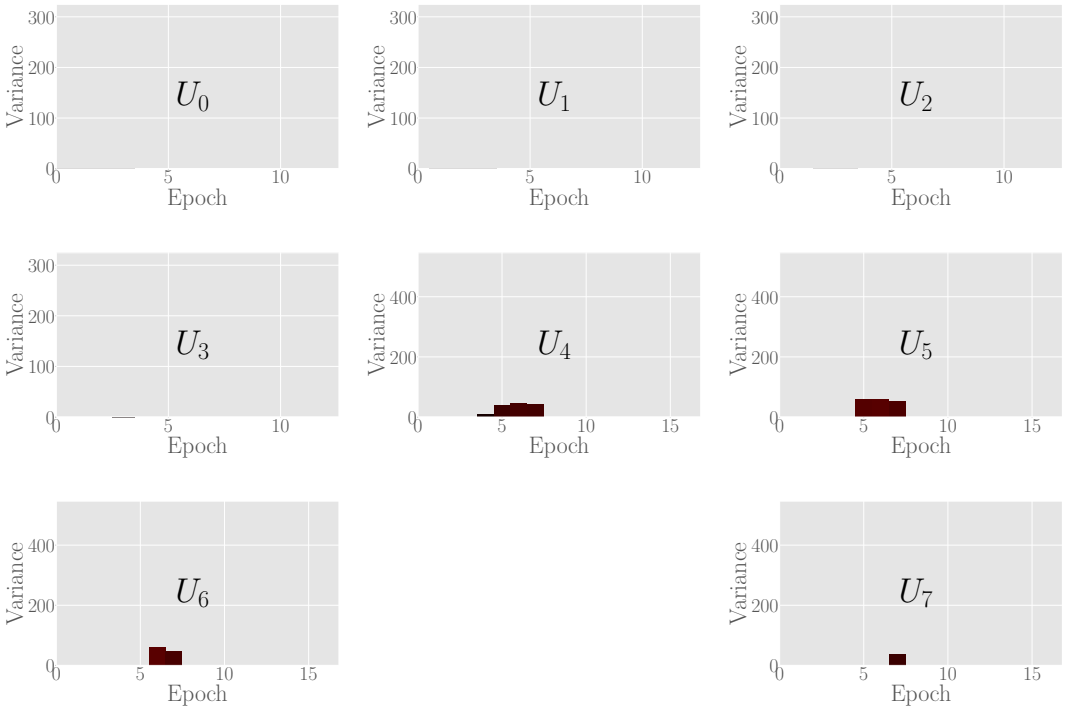


Figure E.6: Three site DMFT system, original ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. As the y-axis is shared between this and the device aware algorithm, we see that in this version we do not record as high variance for the parameters. We also see that the variance is much lower in the middle of the circuit, which is consistent with the findings in [215].

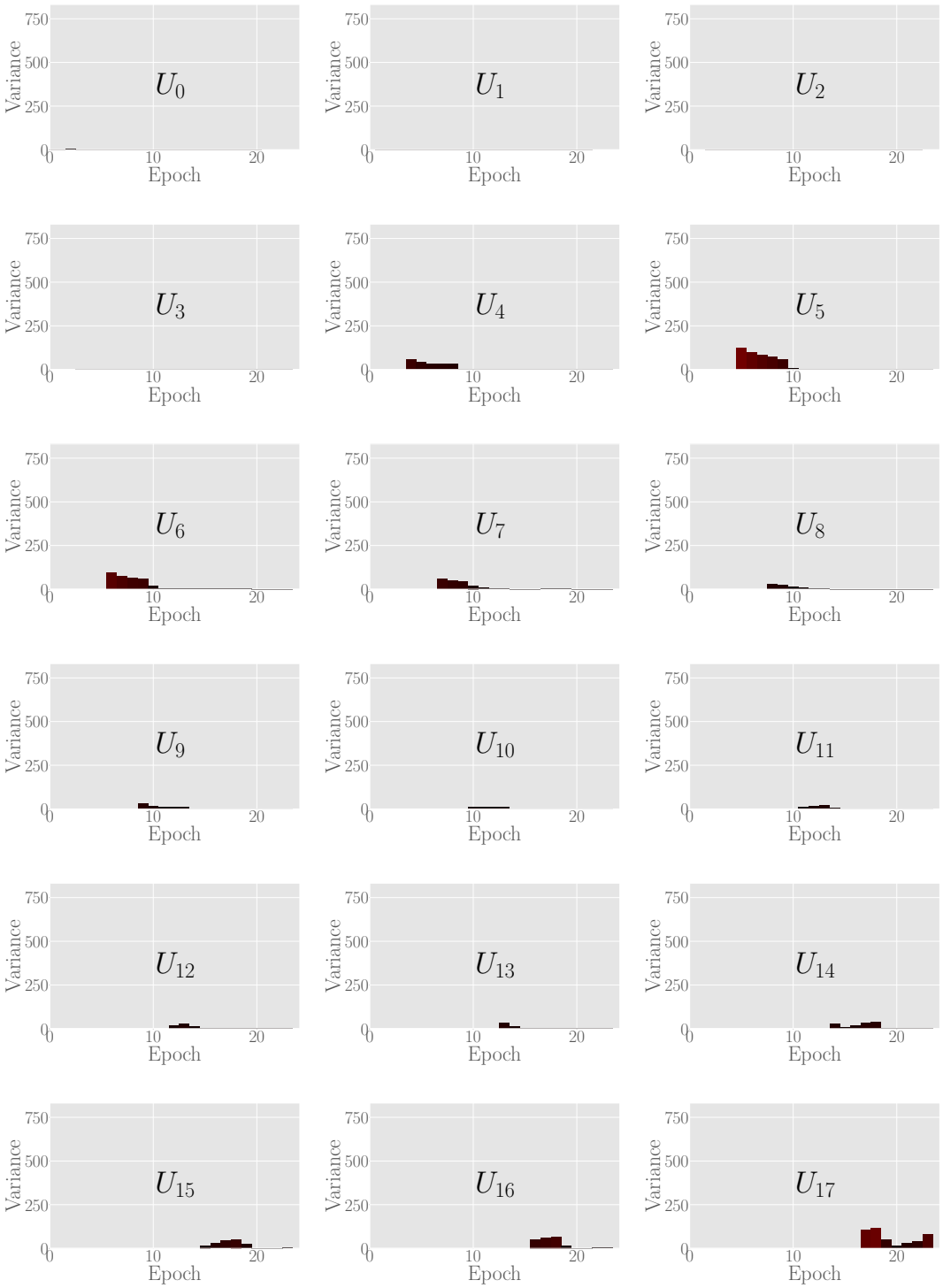


Figure E.7: Four site DMFT system, device aware ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter, first page of graphs.

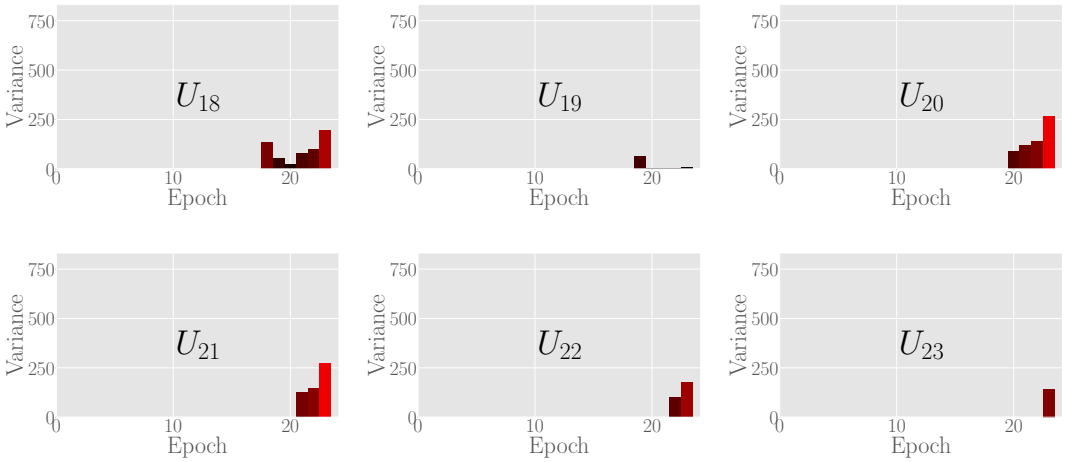


Figure E.8: Four site DMFT system, device aware ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. As the y-axis is shared between this and the device aware algorithm, we see that in this version we do not record as high variance for the parameters. We also see that the variance is much lower in the middle of the circuit, which is consistent with the findings in [215].

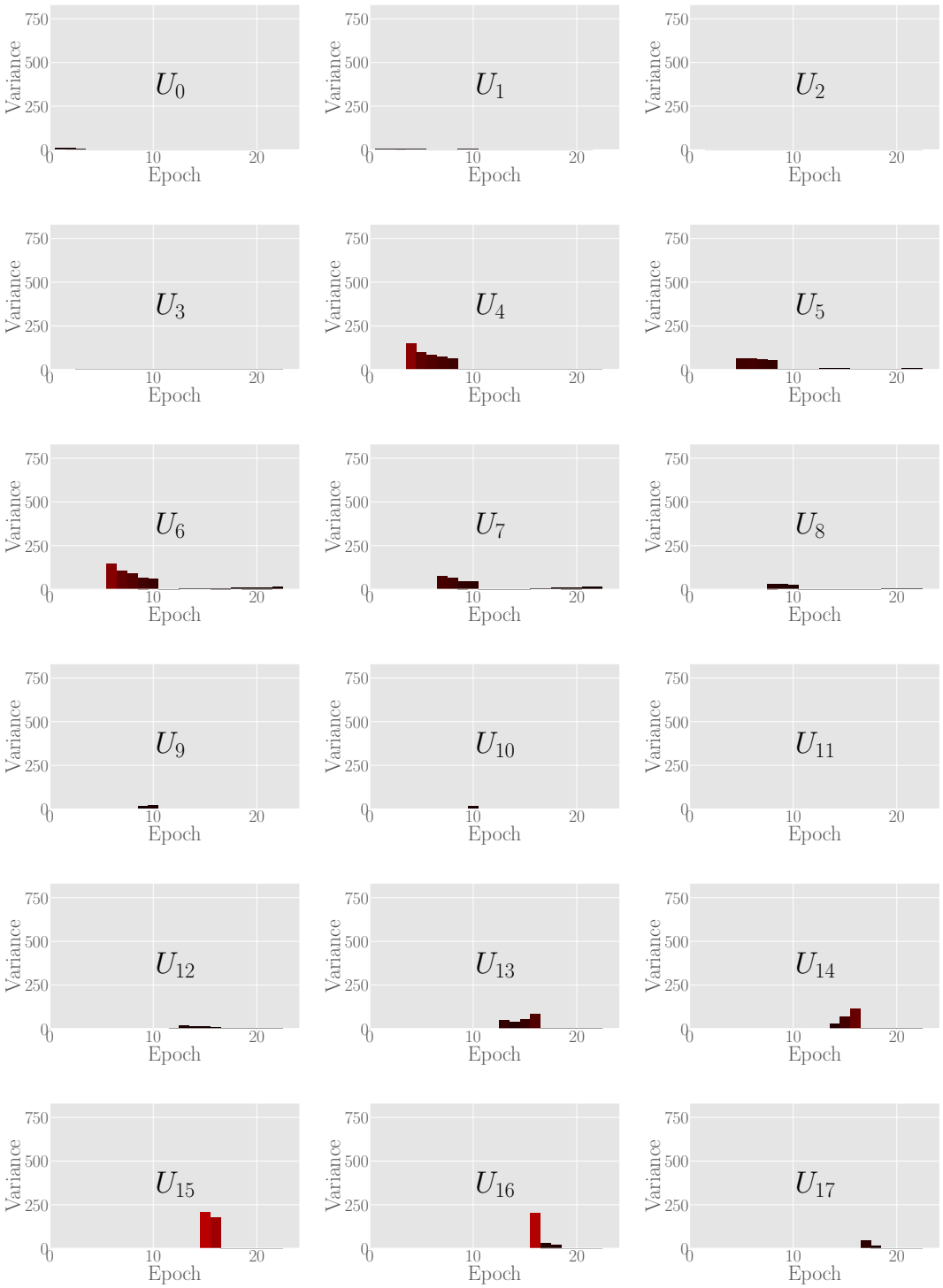


Figure E.9: Four site DMFT system, original ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter, first page of graphs.

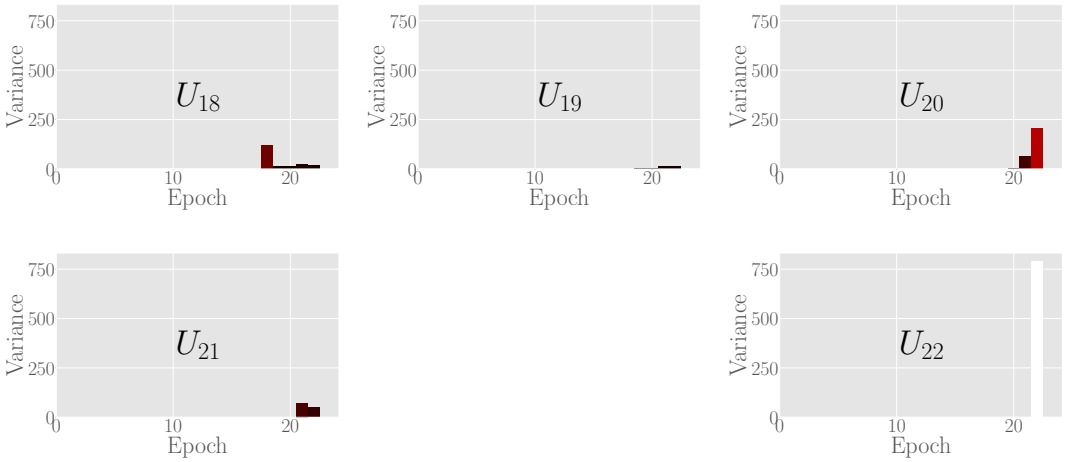


Figure E.10: Four site DMFT system, original ADAPT. Average over all runs of the variance of the gradient of the cost function for each parameter. The tall white bar shows us that the highest variance came in this variant of the algorithm, at the very end. We also see that the variance is much lower in the middle of the circuit, which is consistent with the findings in [215].

Appendix F

Expanding Exponentiated Hamiltonians

To derive the first relation, we must use trigonometric identities and make some substitutions for ease. Let $\omega_0 t/2 := \theta$, and we are dealing only with the operators acting on the internal states of the ion:

$$\exp[i\theta\sigma_z](\sigma_+ + \sigma_-)\exp[-i\theta\sigma_z] \tag{F.1}$$

First, recall the expansion of the Matrix exponential when using Pauli operators:

$$\begin{aligned}\exp[i\theta\sigma_z] &= \mathbf{1} + i\theta\sigma_z - \frac{\theta^2\mathbf{1}}{2!} - i\frac{\theta^3\sigma_z}{3!}\ldots \\ &= \mathbf{1}\cos\theta + i\sigma_z\sin\theta \qquad \text{(F.2)}\\ \exp[-i\theta\sigma_z] &= \mathbf{1}\cos\theta - i\sigma_z\sin\theta \qquad \text{(F.3)}\end{aligned}$$

And so expand the interaction operation:

$$\begin{aligned}\exp[i\theta\sigma_z]\sigma_\pm\exp[i\theta\sigma_z] &= (\mathbf{1}\cos\theta - i\sigma_z\sin\theta)\sigma_\pm(\mathbf{1}\cos\theta + i\sigma_z\sin\theta) \qquad \text{(F.4)}\\ &= \sigma_\pm\cos^2\theta + \sigma_z\sigma_\pm\sigma_z\sin^2\theta + i\sin(2\theta)\frac{1}{2}(\sigma_\pm\sigma_z - \sigma_z\sigma_\pm) \qquad \text{(F.5)}\end{aligned}$$

Finding expressions for the Pauli matrices:

$$\begin{aligned}\sigma_z\sigma_\pm\sigma_z &= \frac{1}{2}(-\sigma_x \mp i\sigma_y) = -\sigma_\pm \\ &= \sigma_\pm\sigma_z - \sigma_z\sigma_\pm = \frac{1}{2}(-i\sigma_y \mp \sigma_x - i\sigma_y \mp \sigma_x) = (-i\sigma_y \mp \sigma_x) \end{aligned} \tag{F.6}$$

And using the trigonometric identities $\sin^2\theta = \frac{1}{2}(1-\cos(2\theta))$ and $\cos^2\theta = \frac{1}{2}(1+\cos(2\theta))$

$$\sigma_\pm\cos^2\theta - \sigma_\pm\sin^2\theta = \sigma_\pm\cos(2\theta) \tag{F.7}$$

So we arrive at:

$$\exp[i\theta\sigma_z]\sigma_\pm\exp[i\theta\sigma_z] = \sigma_\pm\cos(2\theta) \mp i\sigma_\pm\sin(2\theta) \tag{F.8}$$

Which when split into the case for each $\sigma_\pm$ gives:

$$\exp[i\theta\sigma_z]\sigma_+\exp[i\theta\sigma_z] = \sigma_+\cos(2\theta) - i\sigma_+\sin(2\theta) \tag{F.9}$$

$$= e^{2i\theta}\sigma_+\exp[i\theta\sigma_z]\sigma_-\exp[i\theta\sigma_z] \tag{F.10}$$

$$= \sigma_-\cos(2\theta) + i\sigma_+\sin(2\theta) \tag{F.11}$$

$$= e^{-2i\theta}\sigma_- \tag{F.12}$$

Appendix G

Signal Processing and Noise

We will use concepts from signal processing [328] to describe the noise in the driving field - as the desired electromagnetic laser field can be thought of as a perfect signal. The autocorrelation of a signal defines how much information we infer about a signal at time t_2, given the value at time t_1:

$$G(t_1,t_2) = \langle V(t_1)V(t_2)\rangle \tag{G.1}$$

which we can also define in terms of the difference between times t_1 and t_2, τ:

$$G(\tau) = \langle V(t_1)V(t_1+\tau)\rangle\,, \tag{G.2}$$

$$= \lim_{T\to 0}\frac{1}{T}\int_0^T V(t+\tau)V(t)dt. \tag{G.3}$$

A Markov signal is defined as a stochastic signal with zero mean, where the autocorrelation reduces exponentially:

$$G(t_1,t_2) = \frac{e^{-|t_1-t_2|}\tau'}{\tau'} \tag{G.4}$$

where the signals are only correlated for short times, τ'; when τ' approaches 0, we have white noise, and the autocorrelation function is described by a Dirac delta function:

$$G(t_1,t_2) = \sigma^2\delta(t_1-t_2), \tag{G.5}$$

where σ is the variance of the signal, and δ is the Dirac delta.

The Wiener-Khinchin theorem states that the autocorrelation and power spectral density are a Fourier Transform pair:

$$\mathrm{FT}(G(\tau)) = \mathrm{FT}\left(\int_{-\infty}^{\infty} V(\tau)V(\tau-t)d\tau\right) \tag{G.6}$$

$$= \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} V(\tau)V(\tau-t)d\tau\right)e^{-i\omega t}dt \tag{G.7}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} V(\tau)V(T)e^{-i\omega\tau-T}dTd\tau \tag{G.8}$$

$$= \int_{-\infty}^{\infty} V(T)e^{i\omega T}\int_{-\infty}^{\infty} V(\tau)e^{-i\omega\tau}d\tau \tag{G.9}$$

$$= S^{*}(\omega)S(\omega) = |S(\omega)|^{2} \tag{G.10}$$

Where $S(\omega)$ is the Power Spectral Density (PSD). When we are considering white noise, we can again reduce this to the Dirac delta function:

$$G(\tau) = S_0\delta(\tau) \tag{G.11}$$

Where S_0 is the PSD of the white noise.

Appendix H

White Noise in the Liouville von-Neumann Equation

Consider a generic Hamiltonian, where one process is subject to a noise term, ξ:

$$H(t) = H_1(t) + \xi(t)H_2(t) \tag{H.1}$$

As ξ describes some noise term, it is real-valued with zero mean. We can describe the evolution of the system using the Liouville von-Neumann equation:

$$d\rho = [H_1, \rho]dt + \xi(t)[H_2, \rho]dt, \tag{H.2}$$

H.1 White Noise

If there are no time correlations in the noise process it is described as white noise, i.e. a flat spectrum with a very wide bandwidth. It is completely characterised by its autocorrelation function:

$$G(\tau) = \langle \xi(\tau)\xi(0) \rangle = 0(\tau \neq 0) \tag{H.3}$$

$$\int_{-\infty}^{\infty} \langle \xi(\tau)\xi(0) \rangle \, d\tau = \Gamma. \tag{H.4}$$

We can characterise $\xi(t)dt$ as the increment of a Wiener process [321], so that $\xi(\tau)dt \to \sqrt{\Gamma}dW(t)$, giving us the Stratonovich Stochastic Differential Equation

(SDE) [329]:

$$i\hbar d\rho = [H_1, \rho]dt + \sqrt{\Gamma}[H_2, \rho] \circ dW. \tag{H.5}$$

In this form, $\rho(t)$ and $dW(t)$ are not statistically independent, so we cannot safely average over them to get a mean evolution. To do so, we must convert this into the Ito form.

First, we recall the Stratonovich integral:

$$\mathrm{S}\int_{t_0}^{t} G\left[\rho(t'), t'\right] dW(t') = \lim_{n\to\infty} \sum_{i=1}^{n} G\left\{\frac{\rho(t_i) + \rho(t_{i-1})}{2}, t_{i-1}\right\} \left[W(t_i) - W(t_{i-1})\right]. \tag{H.6}$$

Where the limit here is the mean-square limit, and S denotes the Stratonovich integral[1]. . We can use this integral in the definition of a Stratonovich Stochastic Differential Equation (SDE):

$$\rho(t) = \rho(t_0) + \int_{t_0}^{t} dt' \alpha[\rho(t'), t'] + \mathrm{S}\int_{t_0}^{t} dW(t')\beta[\rho(t'), t']. \tag{H.7}$$

We assume that the $\rho(t)$ here solves an Ito SDE, which has the form:

$$d\rho(t) = A[\rho(t), t]dt + B[\rho(t), t]dW(t), \tag{H.8}$$

and use Eqn. H.7, substituting β for G. Writing $\rho(t_i) = \rho(t_{i-1}) + d\rho(t_{i-1})$, and substituting into the Ito SDE (Eqn H.8) we get:

$$d\rho(t_i) = A[\rho(t_{i-1}), t_{i-1}](t_i - t_{i-1}) + B[\rho(t_{i-1}), t_{i-1}][W(t_i) - W(t_{i-1})]. \tag{H.9}$$

[1]The difference between Stratonovich and Ito stochastic integrals is related to where the intermediate points dt are taken [329].

We then apply Ito's formula [329] to the β function:

$$\beta\left[\frac{\rho(t_i)+\rho(t_{i-1})}{2}\right] = \beta\left[\rho(t_{i-1})+\frac{1}{2}d\rho(t_{i-1}),t_{i-1}\right] \tag{H.10}$$

$$= \beta(t_{i-1}) + \left(A(t_{i-1})\partial_\rho\beta(t_{i-1}+\frac{1}{4}B^2(t_{i-1})\right)\left(\frac{1}{2}t_i - t_{i-1}\right)$$

$$+\frac{1}{2}B(t_{i-1})\partial_\rho\beta(t_{i-1})[W(t_i)-W(t_{i-1})] \tag{H.11}$$

we then make the substitutions:

$$dt^2 = dtdW(t) = 0, \tag{H.12}$$

$$dW^2(t) = dt, \tag{H.13}$$

and substitute this into Eqn. H.8:

$$\mathrm{S}\int_{t_0}^{t}\beta[\rho(t'),t']dW(t') = \int_{t_0}^{t}\beta[\rho(t'),t']dW(t') + \frac{1}{2}\int_{t_0}^{t}B[\rho(t'),t']\partial_\rho\beta[\rho(t'),t']dt'. \tag{H.14}$$

So, in our implementation of the problem we have: $\hbar\beta[\rho(t),t] = \sqrt{\Gamma}[H_2,\rho]$, so:

$$\hbar\beta[\rho+\frac{1}{2}d\rho,t] = \sqrt{\Gamma}[H_2,\rho+v\frac{1}{2}d\rho] = \sqrt{\Gamma}\left([H_2,\rho]+\frac{1}{2}([H_2,A]dt+[H_2,B]dW)\right), \tag{H.15}$$

which gives us:

$$\sqrt{\Gamma}[H_2,\rho]\circ dW = \sqrt{\Gamma}\left([H_2,\rho]dW+\frac{1}{2}[H_2,B]dt\right). \tag{H.16}$$

Substituting this into the Stratonovich SDE, eqn. H.5:

$$\hbar d\rho(t) = [H_1,\rho]dt - \frac{i}{2\hbar}\Gamma[H_2,[H_2,\rho]]dt + \sqrt{\Gamma}[H_2,\rho]dW. \tag{H.17}$$

We can now average over many instances of the system, ρ to get the master equation:

$$\hbar\frac{d\langle\rho\rangle}{dt} = [H_1,\langle\rho\rangle] - \frac{i}{2\hbar}\Gamma[H_2,[H_2,\langle\rho\rangle]]. \tag{H.18}$$

In what follows, we will drop the expectation value brackets around rho, as we will always be discussing the ensemble average. The master equation can also be written in the Lindblad form:

$$\frac{d\rho}{dt} = -\frac{i}{\hbar}[H_1, \rho] + \mathcal{L}\rho\mathcal{L}^\dagger - \frac{1}{2}\left\{\mathcal{L}^\dagger\mathcal{L}\rho + \rho\mathcal{L}^\dagger\mathcal{L}\right\}, \tag{H.19}$$

where $\mathcal{L} = \frac{\sqrt{\Gamma}}{\hbar}H_2$. We will use the relationships derived here to move from a description ow white noise on various system parameters to the Lindblad master equation, which can be simulated numerically.

Bibliography

[1] P. W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," in *AT&T Research*, IEEE Computer Society Press, 1994, pp. 20–22. (visited on 01/15/2018).

[2] L. K. Grover, "From Schrödinger's equation to the quantum search algorithm," *Pramana - Journal of Physics*, vol. 56, no. 2-3, pp. 333–348, 2001, ISSN: 03044289. DOI: 10.1119/1.1359518. (visited on 01/15/2018).

[3] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 439, no. 1907, pp. 553–558, Dec. 1992. DOI: 10.1098/rspa.1992.0167. (visited on 07/06/2022).

[4] R. P. Feynman, "Simulating physics with computers," *International Journal of Theoretical Physics*, vol. 21, no. 6-7, pp. 467–488, Jun. 1982, ISSN: 0020-7748. DOI: 10.1007/BF02650179. (visited on 10/05/2021).

[5] D. Lee, J. L. DuBois, and V. Lordi, "Identification of the Local Sources of Paramagnetic Noise in Superconducting Qubit Devices Fabricated on {$\alpha \mathbf{-}\mathrm{Al}_{2}\mathbf{O}_{3}$} Substrates Using Density-Functional Calculations," *Phys. Rev. Lett.*, vol. 112, no. 1, p. 017 001, Jan. 2014. DOI: 10.1103/PhysRevLett.112.017001. (visited on 07/06/2022).

[6] Y. K. Wang, S. M. Fei, Z. X. Wang, J. P. Cao, and H. Fan, "Maximal Holevo quantity based on weak measurements," *Scientific Reports*, 2015, ISSN: 20452322. DOI: 10.1038/srep10727.

[7] P. Kumar, S. Sendelbach, M. A. Beck, *et al.*, "Origin and Reduction of $1/f$ Magnetic Flux Noise in Superconducting Devices," *Phys. Rev. Applied*, vol. 6, no. 4, p. 041 001, Oct. 2016. DOI: 10.1103/PhysRevApplied.6.041001. (visited on 07/06/2022).

[8] S. E. de Graaf, L. Faoro, J. Burnett, *et al.*, "Suppression of low-frequency charge noise in superconducting resonators by surface spin desorption," *Nat Commun*, vol. 9, no. 1, p. 1143, Mar. 2018, ISSN: 2041-1723. DOI: 10.1038/s41467-018-03577-2. (visited on 07/06/2022).

[9] A. Safavi-Naini, E. Kim, P. F. Weck, P. Rabl, and H. R. Sadeghpour, "Influence of monolayer contamination on electric-field-noise heating in ion traps," *Phys. Rev. A*, vol. 87, no. 2, p. 023 421, Feb. 2013. DOI: 10.1103/PhysRevA.87.023421. (visited on 07/06/2022).

[10] S. Mavadia, V. Frey, J. Sastrawan, S. Dona, and M. J. Biercuk, "Prediction and real-time compensation of qubit decoherence via machine learning," *Nat Commun*, vol. 8, no. 1, p. 14 106, Jan. 2017, ISSN: 2041-1723. DOI: 10.1038/ncomms14106. (visited on 07/06/2022).

[11] R. S. Gupta and M. J. Biercuk, "Machine Learning for Predictive Estimation of Qubit Dynamics Subject to Dephasing," *Phys. Rev. Applied*, vol. 9, no. 6, p. 064 042, Jun. 2018. DOI: 10.1103/PhysRevApplied.9.064042. (visited on 07/06/2022).

[12] T. Proctor, M. Revelle, E. Nielsen, *et al.*, "Detecting and tracking drift in quantum information processors," *Nat Commun*, vol. 11, no. 1, p. 5396, Oct. 2020, ISSN: 2041-1723. DOI: 10.1038/s41467-020-19074-4. (visited on 07/06/2022).

[13] S. Majumder, L. Andreta de Castro, and K. R. Brown, "Real-time calibration with spectator qubits," *npj Quantum Inf*, vol. 6, no. 1, pp. 1–9, Feb. 2020, ISSN: 2056-6387. DOI: 10.1038/s41534-020-0251-y. (visited on 07/06/2022).

[14] J. M. Pino, J. M. Dreiling, C. Figgatt, *et al.*, "Demonstration of the trapped-ion quantum-CCD computer architecture," *Nature*, vol. 592, no. 7853, pp. 209–213, Apr. 2021, ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-021-03318-4. arXiv: 2003.01293 [quant-ph]. (visited on 05/18/2022).

[15] A. P. Vepsäläinen, A. H. Karamlou, J. L. Orrell, *et al.*, "Impact of ionizing radiation on superconducting qubit coherence," *Nature*, vol. 584, no. 7822, pp. 551–556, Aug. 2020, ISSN: 1476-4687. DOI: 10.1038/s41586-020-2619-8. (visited on 07/06/2022).

[16] C. D. Wilen, S. Abdullah, N. A. Kurinsky, *et al.*, "Correlated charge noise and relaxation errors in superconducting qubits," *Nature*, vol. 594, no. 7863, pp. 369–373, Jun. 2021, ISSN: 1476-4687. DOI: 10.1038/s41586-021-03557-5. (visited on 07/06/2022).

[17] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018, ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79. (visited on 01/09/2018).

[18] IBM, *Quantum systems - IBM Q*, https://www.research.ibm.com/ibm-q/technology/devices/, 2019. (visited on 07/06/2022).

[19] A. W. Cross, L. S. Bishop, S. Sheldon, P. D. Nation, and J. M. Gambetta, "Validating quantum computers using randomized model circuits," *Phys. Rev. A*, vol. 100, no. 3, p. 032 328, Sep. 2019, ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.100.032328. arXiv: 1811.12926 [quant-ph]. (visited on 07/06/2022).

[20] T. Lubinski, S. Johri, P. Varosy, *et al.*, *Application-Oriented Performance Benchmarks for Quantum Computing*, Dec. 2021. DOI: 10.48550/arXiv.2110.03137. arXiv: 2110.03137 [quant-ph]. (visited on 07/06/2022).

[21] Honeywell, *Honeywell Sets Another Record For Quantum Computing Performance*, https://www.honeywell.com/us/en/news/2021/07/honeywell-

sets-another-record-for-quantum-computing-performance. (visited on 07/06/2022).

[22] F. Arute, K. Arya, R. Babbush, *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, Oct. 2019, ISSN: 14764687. DOI: 10.1038/s41586-019-1666-5. (visited on 09/27/2021).

[23] H.-S. Zhong, H. Wang, Y.-H. Deng, *et al.*, "Quantum computational advantage using photons," *Science*, vol. 370, no. 6523, pp. 1460–1463, Dec. 2020. DOI: 10.1126/science.abe8770. (visited on 04/15/2022).

[24] J. Gray and S. Kourtis, "Hyper-optimized tensor network contraction," *Quantum*, vol. 5, p. 410, Mar. 2021. DOI: 10.22331/q-2021-03-15-410. (visited on 07/06/2022).

[25] L. Egan, D. M. Debroy, C. Noel, *et al.*, "Fault-tolerant control of an error-corrected qubit," *Nature*, vol. 598, no. 7880, pp. 281–286, Oct. 2021, ISSN: 1476-4687. DOI: 10.1038/s41586-021-03928-y. (visited on 07/06/2022).

[26] E. Chertkov, J. Bohnet, D. Francois, *et al.*, "Holographic dynamics simulations with a trapped ion quantum computer," in *Quantum Information and Measurement VI 2021 (2021), Paper W3A.3*, Optica Publishing Group, Nov. 2021, W3A.3. DOI: 10.1364/QIM.2021.W3A.3. (visited on 07/06/2022).

[27] Y. Hu and M. W. Ribbe, "Biosynthesis of nitrogenase FeMoco," *Coordination Chemistry Reviews*, Molybdenum, vol. 255, no. 9, pp. 1218–1224, May 2011, ISSN: 0010-8545. DOI: 10.1016/j.ccr.2010.11.018. (visited on 07/06/2022).

[28] M. Appl, "Ammonia, 2. Production Processes," in *Ullmann's Encyclopedia of Industrial Chemistry*, John Wiley & Sons, Ltd, 2011, ISBN: 978-3-527-30673-2. DOI: 10.1002/14356007.o02_o11. (visited on 07/06/2022).

[29] V. Smil, *Enriching the Earth: Fritz Haber, Carl Bosch, and the Transformation of World Food Production*. Cambridge, MA, USA: MIT Press, Dec. 2000, ISBN: 978-0-262-19449-5.

[30] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, "Elucidating reaction mechanisms on quantum computers," *PNAS*, vol. 114, no. 29, pp. 7555–7560, Jul. 2017, ISSN: 0027-8424, 1091-6490. DOI: 10/gbspk6. (visited on 11/15/2021).

[31] C. Gidney and M. Ekerå, "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits," *Quantum*, vol. 5, p. 433, Apr. 2021. DOI: 10.22331/q-2021-04-15-433. (visited on 07/06/2022).

[32] S. Aaronson and L. Chen, "Complexity-Theoretic Foundations of Quantum Supremacy Experiments," 2016, ISSN: 18688969. DOI: 10.4230/LIPIcs.CCC.2017.22. (visited on 01/03/2018).

[33] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, *et al.*, "Characterizing Quantum Supremacy in Near-Term Devices," Jul. 2016. (visited on 12/02/2017).

[34] M. J. Bremner, A. Montanaro, and D. J. Shepherd, "Achieving quantum supremacy with sparse and noisy commuting quantum computations," *Quantum*, vol. 1, p. 8, 2017, ISSN: 2521-327X. DOI: 10.22331/q-2017-04-25-8. (visited on 01/11/2018).

[35] P. Clifford and R. Clifford, "The Classical Complexity of Boson Sampling," 2017. (visited on 01/11/2018).

[36] A. W. Harrow and A. Montanaro, "Quantum computational supremacy," *Nature*, vol. 549, no. 7671, pp. 203–209, 2017, ISSN: 14764687. DOI: 10.1038/nature23458. (visited on 12/03/2017).

[37] A. P. Lund, M. J. Bremner, and T. C. Ralph, "Quantum Sampling Problems, BosonSampling and Quantum Supremacy," *npj Quantum Information*, vol. 3, no. 1, p. 15, Dec. 2017, ISSN: 2056-6387. DOI: 10.1038/s41534-017-0018-2. (visited on 12/02/2017).

[38] C. Neill, P. Roushan, K. Kechedzhi, *et al.*, "A blueprint for demonstrating quantum supremacy with superconducting qubits," 2017. (visited on 12/03/2017).

[39] S. Aaronson, *Shtetl-Optimized » Blog Archive » Quantum supremacy: The gloves are off*, https://www.scottaaronson.com/blog/?p=4372, 2019. (visited on 10/05/2021).

[40] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943, ISSN: 00074985. DOI: 10.1007/BF02478259. (visited on 08/24/2018).

[41] H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, "An adaptive variational algorithm for exact molecular simulations on a quantum computer," *Nature Communications*, vol. 10, no. 1, 2019, ISSN: 20411723. DOI: 10.1038/s41467-019-10988-2. (visited on 09/17/2020).

[42] H. L. Tang, V. Shkolnikov, G. S. Barron, *et al.*, "Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansätze on a Quantum Processor," *PRX Quantum*, vol. 2, no. 2, p. 020 310, Apr. 2021, ISSN: 2691-3399. DOI: 10.1103/PRXQuantum.2.020310. (visited on 09/23/2021).

[43] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. New York: Cambridge University Press, 2000, ISBN: 978-0-521-63503-5. DOI: 10.2277/0521635039.

[44] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Mod. Phys.*, vol. 90, no. 1, p. 015 002, Jan. 2018. DOI: 10.1103/RevModPhys.90.015002. (visited on 07/06/2022).

[45] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. Van den Nest, "Measurement-based quantum computation," *Nature Phys*, vol. 5, no. 1, pp. 19–26, Jan. 2009, ISSN: 1745-2481. DOI: 10.1038/nphys1157. (visited on 07/06/2022).

[46] I. M. Georgescu, S. Ashhab, and F. Nori, "Quantum simulation," *Rev. Mod. Phys.*, vol. 86, no. 1, pp. 153–185, Mar. 2014. DOI: 10.1103/RevModPhys.86.153. (visited on 07/06/2022).

[47] P. Schindler, D. Nigg, T. Monz, *et al.*, "A quantum information processor with trapped ions," *New Journal of Physics*, vol. 15, no. 12, p. 123 012, Dec. 2013, ISSN: 1367-2630. DOI: 10.1088/1367-2630/15/12/123012. (visited on 01/14/2021).

[48] J. A. Schreier, A. A. Houck, J. Koch, *et al.*, "Suppressing charge noise decoherence in superconducting charge qubits," *Physical Review B - Condensed Matter and Materials Physics*, vol. 77, no. 18, 2008, ISSN: 10980121. DOI: 10.1103/PhysRevB.77.180502. (visited on 01/10/2018).

[49] J. E. Bourassa, R. N. Alexander, M. Vasmer, *et al.*, "Blueprint for a Scalable Photonic Fault-Tolerant Quantum Computer," *Quantum*, vol. 5, p. 392, Feb. 2021, ISSN: 2521-327X. DOI: 10.22331/q-2021-02-04-392. arXiv: 2010.02905 [quant-ph]. (visited on 05/18/2022).

[50] A. Peruzzo, J. McClean, P. Shadbolt, *et al.*, "A variational eigenvalue solver on a photonic quantum processor," *Nature Communications*, vol. 5, no. 1, p. 4213, Sep. 2014, ISSN: 2041-1723. DOI: 10.1038/ncomms5213. (visited on 08/13/2019).

[51] D. Gottesman, "Theory of fault-tolerant quantum computation," *Phys. Rev. A*, vol. 57, no. 1, pp. 127–137, Jan. 1998. DOI: 10.1103/PhysRevA.57.127. (visited on 07/06/2022).

[52] D. Gottesman, "The Heisenberg representation of quantum computers," Los Alamos National Lab. (LANL), Los Alamos, NM (United States), Tech. Rep. LA-UR-98-2848; CONF-980788-, Jun. 1998. (visited on 06/30/2022).

[53] M. Born, "Zur Quantenmechanik der Stoßvorgänge," *Z. Physik*, vol. 37, no. 12, pp. 863–867, Dec. 1926, ISSN: 0044-3328. DOI: 10.1007/BF01397477. (visited on 07/07/2022).

[54] D. Coppersmith, "An approximate Fourier transform useful in quantum factoring," *arXiv:quant-ph/0201067*, Jan. 2002. arXiv: quant-ph/0201067. (visited on 10/25/2021).

[55] J. P. Buhler, H. W. Lenstra, and C. Pomerance, "Factoring integers with the number field sieve," in *The Development of the Number Field Sieve*, A. K. Lenstra and H. W. Lenstra, Eds., ser. Lecture Notes in Mathematics, Berlin, Heidelberg: Springer, 1993, pp. 50–94, ISBN: 978-3-540-47892-8. DOI: 10.1007/BFb0091539.

[56] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, "Simple Pulses for Elimination of Leakage in Weakly Nonlinear Qubits," *Phys. Rev. Lett.*, vol. 103, no. 11, p. 110 501, Sep. 2009, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.103.110501. (visited on 06/30/2022).

[57] D. Hayes, D. Stack, B. Bjork, A. C. Potter, C. H. Baldwin, and R. P. Stutz, "Eliminating Leakage Errors in Hyperfine Qubits," *Physical Review Letters*, vol. 124, 2020. DOI: 10.1103/PhysRevLett.124.170501. (visited on 10/05/2021).

[58] Y. Nam, J.-S. Chen, N. C. Pisenti, *et al.*, "Ground-state energy estimation of the water molecule on a trapped-ion quantum computer," *npj Quantum Inf*, vol. 6, no. 1, pp. 1–6, Apr. 2020, ISSN: 2056-6387. DOI: 10.1038/s41534-020-0259-3. (visited on 06/30/2022).

[59] R. C. Bialczak, M. Ansmann, M. Hofheinz, *et al.*, "Quantum process tomography of a universal entangling gate implemented with Josephson phase qubits," *Nature Physics 2010 6:6*, vol. 6, no. 6, pp. 409–413, Apr. 2010, ISSN: 1745-2481. DOI: 10.1038/nphys1639. (visited on 10/07/2021).

[60] E. Magesan, J. M. Gambetta, A. D. Córcoles, and J. M. Chow, "Machine Learning for Discriminating Quantum Measurement Trajectories and Improving Readout," *Phys. Rev. Lett.*, vol. 114, no. 20, p. 200 501, May 2015, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.114.200501. (visited on 06/30/2022).

[61] F. B. Maciejewski, Z. Zimborás, and M. Oszmaniec, "Mitigation of readout noise in near-term quantum devices by classical post-processing based on

detector tomography," *Quantum*, vol. 4, p. 257, Apr. 2020. DOI: 10.22331/q-2020-04-24-257. (visited on 10/07/2021).

[62] K. E. Hamilton, T. Kharazi, T. Morris, A. J. McCaskey, R. S. Bennink, and R. C. Pooser, "Scalable quantum processor noise characterization," *Proceedings - IEEE International Conference on Quantum Computing and Engineering, QCE 2020*, pp. 430–440, Jun. 2020. (visited on 10/07/2021).

[63] M. R. Geller and M. Sun, "Toward efficient correction of multiqubit measurement errors: Pair correlation method," *Quantum Sci. Technol.*, vol. 6, no. 2, p. 025 009, Feb. 2021, ISSN: 2058-9565. DOI: 10.1088/2058-9565/abd5c9. (visited on 06/30/2022).

[64] I. L. Chuang and M. A. Nielsen, "Prescription for experimental determination of the dynamics of a quantum black box," *Journal of Modern Optics*, vol. 44, no. 11-12, pp. 2455–2467, Nov. 1997, ISSN: 0950-0340, 1362-3044. DOI: 10.1080/09500349708231894. (visited on 06/30/2022).

[65] A. M. Palmieri, E. Kovlakov, F. Bianchi, *et al.*, "Experimental neural network enhanced quantum tomography," *npj Quantum Information 2020 6:1*, vol. 6, no. 1, pp. 1–5, Feb. 2020, ISSN: 2056-6387. DOI: 10.1038/s41534-020-0248-6. (visited on 10/07/2021).

[66] K. E. Hamilton, E. F. Dumitrescu, and R. C. Pooser, "Generative model benchmarks for superconducting qubits," *Physical Review A*, vol. 99, no. 6, p. 062 323, Jun. 2019, ISSN: 2469-9926. DOI: 10.1103/PhysRevA.99.062323. (visited on 10/30/2019).

[67] F. Arute, K. Arya, R. Babbush, *et al.*, "Supplementary information for "Quantum supremacy using a programmable superconducting processor"," *Nature*, vol. 574, no. 7779, pp. 505–510, Oct. 2019, ISSN: 0028-0836, 1476-4687. DOI: 10/ggbnb4. arXiv: 1910.11333. (visited on 11/12/2021).

[68] A. A. Markov and N. M. Nagorny, *The Theory of Algorithms*. Springer Netherlands, Nov. 1988, ISBN: 978-90-277-2773-2.

[69] M. Y. Niu, V. Smelyanskyi, P. Klimov, *et al.*, *Learning Non-Markovian Quantum Noise from Moir\'{e}-Enhanced Swap Spectroscopy with Deep Evolutionary Algorithm*, Dec. 2019. DOI: 10.48550/arXiv.1912.04368. arXiv: 1912.04368 [quant-ph]. (visited on 07/07/2022).

[70] P. Figueroa-Romero, K. Modi, R. J. Harris, T. M. Stace, and M.-H. Hsieh, "Randomized Benchmarking for Non-Markovian Noise," *PRX Quantum*, vol. 2, no. 4, p. 040 351, Dec. 2021. DOI: 10.1103/PRXQuantum.2.040351. (visited on 07/07/2022).

[71] S. B. Bravyi and A. Y. Kitaev, "Quantum codes on a lattice with boundary," *arXiv:quant-ph/9811052*, Nov. 1998. arXiv: quant-ph/9811052. (visited on 04/20/2022).

[72] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, "Topological quantum memory," *Journal of Mathematical Physics*, vol. 43, no. 9, pp. 4452–4505, Sep. 2002, ISSN: 0022-2488, 1089-7658. DOI: 10.1063/1.1499754. arXiv: quant-ph/0110143. (visited on 04/20/2022).

[73] A. Y. Kitaev, "Fault-tolerant quantum computation by anyons," *Annals of Physics*, vol. 303, no. 1, pp. 2–30, Jan. 2003, ISSN: 00034916. DOI: 10.1016/S0003-4916(02)00018-0. arXiv: quant-ph/9707021. (visited on 04/20/2022).

[74] N. Akerman, N. Navon, S. Kotler, Y. Glickman, and R. Ozeri, "Universal gate-set for trapped-ion qubits using a narrow linewidth diode laser," *New Journal of Physics*, vol. 17, no. 11, p. 113 060, Nov. 2015, ISSN: 1367-2630. DOI: 10.1088/1367-2630/17/11/113060. (visited on 03/18/2021).

[75] H. Ball, W. D. Oliver, and M. J. Biercuk, "The role of master clock stability in quantum information processing," *npj Quantum Information*, vol. 2, no. 1, p. 16 033, Nov. 2016, ISSN: 2056-6387. DOI: 10.1038/npjqi.2016.33. (visited on 06/16/2021).

[76] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol, and D. M. Lucas, "High-Fidelity Quantum Logic Gates Using Trapped-Ion Hyperfine Qubits," *Physical Review Letters*, vol. 117, no. 6, 2016, ISSN: 10797114. DOI: 10.1103/PhysRevLett.117.060504. (visited on 01/12/2021).

[77] M. Brownnutt, G. Wilpers, P. Gill, R. C. Thompson, and A. G. Sinclair, "Monolithic microfabricated ion trap chip design for scaleable quantum processors," *New Journal of Physics*, vol. 8, no. 10, pp. 232–232, Oct. 2006, ISSN: 1367-2630. DOI: 10.1088/1367-2630/8/10/232. (visited on 01/27/2021).

[78] W. C. Campbell, J. Mizrahi, Q. Quraishi, *et al.*, "Ultrafast Gates for Single Atomic Qubits," *Physical Review Letters*, vol. 105, no. 9, p. 090 502, Aug. 2010, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.105.090502. (visited on 01/20/2021).

[79] K. Choonee, G. Wilpers, and A. G. Sinclair, "Silicon microfabricated linear segmented ion traps for quantum technologies," in *TRANSDUCERS 2017 - 19th International Conference on Solid-State Sensors, Actuators and Microsystems*, Institute of Electrical and Electronics Engineers Inc., Jul. 2017, pp. 615–618, ISBN: 978-1-5386-2731-0. DOI: 10.1109/TRANSDUCERS.2017.7994124. (visited on 01/27/2021).

[80] T. P. Harty, D. T. C. Allcock, C. J. Ballance, *et al.*, "High-Fidelity Preparation, Gates, Memory, and Readout of a Trapped-Ion Quantum Bit," *Physical Review Letters*, vol. 113, no. 22, p. 220 501, Nov. 2014, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.113.220501. (visited on 01/20/2021).

[81] C. F. Roos, "Ion trap quantum gates with amplitude-modulated laser beams," *New Journal of Physics*, vol. 10, p. 13 002, 2008, ISSN: 13672630. DOI: 10.1088/1367-2630/10/1/013002. (visited on 04/23/2021).

[82] P. See, G. Wilpers, P. Gill, and A. G. Sinclair, "Fabrication of a Monolithic Array of Three Dimensional Si-based Ion Traps," *Journal of Microelectrome-*

chanical Systems, vol. 22, no. 5, pp. 1180–1189, Oct. 2013, ISSN: 1057-7157. DOI: 10.1109/JMEMS.2013.2262573. (visited on 01/27/2021).

[83] J. Thom, B. Yuen, G. Wilpers, E. Riis, and A. G. Sinclair, "Intensity stabilisation of optical pulse sequences for coherent control of laser-driven qubits," *Applied Physics B*, vol. 124, no. 5, p. 90, May 2018, ISSN: 0946-2171. DOI: 10.1007/s00340-018-6955-4. (visited on 01/14/2021).

[84] G. Wilpers, P. See, P. Gill, and A. G. Sinclair, "A monolithic array of three-dimensional ion traps fabricated with conventional semiconductor technology," *Nature Nanotechnology*, vol. 7, no. 9, pp. 572–576, Sep. 2012, ISSN: 1748-3387. DOI: 10.1038/nnano.2012.126. (visited on 01/27/2021).

[85] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Physical Review A*, vol. 52, no. 4, R2493–R2496, Oct. 1995, ISSN: 1050-2947. DOI: 10.1103/PhysRevA.52.R2493. (visited on 10/05/2021).

[86] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, "Decoding quantum errors with subspace expansions," *Nat Commun*, vol. 11, no. 1, p. 636, Jan. 2020, ISSN: 2041-1723. DOI: 10.1038/s41467-020-14341-w. (visited on 04/15/2022).

[87] A. G. Fowler, "Proof of finite surface code threshold for matching," *Phys. Rev. Lett.*, vol. 109, no. 18, p. 180 502, Nov. 2012, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.109.180502. arXiv: 1206.0800. (visited on 04/19/2022).

[88] P. Iyer and D. Poulin, "Hardness of decoding quantum stabilizer codes," *arXiv:1310.3235 [quant-ph]*, Oct. 2013. arXiv: 1310.3235 [quant-ph]. (visited on 04/20/2022).

[89] S. Bravyi, M. Suchara, and A. Vargo, "Efficient Algorithms for Maximum Likelihood Decoding in the Surface Code," *arXiv:1405.4883 [quant-ph]*, May 2014. DOI: 10.1103/PhysRevA.90.032326. arXiv: 1405.4883 [quant-ph]. (visited on 04/20/2022).

[90] A. G. Fowler, "Minimum weight perfect matching of fault-tolerant topological quantum error correction in average $O(1)$ parallel time," *arXiv:1307.1740 [quant-ph]*, Oct. 2014. arXiv: 1307.1740 [quant-ph]. (visited on 04/20/2022).

[91] O. Higgott, "PyMatching: A Python package for decoding quantum codes with minimum-weight perfect matching," *arXiv:2105.13082 [quant-ph]*, Jul. 2021. arXiv: 2105.13082 [quant-ph]. (visited on 04/20/2022).

[92] A. G. Fowler, "2-D color code quantum computation," *Phys. Rev. A*, vol. 83, no. 4, p. 042 310, Apr. 2011, ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.83.042310. arXiv: 0806.4827. (visited on 04/20/2022).

[93] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, "Topological quantum memory," *Journal of Mathematical Physics*, vol. 43, no. 9, pp. 4452–4505, Sep. 2002, ISSN: 0022-2488, 1089-7658. DOI: 10.1063/1.1499754. (visited on 04/20/2022).

[94] M. Vasmer and D. E. Browne, "Three-dimensional surface codes: Transversal gates and fault-tolerant architectures," *Physical Review A*, vol. 100, no. 1, p. 012 312, Jul. 2019, ISSN: 24699934. DOI: 10.1103/PhysRevA.100.012312. (visited on 11/21/2019).

[95] J. P. Bonilla Ataides, D. K. Tuckett, S. D. Bartlett, S. T. Flammia, and B. J. Brown, "The XZZX surface code," *Nat Commun*, vol. 12, no. 1, p. 2172, Dec. 2021, ISSN: 2041-1723. DOI: 10.1038/s41467-021-22274-1. (visited on 04/20/2022).

[96] Q. Xu, N. Mannucci, A. Seif, A. Kubica, S. T. Flammia, and L. Jiang, "Tailored XZZX codes for biased noise," *arXiv:2203.16486 [quant-ph]*, Mar. 2022. arXiv: 2203.16486 [quant-ph]. (visited on 04/20/2022).

[97] C. Chamberland, K. Noh, P. Arrangoiz-Arriola, *et al.*, "Building a fault-tolerant quantum computer using concatenated cat codes," *PRX Quantum*, vol. 3, no. 1, p. 010 329, Feb. 2022, ISSN: 2691-3399. DOI: 10.1103/PRXQuantum.3.010329. arXiv: 2012.04108. (visited on 04/20/2022).

[98] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum Support Vector Machine for Big Data Classification," *Physical Review Letters*, vol. 113, no. 13, p. 130 503, Sep. 2014, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.113.130503. (visited on 05/23/2018).

[99] H.-Y. Huang, M. Broughton, J. Cotler, *et al.*, "Quantum advantage in learning from experiments," *arXiv:2112.00778 [quant-ph]*, Dec. 2021. arXiv: 2112.00778 [quant-ph]. (visited on 12/03/2021).

[100] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman & Co., 1990, ISBN: 978-0-7167-1045-5.

[101] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum Algorithm for Linear Systems of Equations," *Physical Review Letters*, vol. 103, no. 15, p. 150 502, Oct. 2009, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.103.150502. (visited on 08/25/2018).

[102] E. Tang, "A quantum-inspired classical algorithm for recommendation systems," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, ser. STOC 2019, New York, NY, USA: Association for Computing Machinery, Jun. 2019, pp. 217–228, ISBN: 978-1-4503-6705-9. DOI: 10.1145/3313276.3316310. (visited on 06/30/2022).

[103] N.-H. Chia, A. Gilyén, T. Li, H.-H. Lin, E. Tang, and C. Wang, "Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning," in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, Jun. 2020, pp. 387–400. DOI: 10.1145/3357713.3384314. arXiv: 1910.06151 [quant-ph]. (visited on 04/16/2023).

[104] A. Bakshi and E. Tang, *An Improved Classical Singular Value Transformation for Quantum Machine Learning*, Mar. 2023. DOI: 10.48550/arXiv.2303.01492. arXiv: 2303.01492 [quant-ph]. (visited on 04/16/2023).

[105] I. Goodfellow, Y. Bengio, and A. Courvillle, *Deep Learning*. MIT Press, 2016.

[106] M. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2014.

[107] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical Text-Conditional Image Generation with CLIP Latents," *arXiv:2204.06125 [cs]*, Apr. 2022. arXiv: 2204.06125 [cs]. (visited on 04/20/2022).

[108] T. Brown, B. Mann, N. Ryder, *et al.*, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901. (visited on 06/30/2022).

[109] O. Russakovsky, J. Deng, H. Su, *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015, ISSN: 15731405. DOI: 10.1007/s11263-015-0816-y. (visited on 08/11/2020).

[110] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sep. 2017, ISSN: 1476-4687. DOI: 10.1038/nature23474. (visited on 06/30/2022).

[111] N. Wiebe, D. Braun, and S. Lloyd, "Quantum Algorithm for Data Fitting," *Phys. Rev. Lett.*, vol. 109, no. 5, p. 050 505, Aug. 2012. DOI: 10.1103/PhysRevLett.109.050505. (visited on 07/01/2022).

[112] I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20 150 202, Apr. 2016. DOI: 10.1098/rsta.2015.0202. (visited on 07/01/2022).

[113] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, no. 9, pp. 631–633, Sep. 2014, ISSN: 1745-2473. DOI: 10.1038/nphys3029. (visited on 05/29/2018).

[114] V. Giovannetti, S. Lloyd, and L. MacCone, "Quantum random access memory," *Physical Review Letters*, vol. 100, no. 16, 2008, ISSN: 00319007. DOI: 10.1103/PhysRevLett.100.160501. (visited on 08/25/2018).

[115] C. T. Hann, C.-L. Zou, Y. Zhang, *et al.*, "Hardware-Efficient Quantum Random Access Memory with Hybrid Quantum Acoustic Systems," *Phys. Rev. Lett.*, vol. 123, no. 25, p. 250 501, Dec. 2019, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.123.250501. (visited on 06/22/2022).

[116] I. Kerenidis and A. Prakash, "Quantum recommendation system," in *Leibniz International Proceedings in Informatics, LIPIcs*, vol. 67, 2017, ISBN: 978-3-95977-029-3. DOI: 10.4230/LIPIcs.ITCS.2017.49. (visited on 12/08/2020).

[117] S. Huang, M. Newman, and K. R. Brown, "Fault-tolerant weighted union-find decoding on the toric code," *Physical Review A*, vol. 102, no. 1, p. 012 419, Jul. 2020, ISSN: 2469-9926. DOI: 10.1103/PhysRevA.102.012419. (visited on 05/17/2021).

[118] M. W. Johnson, M. H. S. Amin, S. Gildert, *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, pp. 194–198, May 2011, ISSN: 1476-4687. DOI: 10.1038/nature10012. (visited on 07/01/2022).

[119] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983. DOI: 10.1126/science.220.4598.671. (visited on 07/01/2022).

[120] D. Bertsimas and J. Tsitsiklis, "Simulated Annealing," *Statistical Science*, vol. 8, no. 1, pp. 10–15, Feb. 1993, ISSN: 0883-4237, 2168-8745. DOI: 10.1214/ss/1177011077. (visited on 07/01/2022).

[121] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm*, Nov. 2014. DOI: 10.48550/arXiv.1411.4028. arXiv: 1411.4028 [quant-ph]. (visited on 06/30/2022).

[122] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, "From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz," *Algorithms*, vol. 12, no. 2, p. 34, Feb. 2019, ISSN: 1999-4893. DOI: 10.3390/a12020034. (visited on 07/03/2022).

[123] S. Marsh and J. B. Wang, "Combinatorial optimization via highly efficient quantum walks," *Phys. Rev. Research*, vol. 2, no. 2, p. 023 302, Jun. 2020. DOI: 10.1103/PhysRevResearch.2.023302. (visited on 07/04/2022).

[124] E. Farhi and S. Gutmann, "Quantum computation and decision trees," *Physical Review A*, vol. 58, no. 2, pp. 915–928, Aug. 1998, ISSN: 1050-2947. DOI: 10.1103/PhysRevA.58.915. (visited on 05/14/2018).

[125] A. M. Childs and J. Goldstone, "Spatial search by quantum walk," *Phys. Rev. A*, vol. 70, no. 2, p. 022 314, Aug. 2004. DOI: 10.1103/PhysRevA.70.022314. (visited on 07/04/2022).

[126] J. Lynn, I. Tews, S. Gandolfi, and A. Lovato, "Quantum Monte Carlo Methods in Nuclear Physics: Recent Advances," *Annual Review of Nuclear and Particle Science*, vol. 69, no. 1, pp. 279–305, 2019. DOI: 10.1146/annurev-nucl-101918-023600. (visited on 07/05/2022).

[127] T. M. Dieb, S. Ju, J. Shiomi, and K. Tsuda, "Monte Carlo tree search for materials design and discovery," *MRS Communications*, vol. 9, no. 2, pp. 532–536, Jun. 2019, ISSN: 2159-6859, 2159-6867. DOI: 10.1557/mrc.2019.40. (visited on 07/05/2022).

[128] D. Landau and K. Binder, "A brief review of other methods of computer simulation," in *A Guide to Monte Carlo Simulations in Statistical Physics*, Fifth, Cambridge: Cambridge University Press, 2021, pp. 484–518, ISBN: 978-1-108-49014-6. DOI: 10.1017/9781108780346.013. (visited on 07/05/2022).

[129] N. Bolia and S. Juneja, "Monte Carlo methods for pricing financial options," *Sadhana*, vol. 30, no. 2, pp. 347–385, Apr. 2005, ISSN: 0973-7677. DOI: 10.1007/BF02706251. (visited on 07/05/2022).

[130] J. H. Hetherington, "Observations on the statistical iteration of matrices," *Phys. Rev. A*, vol. 30, no. 5, pp. 2713–2719, Nov. 1984. DOI: 10.1103/PhysRevA.30.2713. (visited on 07/05/2022).

[131] A. Montanaro, "Quantum speedup of Monte Carlo methods," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 471, no. 2181, p. 20 150 301, Sep. 2015. DOI: 10.1098/rspa.2015.0301. (visited on 07/01/2022).

[132] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, *Quantum Amplitude Amplification and Estimation*, 2002. DOI: 10.1090/conm/305/05215. arXiv: quant-ph/0005055. (visited on 07/05/2022).

[133] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A Learning Algorithm for Boltzmann Machines*," *Cognitive Science*, vol. 9, no. 1, pp. 147–169, 1985, ISSN: 1551-6709. DOI: 10.1207/s15516709cog0901_7. (visited on 07/05/2022).

[134] M. H. Amin, E. Andriyash, J. Rolfe, B. Kulchytskyy, and R. Melko, "Quantum Boltzmann Machine," *Phys. Rev. X*, vol. 8, no. 2, p. 021 050, May 2018. DOI: 10.1103/PhysRevX.8.021050. (visited on 07/05/2022).

[135] A. Patterson, H. Chen, L. Wossnig, S. Severini, D. Browne, and I. Rungger, "Quantum state discrimination using noisy quantum neural networks," *Physical Review Research*, vol. 3, no. 1, p. 013 063, Jan. 2021, ISSN: 2643-1564. DOI: 10.1103/PhysRevResearch.3.013063. (visited on 09/22/2020).

[136] H. Chen, L. Wossnig, S. Severini, H. Neven, and M. Mohseni, "Universal discriminative quantum neural networks," *Quantum Machine Intelligence*, vol. 3, no. 1, May 2021, ISSN: 2524-4906. DOI: 10.1007/s42484-020-00025-7.

[137] C. H. Bennett, "Quantum cryptography using any two nonorthogonal states," *Physical Review Letters*, vol. 68, no. 21, pp. 3121–3124, May 1992, ISSN: 00319007. DOI: 10.1103/PhysRevLett.68.3121. (visited on 10/14/2019).

[138] A. Chefles, "Quantum state discrimination," *Contemporary Physics*, vol. 41, no. 6, pp. 401–424, Nov. 2000, ISSN: 0010-7514. DOI: 10.1080/00107510010002599. (visited on 10/14/2019).

[139] L.-M. Duan and G.-C. Guo, "Probabilistic Cloning and Identification of Linearly Independent Quantum States," *Physical Review Letters*, vol. 80, no. 22, pp. 4999–5002, Jun. 1998, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.80.4999. (visited on 10/14/2019).

[140] V. Giovannetti, S. Lloyd, and L. Maccone, "Advances in quantum metrology," *Nature Photonics*, vol. 5, no. 4, pp. 222–229, 31/04/2011, ISSN: 1749-4885. DOI: 10.1038/nphoton.2011.35. (visited on 10/14/2019).

[141] S. Lloyd, "Enhanced Sensitivity of Photodetection via Quantum Illumination," *Science*, vol. 321, no. 5895, pp. 1463–1465, Sep. 2008, ISSN: 0036-8075. DOI: 10.1126/science.1160627. (visited on 10/14/2019).

[142] M. Benedetti, E. Grant, L. Wossnig, and S. Severini, "Adversarial quantum circuit learning for pure state approximation," *New Journal of Physics*, vol. 21, no. 4, p. 043 023, Apr. 2019, ISSN: 1367-2630. DOI: 10.1088/1367-2630/ab14b5. (visited on 06/12/2018).

[143] J. A. Bergou, U. Herzog, and M. Hillery, "Quantum Filtering and Discrimination between Sets of Boolean Functions," *Physical Review Letters*, vol. 90, no. 25, p. 4, 2003, ISSN: 10797114. DOI: 10.1103/PhysRevLett.90.257901. (visited on 08/11/2020).

[144] M. Mohseni, A. M. Steinberg, and J. A. Bergou, "Optical realization of optimal unambiguous discrimination for pure and mixed quantum states," *Physical Review Letters*, vol. 93, no. 20, 2004, ISSN: 00319007. DOI: 10.1103/PhysRevLett.93.200403. (visited on 06/11/2018).

[145] V. Negnevitsky, M. Marinelli, K. K. Mehta, H. Y. Lo, C. Flühmann, and J. P. Home, "Repeated multi-qubit readout and feedback with a mixed-species trapped-ion register," *Nature*, vol. 563, no. 7732, pp. 527–531, 2018, ISSN: 14764687. DOI: 10.1038/s41586-018-0668-z. (visited on 11/21/2019).

[146] M. Ježek, J. Řeháček, and J. Fiurášek, "Finding optimal strategies for minimum-error quantum-state discrimination," *Physical Review A - Atomic, Molecular, and Optical Physics*, vol. 65, no. 6, p. 4, Jun. 2002, ISSN: 10941622. DOI: 10.1103/PhysRevA.65.060301. (visited on 08/11/2020).

[147] W. Wang, G. Dong, and M. Li, "Minimum-error quantum state discrimination based on semidefinite programming," in *Proceedings of the 27th Chinese Control Conference, CCC*, 2008, pp. 521–524, ISBN: 978-7-900719-70-6. DOI: 10.1109/CHICC.2008.4605546. (visited on 08/11/2020).

[148] Y. C. Eldar, "A semidefinite programming approach to optimal unambiguous discrimination of quantum states," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 446–456, Feb. 2003, ISSN: 00189448. DOI: 10.1109/TIT.2002.807291. (visited on 08/11/2020).

[149] C. Cîrstoiu, Z. Holmes, J. Iosue, L. Cincio, P. J. Coles, and A. Sornborger, "Variational fast forwarding for quantum simulation beyond the coherence time," *npj Quantum Information*, vol. 6, no. 1, 2020, ISSN: 20566387. DOI: 10.1038/s41534-020-00302-0. (visited on 10/22/2020).

[150] C. Bravo-Prieto, J. Lumbreras-Zarapico, L. Tagliacozzo, and J. I. Latorre, "Scaling of variational quantum circuit depth for condensed matter systems," *Quantum*, vol. 4, 2020, ISSN: 2521327X. DOI: 10.22331/Q-2020-05-28-272. (visited on 10/22/2020).

[151] P. K. Barkoutsos, G. Nannicini, A. Robert, I. Tavernelli, and S. Woerner, "Improving Variational Quantum Optimization using CVaR," *Quantum*, vol. 4, p. 256, 2020, ISSN: 2521-327X. DOI: 10.22331/q-2020-04-20-256. (visited on 10/22/2020).

[152] N. Moll, P. Barkoutsos, L. S. Bishop, *et al.*, "Quantum optimization using variational algorithms on near-term quantum devices," *Quantum Science and Technology*, vol. 3, no. 3, p. 030 503, Jun. 2018, ISSN: 20589565. DOI: 10.1088/2058-9565/aab822. (visited on 01/06/2021).

[153] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. (visited on 06/30/2022).

[154] P. Gokhale, Y. Ding, T. Propson, *et al.*, "Partial Compilation of Variational Algorithms for Noisy Intermediate-Scale Quantum Machines," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture - MICRO '52*, New York, New York, USA: ACM Press, 2019, pp. 266–278, ISBN: 978-1-4503-6938-1. DOI: 10.1145/3352460.3358313. (visited on 10/30/2019).

[155] R. Sweke, F. Wilde, J. Meyer, *et al.*, "Stochastic gradient descent for hybrid quantum-classical optimization," *Preprint*, Oct. 2019.

[156] J.-G. Liu, Y.-H. Zhang, Y. Wan, and L. Wang, "Variational quantum eigensolver with fewer qubits," *Physical Review Research*, vol. 1, no. 2, p. 023 025, Sep. 2019, ISSN: 2643-1564. DOI: 10.1103/PhysRevResearch.1.023025. (visited on 10/30/2019).

[157] A. Neelakantan, L. Vilnis, Q. V. Le, *et al.*, *Adding Gradient Noise Improves Learning for Very Deep Networks*, Nov. 2015. (visited on 06/30/2022).

[158] M. Ostaszewski, E. Grant, and M. Benedetti, "Structure optimization for parameterized quantum circuits," *Quantum*, vol. 5, p. 391, Jan. 2021. DOI: 10/gm85gh. (visited on 10/29/2021).

[159] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Communications*, vol. 9, no. 1, p. 4812, Dec. 2018, ISSN: 2041-1723. DOI: 10.1038/s41467-018-07090-4. (visited on 08/15/2019).

[160] A. Hayashi, T. Hashimoto, and M. Horibe, "State discrimination with error margin and its locality," *Physical Review A - Atomic, Molecular, and Optical Physics*, vol. 78, no. 1, 2008, ISSN: 10502947. DOI: 10.1103/PhysRevA.78.012333. (visited on 08/11/2020).

[161] T. Hashimoto, A. Hayashi, M. Hayashi, and M. Horibe, "Unitary-process discrimination with error margin," *Physical Review A - Atomic, Molecular, and Optical Physics*, vol. 81, no. 6, p. 62 327, 2010, ISSN: 10502947. DOI: 10.1103/PhysRevA.81.062327. (visited on 08/11/2020).

[162] E. Bagan, R. Muñoz-Tapia, G. A. Olivares-Rentería, and J. A. Bergou, "Optimal discrimination of quantum states with a fixed rate of inconclusive outcomes," *Physical Review A - Atomic, Molecular, and Optical Physics*, vol. 86, no. 4, p. 40 303, 2012, ISSN: 10502947. DOI: 10.1103/PhysRevA.86.040303. (visited on 08/11/2020).

[163] U. Herzog, "Optimal measurements for the discrimination of quantum states with a fixed rate of inconclusive results," *Physical Review A - Atomic, Molecular, and Optical Physics*, vol. 91, no. 4, p. 42 338, 2015, ISSN: 10941622. DOI: 10.1103/PhysRevA.91.042338. (visited on 08/11/2020).

[164] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms," *Adv Quantum Tech*, vol. 2, no. 12, p. 1 900 070, Dec. 2019, ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.201900070. (visited on 06/30/2022).

[165] E. Knill, "Quantum computing with realistically noisy devices," *Nature*, vol. 434, no. 7029, pp. 39–44, Mar. 2005, ISSN: 0028-0836. DOI: 10.1038/nature03350. (visited on 04/26/2019).

[166] D. Gottesman, "An Introduction to Quantum Error Correction and Fault-Tolerant Quantum Computation," *Preprint*, pp. 13–58, Apr. 2009. DOI: 10.1090/psapm/068/2762145.

[167] S. S. Hong, A. T. Papageorge, P. Sivarajah, *et al.*, "Demonstration of a parametrically activated entangling gate protected from flux noise," *Physical Review A*, vol. 101, no. 1, p. 012 302, Jan. 2020, ISSN: 2469-9926. DOI: 10.1103/PhysRevA.101.012302.

[168] Y. Chen, C. Neill, P. Roushan, *et al.*, "Qubit architecture with high coherence and fast tunable coupling," *Physical Review Letters*, vol. 113, no. 22, 2014, ISSN: 10797114. DOI: 10.1103/PhysRevLett.113.220502. (visited on 01/10/2018).

[169] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," *New Journal of Physics*, vol. 18, no. 2, p. 023 023, Feb. 2016, ISSN: 1367-2630. DOI: 10.1088/1367-2630/18/2/023023. (visited on 08/13/2019).

[170] M. Abadi, P. Barham, J. Chen, *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'16, USA: USENIX Association, Nov. 2016, pp. 265–283, ISBN: 978-1-931971-33-1. (visited on 06/30/2022).

[171] C. Gidney, D. Bacon, and The Cirq Developers, *Quantumlib/Cirq: A python framework for creating, editing, and invoking Noisy Intermediate Scale Quantum (NISQ) circuits.* https://github.com/quantumlib/Cirq, 2018. (visited on 08/14/2019).

[172] R. Jozsa, "Fidelity for Mixed Quantum States," *Journal of Modern Optics*, vol. 41, no. 12, pp. 2315–2323, Dec. 1994, ISSN: 0950-0340, 1362-3044. DOI: 10/c7v8h3. (visited on 11/12/2021).

[173] S. M. Barnett and S. Croke, "Quantum state discrimination," *Advances in Optics and Photonics*, vol. 1, no. 2, p. 238, Apr. 2009, ISSN: 1943-8206. DOI: 10.1364/AOP.1.000238. (visited on 05/21/2020).

[174] J. Bae, "Optimal state discrimination over quantum channels," in *Quantum Communications and Quantum Imaging XVII*, vol. 11134, SPIE, Sep. 2019, pp. 89–95. DOI: 10/gnfsth. (visited on 11/12/2021).

[175] S. Wang, E. Fontana, M. Cerezo, *et al.*, "Noise-induced barren plateaus in variational quantum algorithms," *Nat Commun*, vol. 12, no. 1, p. 6961, Nov.

2021, ISSN: 2041-1723. DOI: 10.1038/s41467-021-27045-6. (visited on 06/30/2022).

[176] M. Larocca, N. Ju, D. García-Martín, P. J. Coles, and M. Cerezo, *Theory of overparametrization in quantum neural networks*, Sep. 2021. DOI: 10.48550/arXiv.2109.11676. arXiv: 2109.11676 [quant-ph, stat]. (visited on 06/30/2022).

[177] *DuckTigger/Non-Linearities: Training ansatze to non-linear functions.llo*, https://github.com/DuckTigger/AnsatzFinder. (visited on 06/15/2022).

[178] A. Agarwal and E. Fontana, *Non Linearities in Quantum Neural Networks*, 2021.

[179] K. F. Riley, M. P. Hobson, and S. J. Bence, *Mathematical Methods for Physics and Engineering: A Comprehensive Guide*. Cambridge University Press, 2006, ISBN: 978-0-521-86153-3.

[180] S. Yan, H. Qi, and W. Cui, "Nonlinear Quantum Neuron: A Fundamental Building Block for Quantum Neural Networks," *Phys. Rev. A*, vol. 102, no. 5, p. 052 421, Nov. 2020, ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.102.052421. arXiv: 2011.03429. (visited on 04/06/2022).

[181] T. Toffoli, "Reversible computing," in *Automata, Languages and Programming*, J. de Bakker and J. van Leeuwen, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 1980, pp. 632–644, ISBN: 978-3-540-39346-7.

[182] Z. Holmes, N. Coble, A. T. Sornborger, and Y. Subaşı, "On nonlinear transformations in quantum computation," *arXiv:2112.12307 [quant-ph]*, Dec. 2021. arXiv: 2112.12307 [quant-ph]. (visited on 01/05/2022).

[183] N. Guo, K. Mitarai, and K. Fujii, "Nonlinear transformation of complex amplitudes via quantum singular value transformation," *arXiv:2107.10764 [quant-ph]*, Jul. 2021. arXiv: 2107.10764 [quant-ph]. (visited on 04/05/2022).

[184] C. Sünderhauf, E. Campbell, and J. Camps, *Block-encoding structured matrices for data input in quantum computing*, Feb. 2023. DOI: 10.48550/arXiv.2302.10949. arXiv: 2302.10949 [quant-ph]. (visited on 04/15/2023).

[185] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, "Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, Phoenix AZ USA: ACM, Jun. 2019, pp. 193–204, ISBN: 978-1-4503-6705-9. DOI: 10.1145/3313276.3316366. (visited on 04/25/2022).

[186] G. H. Low, T. J. Yoder, and I. L. Chuang, "Methodology of Resonant Equiangular Composite Quantum Gates," *Phys. Rev. X*, vol. 6, no. 4, p. 041 067, Dec. 2016, ISSN: 2160-3308. DOI: 10.1103/PhysRevX.6.041067. (visited on 04/27/2022).

[187] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, "Quantum embeddings for machine learning," *arXiv:2001.03622 [quant-ph]*, Feb. 2020. arXiv: 2001.03622 [quant-ph]. (visited on 04/06/2022).

[188] S. McArdle, S. Endo, A. Aspuru-Guzik, S. Benjamin, and X. Yuan, "Quantum computational chemistry," *Rev. Mod. Phys.*, vol. 92, no. 1, p. 015 003, Mar. 2020, ISSN: 0034-6861, 1539-0756. DOI: 10.1103/RevModPhys.92.015003. arXiv: 1808.10402. (visited on 05/07/2022).

[189] A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory*. Courier Corporation, Jul. 1996, ISBN: 978-0-486-69186-2.

[190] D. A. McQuarrie, *Quantum Chemistry*. University Science Books, 2008, ISBN: 978-1-891389-50-4.

[191] T. Helgaker, P. Jorgensen, and J. Olsen, *Molecular Electronic-Structure Theory*. John Wiley & Sons, Aug. 2014, ISBN: 978-1-119-01955-8.

[192] S. McArdle, "Quantum algorithms for simulating quantum systems," Ph.D. dissertation, University of Oxford, 2020.

[193] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, "Low-Depth Quantum Simulation of Materials," *Phys. Rev. X*, vol. 8, no. 1, p. 011 044, Mar. 2018, ISSN: 2160-3308. DOI: 10.1103/PhysRevX.8.011044. (visited on 05/08/2022).

[194] J. C. Slater, "The Theory of Complex Spectra," *Phys. Rev.*, vol. 34, no. 10, pp. 1293–1322, Nov. 1929, ISSN: 0031-899X. DOI: 10.1103/PhysRev.34.1293. (visited on 05/08/2022).

[195] R. Babbush, D. W. Berry, Y. R. Sanders, *et al.*, "Exponentially more precise quantum simulation of fermions in the configuration interaction representation," *Quantum Sci. Technol.*, vol. 3, no. 1, p. 015 006, Jan. 2018, ISSN: 2058-9565. DOI: 10.1088/2058-9565/aa9463. (visited on 05/08/2022).

[196] I. D. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, "Bounding the costs of quantum simulation of many-body physics in real space," *J. Phys. A: Math. Theor.*, vol. 50, no. 30, p. 305 301, Jun. 2017, ISSN: 1751-8121. DOI: 10.1088/1751-8121/aa77b8. (visited on 05/08/2022).

[197] B. Toloui and P. J. Love, "Quantum Algorithms for Quantum Chemistry based on the sparsity of the CI-matrix," *arXiv:1312.2579 [quant-ph]*, Dec. 2013. arXiv: 1312.2579 [quant-ph]. (visited on 05/08/2022).

[198] P. Jordan and E. Wigner, "Über das Paulische Äquivalenzverbot," *Z. Physik*, vol. 47, no. 9, pp. 631–651, Sep. 1928, ISSN: 0044-3328. DOI: 10.1007/BF01331938. (visited on 10/26/2021).

[199] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, "Simulating physical phenomena by quantum networks," *Phys. Rev. A*, vol. 65, no. 4, p. 042 323, Apr. 2002, ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.65.042323. (visited on 10/26/2021).

[200] J. T. Seeley, M. J. Richard, and P. J. Love, "The Bravyi-Kitaev transformation for quantum computation of electronic structure," *J. Chem. Phys.*, vol. 137, no. 22, p. 224 109, Dec. 2012, ISSN: 0021-9606. DOI: 10.1063/1.4768229. (visited on 10/26/2021).

[201] S. B. Bravyi and A. Y. Kitaev, "Fermionic Quantum Computation," *Annals of Physics*, vol. 298, no. 1, pp. 210–226, May 2002, ISSN: 0003-4916. DOI: 10.1006/aphy.2002.6254. (visited on 10/26/2021).

[202] A. Tranter, P. J. Love, F. Mintert, and P. V. Coveney, "A Comparison of the Bravyi–Kitaev and Jordan–Wigner Transformations for the Quantum Simulation of Quantum Chemistry," *J. Chem. Theory Comput.*, vol. 14, no. 11, pp. 5617–5630, Nov. 2018, ISSN: 1549-9618, 1549-9626. DOI: 10.1021/acs.jctc.8b00450. (visited on 05/16/2022).

[203] C. Derby and J. Klassen, "A Compact Fermion to Qubit Mapping," *Phys. Rev. B*, vol. 104, no. 3, p. 035 118, Jul. 2021, ISSN: 2469-9950, 2469-9969. DOI: 10.1103/PhysRevB.104.035118. arXiv: 2003.06939 [quant-ph]. (visited on 05/16/2022).

[204] C. Derby and J. Klassen, *A Compact Fermion to Qubit Mapping Part 2: Alternative Lattice Geometries*, Jan. 2021. DOI: 10.48550/arXiv.2101.10735. arXiv: 2101.10735 [quant-ph]. (visited on 05/16/2022).

[205] H. F. Trotter, "On the product of semi-groups of operators," *Proc. Amer. Math. Soc.*, vol. 10, no. 4, pp. 545–551, 1959, ISSN: 0002-9939, 1088-6826. DOI: 10.1090/S0002-9939-1959-0108732-6. (visited on 05/16/2022).

[206] M. Suzuki, "General theory of fractal path integrals with applications to many-body theories and statistical physics," *J. Math. Phys.*, vol. 32, no. 2, pp. 400–407, Feb. 1991, ISSN: 0022-2488. DOI: 10.1063/1.529425. (visited on 05/16/2022).

[207] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, "Efficient quantum algorithms for simulating sparse Hamiltonians," *Commun. Math. Phys.*, vol. 270, no. 2, pp. 359–371, Mar. 2007, ISSN: 0010-3616, 1432-0916. DOI: 10.1007/s00220-006-0150-x. arXiv: quant-ph/0508139. (visited on 05/16/2022).

[208] I. D. Kivlichan, C. Gidney, D. W. Berry, *et al.*, "Improved Fault-Tolerant Quantum Simulation of Condensed-Phase Correlated Electrons via Trotterization," *Quantum*, vol. 4, p. 296, Jul. 2020. DOI: 10/gnfsw2. (visited on 11/12/2021).

[209] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, "A Theory of Trotter Error," *Phys. Rev. X*, vol. 11, no. 1, p. 011 020, Feb. 2021, ISSN: 2160-3308. DOI: 10.1103/PhysRevX.11.011020. arXiv: 1912.08854. (visited on 05/16/2022).

[210] P. M. Q. Cruz, G. Catarina, R. Gautier, and J. Fernández-Rossier, "Optimizing quantum phase estimation for the simulation of Hamiltonian eigenstates," *Quantum Sci. Technol.*, vol. 5, no. 4, p. 044 005, Aug. 2020, ISSN: 2058-9565. DOI: 10.1088/2058-9565/abaa2c. arXiv: 1910.06265. (visited on 10/25/2021).

[211] D. Wang, O. Higgott, and S. Brierley, "Accelerated variational quantum eigensolver," *Physical Review Letters*, vol. 122, no. 14, 2019, ISSN: 10797114. DOI: 10.1103/PhysRevLett.122.140504. (visited on 09/17/2020).

[212] A. Kitaev, A. Shen, and M. Vyalyi, *Classical and Quantum Computation* (Graduate Studies in Mathematics). American Mathematical Society, May 2002, vol. 47, ISBN: 978-0-8218-3229-5 978-1-4704-0927-2 978-1-4704-2011-6 978-1-4704-1800-7. DOI: 10.1090/gsm/047. (visited on 06/30/2022).

[213] N. Wiebe, C. Granade, A. Kapoor, and K. M. Svore, *Bayesian inference via rejection filtering*, Dec. 2015. DOI: 10.48550/arXiv.1511.06458. arXiv: 1511.06458 [quant-ph, stat]. (visited on 06/30/2022).

[214] N. Wiebe and C. Granade, "Efficient Bayesian Phase Estimation," *Phys. Rev. Lett.*, vol. 117, no. 1, p. 010 503, Jun. 2016, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.117.010503. (visited on 06/30/2022).

[215] Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles, "Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus," *PRX Quantum*, vol. 3, no. 1, p. 010 313, Jan. 2022. DOI: 10.1103/PRXQuantum.3.010313. (visited on 06/30/2022).

[216] H. R. Grimsley, D. Claudino, S. E. Economou, E. Barnes, and N. J. Mayhall, "Is the Trotterized UCCSD Ansatz chemically well-defined?" *J. Chem. Theory Comput.*, vol. 16, no. 1, pp. 1–6, Jan. 2020, ISSN: 1549-9618, 1549-9626. DOI: 10.1021/acs.jctc.9b01083. arXiv: 1910.10329 [quant-ph]. (visited on 05/16/2022).

[217] A. Kandala, A. Mezzacapo, K. Temme, *et al.*, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature*, vol. 549, no. 7671, pp. 242–246, Sep. 2017, ISSN: 1476-4687. DOI: 10/gbwmbw. (visited on 10/28/2021).

[218] C. S. Forbes, M. Evans, C. Forbes, M. Evans, N. Hastings, and B. Peacock, Eds., *Statistical Distributions*, 4. ed. Hoboken: Wiley, 2011, ISBN: 978-0-470-39063-4.

[219] O. Crawford, B. van Straaten, D. Wang, T. Parks, E. Campbell, and S. Brierley, "Efficient quantum measurement of Pauli operators in the presence of finite sampling error," *Quantum*, vol. 5, p. 385, Jan. 2021, ISSN: 2521-327X. DOI: 10/gkbmw9. arXiv: 1908.06942. (visited on 11/01/2021).

[220] R. Noek, G. Vrijsen, D. Gaultney, *et al.*, "High speed, high fidelity detection of an atomic hyperfine qubit," *Opt. Lett.*, vol. 38, no. 22, p. 4735, Nov. 2013, ISSN: 0146-9592, 1539-4794. DOI: 10.1364/OL.38.004735. (visited on 05/18/2022).

[221] E. Jeffrey, D. Sank, J. Y. Mutus, *et al.*, "Fast accurate state measurement with superconducting qubits," *Physical Review Letters*, vol. 112, no. 19, 2014, ISSN: 10797114. DOI: 10.1103/PhysRevLett.112.190504. (visited on 01/10/2018).

[222] E. A. Sete, M. J. Reagor, N. Didier, and C. T. Rigetti, "Charge- and Flux-Insensitive Tunable Superconducting Qubit," *Physical Review Applied*, vol. 8, no. 2, p. 024 004, Aug. 2017, ISSN: 2331-7019. DOI: 10.1103/PhysRevApplied.8.024004. (visited on 05/10/2018).

[223] P. Gokhale, O. Angiuli, Y. Ding, *et al.*, *Minimizing State Preparations in Variational Quantum Eigensolver by Partitioning into Commuting Families*, Jul. 2019. arXiv: 1907.13623 [quant-ph]. (visited on 05/18/2022).

[224] V. Verteletskyi, T.-C. Yen, and A. F. Izmaylov, "Measurement optimization in the variational quantum eigensolver using a minimum clique cover," *J. Chem. Phys.*, vol. 152, no. 12, p. 124 114, Mar. 2020, ISSN: 0021-9606, 1089-7690. DOI: 10/gnc7pk. (visited on 11/02/2021).

[225] D. Gottesman, *Stabilizer Codes and Quantum Error Correction*, May 1997. arXiv: quant-ph/9705052. (visited on 05/20/2022).

[226] S. Aaronson and D. Gottesman, "Improved Simulation of Stabilizer Circuits," *Phys. Rev. A*, vol. 70, no. 5, p. 052 328, Nov. 2004, ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.70.052328. arXiv: quant-ph/0406196. (visited on 05/20/2022).

[227] K. J. Sung, J. Yao, M. P. Harrigan, *et al.*, "Using models to improve optimizers for variational quantum algorithms," *Quantum Sci. Technol.*, vol. 5, no. 4, p. 044 008, Oct. 2020, ISSN: 2058-9565. DOI: 10.1088/2058-9565/abb6d9. arXiv: 2005.11011 [quant-ph]. (visited on 06/06/2022).

[228] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Physical Review A*, vol. 98, no. 3, p. 032 309, Sep. 2018, ISSN: 2469-9926. DOI: 10.1103/PhysRevA.98.032309. (visited on 06/11/2018).

[229] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, "Evaluating analytic gradients on quantum hardware," *Physical Review A*, vol. 99, no. 3, p. 032 331, Mar. 2019, ISSN: 2469-9926. DOI: 10.1103/PhysRevA.99.032331. (visited on 08/13/2019).

[230] R. M. Parrish, E. G. Hohenstein, P. L. McMahon, and T. J. Martinez, *Hybrid Quantum/Classical Derivative Theory: Analytical Gradients and Excited-State Dynamics for the Multistate Contracted Variational Quantum Eigensolver*, Jun. 2019. DOI: 10.48550/arXiv.1906.08728. arXiv: 1906.08728 [quant-ph]. (visited on 06/30/2022).

[231] J. Duchi, E. Hazan, and Y. Singer, "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization," *J. Mach. Learn. Res.*, vol. 12, no. null, pp. 2121–2159, Jul. 2011, ISSN: 1532-4435.

[232] M. D. Zeiler, *ADADELTA: An Adaptive Learning Rate Method*, Dec. 2012. arXiv: 1212.5701 [cs]. (visited on 05/22/2022).

[233] K. M. Nakanishi, K. Fujii, and S. Todo, "Sequential minimal optimization for quantum-classical hybrid algorithms," *Phys. Rev. Research*, vol. 2, no. 4, p. 043 158, Oct. 2020, ISSN: 2643-1564. DOI: 10/ghp8z3. (visited on 10/29/2021).

[234] R. M. Parrish, J. T. Iosue, A. Ozaeta, and P. L. McMahon, "A Jacobi Diagonalization and Anderson Acceleration Algorithm For Variational Quantum Algorithm Parameter Optimization," *arXiv:1904.03206 [quant-ph]*, Apr. 2019. arXiv: 1904.03206 [quant-ph]. (visited on 10/29/2021).

[235] J. Bergstra and Y. Bengio, "Random Search for Hyper-Parameter Optimization," *Journal of Machine Learning Research*, vol. 13, no. 10, pp. 281–305, 2012, ISSN: 1533-7928. (visited on 06/30/2022).

[236] J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, Jan. 1965, ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308. (visited on 08/13/2019).

[237] J. M. Renes, R. Blume-Kohout, A. J. Scott, and C. M. Caves, "Symmetric informationally complete quantum measurements," *Journal of Mathematical Physics*, vol. 45, no. 6, pp. 2171–2180, Jun. 2004, ISSN: 0022-2488, 1089-7658. DOI: 10.1063/1.1737053. (visited on 05/25/2022).

[238] A. W. Harrow and R. A. Low, "Random Quantum Circuits are Approximate 2-designs," *Commun. Math. Phys.*, vol. 291, no. 1, pp. 257–302, Oct. 2009, ISSN: 0010-3616, 1432-0916. DOI: 10.1007/s00220-009-0873-6. (visited on 05/25/2022).

[239] T. Farshi, D. Toniolo, C. E. González-Guillén, Á. M. Alhambra, and L. Masanes, "Mixing and localization in random time-periodic quantum circuits of Clifford unitaries," *J. Math. Phys.*, vol. 63, no. 3, p. 032 201, Mar. 2022, ISSN: 0022-2488. DOI: 10.1063/5.0054863. (visited on 05/25/2022).

[240] Z. Puchała and J. A. Miszczak, "Symbolic integration with respect to the Haar measure on the unitary group," *Bulletin of the Polish Academy of Sciences Technical Sciences*, vol. 65, no. 1, pp. 21–27, Feb. 2017, ISSN: 2300-1917. DOI: 10.1515/bpasts-2017-0003. arXiv: 1109.4244 [physics, physics:quant-ph]. (visited on 05/27/2022).

[241] C. Hirche, C. Rouzé, and D. S. França, *On contraction coefficients, partial orders and approximation of capacities for quantum channels*, Aug. 2021. arXiv: 2011.05949 [quant-ph]. (visited on 05/29/2022).

[242] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost Function Dependent Barren Plateaus in Shallow Parametrized Quantum Circuits," *Nat Commun*, vol. 12, no. 1, p. 1791, Dec. 2021, ISSN: 2041-1723. DOI: 10.1038/s41467-021-21728-w. arXiv: 2001.00550 [quant-ph]. (visited on 05/30/2022).

[243] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, "Layerwise learning for quantum neural networks," *Quantum Mach. Intell.*, vol. 3, no. 1, p. 5, Jun. 2021, ISSN: 2524-4906, 2524-4914. DOI: 10.1007/s42484-020-00036-4. arXiv: 2006.14904 [quant-ph]. (visited on 05/30/2022).

[244] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti, "An initialization strategy for addressing barren plateaus in parametrized quantum circuits," *Quantum*, vol. 3, p. 214, Dec. 2019, ISSN: 2521-327X. DOI: 10.

22331/q-2019-12-09-214. arXiv: 1903.05076 [quant-ph]. (visited on 05/30/2022).

[245] T. Volkoff and P. J. Coles, "Large gradients via correlation in random parameterized quantum circuits," *Quantum Sci. Technol.*, vol. 6, no. 2, p. 025 008, Apr. 2021, ISSN: 2058-9565. DOI: 10.1088/2058-9565/abd891. (visited on 05/30/2022).

[246] I. Rungger, N. Fitzpatrick, H. Chen, *et al.*, *Dynamical mean field theory algorithm and experiment on quantum computers*, Jan. 2020. DOI: 10.48550/arXiv.1910.04735. arXiv: 1910.04735 [cond-mat, physics:quant-ph]. (visited on 06/30/2022).

[247] D. C. McKay and B. DeMarco, "Cooling in strongly correlated optical lattices: Prospects and challenges," *Rep. Prog. Phys.*, vol. 74, no. 5, p. 054 401, Apr. 2011, ISSN: 0034-4885. DOI: 10.1088/0034-4885/74/5/054401. (visited on 06/07/2022).

[248] R. Orús, "Tensor networks for complex quantum systems," *Nat Rev Phys*, vol. 1, no. 9, pp. 538–550, Sep. 2019, ISSN: 2522-5820. DOI: 10.1038/s42254-019-0086-7. (visited on 06/07/2022).

[249] C. Cade, L. Mineh, A. Montanaro, and S. Stanisic, "Strategies for solving the Fermi-Hubbard model on near-term quantum computers," *Phys. Rev. B*, vol. 102, no. 23, p. 235 122, Dec. 2020, ISSN: 2469-9950, 2469-9969. DOI: 10.1103/PhysRevB.102.235122. arXiv: 1912.06007 [quant-ph]. (visited on 06/06/2022).

[250] S. Stanisic, J. L. Bosse, F. M. Gambetta, *et al.*, *Observing ground-state properties of the Fermi-Hubbard model using a scalable algorithm on a quantum computer*, Dec. 2021. DOI: 10.48550/arXiv.2112.02025. arXiv: 2112.02025 [cond-mat, physics:quant-ph]. (visited on 06/07/2022).

[251] L. Mineh and A. Montanaro, "Solving the Hubbard model using density matrix embedding theory and the variational quantum eigensolver," *Phys.*

Rev. B, vol. 105, no. 12, p. 125 117, Mar. 2022. DOI: 10.1103/PhysRevB.105.125117. (visited on 06/06/2022).

[252] K. Seki and S. Yunoki, "Spatial, spin, and charge symmetry projections for a Fermi-Hubbard model on a quantum computer," *Phys. Rev. A*, vol. 105, no. 3, p. 032 419, Mar. 2022, ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.105.032419. arXiv: 2112.14077 [cond-mat, physics:quant-ph]. (visited on 06/07/2022).

[253] J. Hubbard and B. H. Flowers, "Electron correlations in narrow energy bands," *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 276, no. 1365, pp. 238–257, Nov. 1963. DOI: 10.1098/rspa.1963.0204. (visited on 06/06/2022).

[254] Nature, "The Hubbard model at half a century," *Nature Phys*, vol. 9, no. 9, pp. 523–523, Sep. 2013, ISSN: 1745-2481. DOI: 10.1038/nphys2759. (visited on 06/06/2022).

[255] M. C. Gutzwiller, "Effect of Correlation on the Ferromagnetism of Transition Metals," *Phys. Rev. Lett.*, vol. 10, no. 5, pp. 159–162, Mar. 1963. DOI: 10.1103/PhysRevLett.10.159. (visited on 06/06/2022).

[256] J. Kanamori, "Electron Correlation and Ferromagnetism of Transition Metals," *Progress of Theoretical Physics*, vol. 30, no. 3, pp. 275–289, Sep. 1963, ISSN: 0033-068X. DOI: 10.1143/PTP.30.275. (visited on 06/06/2022).

[257] S. Robaszkiewicz and B. R. Bulka, "Superconductivity in the Hubbard model with pair hopping," *Phys. Rev. B*, vol. 59, no. 9, pp. 6430–6437, Mar. 1999, ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.59.6430. arXiv: cond-mat/9806088. (visited on 06/06/2022).

[258] T. Yanagisawa, "Physics of the Hubbard model and high temperature superconductivity," *J. Phys.: Conf. Ser.*, vol. 108, p. 012 010, Mar. 2008, ISSN: 1742-6596. DOI: 10.1088/1742-6596/108/1/012010. (visited on 06/06/2022).

[259] D. P. Arovas, E. Berg, S. Kivelson, and S. Raghu, "The Hubbard Model," *Annu. Rev. Condens. Matter Phys.*, vol. 13, no. 1, pp. 239–274, Mar. 2022, ISSN: 1947-5454, 1947-5462. DOI: 10.1146/annurev-conmatphys-031620-102024. arXiv: 2103.12097 [cond-mat]. (visited on 06/06/2022).

[260] M. Qin, C.-M. Chung, H. Shi, *et al.*, "Absence of superconductivity in the pure two-dimensional Hubbard model," *Phys. Rev. X*, vol. 10, no. 3, p. 031 016, Jul. 2020, ISSN: 2160-3308. DOI: 10.1103/PhysRevX.10.031016. arXiv: 1910.08931 [cond-mat]. (visited on 06/06/2022).

[261] Z. Cai, "Resource Estimation for Quantum Variational Simulations of the Hubbard Model," *Phys. Rev. Applied*, vol. 14, no. 1, p. 014 059, Jul. 2020, ISSN: 2331-7019. DOI: 10.1103/PhysRevApplied.14.014059. (visited on 06/06/2022).

[262] E. T. Campbell, "Early fault-tolerant simulations of the Hubbard model," *Quantum Sci. Technol.*, vol. 7, no. 1, p. 015 007, Jan. 2022, ISSN: 2058-9565. DOI: 10.1088/2058-9565/ac3110. arXiv: 2012.09238 [quant-ph]. (visited on 06/06/2022).

[263] I. D. Kivlichan, C. Gidney, D. W. Berry, *et al.*, "Improved Fault-Tolerant Quantum Simulation of Condensed-Phase Correlated Electrons via Trotterization," *Quantum*, vol. 4, p. 296, Jul. 2020, ISSN: 2521-327X. DOI: 10.22331/q-2020-07-16-296. arXiv: 1902.10673 [physics, physics:quant-ph]. (visited on 06/06/2022).

[264] A. Georges and G. Kotliar, "Hubbard model in infinite dimensions," *Phys. Rev. B*, vol. 45, no. 12, pp. 6479–6483, Mar. 1992. DOI: 10.1103/PhysRevB.45.6479. (visited on 06/06/2022).

[265] M. Jarrell, "Hubbard model in infinite dimensions: A quantum Monte Carlo study," *Phys. Rev. Lett.*, vol. 69, no. 1, pp. 168–171, Jul. 1992, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.69.168. (visited on 06/06/2022).

[266] P. W. Anderson, "Localized Magnetic States in Metals," *Phys. Rev.*, vol. 124, no. 1, pp. 41–53, Oct. 1961. DOI: 10.1103/PhysRev.124.41. (visited on 06/30/2022).

[267] W. J. de Haas, J. de Boer, and G. J. van dën Berg, "The electrical resistance of gold, copper and lead at low temperatures," *Physica*, vol. 1, no. 7, pp. 1115–1124, May 1934, ISSN: 0031-8914. DOI: 10.1016/S0031-8914(34)80310-2. (visited on 06/06/2022).

[268] J. Kondo, "Resistance Minimum in Dilute Magnetic Alloys," *Progress of Theoretical Physics*, vol. 32, no. 1, pp. 37–49, Jul. 1964, ISSN: 0033-068X. DOI: 10.1143/PTP.32.37. (visited on 06/06/2022).

[269] G. Green, *An Essay on the Application of Mathematical Analysis to the Theories of Electricity and Magnetism*. author, 1828.

[270] T. Matsubara, "A New Approach to Quantum-Statistical Mechanics," *Prog. Theor. Phys.*, vol. 14, no. 4, pp. 351–378, Oct. 1955, ISSN: 0033-068X. DOI: 10.1143/PTP.14.351. (visited on 06/07/2022).

[271] M. Kinza, J. Ortloff, J. Bauer, and C. Honerkamp, "Alternative functional renormalization group approach to the single impurity Anderson model," *PHYSICAL REVIEW B*, p. 13, 2013. DOI: 10.1103/PhysRevB.87.035111.

[272] M. Potthoff, "Two-site dynamical mean-field theory," *Phys. Rev. B*, vol. 64, no. 16, p. 165 114, Oct. 2001. DOI: 10/cmpxbx. (visited on 11/09/2021).

[273] H. Lehmann, "Über Eigenschaften von Ausbreitungsfunktionen und Renormierungskonstanten quantisierter Felder," *Nuovo Cim*, vol. 11, no. 4, pp. 342–357, Apr. 1954, ISSN: 1827-6121. DOI: 10.1007/BF02783624. (visited on 06/07/2022).

[274] J. Kaye, K. Chen, and O. Parcollet, *Discrete Lehmann representation of imaginary time Green's functions*, Feb. 2022. arXiv: 2107.13094 [cond-mat]. (visited on 06/07/2022).

[275] J. M. Kreula, L. García-Álvarez, L. Lamata, S. R. Clark, E. Solano, and D. Jaksch, "Few-qubit quantum-classical simulation of strongly correlated lattice fermions," *EPJ Quantum Technology*, vol. 3, no. 1, p. 11, 2016, ISSN: 2196-0763. DOI: 10.1140/epjqt/s40507-016-0049-1. (visited on 10/04/2018).

[276] B. Bauer, D. Wecker, A. J. Millis, M. B. Hastings, and M. Troyer, "Hybrid Quantum-Classical Approach to Correlated Materials," *Phys. Rev. X*, vol. 6, no. 3, p. 031 045, Sep. 2016, ISSN: 2160-3308. DOI: 10.1103/PhysRevX.6.031045. (visited on 03/17/2022).

[277] T. Keen, T. Maier, S. Johnston, and P. Lougovski, "Quantum-classical simulation of two-site dynamical mean-field theory on noisy quantum hardware," *arXiv:1910.09512 [cond-mat, physics:quant-ph]*, Nov. 2021. DOI: 10.1088/2058-9565/ab7d4c. arXiv: 1910.09512 [cond-mat, physics:quant-ph]. (visited on 03/24/2022).

[278] S. Endo, I. Kurata, and Y. O. Nakagawa, "Calculation of the Green's function on near-term quantum computers," *arXiv:1909.12250 [quant-ph]*, Aug. 2020. DOI: 10.1103/PhysRevResearch.2.033281. arXiv: 1909.12250 [quant-ph]. (visited on 03/24/2022).

[279] Y. Li and S. C. Benjamin, "Efficient Variational Quantum Simulator Incorporating Active Error Minimization," *Phys. Rev. X*, vol. 7, no. 2, p. 021 050, Jun. 2017. DOI: 10/gbk2d9. (visited on 11/13/2021).

[280] K. Temme, S. Bravyi, and J. M. Gambetta, "Error Mitigation for Short-Depth Quantum Circuits," *Phys. Rev. Lett.*, vol. 119, no. 18, p. 180 509, Nov. 2017, ISSN: 0031-9007, 1079-7114. DOI: 10/gcn4nk. (visited on 11/13/2021).

[281] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, "Error mitigation extends the computational reach of a noisy quantum processor," *Nature*, vol. 567, no. 7749, pp. 491–495, Mar. 2019, ISSN: 1476-4687. DOI: 10.1038/s41586-019-1040-7. (visited on 06/30/2022).

[282] J. Nocedal and S. J. Wright, *Numerical Optimization* (Springer Series in Operations Research), 2nd ed. New York: Springer, 2006, ISBN: 978-0-387-30303-1.

[283] I. G. Ryabinkin, S. N. Genin, and A. F. Izmaylov, "Constrained Variational Quantum Eigensolver: Quantum Computer Search Engine in the Fock Space," *J. Chem. Theory Comput.*, vol. 15, no. 1, pp. 249–255, Jan. 2019, ISSN: 1549-9618, 1549-9626. DOI: 10.1021/acs.jctc.8b00943. (visited on 06/07/2022).

[284] K. M. Nakanishi, K. Mitarai, and K. Fujii, "Subspace-search variational quantum eigensolver for excited states," *Phys. Rev. Research*, vol. 1, no. 3, p. 033 062, Oct. 2019, ISSN: 2643-1564. DOI: 10.1103/PhysRevResearch.1.033062. arXiv: 1810.09434 [quant-ph]. (visited on 06/07/2022).

[285] J. R. McClean, M. E. Kimchi-Schwartz, J. Carter, and W. A. de Jong, "Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states," *Phys. Rev. A*, vol. 95, no. 4, p. 042 308, Apr. 2017, ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.95.042308. (visited on 11/12/2021).

[286] O. Higgott, D. Wang, and S. Brierley, "Variational Quantum Computation of Excited States," *Quantum*, vol. 3, p. 156, Jul. 2019, ISSN: 2521-327X. DOI: 10.22331/q-2019-07-01-156. arXiv: 1805.08138 [quant-ph]. (visited on 06/07/2022).

[287] F. Arute, K. Arya, R. Babbush, *et al.*, "Hartree-Fock on a superconducting qubit quantum computer," *Science*, vol. 369, no. 6507, pp. 1084–1089, Aug. 2020, ISSN: 0036-8075, 1095-9203. DOI: 10/ghk7pf. arXiv: 2004.04174. (visited on 10/28/2021).

[288] L. Zhu, H. L. Tang, G. S. Barron, *et al.*, *An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer*, Dec. 2020. DOI: 10.48550/arXiv.2005.10258. arXiv: 2005.10258 [quant-ph]. (visited on 06/30/2022).

[289] P. Murali, N. M. Linke, M. Martonosi, A. J. Abhari, N. H. Nguyen, and C. H. Alderete, "Architecting Noisy Intermediate-Scale Quantum Computers: A Real-System Study," *IEEE Micro*, vol. 40, no. 3, pp. 73–80, May 2020, ISSN: 1937-4143. DOI: 10.1109/MM.2020.2985683.

[290] R. I. Tobey, K. W. Lee, A. M. Hankin, *et al.*, "A high-power, low-noise, ultraviolet laser system for trapped-ion quantum computing," *Optics InfoBase Conference Papers*, vol. Part F181-, no. 0, 2020. DOI: 10.1364/CLEO_AT.2020.AF3K.3.

[291] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, and R. Duncan, "T|ket>: A retargetable compiler for NISQ devices," *Quantum Science and Technology*, vol. 6, no. 1, p. 014 003, Nov. 2020, ISSN: 2058-9565. DOI: 10.1088/2058-9565/ab8e92. (visited on 10/05/2021).

[292] M. D. S. ANIS, H. Abraham, AduOffei, R. Agarwal, and Et.al., *Qiskit: An Open-source Framework for Quantum Computing*, 2021. DOI: 10.5281/zenodo.2573505.

[293] L. Lao and D. E. Browne, "2QAN: A quantum compiler for 2-local qubit Hamiltonian simulation algorithms," *arXiv:2108.02099 [quant-ph]*, Nov. 2021. arXiv: 2108.02099 [quant-ph]. (visited on 05/03/2022).

[294] J. R. McClean, N. C. Rubin, K. J. Sung, *et al.*, "OpenFermion: The electronic structure package for quantum computers," *Quantum Sci. Technol.*, vol. 5, no. 3, p. 034 014, Jun. 2020, ISSN: 2058-9565. DOI: 10.1088/2058-9565/ab8ebc. (visited on 06/13/2022).

[295] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, and R. Duncan, "T$|$ket$\rangle$: A Retargetable Compiler for NISQ Devices," *Quantum Sci. Technol.*, vol. 6, no. 1, p. 014 003, Nov. 2020, ISSN: 2058-9565. DOI: 10/gn37qt. arXiv: 2003.10611. (visited on 01/14/2022).

[296] *DuckTigger/AnsatzFinder: Using a VAE inspired solution to find the best Ansatz*, https://github.com/DuckTigger/AnsatzFinder. (visited on 06/15/2022).

[297] Y. Yao, L. Rosasco, and A. Caponnetto, "On early stopping in gradient descent learning," *Constructive Approximation*, vol. 26, no. 2, pp. 289–315, 2007, ISSN: 01764276. DOI: 10.1007/s00365-006-0663-2. (visited on 08/30/2018).

[298] D. M. Lucas, B. C. Keitch, J. P. Home, *et al.*, *A long-lived memory qubit on a low-decoherence quantum bus*, Oct. 2007. DOI: 10.48550/arXiv.0710.4421. arXiv: 0710.4421 [quant-ph]. (visited on 06/30/2022).

[299] M. J. Biercuk, H. Uys, A. P. VanDevender, N. Shiga, W. M. Itano, and J. J. Bollinger, "Optimized dynamical decoupling in a model quantum memory," *Nature*, vol. 458, no. 7241, pp. 996–1000, Apr. 2009, ISSN: 1476-4687. DOI: 10.1038/nature07951. (visited on 06/30/2022).

[300] J. Emerson, R. Alicki, and K. Zyczkowski, "Scalable noise estimation with random unitary operators," *J. Opt. B: Quantum Semiclass. Opt.*, vol. 7, no. 10, S347–S352, Sep. 2005, ISSN: 1464-4266. DOI: 10.1088/1464-4266/7/10/021. (visited on 10/15/2021).

[301] J. Emerson, M. Silva, O. Moussa, *et al.*, "Symmetrized Characterization of Noisy Quantum Processes," *Science*, vol. 317, no. 5846, pp. 1893–1896, Sep. 2007. DOI: 10.1126/science.1145699. (visited on 10/15/2021).

[302] E. Knill, D. Leibfried, R. Reichle, *et al.*, "Randomized benchmarking of quantum gates," *Physical Review A*, vol. 77, no. 1, p. 012 307, Jan. 2008, ISSN: 1050-2947. DOI: 10.1103/PhysRevA.77.012307. (visited on 02/05/2021).

[303] E. Magesan, J. M. Gambetta, and J. Emerson, "Scalable and Robust Randomized Benchmarking of Quantum Processes," *Phys. Rev. Lett.*, vol. 106, no. 18, p. 180 504, May 2011. DOI: 10.1103/PhysRevLett.106.180504. (visited on 10/15/2021).

[304] J. J. Wallman and J. Emerson, "Noise tailoring for scalable quantum computation via randomized compiling," *Phys. Rev. A*, vol. 94, no. 5, p. 052 325, Nov. 2016, ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.94.052325. (visited on 10/15/2021).

[305] M. Ware, G. Ribeill, D. Ristè, C. A. Ryan, B. Johnson, and M. P. da Silva, "Experimental Pauli-frame randomization on a superconducting qubit," *Phys. Rev. A*, vol. 103, no. 4, p. 042 604, Apr. 2021. DOI: 10.1103/PhysRevA.103.042604. (visited on 06/30/2022).

[306] A. Hashim, R. K. Naik, A. Morvan, *et al.*, "Randomized Compiling for Scalable Quantum Computing on a Noisy Superconducting Quantum Processor," *Phys. Rev. X*, vol. 11, no. 4, p. 041 039, Nov. 2021. DOI: 10.1103/PhysRevX.11.041039. (visited on 06/30/2022).

[307] C. Chamberland, G. Zhu, T. J. Yoder, J. B. Hertzberg, and A. W. Cross, "Topological and Subsystem Codes on Low-Degree Graphs with Flag Qubits," *Physical Review X*, vol. 10, no. 1, 2020, ISSN: 21603308. DOI: 10.1103/PhysRevX.10.011022. (visited on 12/02/2020).

[308] M. Mirrahimi, Z. Leghtas, V. V. Albert, *et al.*, "Dynamically protected cat-qubits: A new paradigm for universal quantum computation," *New J. Phys.*, vol. 16, no. 4, p. 045 014, Apr. 2014, ISSN: 1367-2630. DOI: 10.1088/1367-2630/16/4/045014. (visited on 06/30/2022).

[309] P. Murali, L. Lao, M. Martonosi, and D. Browne, "Designing calibration and expressivity-efficient instruction sets for quantum computing," in *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*, Jun. 2021, pp. 846–859. DOI: 10.1109/ISCA52012.2021.00071. arXiv: 2106.15490 [quant-ph]. (visited on 06/30/2022).

[310] D. K. Tuckett, S. D. Bartlett, and S. T. Flammia, "Ultrahigh Error Threshold for Surface Codes with Biased Noise," *Phys. Rev. Lett.*, vol. 120, no. 5, p. 050 505, Jan. 2018, ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.120.050505. arXiv: 1708.08474 [quant-ph]. (visited on 06/30/2022).

[311] D. K. Tuckett, A. S. Darmawan, C. T. Chubb, S. Bravyi, S. D. Bartlett, and S. T. Flammia, "Tailoring surface codes for highly biased noise," *Phys. Rev. X*, vol. 9, no. 4, p. 041 031, Nov. 2019, ISSN: 2160-3308. DOI: 10.

1103/PhysRevX.9.041031. arXiv: 1812.08186 [quant-ph]. (visited on 06/30/2022).

[312] R. Blume-Kohout, J. K. Gamble, E. Nielsen, J. Mizrahi, J. D. Sterk, and P. Maunz, "Robust, self-consistent, closed-form tomography of quantum logic gates on a trapped ion qubit," *arXiv:1310.4492 [quant-ph]*, Oct. 2013. arXiv: 1310.4492 [quant-ph]. (visited on 10/19/2021).

[313] S. T. Merkel, J. M. Gambetta, J. A. Smolin, *et al.*, "Self-consistent quantum process tomography," *Phys. Rev. A*, vol. 87, no. 6, p. 062 119, Jun. 2013, ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.87.062119. (visited on 10/19/2021).

[314] D. Greenbaum, "Introduction to Quantum Gate Set Tomography," *arXiv:1509.02921 [quant-ph]*, Sep. 2015. arXiv: 1509.02921 [quant-ph]. (visited on 10/19/2021).

[315] R. Blume-Kohout, J. K. Gamble, E. Nielsen, *et al.*, "Demonstration of qubit operations below a rigorous fault tolerance threshold with gate set tomography," *Nat Commun*, vol. 8, no. 1, p. 14 485, Apr. 2017, ISSN: 2041-1723. DOI: 10.1038/ncomms14485. (visited on 10/19/2021).

[316] P. Aliferis, D. Gottesman, and J. Preskill, "Accuracy threshold for postselected quantum computation," *arXiv:quant-ph/0703264*, Sep. 2007. arXiv: quant-ph/0703264. (visited on 10/15/2021).

[317] D. Aharonov and M. Ben-Or, "Fault-Tolerant Quantum Computation with Constant Error Rate," *SIAM J. Comput.*, vol. 38, no. 4, pp. 1207–1282, Jan. 2008, ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539799359385. (visited on 10/15/2021).

[318] D. Leibfried, R. Blatt, C. Monroe, and D. Wineland, "Quantum dynamics of single trapped ions," *Reviews of Modern Physics*, vol. 75, no. 1, pp. 281–324, Mar. 2003, ISSN: 0034-6861. DOI: 10.1103/RevModPhys.75.281. (visited on 01/19/2021).

[319] K. Mølmer and A. Sørensen, "Multiparticle Entanglement of Hot Trapped Ions," *Physical Review Letters*, vol. 82, no. 9, pp. 1835–1838, Mar. 1999, ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.82.1835. (visited on 01/13/2021).

[320] J. R. Johansson, P. D. Nation, and F. Nori, "QuTiP 2: A Python framework for the dynamics of open quantum systems," *Computer Physics Communications*, vol. 184, no. 4, pp. 1234–1240, Nov. 2012, ISSN: 00104655. DOI: 10.1016/j.cpc.2012.11.019. (visited on 05/12/2021).

[321] T. P. Harty, "High-Fidelity Microwave-Driven Quantum Logic in Intermediate-Field 43 Ca +," Ph.D. dissertation, Balliol College, Oxford, 2013. (visited on 05/13/2021).

[322] C. J. Ballance, "High-Fidelity Quantum Logic in Ca +," Ph.D. dissertation, University of Oxford, 2014, ISBN: 978-3-319-68215-0. (visited on 01/21/2021).

[323] P. Panteleev and G. Kalachev, *Degenerate quantum LDPC codes with Good finite length performance*, 2019. (visited on 12/01/2020).

[324] G. Torlai and R. G. Melko, "Machine-Learning Quantum States in the NISQ Era," *Annual Review of Condensed Matter Physics*, vol. 11, no. 1, pp. 325–344, Mar. 2020, ISSN: 1947-5454. DOI: 10.1146/annurev-conmatphys-031119-050651. (visited on 10/05/2021).

[325] J. Biamonte and V. Bergholm, *Tensor Networks in a Nutshell*, Jul. 2017. arXiv: 1708.00006 [cond-mat, physics:gr-qc, physics:hep-th, physics:math-ph, physics:quant-ph]. (visited on 05/28/2022).

[326] P. Sobociński, *Graphical Linear Algebra*, https://graphicallinearalgebra.net/, Jun. 2015. (visited on 05/28/2022).

[327] D. Verdon, *Unitary 2-designs, variational quantum eigensolvers, and barren plateaus – Bristol Quantum Information Theory*, https://qitheory.blogs.bristol.ac.uk/2019/02/25/2-designs-variational-quantum-eigensolving-and-barren-plateaus/, Feb. 2019. (visited on 05/28/2022).

[328] J. Y. Stein, *Digital Signal Processing: A Computer Science Perspective*. USA: John Wiley & Sons, Inc., 2000, ISBN: 978-0-471-29546-4.

[329] C. Gardiner, "Stochastic Methods: A Handbook for the Natural and Social Sciences," *Citeulikeorg*, p. 447, 2009, ISSN: 0172-7389.